



K E R A J A A N M A L A Y S I A



KRISA

PANDUAN KEJURUTERAAN
SISTEM APLIKASI SEKTOR AWAM

Hak cipta terpelihara.

© Unit Pemodenan Tadbiran dan Perancangan Pengurusan Malaysia (MAMPU), 2019.

Hak cipta terpelihara, kecuali untuk tujuan pendidikan tanpa apa-apa kepentingan komersil. Tidak dibenarkan mengulang cetak mana-mana bahagian artikel, ilustrasi dan isi kandungan buku ini dalam apa jua bentuk dan dengan apa jua cara sama ada secara elektronik, rakaman atau cara lain sebelum mendapat kebenaran bertulis dari Ketua Pengarah MAMPU.



Diterbitkan oleh:

Trek Pembangunan Sistem
Bahagian Perundingan ICT
Unit Pemodenan Tadbiran dan Perancangan Pengurusan Malaysia
Jabatan Perdana Menteri
62502 Putrajaya Malaysia
Tel : 603 8000 8000
Fax : 603 8888 3712
Web : <http://www.mampu.gov.my>

KRISA

PANDUAN KEJURUTERAAN SISTEM APLIKASI SEKTOR AWAM



UNIT PEMODENAN TADBIRAN DAN
PERANCANGAN PENGURUSAN MALAYSIA
JABATAN PERDANA MENTERI
2019

KANDUNGAN

KANDUNGAN	IV
SENARAI JADUAL	VIII
SENARAI RAJAH	XII
APENDIKS	XVI
SENARAI DOKUMENTASI	XVII
PRAKATA	XVIII
AKRONIM	XXII
RINGKASAN EKSEKUTIF	1
KENAPA BUKU INI PENTING?	4
BAGAIMANA MENGGUNAKAN BUKU INI?	5
KEPERLUAN MINIMA DOKUMENTASI	6
1 PENGENALAN KEPADA METODOLOGI KEJURUTERAAN SISTEM APLIKASI	9
1.1 Pengenalan Kepada Kitar Hayat Kejuruteraan Pembangunan Sistem Aplikasi	9
1.2 Metodologi Pembangunan Sistem Aplikasi.....	10
1.3 Klasifikasi Metodologi Pembangunan Sistem Aplikasi	14
1.4 Ekosistem Pembangunan Sistem Sektor Awam.....	16
1.5 Metodologi Kejuruteraan Sistem Aplikasi Sektor Awam.....	16
1.6 Faktor Kejayaan dalam Pembangunan Sistem Aplikasi.....	20
1.6.1 Tadbir Urus Pengurusan Projek ICT.....	20
1.6.2 Penglibatan Pemegang Taruh.....	21
1.6.3 Pengurusan Kawalan Pindaan	22
1.6.4 Jaminan Kualiti Perisian (SQA)	23
1.6.5 Faktor Keselamatan ICT.....	27
1.6.6 Kepentingan Pengukuran Saiz Sistem	29
2 FASA PERMULAAN PROJEK	32
2.1 Gambaran Keseluruhan	32
2.2 Perancangan Pembangunan Sistem	33
2.2.1 Pengenalan.....	33
2.2.2 Penglibatan Pemegang Taruh.....	33
2.2.3 Faktor Kejayaan	33
2.2.4 Penyediaan Pelan Pembangunan Sistem [F1.1]	34
2.3 Kajian Keperluan Bisnes	47

2.3.1	Pengenalan.....	47
2.3.2	Penglibatan Pemegang Taruh.....	48
2.3.3	Faktor Kejayaan.....	48
2.3.4	Penentuan Keperluan Bisnes [F1.2].....	49
2.3.5	Pemodelan Fungsi Bisnes [F1.3].....	55
2.3.6	Pemodelan Proses Bisnes [F1.4].....	63
2.3.7	Penyediaan Spesifikasi Keperluan Bisnes [F1.5].....	74
3	FASA ANALISIS.....	79
3.1	Gambaran Keseluruhan.....	79
3.2	Pengenalan.....	79
3.3	Penglibatan Pemegang Taruh.....	80
3.4	Faktor Kejayaan.....	81
3.5	Pemodelan <i>Use Case</i> (Fungsian) [F2.1].....	82
3.6	Pemodelan Fungsi Sistem [F2.2].....	93
3.7	Pemodelan Keperluan Data [F2.3].....	101
3.8	Pemodelan Proses Sistem [F2.4].....	114
3.9	Penentuan Keperluan Bukan Fungsian [F2.5].....	121
3.10	Penyediaan Spesifikasi Keperluan Sistem [F2.6].....	127
4	FASA REKABENTUK.....	132
4.1	Gambaran Keseluruhan.....	132
4.2	Pengenalan.....	132
4.3	Penglibatan Pemegang Taruh.....	133
4.4	Faktor Kejayaan.....	133
4.5	Rekabentuk Arkitek [F3.1].....	134
4.6	Penentuan Teknologi [F3.2].....	153
4.7	Rekabentuk Pangkalan Data [F3.3].....	155
4.8	Rekabentuk Antaramuka Pengguna [F3.4].....	163
4.9	Rekabentuk Transaksi Sistem [F3.5].....	173
4.10	Penyediaan Spesifikasi Rekabentuk Sistem [F3.6].....	178
4.11	Migrasi Data.....	182
4.11.1	Penyediaan Pelan Migrasi Data [F3.7].....	183
4.11.2	Rekabentuk Migrasi Data [F3.8].....	187
4.12	Integrasi Sistem.....	197
4.12.1	Penyediaan Pelan Integrasi Sistem [F3.9].....	198
4.12.2	Rekabentuk Integrasi Sistem [F3.10].....	200

5	PEMBANGUNAN	212
5.1	Gambaran Keseluruhan	212
5.2	Pengenalan	212
5.3	Penglibatan Pemegang taruh	213
5.4	Faktor Kejayaan	214
5.5	Pembangunan Pangkalan Data [F4.1]	215
5.6	Pengaturcaraan Aplikasi [F4.2]	222
5.7	Pengujian Sistem [F4.3]	231
6	FASA PENGUJIAN PENERIMAAN	240
6.1	Gambaran Keseluruhan	240
6.2	Pengenalan	240
6.3	Penglibatan Pemegang Taruh	241
6.4	Faktor Kejayaan	242
6.4	Penyediaan Pelan Induk Pengujian [F5.1]	243
6.5	Penyediaan Dokumentasi Persediaan Ujian [F5.2]	249
6.6	Penyediaan Pelan Ujian (UAT & PAT) [F5.3]	263
6.7	Ujian Penerimaan Pengguna (UAT) [F5.4]	270
6.8	Ujian Penerimaan Provisional (PAT) [F5.5]	273
6.9	Laporan Ujian Penerimaan (UAT & PAT) [F5.6]	276
7	FASA PELAKSANAAN	279
7.1	Gambaran Keseluruhan	279
7.2	Pengenalan	279
7.3	Penglibatan Pemegang taruh	280
7.4	Faktor Kejayaan	281
7.5	Pelaksanaan Migrasi Data [F6.1]	283
7.6	Ujian Penerimaan Akhir [F6.2]	287
7.7	Penyediaan Manual Pengguna [F6.3]	291
7.8	Serahan Sistem Aplikasi [F6.4]	295
8	PENGIRAAN SAIZ SISTEM APLIKASI	298
8.1	Pengenalan kepada <i>Function Points</i>	298
8.2	Pengiraan Saiz Fungsian Sistem Aplikasi	298
8.2.1	Definisi	298
8.2.1.1	Komponen Fungsian	299
8.2.1.2	Jadual Matriks Kompleksiti dan Jadual Penterjemahan	301
8.2.1.3	Empat Belas Ciri-ciri Am Sistem (GSC)	303

8.2.2	Pengiraan <i>Value Adjustment Factor</i> (VAF).....	305
8.2.3	Pengiraan <i>Unadjusted Function Points</i> (UFP)	305
8.2.4	Pengiraan <i>Adjusted Function Points</i> (AFP)	305
8.2.5	Pengiraan Anggaran <i>Effort</i> (<i>man hours</i>) dan Kos Pembangunan Sistem	307
8.3	Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Permulaan Projek	309
8.4	Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Analisis.....	315
8.5	Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Pelaksanaan	323



SENARAI JADUAL

Jadual 1: Rangka Kerja Metodologi Kejuruteraan Pembangunan Sistem Aplikasi.....	2
Jadual 2 : Keperluan Minima Dokumentasi Projek Pembangunan Sistem.....	6
Jadual 3 : Atribut Kualiti Perisian	23
Jadual 4 : Jenis-Jenis Pengujian IV&V	26
Jadual 5 : Contoh Jadual Penerangan Keahlian Pasukan Pembangunan	37
Jadual 6 : Skala Impak dan Tahap Risiko	39
Jadual 7 : Contoh Anggaran Jumlah Projek	44
Jadual 8 : Teknik Pengumpulan Maklumat.....	50
Jadual 9 : Notasi Fungsi Bisnes.....	55
Jadual 10 : Keterangan Label Fungsi Bisnes	57
Jadual 11 : Contoh Keterangan Fungsi Bisnes.....	60
Jadual 12 : Notasi Proses Bisnes	63
Jadual 13 : Keterangan Label Definisi Aktiviti Fungsi Bisnes	71
Jadual 14 : Contoh Senarai Pemegang Taruh.....	75
Jadual 15 : Contoh Senarai Pengguna	76
Jadual 16 : Notasi Rajah <i>Use Case</i>	82
Jadual 17 : Contoh Jadual Terjemahan Peranan kepada Aktor.....	85
Jadual 18 : Keterangan Label <i>Use Case</i>	86
Jadual 19 : Contoh Keterangan <i>Use Case</i>	88
Jadual 20 : Notasi Fungsi Bisnes	93
Jadual 21 : Contoh Pepadanan Aktor Dengan Fungsi Sistem Bagi Submodul Tempahan Bilik Mesyuarat, Sistem Tempahan Bilik Mesyuarat (eTempah)	98
Jadual 22 : Notasi Rajah Hubungan Data	101
Jadual 23 : Contoh Entiti, Atribut, Pola Capaian Dan Peraturan Bisnes	104
Jadual 24 : Notasi Rajah Aliran Data.....	114
Jadual 25 : Keterangan Medan-Medan Definisi Aliran Data	120
Jadual 26 : Keterangan Aspek-aspek Klasifikasi Keperluan Fungsian	123
Jadual 27 : Templat Ciri-ciri Sistem Aplikasi.....	123
Jadual 28 : Keterangan Medan-medan Templat Ciri-ciri Sistem Aplikasi.....	124
Jadual 29 : Senarai Definisi Ciri-ciri Kualiti Sistem Aplikasi.....	125

Jadual 30 : Contoh Senarai Aktor Sistem	128
Jadual 31 : Perbandingan Arkitektur Monolitik dan Mikroservis.....	138
Jadual 32 : Perbandingan Arkitektur Aplikasi Tradisional Vs Aplikasi Moden.....	144
Jadual 33 : Perbandingan Arkitektur Shared-Disk Vs Shared-Nothing	148
Jadual 34 : Nilai Skor Skema Penilaian Teknikal	154
Jadual 35 : Matriks Alternatif Bagi Penentuan Bahasa Pengaturcaraan.....	154
Jadual 36 : Notasi Rekabentuk Keperluan Data	155
Jadual 37 : Keterangan Medan-Medan Templat Pemetaan Data	170
Jadual 38 : Contoh Pengisian Templat Pemetaan Data Bagi Menu Luluskan Tempahan Bilik Mesyuarat, Sistem Tempahan Bilik Mesyuarat (eTempah).....	171
Jadual 39 : Keterangan Medan-medan Templat Senario <i>Use Case</i>	175
Jadual 40 : Pengisian Templat Senario <i>Use Case</i>	176
Jadual 41 : Analisa Keperluan Migrasi Data.....	183
Jadual 42 : Isi Kandungan Pelan Migrasi Data	184
Jadual 43 : Pemetaan Medan Data antara Sistem Legasi dan.....	189
Jadual 44 : Contoh Pemetaan Kod	189
Jadual 45 : Contoh jadual yang menyimpan maklumat kod	190
Jadual 46: jadual sumber asal ref_skimJwtn	190
Jadual 47: Jadual destinasi baru ref_skimGredJwtn	191
Jadual 48: Jadual sumber asal 'pekerja'	191
Jadual 49: Contoh maklumat perubahan gred perjawatan.....	192
Jadual 50: Contoh jadual yang menunjukkan rekod idPekerja '0000001'	193
Jadual 51 : Contoh pemetaan rekod yang menunjukkan maklumat perubahan gred	193
Jadual 52 : Contoh jadual destinasi baru Pesonel	194
Jadual 53 : Contoh pengisian peraturan pemetaan data	194
Jadual 54 : Isi Kandungan Spesifikasi Migrasi Data.....	195
Jadual 55 : Isi Kandungan Pelan Integrasi Sistem	198
Jadual 56 : Isi Kandungan Pelan Integrasi Sistem	199
Jadual 57 : Keterangan Templat Maklumat Servis Integrasi	201
Jadual 58 : Contoh Maklumat Aktiviti yang Memerlukan Integrasi	201
Jadual 59 : Keterangan Templat Format Pertukaran Data	202
Jadual 60 : Contoh Penerangan Servis Integrasi dan Data yang Terlibat.....	203

Jadual 61 : Keterangan Templat Pemetaan Struktur Data	204
Jadual 62 : Contoh Pemetaan Struktur Data	204
Jadual 63 : Pemadanan Istilah antara reka bentuk logikal dan reka bentuk fizikal pangkalan data	218
Jadual 64 : Contoh Konvensyen Penamaan	227
Jadual 65 : <i>Entry Criteria</i> dan <i>Exit Criteria</i> bagi Ujian Unit/Komponen	232
Jadual 66 : <i>Entry Criteria</i> dan <i>Exit Criteria</i> bagi Ujian Sub-Sistem/Modul	234
Jadual 67 : <i>Entry Criteria</i> dan <i>Exit Criteria</i> bag Ujian Integrasi	235
Jadual 68 : <i>Entry Criteria</i> dan <i>Exit Criteria</i> bag Ujian Sistem.....	236
Jadual 69 : Format Laporan Ujian Sistem	237
Jadual 70 : Komunikasi Pengujian	244
Jadual 71 : Contoh Peringkat dan Jenis-jenis Ujian	245
Jadual 72 : Contoh Tahap Severity Hasil Ujian.....	247
Jadual 73 : Contoh Jadual Perancangan Pengujian.....	247
Jadual 74 : Contoh Peranan dan Tanggungjawab dalam Pengujian.....	248
Jadual 75 : Isi Kandungan Templat Senario Ujian	250
Jadual 76 : Contoh Pengisian Templat Senario Ujian (Mendaftar Akaun Baharu)	251
Jadual 77 : Contoh Pengisian Templat Senario Ujian (Mengemaskini Profil)	251
Jadual 78 : Isi Kandungan Templat Kes Ujian	253
Jadual 79 : Contoh Pengisian Templat Kes Ujian (Log Masuk Positif)	254
Jadual 80 : Contoh Pengisian Templat Kes Ujian (Log Masuk Negatif).....	255
Jadual 81 : Contoh Pengisian Templat Kes Ujian (Tempahan Bilik Mesyuarat)	256
Jadual 82 : Isi Kandungan Templat Prosedur/ Skrip Ujian	258
Jadual 83 : Contoh Pengisian Templat Prosedur/ Skrip Ujian	259
Jadual 84 : Kategori Data Ujian	260
Jadual 85 : Isi Kandungan Templat <i>Traceability Matrix</i>	261
Jadual 86 : Contoh Pengisian Templat <i>Traceability Matrix</i>	261
Jadual 87 : Contoh Strategi Ujian Bagi Ujian Bebanan.....	265
Jadual 88 : Contoh Strategi Ujian Bagi Ujian Prestasi	265
Jadual 89 : Contoh Strategi Ujian Bagi Ujian Tekanan	266
Jadual 90 : Contoh Strategi Ujian Bagi Pemantauan System Resources	266
Jadual 91 : Contoh <i>Entry Criteria</i> Dan <i>Exit Criteria</i> Untuk Pelan Ujian UAT/PAT	268

Jadual 92 : Contoh Metrik Pengukuran.....	268
Jadual 93 : Contoh Pengiraan Sampel Data Bagi Ujian Migrasi Data	285
Jadual 94 : Isi Kandungan Laporan Migrasi Data	286
Jadual 95 : Syarat Masuk dan Keluar Ujian Penerimaan akhir	287
Jadual 96 : Aktor dan Aktiviti Ujian Penerimaan akhir	288
Jadual 97 : Isi Kandungan Laporan Penamatan Ujian.....	288
Jadual 98 : Matriks Kompleksiti EI	301
Jadual 99 : Matriks Kompleksiti EO dan EQ.....	302
Jadual 100 : Penterjemahan Saiz EI dan EQ	302
Jadual 101 : Penterjemahan Saiz EO.....	302
Jadual 102 : Matriks Kompleksiti ILF dan EIF.....	302
Jadual 103 : Penterjemahan Saiz ILF.....	302
Jadual 104 : Penterjemahan Saiz EIF	303
Jadual 105 : 14 Ciri-ciri Am Sistem (GSC)	303
Jadual 106 : Formula Pengiraan UFP	305
Jadual 107 : Formula Pengiraan AFP.....	306
Jadual 108 : Penentuan Tahap Kompleksiti Fungsi Data.....	311
Jadual 109 : Penentuan Tahap Kompleksiti Fungsi Transaksi	313
Jadual 110 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data	315
Jadual 111 : Penentuan Tahap Kompleksiti Fungsi Data.....	319
Jadual 112 : Penentuan Tahap Kompleksiti Fungsi Transaksi	321
Jadual 113 : Penentuan Tahap Kompleksiti Fungsi Data.....	325
Jadual 114 : Penentuan Tahap Kompleksiti Fungsi Transaksi	327

SENARAI RAJAH

Rajah 1: Rangka Kerja Panduan Kejuruteraan Sistem Aplikasi Sektor Awam	1
Rajah 2: Kitar Hayat Pembangunan Sistem Aplikasi	9
Rajah 3 : Metodologi <i>Waterfall</i>	10
Rajah 4 : Metodologi <i>Spiral</i>	11
Rajah 5 : Metodologi <i>RAD</i>	11
Rajah 6 : Metodologi <i>Prototype</i>	12
Rajah 7 : Metodologi <i>Incremental Development</i>	12
Rajah 8 : Metodologi <i>RUP</i>	13
Rajah 9 : Model Klasifikasi Metodologi Pembangunan Sistem Aplikasi	15
Rajah 10 : Ekosistem Kejuruteraan Pembangunan Sistem Aplikasi Sektor Awam	16
Rajah 11 : Metodologi Kejuruteraan Pembangunan Sistem Aplikasi Sektor Awam	17
Rajah 12 : Proses Verifikasi dan Validasi.....	24
Rajah 13 : Jenis-Jenis Pengujian V&V	25
Rajah 14 : Gambaran Keseluruhan Fasa I – Permulaan Projek	32
Rajah 15 : Contoh Struktur Pasukan Pembangunan Sistem	37
Rajah 16 : Proses Pemantauan Dan Kawalan Projek	40
Rajah 17 : Contoh Perincian Kerja (WBS)	43
Rajah 18 : Contoh Jadual Perancangan Projek	45
Rajah 19 : Model Bisnes Organisasi	47
Rajah 20 : Contoh Arkitektur Bisnes.....	52
Rajah 21 : Contoh Arkitektur Maklumat	54
Rajah 22 : Contoh Rajah Hierarki Fungsian Bisnes Bagi Sistem Tempahan Bilik Mesyuarat (eTempah).....	58
Rajah 23 : Penyediaan Tajuk PFD dan <i>Swimlane</i>	66
Rajah 24 : Contoh Penyediaan Notasi <i>Activity</i> (Aktiviti Bisnes)	67
Rajah 25 : Contoh Penyediaan Notasi <i>Send Signal Action</i> dan <i>Accept Event Action</i>	67
Rajah 26 : Penggunaan Notasi <i>Decision</i>	68
Rajah 27 : Penggunaan Notasi <i>Activity Final Node</i>	69
Rajah 28 : Penggunaan Notasi <i>Final Flow Node</i>	69
Rajah 29 : Penggunaan Notasi <i>Control Flow/Edge</i> dalam Proses Tempah Bilik Mesyuarat .70	

Rajah 30 : Penggunaan Notasi <i>Note</i>	70
Rajah 31 : Gambaran Keseluruhan Fasa II - Analisis	79
Rajah 32 : Contoh Penggunaan <i>Inherit Behavior</i> (Aktor)	85
Rajah 33 : Rajah Aliran Kerja PFD-BM-MT-TBM Tempah Bilik Mesyuarat	87
Rajah 34 : Contoh Rajah <i>Use Case</i> - Hubungan Aktor Dengan <i>Use Case</i>	88
Rajah 35 : Contoh Hubungan <i>Use Case</i> (Pengkhususan)	90
Rajah 36 : Contoh Hubungan <i>Use Case</i> (Include)	90
Rajah 37 : Contoh Hubungan <i>Use Case</i> (<i>Extend</i>).....	90
Rajah 38 : Contoh Rajah <i>Use Case</i> Modul Pengurusan Tempahan Bilik Mesyuarat	91
Rajah 39 : Contoh Rajah Hierarki Fungsian Sistem Bagi Modul Pengurusan Tempahan Bilik Mesyuarat Di Bawah Sistem Tempahan Bilik Mesyuarat (eTempah).....	97
Rajah 40 : Contoh Entiti	105
Rajah 41 : Contoh Entiti Dan Senarai Atribut	106
Rajah 42 : Contoh Nama Hubungan Entiti	106
Rajah 43 : Contoh Rajah Hubungan Entiti (ERD)	107
Rajah 44 : Contoh <i>Intersection Entity</i>	108
Rajah 45 : Contoh Hubungan Entiti (ERD) – <i>Intersection Entity</i>	109
Rajah 46 : Contoh Hubungan Entiti Lain Kepada Entiti <i>Super-type</i>	110
Rajah 47 : Contoh Hubungan Exclusive Arc.....	111
Rajah 48 : Contoh Carta Organisasi.....	111
Rajah 49 : Contoh Hubungan Rekursif	112
Rajah 50 : Contoh Pemetaan Hierarki Fungsi Sistem kepada Notasi Fungsi	117
Rajah 51 : Contoh Lakaran Fungsi dan Entiti Luaran	117
Rajah 52 : Contoh Rajah Konteks DFD	118
Rajah 53 : Contoh Pemetaan Hierarki Fungsi Sistem kepada Notasi Fungsi DFD	118
Rajah 54 : Contoh Data Flow Diagram (DFD)	119
Rajah 55 : Klasifikasi Keperluan Bukan Fungsian.....	122
Rajah 56 : Gambaran Keseluruhan Fasa III - Rekabentuk.....	132
Rajah 57 : Arkitektur Monolitik Bagi Perisian Sistem Aplikasi.....	137
Rajah 58 : Arkitektur Mikroservis Bagi Perisian Sistem Aplikasi	138
Rajah 59 : Arkitektur Aplikasi Tradisional.....	142
Rajah 60 : Arkitektur Aplikasi Moden.....	143

Rajah 61 : Arkitektur <i>Shared-Disk</i> Bagi Aplikasi Web eTempah	147
Rajah 62 : Arkitektur <i>Shared-Nothing</i> Bagi Aplikasi Web eTempah	148
Rajah 63 : ERD - Entiti Tunggal	157
Rajah 64 : ERD – Penggunaan Entiti Supertype.....	157
Rajah 65 : Kekunci Primer daripada <i>composite key</i>	159
Rajah 66 : ERD – Penggunaan Entiti Supertype.....	160
Rajah 67 : Contoh Model Maklumat Logikal Sistem Tempahan Bilik Mesyuarat (eTempah) 162	
Rajah 68 : Rangka Kerja Ciri-ciri <i>User Experience</i>	164
Rajah 69 : Perbezaan Bagi Penggunaan <i>Line-Height</i> Yang Berkesan Dalam Antaramuka Pengguna.....	168
Rajah 70 : Contoh Penggunaan Ruang Negatif Yang Berkesan	169
Rajah 71 : Contoh Rekabentuk Antaramuka Pengguna Bagi Menu Luluskan Tempahan Bilik Mesyuarat, Sistem Tempahan Bilik Mesyuarat (eTempah).....	169
Rajah 72 : Contoh Antaramuka Pengguna Tempahan Bilik Mesyuarat	174
Rajah 73 : Contoh Pemetaan Jadual antara Pangkalan Data Sistem Legasi dan Sistem Baharu	188
Rajah 74 : Contoh pemetaan data	193
Rajah 75 : Proses Khusus.....	205
Rajah 76 : Proses Berbilang Langkah	206
Rajah 77 : Contoh Senario Proses Integrasi	207
Rajah 78 : Contoh Proses Integrasi Servis Aduan Kerosakan.....	208
Rajah 79 : Arkitektur SOAP	209
Rajah 80 : Arkitektur <i>RESTful</i>	210
Rajah 81 : Arkitektur <i>Messaging</i>	210
Rajah 82 : Arkitektur <i>Messaging</i> untuk Pengintegrasian Berbilang <i>Server</i>	210
Rajah 83 : Gambaran Keseluruhan Fasa IV - Pembangunan.....	212
Rajah 84 : Struktur Storan Secara Logikal Dalam Pangkalan Data.....	216
Rajah 85 : Peringkat Pengujian Sistem	231
Rajah 86 : Ujian Komponen - Stubs dan Drivers.....	233
Rajah 87 : Gambaran Keseluruhan Fasa V – Pengujian Penerimaan	240
Rajah 88 : Contoh Struktur Pasukan Pengujian.....	248
Rajah 89 : Gambaran Keseluruhan Fasa VI – Pelaksanaan	279

Rajah 90 : Komponen Fungsi EI	299
Rajah 91 : Komponen Fungsi EO	299
Rajah 92 : Komponen Fungsi EQ	300
Rajah 93 : Gambaran Komponen-Komponen Fungsi Di Dalam Sistem.....	301



APENDIKS

Apendiks 1 : Borang Permohonan Pembangunan Sistem.....	329
Apendiks 2 : Templat Definisi Aktiviti Fungsi Bisnes	330
Apendiks 3 : Templat Definisi Kamus Data (Entity & Attribute).....	331
Apendiks 4 : Templat Skema Logikal Pangkalan Data	332
Apendiks 5 : Templat Definisi Aliran Data	333
Apendiks 6 : Templat Pemetaan Data (Antaramuka)	334
Apendiks 7 : Templat Senario <i>Use Case</i>	335
Apendiks 8 : Templat Rekabentuk Migrasi Data.....	336
Apendiks 9 : Templat Rekabentuk Integrasi.....	337
Apendiks 10 : Templat Persediaan Ujian	340
Apendiks 11 : Contoh Final Acceptance Certificate.....	344



SENARAI DOKUMENTASI

D01 DOKUMEN PELAN PEMBANGUNAN SISTEM (PPS).....	345
D02 DOKUMEN SPESIFIKASI KEPERLUAN BISNES (BRS)	353
D03 DOKUMEN SPESIFIKASI KEPERLUAN SISTEM (SRS).....	360
D04 DOKUMEN SPESIFIKASI REKABENTUK SISTEM (SDS)	368
D05 DOKUMEN PELAN MIGRASI DATA.....	375
D06 DOKUMEN SPESIFIKASI MIGRASI DATA.....	381
D07 DOKUMEN PELAN INTEGRASI SISTEM.....	388
D08 DOKUMEN SPESIFIKASI INTEGRASI DATA.....	393
D09 DOKUMENTASI PANGKALAN DATA.....	399
D10 DOKUMENTASI KOD SUMBER	404
D11 DOKUMEN LAPORAN UJIAN SISTEM	411
D12 DOKUMEN PELAN INDUK PENGUJIAN.....	414
D13 DOKUMEN PELAN UJIAN PENERIMAAN (UAT-PAT).....	423
D14 LAPORAN UJIAN PENERIMAAN PENGGUNA (UAT/PAT).....	429
D15 DOKUMEN LAPORAN MIGRASI DATA	433
D16 DOKUMEN LAPORAN PENAMATAN UJIAN.....	437
D17 DOKUMEN MANUAL PENGGUNA SISTEM	441
D18 DOKUMEN LAPORAN SERAHAN SISTEM	446



PRAKATA

KETUA PENGARAH MAMPU

Assalamualikum Warahmatullahi Wabarakatuh.

Terlebih dahulu izinkan saya memanjatkan rasa kesyukuran di atas limpah kurnia dan keizinan Allah SWT, Buku Panduan Kejuruteraan Sistem Aplikasi Sektor Awam (KRISA) ini dapat diterbitkan dengan jayanya. Syabas dan tahniah dirakamkan kepada barisan pengarang yang terlibat dalam penghasilan buku ini.

Dunia masa kini berhadapan dengan kemunculan pelbagai teknologi bersifat disruptif yang menuntut kerajaan lebih responsif untuk kekal relevan dan berdaya saing di peringkat global.

Mendepani cabaran ini, penjawat awam perlu bergerak pantas dan efisien dalam mencetus inovasi ke atas model bisnes, tadbir urus dan proses penyampaian perkhidmatan yang dinamik dan progresif menelusuri naratif baharu Kerajaan Digital.

Dalam mengarusperdanakan agenda Kerajaan Digital ke arah peningkatan produktiviti dan kecekapan sistem penyampaian perkhidmatan yang berpaksikan rakyat, pihak Kerajaan telah memberi penekanan serius terhadap inisiatif memperkasakan kualiti Teknologi Maklumat dan Komunikasi (ICT) yang menjadi tunggak terpenting dalam merealisasikan Pelan Pendigitalan Penyampaian Perkhidmatan Kerajaan.

Bagi menyahut cabaran paradigma semasa dalam melestarikan kejayaan dan keberkesanan pembangunan sistem aplikasi kerajaan, MAMPU telah menghasilkan buku panduan KRISA bagi memperkukuh keupayaan intelektual dan kepakaran penjawat awam supaya lebih bersestematik dalam aspek pembangunan sistem aplikasi. Kandungan buku ini disusun berasaskan kepada metodologi dan standard antarabangsa dalam pembangunan sistem, serta diperkasakan dengan garapan pengalaman pakar-pakar ICT Kerajaan bersesuaian dengan domain bisnes sektor awam.

Adalah menjadi harapan saya agar buku ini dapat dimanfaatkan sepenuhnya sebagai repositori ilmu dan panduan kepada agensi dan penjawat awam dalam mengemudi pelaksanaan projek pembangunan sistem aplikasi Kerajaan. Penerbitan buku ini amat tepat pada masanya di saat perkhidmatan awam giat merencanakan pembangunan Kerajaan Digital.

Dato' Dr. Mazlan Yusoff
Ketua Pengarah MAMPU





PRAKATA

KETUA PEGAWAI MAKLUMAT KERAJAAN (GCIO)

Assalamualikum Warahmatullahi Wabarakatuh

Buku Panduan Kejuruteraan Sistem Aplikasi Sektor Awam (KRISA) ditampilkan sebagai wahana ilmu dan rujukan yang praktikal dalam memacu dan melestarikan kecemerlangan penjawat awam khususnya profesional ICT dalam aspek pembangunan sistem aplikasi.

Kebanyakan inisiatif projek-projek ICT Kerajaan melibatkan komponen pembangunan sistem aplikasi, di mana kejayaannya adalah sangat bergantung kepada kecekapan dan kepakaran pasukan dalam melaksanakan proses pembangunan sistem.

Walaupun terdapat banyak buku akademik yang menerangkan pelbagai metodologi pembangunan sistem sebagai rujukan, namun masih terdapat projek ICT Kerajaan menghadapi kelewatan dalam pelaksanaan dan tidak mencapai kualiti seperti yang diharapkan oleh *stakeholder*.

Oleh yang demikian, adalah penting sebuah buku berbentuk praktikal yang komprehensif dalam bidang pembangunan sistem diterbitkan sejajar dengan “Pekeliling perkhidmatan Bilangan 7 Tahun 2016 - Kemajuan Kerjaya Laluan Pakar Bidang Khusus (*Subject Matter Expert* - SME) Bagi Perkhidmatan Awam Persekutuan” untuk mengiktiraf pegawai yang memiliki keupayaan intelektual tinggi dan pengalaman luas dalam bidang kepakaran tertentu di samping menggalakkan pelaksanaan amalan-amalan terbaik dalam pembangunan sistem.

Justeru, MAMPU telah menerbitkan buku KRISA sebagai panduan pembangunan sistem aplikasi berasaskan kepada kitar hayat pembangunan sistem (SDLC) yang meliputi 6 fasa utama bermula dari fasa permulaan projek sehingga fasa pelaksanaan, merangkumi metode, templat dan format dokumentasi. Bukan sekadar penulisan akademik semata-mata, buku KRISA menawarkan panduan praktikal dan *learning experience* yang bagus kepada semua peringkat professional ICT dikalangan penjawat awam. Buku ini juga boleh digunakan oleh

perunding, pakar, instruktur dan fasilitator dalam memberi latihan pembangunan sistem aplikasi kepada agensi kerajaan dan ianya dijadikan sebagai kandungan dalam pelaksanaan Hala Tuju Pembangunan Keupayaan Perkhidmatan ICT Kerajaan.

Akhir kata, sekalung tahniah dan penghargaan diucapkan kepada barisan penasihat dan pengarang dari kalangan pakar-pakar ICT Sektor Awam di atas usaha dalam menghasilkan wacana ilmu yang amat berharga ini. Semoga semangat usahasama dan inovatif ini dapat diteruskan ke arah kecemerlangan penjawat awam.

YBrs. Dr. Suhazimah Dzazali

Ketua Pegawai Maklumat Kerajaan (GCIO)



AKRONIM

BF	Bukan Fungsian
BRS	<i>Business Requirement Specification</i>
CRUD	<i>Create Retrieve Update Delete</i>
DET	<i>Data Element Type</i>
DFD	<i>Data Flow Diagram</i>
EA	<i>Enterprise Architecture</i>
EI	<i>External Inputs</i>
EIF	<i>External Interface File</i>
EO	<i>External Outputs</i>
EQ	<i>External Inquiry</i>
ERD	<i>Entity Relationship Diagram</i>
FAT	<i>Final Acceptance Test</i>
FP	<i>Function Point</i>
FPA	<i>Function Point Analysis</i>
FTR	<i>File Type Referenced</i>
ICT	<i>Information Communication Technology</i>
ICO/IEC	<i>International Organization for Standardization/ International Electrotechnical Commission</i>
IFPUG	<i>International Function Point Users Group</i>
ILF	<i>Internal Logical File</i>
IPR	<i>Intellectual Property Rights</i>
JAD	<i>Joint Application Design</i>
PAT	<i>Provisional Acceptance Test</i>
PFD	<i>Process Flow Diagram</i>
PMO	<i>Project Management Office</i>
PPS	Pelan Pembangunan Sistem
PPrISA	Metodologi Pengurusan Projek ICT Sektor Awam
RAD	<i>Rapid Application Development</i>
RET	<i>Record Element Type</i>
RUP	<i>Rational Unified Process</i>
SDLC	<i>System Development Life Cycle</i>
SDS	<i>System Design Specification</i>
SME	<i>Subject Matter Expert</i>
SRS	<i>System Requirement Specification</i>

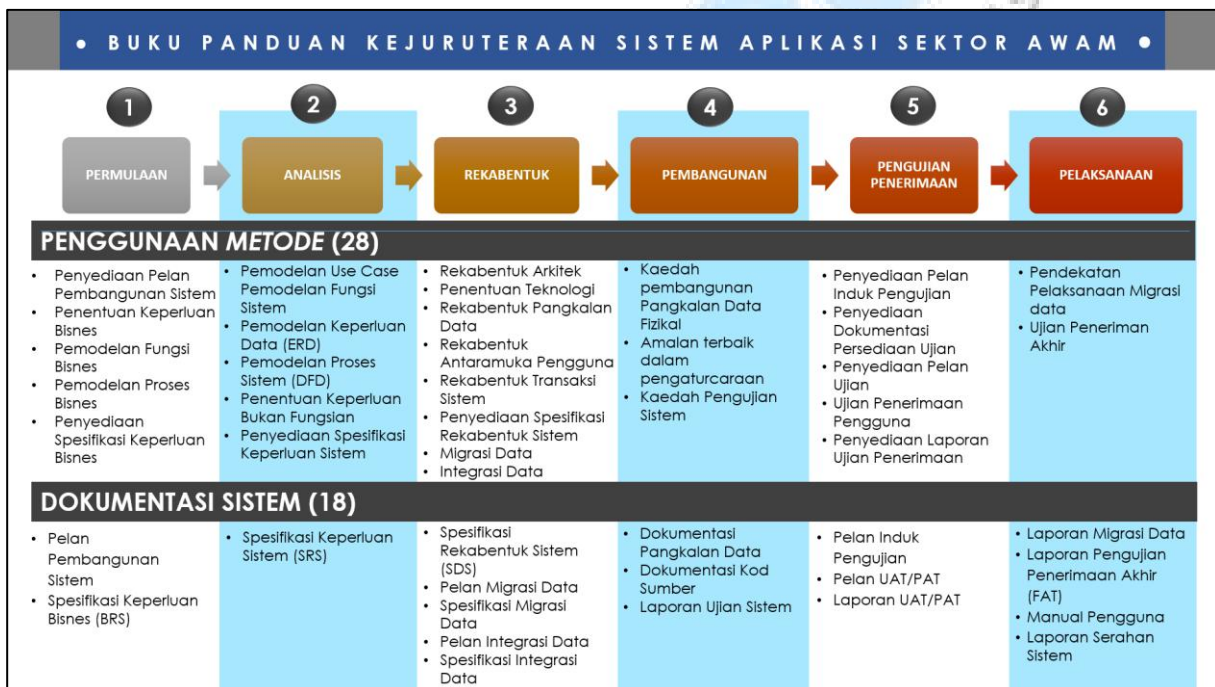
SQA	<i>Software Quality Assurance</i>
SSO	<i>Single Sign-On</i>
UAT	<i>User Acceptance Test</i>
UC	<i>Use Case</i>
UFP	<i>Unadjusted Function Point</i>
UID	<i>Unique ID</i>
UX	<i>User Experiences</i>
V&V	<i>Verifikasi dan Validasi</i>
VAF	<i>Value Adjustment Factor</i>
XP	<i>Extreme Programming</i>



RINGKASAN EKSEKUTIF

Buku ini ditulis dengan mengambil kira amalan terbaik teknik dan kaedah yang digunakan dalam melaksanakan aktiviti-aktiviti pembangunan sistem aplikasi. Buku ini menyediakan panduan dalam proses pembangunan sistem aplikasi dengan memberi fokus kepada bagaimana melaksanakan aktiviti-aktiviti tersebut dan apakah serahan yang perlu dihasilkan pada setiap fasa kitar hayat pembangunan sistem. Proses pembangunan sistem aplikasi yang diperkenalkan dalam buku ini dikenali sebagai **Buku Panduan Kejuruteraan Sistem Aplikasi Sektor Awam (KRISA)**.

Metodologi KRISA meliputi 6 fasa utama yang perlu difahami, diguna pakai dan dilaksanakan oleh agensi. Setiap fasa menerangkan aktiviti, teknik dan pendekatan, serta dokumentasi serahan. **Rajah 1** Rangka Kerja Panduan Kejuruteraan Sistem Aplikasi Sektor Awam dan **Jadual 1** menyenaraikan fasa, aktiviti dan dokumen serahan.



Rajah 1: Rangka Kerja Panduan Kejuruteraan Sistem Aplikasi Sektor Awam

Beberapa pertimbangan utama perlu diambil kira bagi menjamin kejayaan pelaksanaan projek pembangunan sistem aplikasi, di samping memastikan sistem aplikasi yang dibangunkan adalah berkualiti dan memenuhi keperluan pengguna. Pertimbangan- pertimbangan tersebut adalah:

- Tadbir Urus Pengurusan Projek ICT**
- Penglibatan Pemegang Taruh**
- Pengurusan Kawalan Pindaan**
- Jaminan Kualiti Perisian (SQA)**
- Faktor Keselamatan ICT**
- Kepentingan Pengukuran Saiz Sistem**

Jadual 1: Rangka Kerja Metodologi Kejuruteraan Pembangunan Sistem Aplikasi

Fasa		No.	Aktiviti	Dokumen Serahan
F1	Permulaan dan Keperluan Projek	[F1.1]	Penyediaan Pelan Pembangunan Sistem	D01 Pelan Pembangunan Sistem
		[F1.2]	Penentuan Keperluan Bisnes	
		[F1.3]	Pemodelan Fungsi Bisnes	
		[F1.4]	Pemodelan Proses Bisnes	
		[F1.5]	Penyediaan Spesifikasi Keperluan Bisnes	D02 Spesifikasi Keperluan Bisnes (BRS)
F2	Analisis	[F2.1]	Pemodelan Use Case (Fungsian)	D03 Spesifikasi Keperluan Sistem (SRS)
		[F2.2]	Pemodelan Fungsi Sistem (DFD)	
		[F2.3]	Pemodelan Keperluan Data (ERD)	
		[F2.4]	Pemodelan Proses Sistem	
		[F2.5]	Penentuan Keperluan Bukan Fungsian	
		[F2.6]	Penyediaan Spesifikasi Keperluan Sistem	
F3	Rekabentuk	[F3.1]	Rekabentuk Arkitek	D04 Spesifikasi Rekabentuk Sistem (SDS)
		[F3.2]	Penentuan Teknologi	
		[F3.3]	Rekabentuk Pangkalan Data	
		[F3.4]	Rekabentuk Antaramuka Pengguna	
		[F3.5]	Rekabentuk Transaksi Sistem	
		[F3.6]	Penyediaan Spesifikasi Rekabentuk Sistem	
		[F3.7]	<u>Migrasi Data</u> Penyediaan Pelan Migrasi Data	D05 Pelan Migrasi Data
		[F3.8]	Rekabentuk dan Penyediaan	D06 Spesifikasi Migrasi Data

Fasa		No.	Aktiviti	Dokumen Serahan
			Spesifikasi Migrasi Data	
		[F3.9]	Integrasi Data Penyediaan Pelan Integrasi Data	D07 Pelan Integrasi Data
		[F3.10]	Rekabentuk dan Penyediaan Spesifikasi Integrasi Data	D08 Spesifikasi Integrasi Data
F4	Pembangunan	[F4.1]	Pembangunan Pangkalan Data	D09 Dokumentasi Pangkalan Data
		[F4.2]	Pengaturcaraan Aplikasi	D10 Dokumentasi Kod Sumber
		[F4.3]	Pengujian Sistem	D11 Laporan Ujian Sistem
F5	Pengujian	[F5.1]	Penyediaan Pelan Induk Pengujian	D12 Pelan Induk Pengujian
		[F5.2]	Penyediaan Dokumentasi Persediaan Ujian	
		[F5.3]	Penyediaan Pelan Ujian (UAT dan PAT)	D13 Pelan Ujian Penerimaan (UAT/PAT)
		[F5.4]	Ujian Penerimaan Pengguna (UAT)	
		[F5.5]	Ujian Penerimaan Provisional (PAT)	
		[F5.6]	Penyediaan Laporan Ujian Penerimaan	D14 Laporan UAT/PAT
F6	Pelaksanaan	[F6.1]	Pelaksanaan Migrasi Data	D15 Laporan Migrasi Data
		[F6.2]	Ujian Penerimaan Akhir (FAT)	D16 Laporan Penamatan Ujian
		[F6.3]	Penyediaan Manual Pengguna	D17 Manual Pengguna Sistem
		[F6.4]	Serahan Sistem Aplikasi	D18 Laporan Serahan Sistem

KENAPA BUKU INI PENTING?

Pembangunan sistem aplikasi merupakan salah satu bidang dalam teknologi ICT yang menghasilkan sistem aplikasi bagi menyediakan perkhidmatan dalam talian atau secara digital kepada pengguna. Kejayaan pembangunan sistem aplikasi bergantung kepada sejauh mana kecekapan pelaksanaan aktiviti-aktiviti dalam metodologi pembangunan sistem. Terdapat banyak buku berbentuk akademik yang menerangkan metodologi pembangunan sistem. Namun begitu kegagalan dan kelewatan dalam pembangunan sistem masih ketara berlaku. Menurut Standish Group (1995 & 1996) faktor-faktor utama yang menyumbang kepada kegagalan dalam pembangunan sistem aplikasi ialah keperluan tidak lengkap (13.1%), kurang penglibatan pengguna (12.4%) dan kekurangan sumber (10.6%). Manakala faktor-faktor utama kejayaan pula ialah penglibatan pengguna (15.9%), sokongan pihak pengurusan (13.9%) dan keperluan yang jelas (13.0%). Oleh yang demikian adalah penting penghasilan sebuah buku berbentuk praktikal dengan mengambil kira isu-isu dalam pembangunan sistem aplikasi secara komprehensif.

Memandangkan proses pembangunan sistem aplikasi adalah proses yang kompleks dan terdapat pelbagai metodologi dan pendekatan yang berbeza dalam buku-buku akademik, maka MAMPU telah mengambil inisiatif menerbitkan Buku Panduan Kejuruteraan Pembangunan Sistem Aplikasi yang standard bagi kegunaan agensi sektor awam. Buku ini dihasilkan berasaskan kemahiran intelektual dan pengalaman praktikal perunding dan pakar pembangunan sistem aplikasi MAMPU ketika memberi khidmat perundingan dan khidmat nasihat sepanjang perkhidmatan dalam sektor awam. Buku ini sangat penting kerana ianya menyediakan aktiviti-aktiviti pembelajaran melalui pengalaman amali (*experiential learning activities*) yang membolehkan individu dan pasukan pembangunan sistem mempraktikkan skil kaedah-kaedah dalam metodologi pembangunan sistem aplikasi.

Objektif buku ini dikeluarkan adalah untuk:

- a) Memperkenalkan metodologi standard yang praktikal bagi panduan agensi sektor awam dalam proses pembangunan sistem aplikasi.
- b) Meningkatkan kesedaran pemegang taruh tentang kepentingan dokumentasi sistem yang perlu dihasilkan sepanjang pelaksanaan pembangunan sistem aplikasi.
- c) Menjamin kualiti sistem aplikasi yang dibangunkan dan kesinambungan kerja dalam proses pembangunan, penyelenggaraan dan peningkatan sistem aplikasi.

BAGAIMANA MENGGUNAKAN BUKU INI?

Tujuan utama buku ini ialah untuk menyediakan panduan yang seragam dalam pembangunan sistem aplikasi kepada agensi sektor awam. Di samping itu, buku ini boleh digunakan oleh perunding, pakar, pelatih/instruktur dan fasilitator dalam memberi latihan pembangunan sistem aplikasi. Menyedari bahawa pembelajaran melalui pengalaman amali adalah kaedah pembelajaran yang paling berkesan dalam melahirkan individu yang pakar dan berkeyakinan, maka buku ini disusun bagi memenuhi keperluan tersebut. Penggunaan buku ini dengan 2 tujuan iaitu:

a) Pembelajaran bagi Pembangunan Sistem Aplikasi

Pembelajarannya bermula dari Bab 2 hingga Bab 7 iaitu bab yang menerangkan fasa-fasa dalam metodologi pembangunan sistem aplikasi sektor awam secara khusus.

b) Pembangunan bagi tujuan Latihan atau Kursus Pembangunan Sistem Aplikasi

Pembelajarannya bermula dari Bab 1 hingga Bab 7 iaitu bab 1 memberi kefahaman berkaitan metodologi pembangunan sistem aplikasi secara umum dan bab seterusnya menerangkan metodologi pembangunan sistem aplikasi sektor awam secara khusus.

KEPERLUAN MINIMA DOKUMENTASI

Dokumentasi merupakan serahan utama dalam proses pembangunan sistem aplikasi bagi memastikan sistem aplikasi yang dibangunkan adalah berkualiti dan memenuhi kehendak pengguna. Di samping itu, dokumentasi sistem aplikasi juga menjadi sumber bagi memastikan kesinambungan maklumat dari aktiviti kajian keperluan sehingga aktiviti penyelenggaraan sistem aplikasi. Oleh itu, pasukan projek pembangunan aplikasi digalakkan untuk menyediakan semua jenis dokumentasi yang disenaraikan dari D01 hingga D18. Walau bagaimanapun kelonggaran dalam penyediaan dokumentasi adalah bergantung kepada saiz dan kompleksiti sistem aplikasi yang hendak dibangunkan, penglibatan SME serta tempoh pembangunan. **Jadual 2** menggariskan keperluan minima dokumentasi yang perlu dihasilkan untuk sesuatu projek pembangunan sistem aplikasi bagi kesempurnaan dalam pembangunan sistem aplikasi.

Jadual 2 : Keperluan Minima Dokumentasi Projek Pembangunan Sistem

Nama Dokumen	Keterangan Dokumen	Keperluan Minima
D01 Pelan Pembangunan Sistem	Dokumen yang menerangkan aspek-aspek perancangan dalam pengendalian usaha-usaha pembangunan sistem yang akan dijalankan.	√
D02 Spesifikasi Keperluan Bisnes (BRS)	Dokumen ini menerangkan skop keperluan bisnes yang diperlukan oleh pemegang taruh yang terdiri daripada hierarki bisnes, rajah aliran proses dan definisi fungsi bisnes.	√
D03 Spesifikasi Keperluan Sistem (SRS)	Dokumen ini menerangkan keperluan sistem yang terdiri daripada keperluan fungsian, keperluan bukan fungsian, keperluan data dan keperluan aliran fungsi dan data yang	√
D04 Spesifikasi Rekabentuk Sistem (SDS)	Dokumen ini menjelaskan arkitektur aplikasi yang hendak dibangunkan, rekabentuk antaramuka pengguna, rekabentuk struktur pangkalan data dan pemetaan antaramuka data dan logik program yang perlu di programkan.	√
D05 Pelan Migrasi Data	Dokumen ini menjelaskan pelan pelaksanaan aktiviti migrasi data dan pendekatan migrasi data.	Boleh digabungkan dengan D01

Nama Dokumen	Keterangan Dokumen	Keperluan Minima
D06 Spesifikasi Migrasi Data	Dokumen ini menjelaskan pendekatan, arkitektur dan pemetaan migrasi data yang akan dilaksanakan.	Kandungan mungkin boleh digabungkan dengan D04
D07 Pelan Integrasi Data	Dokumen ini menjelaskan pelan pelaksanaan dan pendekatan Integrasi data.	Boleh digabungkan dengan D01
D08 Spesifikasi Integrasi Data	Dokumen ini menjelaskan pendekatan, arkitektur dan rekabentuk integrasi sistem yang akan dibangunkan.	Kandungan mungkin boleh digabungkan dengan D04
D09 Dokumentasi Pangkalan Data	Dokumen yang mengandungi ringkasan maklumat dan skrip pangkalan data yang digunakan untuk membangunkan pangkalan data fizikal.	√
D10 Dokumentasi Kod Sumber	Dokumen yang mengandungi senarai kod aturcara yang meliputi struktur direktori dan hiarki fail. Serta garis panduan yang mengesyorkan perkara berkaitan gaya pengaturcaraan (<i>style</i>), konvensyen penamaan, indentation, ulasan/komen, pengistiharan pembolehubah, pernyataan SQL dan lain-lain yang perlu dipatuhi oleh semua pengaturcara.	√
D11 Laporan Ujian Sistem	Dokumen ini mengandungi laporan berkenaan ujian sistem yang telah dilaksanakan, ia juga meringkaskan dan menyimpulkan jika perisian sedia untuk dilancarkan.	√
D12 Pelan Induk Pengujian	Dokumen yang menyenaraikan semua aktiviti bagi fasa pengujian sistem seperti jadual pelaksanaan, takrifan skop, tugas dan tanggungjawab pasukan penguji sistem, risiko, kriteria input & output serta objektif ujian. Ia termasuk Pelan Ujian Unit (<i>Unit Test Plan</i>), Pelan Ujian Integrasi (<i>Integration Test Plan</i>), Pelan Ujian Sistem (<i>System Test Plan</i>) dan Pelan Ujian Penerimaan (<i>Acceptance Test Plan</i>).	√
D13 Pelan Ujian Penerimaan (UAT/PAT)	Pelan UAT/PAT menggariskan pendekatan, pertimbangan dan langkah-langkah dalam melaksanakan UAT/PAT.	Boleh digabungkan dengan D12

Nama Dokumen	Keterangan Dokumen	Keperluan Minima
D14 Laporan UAT/PAT	a) Dokumen ini mengandungi laporan hasil UAT/PAT, hasil ujian akan menjadi <i>entry criteria</i> kepada aktiviti UAT/PAT. b) Dokumen ini mengandungi laporan hasil UAT/PAT, hasil ujian akan menjadi <i>entry criteria</i> kepada aktiviti Ujian Penerimaan Akhir (FAT).	√
D15 Laporan Migrasi Data	Dokumen ini mengandungi laporan hasil aktiviti migrasi data yang telah dilaksanakan.	√
D16 Laporan Penamatan Ujian	Dokumen ini mengandungi laporan hasil ujian penerimaan akhir yang dilaksanakan sebelum sistem diserahkan kepada pemilik sistem.	√
D17 Manual Pengguna	Dokumen ini menyediakan kaedah penggunaan sistem aplikasi yang dibangunkan.	√
D18 Laporan Serahan Sistem	Dokumen ini menerangkan status pencapaian perancangan dan perhatian yang perlu diberi perhatian oleh pemilik sistem aplikasi dalam penyelenggaraan sistem	√




BAB:01

PENGENALAN KEPADA METODOLOGI KEJURUTERAAN SISTEM APLIKASI

Bab ini merupakan pengenalan dan penerangan ringkas kepada Kitar Hayat Pembangunan Sistem (SDLC).

Bab ini juga menerangkan Ekosistem Pembangunan Sistem Sektor Awam yang terdiri dari Metodologi Pembangunan Sistem dan aspek-aspek pertimbangan yang mempengaruhi kepada kejayaan pelaksanaan projek pembangunan sistem.



1 PENGENALAN KEPADA METODOLOGI KEJURUTERAAN SISTEM APLIKASI

1.1 PENGENALAN KEPADA KITAR HAYAT KEJURUTERAAN PEMBANGUNAN SISTEM APLIKASI

Merujuk kepada *ISTQB Certification*¹, model kitar hayat pembangunan sistem aplikasi (SDLC) adalah terdiri 6 fasa utama iaitu fasa pengumpulan keperluan dan analisis, fasa rekabentuk, fasa pembangunan (*coding*), fasa pengujian, fasa pelaksanaan (*deployment*) dan fasa penyelenggaraan seperti digambarkan dalam **Rajah 1**. Setiap fasa mempunyai siri aktiviti melalui penggunaan teknik-teknik tertentu bagi penghasilan dokumentasi serahan.



Rajah 2: Kitar Hayat Pembangunan Sistem Aplikasi

Terdapat **2 elemen utama** yang perlu difahami dalam SDLC iaitu.

- a) Perlu memahami fasa-fasa dalam SDLC, langkah-langkah yang perlu dilakukan dalam projek pembangunan sistem aplikasi dan teknik-teknik yang diperkenalkan dalam penghasilan serahan yang tertentu.
- b) Perlu memahami bahawa model SDLC melibatkan serahan yang dihasilkan dalam setiap fasa. Serahan dalam setiap fasa akan digunakan sebagai input kepada fasa berikutnya.

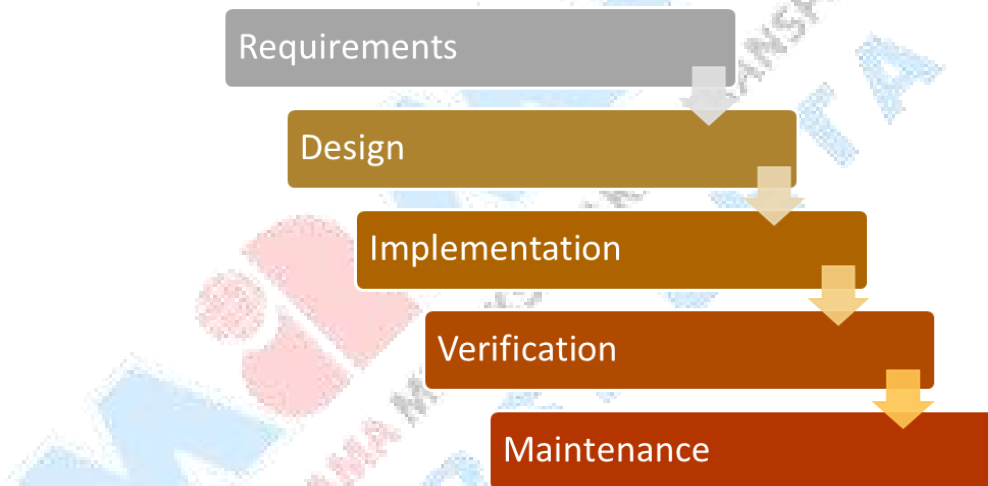
¹ <http://istqbexamcertification.com/what-are-the-software-development-life-cycle-sdlc-phases/>

1.2 METODOLOGI PEMBANGUNAN SISTEM APLIKASI

Metodologi Pembangunan Sistem Aplikasi adalah satu rangka kerja yang digunakan untuk menstruktur, merancang dan mengawal proses pembangunan sistem aplikasi. Terdapat pelbagai metodologi pembangunan sistem aplikasi yang telah diperkenalkan. Kesesuaian sesuatu metodologi pembangunan sistem aplikasi adalah bergantung kepada jenis organisasi, keupayaan dan keperluan teknikal, serta jenis dan pasukan projek. Antara model yang sering digunakan adalah:

a) Metodologi *Waterfall*

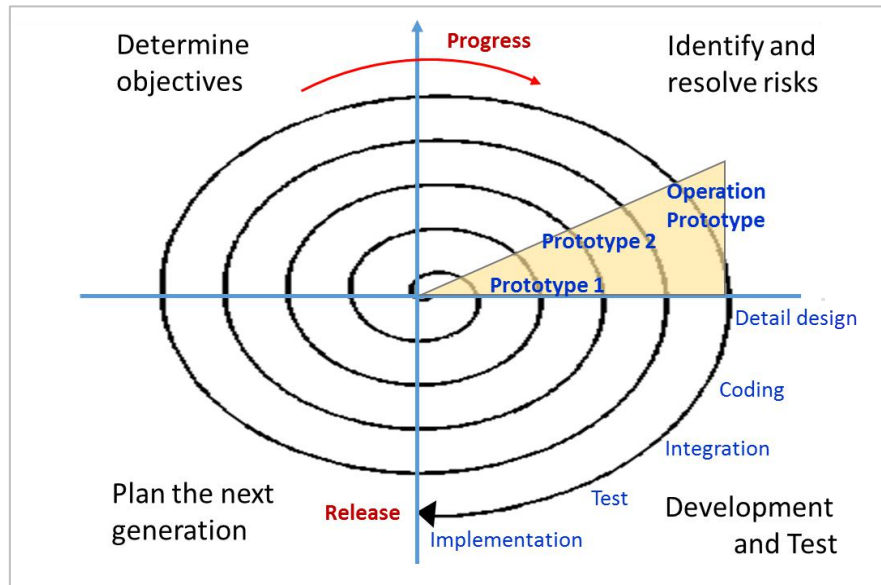
Metodologi *waterfall* juga dikenali Metodologi Jujukan *Linear*. Model ini menerapkan kawalan yang ketat terhadap setiap fasa di dalam SDLC. Semakan dan pengesahan serahan dilakukan secara formal dengan pemegang taruh pada setiap penghujung fasa. Sesuatu fasa seterusnya tidak akan dimulakan sekiranya semakan dan pengesahan serahan bagi fasa semasa tidak disempurnakan.



Rajah 3 : Metodologi *Waterfall*

b) Metodologi *Spiral*

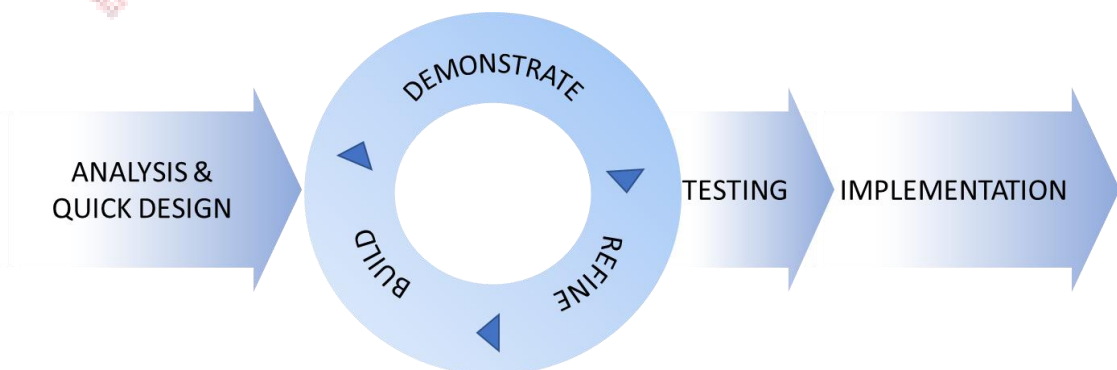
Metodologi ini adalah gabungan konsep berulang di dalam Metodologi *Prototype* dan aspek kawalan yang sistematik seperti yang terdapat pada metodologi Jujukan *Linear*. Metodologi ini memberi fokus kepada penilaian produk dan meminimumkan risiko dalam pembangunan sistem. Model ini memecahkan skop projek kepada segmen-segmen yang kecil bagi memudahkan semakan dan perubahan dilakukan.



Rajah 4 : Metodologi *Spiral*

c) Metodologi *Rapid Application Development (RAD)*

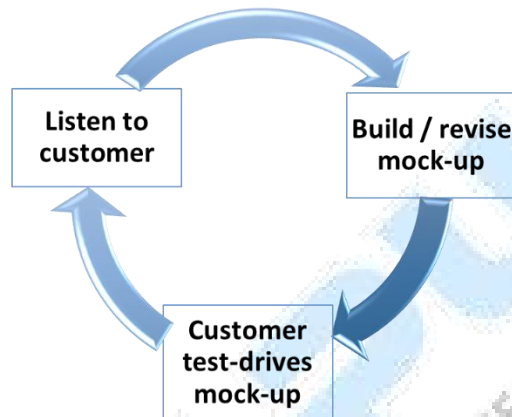
Putaran *RAD* merangkumi 4 fasa utama iaitu perancangan keperluan, rekabentuk, pembangunan dan pelaksanaan. Fasa ini dilaksanakan oleh sekumpulan pembangun aplikasi yang mahir yang bekerjasama rapat dengan pengguna sepanjang tempoh pembangunan. Teknik dan *tools* yang digunakan merupakan faktor utama kejayaan *RAD*. Matlamat utama metodologi ini adalah menghasilkan sistem yang berkualiti tinggi secara cepat dengan memberikan penekanan terhadap keperluan pengguna.



Rajah 5 : Metodologi *RAD*

d) Metodologi *Prototype*

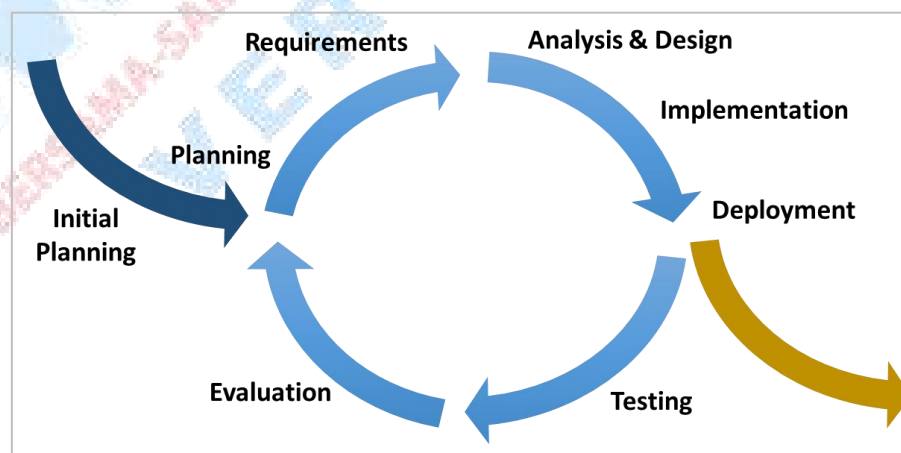
Metodologi ini terdapat sesuai digunakan jika pengguna sistem sukar mengenalpasti keperluan sistem. Terdapat 3 fasa penting dalam metodologi *Prototype* iaitu fasa keperluan pengguna, fasa pembangunan dan fasa pengujian. Proses ini diulang sehinggalah sistem lengkap dibangunkan. Metodologi ini mengaplikasikan proses perubahan lelaran (*iterative modification process*) sehingga sistem prototaip berkembang dan memenuhi kehendak pengguna.



Rajah 6 : Metodologi *Prototype*

e) Metodologi Pembangunan Berperingkat (*Incremental Development*)

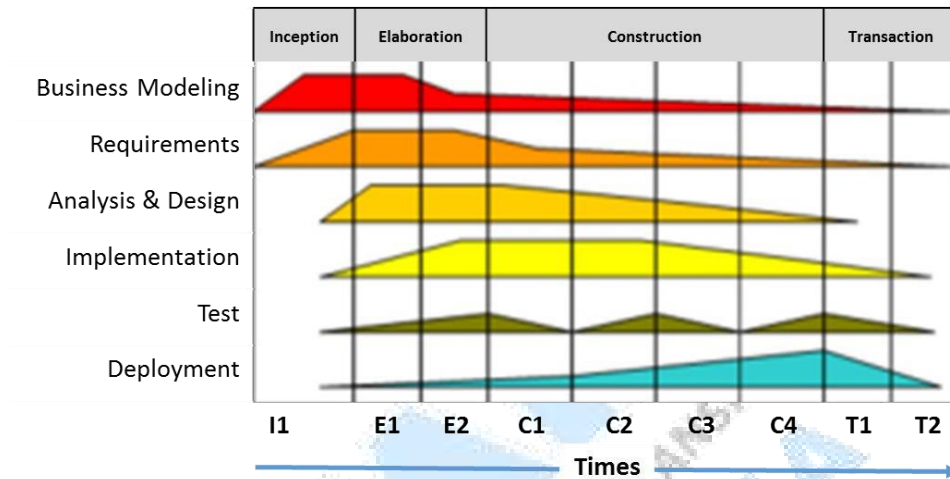
Pendekatan ini mengaplikasikan metodologi *waterfall* dalam pembangunan utama sistem (*system core*) dan diikuti dengan metodologi *Prototype* secara lelaran (*iterative*). Pembangunan secara prototaip sehinggalah prototaip menjadi sistem aplikasi yang lengkap mengikut kehendak pengguna.



Rajah 7 : Metodologi *Incremental Development*

f) Metodologi *Rational Unified Process (RUP)*

Metodologi *RUP* adalah proses pembangunan sistem yang menyediakan satu pendekatan berdisiplin dalam menetapkan tugas dan peranan sesuatu pasukan pembangunan sistem. Matlamat *RUP* adalah untuk memastikan produk yang dihasilkan menepati kehendak pengguna dan di dalam lingkungan peruntukan dan jadual yang dianggarkan. 4 fasa utama *RUP* adalah *Inception*, *Elaboration*, *Construction* dan *Transaction*.



Rajah 8 : Metodologi *RUP*

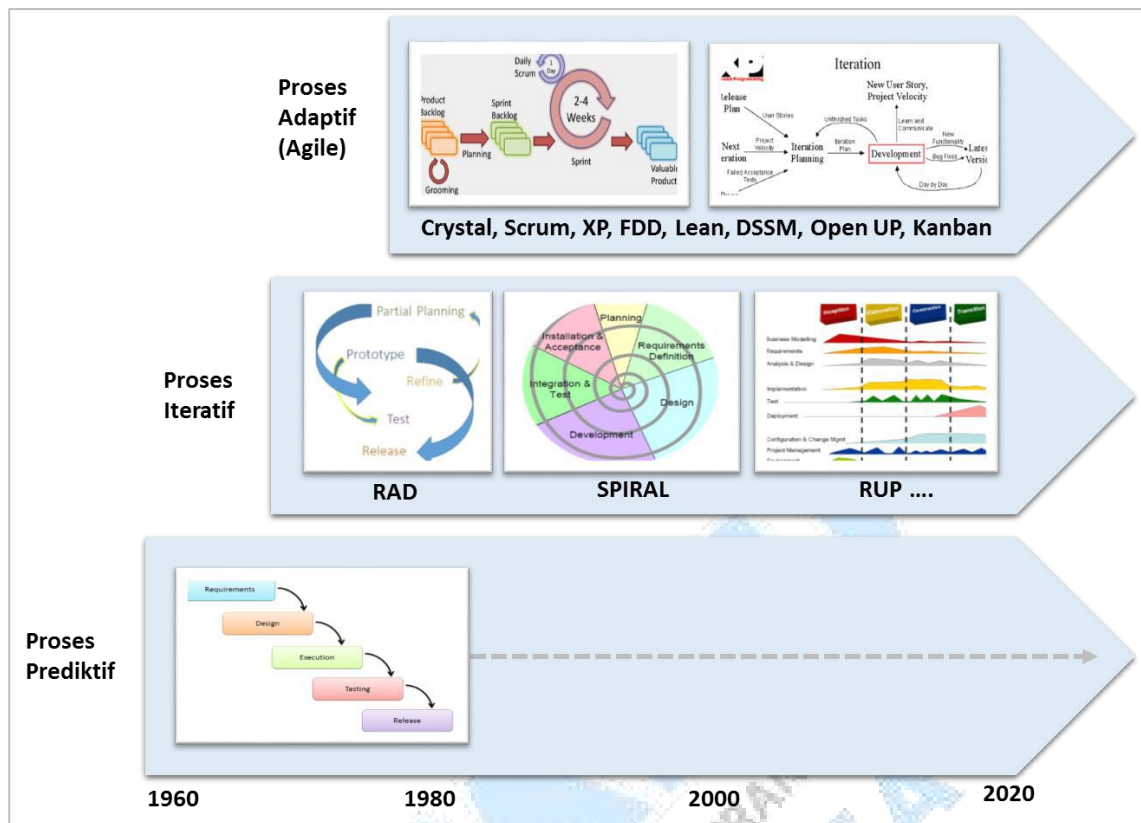
1.3 KLASIFIKASI METODOLOGI PEMBANGUNAN SISTEM APLIKASI

Metodologi-metodologi pembangunan sistem aplikasi yang digunakan boleh diklasifikasikan kepada 3 model utama iaitu model yang berasaskan kepada proses prediktif, model yang berasaskan kepada proses iteratif dan model yang berasaskan kepada proses adaptif. Model-model ini menentukan pendekatan dalam proses pembangunan sistem aplikasi.

Model Prediktif, proses pembangunan sistem aplikasi berlaku dalam satu siri fasa-fasa secara tersusun. Selepas keperluan sistem dipersetujui, rekabentuk sistem dibuat dan diikuti dengan pengaturcaraan. Akhirnya, sistem diuji sebagai pengesahan pematuhan kepada keperluan. Metodologi *Waterfall* termasuk dalam kategori model ini.

Model Iteratif merupakan model yang berasaskan penemuan (*discovery-based*) di mana pada peringkat awal dokumen keperluan sistem lebih ringkas berbanding dengan model prediktif. Bermula dengan dokumen asas yang menerangkan apa yang hendak dibangunkan, keperluan sebenar sistem akan dikenalpasti dan ditemui semasa proses iteratif. Metodologi *Spiral*, *RAD* dan *RUP* termasuk dalam kategori model ini.

Model Adaptif merupakan evolusi daripada model prediktif dan model iteratif. Model ini juga dikenali metodologi *Agile*. Model ini membolehkan organisasi menyelesaikan masalahnya secara holistik melalui persekitaran pembangunan sistem yang transparan dan adaptif. Persekitaran ini diwujudkan melalui pembentukan pasukan pembangunan yang menekankan kolaboratif antara pihak bisnes (SME) dan pihak ICT untuk menghasilkan sistem yang selari dengan strategi organisasi dengan cepat. Tanpa melengkapkan dokumen keperluan lebih awal, aktiviti pengaturcaraan akan dilaksanakan secepat mungkin dan akan terus dinilai oleh SME. Kelemahan atau kekurangan yang ditemui kemudiannya akan terus dibetulkan oleh pihak ICT. Ini bermakna perubahan kepada keperluan sistem boleh berlaku di semua fasa pembangunan sistem dan dokumen-dokumen sistem dimuktamadkan selepas pihak SME berpuas hati dengan produk yang dibangunkan. Metodologi *Scrum*, *Extreme Programming (XP)* dan *Lean* termasuk dalam kategori model ini. **Rajah 9** merupakan Model Metodologi Pembangunan Sistem Aplikasi.



Rajah 9 : Model Klasifikasi Metodologi Pembangunan Sistem Aplikasi

1.4 EKOSISTEM PEMBANGUNAN SISTEM SEKTOR AWAM

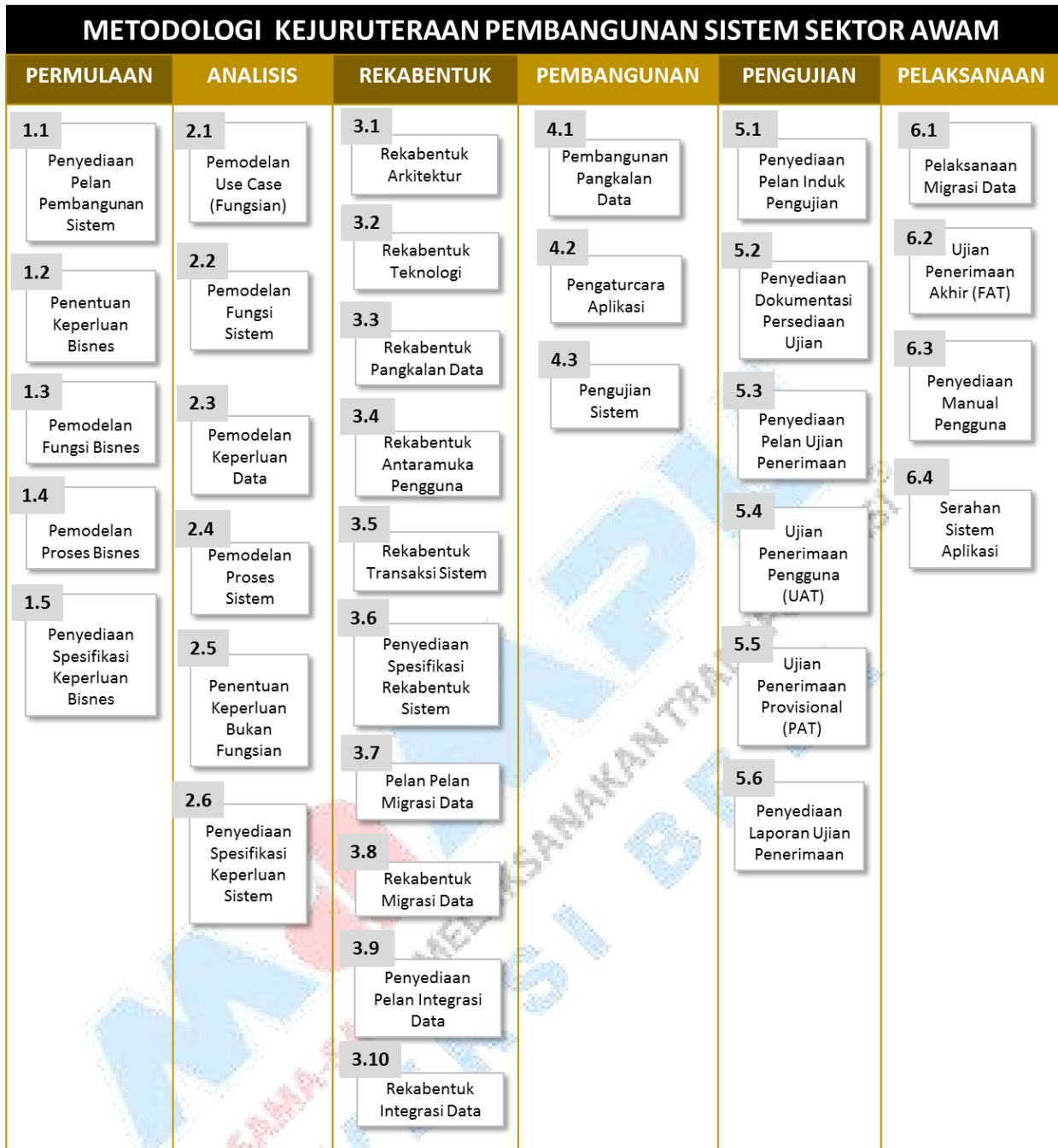
Ekosistem Pembangunan Sistem Aplikasi Sektor Awam mengambil kira elemen-elemen lain dalam persekitaran ICT dan hubungannya dengan metodologi dan pendekatan pembangunan sistem aplikasi di agensi-agensi Sektor Awam. Ekosistem tersebut digambarkan dalam **Rajah 9**. Aspek pertimbangan utama dalam pembangunan sistem aplikasi adalah Anggaran Keperluan Sumber, Pengurusan Kawalan Pindaan, Penglibatan Pemegang Taruh, Jaminan Kualiti Perisian (SQA), Keselamatan Aplikasi dan Tadbir Urus. Aspek-aspek ini perlu dipacu bersama-sama metodologi pembangunan sistem aplikasi ke arah mencapai Pelan Strategik ICT dan Pelan Strategik Bisnes yang dirancang.



Rajah 10 : Ekosistem Kejuruteraan Pembangunan Sistem Aplikasi Sektor Awam

1.5 METODOLOGI KEJURUTERAAN SISTEM APLIKASI SEKTOR AWAM

Metodologi Kejuruteraan Sistem Aplikasi Sektor Awam dibangunkan dengan mengambilkira tinjauan dan pengalaman ke atas metodologi pembangunan sistem yang telah diamalkan dalam industri dan sektor awam. Metodologi ini merangkumi 6 fasa utama iaitu: fasa permulaan projek (*initiation*), fasa analisa (*analysis*), fasa rekabentuk (*design*), fasa pembangunan (*construction*), fasa Pengujian (*testing*) dan fasa pelaksanaan (*implementation*). **Rajah 11** merupakan Metodologi Pembangunan Sistem Aplikasi Sektor Awam yang menunjukkan hubungan antara fasa dan aktiviti dalam setiap fasa.



Rajah 11 : Metodologi Kejuruteraan Pembangunan Sistem Aplikasi Sektor Awam

Fasa 1 : Permulaan Projek

Fasa permulaan projek memberikan penekanan kepada komunikasi dan komitmen Pemegang Taruh dan organisasi yang bertanggungjawab membangunkan sesuatu projek selepas sesuatu permintaan diterima. Fasa ini dibahagikan kepada 2 aktiviti utama iaitu:

a) Penyediaan Pelan Pembangunan Sistem / Piagam Pelanggan

Penyediaan Pelan Pembangunan Sistem (PPS) adalah penting sebagai komitmen pasukan projek dengan pemilik sistem dan pemegang taruh. Butiran komitmen merangkumi skop, tujuan, keperluan sumber, tempoh projek dan faedah/impak sistem kepada bisnes organisasi. Pelan Pembangunan Sistem perlu mendapat persetujuan bersama antara pasukan pembangun dan pemilik sistem.

b) Kajian Keperluan Bisnes

Penentuan Keperluan Bisnes memberikan tumpuan ke atas aktiviti mengumpul keperluan bisnes sesuatu organisasi (organisasi pemegang taruh). Ini untuk memastikan sistem yang dibangunkan menepati keperluan organisasi *Pemegang Taruh* secara menyeluruh (contoh: menepati visi, misi dan objektif agensi) dan menepati keperluan organisasi secara spesifik. Proses-proses yang berlaku di dalam aktiviti ini adalah mendefinisikan unit Bisnes dan Peranan, Pemodelan Fungsi dan Proses Bisnes. Hasil daripada aktiviti ini ialah D02 Spesifikasi Keperluan Bisnes.

Fasa 2 : Analisis

Matlamat utama fasa ini adalah melaksana analisis ke atas keperluan secara terperinci untuk menghasilkan **D02 Spesifikasi Keperluan Bisnes** Output bagi aktiviti ini adalah **D03 Spesifikasi Keperluan Sistem** yang menyatakan keperluan bagi sistem dari perspektif pembangun sistem. Ia menyatakan perkara-perkara atau item-item yang perlu ada didalam sesuatu sistem bagi merealisasikan keperluan bisnes atau pemegang taruh. Proses-proses yang berlaku di dalam aktiviti ini adalah pemodelan use case (fungsi), pemodelan keperluan data dan pemodelan proses sistem.

Fasa 3 : Rekabentuk

Berdasarkan keperluan sistem yang diperolehi di dalam Fasa Analisis, arkitektur keseluruhan sistem akan dihasilkan. Arkitektur sistem ini mendefinisikan komponen, perlakuan dan antaramuka komunikasi bagi sesuatu sistem. Fasa ini juga menerangkan tentang bagaimana sistem ini akan dihasilkan. Ia merangkumi aktiviti-aktiviti seperti rekabentuk arkitektur; rekabentuk sistem; rekabentuk pangkalan data, serta penentuan teknologi yang akan digunakan. Output kepada fasa ini adalah **D04 Spesifikasi Rekabentuk Sistem**.

Fasa 4 : Pembangunan

Aktiviti-aktiviti yang dilaksanakan di dalam fasa pembangunan adalah berkaitan proses penghasilan dan pengujian sistem oleh pasukan pembangun. Fasa pembangunan merealisasikan SDS yang dihasilkan dalam fasa rekabentuk. Komponen dan fungsi sistem dibangunkan melalui proses pengaturcaraan dan diintegrasikan untuk menghasilkan sesuatu produk atau sistem. Aktiviti-aktiviti yang dilaksanakan di dalam fasa ini adalah pembangunan pangkalan data, pengaturcaraan sistem dan pengujian sistem. Di dalam fasa ini, proses penghasilan sistem aplikasi adalah matlamat utama.

Fasa 5 : Pengujian

Aktiviti-aktiviti yang dilaksanakan di dalam fasa pengujian adalah berkaitan dengan penyediaan pelan ujian dan dokumentasi berkaitan ujian, serta pelaksanaan pengujian penerimaan oleh pengguna ke atas sistem. Pengujian yang dimaksudkan adalah Ujian Penerimaan Pengguna (UAT) dan Ujian Penerimaan Sementara (PAT). Ujian ini dilaksanakan sebagai validasi ke atas sistem aplikasi yang dibangunkan berdasarkan keperluan pengguna dan keperluan sistem bagi memastikan keperluan tersebut dipenuhi sebelum sistem aplikasi dilaksanakan.

Fasa 6 : Pelaksanaan

Aktiviti utama di dalam fasa pelaksanaan adalah melaksanakan aktiviti ke arah persediaan pelaksanaan sistem. Aktiviti-aktiviti yang dilaksanakan di dalam fasa ini adalah migrasi data, ujian penerimaan akhir, persediaan manual pengguna dan laporan serahan sistem.

1.6 FAKTOR KEJAYAAN DALAM PEMBANGUNAN SISTEM APLIKASI

Ekosistem Kejuruteraan Sistem Aplikasi Sektor Awam telah menggariskan beberapa faktor kejayaan dalam pembangunan sistem iaitu Tadbir Urus Pengurusan Projek ICT, Pengurusan Kawalan Pindaan, Jaminan Kualiti Perisian (SQA), Faktor Keselamatan ICT, Pengukuran Saiz Sistem Aplikasi dan Penglibatan Pemegang Taruh. Aspek-aspek utama ini perlu dititikberatkan dalam melaksana metodologi kitar hayat pembangunan sistem ke arah mencapai matlamat bisnes yang ditetapkan.

1.6.1 TADBIR URUS PENGURUSAN PROJEK ICT

Merujuk kepada Buku Metodologi PRISA Panduan Pengurusan Projek ICT Sektor Awam, 2016 yang diterbitkan oleh MAMPU, tadbir urus projek adalah penting bagi memastikan kejayaan projek ICT. Ianya meliputi penubuhan struktur pasukan projek pembangunan dan jawatankuasa pemantauan projek.

Organisasi Pasukan Projek Pembangunan diketuai oleh Pengurus Projek dengan bantuan Pejabat Pengurusan Projek (PMO) yang berfungsi sebagai penyelarasan kepada 2 pasukan utama projek seperti berikut:

a) Pasukan Pembangunan

Pasukan pembangunan bertanggungjawab dalam melaksana aktiviti mengkaji, menganalisa, membangun sistem aplikasi dan melaksana pengujian sistem

b) Pasukan SME

Pasukan SME bertanggungjawab dalam menyediakan input D03 Spesifikasi Keperluan Sistem dan melaksana pengujian penerimaan ke atas sistem

Jawatankuasa pemantauan projek adalah terdiri 2 jawatankuasa utama iaitu:

a) Jawatankuasa Pemandu Projek

Jawatankuasa Pemandu Projek bertanggungjawab memutuskan strategi dan halatuju projek pembangunan yang terdiri daripada keperluan sumber manusia, kewangan dan proses. Jawatankuasa ini juga memantau pelaksanaan projek secara keseluruhan.

b) Jawatankuasa Teknikal Projek

Jawatankuasa Teknikal Projek bertanggungjawab memperaku halatuju, serahan dan strategik, menyelesaikan isu-isu teknikal, menyelaras dan memantau pelaksanaan mengikut skop projek yang dipersetujui.

1.6.2 PENGLIBATAN PEMEGANG TARUH

Penglibatan pemegang taruh sepanjang aktiviti pembangunan sistem aplikasi dikira sebagai faktor utama bagi memastikan kejayaan dalam pembangunan dan pelaksanaan sistem aplikasi. Di samping menyumbang kepada kualiti sistem yang dibangunkan, penglibatan pemegang taruh juga dapat menyelesaikan strategi pengurusan perubahan ke arah penggunaan sistem aplikasi yang dibangunkan.

Faedah Yang Diperolehi

- a) Meningkatkan kualiti sistem dan ketetapan kepada kehendak dan keperluan pengguna dalam menyokong peranan dan bisnes agensi
- b) Meningkatkan kefahaman dan penerimaan sistem aplikasi yang dibangunkan
- c) Dengan penglibatan SME dalam mengenalpasti keperluan sistem dalam konteks fungsi organisasi, ini dapat mengurangkan masa pembangunan
- d) Memupuk perasaan pemilikan sistem baru, dengan penglibatan pengguna dalam proses pembangunan akan menyebabkan pengguna lebih komited untuk menggunakan sistem dan untuk memastikan kejayaannya semasa pelaksanaan.

Kategori Pemegang Taruh

Dalam pelaksanaan pembangunan sistem agensi sektor awam, terdapat 5 kategori pemegang taruh iaitu:

a) **Penganjur Projek**

Penganjur projek adalah Ketua Jabatan/Agensi yang melulus dan memberi peruntukan bagi pembangunan sistem aplikasi. Sokongan dari pengurusan atasan agensi adalah faktor utama yang membawa kepada kejayaan pembangunan dan pelaksanaan sistem aplikasi.

b) **Pemilik Projek**

Pemilik projek adalah Ketua Bahagian yang memiliki fungsi dan mempunyai kuasa terhadap dasar dan prosidur sistem aplikasi yang sedang dibangunkan. Mereka bertanggungjawab dalam memastikan sistem yang dibangunkan memenuhi keperluan jabatan.

c) **Subject Matter Expert (SME)**

SME adalah kumpulan pegawai rujukan kepada fungsi bisnes agensi berkaitan dengan sistem yang akan dibangunkan.

d) **Ahli Pasukan Projek**

Pasukan pengguna yang terlibat dalam menyatakan keperluan sistem, terlibat dalam rekabentuk sistem dan pengujian sistem. Mereka akan menjadi pegawai penghubung antara pengguna akhir sistem dan pasukan pembangun.

e) **Penasihat Projek**

Pengurusan atasan agensi yang memberi nasihat dalam keperluan pengguna dan sepanjang pentadbiran dan pengurusan projek dilaksanakan. Selalunya penasihat projek akan mempengerusi Mesyuarat Jawatankuasa Projek.

1.6.3 PENGURUSAN KAWALAN PINDAAN

Permintaan pindaan adalah permohonan untuk membuat pelarasan semula skop atau keperluan sistem yang telah dipersetujui. Permintaan perubahan ini amat penting diuruskan bagi memastikan tiada kesan kelewatan dalam pembangunan sistem aplikasi yang sedang berlaku. Permintaan pindaan keperluan biasanya berpunca dari sebab-sebab berikut:

- a) Masalah yang dilaporkan dikenalpasti sebagai ralat yang mesti diperbetulkan.
- b) Peningkatan sistem yang diminta dari pihak pengguna.
- c) Pertukaran standard, polisi atau akta berkaitan bisnes yang membawa kepada perlunya sistem diubahsuai.
- d) Permintaan daripada pengurusan kanan yang memerlukan pertambahan atau pengubahsuaian fungsi sistem.

Buku Metodologi PRrISA Panduan Pengurusan Projek ICT Sektor Awam, 2016 yang diterbitkan oleh MAMPU, menggariskan pendekatan bagi kawalan pindaan di bawah Fasa Pelaksanaan dan Kawalan, menjelaskan perkara berikut:

- a) Borang *Change Request* - borang standard yang perlu diisi oleh pihak pengguna bagi membuat permintaan pindaan dan sokongan daripada pemilik sistem
- b) Penyata Pindaan – keperluan untuk mendaftar permintaan pindaan, kelulusan tindakan, kaedah penyelesaian dan status tindakan
- c) Log Penyelesaian Isu – catatan kelulusan penyelesaian ke atas pindaan dan aktiviti yang telah dilaksanakan
- d) Lembaga kawalan perubahan – Lembaga /peringkat yang membincangkan dan meluluskan permintaan kawalan
- e) Aliran Proses Kawalan Pindaan – menjelaskan aliran kerja permintaan pindaan.

1.6.4 JAMINAN KUALITI PERISIAN (SQA)

SQA adalah satu pendekatan yang terancang dan sistematik untuk menilai kualiti dan pematuhan kepada standard aplikasi, proses dan prosedur. Ini termasuklah proses bagi menjamin bahawa standard dan prosedur yang diwujudkan dipatuhi sepanjang aktiviti kitar hayat pembangunan sistem. SQA berbeza daripada ujian kerana ia adalah berorientasikan pencegahan manakala ujian adalah berorientasikan pengesanan. SQA adalah proses semakin bermula dari peringkat awal pembangunan, dengan ini dapat membantu dalam mengenal pasti masalah awal seterusnya menyumbang ke arah peningkatan kualiti sistem. Terdapat 2 item utama yang perlu dibincangkan dalam pelaksanaan SQA iaitu ciri-ciri sistem aplikasi yang berkualiti diukur berdasarkan atribut kualiti perisian dan aktiviti pengujian verifikasi dan validasi bagi memastikan sistem aplikasi yang dihasilkan berkualiti dan memenuhi ciri-ciri kualiti perisian.

a) Atribut Kualiti Perisian

Kualiti sistem aplikasi ditentukan oleh atribut kualiti seperti **Jadual 3**.

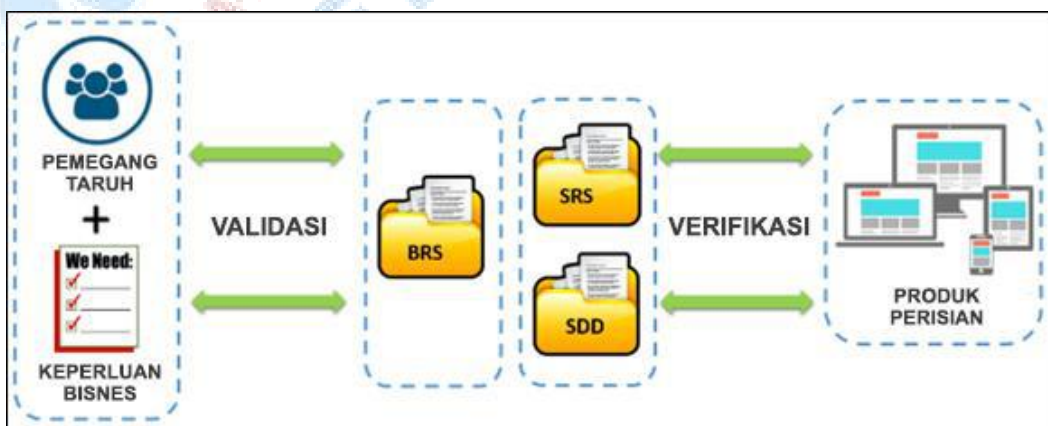
Jadual 3 : Atribut Kualiti Perisian

Atribut	Keterangan	Contoh
Atribut fungsian (Functional attributes)	Input dan output produk perisian	<ul style="list-style-type: none"> i. Sistem dapat memenuhi keperluan pengguna dengan tepat (<i>correctness</i>). ii. Sistem mengawal capaian modul mengikut peranan pengguna (<i>authentication</i>). iii. Sistem memaparkan maklumat sulit kepada pengguna tertentu sahaja (<i>integrity</i>).Aspek keselamatan telah diambil berat semasa pembangunan sistem (<i>security</i>)
Atribut operasi (Operational attributes)	Syarat operasi sesuatu produk perisian	<ul style="list-style-type: none"> i. Sistem berupaya memberikan tindak balas (<i>latency or response time</i>) dalam masa yang ditetapkan ii. Sistem dapat menampung kapasiti (<i>capacity</i>) pengguna serentak yang ditetapkan iii. Sistem dapat beroperasi walaupun melebihi kapasiti pengguna yang ditetapkan.
Atribut kebolegunaan (Usability attributes)	Sejauh mana produk perisian boleh digunakan dan disesuaikan	<ul style="list-style-type: none"> i. Sistem mudah digunakan (<i>ease of use</i>) oleh pengguna ii. Sistem mudah dipelajari (<i>ease of learn</i>) oleh pengguna

Atribut	Keterangan	Contoh
	dengan produk perisian	iii. Sistem mudah disesuaikan (<i>customizable</i>) mengikut keperluan operasi pengguna iv. Sistem sedia berkolaborasi dengan aplikasi lain (<i>interoperability</i>)
Atribut perniagaan (Business attributes)	Kos pembangunan, penggunaan serta perubahan produk perisian	i. Kos pembangunan sistem ii. Kos penyelenggaraan system iii. Penjimatan kos apabila produk atau komponennya boleh diguna semula (<i>reusability</i>) iv. Penjimatan kos apabila produk boleh digunakan pada platform berbeza (<i>portability</i>)
Atribut struktur (Structural attributes)	Struktur dalaman produk perisian	i. Sistem yang dibangunkan berupaya untuk diuji (<i>testability</i>) ii. Sistem mudah diadaptasi (<i>adaptability</i>) mengikut perubahan keperluan pengguna iii. Pembangunan sistem mengambil kira struktur kemandulan (<i>modularity</i>)

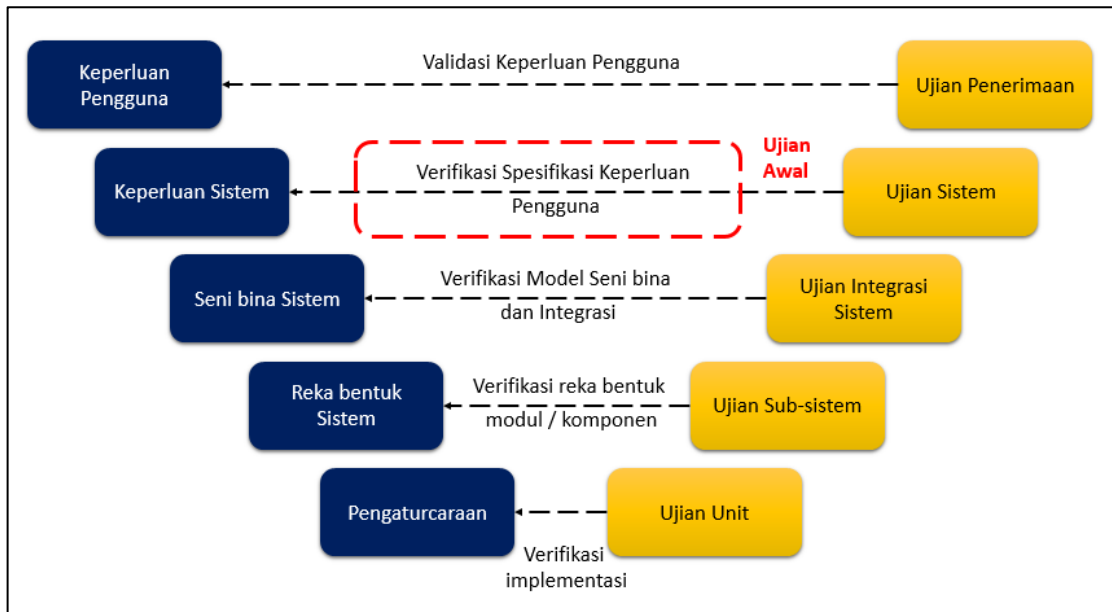
b) Verifikasi dan Validasi

Verifikasi adalah proses semakan dokumentasi dan rekabentuk sistem (*static testing*) untuk memastikan produk yang telah dibangunkan mematuhi keperluan yang ditetapkan (*system was built right*) dalam setiap fasa pembangunan sistem. Antara tujuan verifikasi adalah untuk menghasilkan keperluan yang betul, tepat, lengkap dan konsisten. Validasi adalah aktiviti bagi memastikan produk yang dihasilkan memenuhi spesifikasi keperluan (*right system built*) melalui proses pengujian dinamik. Rujuk rajah di bawah.



Rajah 12 : Proses Verifikasi dan Validasi

Pengujian Verifikasi dan Validasi (V&V) hendaklah dilaksanakan oleh pasukan pembangunan sistem itu sendiri, pada setiap fasa kitaran hayat pembangunan sistem. Rajah di bawah menunjukkan 4 jenis pengujian yang perlu dilaksanakan iaitu:



Rajah 13 : Jenis-Jenis Pengujian V&V

Jadual 4 : Jenis-Jenis Pengujian IV&V

Jenis Ujian	Keterangan
Pengujian Awal (Early Testing)	Jenis pengujian adalah Ujian Statik dalam fasa permulaan projek. Pendekatan ujian adalah secara <i>review</i> atau/dan <i>walkthrough</i> ke atas keperluan fungsian (<i>functional</i>) dan bukan fungsian (<i>non-functional</i>) seperti yang dinyatakan dalam SRS. Artifak yang terlibat adalah seperti: <ul style="list-style-type: none"> • Dokumen tender • Cadangan tender daripada pembekal yang dilantik. • <i>Software Requirement Specification (SRS)</i>.
Ujian Unit (Unit Testing)	Aktiviti pengujian bagi menilai unit terkecil (lowest level) di dalam sistem seperti kod program, <i>function</i> dan sebagainya
Ujian Integrasi (Integration Testing)	Aktiviti pengujian dengan sistem-sistem dalaman dan sistem-sistem luaran yang terlibat secara terus dengan aplikasi yang sedang dibangunkan.
Ujian Sistem (System Testing)	Aktiviti pengujian yang terlibat adalah Pengujian Functional dan Non-Functional untuk keseluruhan modul yang dibangunkan
Ujian Penerimaan (Acceptance Testing)	Ujian Penerimaan Pengguna terbahagi kepada tiga (3) seperti turutan berikut: <ul style="list-style-type: none"> • Ujian Penerimaan Pengguna (UAT) • Ujian Penerimaan Sementara (PAT) • Ujian Penerimaan Akhir (FAT)

Penggunaan SQA dalam proses pembangunan sistem akan membawa kepada faedah-faedah berikut:

- a) Memastikan standard dan prosedur dipatuhi.
- b) Mewujudkan dokumentasi system yang seragam dan mudah difahami.
- c) Memastikan masalah dapat ditemui awal dan diuruskan dengan sewajarnya.
- d) Memantau dan menambah baik proses pembangunan perisian.

1.6.5 FAKTOR KESELAMATAN ICT

Elemen keselamatan dalam pembangunan sistem aplikasi adalah perlu diambil kira dalam proses kitaran hayat pembangunan sistem aplikasi. Sistem aplikasi sering terdedah kepada serangan kerana wujudnya kelemahan semasa aktiviti rekabentuk dan pengaturcaraan. Kelemahan sistem aplikasi boleh menyebabkan ia boleh diubahsuai oleh pihak yang tidak bertanggungjawab sehingga sistem aplikasi tersebut tidak dapat beroperasi dengan baik. Tindakan pencegahan adalah penting untuk mengurangkan ancaman ke atas sistem aplikasi. Aspek yang perlu diambil kira dalam menjamin keselamatan sistem adalah:

a) Prinsip umum Pembangunan Sistem Terselamat

Keselamatan sistem boleh diancam pada mana-mana aktiviti sepanjang kitaran hayatnya, sama ada secara sengaja atau tidak sengaja oleh "orang dalam" atau oleh "orang luar" yang tidak mempunyai hubungan dengan organisasi. Penambahbaikan kelemahan yang dikesan pada peringkat awal sepanjang kitaran hayat sistem aplikasi adalah lebih kos efektif daripada membangun dan mengeluarkan versi/patch keselamatan baru untuk sistem beroperasi. Untuk dianggap selamat, sistem mestilah mempunyai ciri-ciri:

i) Kebergantungan (*Dependability*)

Sistem yang dibangunkan boleh beroperasi dengan lancar di bawah semua keadaan dan platform, termasuklah sekiranya terdapat serangan atau pelbagai niat jahat.

ii) Dipercayai (*Trustworthiness*)

Sistem boleh dipercayai walaupun terdapat kelemahan yang sengaja dieksploitasi untuk mengganggu pengoperasian sistem. Sistem yang boleh dipercayai mesti mengandungi logik yang boleh menghalang sebarang pencerobohan jahat.

iii) Daya Tahan (*Resilience*)

Sistem yang berdaya tahan adalah sistem yang mampu menentang serangan atau dapat dipulihkan dengan cepat sekiranya ada serangan yang tidak boleh ditolak/ ditahan.

b) Ciri-Ciri Keselamatan Utama Pembangunan Sistem

Terdapat 7 ciri-ciri keselamatan yang perlu diambil kira bagi memastikan sistem aplikasi adalah selamat. Ciri-ciri tersebut adalah seperti berikut:

i) Keselamatan Tahap Sistem

Sistem dapat mengawal akses aplikasi pada peranan setiap pengguna. Biasanya sistem dikawal berasaskan paparan pilihan menu yang berbeza mengikut peranan pengguna.

ii) Keselamatan *Row-Level*

Pengguna hanya dapat melihat data yang dibenarkan mengikut peranan mereka di dalam aplikasi yang sama.

iii) *Single Sign-On (SSO)*

Satu proses pengesahan pengguna yang membolehkan pengguna memasukkan id pengguna dan kata laluan mereka pada satu sistem sahaja tetapi pengesahan tersebut memberi kebenaran kepada pengguna untuk mengakses sistem lain yang berkaitan pada sesi yang sama tanpa perlu log masuk ke sistem tersebut. SSO akan mengurangkan bilangan ID Pengguna/kata laluan yang perlu diingat dan memerlukan login pada setiap aplikasi yang diperlukan.

iv) Parameter Keistimewaaan Pengguna

Parameter keistimewaaan pengguna digunakan untuk memperibadikan ciri dan keselamatan kepada pengguna individu atau peranan pengguna. Parameter keistimewaaan pengguna disimpan ke profil pengguna dan boleh diakses pada keseluruhan sistem. Ianya sangat fleksibel, dapat mengawal, menambah atau menyembunyi pilihan pengguna. Contoh: walaupun semua pengguna boleh mengakses aplikasi yang sama, tetapi hanya pengguna tertentu sahaja yang boleh melihat pilihan untuk mengemaskini data.

v) Pilihan Pengesahan Fleksibel (*Flexible Authentication Options*)

Sistem aplikasi yang dibangunkan sepatutnya menawarkan pelbagai pilihan kaedah pengesahan pengguna, sistem sepatutnya fleksibel untuk menggunakan kaedah pengesahan yang sedia ada tanpa perlu wujud atau mengubah kaedah pengesahan semasa.

vi) Sumber Data Pengguna (*User Specific Data Source*)

Ciri keselamatan ini adalah sama dengan row-level keselamatan, tetapi di peringkat pangkalan data. Aplikasi yang dibangunkan boleh mengakses sumber data yang berbeza bergantung kepada pengguna. Contoh: Apabila berlaku penggabungan 2 syarikat yang menggunakan sistem yang sama. Syarikat A perlu akses pangkalan data *local*, manakala pekerja syarikat B perlu akses data dari pangkalan data berlainan.

vii) Pengauditan Aktiviti Sistem

Pengauditan aktiviti sistem membolehkan pembangun log aktiviti pengguna akhir untuk setiap aktiviti *Sign On / Sign Off*. Ini membolehkan sistem menjejaki dengan cepat dan merekod log masuk dan keluar pengguna, fungsi yang dicapai dan aktiviti yang dilakukan terhadap data.

1.6.6 KEPENTINGAN PENGUKURAN SAIZ SISTEM

Dalam perancangan pembangunan projek yang melibatkan pembangunan sistem aplikasi, langkah pertama yang perlu dilakukan ialah membuat anggaran 4 parameter utama iaitu saiz sistem, usaha pembangunan (development effort), masa dan kos. Kesemua parameter ini merupakan input utama bagi pelan keseluruhan projek.

Apabila saiz sistem tidak diukur dengan tepat, maka kebiasaannya anggaran usaha pembangunan, masa dan kos adalah hanya berdasarkan kepada pengalaman dan rekod-rekod sejarah yang lepas. Pengalaman dan rekod-rekod sejarah tersebut pula dikumpulkan berdasarkan kepada cadangan pihak kontraktor di mana kadang-kadang kos yang ditawarkan boleh dipersoalkan sama ada berpatutan atau sebaliknya. Oleh yang demikian, adalah sangat penting saiz sistem diukur dengan lebih tepat dan secara saintifik serta mematuhi piawaian antarabangsa.

Apabila membuat perancangan projek-projek ICT yang melibatkan pembangunan sistem baru atau peningkatan sistem sedia ada, pihak pengurusan memerlukan jawapan kepada tiga persoalan asas berikut:

- a) Berapakah jumlah kos yang terlibat?
- b) Berapa lama masa diambil untuk menyiapkan projek?
- c) Siapa akan melaksanakannya dan adakah pilihan yang dibuat akan mengurangkan kos atau mempercepatkan pelaksanaan projek?

Jawapan kepada persoalan bisnes ini dapat diberikan sekiranya saiz sistem yang hendak dibangunkan diketahui. Pengiraan saiz sistem merupakan salah satu aktiviti dalam kejuruteraan sistem untuk membuat anggaran saiz sistem aplikasi yang hendak dibangunkan. Terdapat beberapa kaedah pengiraan saiz perisian, antaranya ialah COSMIC Function Points, Mk II Function Points, Nesma Function Points, FiSMA Function Points dan kaedah IFPUG. Kaedah yang paling terkenal ialah kaedah IFPUG yang juga dikenali dengan Function Point Analysis (FPA). FPA dimiliki dan diselenggara oleh IFPUG (*International Function Point Users Group*) iaitu sebuah organisasi bukan berasaskan keuntungan yang telah ditubuhkan pada 1986. FPA ini mematuhi standard pengukuran fungsian sistem aplikasi ISO/IEC 14143-1:2007. Metrik pengukuran *function points* (FP) telah menjadi standard *de facto* untuk analisis ekonomik sistem aplikasi, sebagai penanda aras kepada produktiviti dan kualiti sistem aplikasi dan lain-lain pengukuran yang berkaitan. Keterangan yang terperinci mengenai *function points* dan kaedah IFPUG FPA akan diterangkan dalam Bab 8.

Persoalan yang selalu ditanya apabila membuat inisiatif berkaitan pembangunan sistem aplikasi ialah berapa besar atau berapa kompleks sistem aplikasi yang hendak dibangunkan. Adakah sistem yang hendak dibangunkan bersaiz kecil, sederhana atau besar? Apabila kaedah FPA diaplikasikan, penentuan kategori saiz sistem dapat ditentukan dengan yakin dan jelas. Capers Jones Pengarang buku *Software Engineering Best Practices - Lessons from Successful Projects in the Top Companies* (2010), menyatakan perisian kategori kecil ialah perisian yang kurang daripada 100FP,

kategori sederhana bagi perisian bersaiz antara 100FP hingga 999FP dan kategori besar bagi perisian bersaiz 1000FP ke atas. Apabila saiz sistem dapat diketahui, maka kos, tempoh masa dan usaha pembangunan dapat ditentukan dengan merujuk kepada jadual standards antarabangsa mengikut negara atau mengikut standards organisasi sekiranya ada.

FP sudah dan sedang diterima secara meluas sebagai metrik standards untuk mengukur saiz perisian. Kefahaman mengenai saiz perisian merupakan kunci kepada kefahaman produktiviti dan kualiti dalam proses pembangunan sistem. Tanpa metrik pengukuran saiz perisian yang saintifik dan boleh dipercayai, produktiviti (FPs sebulan kerja) atau kualiti (kecacatan per FP) tidak akan dapat dikira. Dengan FP, pemantauan kemajuan dalam produktiviti dan kualiti pembangunan sistem di setiap fasa pembangunan dapat ditambah baik. Apabila perubahan kepada produktiviti dan kualiti dikira dan diplot mengikut masa, organisasi boleh fokus kepada kekuatan dan kelemahan. Fokus kepada kekuatan dan pembetulan serta penambahbaikan kepada kelemahan akan menjadikan proses pembangunan sistem lebih efektif.

Penggunaan pengukuran saiz fungsian sistem aplikasi yang dibangunkan akan menyediakan maklumat-maklumat tambahan yang akan membantu dalam kejayaan dalam pengurusan projek pembangunan sistem aplikasi. Antara maklumat-maklumat tersebut adalah seperti berikut:

a) Produktiviti

- i) Jam per *function point*, membolehkan organisasi membandingkan produktiviti antara projek-projek atau pun membandingkan dengan *standards* industri.
- ii) Produktiviti Keseluruhan, merujuk kepada *function point* per bilangan sumber manusia terlibat (*total work effort*) yang membolehkan organisasi mentadbir dan memantau pelaksanaan projek secara efektif.
- iii) Kadar pengeluaran/serahan (*rate of delivery*), membolehkan pihak pengurusan membuat perancangan dan analisis masa untuk promosi, atau untuk pelaksanaan projek.

b) Kualiti

- i) Saiz fungsian perisian dalam bentuk *function points* boleh digunakan sebagai alat untuk membuat anggaran jadual pelan perancangan projek yang lebih tepat.
- ii) Metrik tahap kesempurnaan projek boleh dikira dengan membandingkan bilangan *function points* berasaskan fungsian yang diminta pengguna berbanding bilangan *function points* berasaskan fungsian yang diserahkan.
- iii) Metrik kadar perubahan kepada skop projek. Organisasi boleh menggunakan maklumat ini untuk membuat justifikasi kepada permohonan bajet tambahan atau perubahan kepada tarikh serahan projek.

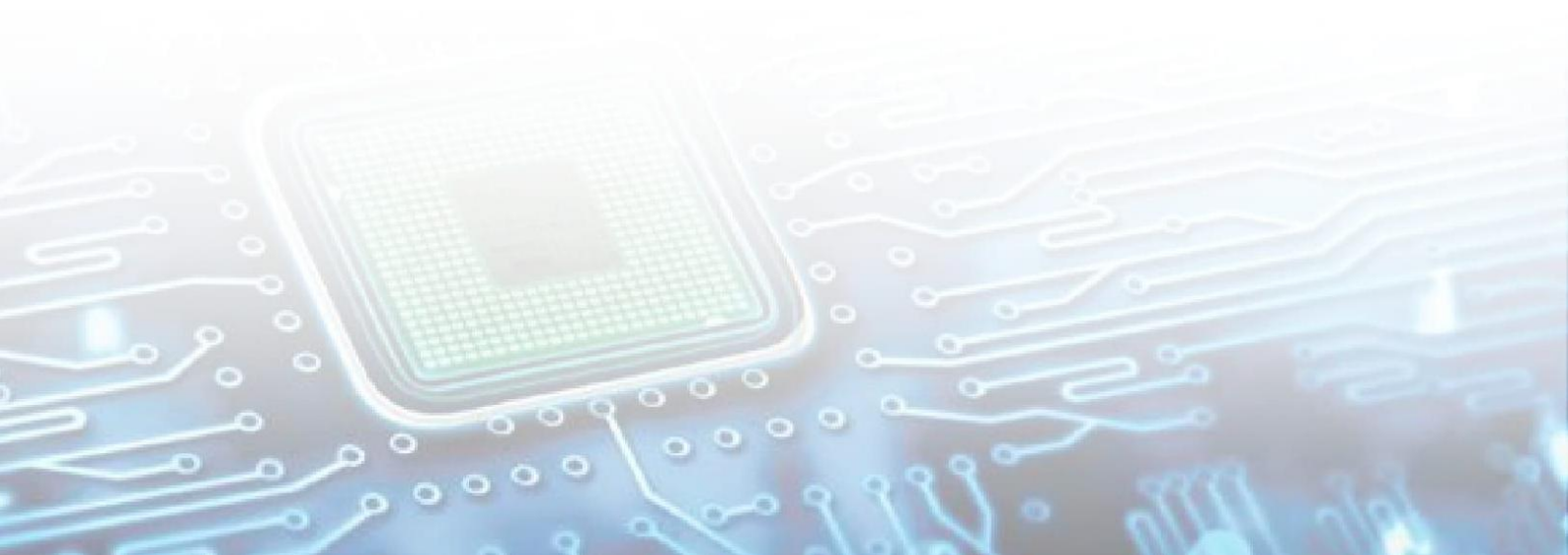
- iv) Kadar kecacatan produk. Ianya boleh digunakan untuk perancangan dan pemantauan di setiap fasa pembangunan perisian bagi memastikan kecacatan yang dikenalpasti dibetulkan sebelum produk dilaksanakan.
- c) Kewangan
- i) Kos per *function point* yang boleh digunakan membuat anggaran kos pembangunan aplikasi dan membantu membuat keputusan dalam pemilihan perisian.
 - ii) Kos pembetulan perisian per *function points*. Ianya digunakan untuk mengukur kos pembetulan perisian selepas dilaksanakan dalam tempoh tertentu.
 - iii) Nilai aset perisian organisasi merupakan satu metrik yang penting untuk menilai aset perisian kepunyaan organisasi.
- d) Penyelenggaraan
- i) *Maintainability* merupakan *effort* dalam bentuk kos penyelenggaraan per *function point* – boleh digunakan untuk membuat perancangan dan pemantauan kos penyelenggaraan aplikasi.
 - ii) *Reliability* ialah bilangan kegagalan aplikasi berbanding dengan saiz fungsinya, boleh digunakan untuk mengukur tahap kebolehpercayaan aplikasi.
 - iii) Bilangan sumber manusia per *function points* untuk membuat penyelenggaraan.



BAB:02

FASA PERMULAAN PROJEK

Bab ini menerangkan aktiviti-aktiviti bagi penyediaan Pelan Perancangan Pembangunan Sistem dan pelaksanaan kajian keperluan bisnes yang merangkumi Pemodelan Fungsi dan Proses Bisnes.



2 FASA PERMULAAN PROJEK

Fasa Permulaan Projek adalah fasa perancangan kepada pasukan pembangun sistem untuk mengenalpasti skop pembangunan, pendekatan pelaksanaan dan aspek-aspek lain yang perlu diberi perhatian sepanjang aktiviti pembangunan dilaksanakan.

Fasa Permulaan Projek memberikan penekanan kepada komunikasi antara pemegang taruh dan organisasi/pasukan yang bertanggungjawab membangunkan sesuatu projek. Penglibatan dan komitmen kedua-dua pihak ini juga ditekankan dalam menjayakan projek.

Fasa ini dibahagikan kepada 2 aktiviti utama iaitu:

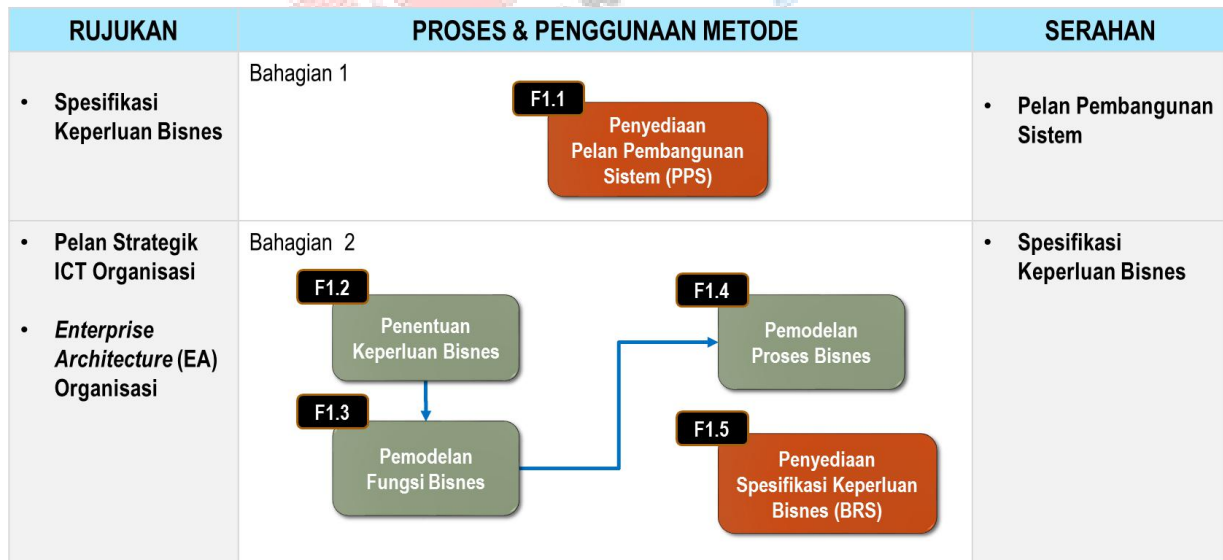
a) Perancangan Pembangunan Sistem

Aktiviti ini membincangkan penyediaan pelan perancangan pembangunan sistem aplikasi dari aspek keperluan sumber projek seperti kos, masa dan modal insan.

b) Kajian Keperluan Bisnes

Aktiviti ini membincangkan pendekatan bagi kajian keperluan bisnes dan mendokumentasikan skop bisnes bagi tujuan pembangunan sistem aplikasi.

2.1 GAMBARAN KESELURUHAN



Rajah 14 : Gambaran Keseluruhan Fasa I – Permulaan Projek

2.2 PERANCANGAN PEMBANGUNAN SISTEM

2.2.1 PENGENALAN

Perancangan pembangunan sistem perlu dilaksanakan secara menyeluruh dan terperinci bagi membolehkan pasukan pembangunan dan organisasi mengurus, melaksana dan memantau pembangunan sistem aplikasi. Perancangan pembangunan perlu didokumenkan dalam Pelan Pembangunan Sistem Aplikasi. Proses penyediaan pelan ini sewajarnya bermula setelah pasukan mengenalpasti skop keperluan bisnes, bagi memastikan perancangan dapat dilakukan dengan tepat. Pelan ini juga perlu dibangunkan sejajar dengan pelan perancangan projek ICT yang berkaitan.

Dokumen Rujukan kepada aktiviti Perancangan Pembangunan adalah **D02 Spesifikasi Keperluan Bisnes**, sekiranya dokumen ini dibangunkan awal.

Dokumen Serahan kepada aktiviti Perancangan Pembangunan adalah **D01 Pelan Pembangunan Sistem**.

2.2.2 PENGLIBATAN PEMEGANG TARUH

Pemegang Taruh utama yang akan terlibat dalam perancangan adalah ketua pasukan pembangunan sistem dan pemilik sistem aplikasi yang dilantik berdasarkan sumber-sumber yang telah ditetapkan. Oleh kerana perancangan meliputi pendekatan pelaksanaan sistem, adalah penting perancangan mendapat komitmen dan perhatian kepada semua ahli pasukan dan persetujuan oleh pemilik sistem.

Cadangan penglibatan kategori pemegang taruh adalah seperti berikut:

- a) Pemilik Sistem.
- b) Ketua Pasukan Pembangunan Sistem.
- c) Ketua SME.

2.2.3 FAKTOR KEJAYAAN

Untuk memastikan kejayaan perancangan, faktor kejayaan utama yang perlu dipertimbangkan sebelum dan semasa aktiviti dilaksanakan adalah seperti berikut:

- a) Menetapkan pemilik sistem.
- b) Memperolehi komitmen daripada pemegang taruh utama.
- c) Kes bisnes untuk pembangunan sistem diluluskan.
- d) Pasukan pembangunan yang mencukupi.

2.2.4 PENYEDIAAN PELAN PEMBANGUNAN SISTEM [F1.1]

KETERANGAN

Pelan Perancangan Sistem merupakan perincian strategi dan pendekatan yang akan digunakan sebagai panduan oleh organisasi dan pasukan projek. Rujuk kepada **D01 Pelan Pembangunan Sistem** untuk melihat format dokumen yang terlibat di mana pengisian kandungan-kandungannya adalah seperti langkah-langkah di bawah.

OBJEKTIF

- Menetapkan tempoh berdasarkan peruntukan sumber yang diberikan
- Sebagai panduan dalam mengendalikan pembangunan sistem
- Panduan dalam pemantuan projek pembangunan

LANGKAH-LANGKAH

Langkah 1 : Tetapkan Pengenalan Projek

Pengenalan projek akan menerangkan gambaran ringkas mengenai projek secara keseluruhannya yang terdiri daripada tujuan projek, skop projek dan serahan projek.

a) Tujuan Projek

Terangkan tujuan projek yang menentukan jangkaan hasil projek yang bersama-sama dengan kekangan keperluan pembangunan (persekitaran, standard, . Tujuan boleh juga diekstrak daripada Cadangan Projek yang didokumenkan dalam D02 Spesifikasi Keperluan Bisnes yang dihasilkan. Utamakan tujuan projek berdasarkan aspek berikut:

- i) Matlamat Fungsian organisasi
- ii) Matlamat Bisnes (kecekapan & faedah)
- iii) Matlamat Kualiti (kualiti produk atau perkhidmatan)

Terangkan juga rasional yang menyebabkan projek pembangunan perlu dilaksanakan seperti keperluan baru, perluasan (*roll-out*) atau memperbaiki proses/perkhidmatan yang disediakan oleh organisasi. Pernyataan ringkas mengenai masalah perkakasan, masalah perisian dan sebagainya disenaraikan bagi menguatkan lagi tujuan projek.

Contoh tujuan projek:

Tujuan projek ini adalah untuk membangunkan aplikasi telefon pintar bagi mencari lokasi tempat makan di Malaysia. Aplikasi ini dapat membantu rakyat atau pelancong semasa percutian di Malaysia. Aplikasi ini akan menjimat masa carian, menyediakan pelbagai pilihan lokasi dengan pelbagai jenis makanan yang terdapat di Malaysia.

b) Skop Projek

Jelaskan apakah yang akan (atau tidak akan) dihasilkan oleh projek bagi mengelak perubahan pada masa depan dalam mencapai tujuan atau matlamat projek. Skop projek perlulah dipersetujui dan akan dijadikan sebagai sempadan (*boundary*) untuk pelaksanaan pembangunan aplikasi dan asas untuk mengukur kejayaan projek.

Contoh Skop projek:

Antara skop projek yang telah dikenalpasti bagi mencapai tujuan projek adalah:

- i) Merangkumi semua lokasi restoran/tempat makan diseluruh Malaysia yang didaftarkan dibawah agensi pelancongan.
- ii) Akses aplikasi melalui semua jenis telefon pintar (Android dan IOS)

c) Serahan Projek

Menyenaraikan jangkaan serahan projek mengikut fasa pembangunan sistem yang berkaitan. Butiran serahan yang perlu meliputi nama serahan, tarikh serahan, kuantiti serahan, penyedia, pengesah dan pelulus serahan. Setiap dokumen serahan perlu mendapat pengesahan atau kelulusan daripada pemilik projek bagi memastikan kualiti dan ketepatan maklumat yang disediakan.

Jadual di bawah sebagai panduan dalam penyenaian serahan projek :

Fasa (SDLC)	Nama & No. Rujukan Serahan	Tarikh serahan	Kuantiti	Nama/jawatan Penyedia	Jawatan Penyemak	Jawatan Pelulus

Contoh Serahan Pembangunan Sistem Aplikasi adalah:

Fasa Analisis	i) Spesifikasi keperluan sistem
Fasa Pembangunan	i) Sistem Aplikasi ii) Dokumentasi Pangkalan Data iii) Dokumentasi Kod Sumber
Fasa Pelaksanaan	i) Manual Pengguna ii) Manual Operasi iii) Latihan Teknikal dan Latihan Pengguna

Langkah 2 : Tetapkan Maklumat Pengendalian Projek

Maklumat bagi pengendalian projek seperti model proses, struktur organisasi dan peranan merupakan input penting dalam penyediaan Pelan Pembangunan Sistem. Kerangka bagi maklumat tersebut adalah seperti berikut:

a) Model Proses

Menerangkan proses pembangunan sistem yang akan digunakan yang mengambilkira kitaran hayat pembangunan sistem (SDLC). Model proses perlu mengambilkira:

i) Proses Pembangunan

Nyata dan jelaskan proses pembangunan sistem aplikasi, kenalpasti juga modul/release/fasa aplikasi yang dirancang, objektif dan aktiviti pembangunan yang akan dilaksanakan di setiap modul/release/fasa.

ii) Metodologi Pembangunan

Nyata dan jelaskan kaedah pembangunan sistem yang akan digunakan (*waterfall, agile, spiral* dll). Ini termasuklah keterangan manual, *tools*/alat dan prosedur yang akan digunakan dalam menyokong kaedah yang digunakan.

iii) Standard

Nyatakan standard yang akan digunakan untuk mempersembahkan keperluan, rekabentuk, coding, kes pengujian, prosedur pengujian dan keputusan pengujian penerimaan.

iv) *Reusable Product*

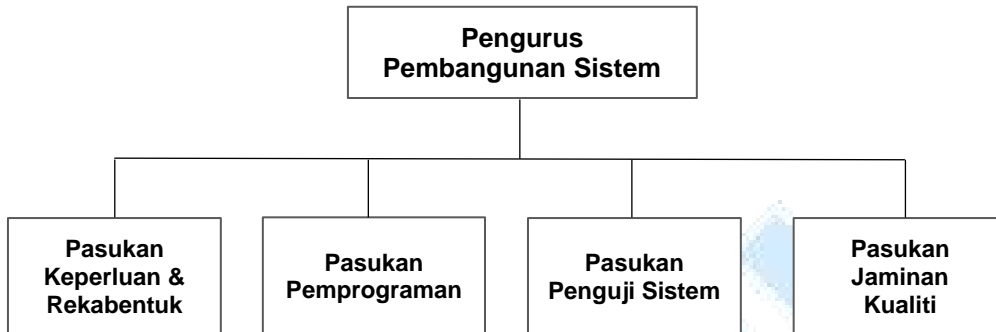
Terangkan pendekatan yang digunakan untuk mengenalpasti, menilai dan melapor peluang untuk customize atau mengintegrasikan dengan aplikasi sedia ada.

b) Struktur Organisasi Pasukan

Pasukan Pembangunan sistem adalah salah satu fungsi/komponen dalam Struktur Organisasi Projek ICT. Pelan Pembangunan Sistem yang dibangunkan hanya memperincikan pasukan pembangunan sistem. Kenalpasti pasukan kerja yang terlibat dalam pasukan pembangunan sistem. Pasukan ini sekurang-kurangnya terdiri daripada sub-pasukan seperti berikut:

- i) Pasukan Keperluan dan Analisis
- ii) Pasukan Rekabentuk (Pangkalan Data & Fungsian)
- iii) Pasukan Migrasi/Integrasi (Jika berkaitan)
- iv) Pasukan Pengujian
- v) Pasukan Jaminan Kualiti

Sediakan rajah struktur organisasi pasukan projek digambarkan dengan menggunakan sama ada carta organisasi atau rajah matriks yang dapat menggambarkan bidang kuasa, tanggungjawab dan komunikasi di dalam projek.



Rajah 15 : Contoh Struktur Pasukan Pembangunan Sistem

c) Tetapkan Peranan dan Tanggungjawab

Pengurus Projek Pembangunan perlu dilantik sebagai ketua dan bertanggungjawab dalam merancang dan mengurus pelaksanaan aktiviti pembangunan sistem. Manakala pelaksanaan akan/boleh diagihkan kepada sub-pasukan mengikut fungsi. Setiap sub-pasukan boleh terdiri daripada ketua pasukan dan ahli pelaksana. Setiap ahli pasukan hendaklah dikenalpasti perlu mengambilkira tahap kemahiran/kepakaran dan tempoh masa yang diperuntukkan dalam projek untuk anggaran tempoh dan kos projek.

Contoh: Jadual Peranan dan Tanggungjawab

Jadual 5 : Contoh Jadual Penerangan Keahlian Pasukan Pembangunan

Fungsi	Nama Ketua	Nama Ahli	Tempoh Penglibatan	Tanggungjawab
Pengurus Pembangunan Sistem				
Pasukan Keperluan dan Rekabentuk				
Pasukan Pemrograman				
Pasukan Penguji				
Pasukan QA				
Pasukan SME				

Langkah 3 : Tetapkan Proses Pengurusan

Proses pengurusan merupakan penerangan faktor yang perlu diambil kira bagi melancarkan pengurusan pelaksanaan projek. Faktor tersebut adalah Andaian, Kebergantungan dan Kekangan; Risiko; Tahap Kebarangkalian Risiko dan Tahap Impak; serta Pemantauan dan Kawalan.

a) Andaian, Kebergantungan dan Kekangan

Menjelaskan andaian sebagai asas dalam pembangunan pelan projek, kebergantungan ke atas pelan projek dan kekangan terhadap pelan projek seperti tempoh projek, peruntukan sumber, kualiti dan fungsi.

- i) Andaian Projek adalah peristiwa atau keadaan yang dijangka berlaku sepanjang tempoh projek pembangunan sistem aplikasi dan memberi kesan positif atau negatif terhadap objektif projek.
- ii) Kebergantungan projek adalah kebergantungan kejayaan projek dengan aktiviti luaran yang lain, ianya juga perlu dikenalpasti bagi memastikan pelaksanaan pembangunan perisian berjalan lancar.
- iii) Kekangan projek adalah apa-apa yang menghadkan tindakan projek pembangunan sistem aplikasi. Kekangan perlu mengambilkira 3 aspek iaitu masa, sumber dan kualiti kerana ketiga-tiga aspek ini yang akan memandu kejayaan projek.

b) Senaraikan Risiko

Risiko projek adalah peristiwa atau keadaan yang tidak pasti, jika ia berlaku, ia akan memberi kesan ke atas tujuan/objektif projek. Risiko dalam pembangunan sistem aplikasi berpunca dari sumber dalaman dan luaran.

- i) Risiko dalaman yang dimaksudkan adalah:
 - Risiko yang berkaitan dengan objektif projek antara pengurus projek, ahli pasukan dan pemilik projek. Sekiranya objektif projek tidak ditakrifkan dengan jelas pada permulaan projek, adalah sangat sukar bagi projek itu mencapai keputusan
 - Risiko yang berkaitan dengan takrifan saiz projek. Takrifan saiz yang salah atau tidak tepat mengakibatkan salah arah tuju pembangunan projek, seperti perubahan tempoh masa pembangunan projek, pertambahan kos dan lain-lain.
 - Risiko yang berkaitan dengan kompleksiti projek. Sekiranya pasukan telah dapat memahami kompleksiti projek dengan tepat mengakibatkan tambahan masa, sumber bagi memahami bsines yang diperlu dibangunkan.
 - Risiko yang berkaitan dengan definisi pelan projek. Sekiranya pelan projek tidak mengambilkira kompleksiti, saiz, kemahiran dan pengalaman ahli

pasukan dalam fungsi masing-masing akan mengakibatkan risiko kegagalan projek

- ii) Risiko luaran pula adalah risiko diluar kawalan pasukan projek. Risiko berkaitan dengan politik, perundangan dan berkaitan dengannya. Masalah teknikal diluar kawalan juga salah satu risiko luaran. Sekiranya projek adalah bergantung kepada teknologi yang masih belum siap, projek ini potensi berisiko, maka penyelesaian alternatif perlu dikenalpasti.

c) Tahap Kebarangkalian Risiko dan Tahap Impak

Pengurusan risiko perlu diberi tumpuan untuk mengenal pasti, menilai risiko dan menguruskan risiko tersebut untuk meminimumkan kesan ke atas projek.

Penilaian risiko perlu dibuat bagi mengenalpasti impak risiko kepada projek. Tahap kebarangkalian risiko (exposure) dinilai berdasarkan formula berikut :

$$\text{Tahap risiko (exposure)} = \text{Kebarangkalian (probability / likelihood)} \times \text{Impak atau kesan risiko (impact)}$$

Kedua-dua kebarangkalian dan impak akan dinilai mengikut skala yang ditetapkan.

Jadual 6 : Skala Impak dan Tahap Risiko

		Impak				
		Tiada Kesan	Kecil	Sederhana	Besar	Sangat Besar
Kebarangkalian		1	2	3	4	5
5	Hampir pasti	5	10	15	20	25
4	Kemungkinan tinggi	4	8	12	16	20
3	Ada Kemungkinan	3	6	9	12	15
2	Kemungkinan rendah	2	4	6	8	10
1	Jarang	1	2	3	4	5

	Tahap Risiko	Skala Tahap Risiko	Penerangan
	Ekstrem	13-25	Risiko sangat tinggi, pelan tindakan terperinci diperlukan
	Tinggi	8-12	Risiko tinggi, diberi perhatian oleh pengurusan projek
	Sederhana	4-7	Risiko sederhana, diuruskan dan diberi perhatian oleh pihak yang dipertanggungjawab
	Rendah	1-3	Risiko Rendah, diuruskan mengikut prosidur sedia ada

Sumber : Overview Pengurusan Risiko Sektor Awam, MAMPU, JPM

d) Pemantauan dan Kawalan

Pemantauan dan kawalan merupakan faktor penting dalam pengurusan projek. Kepentingan pemantauan dan kawalan dalam projek adalah untuk memastikan projek mencapai kemajuan seperti dirancang dan tindakan yang diambil apabila perubahan kepada pelan asal diperlukan. Bagi memastikan pemantauan dan kawalan dilaksanakan secara berkesan, projek perlu menetapkan penggunaan mekanisma atau pendekatan yang bersesuaian.

Mekanisma pemantauan ini perlu mengambil kira:

- i) Struktur pelaporan
- ii) Kaedah dan kekerapan pelaporan
- iii) Format templat pelaporan

Proses dalam melaksanakan pemantauan dan kawalan adalah sebagaimana berikut:



Rajah 16 : Proses Pemantauan Dan Kawalan Projek

Merujuk kepada Ir. Wan Ibrahim Wan Yusoff, buku bertajuk 77 Tips Projek Berjaya perkara yang perlu dipantau atau dikawal adalah:

- i) Kemajuan mengikut pelan perancangan mengawal dari segi pelaksanaan status aktiviti, mengenalpasti isu/masalah dan tindakan pembedahan
- ii) Peruntukan, kawal dari aspek berapa banyak telah dibelanja dan yang tinggal, dan adakah baki kos yang tinggal bagi menyiapkan kerja
- iii) Skop dikawal dari aspek pelaksanaan aktiviti masih dalam skop ataupun diluar skop dan juga kenalpasti perubahan yang melibatkan tambahan skop
- iv) Kualiti, pastikan kerja menepati spesifikasi, sekira wujudnya di luar spesifikasi, apakah tindakan perlu diperbetulkan, dipantau dan perlu dikawal.

Langkah 4 : Menetapkan Proses Teknikal

Proses Teknikal merupakan penerangan faktor yang perlu diambilkira bagi melancarkan pelaksanaan pembangunan sistem aplikasi. Faktor tersebut adalah Pendekatan, teknik dan alat bantu; Dokumen aplikasi dan Dokumentasi Fungsi sokongan.

a) Pendekatan, teknik dan alat bantu

Nyatakan pendekatan, teknik dan alat bantu akan digunakan dalam proses pembangunan sistem aplikasi. Merujuk kepada contrux.com pendekatan, teknik dan alat bantu perlulah mengambilkira item berikut:

i) Persekitaran sistem

Nyatakan apakah perkakasan, sistem pengoperasian dan pengurusan pangkalan data yang akan digunakan dan spesifikasi yang diperlukan bagi persekitaran pembangunan, pengujian dan produksi.

ii) *Tools* Perisian

Nyatakan *tools* yang akan digunakan bagi tujuan analisis, rekabentuk, kawalan kod sumber, pembangunan sistem, debugging aids, defect tracking dan lain-lain.

iii) Metodologi pembangunan

Nyatakan pendekatan dalam kajian keperluan sistem, metodologi dalam rekabentuk dan penggunaan notasi, bahasa pengaturcaraan, piawaian pengaturcaraan, piawaian dokumentasi, prosedur integrasi sistem, dan sebagainya.

iv) Jaminan kualiti

Nyatakan kaedah jaminan kualiti serahan bermula daripada semakan kesinambungan keperluan dan rekabentuk, ujian unit, menyemak logik kod (debugger), ujian sistem, ujian regrasi dan ujian bukan fungsian lain yang berkenaan.

b) Dokumen Aplikasi

Senaraikan semua dokumentasi berkaitan pembangunan perisian yang perlu disediakan termasuk bila sasaran siap, disemak dan ditandatangani. Antara dokumen yang berkaitan adalah:

- i) Spesifikasi Keperluan Bisnes (BRS)
- ii) Spesifikasi Keperluan Sistem (SRS)
- iii) Spesifikasi Rekabentuk Sistem (SRS)
- iv) Pelan Induk Pengujian
- v) Pelan Ujian (UAT & PAT)
- vi) Laporan Ujian Penerimaan

- vii) Pelan Integrasi Sistem
- viii) Pelan Migrasi Sistem
- ix) Manual Pengguna

c) Dokumen Sokongan

Nyatakan dokumentasi rujukan lain yang menyokong usaha kejayaan pembangunan sistem. Antaranya adalah :

- i) Pengurusan Konfigurasi
Dokumen yang akan digunakan bagi merekod dan mengawal semua versi serahan projek.
- ii) Pengurusan Risiko
Dokumen yang akan digunakan bagi merekod dan mengawal risiko yang telah dikenal pasti.
- iii) Pengurusan Perubahan
Dokumen yang akan digunakan bagi merekod dan mengawal perubahan skop dan komponen projek.
- iv) Pelan Jaminan Kualiti
Dokumen yang akan digunakan bagi merekod dan mengawal kualiti produk.
- v) Senarai Semak Instalasi Sistem (*Deployment*)
Dokumen yang akan digunakan bagi merekod semua proses pengujian dan produksi.
- vi) Pelan Latihan Pengguna
Dokumen yang akan digunakan bagi merekod perancangan latihan sistem kepada teknikal dan pengguna akhir sistem.

Langkah 5 : Menetapkan Pakej Kerja, Jadual Dan Peruntukan

Menetapkan pakej kerja, jadual dan peruntukan secara jelas dan terperinci adalah penting dalam kerja-kerja pemantauan dan kawalan pembangunan sistem secara berkesan.

a) Pakej Kerja

Terangkan pakej kerja yang mesti diselesaikan untuk melengkapkan sesuatu sistem aplikasi. Kenal pasti setiap pakej kerja dan gambarkan pecahan pakej kerja melalui struktur perincian kerja (WBS).

<u>Pakej kerja</u>	<u>Aktiviti utama</u>	<u>Sub aktiviti</u>
1. Kajian Keperluan	<u>1.1 <Aktiviti></u> dll	1.1.1 <Sub-aktiviti> 1.1.2....
2. Rekabentuk Sistem	<u>2.1 <Aktiviti></u> dll	2.1.1 <Sub-aktiviti> 2.1.2...
n. Pembangunan Sistem	<u>n.1 <Name></u> dll	n.1.1 <Sub-aktiviti> n.1.2...

Rajah 17 : Contoh Perincian Kerja (WBS)

b) Pakej Kerja dan Kebergantungan

Kebergantungan menerangkan hubungan antara dua atau lebih aktiviti-aktiviti dalam pakej kerja yang sama atau dari pakej kerja yang lain. Oleh itu, pengurus pembangunan sistem perlu mengenal pasti kebergantungan antara aktiviti-aktiviti atau pakej kerja untuk merangka jadual projek yang tepat. Sebagai contoh, pakej kerja bagi membangunkan modul pengurusan rekod secara elektronik, pakej kerja tersebut bergantung kepada kesediaan pakej kerja pengkelasan rekod yang perlu diselesaikan terlebih dahulu.

c) Anggaran Keperluan Sumber

Anggaran sumber adalah dijumlah dari setiap aktiviti, pakej kerja dan keseluruhan projek. Kenalpasti anggaran usaha ahli pasukan dan anggaran kos bagi sumber-sumber lain seperti perkakasan dan perisian.

Contoh: Anggaran sumber**Jadual 7 : Contoh Anggaran Jumlah Projek**

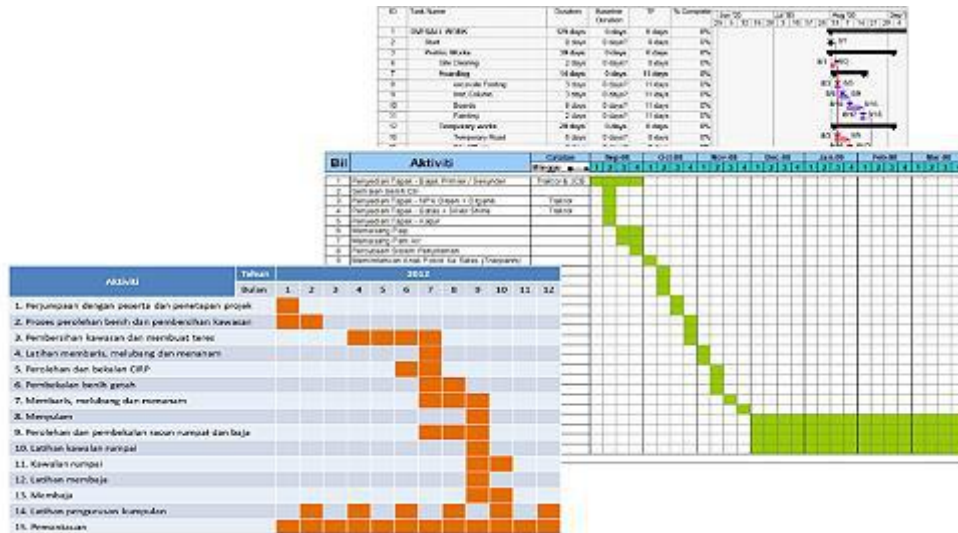
<u>Pakej kerja</u>	<u>Aktiviti utama</u>	<u>Sub aktiviti</u>	<u>Ahli Pasukan Orang/sehari</u>	<u>Kos lain</u>
1. Kajian Keperluan	<u>1.1 <Aktiviti></u>	1.1.1 <Sub- aktiviti>		
		1.1.2....		
	<u>1.2...</u>			
2. Rekabentuk Sistem	<u>2.1 <Aktiviti></u>	2.1.1 <Sub- aktiviti>		
		2.1.2...		
	dll			
n. Pembangunan Sistem	<u>n.1 <Name></u>	n.1.1 <Sub- aktiviti>		
		n.1.2...		
	dll			
Anggaran Jumlah Projek				

d) Peruntukan Kos

Sumber dan jumlah peruntukan yang diperlukan bagi menjalankan projek perlu dinyatakan. Pembahagian dan perincian peruntukan mengikut aktiviti – aktiviti utama seperti pembangunan aplikasi, perkakasan dan penyediaan dokumen kajian keperluan perlu dinyatakan. Keperluan peruntukan boleh dibentangkan dalam bentuk jadual dan / atau menggunakan satu atau lebih angka (seperti histogram).

e) Jadual Perancangan

Tentukan jadual projek menggunakan carta bar atau carta Gantt, mengenal pasti tarikh pencapaian utama, pencapaian, input luaran untuk projek (dan sebarang kebergantungan luaran yang lain), tempoh setiap pakej kerja dipecahkan kepada kepada aktiviti dan sub aktiviti.



Rajah 18 : Contoh Jadual Perancangan Projek

Dua kaedah untuk mengira anggaran masa yang diperlukan untuk pembangunan sistem iaitu:

- i) Berasaskan masa yang diperuntukkan dalam fasa perancangan untuk membuat anggaran masa yang diperlukan dalam fasa berikutnya;

Mengikut standard industri bagi sistem aplikasi yang tipikal, usaha yang diperuntukkan dalam fasa perancangan adalah sebanyak 15% , 20% untuk fasa analisis, 30% untuk fasa rekabentuk dan 35% untuk fasa pelaksanaan. Walau bagaimanapun, peratusan ini lebih tepat sekiranya kita mempunyai rekod-rekod dari pengalaman lepas yang boleh dijadikan asas kepada pengiraan anggaran masa yang diperlukan.

- ii) *Function Point Analysis*

Pengiraan anggaran masa secara saintifik menggunakan pendekatan analisis function point lebih tepat dan lebih dipercayai (*reliable*). Berdasarkan keperluan bisnes, analyst akan membuat pengiraan awal saiz sistem aplikasi. Saiz sistem aplikasi dalam ukuran *function point* ini akan ditukar kepada jumlah usaha diperlukan untuk membangunkan sistem dalam bentuk bilangan person-months. Anggaran usaha ini kemudiannya diterjemahkan kepada anggaran jadual masa dalam bentuk bilangan bulan daripada awal hingga tamat projek.

Langkah 6 : Komponen Tambahan

Komponen tambahan adalah perkara yang ada hubung kait dalam pelaksanaan projek. Contoh komponen tambahan adalah seperti pelan keselamatan, pelan latihan, pelan perolehan perkakasan dan perisian, pelan-pelan fasiliti, pelan pemasangan, pelan penyelenggaraan dan komponen-komponen lain yang berkaitan.

Langkah 7 : Lampiran

Pelan Pembangunan Sistem akan mempunyai dokumen sokongan yang dirujuk dalam pelaksanaan projek. Dokumen ini boleh dijadikan lampiran kepada pelan pembangunan sistem. Contoh dokumen lampiran adalah minit mesyuarat, surat-surat dan sebagainya.

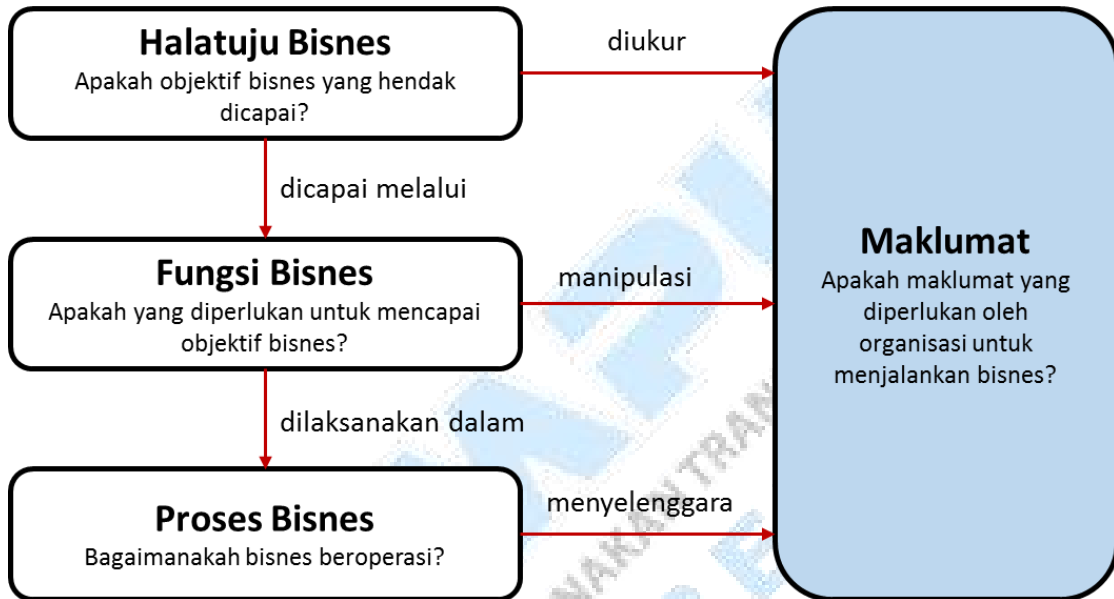
RUJUKAN

1. Gary, B. S. & Harry, J. R. (2012). System Analysis and Design. Ninth Edition. Boston: Course Technology
2. Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems Analysis and Design. Fifth Edition. USA: John Wiley & Sons, Inc.
3. Wiegers, K., E. & Joy, B. (2013). Software Requirements. Third Edition. Washington: Microsoft Press
4. Risks in Information Systems Development Projects, Ozren Đurković Lazar Raković, Management Information Systems, Vol. 4 (2009)
5. Ir. Wan Ibrahim Wan Yusoff, 77 tips kejayaan projek (www.pengurusprojek.com) "SW Development Plan" template, www.computing.dcu.ie/~ltuohey/CA305_2007.../ExampleDevPlanTemplate.doc

2.3 KAJIAN KEPERLUAN BISNES

2.3.1 PENGENALAN

Kajian keperluan bisnes adalah merupakan aktiviti asas yang perlu dilaksanakan untuk memahami halatuju dan objektif organisasi, mengenalpasti keperluan bisnes dan skop yang perlu diliputi bagi tujuan pembangunan sistem aplikasi. Kajian keperluan ini adalah berpandukan model bisnes organisasi seperti dalam **Rajah 19**.



Rajah 19 : Model Bisnes Organisasi

Aktiviti utama yang terlibat dalam kajian keperluan bisnes adalah seperti berikut:

- Penentuan Keperluan Bisnes.
- Pemodelan Fungsian Bisnes.
- Pemodelan Proses Bisnes.

Dokumen Rujukan kepada aktiviti kajian keperluan bisnes adalah Permintaan daripada pemegang taruh samada melalui Pelan Strategik ICT atau *Enterprise Architecture* (EA) organisasi, mesyuarat, kertas kerja, kelulusan Jawatankuasa Pemandu Organisasi atau Jawatankuasa di peringkat Sektor Awam.

Dokumen Serahan kepada aktiviti kajian keperluan bisnes adalah **D02 Spesifikasi Keperluan Bisnes**.

2.3.2 PENGLIBATAN PEMEGANG TARUH

Pemegang taruh utama yang terlibat dalam kajian keperluan adalah pemilik prosidur atau proses, pengurusan atasan dan pengguna. Memandangkan kajian keperluan bisnes merupakan aktiviti asas dan utama dalam pembangunan sistem, maka penglibatan SME yang berautoriti dan kompeten sangat diperlukan. Oleh demikian, pemilihan SME amat penting dalam memastikan kejayaan kajian keperluan bisnes. Pemegang taruh boleh terdiri dari pada kategori berikut:

- a) Pemilik sistem.
- b) Pengurusan atasan.
- c) Pembiaya projek.
- d) Pengguna sistem.
- e) Organisasi yang akan berinteraksi dengan sistem.

2.3.3 FAKTOR KEJAYAAN

Untuk memastikan aktiviti kajian keperluan bisnes berjaya dilaksanakan, faktor kejayaan utama yang perlu dipertimbangkan sebelum dan semasa aktiviti dilaksanakan adalah seperti berikut:

- a) SME yang dilantik arif dengan keperluan bisnes dan penglibatan yang berterusan.
- b) Pasukan kajian mempunyai kemahiran dalam berkomunikasi.
- c) Keperluan bisnes didokumenkan dengan tepat dan mendapat pengesahan pemilik.

2.3.4 PENENTUAN KEPERLUAN BISNES [F1.2]

KETERANGAN

Aktiviti ini adalah bagi memastikan sistem yang diperlukan dapat memenuhi keperluan bisnes organisasi. Maklumat berkaitan keperluan bisnes dapat diperolehi daripada pelbagai sumber seperti nota temuduga, peraturan dan prosedur bertulis, dokumentasi sistem semasa, laporan, borang dan aliran kerja yang sedia ada. Semua maklumat tersebut dikumpulkan untuk memahami keperluan fungsian selanjutnya.

OBJEKTIF

- Mengumpul maklumat berkaitan keperluan bisnes
- Mengenalpasti dan memahami halatuju bisnes yang mendorong kepada apa dan bagaimana proses bisnes dilakukan

LANGKAH-LANGKAH

Langkah 1 : Fomulasi Kes Bisnes

- a) Kenalpasti kes bisnes bagi keperluan pembangunan sistem. Kes bisnes perlu menyatakan sebab-sebab yang membawa kepada keperluan pembangunan sistem.
- b) Hasrat yang telah dikemukakan oleh pemegang taruh akan dipindahkan kepada Apendiks 1 Borang Permohonan Pembangunan Sistem dan digunakan sebagai input utama kepada pembangunan sistem.
- c) Hasrat pemegang taruh mungkin berpunca dari persekitaran bisnes semasa yang berkaitan dengan faktor berikut:
 - i) Perubahan dalam proses bisnes;
 - ii) Perubahan dalam teknologi;
 - iii) Perubahan dalam undang-undang, akta atau prosedur; dan
 - iv) Pematuhan kepada piawaian baru.

Langkah 2 : Fahami Keperluan Bisnes

- a) Fahami keperluan bisnes akan menjawab kepada enam (6) soalan asas kepada sistem baru yang hendak dibangunkan iaitu:
 - i) Apakah objektif projek ini?
 - ii) Apakah perubahan yang perlu dilakukan ke atas bisnes untuk mencapai objektif projek ini?
 - iii) Bila dan apakah urutan proses berlaku?
 - iv) Di mana proses/fungsi yang dilakukan?
 - v) Siapakah yang menjalankan proses/fungsi yang dikenalpasti.

- vi) Apakah dokumen, borang, e-mel, laporan, prosedur yang terlibat dalam proses tersebut?

Langkah 3 : Lakukan Pemilihan SME

Pemilihan SME dan penglibatan mereka amat penting bagi menjayakan kajian keperluan bisnes. SME yang dipilih perlu mempunyai ciri-ciri berikut:

- a) Berpengetahuan dalam proses bisnes;
- b) Merupakan personel utama dalam proses bisnes;
- c) Berkemampuan untuk menterjemah dan melaksanakan strategi berkaitan;
- d) Mempunyai visi mengenai halatuju bisnes; dan
- e) Menyokong objektif projek

Langkah 4 : Pilih Teknik Pengumpulan Maklumat

Terdapat pelbagai teknik untuk pengumpulan maklumat dan mengenalpasti keperluan bagi pembangunan sistem iaitu:

Jadual 8 : Teknik Pengumpulan Maklumat

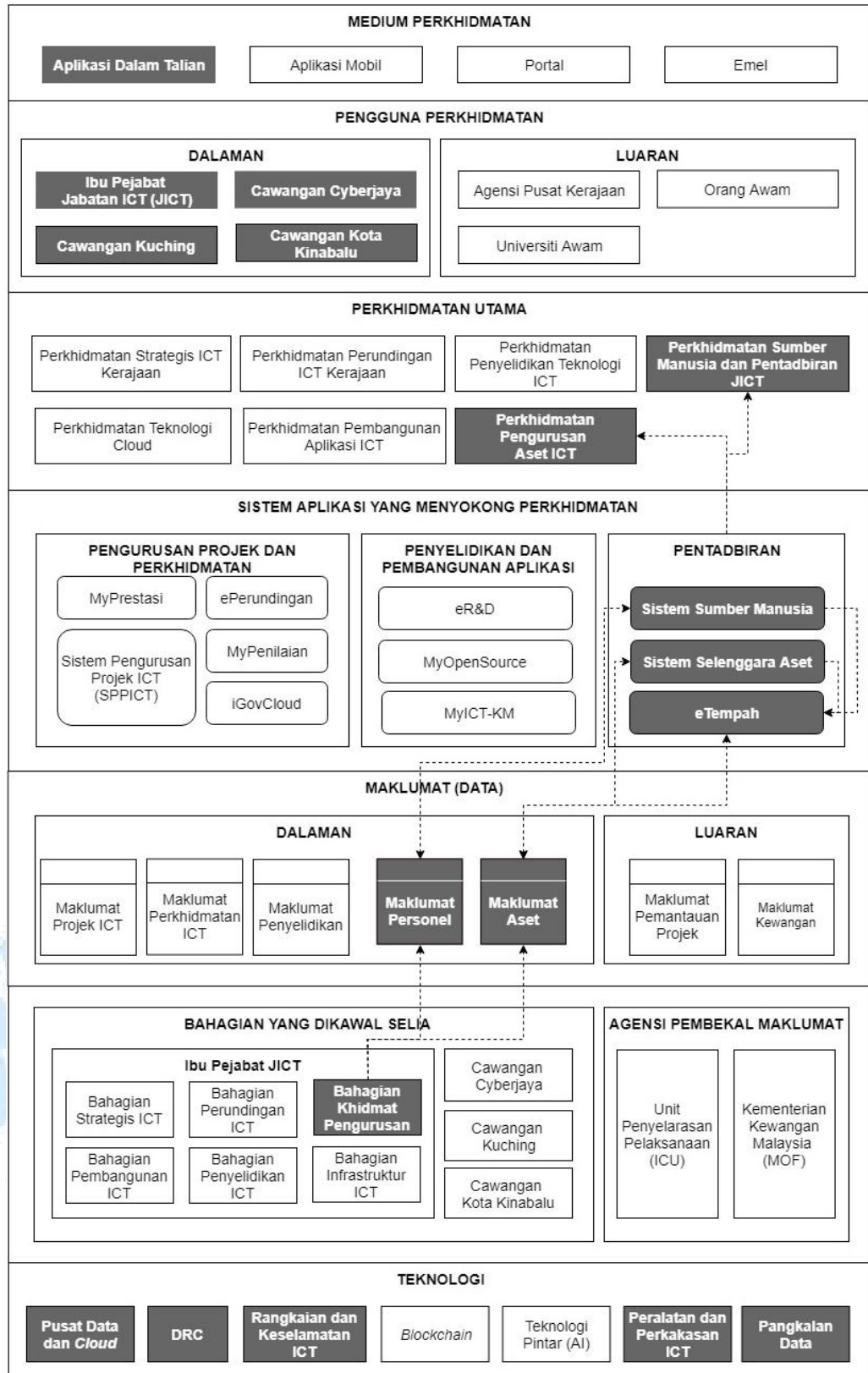
Brainstorming	<ol style="list-style-type: none"> a) Sesi percambahan fikiran bagi mencetuskan idea/strategi; b) Boleh menghasilkan berbagai pilihan/cadangan dengan cepat; c) Dipengaruhi oleh faktor pengalaman dan kreativiti kumpulan; d) Kaedah yang baik bagi mengumpulkan idea; dan e) Dijalankan dalam persekitaran yang tidak terikat dengan peraturan.
Focus Group	<ol style="list-style-type: none"> a) Satu kumpulan persekitaran interaktif untuk mendapatkan idea tentang peluang produk / sistem; b) Dipandu oleh moderator; c) Boleh dijalankan pada bila-bila masa dalam kitaran hayat; dan d) Sesuai untuk mengkaji keperluan baru dan membuat penilaian ke atas produk.
Pemerhatian	<ol style="list-style-type: none"> a) Melihat bagaimana sistem membantu pengguna dalam melaksanakan kerja seharian dan menilai samada sistem perlu diganti atau dinaik taraf; b) Membantu menyediakan aliran kerja keseluruhan pengguna; dan c) Kaedah yang baik bagi mengumpulkan idea.

Temuduga	<ul style="list-style-type: none"> a) Membantu untuk membina hubungan baik dengan pihak berkepentingan; b) Memudahkan tindakan susulan dan penjelasan; c) Kaedah yang ringkas dan mudah; d) Memudahkan pemerhatian terus non-verbal behaviour; dan e) Bergantung kepada pengetahuan pengguna dan kepakaran penemuduga.
Tinjauan dan Soalselidik	<ul style="list-style-type: none"> a) Set soalan untuk dijawab; b) Jawapan dinilai dan digunakan bagi menentukan keperluan; c) Kaedah soalan secara terbuka dan tertutup digunakan; dan d) Maklumat dapat dikumpulkan daripada bilangan kumpulan sasaran yang lebih besar.
Joint Application Design (JAD)	<ul style="list-style-type: none"> a) Pasukan projek, pengguna dan pengurusan bekerjasama; b) Boleh mengurangkan <i>scope creep</i> sebanyak 50%; dan c) Dijalankan oleh fasilitator mahir dalam teknik JAD.

Langkah 5 : Sediakan Arkitektur Bisnes

- a) Arkitektur Bisnes adalah suatu dokumentasi berkaitan bisnes yang menerangkan mengenai organisasi dan digunakan untuk menentukan objektif organisasi yang lebih strategik. Arkitektur bisnes digunakan sebagai salah satu sumber rujukan dalam pengurusan keperluan (*Requirements Management*) pembangunan aplikasi. Arkitektur bisnes merupakan salah satu kerangka di dalam *Enterprise Architecture (EA)* yang dibangunkan oleh organisasi.
- b) Arkitektur bisnes di atas menunjukkan ianya akan menjawab persoalan mengenai:
 - i) Kenapa perlu dilakukan?
 - ii) Apa yang perlu dilakukan?
 - iii) Bagaimana ianya dilaksanakan?
 - iv) Dengan apa ianya dilaksanakan?
- c) Persoalan yang perlu dijawab dalam Arkitektur bisnes akan diterjemahkan ke dalam bentuk kerangka bisnes sebagaimana dibawah. Kerangka bisnes adalah berorientasikan perkhidmatan organisasi dan penyelesaian kepada keperluan bisnes yang dapat menghalang dari berlakunya konflik antara pemegang taruh.

Contoh arkitektur bisnes yang boleh dibangunkan adalah seperti berikut:

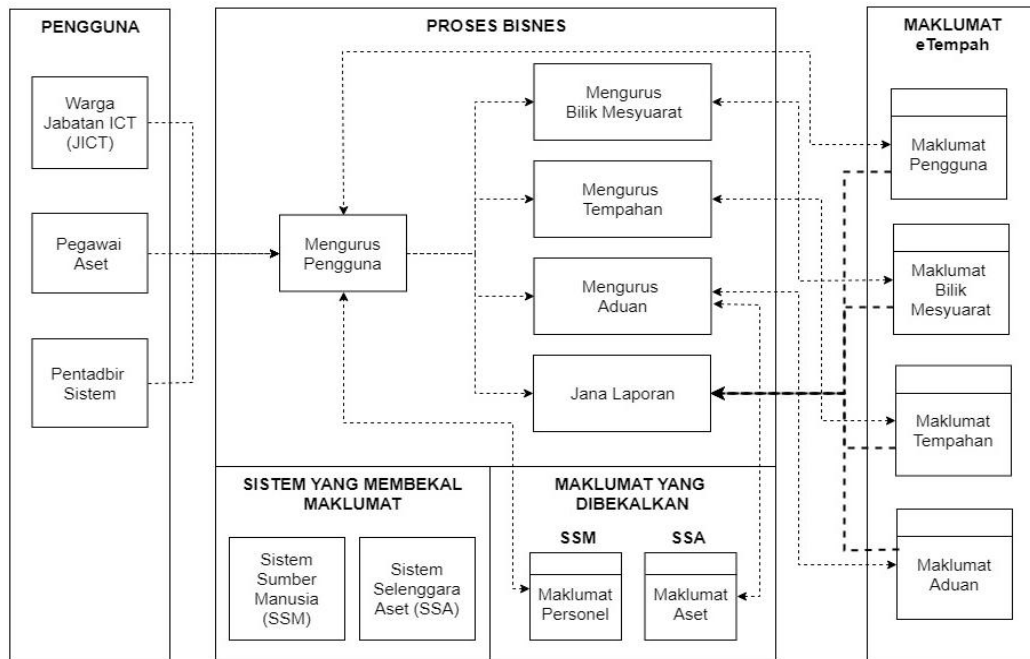


Rajah 20 : Contoh Arkitektur Bisnes

Langkah 6 : Sediakan Arkitektur Maklumat

- a) Arkitektur maklumat adalah rekabentuk berstruktur bagi persekitaran maklumat. Model konsep maklumat digunakan bagi melaksanakan aktiviti yang melibatkan struktur maklumat yang kompleks. Fokus utama arkitektur maklumat adalah untuk mengumpul, menstrukturkan, melabelkan kandungan maklumat secara berkesan dan mampan (*sustainable*). Tujuan utama arkitektur maklumat adalah bagi membantu dalam penggunaan dan pengaliran maklumat.
- b) Arkitektur maklumat berkait rapat dengan ekologi dan persekitaran maklumat. Ekologi maklumat merangkumi konteks (*context*), kandungan (*content*), dan pengguna (*users*).
- i) Konteks (*Context*): Sasaran bisnes, peruntukan, politik, persekitaran, teknologi, sumber dan kekangan.
 - ii) Kandungan (*Content*): Objektif, dokumentasi dan jenis data, jumlah data, struktur data terkini, pemilik data dan tadbir urus data.
 - iii) Pengguna (*Users*): Pemegang taruh, tugas, keperluan, cara carian dan penggunaan maklumat dan pengalaman.
- c) Arkitektur maklumat dikenal pasti dengan menentukan jenis kumpulan maklumat yang berkaitan dengan domain bisnes, unit organisasi yang mengumpul atau memproses data berkenaan, dan juga peranan serta tanggungjawab organisasi tersebut ke atas jenis data yang telah dikenal pasti. Berikut adalah keterangan lanjut berkenaan elemen-elemen di dalam arkitektur maklumat :
- i) Pengguna**
 Kenal pasti unit organisasi atau personel yang terlibat dengan jenis kumpulan maklumat dan proses yang berkenaan dengan domain bisnes. Pengguna boleh terdiri daripada pegawai yang bertanggungjawab seperti pemilik data dan pentadbir bahagian, nama jabatan, bahagian sehingga ke unit yang terkecil.
 - ii) Proses**
 Kenal pasti proses utama yang terlibat dalam domain bisnes serta interaksi dengan sistem lain. Proses utama boleh berpandukan kepada fungsi-fungsi bisnes tahap 1 dalam Pemodelan Fungsi Bisnes [F1.3].
 - iii) Kumpulan Maklumat**
 Tentukan Kumpulan maklumat yang berkaitan dengan domain bisnes, contohnya seperti maklumat kewangan, maklumat sumber manusia, maklumat gaji dan pelbagai lagi. Elakkan daripada menyenaraikan nama maklumat secara terperinci.

Contoh arkitektur maklumat yang boleh dibangunkan adalah seperti berikut:



Rajah 21 : Contoh Arkitektur Maklumat

Langkah 7 : Dokumenkan Maklumat

Dokumenkan maklumat yang dikumpul dari pihak SME. Susunkan maklumat mengikut kategori berikut:

- Halatujua Bisnes - terdiri daripada senarai objektif dan strategi bisnes termasuk bisnes unit yang terlibat.
- Keperluan Bisnes - terdiri daripada senarai proses/fungsi utama keperluan bisnes
- Model Data Luar - terdiri daripada senarai maklumat utama mengikut perspektif pemegang taruh
- Bisnes Unit dan Definisi Peranan - terdiri daripada peranan dan data elemen

RUJUKAN

- Gary, B. S. & Harry, J. R. (2012). System Analysis and Design. Ninth Edition. Boston: Course Technology
- Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems Analysis and Design. Fifth Edition. USA: John Wiley & Sons, Inc.
- Wieggers, K., E. & Joy, B. (2013). Software Requirements. Third Edition. Washington: Microsoft Press.

2.3.5 PEMODELAN FUNGSI BISNES [F1.3]

KETERANGAN



Permodelan Fungsi Bisnes adalah teknik untuk mengenal pasti fungsi yang sedang dan akan dijalankan oleh organisasi. Teknik ini menghasilkan model fungsi bisnes organisasi kepada pengguna yang direpresentasi dalam bentuk hirarki fungsi bisnes.

OBJEKTIF

- Mengenal pasti fungsi-fungsi yang dijalankan untuk mencapai objektif bisnes.
- Menghasilkan model fungsi yang tepat dan memenuhi keperluan sesebuah organisasi.

NOTASI FUNGSI BISNES

Jadual 9 : Notasi Fungsi Bisnes

Elemen	Keterangan
<p>Fungsi Bisnes</p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili setiap fungsi bisnes. • Labelkan ID Fungsi Bisnes (rujuk notasi di bawah) dan Nama Fungsi Bisnes di dalam notasi dengan menggunakan ayat yang bermula dengan kata kerja.
<p>Sintaks Nama Fungsi</p> <p><kata kerja><kata nama></p>	<ul style="list-style-type: none"> • Nama fungsi bisnes hendaklah dimulakan dengan kata kerja dan diikuti kata nama. <p>Contoh: Merekod Aset.</p>
<p>ID Fungsi Bisnes</p> <p>BF - FF - SF - FTA</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama bagi setiap fungsi bisnes yang dibangunkan. • Keterangan Kandungan ID: <ul style="list-style-type: none"> BF = Ringkasan Nama Teknik FF = Ringkasan Nama Fungsi SF = Ringkasan Nama Subfungsi FTA = Ringkasan Nama Fungsi Asas
<p>Penghubung antara fungsi mendatar atau menegak</p> 	<ul style="list-style-type: none"> • Penghubung antara fungsi dan sub fungsi menggunakan garis lurus mendatar atau menegak

<p>Keterangan Fungsian Bisnes</p> <p><pemegang taruh></p> <p><kata kerja ></p> <p><prestasi/kekerapan></p> <p><peristiwa></p> <p><syarat-syarat></p>	<ul style="list-style-type: none"> • Pernyataan yang akan menerangkan suatu fungsian bisnes.
---	---

LANGKAH-LANGKAH

Langkah 1 : Lakukan Analisis Keperluan Bisnes

- a) Laksanakan analisis ke atas maklumat yang didokumenkan dalam proses Penentuan Keperluan Bisnes [F1.2] untuk mengenal pasti fungsi-fungsi bisnes. Fungsi bisnes terdiri daripada fungsi utama (*root function*) dan sub-sub fungsi. Fungsi utama boleh diperolehi dengan mengenal pasti apa yang perlu dilakukan untuk mencapai misi organisasi atau tujuan utama projek dibangunkan. Sub fungsi ialah fungsi yang perlu dilaksana untuk merealisasikan fungsi utama atau fungsi-fungsi di atasnya mengikut hierarki fungsi. Semua fungsi ini dapat dikenal pasti melalui perkataan kata kerja yang bersifat aktif, kukuh, tepat dan difahami dalam dokumen kes bisnes.

Contoh cabutan sebahagian daripada kes bisnes:

“Maklumat bilik mesyuarat perlu didaftar dan dikemaskini melalui sistem. Pengguna sistem boleh menyemak jadual tempahan dan fasiliti bilik mesyuarat serta melakukan permohonan tempahan bilik mesyuarat secara melalui sistem.”

- b) Perkataan yang bergaris merupakan perkataan berbentuk kata kerja yang boleh menentukan fungsi yang dilaksanakan oleh bisnes. Analisis ini akan menghasilkan senarai fungsi yang akan digunakan dalam langkah seterusnya.

Langkah 2 : Modelkan Fungsi Bisnes

Fungsi utama dan sub fungsi diwakilkan dalam bentuk rajah berstruktur yang dihasilkan melalui proses penguraian fungsi (*function decomposition*). Proses penguraian fungsi ini dilaksanakan ke atas semua fungsi bisnes bermula dari fungsi utama diikuti dengan sub fungsi tahap pertama, kedua dan seterusnya (bergantung kepada kompleksiti sistem) sehingga sub fungsi terkecil. Sub fungsi terkecil ini tercapai apabila fungsi tersebut mempunyai ciri-ciri seperti berikut:

- Bermakna dan memenuhi keperluan pengguna
- Merupakan transaksi yang lengkap
- Tidak memerlukan langkah atau fungsi lain bagi memulakan atau melengkapkannya (*self-contained*)
- Menjadikan aplikasi bisnes dalam keadaan yang konsisten.

Subfungsi terkecil ini juga dikenali sebagai Fungsi Asas (*elementary transactional function*) yang digunakan sebagai asas kepada pengiraan awal saiz sistem aplikasi.

Berikut ialah langkah-langkah dalam pemodelan fungsi bisnes:

a) Kenal Pasti Fungsi Utama Bisnes

Fungsi utama boleh diperolehi dengan mengenalpasti apa yang perlu dilakukan untuk mencapai misi atau tujuan projek ini hendak dibangunkan. Fungsi utama ini berada di tahap 0. Berikan nama fungsi dan ID fungsi dengan menggunakan notasi yang ditetapkan pada Sintaks Nama Fungsi dan ID Fungsi Bisnes. Contoh:

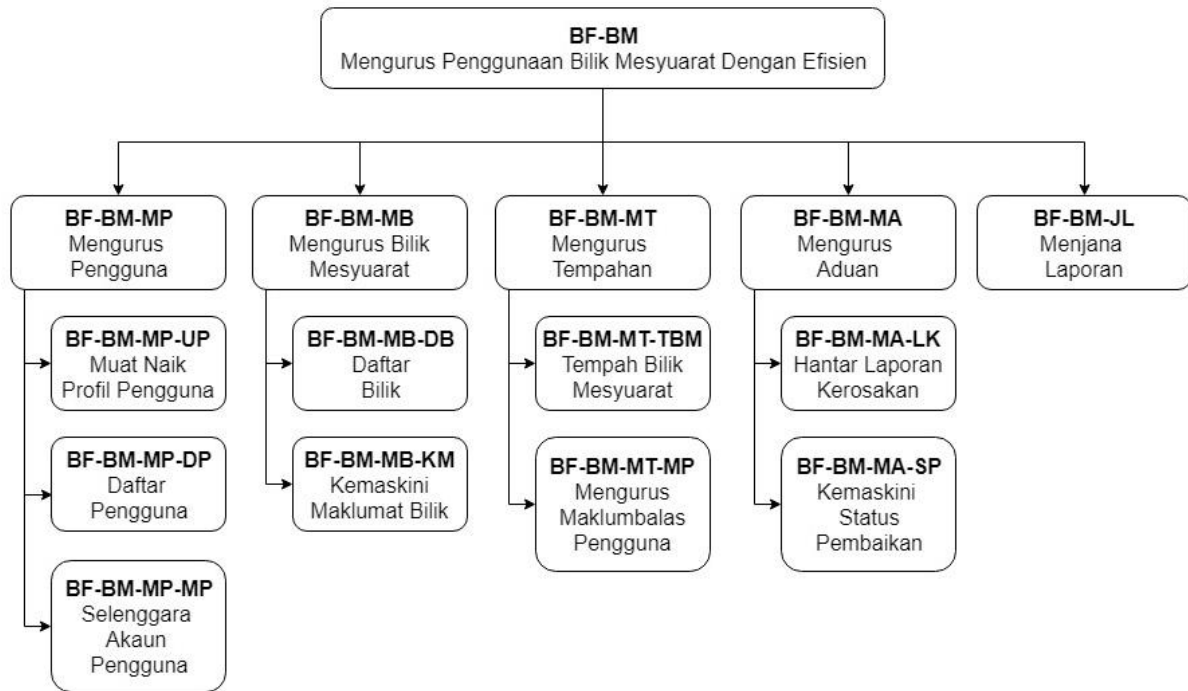
BF-BM Mengurus Penggunaan Bilik Mesyuarat Dengan Efisien

b) Kenalpasti dan susun sub fungsi di bawah fungsi utama yang berada di tahap 1 dalam rajah hierarki fungsi. Berikan nama dan ID setiap fungsi dengan menggunakan notasi Sintaks Nama Fungsi dan ID Fungsi Bisnes yang ditetapkan.

Jadual 10 : Keterangan Label Fungsi Bisnes

Bil.	Label	Keterangan
1	ID Fungsi Bisnes	Setiap fungsi bisnes perlu dilabelkan mengikut <i>naming convention</i> seperti yang diterangkan dalam notasi Fungsi Bisnes. ID fungsi bisnes perlu dilabelkan bermula dengan fungsi utama sehingga fungsi asas dicapai. Contoh ID fungsi bisnes bagi sub fungsi adalah sebagaimana berikut: BF - BM - MP Keterangan: BF - Nama teknik yang digunakan iaitu Fungsi Bisnes BM - Singkatan nama bagi fungsi utama iaitu Bilik Mesyuarat MP - Singkatan nama bagi sub fungsi iaitu Mengurus Pengguna
2	Nama Fungsi Bisnes	Nama fungsi bisnes adalah berdasarkan kepada <i>naming convention</i> yang telah ditetapkan dalam notasi fungsi bisnes yang bermula dengan kata kerja dan diikuti kata nama . Contoh kata kerja: <ul style="list-style-type: none"> • Mengurus • Menjana • Mengemaskini Contoh kata nama: <ul style="list-style-type: none"> • Bilik Mesyuarat • Tempahan • Aduan

Contoh Rajah Hierarki Fungsian Bisnes adalah seperti berikut:



Rajah 22 : Contoh Rajah Hierarki Fungsian Bisnes Bagi Sistem Tempahan Bilik Mesyuarat (eTempah)

- c) Kenalpasti dan susun sub fungsi yang menyokong kepada semua fungsi di tahap 1. Berikan nama dan ID fungsi dengan menggunakan notasi yang ditetapkan. Proses ini diulangi bagi semua tahap dalam rajah hierarki fungsi sehingga fungsi asas dicapai. Rajah hierarki fungsi boleh disediakan dalam bentuk menegak, mendatar atau hibrid.

Langkah 3 : Sertakan Keterangan Fungsi Bisnes

Hierarki fungsi yang disediakan akan lebih mudah difahami jika disertakan dengan keterangan bagi setiap fungsi tersebut berpandukan format yang telah ditetapkan dalam notasi fungsi bisnes. Keterangan fungsi adalah penerangan ringkas mengenai apa yang dijalankan oleh fungsi tersebut dalam menyokong fungsi di atasnya. Keterangan bagi fungsi ini boleh disediakan bersama SME bagi memastikan keterangan yang disediakan adalah tepat.

Langkah Penyediaan Keterangan Fungsi

- a) Pastikan setiap fungsi yang dihasilkan telah mempunyai ID dan nama fungsi. ID fungsi dan nama fungsi disusun dalam jadual sebagaimana contoh di bawah dan diberi keterangan yang tepat untuk fungsi tersebut.
- b) Contoh keterangan fungsi bisnes mengikut notasi yang telah ditetapkan adalah sebagaimana berikut:

<pemegang taruh> <kata kerja> <prestasi/kekerapan> <peristiwa>
<syarat-syarat>

Contoh:

Pemegang taruh - Pentadbir
Kata kerja – meluluskan
Prestasi/kekerapan – apabila menerima
Peristiwa – permohonan penggunaan bilik
Syarat-syarat – jika terdapat kekosongan

Pentadbir dapat meluluskan tempahan bilik mesyuarat apabila menerima permohonan penggunaan bilik mesyuarat jika terdapat kekosongan.

- c) Senaraikan fungsi yang perlu diberikan keterangan bermula dengan sub fungsi yang berada di tahap 2 sehingga fungsi ke fungsi asas. Jika fungsi yang berada di tahap 1 merupakan fungsi asas, fungsi tersebut perlu diberikan keterangan fungsi.

Jadual 11 : Contoh Keterangan Fungsi Bisnes

ID Fungsi Bisnes	Nama Fungsi Bisnes	Keterangan Fungsi Bisnes
BF-BM-MP-UP	Muat Naik Profil	Pengguna akan menggunakan fungsi memuat naik profil apabila ada penambahan pengguna baharu ke dalam sistem. Data yang dimuat naik diambil dari sistem maklumat pekerja.
BF-BM-MP-DP	Daftar Pengguna	Pengguna akan menggunakan fungsi Daftar Pengguna untuk mendaftar pengguna baharu sebagai pilihan bagi fungsi Muat Naik Profile. Data dimasukkan melalui borang yang disediakan.
BF-BM-JL	Menjana Laporan	Pengguna akan menggunakan fungsi Menjana Laporan untuk menjana dan mencetak laporan. Laporan yang dijana boleh pelbagai mengikut keperluan yang telah ditetapkan.

Langkah 4 : Kenalpasti Dan Senaraikan Pengguna Yang Terlibat

- Pengguna boleh dikenalpasti melalui keterangan fungsi yang telah disediakan. Keterangan fungsi boleh memberi gambaran yang lebih jelas mengenai peranan pengguna dan ianya boleh dikumpulkan dalam satu kumpulan yang sama.
- Sebagai contoh, keterangan fungsi Mengurus Bilik Mesyuarat dan Mengurus Aduan menggambarkan peranan yang dilakukan oleh Pegawai Aset Jabatan. Fungsi lain yang akan melibatkan Pegawai Aset Jabatan boleh dikumpulkan di bawah satu peranan pengguna yang sama.
- Senaraikan semua pengguna-pengguna yang telah dikenalpasti dan nyatakan keterangan bagi setiap pengguna tersebut. Rujuk kepada langkah 3a – perkara iii di dalam Penyediaan Spesifikasi Keperluan Bisnes [F1.5] untuk keterangan lanjut berhubung dengan kaedah untuk menghasilkan senarai pengguna tersebut.

Langkah 5 : Semak Dan Tambah Baik Model Fungsi Bisnes

- Fungsi-fungsi kerja yang telah dikenal pasti dalam langkah 2 perlu disemak dan ditambah baik mengikut keperluan. Dalam usaha untuk melengkapkan model fungsi bisnes yang telah dihasilkan, pasukan pembangun perlu melibatkan SME bagi mendapatkan input yang lebih tepat. Pelbagai pendekatan boleh dilakukan. Bagi tujuan pembangunan sistem yang besar dan kompleks teknik JAD adalah lebih praktikal.

b) Pendekatan semakan yang sering digunakan adalah:

- i) Semak dengan kitar hayat entiti bisnes/maklumat yang berkaitan
Gunakan dokumen proses kerja atau SOP untuk menyemak kitaran hayat entiti bisnes dan dibandingkan dengan model fungsian yang telah dihasilkan. Perhatikan bagaimana entiti-entiti digunakan dalam aktiviti-aktiviti kerja organisasi. Bandingkan entiti ini dengan fungsi model bisnes yang telah dihasilkan. Tambah baik model fungsi jika perlu.
- ii) Semak dengan strategi dan peristiwa bisnes
Senaraikan fungsi-fungsi yang bertindak balas terhadap peristiwa penting bisnes dan gunakan senarai tersebut sebagai perbandingan dengan model fungsian yang telah dihasilkan. Ini adalah untuk menentukan sama ada terdapat fungsi-fungsi lain yang belum diambil kira dalam model fungsian.
- iii) Semak dengan peranan individu
Gunakan peranan yang dimainkan oleh individu untuk dibandingkan dengan model fungsian bagi memastikan tiada fungsi individu yang tertinggal dalam model fungsian.

Langkah 6 : Muktamadkan Model Fungsi Bisnes Dengan SME

Model fungsi bisnes yang telah dihasilkan perlu dibentangkan kepada SME supaya model tersebut diterima dan dimuktamadkan.

Langkah 7 : Sediakan Model Fungsi Bisnes Secara Iteratif

- a) Penyediaan dan pembangunan Pemodelan Fungsi Bisnes perlu dilaksanakan secara iteratif bagi meningkatkan tahap komprehensif dan keperincian fungsi yang direkodkan.
- b) Pengemaskinian perlu dilakukan berdasarkan input yang diperolehi dalam penyediaan Model Proses Bisnes.

Langkah 8 : Dokumentasikan Fungsi Bisnes

- a) Dokumen dan masukkan hasil Pemodelan Fungsi Bisnes (Hierarki Bisnes) dan Senarai Pengguna yang telah disediakan ke dalam D02 Spesifikasi Keperluan Bisnes.
- b) Rujuk kepada Penyediaan Spesifikasi Keperluan Bisnes [F1.5] bagi mengenalpasti ruangan di mana maklumat-maklumat dan model tersebut perlu diletakkan

RUJUKAN

1. Gary, B. S. & Harry, J. R. (2012). System Analysis and Design. Ninth Edition. Boston: Course Technology
2. Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems Analysis and Design. Fifth Edition. USA: John Wiley & Sons, Inc.
3. Wieggers, K., E. & Joy, B. (2013). Software Requirements. Third Edition. Washington: Microsoft Press



2.3.6 PEMODELAN PROSES BISNES [F1.4]

KETERANGAN





Pemodelan proses bisnes merupakan teknik yang digunakan untuk membangunkan Model Proses Bisnes. Model proses bisnes yang direpresentasi melalui Rajah Aliran Proses (*Process Flow Diagram* (PFD)) menerangkan bagaimana gabungan aktiviti-aktiviti yang berbeza menyokong kepada satu-satu proses bisnes. PFD menggambarkan kompleksiti dan susunan aktiviti-aktiviti bisnes dijalankan sama ada secara berjujukan, selari atau serentak. Setiap aktiviti yang telah diplotkan di dalam rajah akan diperincikan di dalam jadual Definisi Aktiviti Fungsi Bisnes.



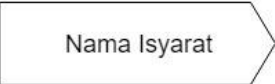

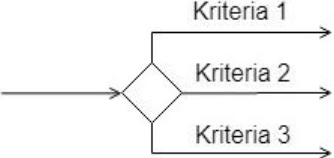
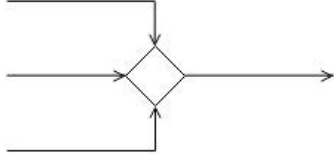
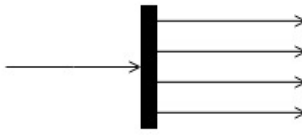
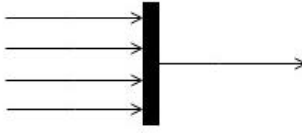
OBJEKTIF

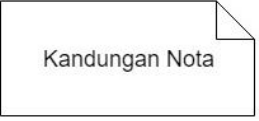
- Menghasilkan PFD *to-be* berdasarkan Fungsi Asas yang telah dikenal pasti di dalam Pemodelan Fungsi Bisnes[F1.3].
- Menyediakan Definisi Aktiviti Fungsi Bisnes yang memerincikan lagi keterangan bagi setiap aktiviti yang terkandung dalam PFD.

NOTASI

Jadual 12 : Notasi Proses Bisnes

Elemen	Keterangan
<p><i>Activity</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili setiap aktiviti. • Labelkan ID Aktiviti Bisnes (rujuk notasi di bawah) dan Nama Aktiviti Bisnes di dalam notasi dengan menggunakan ayat yang bermula dengan kata kerja.
<p><i>Control Flow/Edge</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk menghubungkan satu elemen notasi kepada notasi-notasi berikutnya.
<p><i>Swimlane</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk mengumpulkan aktiviti-aktiviti yang di bawah satu-satu peranan. • Nama peranan dilabelkan berdasarkan individu, unit bisnes, organisasi atau sistem luaran.
<p><i>Initial Node</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk menggambarkan permulaan bagi proses. • Perkataan 'Mula' dilabelkan di atas notasi berkenaan.

<p><i>Activity Final Node</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk menamatkan keseluruhan aliran aktiviti. • Perkataan 'Tamat' dilabelkan di atas notasi berkenaan.
<p><i>Flow Final Node</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk memberhentikan aliran aktiviti yang spesifik sahaja. • Perkataan 'Tamat' dilabelkan di atas notasi berkenaan.
<p><i>Send Signal Action</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk penghantaran isyarat (notifikasi). • Nama isyarat yang dihantar perlu dicatatkan di dalam notasi berkenaan.
<p><i>Accept Event Action</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk penerimaan isyarat (notifikasi). • Nama isyarat yang diterima perlu dicatatkan di dalam notasi berkenaan.
<p><i>Decision</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk memisahkan satu aliran proses kepada beberapa aliran proses berbeza berdasarkan salah satu kriteria keputusan yang ingin dipenuhi. • Labelkan kriteria keputusan seperti "Ya" atau "Tidak" bagi setiap cabang aliran proses berkenaan.
<p><i>Merge</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk menggabungkan semula aliran-aliran proses yang berlainan supaya hanya satu aliran proses sahaja yang disambungkan kepada aktiviti seterusnya.
<p><i>Fork</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk memisahkan aliran proses kepada beberapa aliran proses berbeza bagi aktiviti-aktiviti yang dilaksanakan secara selari.
<p><i>Join</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk menggabungkan semula aliran-aliran proses bagi aktiviti-aktiviti yang berjalan secara selari supaya proses dapat disambungkan kepada aktiviti seterusnya.

<p style="text-align: center;"><i>Note</i></p> 	<ul style="list-style-type: none"> • Digunakan untuk mencatatkan nota ringkas bagi satu-satu aktiviti. Kandungan nota dicatatkan di dalam ruangan notasi tersebut.
<p style="text-align: center;">ID Fungsi Bisnes</p> <p style="text-align: center;">PFD - FF - SF - FTA</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama dan nombor bagi setiap tajuk rajah PFD yang dibangunkan. • Keterangan Kandungan ID: <ul style="list-style-type: none"> PFD = Ringkasan Nama Teknik FF = Ringkasan Nama Fungsi SF = Ringkasan Nama Subfungsi FTA = Ringkasan Nama Fungsi Asas
<p style="text-align: center;">ID Aktiviti Bisnes</p> <p style="text-align: center;">PFD - FF - SF – FTA - 99</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama dan nombor bagi setiap aktiviti yang terkandung dalam satu-satu PFD. • Keterangan Kandungan ID: <ul style="list-style-type: none"> PFD = Ringkasan Nama Teknik FF = Ringkasan Nama Fungsi SF = Ringkasan Nama Subfungsi FTA = Ringkasan Nama Fungsi Transaksi Asas 99 = Nombor Rujukan Aktiviti Bisnes

LANGKAH-LANGKAH

Langkah 1 : Kenal Pasti Fungsi-fungsi Asas

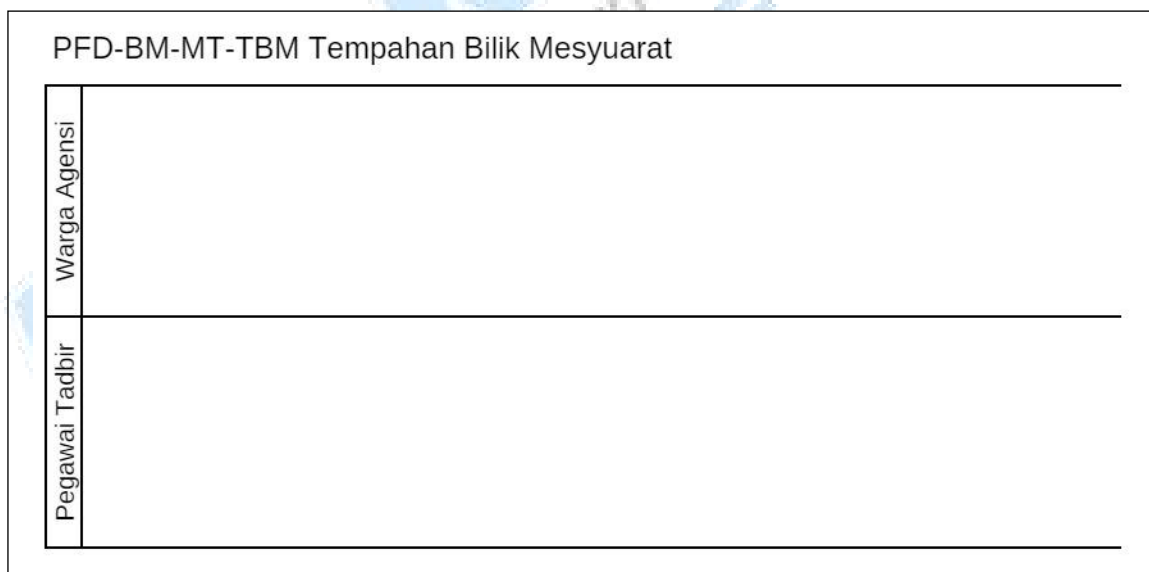
- a) Berdasarkan Hirarki Fungsian Bisnes yang telah disediakan di dalam Pemodelan Fungsi Bisnes [F1.3], kenal pasti setiap Fungsi Asas yang terlibat.
- b) Setiap Fungsi Asas yang telah dikenalpasti akan digunakan sebagai rujukan untuk menyediakan PFD dan juga Definisi-definisi Keperluan Bisnes yang berkaitan.
- c) Pada kebiasaanya, setiap satu Fungsi Asas akan diwakili oleh satu PFD sahaja. Namun begitu, Fungsi-fungsi Asas boleh digabungkan di dalam satu PFD sekiranya proses yang terlibat adalah ringkas, ataupun satu-satu Fungsi Asas boleh juga dipecahkan kepada beberapa PFD sekiranya proses terbabit adalah kompleks.

Langkah 2 : Kenal Pasti Pengguna Yang Terlibat

- Rujuk kepada Senarai Pengguna yang telah disediakan pada langkah 4 di dalam Pemodelan Fungsi Bisnes [F1.3] dan langkah 3a – perkara iii di dalam Penyediaan Spesifikasi Keperluan Bisnes [F1.5].
- Kenal pasti setiap pengguna yang terlibat dengan satu-satu PFD yang akan disediakan.
- Pengguna-pengguna tersebut diwakili dengan menggunakan notasi *swimlane* yang akan dibangunkan di dalam langkah selanjutnya (langkah 3b).

Langkah 3 : Labelkan Tajuk Dan Lukiskan Swimlane Bagi Pengguna Yang Terlibat

- Catatkan ID Fungsi Bisnes dan Nama Fungsi Asas di sebelah atas rajah (di atas notasi *swimlane*) sebagai tajuk kepada PFD yang ingin dibangunkan. Rujuk kepada ruangan notasi di atas bagi kaedah untuk menyediakan tajuk tersebut.
- Lukiskan *swimlane* yang mewakili setiap pengguna yang terlibat dengan proses dan PFD berkenaan. Notasi ini boleh dilukis sama ada secara mendatar (*horizontal*) ataupun secara menegak (*vertical*). Contoh penyediaan *swimlane* adalah seperti rajah berikut.



Rajah 23 : Penyediaan Tajuk PFD dan *Swimlane*

Langkah 4 : Sediakan Aliran Proses Bisnes

- a) Mulakan setiap aliran proses dengan menggunakan notasi *Initial Node* ● .
- b) Lukis dan plotkan aktiviti-aktiviti bisnes dengan menggunakan notasi *Activity*. Catatkan setiap notasi tersebut dengan ID Aktiviti Bisnes dan Nama Aktiviti Bisnes seperti yang telah dinyatakan di dalam ruangan notasi di atas. Contoh penyediaan aktiviti bisnes adalah seperti berikut:



Rajah 24 : Contoh Penyediaan Notasi *Activity* (Aktiviti Bisnes)

6.

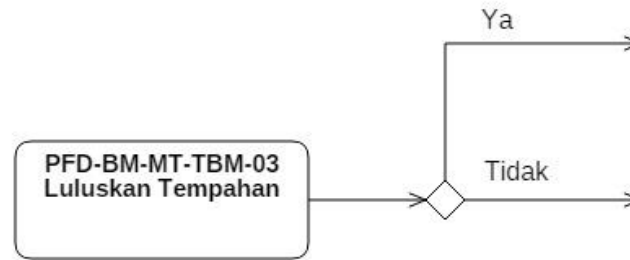
- c) Lukiskan, sekiranya perlu, notasi *Send Signal Action* dan *Accept Event Action* bagi sebarang penghantaran dan penerimaan notifikasi. Setiap notasi *Send Signal Action* perlu diikuti dengan notasi *Accept Event Action* di mana notasi-notasi tersebut tidak boleh diplotkan secara bersendirian. Catatkan keterangan di dalam ruangan notasi-notasi tersebut seperti contoh berikut:



Rajah 25 : Contoh Penyediaan Notasi *Send Signal Action* dan *Accept Event Action*

- d) Aliran proses boleh dipisahkan atau digabungkan dengan menggunakan notasi-notasi di bawah :
 - i) **Decision**

Memisahkan satu aliran proses kepada beberapa aliran proses berbeza. Walaupun notasi ini memisahkan kepada beberapa aliran proses, hanya satu aliran sahaja yang akan digunakan berdasarkan kepada kriteria keputusan atau guard yang ingin dipenuhi. Labelkan setiap kriteria keputusan pada setiap aliran proses yang keluar dari notasi ini. Pastikan Nama Aktiviti Bisnes di dalam notasi sebelum dapat menggambarkan maksud setiap kriteria keputusan yang akan digunakan. Contoh penggunaan notasi Decision adalah seperti berikut:



Rajah 26 : Penggunaan Notasi *Decision*

ii) **Merge**

Menggabungkan semula aliran-aliran proses berlainan yang terpisah akibat penggunaan notasi *Decision* atau *Fork*. Penggunaan notasi ini diperlukan supaya hanya satu aliran proses sahaja yang akan dipanjangkan kepada aktiviti bisnes seterusnya. Notasi ini menerima mana-mana aktiviti bisnes yang telah selesai dilaksanakan tanpa perlu menanti aktiviti-aktiviti lain yang sedang di dalam tindakan.

iii) **Fork**

Memisahkan aliran proses kepada beberapa aliran proses berbeza. Perbezaannya dengan notasi *Decision* ialah notasi ini membolehkan aliran-aliran proses yang berbeza dilaksanakan secara selari tanpa perlu melalui sebarang kriteria keputusan.

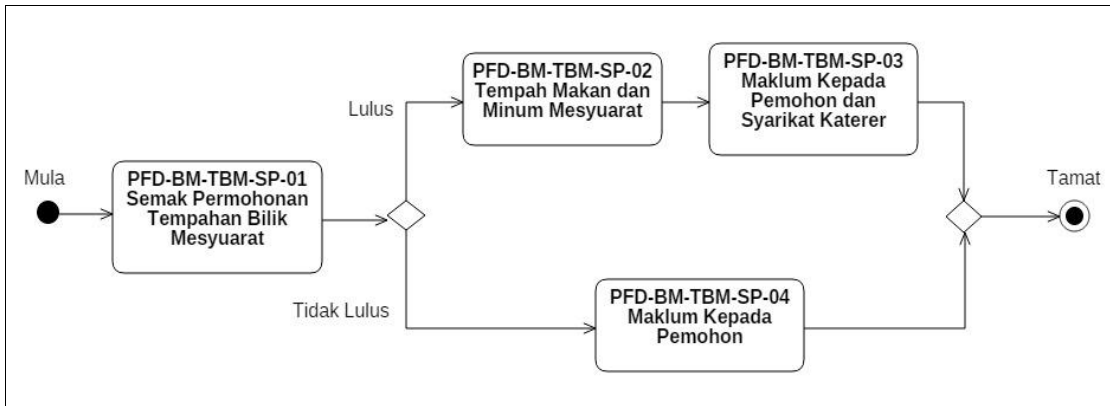
iv) **Join**

Menggabungkan semula aliran-aliran proses bagi aktiviti-aktiviti berbeza yang berjalan secara selari melalui penggunaan notasi *Fork* di aliran sebelumnya. Penggunaan notasi ini diperlukan supaya hanya satu aliran proses sahaja yang akan disambungkan kepada aktiviti bisnes seterusnya. Notasi ini perlu menerima dan menantikan terdahulu kesemua aktiviti bisnes yang berkait dengannya selesai dilaksanakan sebelum aliran proses bersambung kepada aktiviti bisnes berikutnya.

- e) Penerangan dan catatan tambahan bagi satu-satu aktiviti diwakili dengan notasi *Note*.
- f) Tamatkan aliran proses sama ada dengan menggunakan notasi *Activity Final Node* atau *Final Flow Node*. Contoh penggunaan notasi Nod-nod tersebut adalah seperti berikut :

i) **Notasi *Activity Final Node***

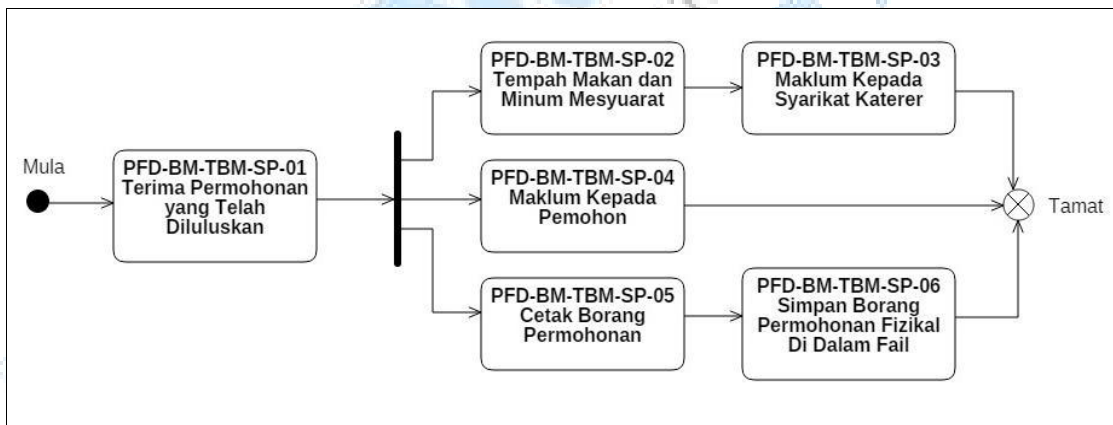
Berdasarkan kepada contoh dalam rajah di bawah, notasi *Activity Final Node* digunakan untuk menamatkan keseluruhan proses sekiranya salah satu daripada aliran proses yang berkriteria 'Lulus' atau 'Tidak Lulus' selesai dilaksanakan.



Rajah 27 : Penggunaan Notasi *Activity Final Node*

ii) **Notasi *Final Flow Node***

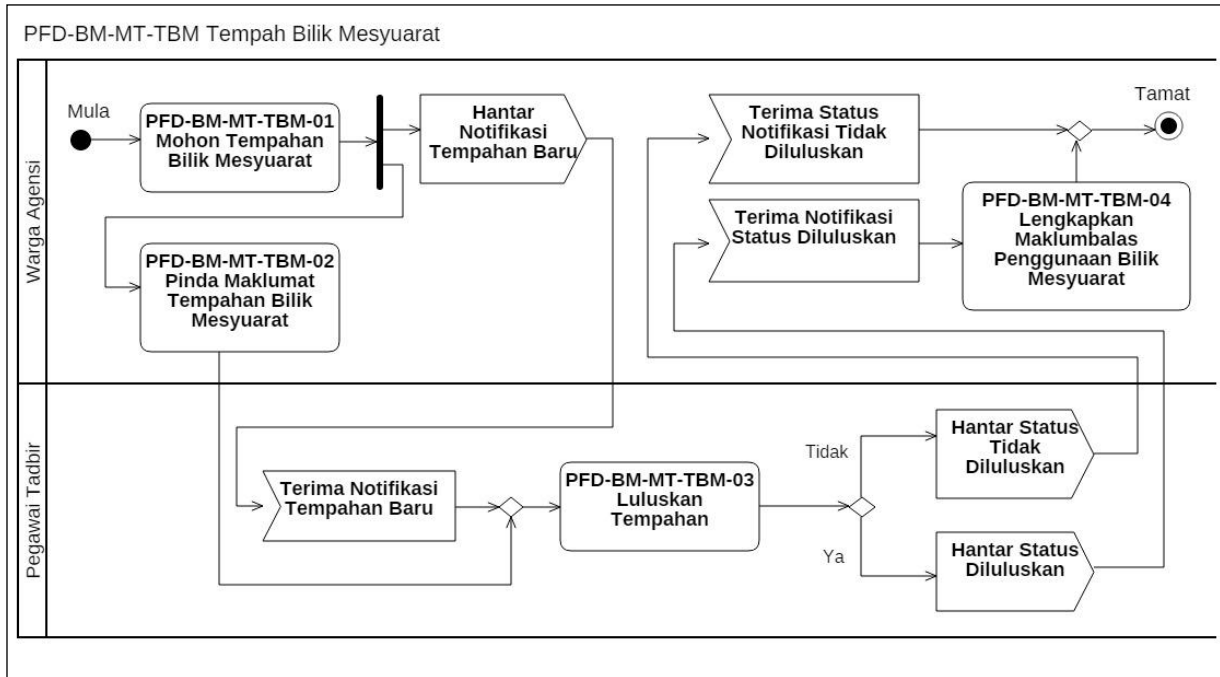
Berdasarkan kepada contoh dalam rajah di bawah, notasi *Final Flow Node* digunakan untuk menamatkan salah satu aliran proses sahaja berdasarkan kriteria yang telah dipenuhi. Aliran-aliran proses yang lain masih lagi perlu dilaksanakan sebelum proses keseluruhan ditamatkan.



Rajah 28 : Penggunaan Notasi *Final Flow Node*

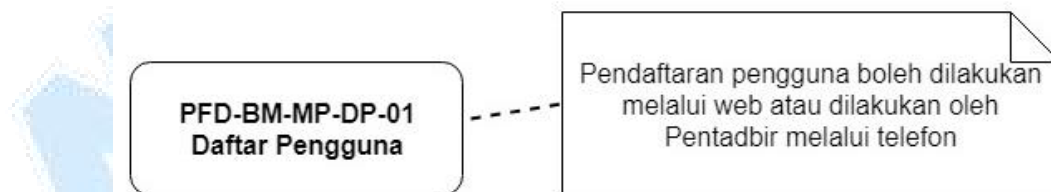
Langkah 5 : Sediakan Hubungan Di Antara Notasi-notasi

- a) Hubungkan notasi-notasi *Initial Node*, *Activity*, *Decision*, *Merge*, *Fork*, *Join*, *Send Signal Action*, *Accept Event Action* dan *Final Nodes* dengan menggunakan notasi *Control Flow* ataupun nama umumnya, *Edge* \longrightarrow . Contoh penyediaan hubungan dengan menggunakan notasi *Control Flow/Edge* adalah seperti rajah berikut:



Rajah 29 : Penggunaan Notasi *Control Flow/Edge* dalam Proses Tempah Bilik Mesyuarat

- b) Bagi notasi *Note*, notasi ini dihubungkan dengan notasi-notasi lain menggunakan garisan putus-putus. Contoh penggunaan hubungan di antara notasi *Note* dengan notasi lain adalah seperti berikut:



Rajah 30 : Penggunaan Notasi *Note*

Langkah 6 : Lengkapi Definisi Aktiviti Fungsi Bisnes

- a) Isi dan lengkapkan jadual Definisi Aktiviti Fungsi Bisnes bagi setiap aktiviti di dalam Rajah Aliran Proses. Maklumat-maklumat di dalam Definisi Aktiviti Fungsi Bisnes bukan sahaja menerangkan berkenaan dengan keperluan fungsian bisnes tetapi ia juga boleh membantu dalam mengenal pasti keperluan bukan fungsian dan keperluan maklumat/data yang berkaitan dengan aktiviti bisnes berkenaan.

b) Isikan maklumat-maklumat berikut:

Jadual 13 : Keterangan Label Definisi Aktiviti Fungsi Bisnes

Label	Keterangan
ID Fungsi Bisnes	Masukkan ID fungsi bisnes seperti yang dinyatakan dalam jadual notasi di atas.
Nama Fungsi Bisnes	Masukkan nama fungsi bisnes berdasarkan fungsi asas yang dipilih dari rajah Hierarki Fungsi Bisnes.
ID Aktiviti Bisnes	Masukkan ID aktiviti bisnes seperti yang dinyatakan dalam jadual notasi di atas.
Nama Aktiviti Bisnes	Masukkan nama aktiviti bisnes yang terlibat berdasarkan Rajah Aliran Proses yang telah dibangunkan.
Keterangan Aktiviti	Nyatakan keterangan secara ringkas aktiviti bisnes yang terlibat.
Aktor	Tentukan aktor yang melaksanakan aktiviti bisnes berkenaan.
Tanggungjawab	Nyatakan unit organisasi yang bertanggungjawab kepada aktiviti bisnes terlibat (disyorkan untuk merekodkan tahap Unit Organisasi yang terkecil seperti cawangan, unit, sektor, seksyen dan lain-lain).
Kekerapan	Nyatakan nilai kekerapan aktiviti bisnes tersebut dilaksanakan dalam satu-satu jangkamasa tertentu.
Unit Kekerapan (jam/hari/bulan)	Nyatakan unit untuk mengukur kekerapan pelaksanaan aktiviti bisnes berkenaan. Contoh, rekodkan unit kekerapan kepada setiap hari, minggu ataupun setiap bulan berdasarkan keperluan.
Aktiviti Bisnes Sebelum dan Selepas	Masukkan nama aktiviti bisnes yang sebelum dan selepas bersekali dengan nombor rujukan aktiviti masing-masing.
Kaedah Operasi	Terangkan secara terperinci bagaimana aktiviti tersebut dilaksanakan. Pengisian di dalam Kaedah Operasi boleh merangkumi keterangan berkenaan aktiviti bisnes yang terlibat, keperluan migrasi dan integrasi data dengan sistem luaran (entiti luar).

Label	Keterangan
Penggunaan Maklumat	Senaraikan maklumat atau data yang digunakan untuk dipapar, disimpan, dihapus atau/dan dikemaskini (CRUD). Maklumat atau data yang disenaraikan boleh terdiri sama ada dari sistem yang ingin dibangunkan, serta maklumat atau data dari sistem luaran (entiti luar) yang diperlukan bagi proses integrasi dan migrasi.
Polisi dan Dasar Berkaitan	Senaraikan polisi dan dasar yang perlu dipatuhi dan berkaitan dengan aktiviti yang terlibat, contohnya seperti Surat Pekeliling Perkhidmatan, Arahan Perbendaharaan, Manual Prosedur Kerja dan <i>Standard Operating Procedure</i> (SOP).
Kaedah Alternatif	Nyatakan kaedah alternatif yang perlu diambil sekiranya sistem yang akan dibangunkan menghadapi isu dan masalah sementara.
Ciri-ciri Kualiti (Keperluan Bukan Fungsian)	Nyatakan ciri-ciri kualiti bagi aktiviti berkenaan sama ada dari segi masa pemprosesan, ketersediaan, kebolehcapaian dan keselamatan.
Catatan Tambahan	Isikan catatan tambahan berkenaan aktiviti bisnes yang terlibat.

- c) **Rujuk** Apendiks 2 Template Definisi Aktiviti Fungsi Bisnes.

Langkah 7 : Muktamadkan Model Proses Bisnes Dengan SME

- a) Rajah Aliran Proses dan Definisi Fungsi Bisnes yang telah disediakan perlu disemak dan disahkan oleh pihak SME. Semakan perlulah mengambil kira perkara-perkara seperti di bawah :
- i) Rajah Aliran Proses dan Definisi Fungsi Bisnes yang disediakan perlulah komprehensif, lengkap, terperinci dan memenuhi keperluan semua fungsi bisnes yang terlibat.
 - ii) Ketepatan aliran proses dan hubungan dengan aktor
 - iii) Ketepatan definisi fungsi/proses bisnes dan aktor-aktor yang terlibat
- b) Pihak SME dan juga pasukan pembangunan perlu mempunyai kefahaman yang jelas semasa proses semakan berkenaan dengan perkara-perkara berikut :
- i) Fungsi dan aktiviti bisnes yang akan dibangunkan

- ii) Kaitan fungsi bisnes dan aktiviti yang dibangunkan

Langkah 8 : Sediakan Model Proses Bisnes Secara Iteratif

- a) Penyediaan dan pembangunan Model Proses Bisnes perlu dilaksanakan secara iteratif bagi meningkatkan tahap komprehensif dan keperincian maklumat yang direkodkan.
- b) Pengemaskinian juga perlu dilakukan kepada maklumat di dalam Pemodelan Fungsi Bisnes [F1.3] berdasarkan input yang diperolehi dalam penyediaan Model Proses Bisnes.

Langkah 9 : Dokumentasikan Proses Bisnes

- a) Dokumen dan masukkan hasil Pemodelan Proses Bisnes (Rajah Aliran Proses dan Definisi Fungsi Bisnes) yang telah dibangunkan ke dalam D02 Spesifikasi Keperluan Bisnes.
- b) Rujuk kepada Penyediaan Spesifikasi Keperluan Bisnes [F1.5] bagi mengenalpasti ruangan di mana model tersebut perlu diletakkan.

RUJUKAN

1. Alan Dennis, Barbara Haley Wixom, David Tegarden (2012). Systems Analysis and Design with UML Version 2.0, Second Edition.
2. Uml-diagrams.org (2009-2017). Activity Diagrams. <https://www.uml-diagrams.org/activity-diagrams.html>.

2.3.7 PENYEDIAAN SPESIFIKASI KEPERLUAN BISNES [F1.5]

KETERANGAN

Spesifikasi Keperluan Bisnes (BRS) adalah merupakan dokumen yang digunakan untuk merekod segala keperluan bisnes *to-be* yang diperolehi daripada pemegang taruh seperti pengguna, organisasi dan SME sebagai panduan bagi fasa-fasa pembangunan sistem yang selanjutnya. Rujuk kepada **D02 Spesifikasi Keperluan Bisnes** untuk melihat format dokumen yang terlibat di mana pengisian kandungan-kandungannya adalah seperti langkah-langkah di bawah.

OBJEKTIF

- Menyediakan Keperluan Pengurusan Bisnes yang diperolehi daripada Penentuan Keperluan Bisnes [F1.2].
- Menyediakan Keperluan Pengoperasian Bisnes yang terdiri daripada Keperluan Fungsi Bisnes dan Keperluan Proses Bisnes hasil daripada aktiviti yang dilaksanakan di dalam Pemodelan Fungsi Bisnes [F1.3] dan Pemodelan Proses Bisnes [F1.4].

LANGKAH-LANGKAH

Langkah 1 : Sediakan Pengenalan Kepada Keperluan Bisnes

Sedia dan lengkapkan pengenalan bisnes bagi perkara-perkara berikut:

a) Tujuan Bisnes

Terangkan secara ringkas hasrat dan tujuan bisnes baru yang ingin dicapai oleh satu-satu organisasi melalui sistem yang akan dibangunkan.

b) Skop Bisnes

Tentukan skop bisnes dengan mengenalpasti nama-nama dan liputan domain bisnes yang akan dirangkumkan di dalam sistem. Skop ini juga boleh ditentukan dengan mengambil kira organisasi, sama ada dalam atau luar, ataupun sistem-sistem lain yang terlibat secara langsung dengan domain bisnes yang telah ditetapkan.

c) Gambaran Keseluruhan Bisnes

Jelaskan secara rumusan domain bisnes yang akan dirangkumkan dalam sistem. Terangkan struktur organisasi-organisasi utama yang berkaitan dengan domain bisnes berkenaan serta hubungannya di antara satu sama lain ataupun dengan entiti luar sekiranya berkaitan. Penggunaan rajah adalah digalakkan untuk menerangkan struktur organisasi berkenaan.

d) Senarai Pemegang Taruh

Senaraikan pemegang-pemegang taruh yang terlibat dengan domain bisnes yang telah dipilih. Pemegang taruh boleh terdiri daripada seseorang individu, kumpulan ataupun unit organisasi. Contoh pengisian bagi senarai pemegang taruh adalah seperti jadual berikut:

Jadual 14 : Contoh Senarai Pemegang Taruh

Pemegang Taruh	Keterangan
Pengurusan Tertinggi	Pegawai-pegawai yang terdiri daripada Ketua Pengarah, Timbalan-timbalan Ketua Pengarah serta Pengarah Bahagian Sumber Manusia yang bertanggungjawab dalam memastikan sistem yang akan dibangunkan selaras dengan misi, visi dan hala tuju yang telah ditetapkan oleh organisasi.
Bahagian Sumber Manusia	Pemilik Proses kepada domain bisnes di dalam sistem yang akan dibangunkan.
Warga Agensi	Pengguna-pengguna yang melaksanakan dan menerima perkhidmatan bagi domain bisnes berkenaan.

Langkah 2 : Nyatakan Keperluan Pengurusan Bisnes

Nyata dan sediakan perkara-perkara yang diperlukan di bawah **Keperluan Pengurusan Bisnes** seperti berikut:

a) Objektif Bisnes

Senarai dan terangkan matlamat dan hasil bisnes yang ingin dicapai melalui pelaksanaan sistem yang akan dibangunkan. Pernyataan objektif perlulah spesifik dan mempunyai parameter yang boleh diukur berdasarkan kepada ciri-ciri S.M.A.R.T. (*Smart, Measureable, Achievable, Realistic & Time-Bound*).

b) Arkitektur Bisnes

Sertakan rajah Arkitektur Bisnes organisasi dan hubungannya dengan sistem aplikasi yang akan dibangunkan. Rujuk kepada langkah 5 di dalam **Penentuan Keperluan Bisnes [F1.2]** untuk mendapatkan penerangan lanjut berkenaan dengan penyediaan Arkitektur Bisnes.

c) Arkitektur Maklumat

Sertakan rajah Arkitektur Maklumat organisasi dan hubungannya dengan sistem aplikasi yang akan dibangunkan. Rujuk kepada langkah 6 di dalam Penentuan Keperluan Bisnes [F1.2] untuk mendapatkan penerangan lanjut berkenaan dengan penyediaan Arkitektur Maklumat.

Langkah 3 : Sediakan Keperluan Pengoperasian Bisnes

Keperluan Pengoperasian Bisnes terdiri daripada perkara-perkara berikut:

a) **Pemodelan Fungsi Bisnes**

i) **Penggunaan Notasi**

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi. Rujuk kepada Pemodelan Fungsi Bisnes [F1.3] untuk menyediakan senarai notasi berkenaan.

ii) **Model Fungsi Bisnes**

Sediakan Model Fungsi Bisnes yang terdiri daripada Rajah Hierarki Fungsi serta keterangan bagi fungsi-fungsi berkenaan. Sila rujuk kepada langkah-langkah di dalam Pemodelan Fungsi Bisnes [F1.3] bagi penyediaan Model Fungsi Bisnes.

iii) **Senarai Pengguna**

Berbeza dengan senarai pemegang taruh pada langkah 1, senarai pengguna merupakan perincian yang lebih lanjut kepada individu, kumpulan atau unit organisasi yang terlibat secara terus dengan fungsi bisnes. Daripada Model Fungsi yang telah disediakan, kenal pasti dan senaraikan pengguna-pengguna yang terlibat serta keterangan ringkas mengenai peranannya di dalam fungsi bisnes berkenaan. Contoh pengisian bagi senarai pengguna adalah seperti jadual berikut:

Jadual 15 : Contoh Senarai Pengguna

Pengguna	Keterangan
Pegawai Tadbir	Pengguna yang bertugas sebagai Ketua Unit yang bertanggungjawab untuk memantau fungsi bisnes yang dilaksanakan oleh unit di bawah seliaannya.
Penolong Pegawai Tadbir	Pengguna yang bertugas sebagai pegawai yang memberikan kelulusan kepada permohonan yang telah disokong.
Pembantu Tadbir	Pengguna yang bertugas sebagai pegawai yang mengesahkan permohonan-permohonan yang telah diterima.
Pemohon	Pengguna yang terdiri daripada warga agensi yang ingin melakukan permohonan tempahan bilik mesyuarat

b) Pemodelan Proses Bisnes

i) **Penggunaan Notasi**

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Proses Aliran Kerja. Sila rujuk kepada bab Pemodelan Proses Bisnes [F1.4] untuk menyediakan senarai notasi berkenaan.

ii) **Model Proses Bisnes**

- Aliran Proses Bisnes

Aliran Proses Bisnes digambarkan dengan menggunakan Rajah Aliran Proses. Sila rujuk kepada langkah 1 sehingga 3 di dalam Pemodelan Proses Bisnes [F1.4] bagi penyediaan Rajah Aliran Proses.

- Definisi Fungsi Bisnes

Sedia dan lengkapkan Definisi Aktiviti Fungsi Bisnes bagi setiap aktiviti yang telah disediakan di dalam Rajah Aliran Proses. Sila rujuk kepada langkah 4 di dalam Pemodelan Proses Bisnes [F1.4] dan format templat di Apendiks 2 - Templat Definisi Aktiviti Fungsi Bisnes

Langkah 4 : Kirakan Saiz Sistem Aplikasi

Pengenalan kepada kepentingan pengiraan saiz sistem aplikasi telah diperjelas dan diterangkan di dalam **1.6.5 Kepentingan Pengukuran Saiz Sistem Aplikasi**. Manakala kaedah pengiraan saiz sistem aplikasi pula telah diterangkan di dalam **Bab 8 Pengiraan Saiz Sistem Aplikasi**. Sehubungan dengan itu, laksanakan pengiraan saiz awal sistem aplikasi berdasarkan kepada elemen-elemen yang telah dikenalpasti di dalam BRS. Rujuk kepada bab 8 di dalam **Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Permulaan Projek** untuk mendapatkan penerangan terperinci berkenaan kaedah serta contoh pengiraan awal saiz sistem aplikasi.

Langkah 5 : Sertakan Dokumen-dokumen Sokongan Sebagai Lampiran

Sertakan dokumen-dokumen sokongan, sekiranya ada, yang perlu dirujuk seperti pekeliling, minit mesyuarat, borang-borang fizikal, surat-surat dan sebagainya.

Langkah 6 : Lakukan Semakan Dan Pengesahan Ke Atas Dokumen BRS

Dokumen BRS perlu dilakukan semakan oleh Ketua Pasukan Analisis dan Rekabentuk, atau pegawai-pegawai yang lain yang bersesuaian. Setelah semakan dilakukan, dokumen BRS

yang telah disediakan perlu disahkan oleh Ketua SME dan Pengurus Projek atau Pengarah Bahagian atau pegawai-pegawai yang lain yang bersesuaian.

RUJUKAN

1. ISO/IEC/IEEE 29148-2011 Systems and software engineering — Life cycle processes — Requirements engineering (2011)
2. Dokumen BRS Projek Sistem Bahagian Pengurusan Hartanah Versi 2 (2016)
3. Dokumen BRS Projek Sistem ePPAx (2016)






BAB:03

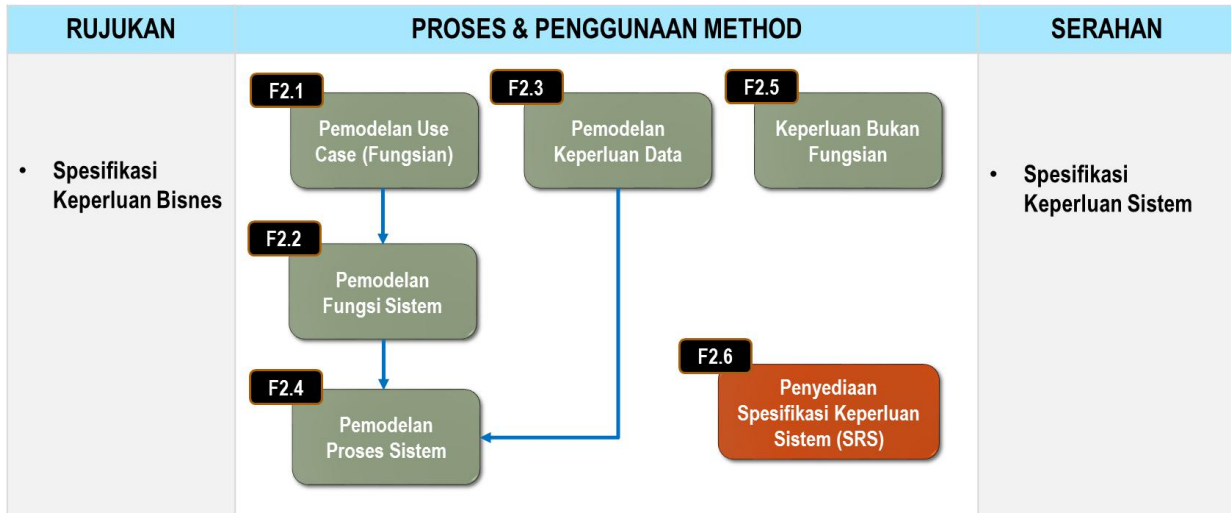
FASA ANALISIS

Bab ini menerangkan aktiviti-aktiviti analisis terhadap keperluan bisnes dan keperluan data yang telah dikenalpasti pada fasa sebelumnya. Bab ini juga menerangkan langkah-langkah untuk menyediakan pemodelan *Use Case*, fungsi sistem, proses sistem, data dan keperluan bukan fungsian bagi memenuhi kehendak pemegang taruh.



3 FASA ANALISIS

3.1 GAMBARAN KESELURUHAN



Rajah 31 : Gambaran Keseluruhan Fasa II - Analisis

3.2 PENGENALAN

Fasa analisis keperluan sistem merangkumi aktiviti-aktiviti analisis terhadap keperluan bisnes yang dikenalpasti dalam fasa permulaan projek dan menterjemahkan keperluan tersebut kepada keperluan sistem bagi tujuan pembangunan sistem baru ataupun penambahbaikan sistem sedia ada. Keperluan sistem adalah keterangan terperinci sistem aplikasi yang akan dibangunkan dari aspek fungsian, perkhidmatan dan kekangan operasi sistem. Dokumen keperluan sistem seharusnya menentukan dengan tepat apakah yang perlu dibangunkan dan dilaksanakan serta secara tidak langsung, ia akan dijadikan sebagai satu tanda persetujuan di antara pemegang taruh dengan pasukan pembangunan sistem aplikasi. Keperluan sistem aplikasi pada kebiasaannya diklasifikasikan kepada dua jenis keperluan, iaitu:

Keperluan Fungsian Keperluan fungsian menyatakan apakah yang boleh dilakukan oleh sistem dan bagaimana sistem tersebut seharusnya bertindakbalas terhadap satu-satu input atau situasi tertentu. Terdapat juga situasi di mana keperluan fungsian menjelaskan perkara-perkara yang tidak sepatutnya dilakukan oleh sesebuah sistem. Keperluan fungsian merangkumi fungsian yang berorientasikan kepada proses dan informasi. IFPUG menyatakan bahawa keperluan fungsian dikategorikan kepada dua jenis, iaitu:

- i) Keperluan Fungsi Transaksi
- ii) Keperluan Fungsi Data

Keperluan Bukan Fungsian Keperluan bukan fungsian adalah kekangan pada perkhidmatan atau fungsi di dalam satu-satu sistem aplikasi termasuklah kekangan masa, kekangan dalam proses pembangunan dan kekangan pada standard. Keperluan bukan fungsian sering dikaitkan kepada sistem aplikasi secara keseluruhan, dan bukannya kepada fungsi sistem itu sendiri.

la menentukan kriteria atau ciri-ciri yang boleh digunakan untuk mengukur prestasi sesuatu sistem.

Penghasilan keperluan sistem perlu melalui kajian yang terperinci ke atas maklumat keperluan bisnes yang telah diperolehi. Pelaksanaan kajian terperinci ini melibatkan empat (4) aktiviti berikut:

- a) Pemodelan *Use Case*
- b) Pemodelan Keperluan Data
- c) Pemodelan Proses Sistem
- d) Penentuan Keperluan Bukan Fungsian

Dokumen Rujukan kepada Fasa Analisis adalah **D02 Spesifikasi Keperluan Bisnes.**

Dokumen Serahan kepada Fasa Analisis adalah **D03 Spesifikasi Keperluan Sistem.**

3.3 PENGLIBATAN PEMEGANG TARUH

Pemegang Taruh utama yang terlibat di dalam analisis keperluan sistem adalah *Business* Juruanalisa Sistem. Juruanalisa sistem perlulah berkeupayaan untuk menterjemahkan keperluan bisnes daripada pengguna kepada keperluan sistem bagi memastikan sistem yang dibangunkan bertepatan dan sejajar dengan keperluan bisnes.

Cadangan penglibatan kategori pemegang taruh adalah seperti berikut:

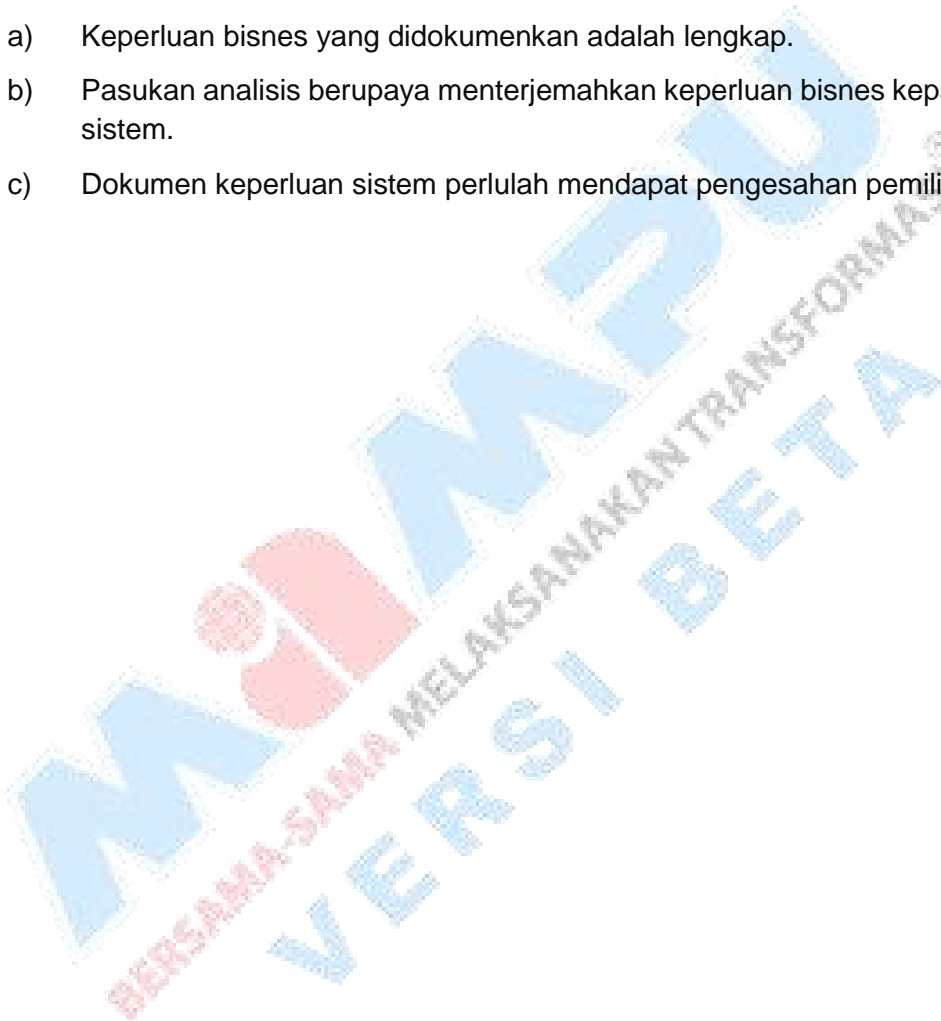
- a) Juruanalisis Sistem

- b) *Business Analyst*
- c) SME
- d) Pemilik Sistem
- e) Pengguna Sistem

3.4 FAKTOR KEJAYAAN

Untuk memastikan aktiviti berjaya dilaksanakan, berikut adalah faktor kejayaan utama yang perlu dipertimbangkan sebelum dan semasa aktiviti dilaksanakan:

- a) Keperluan bisnes yang didokumenkan adalah lengkap.
- b) Pasukan analisis berupaya menterjemahkan keperluan bisnes kepada keperluan sistem.
- c) Dokumen keperluan sistem perlulah mendapat pengesahan pemilik sistem.



3.5 PEMODELAN *USE CASE* (FUNGSIAN) [F2.1]

KETERANGAN

Pemodelan *Use Case* adalah teknik bagi mengenalpasti keperluan fungsian bagi sesuatu sistem. Teknik ini menterjemahkan fungsi-fungsi bisnes yang terdapat dalam Hierarki Fungsian Bisnes hasil daripada **Pemodelan Fungsi Bisnes [F1.3]** dan aktiviti-aktiviti proses bisnes yang terdapat dalam Rajah Aliran Proses Bisnes (PFD) daripada **Pemodelan Proses Bisnes [F1.4]**.

Hasil pemodelan *Use Case* dipersembahkan dalam bentuk Rajah *Use Case* yang menerangkan fungsian sistem secara menyeluruh dan komprehensif. Rajah *Use Case* menerangkan tabiat sistem iaitu bagaimana sistem berinteraksi dengan pengguna atau sistem luaran untuk mencapai matlamat fungsian bisnes.

Pemodelan *Use Case* terdiri daripada set berikut:


- a) Koleksi aktor yang berinteraksi dengan sistem;
- b) Koleksi interaksi antara *Use Case* dalam sistem; dan
- c) Lain-lain komponen yang menjelaskan gambarajah *Use Case*.




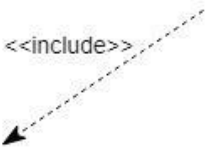
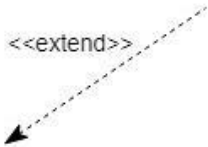

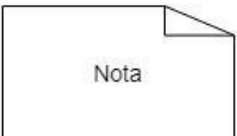
OBJEKTIF

- Mengenalpasti fungsi-fungsi sistem yang perlu dibangunkan dalam menyokong fungsi bisnes; dan
- Menghasilkan Rajah *Use Case* yang menerangkan bagaimana sistem berinteraksi dengan pengguna atau sistem yang lain untuk mencapai matlamat fungsian bisnes.

NOTASI

Jadual 16 : Notasi Rajah *Use Case*

Elemen	Keterangan
<p><i>Actor</i></p>  <p>Nama Aktor</p>	<ul style="list-style-type: none"> • Digunakan bagi mewakili peranan yang dimainkan oleh entiti luar seperti pengguna atau sistem luaran yang berinteraksi dengan sistem. • Setiap aktor perlu dinyatakan nama peranan.

<p><i>Use Case</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili fungsi sistem yang dilakukan oleh satu atau lebih aktor dalam mencapai sesuatu matlamat. • Setiap <i>Use Case</i> perlu diberi nama aktiviti/fungsi.
<p><i>Association</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili interaksi aktor dengan <i>Use Case</i>.
<p><i>Boundary</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi menetapkan sempadan sistem/modul yang menakrifkan skop sistem/modul yang akan dibangunkan.
<p><i>Include</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mentakrifkan sesuatu <i>Use Case</i> menggunakan fungsi dalam <i>Use Case</i> yang lain. Ini bermaksud <i>Use Case</i> tersebut tidak boleh berfungsi dengan sendiri. • Arah anak panah menghala kepada <i>Use Case</i> yang dipanggil.
<p><i>Extend</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mentakrifkan sesuatu <i>Use Case</i> menggunakan fungsi dalam <i>Use Case</i> yang lain dengan syarat-syarat tertentu. • Arah anak panah menghala kepada <i>Use Case</i> yang memanggil <i>Use Case</i> yang lain. • Setiap hubungan <extend> perlu menyatakan extension point di bawah use use utama atau pada line <extend>
<p><i>Inheritance / Generalization</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi menggambarkan kaitan antara elemen (aktor atau <i>Use Case</i>) umum dan khusus • Arah anak panah menghala kepada elemen umum • Elemen khusus mewarisi ciri-ciri elemen umum.
<p><i>Comment</i></p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili nota ringkas yang ingin dimasukkan di dalam aliran proses sebagai tambahan nota kepada gambarajah.

<p style="text-align: center;"><i>Artifact</i></p> <div style="border: 1px solid black; width: 100px; height: 50px; margin: 0 auto; text-align: center; line-height: 50px;"> <Artifak> </div>	<ul style="list-style-type: none"> • Bagi mewakili hubungan kepada gambarajah atau dokumen yang lain. Ia boleh dihubungkan dengan kebergantungan kepada mana-mana elemen lain pada gambarajah.
<p style="text-align: center;">ID <i>Use Case</i></p> <p style="text-align: center;">UC - SS - SBS - MM - SBM - 99</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama dan nombor bagi setiap Rajah <i>Use Case</i> berpandukan kepada Model Fungsi Sistem yang dibangunkan. • Keterangan Kandungan ID: <ul style="list-style-type: none"> SF = Ringkasan Nama Teknik SS = Ringkasan Nama Sistem SBS = Ringkasan Nama Subsistem MM = Ringkasan Nama Modul SBM = Ringkasan Nama Submodul *99 = Nombor Rujukan <i>Use Case</i> <p>* hanya diperlukan bagi penglabelan notasi <i>Use Case</i> sahaja.</p>

LANGKAH-LANGKAH

Langkah 1 : Terjemahkan Peranan (PFD) Kepada Aktor

- a) Pilih salah satu Rajah Aliran Proses Bisnes (PFD) yang didokumenkan dalam BRS yang dihasilkan dari fasa Pemodelan Proses Bisnes[F1.4].
- b) Terjemahkan **Peranan** (*swimlane*) dalam PFD kepada **Aktor** dalam Rajah *Use Case*. Penentuan aktor sistem adalah lebih tertumpu kepada peranan pengguna di dalam sistem aplikasi berbanding dengan nama jawatan seseorang individu.

Contoh :

Rujuk kepada **Pemodelan Proses Bisnes [F1.4]** yang menerangkan aliran proses bagi Permohonan Tempahan Bilik Mesyuarat. PFD tersebut mempunyai dua *swimlane* iaitu Warga Agensi dan dan Pegawai Tadbir. Contoh penterjemahan peranan kepada aktor adalah seperti jadual berikut:

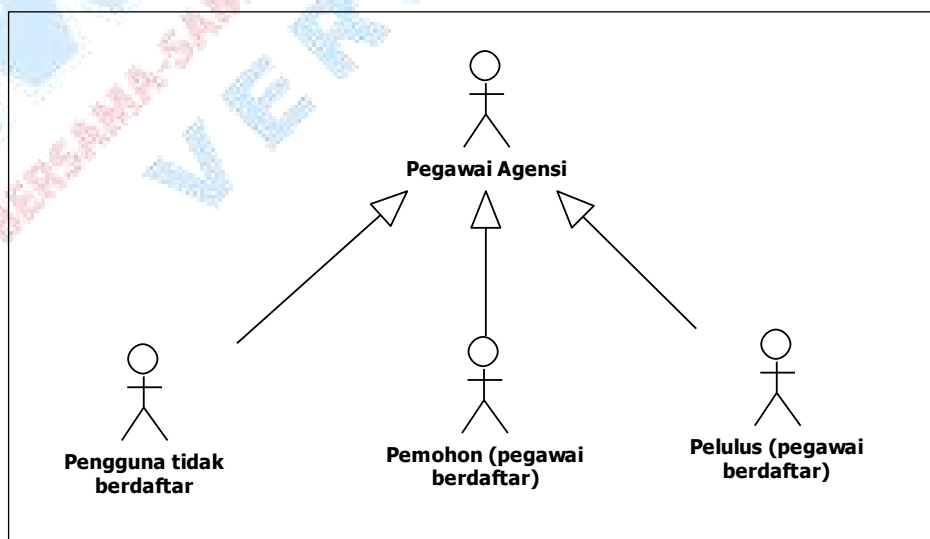
Jadual 17 : Contoh Jadual Terjemahan Peranan kepada Aktor

Peranan	Keterangan	Aktor
Warga Agensi	Pegawai agensi yang mempunyai akses untuk melakukan <u>permohonan</u> tempahan bilik mesyuarat.	Pemohon
Pegawai Tadbir	Pegawai yang dipertanggungjawab untuk <u>melulus</u> permohonan tempahan bilik mesyuarat	Pelulus

- c) Lakarkan koleksi Aktor yang dikenalpasti sebagai komponen dalam Rajah *Use Case*. Dalam melakarkan aktor, syarat-syarat berikut perlu dipatuhi:
- i) Aktor terletak di luar sempadan sistem/modul.
 - ii) Aktor boleh memainkan peranan sebagai pelaksana kepada *Use Case* dan juga penerima hasil daripada tindakan *Use Case* dalam berinteraksi dalam sistem
- d) Laksanakan pengkhususan Aktor. Multi-Aktor boleh dikumpulkan (*inherit behaviour*) bagi menjelaskan peranan aktor dalam Rajah *Use Case*.

Contohnya:

Bagi kes pegawai yang diberi peranan untuk menyemak kekosongan, pegawai yang diberi peranan membuat tempahan dan pegawai yang melulus tempahan. Bagi pegawai yang diberi peranan membuat dan melulus tempahan beliau perlu berdaftar dengan sistem.

**Rajah 32 : Contoh Penggunaan *Inherit Behavior* (Aktor)**

Langkah 2 : Terjemahkan Aktiviti Kepada Use Case

- a) Analisis aktiviti-aktiviti yang terdapat dalam PFD yang dipilih.
- b) Terjemahkan aktiviti-aktiviti tersebut kepada beberapa *Use Case* dalam Rajah *Use Case* yang dilakarkan. Dalam terjemahan kepada *Use Case* ianya tidak terikat samada satu aktiviti kepada satu *Use Case* atau beberapa aktiviti-aktiviti kepada satu *Use Case* atau satu aktiviti kepada beberapa *Use Case*. Pastikan semua aktiviti dalam PFD telah diambilkira dalam mengenalpasti *Use Case*. Ciri-ciri yang perlu diambilkira dalam melakar *Use Case* adalah:
 - i) *Use Case* perlulah lengkap, bermakna dan memberi hasil kepada pengguna.
 - ii) Elakkan daripada penstrukturan *Use Case* dalam peringkat tinggi kerana ini *Use Case* sukar difahami.
 - iii) *Use Case* boleh dipecahkan berdasarkan kepada kompleksiti fungsinya dan juga penggunaan fungsi secara berulang.
 - iv) *Use Case* yang dikenalpasti perlu dilakarkan di dalam sempadan Sistem atau Modul.
- c) Senaraikan *Use Case* yang dikenalpasti. Namakan, labelkan dan sediakan keterangan setiap *Use Case* seperti Jadual di bawah.

Jadual 18 : Keterangan Label Use Case

Label	Keterangan
ID <i>Use Case</i>	Setiap <i>Use Case</i> dilabelkan mengikut <i>namig convention</i> seperti yang diterangkan dalam notasi Pemodelan <i>Use Case</i> .
Nama <i>Use Case</i>	Nama adalah berdasarkan kepada kata kerja, dan <i>Use Case</i> ini merupakan fungsi sistem.
Keterangan <i>Use Case</i>	Keterangan bagi fungsi yang berkaitan, ianya sering dikaitkan dengan aktor dan logik fungsi

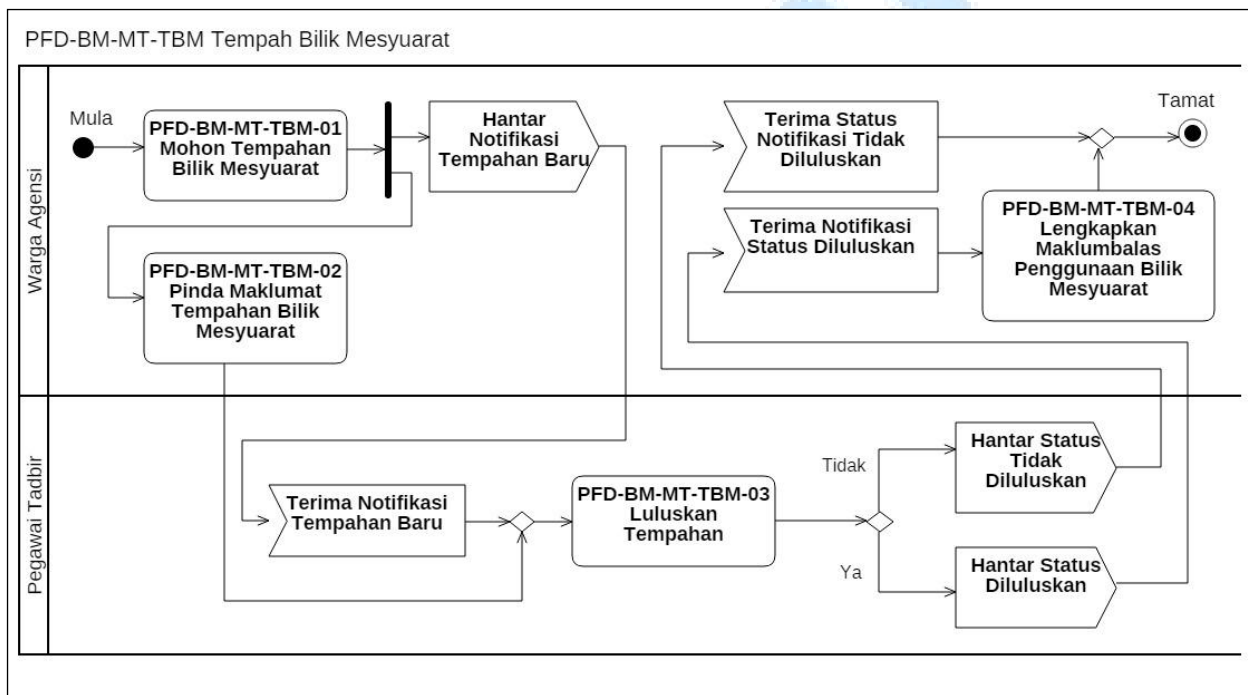
Langkah 3 : Sediakan Hubungan-hubungan Di Antara Aktor Dengan Use Case

- a) Lukiskan Aktor yang diterjemahkan dalam langkah 1 menggunakan notasi *Actor*.
- b) Lukiskan *Use Case* yang dikenalpasti dari langkah 2 menggunakan notasi *Use Case* dan labelkan setiap *Use Case*.
- c) Lakarkan sempadan sistem/modul merangkum semua *Use Case* yang dilukiskan menggunakan notasi *Boundary*. Manakala *Actor* perlu berada di luar sempadan.

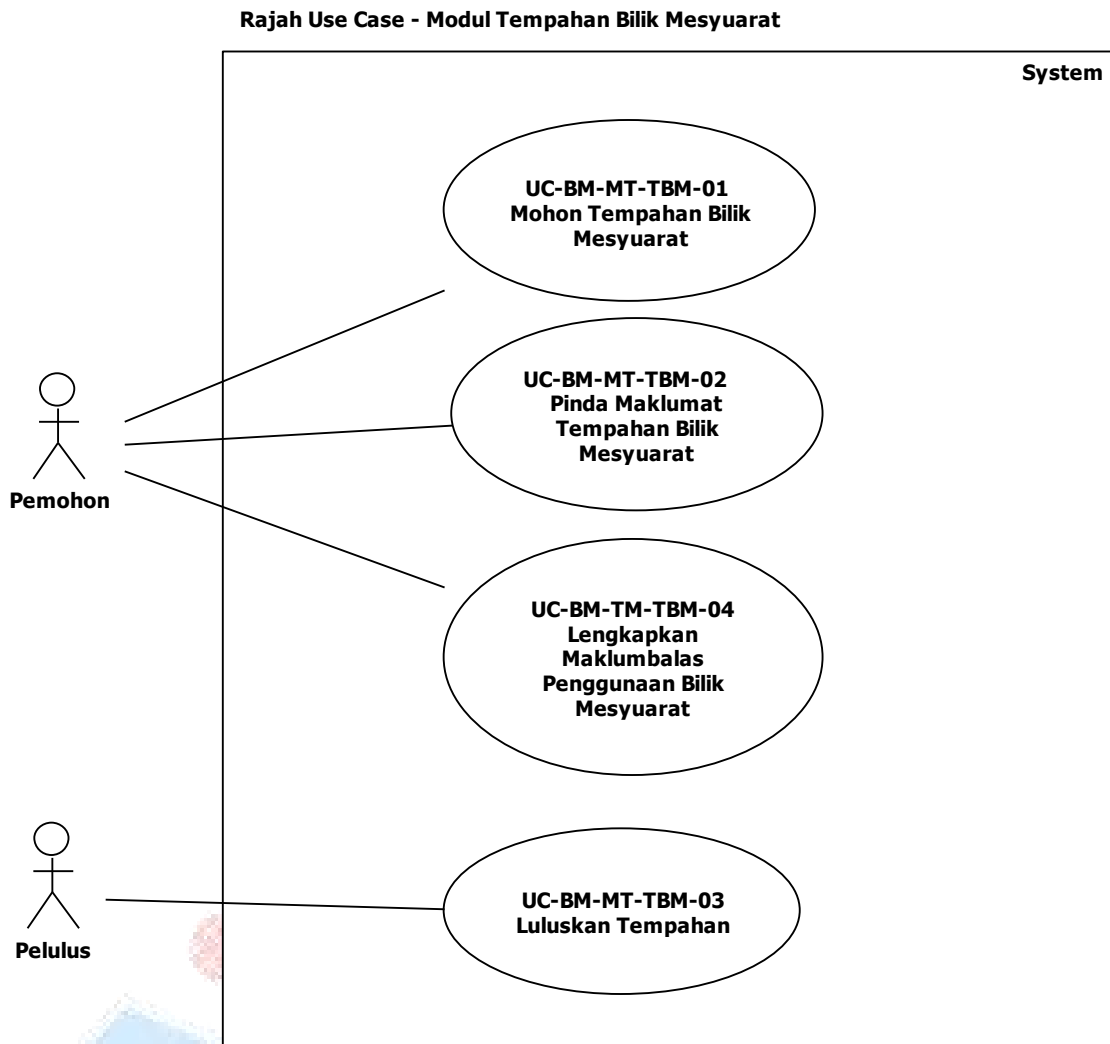
- d) Lakarkan hubungan antara Actor dan *Use Case*. Hubungan ini menunjukkan interaksi aktor kepada *Use Case*. Dalam hubungan ini notasi *association* sahaja digunakan.
- e) Sediakan keterangan setiap use case yang dilakarkan.

Contoh:

Menterjemahkan Proses Tempahan Bilik Mesyuarat (PFD-BM-MT-TBM) kepada **Rajah Use Case UC-BM-MT-TBM**. Berikut adalah Rajah Aliran Proses yang telah dibangunkan dalam **Pemodelan Proses Bisnes[F1.4]** bagi proses PFD-BM-MT-TBM Tempah Bilik Mesyuarat.



Rajah 33 : Rajah Aliran Kerja PFD-BM-MT-TBM Tempah Bilik Mesyuarat



Rajah 34 : Contoh Rajah *Use Case* - Hubungan Aktor Dengan *Use Case*

Jadual 19 : Contoh Keterangan *Use Case*

Label	Nama <i>Use Case</i>	Keterangan
UC-BM-MT-TBM-01	Memohon Tempahan Bilik Mesyuarat	Pemohon boleh membuat permohonan tempahan bilik mesyuarat
UC-BM-MT-TBM-02	Pinda Maklumat Tempahan Bilik Mesyuarat	Pemohon boleh meminda butiran tempahan yang telah dilakukan
UC-BM-MT-TBM-03	Luluskan Tempahan	Pelulus menyemak dan meluluskan permohonan

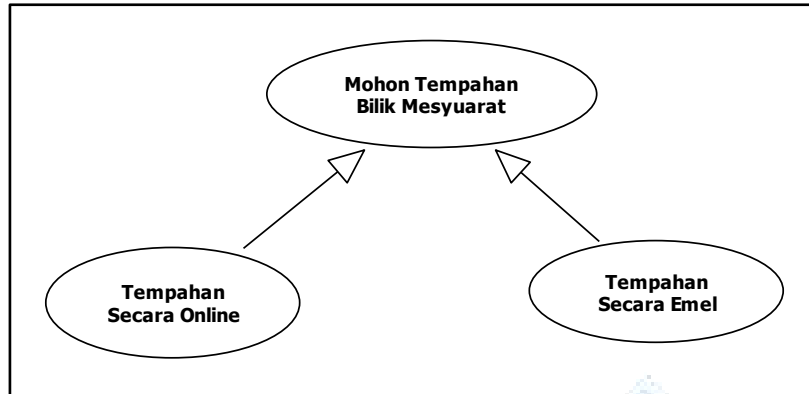
		tempahan yang dilakukan oleh pemohon
UC-BM-MT-TBM-04	Lengkapkan maklumbalas Penggunaan Bilik Mesyuarat	Selepas penggunaan bilik pemohon diminta untuk melengkapkan borang maklumbalas penggunaan Bilik

Langkah 4 : Perincian Hubungan Antara Use Case

- a) Menganalisis Rajah *Use Case* yang dihasilkan dari langkah 3 supaya bermakna kepada pasukan pembangunan. Analisis dilakukan berpandukan kepada kriteria berikut:
- i) Adakah terdapat *Use Case* yang mempunyai hubungan urutan yang boleh dikongsi?.
 - ii) *Use Case* yang mempunyai interaksi dengan banyak aktor perlu di pecahkan untuk mengelakkan *Use Case* yang terlalu banyak fungsi.
 - iii) *Use Case* tidak boleh beroperasi secara tersendiri tanpa tindakan daripada aktor atau dihubungkan dengan *Use Case* yang lain.
 - iv) *Use Case* yang tiada tindakan oleh aktor, bukan *Use Case* utama tetapi ianya sub *Use Case* yang akan menggambarkan urutan proses.
- b) Penentuan hubungan antara *Use Case* adalah berpandukan kepada 3 jenis hubungan seperti berikut:
- i) Pengkhususan (*Specializes*) – *Use Case* boleh mewarisi *Use Case* yang lain. Ini bertujuan untuk mengelak kompleksiti dan definisi semula langkah-langkah dan penambahan proses, kita boleh laksanakan pengkhususan kepada *Use Case* utama. Hubungan ini hanya untuk mengumpulkan *Use Case* yang berkaitan.

Contoh:

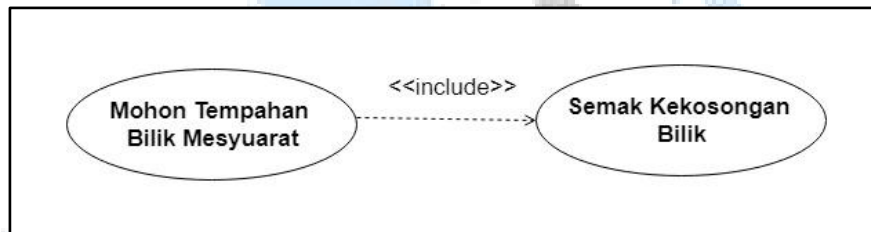
Tempahan Bilik Mesyuarat boleh dilakukan dengan 2 cara iaitu secara online yang disediakan atau tempahan boleh dilakukan secara emel kepada Pegawai Tadbir tempahan.



Rajah 35 : Contoh Hubungan Use Case (Pengkhususan)

- ii) <Include> – Use Case yang mempunyai langkah proses disediakan oleh Use Case yang lain dipanggil <include>. Ini bermaksud Use Case tersebut tidak boleh berfungsi dengan sendiri.

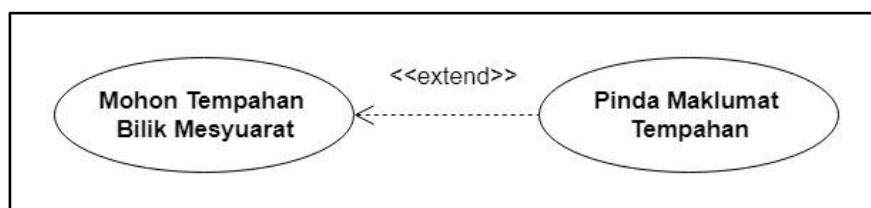
Contoh: Use Case Mohon Tempahan Bilik Mesyuarat perlu memanggil fungsi Semak kekosongan Bilik bagi melengkapkan proses.



Rajah 36 : Contoh Hubungan Use Case (Include)

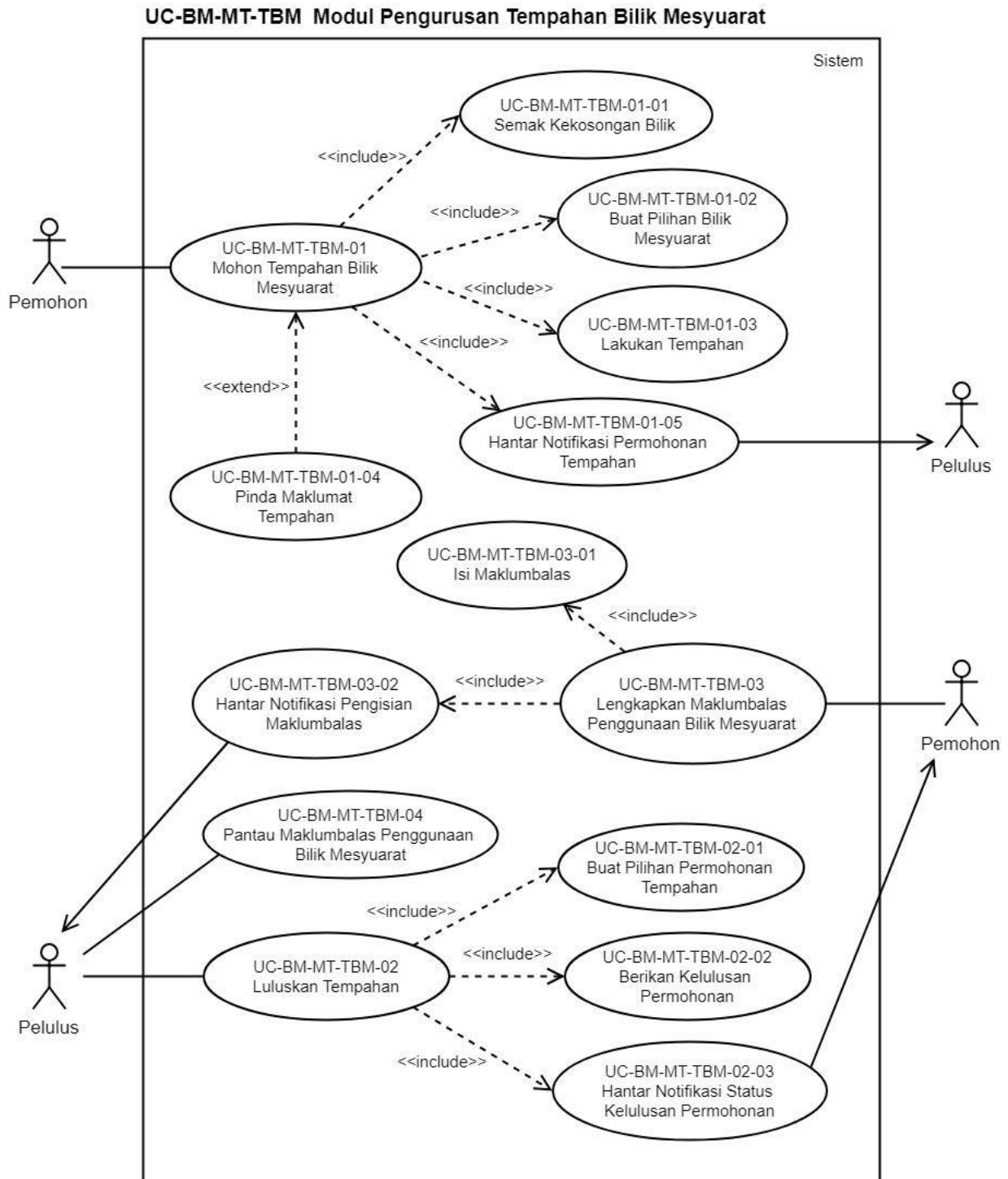
- iii) <Extend> – Use Case yang memberi nilai tambah kepada Use Case yang lain. Ianya digunakan bagi mentakrifkan sesuatu Use Case menggunakan fungsi dalam Use Case yang lain dengan syarat-syarat tertentu.

Contoh: Fungsi Pinda Maklumat Tempahan hanya akan digunakan sekiranya pemohon ingin melakukan perubahan kepada maklumat tempahan yang telah dibuat sebelum ini berkenaan.



Rajah 37 : Contoh Hubungan Use Case (Extend)

- c) Lakarkan keseluruhan Rajah Use Case yang dihasilkan selepas tindakan yang telah dilakukan seperti di atas. Contoh Rajah Use Case yang lengkap adalah seperti dalam rajah di bawah.



Rajah 38 : Contoh Rajah Use Case Modul Pengurusan Tempahan Bilik Mesyuarat

Langkah 5 : Dapatkan Pengesahan SME

- a) Semak Rajah *Use Case* yang dihasilkan dengan SME. Semakan perlulah mengambilkira
 - i) Lengkap – mengambilkira semua fungsi yang perlu
 - ii) Terperinci – peringkat perincian yang dihasilkan
 - iii) Hubungan antara *Use Case*
 - iv) Nama dan keterangan
 - v) Aktor-aktor yang terlibat

- b) Rajah *Use Case* yang dihasilkan perlulah difahami oleh kedua-dua pihak iaitu pasukan pembangunan dan SME iaitu:
 - i) Pasukan Pembangunan
Fahami fungsi sistem yang akan dibangunkan.

 - ii) SME
Fahami kaitan proses bisnes dengan fungsi yang dibangunkan.

Langkah 6 : Dokumentasikan Rajah *Use Case*

Dokumentasikan hasil Pemodelan *Use Case* (Rajah *Use Case* dan keterangan *Use Case*) ke dalam **D03 Spesifikasi Keperluan Sistem**

RUJUKAN

1. <https://www.uml-diagrams.org/use-case-subject.html>
2. Alan Dennis, Barbara Haley Wixom, David Tegarden (2012). *Systems Analysis and Design with UML Version 2.0*, Second Edition.

3.6 PEMODELAN FUNGSI SISTEM [F2.2]

KETERANGAN

Pemodelan Fungsi Sistem adalah teknik untuk mengenal pasti dan menstruktur fungsi-fungsi di dalam sistem yang ingin dibangunkan. Teknik ini adalah berpandukan kepada rajah-rajah *Use Case* yang telah disediakan dalam **Pemodelan Use Case [F2.1]** dan Rajah Hierarki Fungsi Bisnes di dalam **Pemodelan Fungsi Bisnes [F1.3]**.


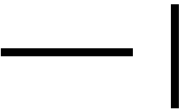
Hasil Pemodelan Fungsi Sistem direpresentasi dalam bentuk Rajah Hierarki Fungsian Sistem. Rajah ini memaparkan struktur sistem di mana tahap hierarkinya bermula dari sistem itu sendiri dan diikuti oleh subsistem-subsistem, modul-modul, submodul-submodul dan transaksi-transaksi yang berkaitan dengannya. Rajah Hierarki Fungsian Sistem akan diiringi dengan senarai peranan-peranan pengguna yang terlibat di bawah satu-satu transaksi sistem.

OBJEKTIF

- Menyediakan Rajah Hierarki Fungsian Sistem yang memaparkan struktur fungsi sistem yang ingin dibangunkan.
- Menyediakan Jadual Pemanduan Aktor dengan Fungsi Sistem pada tahap transaksi.

NOTASI

Jadual 20 : Notasi Fungsi Bisnes

Elemen	Keterangan
<p>Fungsi Sistem</p> 	<ul style="list-style-type: none"> • Digunakan bagi mewakili setiap fungsi-fungsi sistem yang terdiri dari sistem, subsistem, modul, submodul dan transaksi. • Labelkan ID Fungsi Sistem (rujuk notasi di bawah) dan Nama Fungsi Sistem di dalam ruangan notasi tersebut.
<p>Penghubung</p> 	<ul style="list-style-type: none"> • Penghubung di antara notasi Fungsi-fungsi Sistem menggunakan garis lurus mendatar atau menegak.

<p>ID Fungsi Sistem</p> <p>SF - SS - SBS - MM - SBM - 99 - 88</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama dan nombor bagi setiap Rajah Hierarki Fungsian Sistem yang dibangunkan. • Keterangan Kandungan ID: <ul style="list-style-type: none"> SF = Ringkasan Nama Teknik SS = Ringkasan Nama Sistem *SBS = Ringkasan Nama Subsistem *MM = Ringkasan Nama Modul *SBM = Ringkasan Nama Submodul *99 = Nombor Rujukan Transaksi *88 = Nombor Rujukan Subtransaksi <p>* hanya diperlukan sekiranya tahap hierarki Fungsi Sistem adalah merupakan subsistem, modul, submodul, transaksi atau subtransaksi.</p>
--	---

LANGKAH-LANGKAH

Langkah 1 : Terjemahkan Model Use Case Kepada Fungsi Sistem

- a) Fungsi-fungsi Sistem terbahagi kepada lima tahap, iaitu:
- i) Sistem
 - ii) Subsistem
 - iii) Modul
 - iv) Submodul
 - v) Transaksi

Namun begitu, satu-satu sistem tidak semestinya perlu merangkumi kelima-lima tahap tersebut oleh kerana pemilihan tahap Fungsi Sistem adalah bergantung kepada saiz dan juga kompleksiti aplikasi yang ingin dibangunkan. Bagi aplikasi yang bersaiz besar dan sangat kompleks, struktur aplikasinya berkemungkinan memerlukan kesemua tahap Fungsi Sistem seperti yang dinyatakan di atas. Manakala aplikasi yang bersaiz kecil pula perlu sekurang-kurangnya mengandungi struktur pada tahap sistem, modul dan transaksi sahaja.

- b) Pendekatan yang diambil untuk membangunkan Model Fungsi Sistem adalah secara *bottom-up* di mana ia dimulakan dengan mengenal pasti Fungsi Sistem di tahap transaksi dahulu.

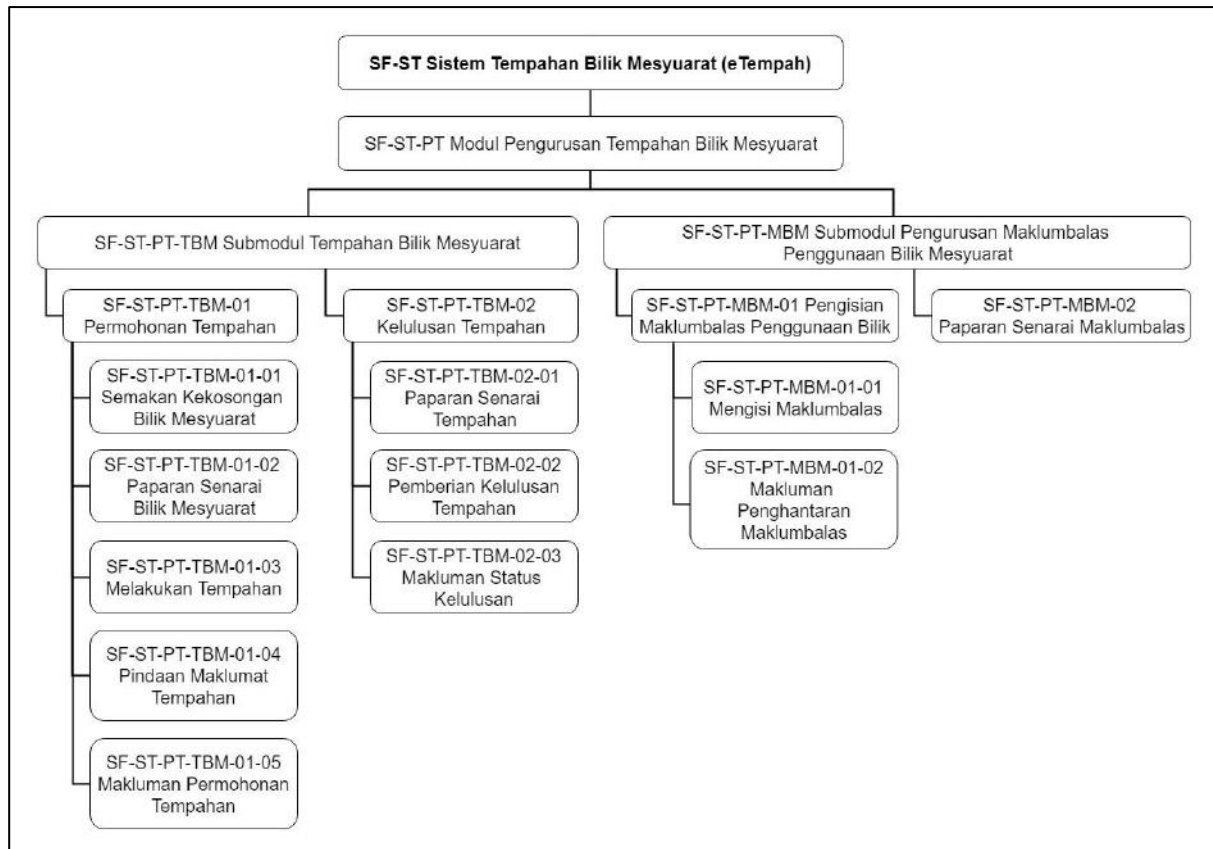
- c) Sekiranya sistem aplikasi yang hendak dibangunkan adalah besar dan kompleks, Model Fungsi Sistem boleh disediakan secara berasingan mengikut subsistem, modul atau submodul bagi tujuan kekemasan serta mengelakkan rajah hierarki menjadi terlalu besar dan kompleks.
- d) Dengan merujuk kepada semua *Use Case* yang telah disediakan di dalam Pemodelan *Use Case* [F2.1], kenal pasti setiap *Use Case* yang terlibat khususnya *Use Case* yang berinteraksi secara terus dengan aktor-aktor.
- e) Setiap *Use Case* yang berinteraksi secara terus dengan aktor akan terus diterjemahkan sebagai Fungsi Sistem pada tahap transaksi bagi satu-satu modul atau submodul. *Use Case* sebegini dikategorikan sebagai *Use Case* induk.
- f) *Use Case* yang tidak berinteraksi secara terus dengan aktor perlu juga dianggap sebagai salah satu komponen Fungsi Sistem pada tahap transaksi, tetapi kedudukan lokasinya mungkin berbeza dan berada di bawah modul atau submodul yang berlainan. *Use Case* sebegini berinteraksi dengan *Use Case* induk melalui hubungan seperti *include*, *extend* dan *generalization*. Kedudukan di mana komponen Fungsi Sistem ini diletakkan adalah bergantung sama ada *Use Case* sebegini adalah :
- i) **Transaksi Am**
 Transaksi Am merupakan komponen Fungsi Sistem yang diterjemahkan dari *Use Case* yang berhubung bukan sahaja dengan satu *Use Case* induk malah juga dengan *Use Case* induk yang lain di dalam modul/submodul yang berbeza. Kesemua Transaksi Am yang telah dikenalpasti akan dikumpulkan di bawah satu modul atau submodul yang umum. Sebagai contoh, transaksi seperti Carian Pengguna dan Muat Naik Dokumen akan diletakkan di bawah Modul atau Submodul Penyelenggaraan.
 - ii) **Transaksi Khusus**
 Transaksi Khusus merupakan komponen Fungsi Sistem yang diterjemahkan dari *Use Case* yang hanya berhubung dengan *Use Case* induk yang berkait dengannya sahaja dalam satu-satu transaksi atau submodul yang sama. *Use Case* tersebut tidak akan berkongsi dan berhubung dengan *Use Case* induk yang berada di bawah modul atau submodul yang lain. Kedudukan Transaksi Khusus ini akan diletakkan di bawah modul atau submodul yang ia terlibat.
- g) Setelah satu-satu Fungsi Sistem pada tahap transaksi selesai dilakukan, terjemahkan pula tajuk Rajah *Use Case* kepada Fungsi Sistem pada tahap modul atau/dan submodul. Gunakan Rajah Hierarki Fungsi Bisnes yang disediakan di Pemodelan Fungsi Bisnes [F1.3] sebagai rujukan untuk menentukan struktur dan kedudukan satu-satu Fungsi Sistem pada tahap modul dan submodul itu berada.
- h) Sekiranya sistem aplikasi yang dibangunkan adalah kompleks dan memerlukan kedua-dua Fungsi Sistem pada tahap modul dan juga submodul, pastikan semua submodul

yang telah dikenalpasti adalah berkaitan dan dikumpulkan di bawah modul yang relevan dengannya.

- i) Fungsi Sistem pada tahap subsistem hanya dikenalpasti sekiranya satu-satu sistem aplikasi yang dibangunkan itu adalah besar dan terdapat keperluan untuk sistem tersebut dipecahkan kepada sub-subsistem yang berasingan. Bagi kes ini, kumpulkan semua Fungsi Sistem pada tahap modul, submodul dan transaksi yang berkaitan dengan subsistem yang sesuai.
- j) Kandungan Fungsi Sistem pada tahap sistem adalah merupakan nama sistem aplikasi yang telah dipersetujui dan dipilih oleh kementerian/jabatan/agensi/organisasi yang terbabit. Perlu diingati bahawa Fungsi Sistem pada tahap ini hanya boleh terdiri dari satu komponen sahaja bagi satu-satu sistem aplikasi.

Langkah 2 : Sediakan Hubungan Di Antara Fungsi-fungsi Sistem

- a) Sebelum hubungan-hubungan dilukis, pastikan Fungsi-fungsi Sistem diletakkan dalam kedudukan hierarki berbentuk hibrid di mana pada kebiasaanya Fungsi-fungsi Sistem pada tahap sistem, subsistem dan modul ditempatkan secara menegak (*vertical*), manakala Fungsi Sistem pada tahap submodul dan transaksi pula berada dalam kedudukan mendatar (*horizontal*). Namun begitu, kedudukan Fungsi Sistem pada setiap tahap-tahap ini boleh berubah dan bergantung kepada kompleksiti dan saiz sistem aplikasi ataupun rajah hierarki yang ingin dibangunkan.
- b) Lakarkan hubungan di antara semua Fungsi-fungsi Sistem bermula dari tahap sistem, subsistem, modul, submodul sehingga ke tahap transaksi. Hubungan-hubungan ini dilukiskan dengan menggunakan notasi Penghubung sama ada ia merupakan garis mendatar atau menegak. Contoh hubungan di antara Fungsi-fungsi Sistem bagi Modul Pengurusan Tempahan di bawah Sistem Tempahan Bilik Mesyuarat (eTempah) adalah seperti rajah hierarki berikut:



Rajah 39 : Contoh Rajah Hierarki Fungsian Sistem Bagi Modul Pengurusan Tempahan Bilik Mesyuarat

Langkah 3 : Padankan Aktor Sistem Dengan Fungsi Sistem

- Rujuk kepada aktor-aktor dengan setiap *Use Case* yang berinteraksi di antara satu sama lain dalam Pemodelan *Use Case* (Fungsian) [F2.1] sebelum satu-satu pemadanan dilakukan. Pemadanan setiap aktor sistem dengan Fungsi Sistem dikenalpasti berdasarkan kepada hubungan semua aktor yang terlibat dengan satu-satu *use case*.
- Use Case* yang telah dikenalpasti kemudiannya diterjemahkan kepada Fungsi Sistem pada tahap transaksi, seperti yang telah diperjelaskan dalam Langkah 1.
- Aktor-aktor sistem yang telah dipadankan dengan setiap Fungsi Sistem pada tahap transaksi boleh dipapar sama ada dengan menggunakan jadual matriks ataupun jadual-jadual lain yang bersesuaian. Contoh jadual yang boleh diguna pakai dalam pemadanan aktor sistem dengan transaksi adalah seperti berikut:

Jadual 21 : Contoh Pemadanan Aktor Dengan Fungsi Sistem Bagi Submodul Tempahan Bilik Mesyuarat

Nama Modul : Modul Pengurusan Tempahan Bilik Mesyuarat			
Nama Submodul : Submodul Tempahan Bilik Mesyuarat			
Bil.	ID Fungsi Sistem	Nama Transaksi	Aktor Sistem
1	SF-ST-PT-TBM-01	Permohonan Tempahan	Pemohon Pelulus
2	SF-ST-PT-TBM-01-01	Permohonan Tempahan >> Semakan Kekosongan Bilik Mesyuarat	Pemohon
3	SF-ST-PT-TBM-01-02	Permohonan Tempahan >> Paparan Senarai Bilik Mesyuarat	Pemohon
4	SF-ST-PT-TBM-01-03	Permohonan Tempahan >> Membuat Tempahan	Pemohon
5	SF-ST-PT-TBM-01-04	Permohonan Tempahan >> Pindaan Maklumat Tempahan	Pemohon
6	SF-ST-PT-TBM-01-05	Permohonan Tempahan >> Makluman Permohonan Tempahan	Pemohon Pelulus
7	SF-ST-PT-TBM-02	Kelulusan Tempahan	Pemohon Pelulus
8	SF-ST-PT-TBM-02-01	Kelulusan Tempahan >> Paparan Senarai Tempahan	Pelulus
9	SF-ST-PT-TBM-02-02	Kelulusan Tempahan >> Pemberian Kelulusan Tempahan	Pelulus
10	SF-ST-PT-TBM-02-03	Kelulusan Tempahan >> Makluman Status Kelulusan	Pemohon Pelulus

Langkah 4 : Kemaskini ID Use Case

Kemaskini ID *Use Case* Sementara di dalam rajah-rajah *Use Case* yang telah dibangunkan di **Pemodelan Use Case [F2.2]** kepada ID *Use Case* seperti notasi di atas yang berlandaskan kepada struktur ID Fungsi Sistem yang telah disediakan.

Langkah 5 : Muktamadkan Model Fungsi Sistem Dengan SME

- a) Rajah Hierarki Fungsian Sistem dan Jadual Pemandanan Aktor dengan Fungsi Sistem yang telah disediakan perlu disemak dan disahkan oleh pihak SME. Semakan perlulah mengambil kira perkara-perkara seperti di bawah :
 - i) Lengkap dan terperinci
 - ii) Ketepatan Fungsi Sistem dan hubungannya dengan aktor sistem
- b) Pihak SME dan juga pasukan pembangunan perlu mempunyai kefahaman yang jelas semasa proses semakan berkenaan dengan Fungsi-fungsi Sistem terlibat merangkumi sistem, subsistem, modul, submodul dan transaksi yang ingin dibangunkan.

Langkah 6 : Sediakan Model Fungsi Sistem Secara Iteratif

Penyediaan dan pembangunan Model Fungsi Sistem perlu dilaksanakan secara iteratif bagi meningkatkan tahap komprehensif dan keperincian rajah hierarki dan maklumat yang direkodkan. Perubahan dan pindaan perlu dilakukan berdasar kepada sebarang tambahan baharu kepada keperluan bisnes dan sistem.

Langkah 7 : Dokumentasikan Model Fungsi Sistem

- a) Dokumen dan masukkan hasil Pemodelan Fungsi Sistem (Rajah Hierarki Fungsian Sistem dan Jadual Pemandanan Aktor dengan Fungsi Sistem) yang telah dibangunkan ke dalam D03 Spesifikasi Keperluan Sistem.
- b) Rujuk kepada Penyediaan Spesifikasi Keperluan Sistem [F2.6] bagi mengenalpasti ruangan di mana model tersebut perlu diletakkan.

RUJUKAN

1. Gary, B. S. & Harry, J. R. (2012). System Analysis and Design. Ninth Edition. Boston: Course Technology
2. Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems Analysis and Design. Fifth Edition. USA: John Wiley & Sons, Inc.
3. Wiegers, K., E. & Joy, B. (2013). Software Requirements. Third Edition. Washington: Microsoft Press



3.7 PEMODELAN KEPERLUAN DATA [F2.3]

KETERANGAN

Analisis Keperluan Data ialah kaedah yang digunakan untuk memodelkan maklumat yang diperlukan oleh sistem aplikasi. Model maklumat ini menerangkan jenis maklumat yang diperlukan, pola capaian maklumat dan peraturan bisnes yang menghubungkan antara maklumat-maklumat tersebut. Teknik yang digunakan bagi perwakilan model maklumat ialah Rajah Hubungan Entiti, atau lebih dikenali sebagai Entity Relationship Diagram (ERD). Rajah ER yang dihasilkan merupakan model maklumat konseptual yang bebas daripada sebarang teknologi dan penyelesaian teknikal. Model ini menerangkan komponen data yang terdiri daripada entiti, atribut dan hubungan di antaranya. Bilangan entiti, atribut dan hubungan di antaranya akan digunakan untuk membuat pengiraan saiz sistem aplikasi dari aspek keperluan data.

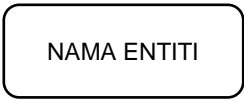
OBJEKTIF


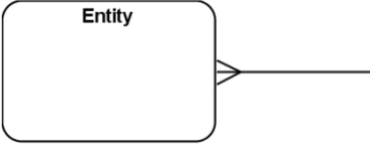


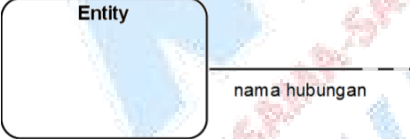
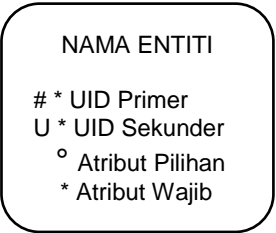
- Menyediakan model maklumat yang tepat berdasarkan keperluan di sesebuah organisasi. Model maklumat ini boleh dijadikan rangka kerja/panduan untuk menaiktaraf aplikasi sedia ada, atau pembangunan sistem aplikasi yang baharu di organisasi berkenaan.
- Menyediakan model yang tidak bersandar dengan mana-mana pelaksanaan teknikal dan teknologi.
- Mengumpul, mendefinisikan dan memahami elemen-elemen maklumat yang diperlukan dan hubungkait di antaranya.
- Mendapatkan saiz sistem aplikasi dari aspek fungsi data (*data function*).

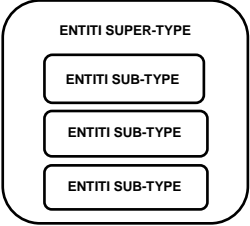
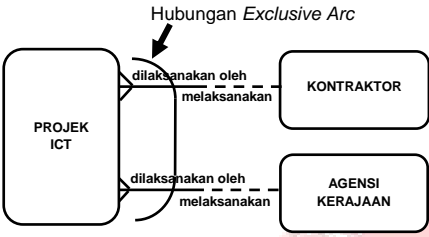
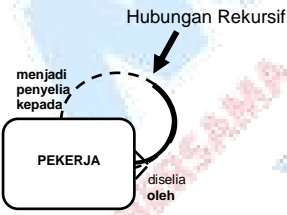
NOTASI

Pemodelan ERD adalah berdasarkan notasi Richard Barker:

Jadual 22 : Notasi Rajah Hubungan Data

Elemen	Keterangan
Entiti 	<ul style="list-style-type: none"> • Digunakan bagi mewakili setiap entiti • Entiti adalah objek yang signifikan, sama ada nyata atau tidak di mana maklumat mengenainya perlu disimpan. • Nama entiti ditulis dengan huruf besar (<i>uppercase</i>) dan dalam bentuk tunggal (<i>singular</i>).

<p>Kardinaliti : Tahap hubungan 'satu dan hanya satu'</p> 	<p>Kardinaliti 'berkaki satu' menunjukkan tahap hubungan (<i>degree of relationship</i>) 'satu dan hanya satu' dengan entiti lain.</p>
<p>Kardinaliti : Tahap hubungan 'satu atau lebih'</p> 	<p>Kardinaliti 'berkaki tiga' (<i>crowsfoot</i>) menunjukkan tahap hubungan (<i>degree of relationship</i>) 'satu atau lebih' dengan entiti lain.</p>
<p>Modaliti : hubungan mandatori</p> 	<ul style="list-style-type: none"> • Modaliti mandatori menunjukkan sifat kebergantungan (<i>dependence</i>) bagi hubungan antara dua entiti.
<p>Modaliti : hubungan pilihan</p> 	<ul style="list-style-type: none"> • Modaliti pilihan menunjukkan sifat tiada kebergantungan (<i>independence</i>) bagi hubungan antara dua entiti.
<p>Nama hubungan</p> 	<ul style="list-style-type: none"> • Nama hubungan yang menunjukkan hubung kait antara dua entiti.
<p>Atribut</p> 	<ul style="list-style-type: none"> • Atribut adalah ciri-ciri yang menggambarkan sesebuah entiti. Contoh atribut adalah id, nama, keterangan, tarikh dan lain-lain. • Jenis-jenis atribut adalah seperti berikut: <ul style="list-style-type: none"> ○ Pengenal Unik (UID) Primer diwakili dengan notasi # ○ Pengenal Unik (UID) Sekunder diwakili dengan notasi U

	<ul style="list-style-type: none"> ○ Atribut Mandatori (<i>Mandatory</i>) iaitu maklumat wajib diisi semasa pengwujudan <i>instance</i> bagi entiti tersebut diwakili dengan notasi * ○ Atribut Pilihan (<i>Optional</i>) iaitu maklumat tidak wajib diisi pengwujudan <i>instance</i> bagi entiti tersebut diwakili dengan notasi °
<p>Entiti <i>Super-type</i> / <i>Sub-type</i></p> 	<ul style="list-style-type: none"> • Entiti <i>Super-type</i> merupakan entiti generik (umum) yang mempunyai hubungan dengan satu atau lebih entiti sub-type • Entiti <i>Sub-type</i> merupakan sub bagi kumpulan entiti yang terdapat dalam entiti <i>Super-type</i>.
<p>Hubungan <i>Exclusive Arc</i></p> 	<ul style="list-style-type: none"> • Hubungan <i>Exclusive Arc</i> menunjukkan hubungan <u>atau</u> di antara dua entiti. • Hubungan ini diwakili dengan notasi garis separuh melengkung ')’.
<p>Hubungan Rekursif</p> 	<ul style="list-style-type: none"> • Hubungan Rekursif merupakan model hierarki di mana hubungan wujud pada entiti yang sama.

LANGKAH-LANGKAH

Langkah 1 : Kenalpasti Entiti, Atribut, Pola Capaian Dan Peraturan Bisnes (*Business Rule*)

- a) Kenal pasti dan senaraikan calon entiti dan atribut daripada proses pemodelan fungsi bisnes (rujuk Apendiks 2 Definisi Aktiviti Fungsi Bisnes).

Contoh cabutan sebahagian daripada kes bisnes:

“Maklumat bilik mesyuarat perlu didaftarkan dan dikemaskini melalui sistem. Pengguna sistem boleh menyemak jadual tempahan dan fasiliti bilik mesyuarat serta melakukan permohonan tempahan bilik mesyuarat secara melalui sistem.”

- b) Perkataan yang bergaris merupakan perkataan sebagai calon entiti dan atribut yang boleh digunakan dalam pemodelan maklumat. Maklumat yang dikenalpasti diterjemahkan kepada elemen pemodelan maklumat.
- c) Pola capaian merujuk kepada bagaimana operasi CRUD dilaksanakan ke atas data.
- d) Perkasa asas (*fundamental*) dan penting dalam pemodelan maklumat ialah memahami peraturan bisnes dalam konteks sistem yang hendak dibangunkan. Maklumat mengenai bagaimana sesuatu sistem itu bertindak dalam situasi tertentu dikenali sebagai peraturan bisnes (contohnya, sekiranya pengguna hendak membuat tempahan aset tetapi tidak memilih sebarang aset untuk ditempah, maka permohonan tersebut akan terbatal).
- e) Peraturan bisnes juga merujuk kepada ketetapan/syarat yang perlu dipatuhi, termasuklah peraturan yang ditetapkan mengikut sesuatu akta, pekeliling dan undang-undang yang berkuatkuasa.

Jadual 23 : Contoh Entiti, Atribut, Pola Capaian Dan Peraturan Bisnes

Entiti	PENGGUNA
Atribut	i) id pengguna ii) nama pengguna iii) emel pengguna iv) nombor telefon pengguna v) alamat pengguna

Pola Capaian	<p><i>Create (C):</i></p> <ul style="list-style-type: none"> i) Pengguna perlu diwujudkan terlebih dahulu sebelum boleh membuat sebarang tempahan aset ii) Pengguna mesti didaftarkan bawah satu bahagian
Peraturan Bisnes	<ul style="list-style-type: none"> i) Hanya pengguna berdaftar sahaja yang boleh membuat tempahan aset ii) Dalam satu masa, pengguna boleh menempah banyak aset sekaligus iii) Sekiranya pengguna hendak membuat tempahan aset, tetapi tidak memilih sebarang aset untuk ditempah, permohonan tersebut akan terbatal.

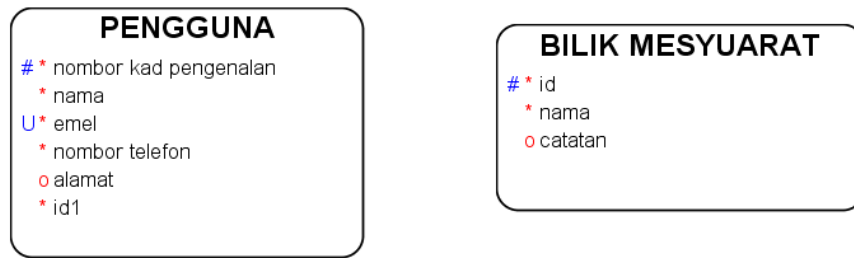
Langkah 2 : Bina Model Maklumat

- a) Lukiskan setiap entiti tunggal (bukan jenis Entiti *super-type* dan *sub-type*) yang telah dikenalpasti. Contoh entiti adalah seperti rajah berikut.



Rajah 40 : Contoh Entiti

- b) Senaraikan atribut yang telah dikenalpasti berpandukan peraturan-peraturan yang berikut.
- i) Nama entiti dalam bentuk tunggal (*singular*), manakala atribut biasanya ditulis dengan huruf kecil.
 - ii) Simbol * diletakkan di hadapan atribut yang dipilih menjadi Atribut Mandatori (wajib diisi).
 - iii) Simbol ° diletakkan di hadapan atribut yang dipilih menjadi Atribut Pilihan (tidak wajib diisi).
 - iv) Simbol # diletakkan di hadapan atribut yang dipilih menjadi UID Primer (UID yang paling kerap digunakan).
 - v) Simbol U diletakkan di hadapan atribut yang dipilih menjadi UID Sekunder (UID selain daripada UID Primer).

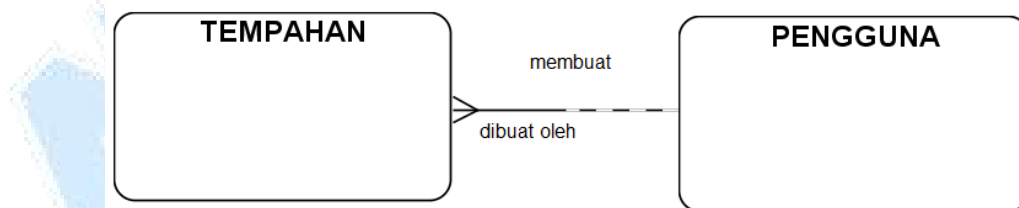


Rajah 41 : Contoh Entiti Dan Senarai Atribut

- vi) Dalam satu entiti, satu atau lebih atribut boleh dijadikan Pengenal Unik (Unique Identifier-UID).
- vii) UID/Kombinasi UID yang paling kerap digunakan akan menjadi UID Primer, manakala UID/Kombinasi UID yang lain akan menjadi UID Sekunder.
- viii) Maklumat yang disimpan dalam UID/Kombinasi UID mestilah unik (maklumat yang sama tidak boleh disimpan dua kali).

Langkah 3 : Kenal Pasti Hubungan Antara Entiti

- a) Berdasarkan peraturan bisnes yang disenaraikan, kenalpasti hubungan antara entiti-entiti tersebut dalam aspek kardinaliti dan modaliti.
- b) Tentukan nama hubungan yang sesuai antara entiti-entiti tersebut. Contoh diberikan seperti rajah dibawah.



Rajah 42 : Contoh Nama Hubungan Entiti

Jika dibaca dari kiri ke kanan:

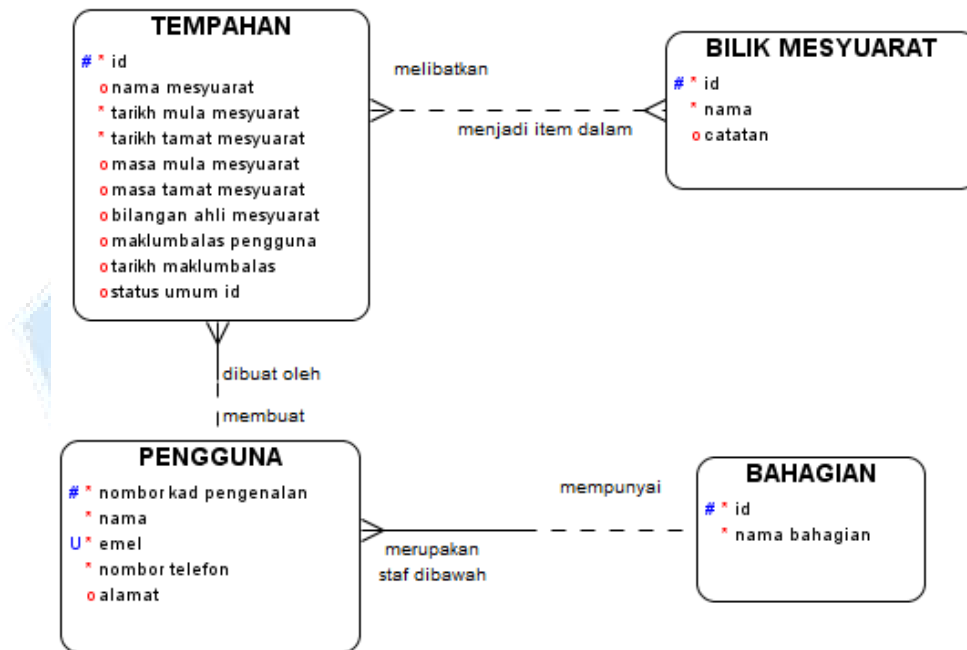
- i) Kardinaliti hubungan menunjukkan setiap TEMPAHAN dibuat oleh satu, dan hanya satu PENGGUNA. Ini bermaksud pada satu-satu tarikh dan masa tertentu, tempahan aset hanya boleh dibuat oleh seorang pengguna sahaja.
- ii) Dari segi modaliti, hubungan adalah mandatori (mesti) iaitu rekod TEMPAHAN tidak boleh diwujudkan tanpa wujudnya rekod PENGGUNA.

- iii) Bacaan adalah 'setiap TEMPAHAN mesti dibuat oleh satu, dan hanya satu PENGGUNA'.

Jika dibaca dari kanan ke kiri:

- i) Kardinaliti hubungan menunjukkan setiap PENGGUNA membuat satu atau lebih TEMPAHAN.
 - ii) Dari segi modaliti, hubungan adalah pilihan. Ini bermaksud rekod pengguna boleh wujud tanpa membuat sebarang tempahan bilik mesyuarat, dan beberapa tempahan bilik mesyuarat boleh dibuat oleh pengguna.
 - iii) Bacaan adalah 'setiap PENGGUNA mungkin membuat satu atau lebih TEMPAHAN'.
- c) Lengkapkan ERD dengan melukis hubungan di antara dua entiti, dan pastikan setiap hubungan dilengkapi dengan tiga perkara asas berikut iaitu nama hubungan (ditulis dalam huruf kecil), kardinaliti dan modaliti.

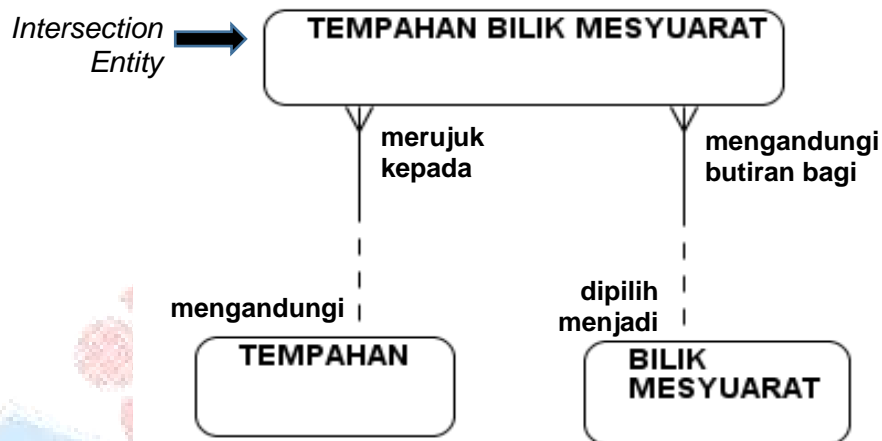
Contoh ERD yang dihasilkan adalah seperti yang di bawah.



Rajah 43 : Contoh Rajah Hubungan Entiti (ERD)

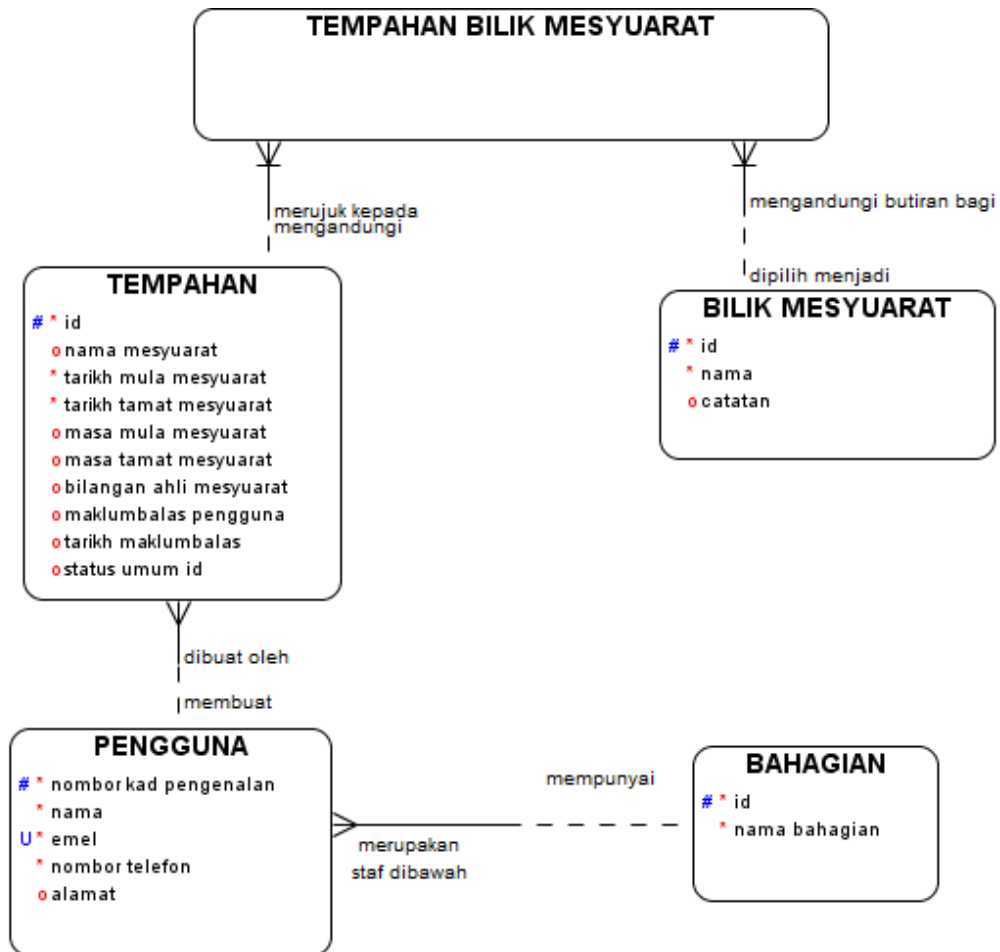
Langkah 4 : Perkemaskan Model Maklumat

- a) Setelah model maklumat siap dilukis, semak sekali lagi model ini dengan maklumat yang terdapat dalam proses pemodelan fungsi bisnes (rujuk Apendiks 2 Definisi Aktiviti Fungsi Bisnes dan Apendiks 5 Definisi Aliran Data) bagi memastikan semua maklumat yang diperlukan tidak tercicir. Lengkapkan ERD tersebut dengan menambah entiti-entiti yang baru (jika ada).
- b) Perkemaskan lagi ERD tersebut dengan menggunakan teknik-teknik di bawah sekiranya perlu:
 - i) Wujudkan *Intersection Entity* bagi menyelesaikan hubungan banyak-ke-banyak (*many-to-many*) di antara dua entiti. Contoh adalah seperti rajah dibawah.



Rajah 44 : Contoh *Intersection Entity*

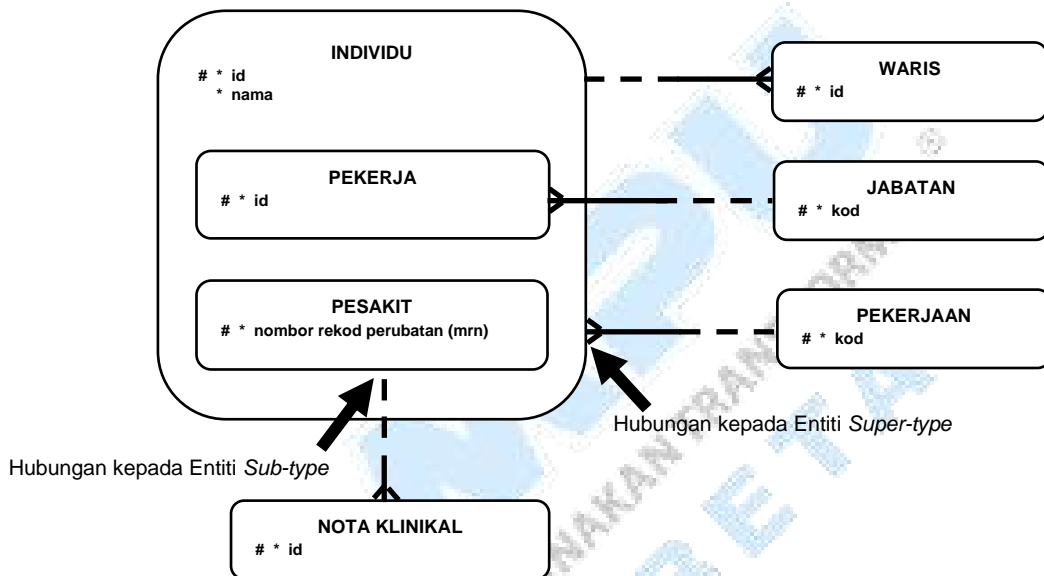
Contoh ERD yang telah diperkemas dengan mewujudkan *Intersection Entity* 'TEMPAHAN BILIK MESYUARAT' bagi menyelesaikan hubungan banyak-ke-banyak (*many-to-many*) adalah seperti berikut.



Rajah 45 : Contoh Hubungan Entiti (ERD) – *Intersection Entity*

- ii) Wujudkan Entiti *Super-type* dan *Sub-type* jika terdapat entiti dalam kumpulan yang sama. Lukis juga hubungan di antara Entiti *Super-type* dan *Sub-type* ini kepada entiti yang lain. Hubungan dengan entiti lain boleh sama ada kepada Entiti *Super-type* atau *Sub-type*.

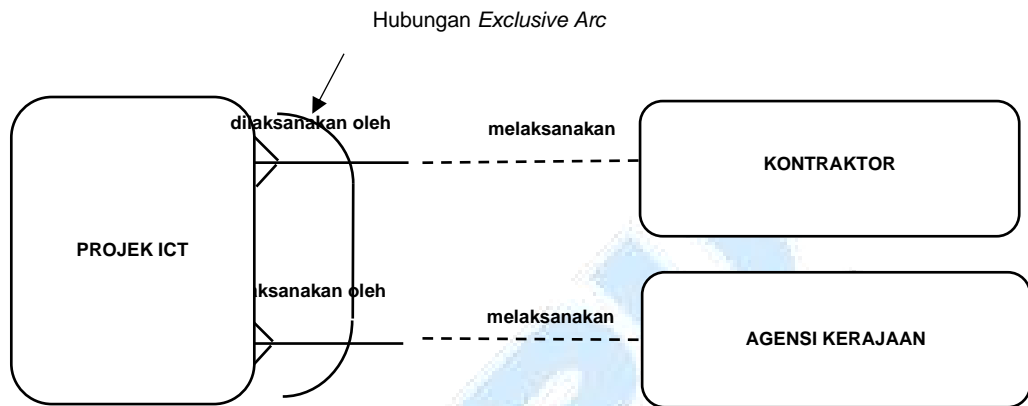
Contoh hubungan entiti lain kepada Entiti *Super-type* atau *Sub-type* adalah seperti rajah dibawah.



Rajah 46 : Contoh Hubungan Entiti Lain Kepada Entiti *Super-type*

- Hubungan dengan entiti lain boleh sama ada kepada Entiti *Super-type* atau *Sub-type*.
- Entiti ini dibaca seperti berikut:
 - Setiap INDIVIDU mestilah sama ada dalam kategori PEKERJA, PESAKIT atau KONTRAKTOR.
 - PESAKIT merupakan seseorang INDIVIDU

- iii) Wujudkan Hubungan *Exclusive Arc*. Hubungan *Exclusive Arc* menunjukkan hubungan atau di antara dua entiti. Hubungan ini diwakili dengan notasi garis separuh melengkung ')'.

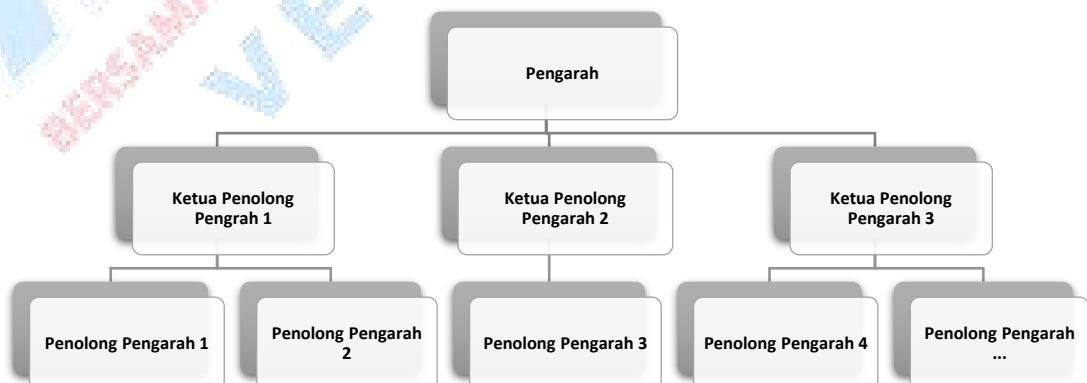


Rajah 47 : Contoh Hubungan Exclusive Arc

Cara pembacaan adalah:

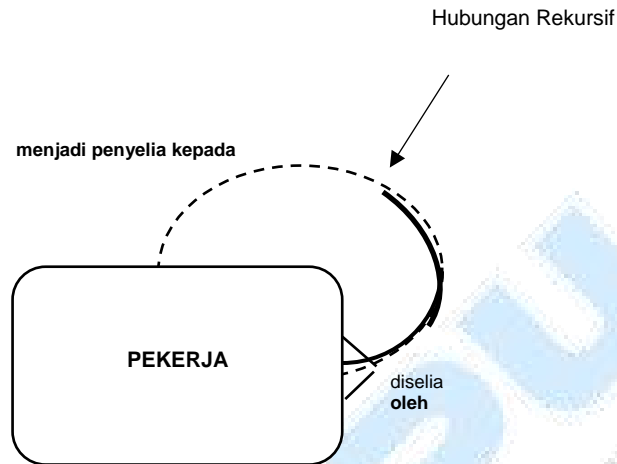
Setiap PROJEK ICT mesti dilaksanakan oleh satu, dan hanya satu KONTRAKTOR atau dilaksanakan oleh satu, dan hanya satu AGENSI KERAJAAN

- iv) Wujudkan Hubungan Rekursif. Hubungan Rekursif merupakan model hierarki di mana hubungan wujud pada entiti yang sama. Contohnya adalah seperti carta organisasi seperti di bawah:



Rajah 48 : Contoh Carta Organisasi

Dalam model maklumat, hubungan yang wujud pada entity yang sama digambarkan seperti rajah dibawah.



Rajah 49 : Contoh Hubungan Rekursif

Cara pembacaan adalah:

- Setiap PEKERJA mungkin menjadi penyelia kepada satu atau lebih PEKERJA
- Setiap PEKERJA mesti diselia oleh satu, dan hanya satu PEKERJA

Langkah 5 : Merekod Definisi Kamus Data

Penerangan terperinci mengenai ERD perlu dijelaskan dalam definisi kamus data. Sila rujuk **Apendiks 3 Definisi Kamus Data**. Definisi Kamus Data adalah untuk sistem yang dimodelkan perlu terdiri daripada definisi entiti dan definisi atribut.

Langkah 6 : Muktamadkan Model Maklumat Dengan SME

Bergantung kepada saiz dan kompleksiti projek, aktiviti ini mungkin memerlukan satu sesi bengkel dengan pengguna atau SME untuk memuktamadkan model maklumat tersebut.

Langkah 7 : Pengiraan Saiz Fungsi Data (*Data Function*)

Berpandukan elemen-elemen dan peraturan-peraturan hubungan yang telah dikenalpasti dalam modelan maklumat, pengiraan saiz sistem dari aspek keperluan Fungsi Data dapat dikira. Kaedah pengiraan ini boleh dirujuk dalam **Bab 8 Pengiraan Saiz Sistem Aplikasi**.

Langkah 8 : Dokumentasikan Model Maklumat Konseptual

Dokumentasikan semua output yang dihasilkan sebagai hasil serahan proses analisis keperluan data ke dalam **D03 Spesifikasi Keperluan Sistem**. Dokumentasikan mengikut susunan seperti berikut:

- a) Rajah Hubungan Entiti (ERD)
- b) Definisi Kamus Data (Entiti dan Atribut)

RUJUKAN

1. Richard Barker; CASE*Method: Entity Relationship Modeling (1990)
2. Graeme C. Simsion and Graham C. Witt; Data Modeling Essentials (2005)

3.8 PEMODELAN PROSES SISTEM [F2.4]

KETERANGAN

Permodelan Proses Sistem bertujuan untuk menyusun atur dan mendokumentasikan struktur dan aliran data antara proses logikal. Teknik pemodelan proses yang digunakan dalam metodologi ini adalah *Data Flow Diagram* (DFD). DFD menerangkan pergerakan, pertukaran atau transformasi data antara fungsi-fungsi bisnes. Terdapat dua peringkat DFD yang biasa digunakan iatu:

- a) Rajah Konteks (Aras Tinggi) : menunjukkan skop sistem dan interaksi dengan subsistem.
- b) DFD Aras n: menunjukkan kebergantungan fungsi-fungsi sistem


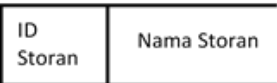
OBJEKTIF

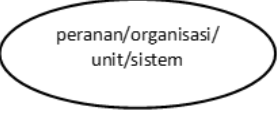
- Menghasilkan model proses sistem yang menerangkan pemprosesan yang akan dilaksanakan oleh sistem
- Mengenal pasti sumber data dan destinasi maklumat yang diperlukan oleh proses
- Menetapkan sempadan proses dan sistem yang dibangunkan

NOTASI

Permodelan DFD adalah berdasarkan notasi Gane & Sarson.

Jadual 24 : Notasi Rajah Aliran Data

Elemen	Keterangan
<p>Fungsi</p> 	<ul style="list-style-type: none"> • Mewakili tindakan yang dilaksanakan ke atas data sama ada ditransformasi, disimpan atau diagihkan ke proses lain untuk menghasilkan output. • Nama Fungsi hendaklah dimulakan dengan kata kerja dan mempunyai sekurang-kurangnya satu aliran input dan satu aliran output. • Minimum mempunyai SATU aliran maklumat input dan SATU aliran maklumat output.
<p>Storan Data</p> 	<ul style="list-style-type: none"> • Storan untuk penyimpanan data yang diterjemahkan daripada entiti ERD. • Nama Storan dilabelkan dengan menggunakan kata kerja berhuruf besar.

<p>Pengulangan Storan Data</p> 	<ul style="list-style-type: none"> • MESTI berhubung dengan fungsi melalui aliran maklumat (input, output atau kedua-duanya). • TIDAK boleh berhubung terus dengan entiti luaran tanpa melalui fungsi. • Pengulangan storan data ditunjukkan melalui garis menegak untuk setiap pengulangan yang dibuat
<p>Aliran Maklumat</p> <p>ID Aliran Maklumat Nama Aliran Maklumat</p> 	<ul style="list-style-type: none"> • Mewakili arah pergerakan maklumat antara entiti luaran, fungsi dan storan data atau sebaliknya. • MESTI menunjukkan satu arah aliran sahaja untuk setiap aliran maklumat input/output.
<p>Entiti Luaran</p>  <p>Pengulangan Entiti Luaran</p> 	<ul style="list-style-type: none"> • Mewakili sumber dan/atau destinasi maklumat kepada proses yang terdiri dari entiti lain seperti individu, peranan, unit organisasi, organisasi lain atau sistem-sistem lain yang berinteraksi dengan sistem. • MESTI berhubung dengan fungsi melalui aliran maklumat (input, output atau kedua-duanya). • Pengulangan entiti luaran ditunjukkan melalui garis menegak untuk setiap pengulangan yang dibuat
<p>ID Rajah Konteks</p> <p>DFD - SS</p>	<ul style="list-style-type: none"> • Digunakan sebagai konvensyen nama dan nombor bagi Rajah Konteks yang dibangunkan. • Keterangan Kandungan ID: DFD = Ringkasan Nama Teknik SS = Ringkasan Nama Sistem
<p>ID Fungsi (ID DFD Aras n / n-1)</p> <p>DFD - SS - SBS - MM - SBM - 99 - 88</p>	<p>Digunakan sebagai konvensyen nama dan nombor bagi setiap DFD aras n dan n-1 yang dibangunkan.</p> <p>Keterangan Kandungan ID: DFD = Ringkasan Nama Teknik SS = Ringkasan Nama Sistem SBS = Ringkasan Nama Subsistem MM = Ringkasan Nama Modul</p>

	<p>SBM = Ringkasan Nama Submodul *99 = Nombor Rujukan Transaksi *88 = Nombor Rujukan Subtransaksi</p> <p>* hanya diperlukan sekiranya tahap hierarki Fungsi Sistem adalah merupakan subsistem, modul, submodul, transaksi atau subtransaksi.</p>
<p>ID Storan</p> <p>DS - 99</p>	<ul style="list-style-type: none"> Digunakan sebagai konvensyen nama dan nombor bagi setiap storan data dibangunkan. Keterangan Kandungan ID: <ul style="list-style-type: none"> DS = Ringkasan Nama Storan Data 99 = Nombor Rujukan Storan Data
<p>ID Aliran Data</p> <p>DF - 99</p>	<ul style="list-style-type: none"> Digunakan sebagai konvensyen nama dan nombor bagi setiap DFD aras n yang dibangunkan. Keterangan Kandungan ID: <ul style="list-style-type: none"> DF = Ringkasan Aliran Data 99 = Nombor Rujukan Aliran Data

LANGKAH-LANGKAH

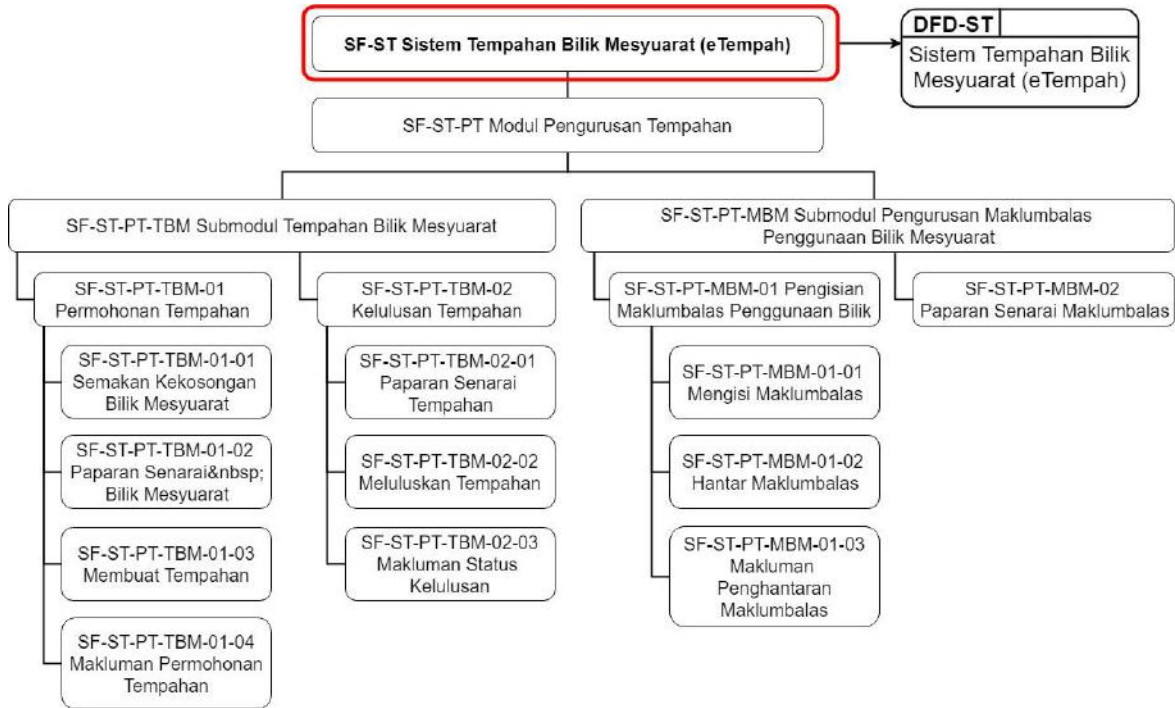
Langkah 1 : Sediakan Rajah Konteks

Langkah pertama dalam pemodelan proses sistem ialah penyediaan Rajah Konteks yang merupakan gambaran secara kasar bagi keseluruhan skop sistem. Terdapat 3 perkara utama dalam sesebuah rajah konteks:

- Kesemua entiti luaran yang terlibat
- Fungsi yang mewakili keseluruhan skop sistem
- Aliran maklumat input/output antara entiti luaran dan fungsi

Langkah-langkah untuk menyediakan Rajah Konteks adalah seperti berikut:

- Ambil fungsi aras paling atas daripada rajah hierarki fungsi sistem yang dihasilkan dari Pemodelan Fungsi Sistem [F2.2] dan lukiskan notasi fungsi sebagai mewakili sistem aplikasi yang hendak dibangunkan.
- Labelkan id fungsi serta nama fungsi dalam notasi fungsi berkenaan. Contoh pemetaan daripada rajah hierarki fungsi sistem bagi melukis notasi fungsi adalah seperti ditunjukkan dalam rajah di bawah.



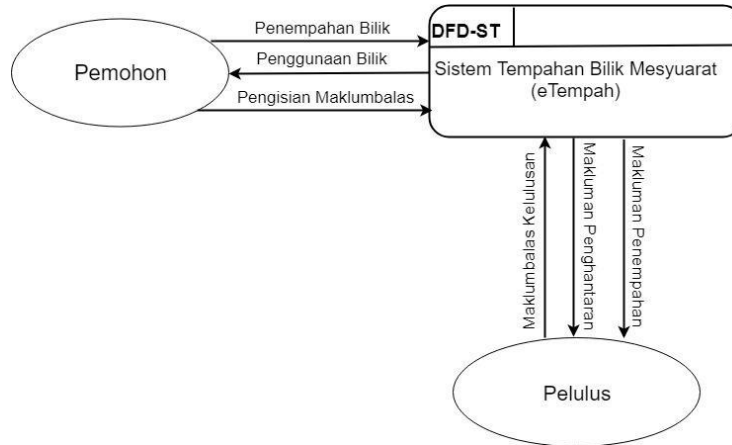
Rajah 50 : Contoh Pemetaan Hierarki Fungsi Sistem kepada Notasi Fungsi

- c) Terjemahkan 'aktor' daripada jadual *Pemadanan Aktor dengan Fungsi Sistem*, rujuk Pemodelan Fungsi Sistem [F2.2] kepada notasi entiti luaran. Contoh seperti Rajah di bawah



Rajah 51 : Contoh Lakaran Fungsi dan Entiti Luaran

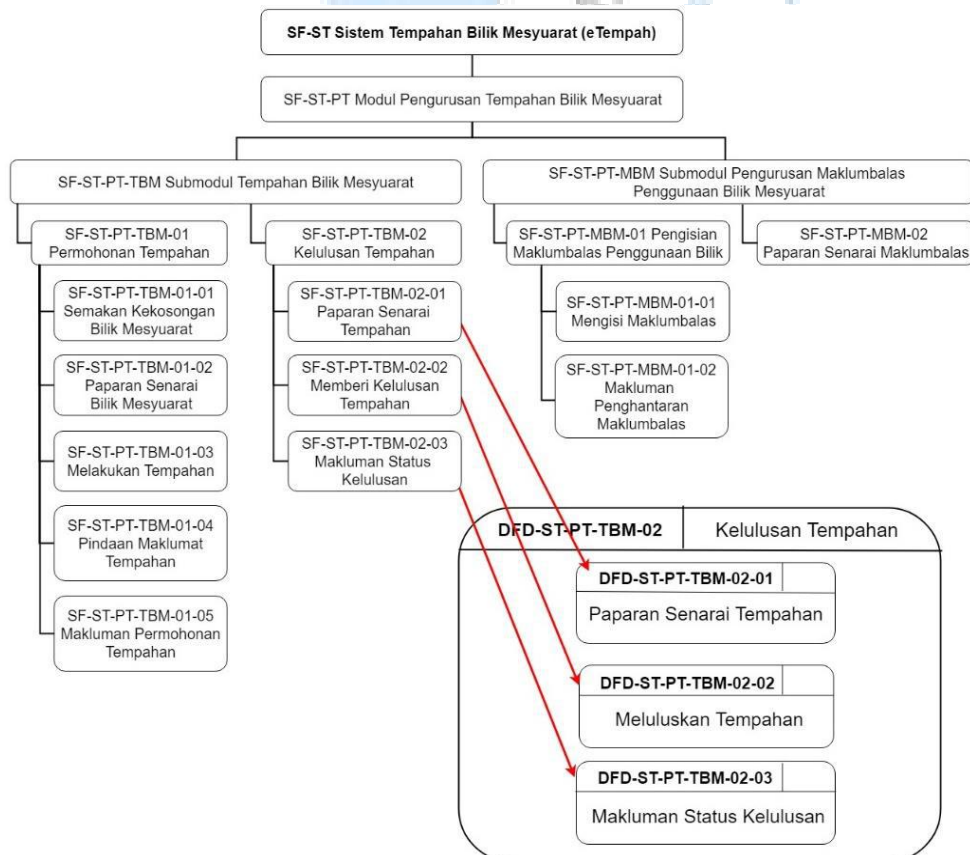
- d) Hubungkan antara entiti luaran dan fungsi melalui penggunaan notasi aliran maklumat. Hubungan di antara entiti luaran dan fungsi serta input dan output yang terlibat boleh dianalisis daripada keterangan di dalam Definisi Fungsi Bisnes yang dihasilkan melalui Pemodelan Proses Bisnes [F1.4]. Contoh Rajah Konteks DFD yang dihasilkan adalah seperti rajah di bawah.



Rajah 52 : Contoh Rajah Konteks DFD

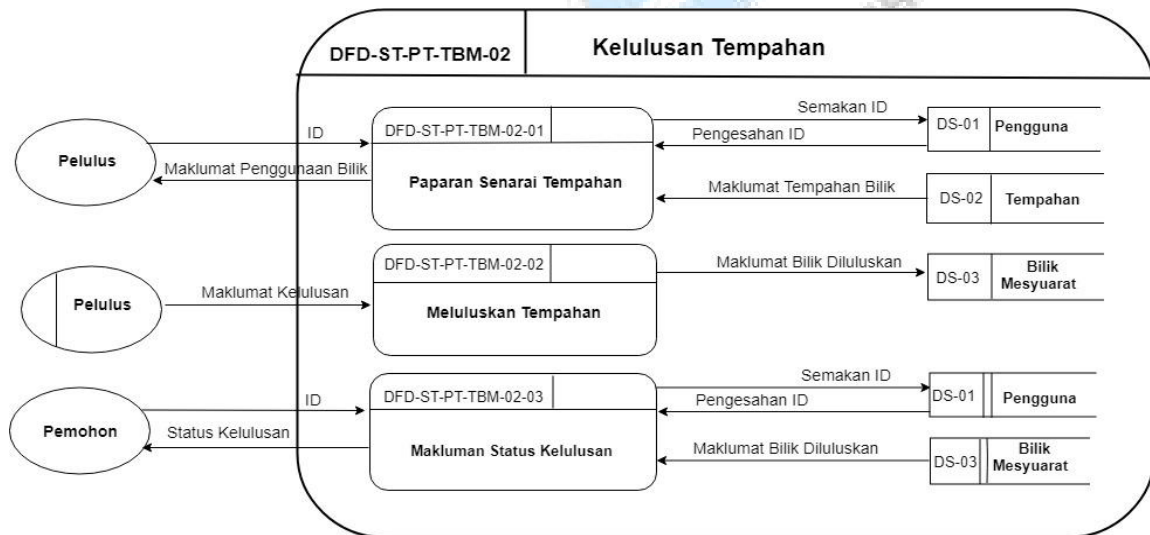
Langkah 2 : Menyediakan DFD Aras n

- a) Pemetaan hierarki fungsi sistem yang dihasilkan dari Pemodelan Fungsi Sistem [F2.2] kepada notasi fungsi DFD dilakukan dengan memilih fungsi transaksi seperti yang ditunjukkan dalam rajah di bawah.



Rajah 53 : Contoh Pemetaan Hierarki Fungsi Sistem kepada Notasi Fungsi DFD

- b) Tentukan juga storan data yang terlibat berdasarkan ERD dalam Pemodelan Keperluan Data [F2.3].
- Lukiskan storan data yang telah dikenal pasti dan labelkan setiap storan tersebut seperti yang telah dinyatakan dalam notasi di atas.
 - Mana-mana storan data yang diselenggara sepenuhnya serta digunakan dalam fungsi, mesti dilukis dalam rangka. Manakala data storan yang lain dilukis di luar rangka.
 - Sekiranya perlu, ulang melukis entiti luaran dan storan data bagi mengelakkan banyak aliran data yang bersilang.
- c) Hubungkan entiti luaran dan storan data ke fungsi dengan menggunakan aliran maklumat bagi menunjukkan pergerakan maklumat yang berlaku antaranya. Labelkan setiap aliran maklumat tersebut seperti dalam di bawah:



Rajah 54 : Contoh Data Flow Diagram (DFD)

Ulang langkah-langkah di atas untuk fungsi transaksi yang lain termasuk fungsi sistem pada tahap transaksi yang umum sekiranya perlu.

Langkah 3: Dokumentasikan Definisi DFD

- Penuhi templat Definisi Aliran Data dengan mengisi maklumat berikut. Sila rujuk templat di Apendiks 5 Definisi Aliran Data.

Jadual 25 : Keterangan Medan-Medan Definisi Aliran Data

Medan	Keterangan
Nama Aliran Data	Nama aliran data adalah merupakan nama yang diberikan untuk aliran data yang dibuat.
Sumber	Sumber aliran data iaitu sama ada daripada entiti luaran, fungsi atau storan data.
ID>Nama	Label ID atau nama bagi entiti luaran, fungsi atau storan data.
Destinasi	Destinasi aliran data iaitu sama ada ke entiti luaran, fungsi atau storan data.
Atribut	Atribut adalah merupakan atribut yang digunakan dalam fungsi tersebut.
CRUD	Operasi tertentu yang dibenarkan untuk setiap atribut.

- b) Pengemaskinian juga perlu dilakukan kepada maklumat di dalam Pemodelan Fungsi Sistem [F2.2] berdasarkan input yang diperolehi dalam penyediaan DFD.

Langkah 4: Memuktamadkan Model DFD Dengan SME

Semak DFD yang dihasilkan dengan SME DFD yang dihasilkan perlulah difahami oleh kedua-dua pihak iaitu pasukan pembangunan dan SME.

Langkah 5 : Dokumentasikan DFD

Dokumentasikan semua output yang dihasilkan sebagai hasil serahan analisis proses sistem ke dalam D03 Spesifikasi Keperluan Sistem. Dokumentasikan mengikut susunan berikut:

- a) DFD Model - Data Flow Diagram
- b) Definisi DFD

RUJUKAN

1. Joseph S.Valacich, Joey F. George, Jeffrey A. Hoffer; Essentials of Systems Analysis & Design (2012).
2. Arthur M.Langer; Analysis and design of Information Systems (2008)
3. Oracle, System Modelling Techniques (1992)

3.9 PENENTUAN KEPERLUAN BUKAN FUNGSIAN [F2.5]

KETERANGAN

Keperluan bukan fungsian, adalah keperluan yang tidak berkenaan secara langsung dengan servis atau fungsi yang ditawarkan oleh sistem untuk penggunaanya. Ianya mungkin berkaitan dengan ciri-ciri sistem seperti kebolehpercayaan, prestasi, keselamatan dan ketersediaan. Kegagalan memenuhi keperluan bukan fungsian boleh membawa kepada sistem tidak boleh digunakan. Sebagai contoh, jika sistem kawalan tidak memenuhi keperluan kebolehpercayaan, ia tidak akan disahkan selamat untuk operasi.

Oleh itu cabaran yang paling penting yang dihadapi dalam kejuruteraan sistem aplikasi adalah keyakinan dan kepercayaan bahawa sistem aplikasi yang dibangunkan sentiasa tersedia apabila dilaksanakan sepertimana yang diharapkan. Ini bermakna, isu-isu kewibawaan sistem aplikasi dan keselamatan data sistem adalah sangat penting di samping fungsi sistem yang dibangunkan.

OBJEKTIF

- Memahami kepentingan keperluan bukan fungsian.
- Memahami bagaimana ciri-ciri keperluan bukan fungsian diperlukan untuk menyokong pelaksanaan keperluan fungsian

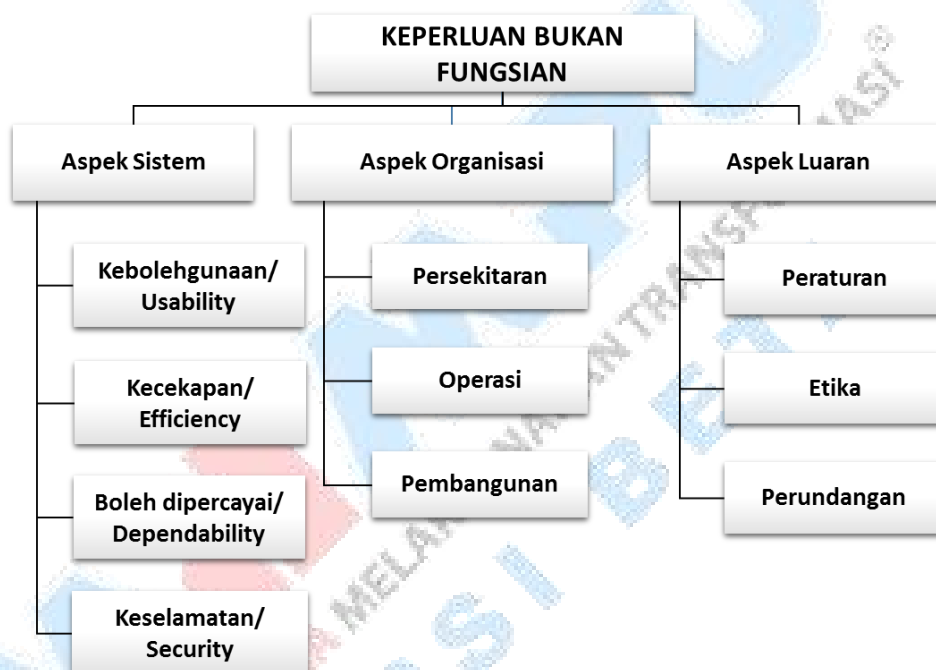
LANGKAH-LANGKAH

Langkah 1 : Analisis Spesifikasi Keperluan Bisnes

- a) Sumber maklumat keperluan bukan fungsian sistem adalah daripada dokumen Definisi Fungsi Bisnes yang telah didokumenkan dalam BRS semasa aktiviti Pemodelan Proses Bisnes. Butiran keperluan bukan bisnes selalunya dicatat dalam ruangan nota
- b) Senaraikan keperluan bukan fungsian yang dikenalpasti. Contoh catatan keperluan bukan fungsian adalah seperti berikut:
 - i) Sistem perlulah mudah dan senang digunakan.
 - ii) Sistem boleh berfungsi pada mana-mana perkakasan dan perisian.
 - iii) Masa tindak balas sistem adalah 2 saat bagi setiap transaksi.
 - iv) Mudah dipindah dan diselenggara.
 - v) Sistem perlu mempunyai tahap ketersediaan yang tinggi bagi menyokong operasi harian.

Langkah 2 : Terjemahkan Keperluan Bukan Fungsian Kepada Ciri-ciri Kualiti

- a) Keperluan bukan fungsian seringkali ditimbulkan kerana kekangan peruntukan, polisi organisasi, keperluan untuk beroperasi dengan lain-lain perisian atau perkakasan, atau faktor luaran seperti peraturan keselamatan atau perundangan privasi. Merujuk kepada buku Software Engineering 9th edition, penulis Ian Sommerville telah mengklasifikasikan keperluan bukan fungsian kemungkinan diperolehi dari ciri-ciri sistem yang diperlukan, organisasi yang membangun sistem berkenaan ataupun dari sumber luaran. Klasifikasi keperluan bukan fungsian yang dimaksudkan adalah terbahagi kepada tiga (3) aspek seperti rajah di bawah.



Rajah 55 : Klasifikasi Keperluan Bukan Fungsian

- b) Penerangan lanjut bagi aspek-aspek klasifikasi keperluan fungsian adalah seperti berikut:

Jadual 26 : Keterangan Aspek-aspek Klasifikasi Keperluan Fungsian

Aspek Sistem	Keperluan ini menentukan atau mengekang tindakan perisian. Sebagai contoh, prestasi sistem iaitu berapa cepat sistem mesti melaksanakan dan berapa banyak memori diperlukan, kebolehpercayaan yang dinyatakan iaitu kadar yang boleh diterima kegagalan, keselamatan dan keperluan kebolehgunaan.
Aspek Organisasi	Keperluan yang diperolehi daripada dasar dan prosedur dalam organisasi pembangun dan pelanggan. Sebagai contoh, keperluan proses operasi yang menentukan bagaimana sistem ini akan digunakan, keperluan proses pembangunan yang menentukan bahasa pengaturcaraan, persekitaran pembangunan atau standard proses yang akan digunakan, dan keperluan yang menentukan persekitaran operasi sistem.
Aspek Luaran	Keperluan luaran meliputi keperluan yang berasaskan kepada beberapa faktor luaran yang memberi impak kepada sistem dan proses pembangunan. Ini termasuklah keperluan peraturan yang perlu diikuti oleh sistem; keperluan perundangan yang perlu dipatuhi untuk memastikan bahawa sistem tersebut beroperasi mengikut undang-undang; dan keperluan etika yang memastikan bahawa sistem itu akan diterima oleh pengguna dan orang awam.

Langkah 3 : Lengkapkan Ciri-ciri Kualiti Sistem Aplikasi

- a) Lengkapkan templat Ciri-ciri Sistem Aplikasi mengikut kategori bukan fungsian seperti jadual berikut:

Jadual 27 : Templat Ciri-ciri Sistem Aplikasi

ID	Ciri-ciri Kualiti	Catatan

- b) Penerangan berkenaan medan-medan di dalam Templat Ciri-ciri Sistem Aplikasi adalah seperti berikut:

Jadual 28 : Keterangan Medan-medan Templat Ciri-ciri Sistem Aplikasi

Medan	Keterangan						
ID	<p>ID merupakan pengenalan kepada setiap ciri-ciri kualiti atau keperluan bukan fungsian yang dikenalpasti. Contoh format ID adalah seperti BF-XX-99, di mana:</p> <table border="1" data-bbox="454 501 1249 808"> <tr> <td data-bbox="454 501 572 562">BF</td> <td data-bbox="572 501 1249 562">Bukan Fungsian</td> </tr> <tr> <td data-bbox="454 562 572 748">XX</td> <td data-bbox="572 562 1249 748"> Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran </td> </tr> <tr> <td data-bbox="454 748 572 808">99</td> <td data-bbox="572 748 1249 808">Siri nombor</td> </tr> </table>	BF	Bukan Fungsian	XX	Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran	99	Siri nombor
BF	Bukan Fungsian						
XX	Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran						
99	Siri nombor						
Ciri-ciri Kualiti	<p>Ciri-ciri kualiti perlu dinyatakan mengikut kategori bukan fungsian masing-masing.</p> <ul style="list-style-type: none"> i) Aspek Sistem Ciri-ciri kualiti bagi sistem merujuk kepada Keperluan bukan fungsian di bawah aspek sistem adalah merujuk kepada standard ISO/IEC 25010:2011 <i>System And Software Quality Models</i>. ii) Aspek Organisasi Keperluan yang meliputi persekitaran, operasi dan pembangunan iii) Aspek Luaran Keperluan yang meliputi peraturan, akta dan perundangan 						
Catatan	<p>Catatan keterangan ciri-ciri kualiti atau keperluan pengguna yang dipetik dari Definisi Fungsi Bisnes.</p>						

- c) ISO/IEC 25010:2011 *System And Software Quality Models* menyenaraikan definisi ciri-ciri kualiti sistem aplikasi seperti berikut:

Jadual 29 : Senarai Definisi Ciri-ciri Kualiti Sistem Aplikasi

Ciri	Sub Ciri	Definisi
Functional suitability	i) Functional completeness ii) Functional correctness iii) Functional appropriateness	<p>Sejauh mana sistem menyediakan fungsi yang memenuhi kehendak yang nyata dan tersirat apabila digunakan dalam keadaan tertentu.</p> <p>i) Sejauh mana set fungsi meliputi semua tugas-tugas yang ditetapkan dan objektif pengguna</p> <p>ii) Sejauh mana sistem menyediakan keputusan yang betul dan tepat</p> <p>iii) Sejauh mana fungsi memudahkan pencapaian tugas dan objektif yang dinyatakan</p>
Performance efficiency	i) Time behaviour ii) Resource utilization iii) Capacity	Sejauh mana prestasi sistem dengan membandingkan jumlah sumber yang digunakan di bawah syarat-syarat yang dinyatakan
Compatibility	i) Co-existence ii) Interoperability	Sejauh mana sistem atau komponen boleh bertukar-tukar maklumat dengan sistem atau komponen lain, dan / atau melaksanakan fungsi yang diperlukan, sambil berkongsi persekitaran perkakasan atau perisian yang sama.
Usability	i) Appropriateness recognizability ii) Learnability iii) Operability iv) User error protection v) User interface aesthetics vi) Accessibility	Sejauh mana produk atau sistem boleh digunakan oleh pengguna tertentu untuk mencapai matlamat mereka dengan berkesan, cekap dan puas dalam konteks yang dinyatakan penggunaan

Reliability	i) Maturity ii) Availability iii) Fault tolerance iv) Recoverability	Sejauh mana sistem atau komponen melaksanakan fungsi tertentu di bawah syarat-syarat yang ditetapkan untuk tempoh masa yang tertentu
Security	i) Confidentiality ii) Integrity iii) Non-repudiation iv) Accountability v) Authenticity	Sejauh mana produk atau sistem melindungi maklumat dan data supaya pengguna atau sistem lain mempunyai tahap akses data tertentu dengan jenis dan tahap kebenaran mereka
Maintainability	i) Modularity ii) Reusability iii) Analysability iv) Modifiability v) Testability	Tahap keberkesanan dan kecekapan penyelenggara sistem yang dipertanggungjawab.
Portability	i) Adaptability ii) Installability iii) Replaceability	Tahap keberkesanan dan kecekapan dengan mana sistem, produk atau komponen boleh dipindahkan dari satu perkakasan, perisian atau persekitaran operasi yang lain.

Langkah 4 : Dokumentasikan Keperluan Bukan Fungsian

Dokumentasikan hasil analisis bukan fungsian ke dalam D03 Spesifikasi Keperluan Sistem di bawah seksyen keperluan bukan fungsian

RUJUKAN

1. Ian Sommerville, Software Engineering 9th edition
2. ISO/IEC 25010:2011 System And Software Quality Models

3.10 PENYEDIAAN SPESIFIKASI KEPERLUAN SISTEM [F2.6]

KETERANGAN

Spesifikasi Keperluan Perisian (SRS) adalah penerangan terperinci berkenaan sistem aplikasi yang akan dibangunkan merangkumi keperluan fungsian dan bukan fungsian. Dokumen SRS menjelaskan secara menyeluruh keupayaan yang diperlukan oleh satu-satu sistem aplikasi serta merekodkan syarat-syarat (*conditions*) dan kekangan (*constraints*) bagi sistem aplikasi berkenaan. Rujuk kepada D03 **Spesifikasi Keperluan Sistem** untuk melihat format dokumen SRS di mana pengisian kandungan-kandungannya adalah seperti langkah-langkah di bawah.

OBJEKTIF

- Menyediakan Model Fungsi Sistem hasil daripada langkah-langkah yang dilaksanakan di dalam Pemodelan Fungsi Sistem [F2.2].
- Menyediakan Model Maklumat, Definisi Entiti dan Definisi Atribut dengan merujuk kepada langkah-langkah di bawah Pemodelan Keperluan Data [F2.3].
- Menyediakan Model dan Definisi Proses Sistem dengan merujuk kepada langkah-langkah di bawah Pemodelan Proses Sistem [F2.4].
- Menyertakan maklumat yang telah dikenal pasti di dalam Penentuan Keperluan Bukan Fungsian [F2.5].
- Membangunkan dokumen D03 Spesifikasi Keperluan Sistem yang mengandungi model-model seperti di atas serta elemen-elemen lain yang berkaitan.

LANGKAH-LANGKAH

Langkah 1 : Sediakan Pengenalan Kepada Keperluan Sistem

Sila sedia dan lengkapkan pengenalan bisnes bagi perkara-perkara berikut:

a) Tujuan Sistem

Terangkan tujuan, objektif dan matlamat sistem aplikasi ini dibangunkan selaras dengan objektif bisnes yang ingin dicapai.

b) Skop Sistem

Jelaskan skop sistem aplikasi berdasarkan skop keperluan bisnes yang telah diperolehi di dalam D02 Spesifikasi Keperluan Bisnes. Skop sistem merangkumi penerangan secara ringkas berkenaan sistem aplikasi yang ingin dibangunkan serta komponen-komponen sistem yang berkaitan dengannya.

c) Senarai Aktor Sistem

Senarai aktor-aktor sistem yang terlibat serta keterangan fungsinya di dalam sistem aplikasi yang akan dibangunkan. Berbeza dengan senarai pemegang taruh dan

pengguna di dalam Penyediaan Spesifikasi Keperluan Bisnes [F1.5], penentuan aktor sistem adalah lebih tertumpu kepada peranan pengguna di dalam sistem aplikasi berbanding dengan nama jawatan seseorang individu ataupun nama unit organisasi yang terlibat. Contoh pengisian bagi senarai aktor sistem adalah seperti jadual berikut:

Jadual 30 : Contoh Senarai Aktor Sistem

Aktor	Keterangan
Pentadbir	Aktor yang berperanan untuk melakukan pemantauan dan semakan ke atas transaksi-transaksi permohonan yang dilaksanakan di dalam sistem aplikasi.
Pelulus	Aktor yang berperanan untuk melaksanakan proses kelulusan permohonan di dalam sistem aplikasi.
Pengesah	Aktor yang berperanan untuk melaksanakan proses pengesahan permohonan yang diterima di dalam sistem aplikasi.
Pemohon	Aktor yang berperanan untuk melakukan permohonan untuk mendapatkan perkhidmatan yang ditawarkan di dalam sistem aplikasi.

Langkah 2 : Dokumentasikan Model Fungsi Sistem

Pemodelan Fungsi Sistem

a) Penggunaan Notasi

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi Sistem. Rujuk kepada Pemodelan Fungsi Sistem [F2.2] untuk menyediakan senarai notasi berkenaan.

b) Rajah Hierarki Fungsian Sistem

Sediakan Rajah Hierarki Fungsian Sistem yang terdiri dari komponen-komponen Fungsi Sistem pada tahap sistem, subsistem, modul, submodul dan transaksi. Sila rujuk kepada langkah-langkah di dalam Pemodelan Fungsi Sistem [F2.2] bagi penyediaan rajah serta jadual berkenaan.

c) Jadual Pemadanan Aktor Dengan Fungsi Sistem

Sediakan Jadual Pemadanan Aktor dengan Fungsi Sistem pada tahap transaksi. Rujuk kepada langkah 3 di dalam **Pemodelan Fungsi Sistem [F2.2]** bagi penyediaan rajah serta jadual berkenaan.

Langkah 3 : Dokumentasikan Model Use Case

Pemodelan Use Case

a) Penggunaan Notasi

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Use Case. Sila rujuk kepada Pemodelan Use Case [F2.1] untuk menyediakan senarai notasi berkenaan.

b) Model Use Case

Sediakan Model Use Case yang terdiri daripada Rajah Use Case serta keterangan bagi Use Case yang terlibat. Sila rujuk kepada langkah-langkah di dalam Pemodelan Use Case [F2.1] bagi penyediaan rajah dan keterangan berkenaan.

Langkah 4 : Dokumentasikan Model Dan Definisi Maklumat

Pemodelan Maklumat

a) Penggunaan Notasi

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Maklumat. Sila rujuk kepada Pemodelan Keperluan Data [2.3] untuk menyediakan senarai notasi berkenaan.

b) Model Maklumat

Sediakan Model Maklumat yang terdiri daripada Rajah Hubungan Entiti. Sila rujuk kepada langkah 1 sehingga 3 di dalam Pemodelan Keperluan Data [2.3] bagi penyediaan rajah berkenaan.

c) Definisi Kamus Data

Sedia dan lengkapkan Definisi Kamus Data bagi setiap entiti dan atribut yang telah disediakan di dalam Rajah Hubungan Entiti. Sila rujuk kepada langkah-langkah di dalam Pemodelan Keperluan Data [2.3] dan format templat di Apendiks 3 Templat Definisi Kamus Data.

Langkah 5 : Dokumentasikan Model Dan Definisi Proses Sistem

Pemodelan Proses Sistem

a) Penggunaan Notasi

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Proses Sistem. Sila rujuk kepada Pemodelan Proses Sistem [F2.4] untuk menyediakan senarai notasi berkenaan.

b) Model Proses Sistem

Sediakan Model Maklumat yang terdiri daripada Rajah Konteks dan Rajah Aliran Data. Sila rujuk kepada langkah 1 sehingga 3 di dalam Pemodelan Proses Sistem [F2.4] bagi penyediaan rajah-rajah berkenaan.

c) Definisi Aliran Data

Sedia dan lengkapkan Definisi Aliran Data berdasarkan kepada Rajah Konteks dan Rajah Aliran Data yang telah dibangunkan. Sila rujuk kepada langkah-langkah di dalam Pemodelan Proses Sistem [F2.4] dan format templat di Apendiks 5 Definisi Aliran Data.

Langkah 6 : Rekodkan Maklumat Keperluan Bukan Fungsian

Maklumat Keperluan Bukan Fungsian yang perlu direkodkan perlu mengambil kira aspek-aspek seperti berikut:

- a) Aspek Sistem
- b) Aspek Organisasi
- c) Aspek Luaran

Lengkap dan sertakan jadual yang mengandungi maklumat keperluan bukan fungsian seperti yang diterangkan di dalam **Penentuan Keperluan Bukan Fungsian [F2.4]**.

Langkah 7 : Kirakan Saiz Sistem Aplikasi

Pengenalan kepada kepentingan pengiraan saiz sistem aplikasi telah diperjelas dan diterangkan di dalam **1.6.6 Kepentingan Pengukuran Saiz Sistem Aplikasi**. Manakala kaedah pengiraan saiz sistem aplikasi pula telah diterangkan di dalam **Bab 8 Pengiraan Saiz Sistem Aplikasi**. Sehubungan dengan itu, laksanakan pengiraan saiz sistem aplikasi dengan berdasarkan kepada elemen-elemen yang telah dikenalpasti di dalam SRS. Rujuk kepada bab 8 di dalam **Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Analisis** untuk mendapatkan penerangan terperinci berkenaan kaedah serta contoh pengiraan saiz sistem aplikasi.

Langkah 8 : Sertakan Dokumen-dokumen Sokongan Sebagai Lampiran

Sertakan dokumen-dokumen sokongan yang perlu dirujuk seperti format borang fizikal, format laporan dan pelbagai lagi dokumen-dokumen lain yang berkaitan.

Langkah 9 : Lakukan Semakan Dan Pengesahan Ke Atas Dokumen SRS

Dokumen SRS perlu dilakukan semakan oleh Ketua Pasukan Analisis dan Rekabentuk, atau pegawai-pegawai yang lain yang bersesuaian. Setelah semakan dilakukan, dokumen SRS yang telah disediakan perlu disahkan oleh Pengurus Projek atau Pengarah Bahagian atau pegawai-pegawai yang lain yang bersesuaian.

RUJUKAN


1. ISO/IEC/IEEE 29148-2011 Systems and software engineering — Life cycle processes — Requirements engineering (2011)
2. IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications (1998)
3. IEEE 1233-1998 - IEEE Guide for Developing System Requirements Specifications (1998)
4. Dokumen SRS Projek Sistem Bahagian Pengurusan Hartanah Versi 2 (2016)
5. Dokumen SRS Projek Sistem ePPAx (2016)



BAB:04

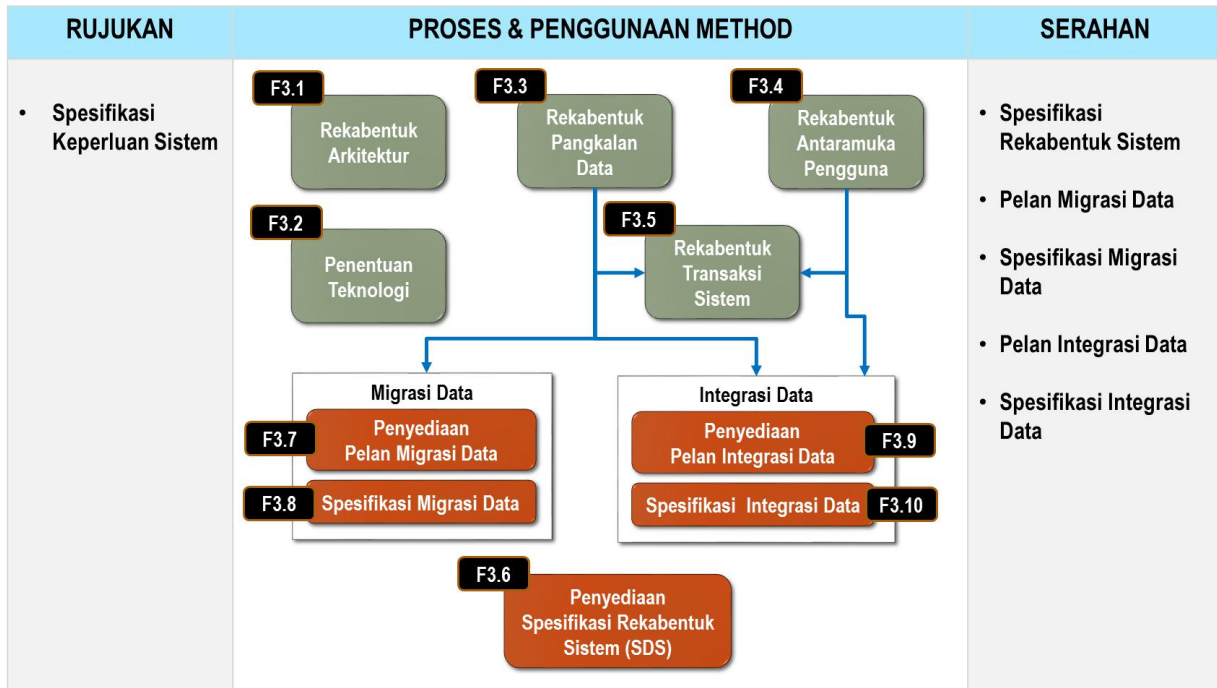
FASA REKABENTUK

Bab ini menerangkan berkenaan dengan aktiviti-aktiviti penyediaan Arkitektur Sistem, Rekabentuk Pangkalan Data, Rekabentuk Antara Muka Pengguna termasuk Pemetaan Data, Rekabentuk Transaksi Sistem, Rekabentuk Migrasi dan Rekabentuk Integrasi berdasarkan kepada maklumat keperluan sistem yang telah dikenalpasti.



4 FASA REKABENTUK

4.1 GAMBARAN KESELURUHAN



Rajah 56 : Gambaran Keseluruhan Fasa III - Rekabentuk

4.2 PENGENALAN

Fasa reka bentuk adalah fasa bagi merancang penyelesaian masalah dan ekspektasi yang dinyatakan dalam **D03 Spesifikasi Keperluan Sistem**. Fasa ini adalah langkah permulaan untuk terjemahkan dari domain masalah kepada domain penyelesaian iaitu alihan daripada 'Apa?' kepada 'Bagaimana?'. Reka bentuk sistem adalah faktor yang paling kritikal yang akan menjejaskan kualiti perisian dan mempunyai kesan yang besar kepada aktiviti pembangunan/pembinaan sistem. Fasa Rekabentuk Sistem menggariskan 7 aktiviti utama iaitu:

- Rekabentuk Arkitek
- Penentuan Teknologi
- Rekabentuk Transaksi Sistem
- Rekabentuk Pangkalan Data
- Rekabentuk Antaramuka Pengguna
- Migrasi Data
- Integrasi Data

Dokumen Rujukan kepada Fasa Rekabentuk adalah **D03 Spesifikasi Keperluan Sistem**.

Dokumen Serahan kepada Fasa Rekabentuk adalah seperti berikut:

- a) **D04 Spesifikasi Rekabentuk Sistem**
- b) **D05 Pelan Migrasi Data**
- c) **D06 Spesifikasi Migrasi Data**
- d) **D07 Pelan Integrasi Sistem**
- e) **D08 Spesifikasi Integrasi Sistem**

4.3 Penglibatan Pemegang Taruh

Pemegang Taruh utama yang patut terlibat dalam rekabentuk sistem adalah Juruanalisa Sistem. Juruanalisa sistem perlulah berkeupayaan untuk menterjemahkan keperluan sistem yang didokumen kepada logik pembangunan untuk menghasilkan sistem yang berkualiti dan memenuhi kehendak pelanggan.

Cadangan penglibatan kategori pemegang taruh adalah seperti berikut:

- a) Juruanalisis Sistem yang membangunkan rekabentuk
- b) SME untuk menyemak rekabentuk fungsian, data dan antaramuka sistem

4.4 Faktor Kejayaan

Untuk memastikan aktiviti rekabentuk berjaya dilaksanakan, faktor kejayaan utama yang perlu dipertimbangkan sebelum dan semasa aktiviti dilaksanakan adalah seperti berikut:

- a) **D03 Spesifikasi Keperluan Sistem** perlu mendapat pengesahan Pemilik Sistem.
- b) Pasukan Rekabentuk Sistem mempunyai kompetensi dan kemahiran untuk menterjemahkan keperluan sistem kepada rekabentuk.
- c) **D04 Spesifikasi Rekabentuk Sistem** disemak dan mendapat pengesahan pemegang taruh.

4.5 REKABENTUK ARKITEK [F3.1]

PENGENALAN

Rekabentuk arkitektur adalah penyusunan dan pengaturan struktur-struktur bagi sesuatu sistem yang ingin dibangunkan. Rekabentuk arkitektur merupakan hubungan yang kritikal di antara rekabentuk dan kejuruteraan keperluan, di mana penyediaannya bertujuan untuk mengenal pasti komponen-komponen berstruktur yang utama di dalam sistem serta hubungan-hubungan di antara setiap komponen tersebut. Rekabentuk arkitektur adalah penting untuk memenuhi keperluan fungsian dan juga bukan fungsian oleh kerana impaknya kepada prestasi, keteguhan (*robustness*), pengagihan (*distributability*) dan kebolehsenggaraan sistem aplikasi.

Lapisan-lapisan bisnes, maklumat/data, aplikasi dan teknologi yang terkandung di dalam arkitektur *enterprise* boleh dijadikan sebagai input dan rujukan semasa penyediaan rekabentuk arkitektur yang dikehendaki. Output kepada proses rekabentuk arkitektur adalah arkitektur perisian yang terdiri daripada arkitektur perisian sistem aplikasi, arkitektur aplikasi dan arkitektur data. Arkitektur yang akan dihasilkan ini menerangkan bagaimana sesuatu sistem disusun atur sebagai set komponen yang saling berkomunikasi di antara satu sama lain.

Arkitektur Sistem Aplikasi

Arkitektur sistem aplikasi merupakan struktur bagi satu-satu sistem yang merangkumi komponen-komponen perisian, sifat-sifat (*properties*) komponen berkenaan serta hubungan di antara satu komponen dengan komponen-komponen yang lain. Arkitektur sistem aplikasi boleh diperincikan kepada sub-sub arkitektur seperti berikut :

- a) **Arkitektur Aplikasi**
- b) **Arkitektur Pangkalan Data**

OBJEKTIF

- o Menyediakan arkitektur sistem aplikasi yang terdiri daripada arkitektur keseluruhan sistem aplikasi, arkitektur aplikasi dan arkitektur pangkalan data berpandukan kepada keperluan-keperluan yang diperolehi di dalam fasa permulaan projek dan fasa analisis.

LANGKAH-LANGKAH

Langkah 1 : Kenal Pasti Keperluan Bisnes Dan Sistem

Rujuk dokumen-dokumen **D02 Spesifikasi Keperluan Bisnes** dan **D03 Spesifikasi Keperluan Sistem** bagi mendapatkan keperluan fungsian dan bukan fungsian yang telah diperolehi daripada pemegang-pemegang taruh. Sila rujuk juga arkitektur *enterprise* untuk dijadikan sebagai input kepada arkitektur perisian yang ingin dibangunkan.

Langkah 2 : Laksanakan Pertimbangan Rekabentuk

Pertimbangan rekabentuk perlu dilakukan berdasarkan keperluan yang telah didapati semasa kajian keperluan sistem. Penilaian ke atas pertimbangan rekabentuk perlu mengambil kira perkara-perkara seperti berikut:

a) Penentuan Jenis Aplikasi

Pemilihan jenis aplikasi yang sesuai adalah penting bagi proses rekabentuk aplikasi. Pemilihan adalah tertakluk kepada keperluan yang telah diperolehi dan batasan infrastruktur serta kemampuan teknologi sedia ada. Contoh jenis-jenis aplikasi adalah seperti aplikasi mudah alih, *client-server application*, portal, aplikasi web tradisional atau aplikasi web moden (*Rich Internet Application*).

b) Penentuan Strategi Pelaksanaan

Pelaksanaan aplikasi boleh dilakukan di dalam pelbagai jenis persekitaran di mana setiap persekitaran mempunyai kekangan-kekangan tersendiri, Contohnya komponen-komponen yang perlu diasingkan secara fizikal dan perlu merentasi pelayan yang berbeza-beza, kekangan konfigurasi bagi peranti-peranti dan kekangan-kekangan yang lain. Keseimbangan di antara keperluan aplikasi dengan kemampuan perkakasan adalah penting di mana faktor-faktor ini boleh mempengaruhi kepada rekabentuk arkitektur yang dibangunkan.

c) Penentuan Ciri-ciri Kualiti

Ciri-ciri kualiti atau keperluan bukan fungsian seperti modular, keselamatan, prestasi, capaian, ketersediaan dan kebolehgunaan semula merupakan faktor-faktor yang perlu dipertimbangkan dalam penyediaan rekabentuk arkitektur. Pemilihan ciri-ciri kualiti yang ingin diutamakan adalah bergantung kepada keperluan yang telah diperolehi dan senario pelaksanaan aplikasi yang akan dijalankan kelak. Pemilihan keutamaan ini perlu dilakukan terlebih dahulu sebelum sebarang rekabentuk arkitektur dibangunkan bagi mengelakkan percanggahan di antara ciri-ciri kualiti yang perlu disediakan di dalam sistem. Sebagai contoh, keutamaan kepada keselamatan berkemungkinan boleh membebankan prestasi dan menyukarkan akses pengguna kepada aplikasi.

Langkah 3 : Kenal Pasti Corak Arkitektur Yang Sesuai

a) Corak arkitektur yang sesuai perlulah dikenal pasti berdasarkan kepada keperluan dan hasil pertimbangan rekabentuk. Corak rekabentuk arkitektur adalah bergantung kepada jenis sistem yang akan dibangunkan. Antara corak rekabentuk arkitektur yang sering digunakan adalah arkitektur lapisan (*layered architecture*) atau dikenali juga sebagai arkitektur *n-tier*. Berikut adalah fakta-fakta penting mengenai arkitektur lapisan:

i) Arkitektur lapisan merupakan gambaran aturan sistem dalam bentuk lapisan dan dihubungkan setiap lapisan tersebut dengan fungsi-fungsi yang berkaitan. Setiap

lapisan di dalam arkitektur mengandungi komponen-komponen sistem yang berbeza dan melaksanakan tugas serta logik yang tertentu. Arkitektur lapisan sesuai untuk digunakan bagi pembangunan aplikasi web atau aplikasi mudah alih berbentuk sistem maklumat kerana ia mudah untuk difahami serta pengasingan komponen-komponen sistem kepada lapisan yang berbeza meningkatkan kebolehsenggaraan dan keboleh-skalaan (*scalability*) sesuatu sistem.

- ii) Secara amnya, arkitektur ini merangkumi lapisan-lapisan seperti lapisan persembahan (*presentation tier*), lapisan pertengahan (*middle tier*) dan lapisan data (*data tier*).
 - Lapisan persembahan merupakan lapisan di mana pengguna berinteraksi dengan aplikasi. Lapisan ini bertanggungjawab dalam mengendalikan kesemua antaramuka pengguna serta logik komunikasi pelayar web.
 - Lapisan pertengahan atau dikenali juga sebagai lapisan aplikasi (*application tier*) atau lapisan bisnes merupakan lapisan pengantaraan yang menghubungkan di antara lapisan persembahan dan lapisan data. Lapisan pertengahan ini terdiri dari komponen-komponen yang melibatkan fungsi logik bisnes aplikasi, perkhidmatan aplikasi atau/dan komponen utiliti (perisian utiliti) yang digunakan oleh komponen-komponen aplikasi yang lain.
 - Lapisan data pula merupakan lapisan di mana maklumat-maklumat aplikasi disimpan di dalam pelayan pangkalan data.
- b) Corak-corak arkitektur lain seperti *event-driven*, *microkernel*, *space-based* dan *client-server* boleh juga diguna pakai tetapi bergantung kepada kesesuaian dan juga keperluan yang telah diperolehi. Untuk mendapatkan keterangan lanjut berkenaan corak-corak arkitektur yang lain, buku-buku seperti di bawah boleh dirujuk:
 - i) *Software Architecture Patterns - Understanding Common Architecture Patterns and When to Use Them* (2015) oleh Mark Richards
 - ii) *Software Engineering – Chapter 6 Architectural Design (9th Edition 2011)* oleh Ian Sommerville

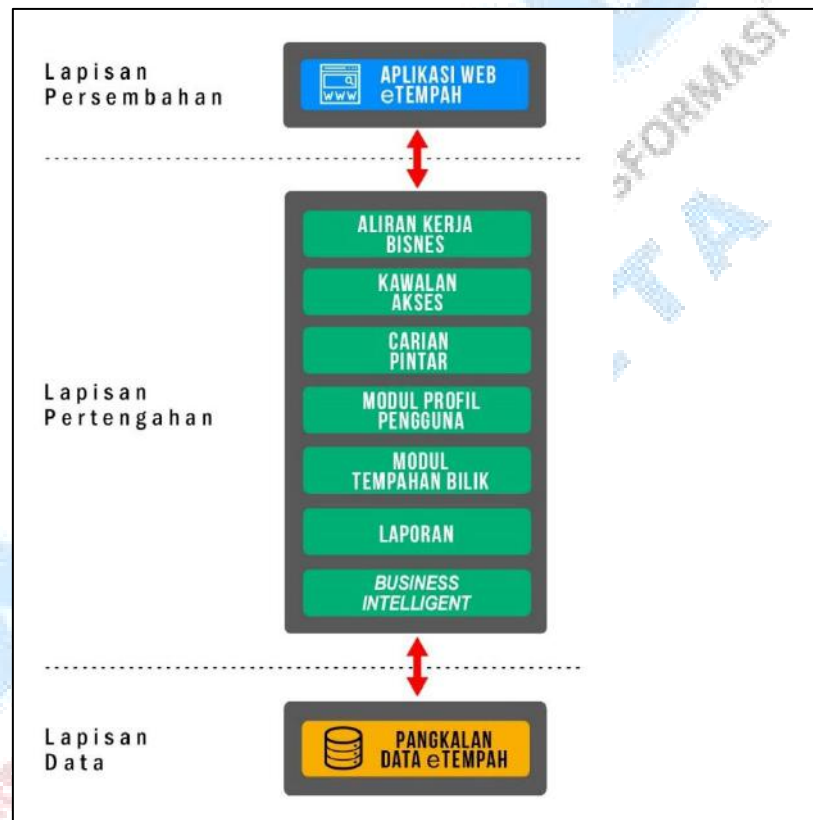
Langkah 4 : Sediakan Arkitektur Keseluruhan Sistem Aplikasi

- a) Arkitektur Keseluruhan Sistem Aplikasi adalah merupakan:
 - i) komponen-komponen perisian iaitu antaramuka sistem, aplikasi dan repositori (data)
 - ii) sifat-sifat luaran (*external properties*) komponen berkenaan
 - iii) hubungan di antara satu komponen dengan komponen-komponen yang lain

b) Arkitektur Sistem Aplikasi boleh dibahagikan kepada dua jenis seperti berikut:

i) Arkitektur Monolitik

Arkitektur monolitik adalah arkitektur yang menggabungkan semua komponen fungsian perisian seperti kawalan akses, aliran kerja, modul profil pengguna dan laporan, menjadi satu unit sahaja. Perisian yang menggunakan pakai arkitektur monolitik direkabentuk supaya ia bersifat *self-contained*, di mana komponen-komponen perisian berkenaan saling berhubung (*interconnected*) dan saling bergantung (*interdependent*) di antara satu sama lain. Dengan kata lain, arkitektur monolitik merupakan arkitektur yang bersifat *tightly-coupled*, setiap komponen yang berkaitan perlu disediakan bersama bagi membolehkan ia dilaksanakan. Berikut adalah contoh arkitektur bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur monolitik:

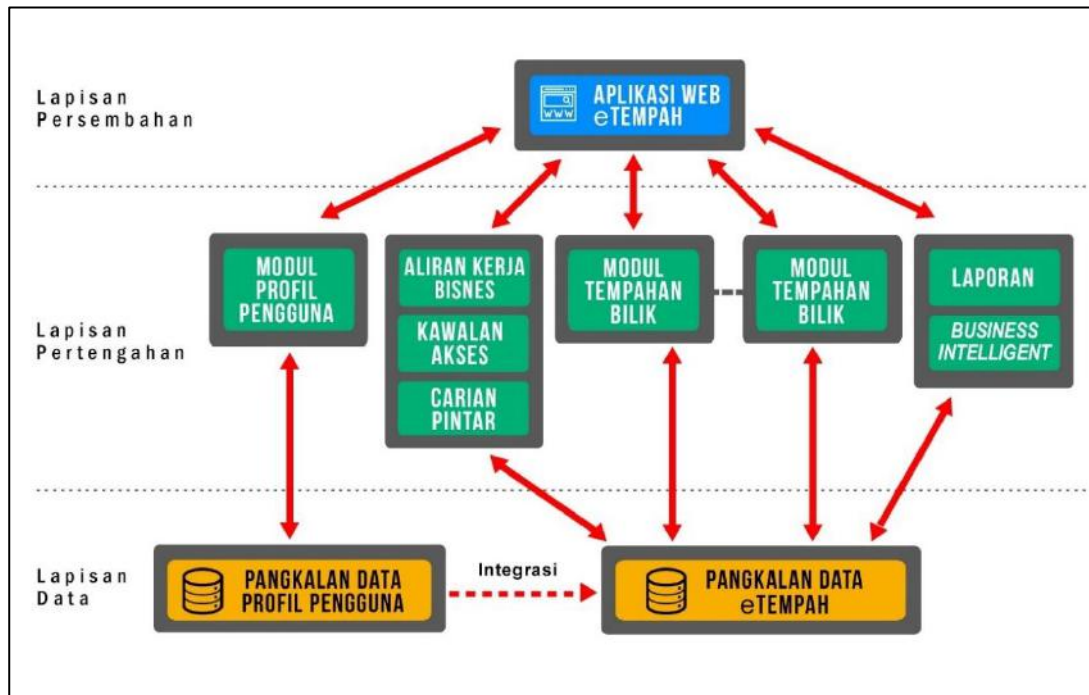


Rajah 57 : Arkitektur Monolitik Bagi Perisian Sistem Aplikasi

ii) Arkitektur Mikroservis

Arkitektur mikroservis adalah arkitektur yang mengagihkan perisian atau sistem aplikasi di pihak pelayan (*server side*) kepada servis-servis atau komponen-komponen perisian yang berasingan. Setiap servis atau komponen merupakan satu proses kecil yang tidak bergantung kepada mana-mana proses lain (*independent*). Mikroservis melaksanakan satu-satu tugas (*task*) secara autonomi tanpa mengganggu komponen-komponen lain di dalam sistem aplikasi. Komunikasi

di antara satu servis dengan servis-servis yang lain dilaksanakan secara dalam talian. Berikut adalah contoh arkitektur bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur mikroservis:



Rajah 58 : Arkitektur Mikroservis Bagi Perisian Sistem Aplikasi

c) Perbandingan di antara arkitektur monolitik dengan mikroservis adalah seperti berikut:

Jadual 31 : Perbandingan Arkitektur Monolitik dan Mikroservis

Ciri-ciri	Arkitektur Monolitik	Arkitektur Mikrosevis
Pengurusan Sumber	Sukar untuk menentukan jumlah sumber yang diperlukan oleh setiap komponen-komponen perisian.	i) Keperluan sumber bagi setiap komponen/servis boleh diukur secara berasingan. ii) Arkitektur ini juga membolehkan penggunaan sumber dilaksanakan secara optimum mengikut keperluan modul dan komponen/servis.

Ciri-ciri	Arkitektur Monolitik	Arkitektur Mikrosevis
Pembangunan	Pembangunan perisian dilaksanakan secara konvensional. Variasi di dalam penggunaan teknologi, rangka kerja serta ahli pasukan yang terlibat dalam satu-satu projek sukar untuk dilaksanakan.	<ul style="list-style-type: none"> i) Pembangunan boleh dilakukan dengan menggunakan teknologi, bahasa pengaturcaraan dan rangka kerja yang berbeza oleh kerana setiap servis/komponen boleh berinteraksi dalam talian menggunakan kemudahan integrasi seperti <i>Application Programming Interface</i> (API). ii) Disebabkan servis/komponen di dalam arkitektur ini berinteraksi secara dalam talian, ia juga boleh digunakan oleh perisian atau aplikasi yang lain. iii) Pengasingan komponen-komponen perisian membolehkan aplikasi dibangunkan oleh pasukan-pasukan kecil yang berlainan. iv) Pengaturcaraan lebih sukar kerana ia perlu melibatkan pengaturcaraan yang melibatkan interaksi di antara servis/komponen sama ada melalui panggilan API atau integrasi data.
Komunikasi dan Rangkaian	Perisian/Aplikasi monolitik adalah bersifat <i>self-contained</i> di mana semua komponen dikumpulkan dalam satu unit sahaja. Justeru, tidak timbul isu <i>latency</i> dan kebergantungan kepada prestasi rangkaian.	Komunikasi di antara servis/komponen berkemungkinan perlahan kerana bergantung kepada prestasi rangkaian.
Pengujian	i) Langkah pengujian adalah lebih ringkas kerana tidak memerlukan semakan lanjut kepada elemen-elemen tambahan yang melibatkan integrasi di antara komponen ataupun data.	i) Langkah pengujian lebih kompleks berbanding dengan arkitektur monolitik kerana semakan perlu melibatkan elemen-elemen integrasi di antara servis/komponen dan data.

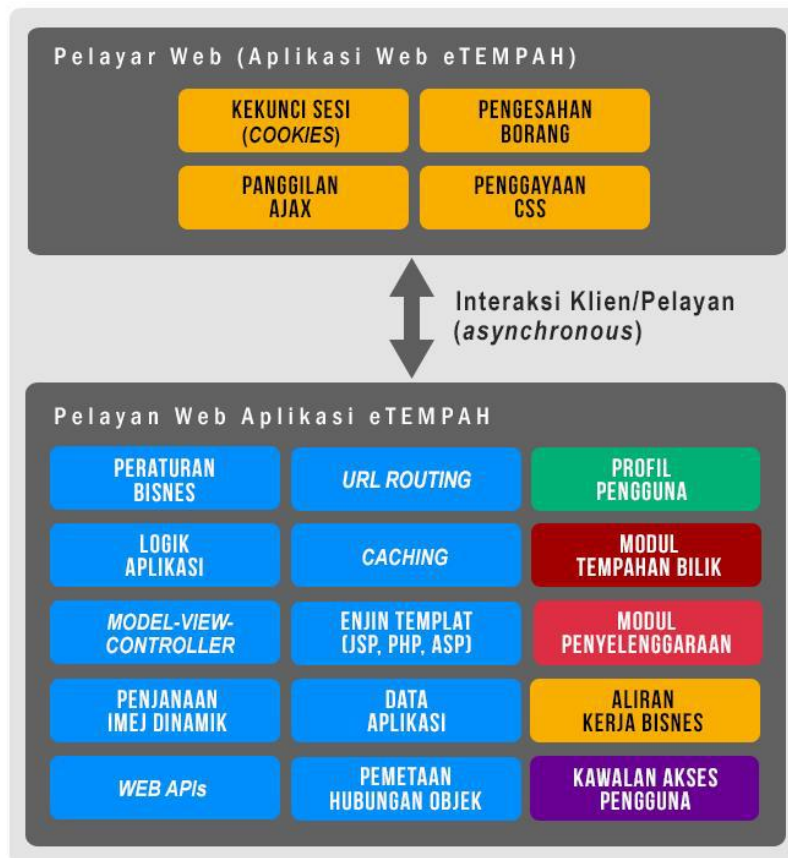
Ciri-ciri	Arkitektur Monolitik	Arkitektur Mikrosevis
	ii) Namun begitu, pengujian dan <i>debugging</i> secara berasingan untuk setiap komponen sukar untuk dilaksanakan.	ii) Pengujian dan <i>debugging</i> secara berasingan lebih mudah untuk dilakukan kerana servis/komponen sudah sedia teragih.
Kebolehskalaan (<i>Scalability</i>)	Peningkatan sumber (memori dan CPU) adalah tidak fleksibel kerana peningkatan perlu mengambil kira kesemua komponen di dalam sistem aplikasi.	Peningkatan sumber (memori dan CPU) secara penskalaan menegak (<i>vertical scaling</i>) dan mendatar (<i>horizontal scaling</i>) lebih anjal kerana peningkatan boleh dibuat mengikut keperluan modul dan komponen/servis.
Kebolehsediaan (<i>Availability</i>)	Arkitektur ini mempunyai ciri kebolehsediaan yang tinggi (<i>High Availability</i> (HA)) di mana semua modul dan komponen/servis perlu dimasukkan ke dalam kluster HA.	Arkitektur ini mempunyai ciri kebolehsediaan yang tinggi (HA) di mana modul-modul dan komponen/servis yang tertentu boleh dipilih untuk dimasukkan ke dalam kluster HA.
Pelaksanaan	Pelaksanaan kebiasaannya perlu dilaksanakan secara menyeluruh kerana setiap komponen perisian bergantung di antara satu sama lain.	Pelaksanaan boleh dilaksanakan secara berasingan tanpa perlu melibatkan servis/komponen lain yang masih belum selesai dibangunkan.
Penyelenggaraan	<p>i) Sebarang perubahan berkemungkinan memerlukan seluruh perisian/aplikasi untuk dibina semula (<i>rebuild</i>) atau <i>redeploy</i> disebabkan oleh kebergantungan komponen-komponen di antara satu sama lain.</p> <p>ii) Oleh yang demikian, masa yang diambil untuk sebarang penambahbaikan ataupun pembaikan (<i>error fixing</i>) adalah lama dan mudah terdedah dengan pelbagai kesilapan.</p> <p>iii) Sebarang kesilapan bukan sahaja boleh</p>	Sebarang perubahan kepada satu-satu servis hanya melibatkan penyelenggaraan kepada servis/komponen berkenaan sahaja dan tidak mengganggu kepada yang lain. Justeru, proses penambahbaikan dan pembaikan dapat diselesaikan dengan lebih pantas.

Ciri-ciri	Arkitektur Monolitik	Arkitektur Mikrosevis
	menjejaskan komponen/ servis yang lain malah ia juga berkemungkinan akan menyebabkan seluruh perisian/aplikasi terjejas.	

- d) Tentukan sama ada arkitektur perisian sistem aplikasi yang ingin dibangunkan adalah berpandukan arkitektur monolitik atau mikroservis. Penentuan arkitektur ini perlu diseimbangkan di antara halatuju agensi, kekangan-kekangan dan infrastruktur sedia ada dengan keperluan fungsian dan bukan fungsian yang telah diperolehi di fasa Analisis Keperluan.
- e) Berdasarkan dua jenis arkitektur perisian sistem aplikasi seperti di atas dengan mengambil kira perbandingan di antara kedua-duanya, sediakan arkitektur yang ideal dan sesuai bagi rekabentuk perisian sistem aplikasi yang ingin dibangunkan.

Langkah 5 : Sediakan Arkitektur Aplikasi

- a) Arkitektur aplikasi merupakan subset kepada Arkitektur Keseluruhan Sistem Aplikasi di mana ia tertumpu kepada penyusunan dan pengaturcaraan sistem aplikasi sahaja. Kemajuan teknologi di dalam pembangunan sistem aplikasi telah banyak mempengaruhi perubahan kepada arkitektur aplikasi.
- b) Dalam persekitaran web, arkitektur aplikasi boleh dibahagikan kepada dua jenis seperti berikut:
- i) Arkitektur Aplikasi Tradisional**
- Arkitektur aplikasi tradisional lebih tertumpu kepada penggunaan dan utiliti di pihak pelayan (*server-side*), di mana pembentuk setiap muka web adalah berpandukan kepada skrip-skrip HTML dan kod pengaturcaraan lain yang dijana dan ditarik dari pelayan. Arkitektur ini adalah terdiri daripada aplikasi yang merupakan muka-muka web (*web pages*). Bagi mendapatkan kandungan yang dinamik, web aplikasi tradisional perlu sentiasa berhubung dengan pelayan web bagi menjana semula muka web dan memaparkan permohonan-permohonan yang dilakukan oleh pengguna (*user request*). Berikut adalah contoh arkitektur aplikasi bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur aplikasi tradisional:



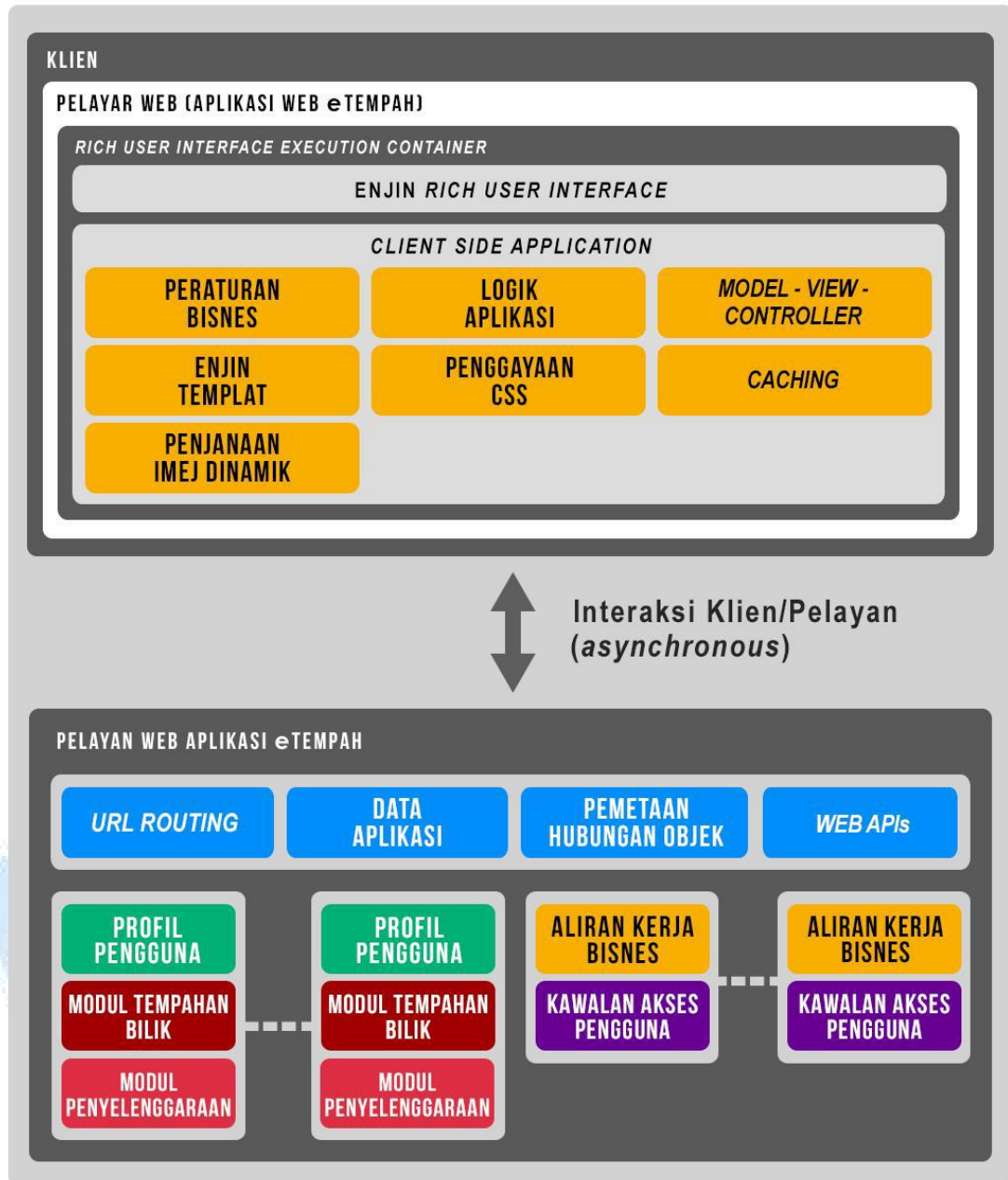
Rajah 59 : Arkitektur Aplikasi Tradisional

ii) Arkitektur Aplikasi Moden

Perkembangan teknologi bahasa pengaturcaraan seperti HTML5 dan CSS3, serta pengenalan kepada persekitaran *runtime* Javascript telah banyak merubah arkitektur aplikasi tradisional. Arkitektur aplikasi moden yang menggunakan teknologi terkini adalah lebih tertumpu kepada utiliti di pihak klien (*client-side*) dan meminimalkan penggunaan serta beban di pihak pelayan. Aplikasi dalam arkitektur ini bertindak sebagai aplikasi *single page* di mana :

- interaksi di antara klien dengan pelayan hanya melibatkan *unformatted data* sahaja
- pengesahan dapat dibuat secara langsung di klien (*live validation*)
- *unnecessary page reload*
- kemudahan *drag and drop*
- kemudahan animasi multimedia
- masa respon yang singkat
- *auto completion*,
- *periodic refresh*
- *rich text editors*
- *pull technology*

Ini menjadi aplikasi lebih kukuh (*robust*), meningkatkan *user experience*, prestasi serta tahap responsif sistem. Berikut adalah contoh arkitektur aplikasi bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur aplikasi moden:



Rajah 60 : Arkitektur Aplikasi Moden

- c) **Perbandingan di antara arkitektur aplikasi tradisional dengan aplikasi moden adalah seperti berikut:**

Jadual 32 : Perbandingan Arkitektur Aplikasi Tradisional Dan Aplikasi Moden

Ciri-ciri	Arkitektur Aplikasi Tradisi	Arkitektur Aplikasi Moden
Pengurusan Sumber	<p>i) Arkitektur web aplikasi tradisional membebankan dan meningkatkan kompleksiti di pihak pelayan. Pelayan aplikasi perlu melaksanakan sebahagian besar daripada pemprosesan perisian termasuk juga pemprosesan yang dilaksanakan di dalam pelayar web dan antaramuka pengguna (UI).</p> <p>ii) Keperluan sumber yang tinggi di pihak pelayan ini bukan sahaja memerlukan kos yang tinggi malah ia juga boleh menjejaskan prestasi keseluruhan aplikasi sekiranya keperluan tersebut tidak dapat dipenuhi.</p>	<p>Arkitektur web aplikasi moden menawarkan <i>native-app-like experience</i> kepada pengguna di mana sebahagian besar daripada <i>runtime</i> aplikasi dilaksanakan di pihak klien. Oleh yang demikian, utiliti di pihak server dapat dilaksanakan secara efisien tanpa memerlukan sumber yang tinggi untuk memenuhi keperluan operasi satu-satu aplikasi.</p>
Pembangunan	<p>i) Disebabkan arkitektur ini adalah bersifat <i>tightly coupled</i> dan komponen aplikasi kian bergantung di antara satu sama lain, pembangunan perlu dilaksanakan secara menyeluruh dan bersekali di pihak klien dan juga pelayan.</p> <p>ii) Pendekatan pembangunan lebih kepada pembangunan muka web (<i>web pages</i>).</p>	<p>i) Pengasingan dalam kebergantungan (<i>loosely coupled</i>) di antara pihak klien dan pelayan membolehkan pembangunan aplikasi dijalankan secara teragih. Pembangunan boleh dilaksanakan oleh dua pasukan yang berbeza, di mana satu kumpulan akan lebih fokus kepada pengaturcaraan di pihak klien (pengaturcaraan HTML, CSS dan Javascript boleh dilakukan tanpa perlu untuk mengaturcara terlebih dahulu teknologi</p>

Ciri-ciri	Arkitektur Aplikasi Tradisi	Arkitektur Aplikasi Moden
		<p><i>templating</i> aplikasi seperti JSP, ASP dan PHP) manakala kumpulan yang lain membangunkan aplikasi dan juga API web di pihak klien.</p> <p>ii) Pendekatan pembangunan lebih kepada pembangunan aplikasi.</p>
Komunikasi dan Rangkaian	Sebarang isu rangkaian yang timbul akan mengakibatkan aplikasi tidak boleh dicapai dan digunakan. Ini kerana, arkitektur aplikasi tradisional memerlukan komunikasi dengan pelayan untuk menjana fail-fail program seperti HTML dan imej dinamik (antaramuka pengguna) kepada pihak klien.	Arkitektur aplikasi moden boleh dimuatkan ke dalam pelayar web atau disimpan secara luar talian melalui <i>cache manifest</i> . Ini membolehkan fungsi-fungsi dalam aplikasi untuk terus digunakan walaupun terdapat gangguan rangkaian.
Pengujian	Pengujian aplikasi bagi arkitektur ini hanya boleh dilaksanakan secara <i>end-to-end</i> dan menyeluruh. Pengujian secara teragih atau <i>unit test</i> sukar untuk dilakukan oleh kerana kebergantungan yang tinggi di antara komponen-komponen aplikasi.	Arkitektur ini membolehkan pengujian dilaksanakan sama ada secara menyeluruh dan juga teragih (<i>unit test</i>) kerana kebergantungan di antara komponen-komponen aplikasi adalah rendah. Contohnya, pengujian API web atau <i>Javascript</i> boleh dilaksanakan secara berasingan tanpa perlu melibatkan komponen teknologi <i>templating</i> aplikasi seperti JSP, ASP dan PHP.
Pelaksanaan	Pelaksanaan dijalankan secara konvensional di mana aplikasi diakses melalui alamat URL di dalam pelayar web.	Pelaksanaan adalah lebih fleksibel dan menawarkan pelbagai opsyen. Pelaksanaan aplikasi boleh dijalankan dengan menggunakan pelayan web (sama seperti arkitektur aplikasi tradisional) dan ia juga boleh dipakejkan dengan rangkaian kerja seperti <i>Apache</i> dan <i>Cordova</i> bagi pengedaran melalui <i>App Store</i> atau <i>Google Play</i> .

Ciri-ciri	Arkitektur Aplikasi Tradisi	Arkitektur Aplikasi Moden
Prestasi	Aplikasi merupakan satu set <i>web pages</i> yang akan dijana di pelayan apabila pengguna klik pada alamat URL yang berkaitan. Proses ini akan memperlahankan sistem dan menjadi lebih kritikal apabila ramai pengguna membuat capaian secara serentak.	Merupakan aplikasi yang dimuatkan dalam pelayar web di mana hanya data sahaja yang ditarik daripada pelayan dan antaramuka pengguna (UI) dijana di pihak klien. Ini memberikan prestasi sistem yang lebih baik.
Penyelenggaraan	Penyelenggaraan aplikasi perlu dilaksanakan secara menyeluruh disebabkan oleh kebergantungan yang tinggi di antara komponen-komponen aplikasi (<i>tightly coupled</i>). Sebarang perubahan kod pengaturcaraan sama ada di klien atau pelayan perlu diselaraskan bagi membolehkan rekabentuk, ciri-ciri atau fungsi yang baru dipinda dapat berjalan dengan lancar.	Oleh kerana arkitektur ini adalah bersifat <i>loosely coupled</i> di mana kesemua logik persembahan diasingkan daripada komponen di pihak pelayan, sebarang pindaan di antaramuka aplikasi boleh dilaksanakan tanpa mengganggu kod pengaturcaraan di pihak pelayan. Justeru, sebarang perubahan dan evolusi kepada rekabentuk dan ciri-ciri (<i>features</i>) aplikasi dapat diselesaikan dengan lebih pantas.

- d) Tentukan sama ada arkitektur aplikasi yang ingin dibangunkan merupakan arkitektur aplikasi tradisional ataupun aplikasi moden. Penentuan arkitektur ini perlu diseimbangkan di antara halatuju agensi, kekangan-kekangan dan infrastruktur sedia ada dengan keperluan fungsian dan bukan fungsian yang telah diperolehi di fasa Analisis Keperluan.
- e) Berdasarkan dua jenis arkitektur aplikasi seperti di atas dengan mengambil kira perbandingan di antara kedua-duanya, sediakan arkitektur yang ideal dan sesuai bagi rekabentuk aplikasi yang ingin dibangunkan.

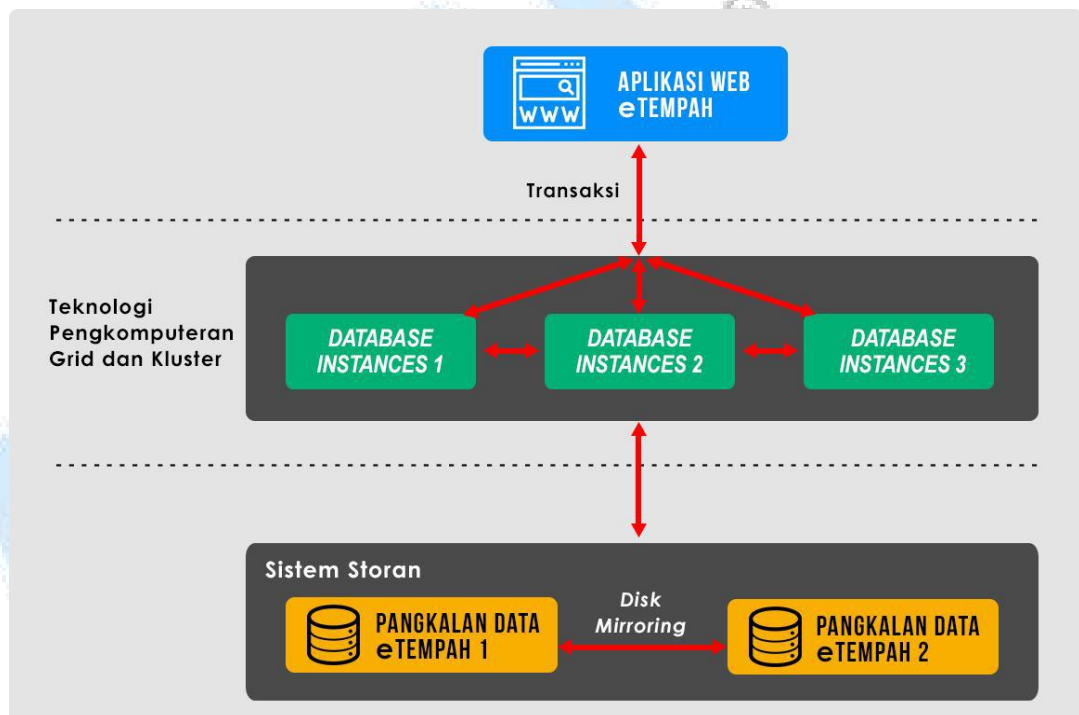
Langkah 6 : Sediakan Arkitektur Pangkalan Data

- a) Arkitektur pangkalan data merupakan subset kepada arkitektur perisian sistem aplikasi di mana ia tertumpu kepada penyusunan dan pengaturan pangkalan data sahaja. Penyediaan arkitektur tersebut adalah berdasarkan kepada keperluan fungsian dan bukan fungsian yang telah diperolehi serta corak arkitektur yang telah ditentukan.

b) **Arkitektur pangkalan data boleh dibahagikan kepada dua jenis seperti berikut:**

i) **Arkitektur *Shared-Disk***

Sistem-sistem yang wujud di dalam persekitaran *shared-disk* berkongsi peranti cakera (*disk devices*) yang sama. Setiap pemproses mempunyai memori tersendiri dan ia boleh mengakses kesemua cakera peranti. Arkitektur *shared disk* membolehkan setiap nod mengakses keseluruhan set data di mana setiap nod berkenaan akan memberi maklum balas kepada sebarang permohonan yang diterima dari pangkalan data. Arkitektur *shared-disk* adalah sesuai bagi aplikasi yang hanya memerlukan *shared access* yang sederhana ke pangkalan data, serta aplikasi yang mempunyai beban kerja (*workload*) yang sukar untuk diagihkan (*partition*). Berikut adalah contoh arkitektur pangkalan data bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur *shared-disk*:

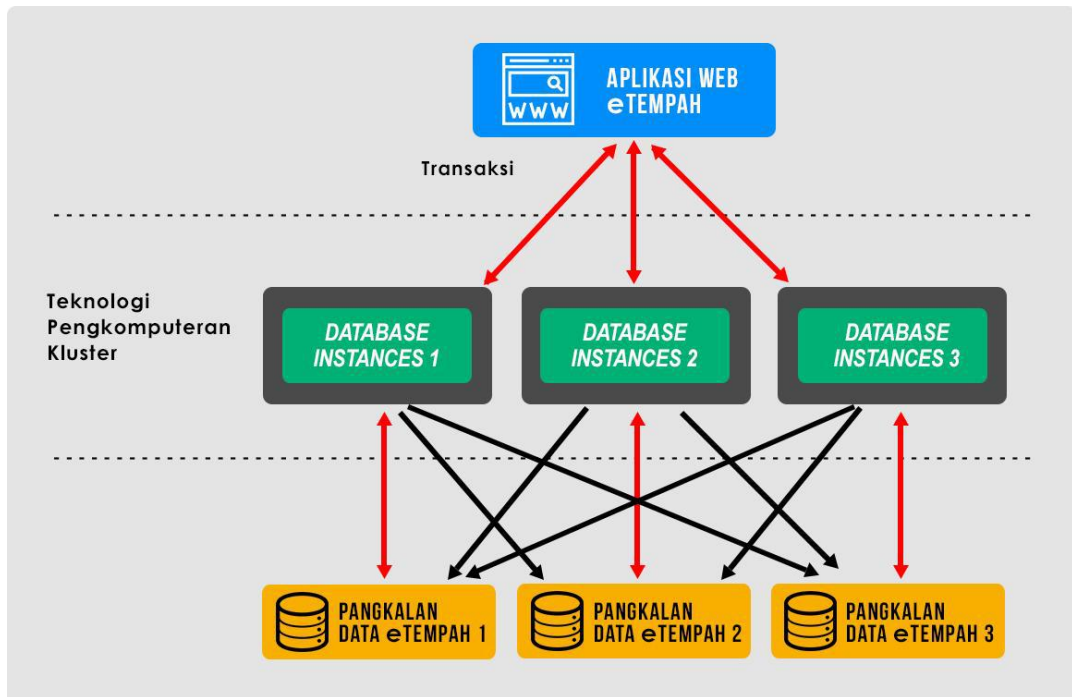


Rajah 61 : Arkitektur *Shared-Disk* Bagi Aplikasi Web eTempah

ii) **Arkitektur *Shared-Nothing***

Sistem-sistem yang wujud di dalam persekitaran *shared-nothing* mempunyai memori dan cakera peranti yang tersendiri. Arkitektur *shared-nothing* mempunyai kemampuan penskalaan yang tinggi (*high scalability*) dan sesuai untuk aplikasi yang kerap mengakses dan melaksanakan transaksi ke pangkalan data. Berikut

adalah contoh arkitektur pangkalan data bagi Sistem Tempahan Bilik Mesyuarat (eTempah) yang direkabentuk berpandukan arkitektur *shared-nothing*.



Rajah 62 : Arkitektur *Shared-Nothing* Bagi Aplikasi Web eTempah

- c) Perbandingan di antara arkitektur *shared-disk* dengan *shared-nothing* adalah seperti berikut:

Jadual 33 : Perbandingan Arkitektur Shared-Disk Vs Shared-Nothing

Ciri-ciri	Arkitektur <i>Shared-Disk</i>	Arkitektur <i>Shared-nothing</i>
Pemasangan dan Penyelenggaraan	<p>Tidak Memerlukan Usaha Tambahan</p> <p>Arkitektur ini tidak bergantung kepada <i>data partitioning</i> kerana setiap nod di dalam kluster mempunyai akses penuh untuk memaparkan dan mengemaskini seluruh data.</p>	<p>Jadual <i>Re-Partitioning</i> Dan <i>Routing</i></p> <p>Berikut adalah langkah-langkah pemasangan bagi pangkalan data <i>shared-nothing</i>:</p> <ol style="list-style-type: none"> i) Sediakan skema <i>partitioning</i> untuk meminimakan atau menghapuskan secara terus mesej antara-nodal. ii) Asingkan data di antara server-server dalam kluster. iii) Selenggara jadual <i>routing</i>.

Ciri-ciri	Arkitektur <i>Shared-Disk</i>	Arkitektur <i>Shared-nothing</i>
		iv) Ulangi semula langkah-langkah di atas bagi setiap <i>re-partition</i> .
Prestasi dan Overhead		
Fungsi / <i>Data Shipping</i>	Meningkatkan Prestasi Tiada fungsi/ <i>data shipping</i> , <i>overhead</i> tambahan tidak diperlukan.	Mengehadkan Prestasi Dan Masalah Penalaan (<i>Tuning</i>) Arkitektur ini bergantung kepada fungsi/ <i>data shipping</i> untuk memenuhi permintaan (<i>request</i>) kepada pangkalan data (seperti <i>joins</i>) yang merentasi pelbagai nod. Arkitektur ini boleh mengehadkan prestasi serta kebolehskalaan nodal berdasarkan skema <i>partitioning</i> yang dibangunkan, saiz data dan bilangan nod yang terlibat.
<i>2-Phase Commit</i> (2PC)	Prestasi yang Lebih Pantas Setiap nod boleh menyelaras nod masing-masing tanpa perlu menunggu nod yang paling perlahan untuk dilaksanakan.	Prestasi 2PC Teragih (<i>Distributed</i>) Merentangi Pelbagai Nod Adalah Sangat Perlahan 2PC teragih menguncikan data pada seluruh nod yang terlibat. Ini mengakibatkan prestasi semua permintaan data-data menjadi perlahan.
Kelajuan Mengakses Data	Latensi Mengakses Data SAN Bagi kluster-kluster penskalaan secara luaran (<i>scale-out</i>), kelajuan antarahubungan (<i>interconnect</i>) <i>shared-disk</i> adalah lebih perlahan berbanding dengan kelajuan <i>bus</i> (<i>bus speed</i>).	Cakera Setempat (<i>Local disk</i>), Kelajuan Mengakses <i>Bus</i> Akses kepada data setempat (<i>local data</i>) dapat dilaksanakan dengan pantas.
<i>Load Balancing</i>	Mengoptimalkan Utilasi CPU Merentasi Pelayan-pelayan (<i>Server</i>) Setiap pelayan dapat berfungsi pada tahap utilasi	Mengoptimalkan Utiliasi CPU Pada Setiap Nod Setiap nod dan pelayan perlu menguruskan beban secara sendiri

Ciri-ciri	Arkitektur <i>Shared-Disk</i>	Arkitektur <i>Shared-nothing</i>
	CPU yang lebih tinggi kerana beban pemprosesan dapat diagihkan kepada pelayan-pelayan lain di dalam kluster yang sama.	tanpa sebarang <i>load balancing</i> yang dinamik.
Mesej Antara-Nodal (<i>Inter-Nodal Messaging</i>)	<i>Messaging Overhead</i> Arkitektur ini bergantung kepada mesej antara-nodal.	Tiada Mesej Antara-Nodal Arkitektur ini tidak memerlukan mesej antara-nodal.
Ketersediaan Tinggi (<i>High Availability</i>)	<i>Master-master Failover</i> a) Tidak memerlukan <i>partitioning</i> , justeru, <i>planned downtime</i> dapat dikurangkan secara dramatik. b) <i>Unplanned downtime</i> juga dapat dikurangkan. Sekiranya terdapat pelayan yang bermasalah (<i>failed</i>), arkitektur ini secara dinamik akan mengubah haluannya kepada server-server yang lain tanpa sebarang <i>downtime</i> .	Keperluan <i>Planned</i> dan <i>Unplanned Downtime</i> a) <i>Planned downtime</i> diperlukan untuk melaksanakan <i>re-partitioning</i> . b) Kegagalan (<i>failed</i>) kepada nod <i>master</i> akan mengakibatkan <i>downtime</i> sehingga satu-satu nod <i>slave</i> dinaik taraf menjadi nod <i>master</i> .
Konsistensi Data	Tiada Masalah Konsistensi Data Arkitektur ini tidak mengalami masalah konsistensi data oleh kerana ia mempunyai pelbagai salinan data sama dalam server yang berlainan.	Mempunyai Masalah Konsistensi Data Perlu melakukan konfigurasi kepada transaksi supaya ia merangkumi replikasi secara <i>synchronous</i> dalam transaksi berkenaan. Replikasi ini akan menyebabkan prestasi kelajuan satu-satu pangkalan data menjadi perlahan.

Ciri-ciri	Arkitektur <i>Shared-Disk</i>	Arkitektur <i>Shared-nothing</i>
Kebolehskaaan (<i>Scalability</i>)		
Penskalaan Secara Dalam (Scale-in) : Satu Pendekatan Kebolehskaaan Dalam Pelayan-pelayan Moden <i>MultiCore</i>	<p>Menyokong Pendekatan Penskalaan Secara Dalam</p> <p>Arkitektur ini menyokong penskalaan secara dalam kerana ia mengguna pakai dengan menyuluruh keupayaan <i>multiple cores</i> di dalam pelayan moden. Enjin <i>shared-disk</i> membolehkan permintaan kepada pangkalan data diagihkan kepada mana-mana <i>instances</i>.</p>	<p><i>Partitioning Data</i> Kurang Sesuai Bagi Penskalaan Secara Dalam</p> <p>Arkitektur ini perlu merangkumi serta melibatkan pangkalan data dan enjin storan untuk menggunakan pendekatan penskalaan secara dalam. Ia juga memerlukan capaian aplikasi kepada setiap <i>instance</i> secara berasingan. Selain itu, arkitektur ini memerlukan setiap cakera dibahagikan kepada storan yang berbeza bagi setiap <i>instance</i>. Oleh yang demikian, keperluan-keperluan ini akan mengakibatkan pangkalan data sukar untuk diuruskan.</p>
Pengedaran Berdasarkan Geografi (<i>Geo-Distribution</i>)	<p>Isu <i>Latency</i> Bagi Lokasi yang Berbeza</p> <p>Arkitektur ini mempunyai isu <i>latency</i> bagi lokasi pelayan-pelayan yang berbeza dari segi geografinya. Isu ini hanya boleh diselesaikan sekiranya dua atau lebih sistem <i>shared-disk</i> dibangunkan secara berasingan.</p>	<p>Isu <i>Latency</i> Bagi Lokasi yang Berbeza</p> <p>Sama seperti arkitektur <i>shared-disk</i>, arkitektur ini mempunyai isu <i>latency</i> bagi lokasi server-server yang berbeza dari segi geografinya. Isu ini boleh diminimalkan dengan membahagi dan mengagih data berdasarkan geo-lokasi. Malah ia juga dapat mengurangkan atau menghapuskan fungsi/<i>data shipping</i> di antara dua lokasi berkenaan.</p>
Pengkomputeran Cloud : <i>Database-as-a-Service</i> (DaaS)	<p>Menyokong Kebolehskaaan Nodal Secara Dinamik</p> <p>Arkitektur ini adalah sesuai bagi penskalaan nodal atau keanjalan nodal (<i>nodal elasticity</i>). Memandangkan semua nod berhubung dengan <i>shared data store</i> yang</p>	<p>Pembahagian Tidak Optimal</p> <p>Pembahagian atau <i>partitioning</i> merupakan isu yang kerap dihadapi bagi pangkalan data <i>cloud</i>. Pembahagian ini dilaksanakan secara automasi dan sesuai bagi keanjalan nodal. Namun begitu, pembahagian secara automasi ini hanyalah separa optimal sahaja. Ia akan meningkatkan fungsi/<i>data-</i></p>

Ciri-ciri	Arkitektur <i>Shared-Disk</i>	Arkitektur <i>Shared-nothing</i>
	sama, bilangan nod-nod boleh diubah dengan mudah tanpa memberi sebarang implikasi kepada data.	<i>shipping</i> yang akan menyebabkan prestasi keseluruhan pangkalan data terganggu. Prestasi pangkalan data akan kian merosot sekiranya bilangan nod semakin bertambah.

- d) Tentukan sama ada arkitektur pangkalan data yang ingin dibangunkan merupakan arkitektur *shared-disk* ataupun *shared-nothing*. Penentuan arkitektur ini perlu diseimbangkan di antara halatuju agensi, kekangan-kekangan dan infrastruktur sedia ada dengan keperluan fungsian dan bukan fungsian yang telah diperolehi di fasa Analisis Keperluan.
- e) Berdasarkan dua jenis arkitektur pangkalan data seperti di atas dengan mengambil kira perbandingan di antara kedua-duanya, sediakan arkitektur yang ideal dan sesuai untuk rekabentuk pangkalan data yang akan dibangunkan.

Langkah 7 : Sediakan Arkitektur-arkitektur Secara Iteratif

Penyediaan dan pembangunan arkitektur perlu dilaksanakan secara iteratif bagi meningkatkan tahap komprehensif dan memenuhi keperluan-keperluan yang telah diperolehi. Disyorkan supaya iterasi proses penyediaan arkitektur dilaksanakan sekurang-kurangnya sebanyak dua kali untuk meminimalkan sebarang kesilapan dan kekurangan di dalam rekabentuk arkitektur.

RUJUKAN

1. Narayan Prusty, Modern Javascripts Application (2016).
2. Mark Richards, Software Architecture Patterns (2015).
3. Danny Brian, Kirk Knoernschild. Modern Web Architecutre (2016).
<http://www.gartner.com/>.
4. Ben Stopford. Shared Nothing v.s. Shared Disk Architectures: An Independent View (2009). <http://www.benstopford.com/2009/11/24/understanding-the-shared-nothing-architecture/>

4.6 Penentuan Teknologi [F3.2]

PENGENALAN

Penentuan teknologi dan *tool* yang sesuai bagi pembangunan sistem aplikasi merupakan salah satu aspek yang penting untuk dipertimbangkan. Teknologi dan *tool* yang dipilih akan digunakan bagi memandu dalam reka bentuk antaramuka, reka bentuk proses dan reka bentuk logikal pangkalan data. Teknologi yang dipilih juga akan digunakan semasa pelaksanaan fasa pembangunan, pengujian, pengoperasian dan penyelenggaraan sistem aplikasi. Bagi menentukan teknologi yang akan digunakan, beberapa aspek perlu diambil kira :

- a) Memenuhi dan mematuhi reka bentuk arkitektur sistem aplikasi
- b) Keserasian dengan keperluan fungsian dan bukan fungsian
- c) Selaras dengan visi dan misi organisasi
- d) Teknologi-teknologi yang mudah diperolehi dan diselenggara

MATRIKS ALTERNATIF

Matriks alternatif ialah salah satu kaedah yang digunakan untuk memilih teknologi yang memenuhi keperluan. Matriks alternatif digunakan sebagai kaedah untuk menentukan rekabentuk yang akan dibangunkan. Matriks alternatif juga digunakan untuk mengurus dan menyusun reka bentuk alternatif supaya penyelesaian yang terbaik dapat diperolehi. Matriks alternatif menggabungkan beberapa analisa kebolehlaksanaan seperti berikut:

- a) Kebolehlaksanaan Teknikal - penilaian kematangan atau keupayaan teknologi untuk berfungsi dengan teknologi yang lain.
- b) Kebolehlaksanaan Operasi - kesesuaian dan kesesuaian pihak pengurusan, pegawai dan pengguna dengan teknologi yang dicadangkan.
- c) Kebolehlaksanaan Ekonomi - penilaian sama ada teknologi yang digunakan berpatutan dan kos efektif.

Bagi setiap kategori *tools* yang diperlukan, analisis matriks alternatif perlu dijalankan. Ini kerana setiap kategori *tool* mempunyai kriteria-kriteria yang berebeza. Matriks alternatif boleh disediakan seperti dalam contoh di bawah. Tahap/Pemberat dan skor-skor dirangkumi bersekali bagi mewujudkan kad skor yang mengenalpasti kriteria-kriteria utama projek dan pilihan alternatif yang terbaik.

Contoh analisis matriks alternatif adalah seperti berikut:

Sebuah organisasi ingin membuat pilihan yang terbaik di antara bahasa pengaturcaraan yang akan digunakan dalam sistem aplikasi yang ingin dibangunkan. Sebanyak 10 kriteria telah ditentukan bagi membantu organisasi dalam membuat pemilihan tersebut. Terdapat tiga jenis skema penilaian yang merangkumi nilai skor dari 1 sehingga 10 bertujuan untuk menyediakan kad skor dalam menentukan alternatif yang terbaik. Jadual matriks alternatif

seperti di bawah digunakan untuk membantu organisasi berkenaan melakukan penilaian kepada bahasa pengaturcaraan yang sesuai. Bahasa pengaturcaraan **Java** telah dikenalpasti sebagai pilihan yang terbaik bagi sistem aplikasi yang ingin dibangunkan.

Jadual 34 : Nilai Skor Skema Penilaian Teknikal

Skema Penilaian	
Tinggi	8 sehingga 10
Sederhana	5 sehingga 7
Rendah	1 sehingga 4

Jadual 35 : Matriks Alternatif Bagi Penentuan Bahasa Pengaturcaraan

No.	Kriteria	Bahasa Pengaturcaraan		
		Java	C#	Phyton
		Nilai Skor		
1.	Pengetahuan, pengalaman dan kemahiran pasukan pembangun	8	6	1
2.	Ketersediaan (<i>availability</i>) pengaturcara	9	5	1
3.	Ketersediaan <i>Integrated Development Environment</i> (IDE) dan <i>tools</i> di pasaran	8	8	6
4.	Kemudahan integrasi	8	8	8
5.	Penjimatan kos	9	6	9
6.	Prestasi	8	8	8
7.	Sekuriti	8	8	8
8.	Sokongan dan komuniti	8	9	6
9.	Keanjalan bahasa pengaturcaraan	8	8	8
10.	Tren semasa	9	6	4
JUMLAH		83	72	59

4.7 REKABENTUK PANGKALAN DATA [F3.3]

KETERANGAN

Rekabentuk Pangkalan Data Logikal merupakan aktiviti untuk menterjemah model maklumat konseptual kepada model maklumat logikal. Model maklumat logikal merupakan model perantara yang akan digunakan untuk merekabentuk pangkalan data fizikal, seterusnya membangunkan pangkalan data yang sebenar. Model ini menerangkan empat (4) komponen utama iaitu spesifikasi jadual (*table specification*), spesifikasi medan (*column specification*), spesifikasi kekunci primer (*primary key specification*) dan spesifikasi kekunci asing (*foreign key specification*).

Peristilahan yang digunakan dalam penterjemahan model maklumat konseptual kepada model maklumat logikal adalah seperti berikut.

Model Konseptual		Model Logikal
Entiti	→	Jadual
Atribut	→	Medan
UID Primer	→	Kekunci Primer
UID Sekunder	→	Kekunci Unik
Hubungan antara entiti	→	Kekunci Asing
Peraturan bisnes	→	<i>Check Constraint</i>

OBJEKTIF

- Menyediakan model maklumat logikal.
- Mengenalpasti spesifikasi jadual dan medan.
- Mengenalpasti spesifikasi kekunci utama dan kekunci asing.
- Implementasi entiti berjenis *Super-type* dan *Sub-type*.

NOTASI

Model maklumat logikal adalah berdasarkan model maklumat konseptual rujuk **Pemodelan Keperluan Data [F2.2]**:

Jadual 36 : Notasi Rekabentuk Keperluan Data

Elemen	Keterangan
Jadual <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> NAMA_JADUAL </div>	<ul style="list-style-type: none"> • Digunakan bagi mewakili setiap jadual.

Medan				
NAMA_JADUAL				
jenis kunci	simbol *	nama medan	jenis data	Panjang data

- Jenis Kunci
Terdapat beberapa jenis kunci bagi medan yang terdapat dalam jadual. Antaranya ialah:
P – Kunci primer (*Primary key*)
U – Kunci unik (*Unique key*)
F – Kunci asing (*Foreign key*)
- Simbol * menunjukkan medan wajib diisi (*not null column*) manakala sekiranya tiada simbol * menunjukkan medan tidak wajib diisi (*null column*).
- Jenis data bagi setiap medan boleh sama ada dalam format numerik, alfanumerik, aksara, tarikh, masa atau fail (seperti fail imej/audio/video/multimedia).
- Panjang (*length*) data bagi jenis data tersebut dinyatakan dalam tanda kurungan ‘()’ – jika ada.

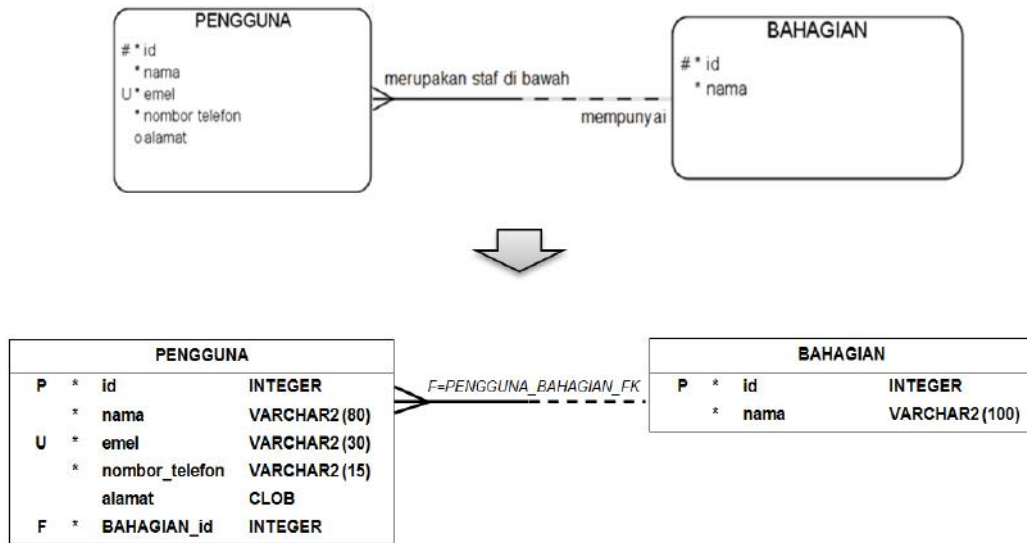
LANGKAH-LANGKAH

Langkah 1 : Sediakan Spesifikasi Jadual

- Kenalpasti entiti yang mempunyai maklumat yang tidak perlu disimpan dalam pangkalan data terlebih dahulu. Entiti tersebut tidak perlu diterjemahkan ke dalam model maklumat logikal.
- Setiap entiti tunggal (bukan berjenis Entiti *Super-type* atau *Sub-type*) akan diterjemahkan terus kepada jadual. Contoh penterjemahan adalah seperti berikut.

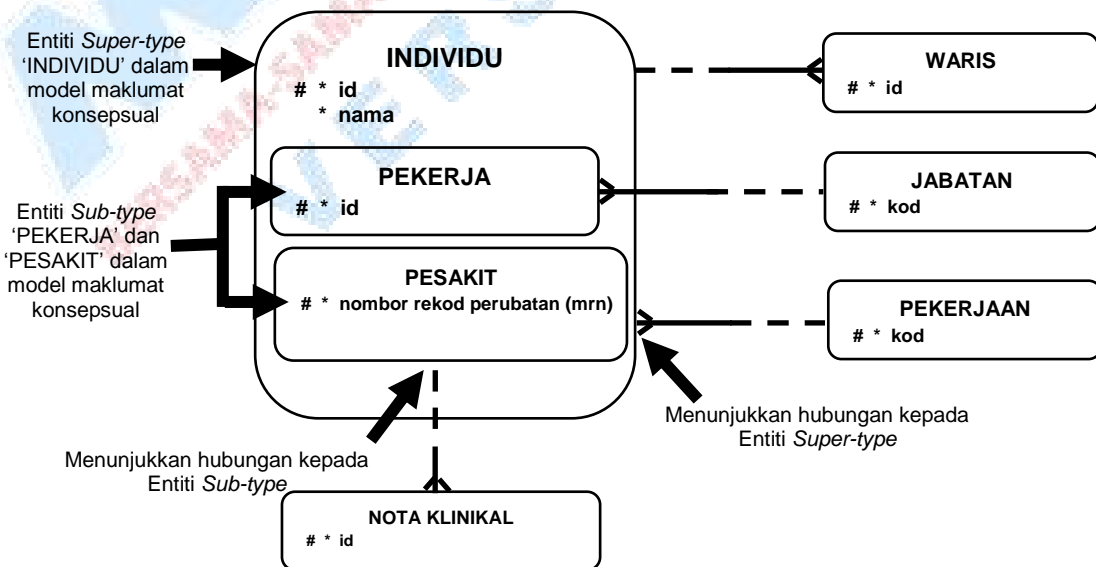
Model Konseptual		Model Logikal
Nama Entiti		Nama Jadual
PENGGUNA	→	PENGGUNA
ASET	→	ASET
KATEGORI ASET	→	KATEGORI_ASET
TEMPAHAN ASET	→	TEMPAHAN_ASET

Contoh entiti tunggal yang diterjemahkan terus kepada jadual adalah seperti di rajah dibawah.



Rajah 63 : ERD - Entiti Tunggal

- c) Sekiranya *Intersection Entity* diwujudkan bagi menyelesaikan hubungan banyak-ke-banyak (*many-to-many*) di antara dua entiti, terjemahkan model tersebut kepada model maklumat logikal adalah sama seperti entiti tunggal.
- d) Sekiranya Entiti Super-type atau Sub-type wujud, penterjemahan ke model logikal akan membabitkan hubungan dengan entiti lain sama ada kepada Entiti *Super-type* atau *Sub-type* tersebut.



Rajah 64 : ERD – Penggunaan Entiti Supertype

Terdapat **tiga pilihan** untuk menterjemahkan model maklumat konseptual ini kepada model maklumat logikal iaitu :

i) Pilihan 1 – Pelaksanaan *Super-type*

Pilihan ini akan menghasilkan satu jadual tunggal sahaja bagi pelaksanaan ketiga-tiga Entiti INDIVIDU, PEKERJA dan PESAKIT. Pelaksanaan ini juga dikenali sebagai **pelaksanaan satu jadual tunggal**.

ii) Pilihan 2 – Pelaksanaan *Sub-type*

Pilihan ini akan menghasilkan **satu jadual bagi setiap Entiti *Sub-type***. Bagi contoh ini, dua jadual akan dihasilkan iaitu Jadual PEKERJA dan PESAKIT.

iii) Pilihan 3 – Pelaksanaan kedua-dua *Super-type* dan *Sub-type* (“Arc”)

Pilihan ini akan menghasilkan **satu jadual bagi setiap entiti** sama ada entiti tersebut berjenis *Super-type* atau *Sub-type*. Bagi contoh ini, tiga jadual akan dihasilkan iaitu Jadual INDIVIDU, PEKERJA dan PESAKIT

- e) Nama jadual mesti dimulakan dengan huruf. Gantikan ruang kosong atau aksara khas yang tidak dibenarkan seperti %, *, –, !, / dan lain-lain kepada aksara garis bawah ‘_’ (underscore). Elakkan penggunaan perkataan-perkataan rizab (reserved words) yang biasa terdapat dalam bahasa pengaturcaraan seperti *number*, *value*, *type* dan lain-lain.
- f) Nama jadual mestilah unik (nama yang sama tidak boleh digunakan berulang kali) dalam satu skema pangkalan data.

Langkah 2 : Sediakan Spesifikasi Medan

- a) Setiap atribut akan diterjemahkan kepada medan (*field*).
- b) Nama atribut akan menjadi nama medan (*field*). Nama medan mesti dimulakan dengan huruf. Gantikan ruang kosong/aksara khas yang tidak dibenarkan kepada aksara garis bawah ‘_’. Elakkan penggunaan perkataan-perkataan rizab, dan beri nama singkatan jika boleh.
- c) Nama medan mestilah unik (nama yang sama tidak boleh digunakan berulang kali) dalam satu jadual.
- d) Atribut Mandatori akan menjadi medan wajib diisi (*not-null column*), manakala Atribut Pilihan menjadi medan tidak wajib diisi (*null column*).
- e) Tentukan jenis data (*data type*) bagi setiap medan. Jenis data boleh sama ada dalam format numerik, alfanumerik, aksara, tarikh, masa atau fail (seperti fail imej/audio/video/multimedia). Nyatakan panjang (*length*) bagi jenis data tersebut dalam tanda kurungan ‘()’ – jika ada.
- f) Sesetengah peraturan bisnes diterjemahkan kepada CHECK Constraint bagi memastikan data yang sah sahaja diterima. Contohnya medan ‘tarikh_tamat_guna’

dalam jadual `TEMPAHAN_ASET` mestilah lebih besar atau sama dengan medan 'tarikh_mula_guna'.

1. tarikh_tamat_guna >= tarikh_mula_guna

- g) Bagi peraturan bisnes yang lebih kompleks, pengekodan/pengaturcaraan tambahan mungkin diperlukan sama ada di komputer pelayan (*server site*), di komputer pelanggan/klien (*client site*), atau kedua-duanya sekali.

Langkah 3 : Sediakan Spesifikasi Kekunci Primer

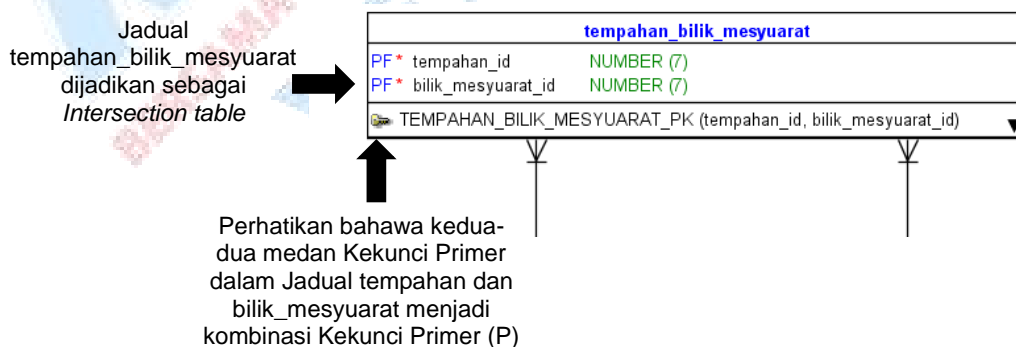
- a) Terjemahkan UID Primer menjadi Kekunci Primer (*Primary Key*).
- b) Terjemahkan UID Sekunder akan menjadi Kekunci Unik (*Unique Key*).

Langkah 4 : Sediakan Spesifikasi Kekunci Asing

- a) Hubungan antara dua entiti akan diterjemah menjadi Kekunci Asing (*Foreign Key*).
- b) Medan Kekunci Asing biasanya dinamakan dengan menggabungkan nama jadual yang dirujuk, dengan nama atribut yang menjadi Kekunci Primer dalam jadual yang dirujuk itu.

Langkah 5 : Sediakan Spesifikasi Medan *Intersection Entity*

- a) Bagi model yang mempunyai *Intersection Entity* yang menyelesaikan hubungan banyak-ke-banyak (*many-to-many*) di antara dua entiti, UID Primer daripada kedua-dua entiti akan digabungkan untuk menjadi UID Primer bagi *Intersection Entity* tersebut.



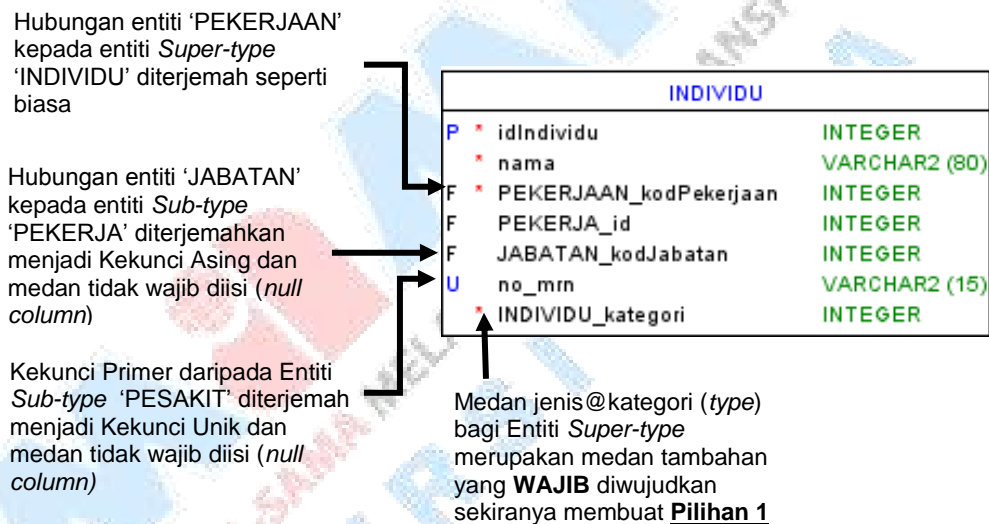
Rajah 65 : Kekunci Primer daripada *composite key*

- b) Dalam model logikal, gabungan UID Primer daripada kedua-dua entiti akan menjadi *Composite Primary key*.

- c) Merujuk kepada rajah di atas, medan-medan yang terdapat dalam *Intersection Table* 'ITEM_TEMPAHAN', hubungan antara dua entiti diterjemah menjadi Kekunci Asing F₁ dan F₂, dan kedua-dua Kekunci Asing ini membentuk kombinasi Kekunci Primer (P) (*composite primary key*) dalam jadual berkenaan. Sama ada modaliti hubungan bersifat mandatori atau sebaliknya, medan Kekunci Asing dalam *Intersection Table* akan sentiasa menjadi medan wajib diisi (*not-null column*).

Langkah 6 : Sediakan Spesifikasi Medan Entiti *Super-Type* Dan *Sub-Type*

- a) Terjemahan bagi atribut yang terlibat dalam Entiti *Super-type* atau *Sub-type*, terdapat beberapa peraturan bagi pilihan yang perlu dipatuhi mengikut jenis terjemahan model maklumat.
- b) **Pilihan 1 – Pelaksanaan *Super-type***



Rajah 66 : ERD – Penggunaan Entiti Supertype

- Atribut yang terdapat dalam Entiti *Super-type* diterjemahkan terus seperti biasa.
- Kesemua atribut daripada Entiti *Sub-type* akan menjadi medan tidak wajib diisi (*null column*). Sekiranya atribut tersebut merupakan UID, atribut diterjemah menjadi Kekunci Unik (*Unique Key*) atau *CHECK Constraint*.
- Satu medan mandatori **WAJIB** diwujudkan bagi membezakan antara Entiti *Sub-type* tersebut. Medan tersebut biasanya dinamakan dengan **<nama_entiti_supertype>_jenis@kategori** seperti **INDIVIDU_kategori** kerana medan inilah yang akan membezakan sama ada INDIVIDU tersebut kategori

PESAKIT atau PEKERJA. Penggunaan *CHECK Constraint* atau tambahan pengekodan dalam aturcara aplikasi boleh diaplikasikan di sini bagi menjamin integriti/kesahihan data.

- iv) Hubungan kepada Entiti *Super-type* juga diterjemahkan terus seperti biasa, manakala Hubungan kepada Entiti *Sub-type* akan diterjemah menjadi Kekunci Asing dan menjadi medan yang tidak wajib diisi (*null column*).

c) **Pilihan 2 – Pelaksanaan Sub-type**

Atribut yang terdapat dalam Entiti *Sub-type* diterjemahkan terus seperti biasa ke dalam jadual yang berasingan. Manakala Kekunci Primer bagi atribut yang terdapat dalam Entiti *Sub-type* adalah UID Primer Entiti *Super-type*.

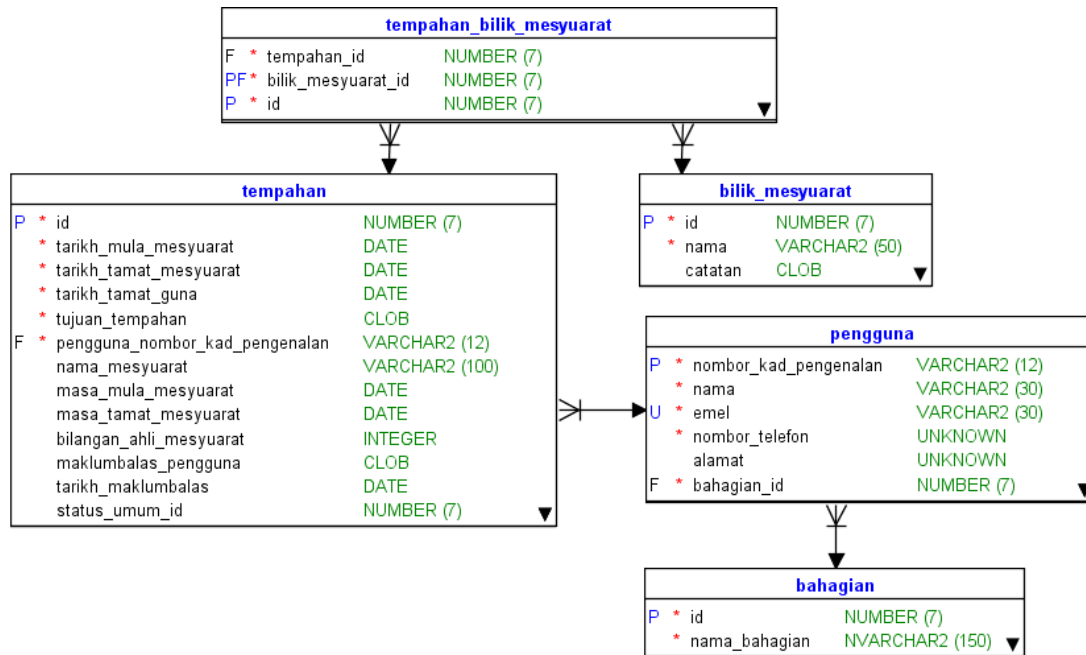
d) **Pilihan 3 – Pelaksanaan kedua-dua Super-type dan Sub-type (Arc)**

- i) Atribut yang terdapat dalam kesemua entiti tersebut diterjemahkan seperti biasa ke dalam jadual yang berasingan. Manakala Kekunci Primer bagi atribut yang terdapat dalam Entiti *Super-type* dan *Sub-type* adalah UID Primer Entiti *Super-type*.
- ii) Bagi Entiti *Super-type*, satu medan mandatori **WAJIB** diwujudkan bagi membezakan antara Entiti *Sub-type* tersebut. Medan tersebut biasanya dinamakan dengan **<nama_entiti_supertype>_jenis@kategori** seperti **INDIVIDU_kategori** kerana medan inilah yang akan membezakan sama ada INDIVIDU tersebut kategori PESAKIT atau PEKERJA. Penggunaan *CHECK Constraint* atau tambahan pengekodan dalam aturcara aplikasi boleh diaplikasikan di sini bagi menjamin integriti/kesahihan data.

Langkah 6 : Perkemaskan Model Maklumat Logikal

Setelah semua entiti siap diterjemah ke model logikal, lengkapkan dan perkemaskan lagi model maklumat logikal mengikut jenis teknologi yang hendak digunakan.

Contoh model maklumat logikal adalah seperti rajah berikut.



Rajah 67 : Contoh Model Maklumat Logikal Sistem Tempahan Bilik Mesyuarat (eTempah)

Langkah 7 : Dokumentasikan Model Maklumat Logikal

Dokumentasikan semua output yang dihasilkan sebagai hasil serahan proses rekabentuk pangkalan data logikal ke dalam **D04 Spesifikasi Rekabentuk Sistem**. Dokumentasi mengikut susunan berikut:

- Model Maklumat Logikal
- Rujuk **Apendiks 4 Template Skema Logikal Pangkalan Data (Database Logical Schema)**.

RUJUKAN

- https://en.wikipedia.org/wiki/Data_modeling#Conceptual.2C_logical_and_physical_schemas
- Jan Speelpenning, Patrice Daux and Jeff Gallus; Data Modeling and Relational Database Design (2001)

4.8 REKABENTUK ANTARAMUKA PENGGUNA [F3.4]

KETERANGAN

Penyediaan Rekabentuk Antaramuka Pengguna (UI) adalah proses untuk menentukan kaedah interaksi di antara pengguna dengan sistem yang akan dibangunkan. Memberi keutamaan kepada rekabentuk antaramuka pengguna, khususnya kepada peningkatan *user experience* (UX), dapat menjadikan sesuatu aplikasi mudah untuk dilayari, efektif dan selesa untuk digunakan.

Di samping itu, antaramuka-antaramuka pengguna yang telah dibangunkan perlu dipadankan dengan medan-medan data di dalam jadual (*table*) pangkalan data. Pemetaan data ini bertujuan untuk memudahkan seseorang pembangun sistem mengetahui senarai medan data yang diperlukan bagi satu-satu antaramuka pengguna yang telah dibangunkan.

OBJEKTIF

- Membangunkan antaramuka pengguna yang berpandukan kepada rangka kerja, prinsip dan elemen asas UI/UX dan selaras dengan *trend* rekabentuk yang terkini.
- Menyediakan jadual rujukan bagi pemetaan di antara antaramuka pengguna dengan medan di dalam pangkalan data.

LANGKAH-LANGKAH

Langkah 1 : Kenalpasti Dan Pilih Use Case

- a) Rujuk kepada rajah *Use Case* yang telah dibangunkan di dalam Pemodelan *Use Case* (Fungsian) [F2.1].
- b) Kenalpasti dan pilih setiap *Use Case* yang terlibat untuk menyediakan semua rekabentuk antaramuka pengguna yang terlibat.

Langkah 2 : Kenalpasti Aliran Data

- a) Rujuk kepada Rajah Aliran Data (DFD) yang telah disediakan di dalam Pemodelan Proses Sistem [F2.3].
- b) Kenalpasti aliran data yang keluar dan masuk dari fungsi yang berkaitan dengan rekabentuk antaramuka pengguna yang ingin dibangunkan

Langkah 3 : Kenalpasti Elemen-elemen Data

- a) Rujuk kepada rajah hubungan entiti (ERD) dan logikal pangkalan data yang telah disediakan di dalam Pemodelan Keperluan Data [F2.2] dan Rekabentuk Pangkalan Data [F3.4].

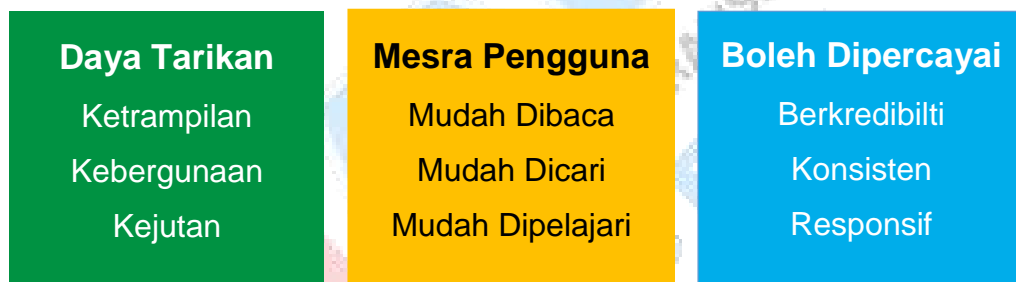
- b) Kenalpasti elemen-elemen data yang akan diguna pakai di dalam ruangan-ruangan teks rekabentuk antaramuka pengguna.

Langkah 4 : Pertimbangkan Keperluan *User Experience* (UX)

- a) Lakukan pertimbangan keperluan UX sebelum antaramuka-antaramuka pengguna dibangunkan. Pertimbangan keperluan UX ini boleh dilaksanakan dengan berpandukan kepada rangka kerja seperti berikut:

Rangka Kerja Ciri-ciri *User Experience* (UX)

Rangka Kerja Ciri-ciri UX merupakan panduan asas kepada pereka dan pembangun sistem dalam menyediakan rekabentuk antaramuka pengguna yang efektif serta memiliki tahap kebolegunaan yang tinggi. Rangka kerja ini mengambil kira kajian-kajian yang telah dilaksanakan berkenaan dengan *human-computer interaction* (HCI), *human factors* (HFs) dan *user-centered design* (UCD). Rangka kerja yang dimaksudkan adalah seperti berikut :-



Rajah 68 : Rangka Kerja Ciri-ciri *User Experience*

i) Daya Tarikan

Daya tarikan adalah merupakan keupayaan untuk menarik perhatian dan membangkitkan minat pengguna untuk melayari aplikasi yang dibangunkan. Ciri-cirinya merangkumi kepada ketrampilan sesuatu antaramuka pengguna, kebolegunaan sesuatu aplikasi dan persembahan elemen-elemen kejutan kepada pengguna.

- **Keterampilan**

Keterampilan visual akan mempengaruhi kepada tanggapan dan ekspektasi pengguna kepada sesuatu aplikasi. Penekanan kepada warna, *font*, imej dan susun atur yang kemas dan konsisten mampu untuk meningkatkan keselesaan pengguna kepada aplikasi yang dilayari. Trend-trend terkini yang selaras dengan Rekabentuk *Flat 2.0*, seperti rekabentuk material yang dipopularkan oleh Syarikat Google dan Metro UI yang diguna pakai di dalam sistem pengoperasian Windows, boleh dijadikan sebagai rujukan dalam usaha untuk mempertingkatkan ketrampilan visual aplikasi yang dibangunkan.

- **Kebergunaan**

Kebergunaan antaramuka yang dibangunkan akan menentukan sama ada aplikasi akan terus dan kerap digunakan oleh pengguna. Aplikasi yang dibangunkan perlu sentiasa menitik beratkan nilai praktikaliti serta dapat membantu pengguna melaksanakan tugas dengan lancar.

- **Kejutan**

Inovasi di dalam rekabentuk antaramuka yang melangkaui ekspektasi pengguna mampu untuk meningkatkan daya tarikan sesuatu aplikasi. Inovasi di dalam rekabentuk antaramuka bukan sahaja tertakluk kepada kemajuan perisian terkini, malah kemudahan-kemudahan yang dimiliki oleh perkakasan pintar seperti kemudahan GPS atau kamera pada telefon bimbit boleh juga diintegrasikan dengan aplikasi untuk menjadikan sesuatu rekabentuk antaramuka kelihatan lebih moden dan canggih.

ii) **Mesra Pengguna**

Mesra pengguna merupakan keupayaan pengguna untuk mempelajari dan memahami penggunaan sesuatu aplikasi. Ciri-cirinya merangkumi kebolehan sesuatu aplikasi untuk memudahkan pengguna membaca kandungannya, melakukan carian dan mempelajari penggunaannya.

- **Mudah Dibaca**

Pereka bentuk perlulah terdahulu mengenal pasti kategori dan jenis pengguna aplikasi sebelum sesuatu rekabentuk antaramuka dibangunkan. Pengguna-pengguna aplikasi boleh terdiri sama ada daripada orang awam, kakitangan kerajaan, kumpulan profesional, golongan belia ataupun muda. Oleh yang demikian, seseorang pereka bentuk perlulah bijak dalam menentukan saiz dan warna *font* yang sesuai serta cerdik dalam menyusun atur kandungan aplikasi dengan kemas dan tidak mengelirukan.

- **Mudah Dicari**

Kebolehcarian sesuatu aplikasi diukur melalui bagaimana mudahnya seseorang pengguna melakukan carian kepada sesuatu maklumat yang tertentu. Contohnya, antaramuka yang dibangunkan perlulah dilengkapi dengan fungsi carian yang pintar, serta susun atur kandungan dan menu perlu diletakkan di ruangan yang strategik.

- **Mudah Dipelajari**

Rekabentuk antaramuka yang efisien perlu mempunyai konsep dan aliran kerja yang konsisten bagi membolehkan aplikasi mudah untuk difahami dan dipelajari oleh pengguna. Penyediaan manual pengguna atau fungsi bantuan

berbentuk interaktif bukan sahaja dapat membantu, malah dapat mempercepatkan lagi proses pembelajaran bagi seseorang pengguna.

iii) **Boleh Dipercayai**

Boleh dipercayai merupakan keupayaan sesuatu aplikasi untuk mendapatkan kepercayaan daripada penggunanya. Ciri-cirinya merangkumi kebolehan aplikasi untuk meningkatkan imej kredibiliti organisasi yang memiliki aplikasi berkenaan (*owner*), menunjukkan konsistensi di dalam rekabentuk antaramukanya serta memiliki tahap responsif yang tinggi.

- **Berkredibiliti**

Kualiti dan pemilihan visual yang bersesuaian bagi antaramuka pengguna sesuatu aplikasi, contohnya dari segi gambar yang dipaparkan, ikon yang digunakan serta susun atur yang kemas, mampu untuk meningkatkan imej kredibiliti sesuatu organisasi dari perspektif pengguna. Pemilihan dan kawalan kualiti visual ini bukan sahaja bertujuan untuk menarik perhatian pengguna, malah ia juga menggambarkan tahap profesional organisasi yang memiliki aplikasi berkenaan.

- **Konsisten**

Elemen-elemen antarmuka pengguna seperti warna, *font*, susun atur dan label-label perlu diselaraskan bagi mencapai aplikasi yang konsisten. Penyelarasan antaramuka pengguna juga dapat membantu pengguna untuk memahami dan mempelajari penggunaan aplikasi dengan lebih pantas.

- **Responsif**

Kelajuan maklumbalas dan prestasi sesuatu aplikasi perlu dititik beratkan di dalam rekabentuk antaramuka pengguna. Saiz imej yang dipaparkan serta teknologi antaramuka yang dipilih perlu bersesuaian dengan kebolehan infrastruktur sistem dan selari dengan keperluan yang diperolehi.

Langkah 5 : Bangunkan Rekabentuk Antaramuka Pengguna

Bangunkan semua rekabentuk antaramuka pengguna untuk memenuhi setiap *use case* yang telah dikenalpasti di dalam fasa analisa dengan berpandukan kepada Rangka Kerja Ciri-ciri *User Experience* (UX) termasuk Prinsip dan Elemen Asas Rekabentuk Antaramuka Pengguna seperti berikut:

Prinsip Dan Elemen Asas Rekabentuk Antaramuka Pengguna (UI)

Prinsip dan Elemen Asas Rekabentuk Antaramuka Pengguna adalah seperti berikut:

a) Garisan

Garisan (*line*) merupakan asas pembentukkan semua komponen dalam rekabentuk visual. Penggunaan garisan di dalam antaramuka pengguna adalah bertujuan untuk memberi penekanan dan menarik perhatian pengguna kepada ruangan-ruangan tertentu, contohnya garisan disertakan dalam satu-satu muka web untuk memisah dan menonjolkan ruangan seperti kandungan, tajuk, label, pautan dan panel sisi.

b) Warna

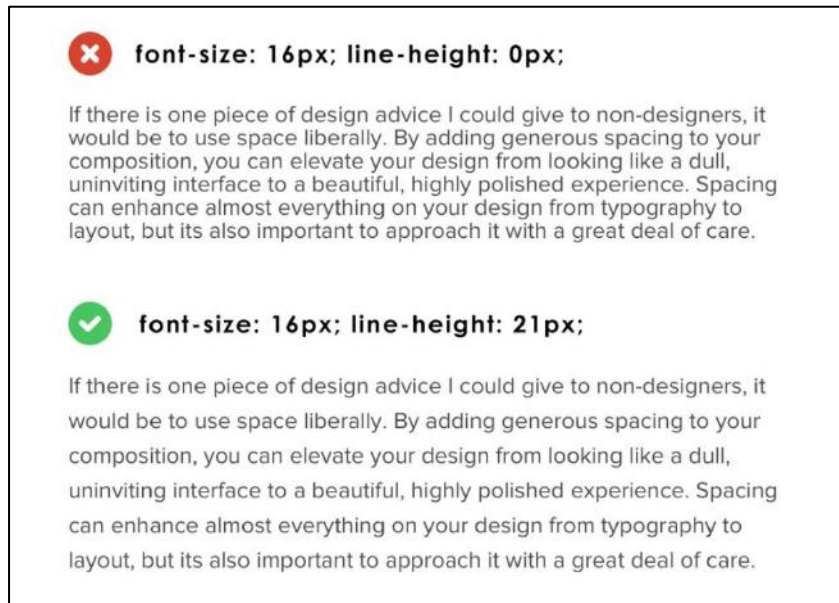
Penerapan warna di dalam antaramuka pengguna adalah bertujuan untuk mencetus suasana dan emosi yang spesifik kepada pengguna, malah ia juga merupakan satu mekanisme penceritaan berkenaan dengan satu-satu organisasi, produk dan perkhidmatan yang terlibat. Penggunaan warna di dalam antaramuka pengguna boleh berdasarkan kepada peraturan 60-30-10, di mana 60% ruangan dalam satu-satu muka web perlu mengandungi penggunaan warna dominan, 30% merupakan warna sekunder dan 10% pula adalah warna *accent*. Namun begitu, pematuhan kepada peraturan ini bukanlah mandatori oleh kerana bilangan dan jumlah peratusan warna yang ingin digunakan adalah bergantung kepada kompetensi dan kreativiti seseorang pereka antaramuka dalam mengharmonikan serta mengimbangi komposisi warna-warna yang dipilih.

c) Tipografi

Tipografi merupakan elemen utama antaramuka untuk menyampaikan mesej dan maklumat kepada pengguna. Oleh yang demikian, pereka antaramuka perlulah memilih *typeface* atau *font family* yang sesuai mengikut ruangan-ruangan di dalam satu-satu antaramuka pengguna, sama ada ruangan tersebut merupakan tajuk, logo, label dan teks kandungan. *Font family* yang dipilih secara amnya perlu mudah untuk dibaca, jelas dan bersesuaian dengan tema yang ingin disampaikan. Sebagai contoh, *font family* yang standard seperti Arial, Sans Serif, Calibri dan Trebuchet MS sesuai digunakan bagi antaramuka yang berteraskan kepada aplikasi bisnes dan profesional. Bagi ruangan-ruangan yang ingin ditonjolkan seperti tajuk atau logo, pereka antaramuka boleh mempraktikkan kreativiti masing-masing dengan menggunakan *font family* yang lebih progresif dan unik bertujuan untuk menarik perhatian pengguna kepada aplikasi yang dibangunkan. Namun begitu, perlu diingatkan bahawa bilangan jenis *font family* dalam antaramuka pengguna tidak seharusnya digunakan secara berlebihan. Secara amnya, jumlah bilangan *font family* yang dipilih dalam satu-satu antaramuka aplikasi tidak seharusnya melebihi dari tiga jenis.

Salah satu aspek lain yang perlu juga dilihat dalam tipografi adalah jarak di antara satu baris ayat dengan baris ayat seterusnya, atau terma teknikalnya lebih dikenali sebagai *line-height*. Pereka antaramuka perlulah menyediakan jarak yang bersesuaian supaya kandungan maklumat atau label yang dipaparkan tidaklah terlalu rapat dan menyukarkan untuk dibaca oleh pengguna. Secara amnya, jumlah jarak di antara baris ayat atas dan bawah adalah 30% lebih besar dari saiz *font* yang ditetapkan. Contohnya, sekiranya saiz *font* yang diguna pakai adalah 16 piksel, nilai *line-height* atau jarak di antara baris adalah

sebanyak 21 piksel. Rujuk rajah di bawah bagi melihat contoh perbezaan di antara penggunaan *line height* yang ideal dalam antaramuka pengguna.



Rajah 69 : Perbezaan Bagi Penggunaan *Line-Height* Yang Berkesan Dalam Antaramuka Pengguna

d) Ruang Negatif (*Negative Space*)

Ruang negatif, atau juga dikenali sebagai *white space*, adalah ruangan yang dibiarkan kosong di antara satu komponen antaramuka pengguna dengan komponen-komponen yang lain. Ruang negatif diimplementasi bertujuan untuk menarik fokus pengguna kepada maklumat atau mesej yang ingin ditonjolkan, memudahkan pengguna berinteraksi dan melakukan navigasi dalam aplikasi, serta ia dapat meningkatkan kekemasan satu-satu antaramuka. Penggunaan ruang negatif yang efisien adalah sejajar dengan pendekatan dan trend rekabentuk minimalis yang kini kian popular digunakan.

Penggunaan ruang negatif yang efisien boleh dilaksanakan dengan meningkatkan saiz margin di antara komponen-komponen antaramuka seperti gambar, logo, tajuk, label dan kotak teks. Dalam masa yang sama, komponen-komponen, informasi dan kandungan yang disertakan dalam setiap antaramuka tidak perlulah terlalu banyak sehingga membuatkan skrin tersebut menjadi terlalu sesak dan sukar untuk dibaca. Rujuk rajah di bawah untuk melihat contoh penggunaan ruang negatif yang efektif dalam satu-satu muka web atau aplikasi.

Rajah 70 : Contoh Penggunaan Ruang Negatif Yang Berkesan

Berikut adalah satu contoh rekabentuk antaramuka pengguna bagi menu Kemaskini Profil Pengguna di bawah Sistem Tempahan Bilik Mesyuarat (eTempah):

Rajah 71 : Contoh Rekabentuk Antaramuka Pengguna Bagi Menu Luluskan Tempahan Bilik Mesyuarat, Sistem Tempahan Bilik Mesyuarat (eTempah)

Langkah 5 : Padankan Rekabentuk Antaramuka Pengguna Dengan Elemen Data

- Padankan rekabentuk antaramuka pengguna dan medan data yang berkaitan dengan melengkapkan Apendiks 6 Templat Pemetaan Data Antaramuka seperti di.
- Selain daripada medan data, masukkan juga objek-objek lain, seperti butang-butang atau pautan, di dalam Jadual Pemetaan Data. Gunakan notasi '<Nama Objek>' apabila merekodkan objek berkenaan. Contoh penulisan menggunakan notasi objek butang adalah seperti <Hantar>, <Simpan> dan <Keluar>.
- Isikan maklumat-maklumat berikut:

Jadual 37 : Keterangan Medan-Medan Templat Pemetaan Data

Medan	Keterangan
Nama Label	Nama label bagi elemen-elemen seperti <i>textbox</i> , <i>textarea</i> atau <i>dropdown menu</i> di dalam rekabentuk antaramuka pengguna.
Nama Jadual	Nama jadual (<i>table</i>) di mana medan data disimpan bagi elemen di dalam rekabentuk antaramuka pengguna.
Nama Medan Data	Nama medan data yang terlibat bagi elemen di dalam rekabentuk antaramuka pengguna.
Create, Update, Read, Delete (CRUD)	<p>CRUD merujuk kepada empat fungsi utama yang dilaksanakan dalam pangkalan data. Isikan jenis fungsi utama berdasarkan keperluan proses dan rekabentuk antaramuka yang telah dibangunkan. Penerangan lanjut berkenaan fungsi-fungsi CRUD adalah seperti berikut :</p> <ol style="list-style-type: none"> <i>Create</i> (C) adalah merujuk kepada data yang baru diwujudkan atau direkodkan <i>Read</i> (R) adalah merujuk kepada data yang dikeluarkan untuk paparan <i>Update</i> (U) adalah merujuk kepada data yang telah wujud sebelum ini dan hanya perlu dikemaskini. <i>Delete</i> (D) adalah merujuk kepada data yang ingin dihapuskan

Catatan	<ul style="list-style-type: none"> • Ruang catatan adalah bertujuan untuk memasukkan keterangan tambahan ataupun peraturan yang berkenaan bagi medan-medan data yang terlibat. • Ruang ini adalah bukanlah mandatori dan boleh digabungkan dengan medan-medan data yang lain sekiranya catatan yang dimasukkan mempunyai ciri-ciri yang sama.
----------------	---

- d) Rujuk jadual di bawah bagi contoh pengisian Templat Pemetaan Data berdasarkan kepada contoh antaramuka pengguna yang telah disediakan pada langkah 4.

Jadual 38 : Contoh Pengisian Templat Pemetaan Data Bagi Menu Luluskan Tempahan Bilik Mesyuarat, Sistem Tempahan Bilik Mesyuarat (eTempah)

Nama Label	Jenis Objek	Nama Jadual	Nama Medan Data	CRUD	Catatan
Nama Mesyuarat	Data	tempahan	nama_mesyuarat	R	
Nama Bilik Mesyuarat	Data	tempahan tempahan_bilik_mesyuarat bilik_mesyuarat	id tempahan_id_bilik_mesyuarat_id id_nama	R	Paparkan semua senarai bilik dalam satu-satu tempahan
Tarikh Mula Mesyuarat	Data	tempahan	tarikh_mula_mesyuarat	R	Format tarikh adalah dalam bentuk 'DD/MM/YYYY'.
Tarikh Tamat Mesyuarat	Data	tempahan	tarikh_tamat_mesyuarat	R	Format tarikh adalah dalam bentuk 'DD/MM/YYYY'.
Masa Mula Mesyuarat	Data	tempahan	masa_mula_mesyuarat	R	Format tarikh adalah dalam bentuk 'DD/MM/YYYY'.
Masa Tamat Mesyuarat	Data	tempahan	masa_tamat_mesyuarat	R	Paparan masa adalah dalam format 12-jam.
Bil. Ahli Mesyuarat	Data	tempahan	bil_ahli_mesyuarat	R	
Ditempah Oleh	Data	tempahan pengguna	no_kad_pengenalan no_kad_pengenalan_nama	R	

Status Tempahan	Data	tempahan status_umum	status_umum_id id nama	C/R/U	
<Simpan>	Butang			C/U	Rekod atau pinda maklumat di dalam jadual yang berkaitan setelah butang Simpan diklik. Keluarkan notifikasi <i>pop-up</i> bagi mengesahkan pengguna ingin melakukan operasi simpan.
<Keluar>	Butang				Batalkan segala operasi dalam menu. Pengguna akan dikembalikan ke muka web/borang yang sebelumnya.

Langkah 6 : Dokumentasikan Antaramuka Pengguna dan Jadual Pemetaan Data

Dokumentasikan semua antaramuka pengguna dan Jadual Pemetaan Data yang telah disediakan ke dalam **D04 Spesifikasi Rekabentuk Sistem**.

RUJUKAN

1. Danny Brian. Attributes of a Great Web User Experience (2013). <https://www.gartner.com/document/2447715>
2. Danny Brian. Building a Great User Experience (2012). <https://www.gartner.com/document/2251816>
3. Ashley Gainer. Tips For Adding White Space To Your Website (2015). <https://getflywheel.com/layout/tips-for-adding-white-space-to-your-website/>
4. Maryam Taheri. 10 Basic Elements of Design (2018). <https://creativemarket.com/blog/10-basic-elements-of-design>
5. Mary Stribley. Design Elements & Principles. <https://www.canva.com/learn/design-elements-principles/>

4.9 REKABENTUK TRANSAKSI SISTEM [F3.5]

KETERANGAN

Rekabentuk Transaksi Sistem merupakan spesifikasi terperinci sesuatu *Use Case*. Ianya juga dipanggil Senario *Use Case*. Senario *Use Case* terdiri daripada langkah-langkah aktiviti yang akan dilakukan dalam sesuatu *Use Case* dengan disokong oleh rekabentuk antaramuka pengguna dan syarat-syarat/kekangan bagi setiap langkah yang dinyatakan. Ia akan dijadikan sebagai panduan kepada pasukan pembangunan untuk membangunkan kod pengaturcaraan yang lebih tersusun bagi mencapai hasil yang diperlukan.

OBJEKTIF

Membangun Senario *Use Case* bagi setiap *Use Case* selari dengan rekabentuk antaramuka pengguna yang telah dibangunkan.

LANGKAH-LANGKAH

Langkah 1 : Penerangan Ringkas *Use Case*

- a) Berdasarkan Rajah *Use Case* yang telah dibangunkan di dalam Pemodelan *Use Case* [F2.1], pilih salah satu Rajah *Use Case* (Modul) sistem.
- b) Ambil salah satu *Use Case* dan terangkan secara ringkas berkaitan *Use Case*
- c) Kenalpasti aktor yang akan berinteraksi dengan *Use Case* yang dipilih
- d) Nyatakan prasyarat atau aktiviti yang perlu dilaksanakan terlebih dahulu sebelum *Use Case* yang terlibat dilaksanakan.

Contoh pra syarat bagi Mohon Tempahan Bilik Mesyuarat adalah:

“Pengguna yang sah berjaya log masuk sistem”.

Langkah 2 : Tentukan Langkah-langkah Aktiviti *Use Case*

- a) Tentukan langkah-langkah aktiviti *Use Case* yang dipilih.
- b) Bagi setiap langkah aktiviti, kenalpasti apakah input (butiran), antaramuka pengguna dan keperluan/syarat-syarat/ kekangan.

Contoh :

Rajah *Use Case* Modul Tempahan Bilik Mesyuarat terdapat 9 *Use Case*, maka sebanyak 9 Senario *Use Case*. Ambil *Use Case* pertama iaitu **UC-BM-MT-TBM-01 Mohon Tempahan Bilik Mesyuarat** dan senaraikan aktiviti yang terlibat dalam memohon tempahan.

Langkah aktiviti Mohon Tempahan Bilik Mesyuarat:

- i) Kemasukan profil pengguna untuk dihubungi
- ii) Pengguna cari kekosongan bilik berdasarkan Tarikh yang diperlukan
- iii) Sistem akan paparkan senarai bilik mesyuarat dan kemudahan yang disediakan
- iv) Pengguna Pilih bilik mesyuarat yang dikehendaki dan tempoh penggunaan
- v) Pengguna hantar permohonan tempahan

Antaramuka Pengguna yang berkaitan:

UI-TBM-01 : Permohonan Tempahan Bilik Mesyuarat

PERMOHONAN TEMPAHAN BILIK MESYUARAT

Butiran pegawai untuk dihubungi:

Nama : Ahmad bin Ali

Emel : ahmad.ali@mampu.gov.my

Carian kekosongan bilik

Tarikh Mula : 31/12/2016 Tarikh Tamat : 31/12/2016 Semak

Senarai Bilik Mesyuarat

	Nama Bilik	Kapasiti	Status
<input checked="" type="checkbox"/>	Bilik Mesyuarat Cyber 1	30 tempat	free
<input type="checkbox"/>	Bilik Bincang Cyber 3	15 tempat	free
<input type="checkbox"/>	Bilik Bincang Cyber Utara 5	15 tempat	free

Hantar

Rajah 72 : Contoh Antaramuka Pengguna Tempahan Bilik Mesyuarat

Langkah 3 : Kenalpasti Pasca Syarat Use Case

Nyatakan pasca syarat selepas semua langkah aktiviti *Use Case* dilaksanakan. Contoh pasca syarat *Use Case* Permohonan Tempahan Bilik Mesyuarat adalah:

“Permohonan tempahan bilik mesyuarat telah dihantar notifikasi kepada pelulus bagi kelulusan”

Langkah 4 : Nyatakan Proses Alternatif Use Case

Proses alternatif merupakan proses pilihan kepada sesuatu langkah aktiviti yang tidak dapat dilaksanakan secara normal. Contohnya, sekiranya tempahan online tidak dapat dilaksanakan maka pengguna boleh dilakukan secara emel dengan menyatakan tarikh, nama bilik mesyuarat.

Langkah 5 : Sedia Dan Lengkapkan Templat Senario Use Case

- a) Lengkapkan dengan terperinci **Apendiks 7 Templat Senario Use Case** bagi setiap aktiviti Use Case yang telah dikenalpasti.
- b) Maklumat-maklumat Senario Use Case adalah seperti berikut :

Jadual 39 : Keterangan Medan-medan Templat Senario Use Case

Medan	Keterangan
Rujukan Use Case	Rujukan bagi setiap aktiviti Use Case berdasarkan konvesyen nama dan nombor yang selaras.
Nama Use Case	Nama bagi aktiviti Use Case yang terlibat berdasarkan Rajah Use Case yang telah dibangunkan.
Keterangan	Keterangan secara ringkas aktiviti Use Case yang terlibat.
Pra Syarat	Syarat atau operasi yang perlu dilaksanakan dahulu sebelum aktiviti yang terlibat dilaksanakan.
Aktor	Aktor yang terlibat dengan aktiviti Use Case berkenaan.
Input	Maklumat atau/dan dokumen yang diperlukan bagi setiap proses di dalam aktiviti Use Case yang terlibat.
Proses	Langkah untuk menavigasi dan melaksanakan operasi berdasarkan rekabentuk antaramuka pengguna yang berkaitan.
Rujukan Rekabentuk Antaramuka	Nama dan nombor rujukan rekabentuk antaramuka pengguna yang terlibat dengan aktiviti Use Case berkenaan.
Keperluan Keterangan / Syarat / Kekangan	Keterangan lanjut atau syarat tambahan atau kekangan yang dihadapi untuk melaksanakan operasi di dalam rekabentuk antaramuka pengguna yang berkaitan.

Pasca Syarat	Syarat atau operasi yang menyusuli selepas aktiviti <i>Use Case</i> berkenaan selesai dilaksanakan.
Proses Alternatif	Proses alternatif sekiranya aktiviti <i>Use Case</i> berkenaan tidak dapat dilakukan.

c) Sila rujuk jadual di bawah bagi contoh pengisian Jadual Senario *Use Case*.

Jadual 40 : Pengisian Templat Senario *Use Case*

Rujukan <i>Use Case</i>	UC-BM-TM-TBM-01		
Nama <i>Use Case</i>	Mohon Tempahan Bilik Mesyuarat		
Keterangan	Transaksi bagi memohon tempahan penggunaan bilik Mesyuarat		
Pra Syarat	Pengguna yang sah berjaya log masuk sistem		
Aktor	Pemohon		
Input	Langkah	Rujukan Antaramuka Pengguna	Keperluan Keterangan / Syarat / Kekangan
Nama dan emel pegawai untuk dihubungi	1. Kemasukan profil pengguna untuk dihubungi	UI-TBM-01	Secara <i>default</i> nama dan emel login yang dipaparkan. Nama & emel yang dikemaskini perlu divalidasi dengan senarai warga agensi. Sekira nama @ emel bukan warga agensi, mesej perlu dipaparkan
Tarikh Mula dan Tarikh Tamat	2. Pengguna cari kekosongan bilik berdasarkan Tarikh yang diperlukan. Masukkan Tarikh mula dan tamat dan tekan semak	UI-TBM-01	Perlu semak: a) Tarikh mula \geq Tarikh sistem semasa b) Tarikh tamat $>$ Tarikh mula
	3. Sistem akan paparkan senarai bilik mesyuarat dan	UI-TBM-01	Sistem akan menyemak semua aset bilik yang berstatus kekosongan dan

	kemudahan yang disediakan		paparkan butiran Nama, kapasiti dan status
	4. Pengguna memilih bilik mesyuarat yang dikehendaki dengan <i>tick</i> pada ruangan yang disediakan	UI-TBM-01	Pengguna boleh memilih lebih daripada satu bilik untuk sesuatu tempahan.
	5. Pengguna hantar permohonan tempahan	UI-TBM-01	Sistem perlu papar mesej sekiranya hantar permohonan tanpa tick pada senarai bilik.
Pasca Syarat	Sistem akan menghantar notifikasi kelulusan permohonan tempahan kepada pelulus.		
Proses Alternatif	Pengguna dapat mengemelkan tarikh dan kapasiti bilik yang diperlukan kepada pentadbir untuk tempahan.		

Langkah 4 : Dokumentasikan Senario Use Case

Dokumentasikan semua senario *Use Case* yang telah disediakan ke dalam **D04 Spesifikasi Rekabentuk Sistem**.

RUJUKAN

1. Dokumen Spesifikasi Rekabentuk Sistem (SDS) ePPAx
2. Slaid Bootcamp Pembangunan Sistem Pasukan Perunding ICT MAMPU

4.10 PENYEDIAAN SPESIFIKASI REKABENTUK SISTEM [F3.6]

KETERANGAN

Spesifikasi Rekabentuk Sistem (SDS) adalah penerangan terperinci berkenaan rekabentuk-rekabentuk arkitektur, fungsi sistem, pangkalan data, migrasi data dan integrasi data bagi sistem aplikasi yang akan dibangunkan. Dokumen SDS merupakan dokumen yang disediakan sebagai panduan utama kepada pasukan pengaturcara kepada senibina dan rekabentuk satu-satu sistem aplikasi.

OBJEKTIF

- Menyediakan Rekabentuk Arkitektur Keseluruhan Sistem Aplikasi, Arkitektur Aplikasi dan Arkitektur Pangkalan Data berdasarkan kepada langkah-langkah yang terkandung di dalam Rekabentuk Arkitektur [F3.1]
- Mengemaskini Model Fungsi Sistem yang telah dibangunkan di dalam Pemodelan Fungsi Sistem [F2.2] supaya selaras dengan keperluan rekabentuk sistem.
- Menyediakan Rekabentuk Transaksi Sistem, Rekabentuk Antaramuka Pengguna dan Pemetaan Data seperti yang dinyatakan dalam langkah-langkah di bawah Rekabentuk Transaksi Sistem [F3.5] dan Rekabentuk Antaramuka Sistem [F3.5].
- Menyediakan Rekabentuk serta Skema Pangkalan Data Logikal dengan merujuk kepada langkah-langkah di bawah Rekabentuk Pangkalan Data [F3.3].
- Menyertakan keterangan ringkas berkenaan Rekabentuk Migrasi Data dan Rekabentuk Integrasi Data.

LANGKAH-LANGKAH

Langkah 1 : Sediakan Pengenalan Kepada Rekabentuk Sistem

Sila sedia dan lengkapkan pengenalan bisnes bagi perkara-perkara berikut:

a) Tujuan Rekabentuk

Terangkan tujuan, objektif dan matlamat yang ingin dicapai di dalam rekabentuk sistem aplikasi selaras dengan objektif bisnes dan keperluan sistem yang ingin dipenuhi.

b) Skop Rekabentuk

Nyatakan skop rekabentuk sistem aplikasi berdasarkan skop keperluan bisnes dan keperluan sistem yang telah diperolehi di dalam dokumen BRS dan SRS. Skop rekabentuk merupakan penentuan sempadan kepada rekabentuk fungsi sistem dan pangkalan data yang disediakan. Sempadan rekabentuk ini boleh ditentukan dengan merujuk kepada bilangan modul, menu dan submenu yang akan dirangkumkan di dalam sistem aplikasi.

Langkah 2 : Sediakan Rekabentuk Arkitektur Sistem Aplikasi

Rekabentuk Arkitektur

a) Arkitektur Keseluruhan Sistem Aplikasi

Sediakan Arkitektur Keseluruhan Sistem Aplikasi yang merupakan gambaran menyeluruh (*bird's eye view*) bagi komponen-komponen antaramuka sistem, aplikasi dan pangkalan data. Sila rujuk kepada langkah 4 di dalam Rekabentuk Arkitektur [F3.1] untuk menyediakan arkitektur berkenaan.

b) Arkitektur Aplikasi

Sediakan Arkitektur Aplikasi yang merupakan gambaran komponen-komponen aplikasi terperinci yang terkandung di dalam Arkitektur Keseluruhan Sistem Aplikasi. Sila rujuk kepada langkah 5 di dalam Rekabentuk Arkitektur [F3.1] untuk menyediakan arkitektur berkenaan.

c) Arkitektur Pangkalan Data

Sediakan Arkitektur Pangkalan Data yang merupakan gambaran komponen-komponen pangkalan data terperinci yang terkandung di dalam Arkitektur Keseluruhan Sistem Aplikasi. Sila rujuk kepada langkah 6 di dalam **Rekabentuk Arkitektur [F3.1]** untuk menyediakan arkitektur berkenaan.

Langkah 3 : Kemaskini Model Fungsi Sistem

a) Penggunaan Notasi

Senaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi Sistem. Rujuk kepada Pemodelan Fungsi Sistem [F2.2] untuk menyediakan senarai notasi berkenaan.

b) Rajah Hierarki Fungsian Sistem

Rajah Hierarki Fungsian Sistem yang telah disediakan di dalam Pemodelan Fungsi Bisnes [F2.2] dan dipaparkan di dalam dokumen D03 Spesifikasi Keperluan Sistem perlu dikemaskini dan dibangunkan semula sekiranya terdapat perubahan dari segi struktur Fungsi Sistem dalam fasa rekabentuk ini. Namun begitu sekiranya tiada sebarang perubahan kepada struktur berkenaan, Rajah Hierarki Fungsian Sistem yang sama boleh diguna dan direkodkan di dalam dokumen SDS.

c) Jadual Pemandanan Aktor Dengan Fungsi Sistem

Kemaskini Jadual Pemandanan Aktor dengan Fungsi Sistem yang telah dibangunkan dalam **Pemodelan Fungsi Bisnes [F2.2]** sekiranya terdapat sebarang perubahan kepada struktur Fungsi Sistem. Sekiranya tiada sebarang perubahan, paparkan sahaja jadual yang sama yang telah dibangunkan dalam fasa sebelum ini. Jadual pemandanan

ini akan digunakan sebagai sumber rujukan bagi kawalan akses pengguna dengan sistem aplikasi yang akan dibangunkan.

Langkah 4 : Dokumentasikan Rekabentuk Fungsian

a) Rekabentuk Antaramuka Pengguna dan Pemetaan Data

Sertakan imej-imej Rekabentuk Antaramuka Pengguna bagi setiap skrin di bawah fungsi, modul, menu dan submenu aplikasi. Rekabentuk Antaramuka Pengguna yang telah disediakan kemudiannya akan dipadankan dengan entiti (jadual) dan atribut yang terkandung di dalam pangkalan data. Sila rujuk kepada langkah-langkah di bawah Rekabentuk Antaramuka [F3.4] bagi penyediaan rekabentuk tersebut dan pemetaan kepada data-data yang terlibat.

b) Rekabentuk Transaksi Sistem

Sertakan Rekabentuk Transaksi Sistem yang terdiri dari jadual-jadual Senario *Use Case*. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Transaksi Sistem [F3.5]** bagi penyediaan rekabentuk berkenaan.

Langkah 5 : Dokumentasikan Rekabentuk Pangkalan Data

a) Rekabentuk Pangkalan Data

Sertakan Rekabentuk Pangkalan Data Logikal yang merupakan perincian lanjut kepada Rajah Hubungan Entiti (ERD) yang telah disediakan di dalam dokumen SRS. Sila rujuk kepada langkah-langkah di dalam Rekabentuk Pangkalan Data [F3.3] bagi penyediaan rekabentuk berkenaan.

b) Skema Logikal Pangkalan Data

Sertakan juga Skema Pangkalan Data Logikal berdasarkan kepada langkah-langkah yang telah dinyatakan di dalam Rekabentuk Pangkalan Data [F3.3].

Langkah 6 : Nyatakan Secara Ringkas Rekabentuk Migrasi Dan Integrasi Data

a) Rekabentuk Migrasi Data

Sediakan keterangan ringkas berkenaan Rekabentuk Migrasi Data dan nyatakan juga rujukan kepada dokumen [D05] Pelan Migrasi Data dan [D06] Spesifikasi Migrasi Data bagi penerangan lanjut kepada rekabentuk berkenaan.

b) Rekabentuk Integrasi Data

Sediakan keterangan ringkas berkenaan Rekabentuk Integrasi Data dan nyatakan juga rujukan kepada dokumen [D06] Pelan Integrasi Data dan [D07] Spesifikasi Integrasi Data bagi penerangan lanjut kepada rekabentuk berkenaan.

Langkah 7 : Sertakan Dokumen-dokumen Sokongan Sebagai Lampiran

Sertakan dokumen-dokumen sokongan, sekiranya ada, yang perlu dirujuk sebagai penerangan lanjut kepada rekabentuk-rekabentuk yang disertakan di dalam dokumen SDS.

Langkah 8 : Lakukan Semakan Dan Pengesahan Ke Atas Dokumen SDS

Dokumen SDS perlu dilakukan semakan oleh Ketua Pasukan Analisis dan Rekabentuk, atau pegawai-pegawai yang lain yang bersesuaian. Setelah semakan dilakukan, dokumen SDS yang telah disediakan perlu disahkan oleh Pengurus Projek atau Pengarah Bahagian atau pegawai-pegawai yang lain yang bersesuaian.

RUJUKAN

1. ISO/IEC/IEEE 29148-2011 Systems and software engineering — Life cycle processes — Requirements engineering (2011)
2. IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications (1998)
3. IEEE 1233-1998 - IEEE Guide for Developing System Requirements Specifications (1998)
4. IEEE Std 1016-2009 : IEEE Standard for Information Technology—Systems Design—Software Design Descriptions (2009)

4.11 MIGRASI DATA

Migrasi data merupakan proses memindahkan data daripada satu sumber asal (sama ada daripada sumber sistem legasi, format excel, *hardcopies* atau sebagainya) ke destinasi baharu yang berbeza daripada sumber asal.

Proses migrasi ini biasanya dijalankan atas sebab-sebab seperti berikut:

- a) Pengembangan skop bisnes yang memerlukan pembangunan/peningkatan sistem;
- b) Penggabungan beberapa sistem kepada satu sistem;
- c) Penaiktarafan perkakasan dan perisian; dan
- d) Proses normalisasi/ semakan semula pangkalan data.

Objektif migrasi data adalah untuk memastikan data daripada sistem legasi dapat digunakan dalam sistem baharu selaras dengan keperluan bisnes.

Dua (2) aktiviti utama dalam migrasi data iaitu:

- a) Penyediaan Pelan Migrasi Data; dan
- b) Rekabentuk Migrasi Data.

4.11.1 PENYEDIAAN PELAN MIGRASI DATA [F3.7]

KETERANGAN

Keperluan migrasi data dapat dikenal pasti semasa proses mengenal pasti proses bisnes sebagaimana diterangkan dalam **Pemodelan Proses Bisnes [F1.4]**. Analisis keperluan migrasi data perlu dijalankan bagi menentukan skop kerja proses migrasi data, impak migrasi kepada bisnes dan mengenal pasti risiko serta isu-isu berkaitan proses migrasi data.

Pelan Migrasi Data akan dibangunkan sebagai panduan dan rujukan bagi keseluruhan pelaksanaan migrasi data hasil daripada analisis keperluan data. Pelan ini menggariskan strategi, kaedah dan jadual pelaksanaan migrasi data yang perlu dipatuhi semasa pelaksanaan migrasi data. Penyediaan pelan ini dapat melancarkan pelaksanaan migrasi data.

OBJEKTIF

Menjalankan analisis keperluan data bagi menghasilkan Pelan Migrasi Data sebagai sumber rujukan yang menetapkan strategi, kaedah dan jadual pelaksanaan yang akan digunakan dalam pelaksanaan migrasi data.

LANGKAH-LANGKAH

Langkah 1 : Analisis Keperluan

Analisis keperluan migrasi akan melibatkan kajian dan perbincangan antara pemilik proses, pembangun sistem dan pasukan migrasi data. Proses ini dijalankan bagi menentukan perkara-perkara seperti berikut:

Jadual 41 : Analisa Keperluan Migrasi Data

Bil.	Perkara	Penerangan
1	Apakah data yang ingin dipindahkan?	Kategori data yang ingin dipindahkan sebagai contoh, data maklumat peribadi pengguna bagi Sistem Tempahan Bilik Mesyuarat.
2	Apakah sistem yang terlibat?	Nyatakan sistem yang terlibat. Pemindahan data melibatkan data daripada sistem ke sistem atau manual ke sistem.
3	Bilakah data diperlukan?	Nyatakan bilakah data diperlukan dan maklumat ini perlu diambil kira dalam penyediaan jadual pelaksanaan.

4	Apakah impak migrasi kepada bisnes?	Impak sekiranya migrasi tidak dilaksanakan atau tidak dijalankan mengikut jadual kepada bisnes agensi. Adakah akan mengganggu aliran proses bisnes?
5	Apakah risiko proses migrasi ini?	Risiko dalam pelaksanaan migrasi sebagai contoh, sistem legasi perlu menjalani <i>downtime</i> untuk proses migrasi. Data perlu dipindahkan pada masa yang bersesuaian supaya tidak mengganggu aktiviti proses bisnes yang telah dirancang.
6	Apakah jenis pangkalan data yang digunakan di sistem legasi?	Maklumat persekitaran teknologi yang digunakan seperti jenis pangkalan data dan jenis data yang terdapat pada sistem legasi diperlukan sebagai input kepada strategi dan kaedah migrasi.
7	Apakah strategi dan kaedah yang bersesuaian digunakan untuk migrasi?	Strategi pelaksanaan migrasi data dilaksanakan sama ada secara <i>one-off</i> atau berfasa, maklumat penggunaan <i>tools</i> , secara <i>scripting</i> atau lain-lain kaedah dikenal pasti berdasarkan keperluan bisnes dan persekitaran semasa.
8	Siapa yang terlibat dalam proses migrasi data?	Pasukan migrasi yang akan terlibat dan peranan setiap ahli.

Langkah 2 : Bangunkan Pelan Migrasi Data

Pelan Migrasi Data dihasilkan selepas analisis keperluan migrasi data selesai dijalankan dan mengandungi **sekurang-kurangnya** perkara seperti berikut :

Jadual 42 : Isi Kandungan Pelan Migrasi Data

Tajuk	Isi Kandungan
Tujuan Dokumen	Penerangan tujuan dokumen dihasilkan adalah untuk dijadikan panduan bagi pelaksanaan migrasi data bagi sistem legasi kepada sistem baharu. Nama sistem perlu dinyatakan dengan jelas.
Latar Belakang	Pengenalan ringkas sistem iaitu objektif dan fungsi sistem serta impak pelaksanaan migrasi data kepada bisnes.

Objektif Migrasi	Penerangan objektif migrasi data dilaksanakan.
Skop Migrasi	<p>Penerangan sistem dan modul yang terlibat dan skop data yang terlibat.</p> <ul style="list-style-type: none"> a) Nama sistem terlibat – contoh : data daripada Sistem Sumber Manusia akan dipindahkan ke Sistem Tempahan Bilik Mesyuarat b) Modul yang terlibat – contoh : data bagi Modul Pengurusan Pengguna sahaja c) Julat data yang terlibat – contoh : data daripada tahun 2010 hingga 2017 sahaja atau data pegawai gred 41 dan ke atas sahaja.
Pendekatan Migrasi Data	<p>Penjelasan tentang pendekatan pelaksanaan migrasi data sama ada berfasa mengikut modul atau sebaliknya. Kaedah migrasi dijalankan iaitu sama ada menggunakan <i>tools</i>, <i>scripting</i> atau lain-lain kaedah juga diterangkan dalam bab ini.</p> <p>Bab ini juga mengandungi maklumat sekurang-kurangnya seperti berikut :</p> <ul style="list-style-type: none"> a) Penambahbaikan atau penyediaan <i>Service Level Agreement</i> (SLA) sekiranya perlu; b) Persediaan perkakasan (<i>hardware</i>) dan perisian (<i>software</i>) diperlukan; c) Aspek teknologi mengenai persekitaran semasa dan persekitaran migrasi yang perlu diambil kira; dan d) Kaedah pengujian dan verifikasi data yang akan digunakan
Pasukan Projek	Struktur organisasi bagi pasukan projek dan menyertakan nama-nama pegawai terlibat.
Jadual Pelaksanaan	<p>Penerangan jadual pelaksanaan migrasi data yang dicadangkan dalam bentuk gantt chart atau yang bersesuaian. Ia mengandungi aktiviti migrasi bermula daripada analisis keperluan sehingga penyediaan laporan. Jadual pelaksanaan juga perlu mengambil kira aspek seperti berikut :</p> <ul style="list-style-type: none"> a) Susunan keutamaan data yang ingin dipindahkan; dan b) Tempoh <i>downtime</i> yang dibenarkan untuk sistem semasa sekiranya melibatkan sistem yang sedang beroperasi.

Pelaksanaan migrasi data akan dilaksanakan berdasarkan pelan yang dihasilkan. Rujuk format **D05 Pelan Migrasi Data**.

Langkah 3 : Sahkan Pelan Migrasi Data

Pelan Migrasi Data yang didokumenkan perlu dibentang dan mendapat pengesahan pemilik sistem bagi memastikan strategi, kaedah dan jadual pelaksanaan migrasi data memenuhi keperluan dan mendapat sokongan serta kerjasama daripada pemilik sistem dan pembangun migrasi data.

RUJUKAN

1. Pelan Migrasi Data Sistem eRoses
2. Oracle White Paper (2011). Successful Data Migration. <http://www.oracle.com/technetwork/middleware/oedq/successful-data-migration-wp-1555708.pdf>
3. Credosoft White Paper. Eight key steps which help ensure a successful data migration project: A white paper for inspection management professionals. <http://credosoft.com/wp/wp-content/uploads/2014/01/Eight-key-steps-which-help-ensure-a-successfu-data-migration-project.pdf>
4. SAGA Group (2012). Methods of Data Migration.

4.11.2 REKABENTUK MIGRASI DATA [F3.8]

KETERANGAN

Proses reka bentuk dan pemetaan akan dilakukan selepas selesai pembangunan pelan migrasi. Proses ini terbahagi kepada empat langkah utama iaitu :

- a) Pemetaan jadual (*table*);
- b) Pemetaan medan data (*field*);
- c) Pemetaan kod; dan
- d) Pemetaan rekod (data) disebabkan perubahan kod/id.

Proses ini boleh dilakukan secara manual atau menggunakan *tools*. Antara *tools* yang boleh digunakan dalam rekabentuk migrasi ialah Altova MapForce, Talend dan Navicat.

OBJEKTIF

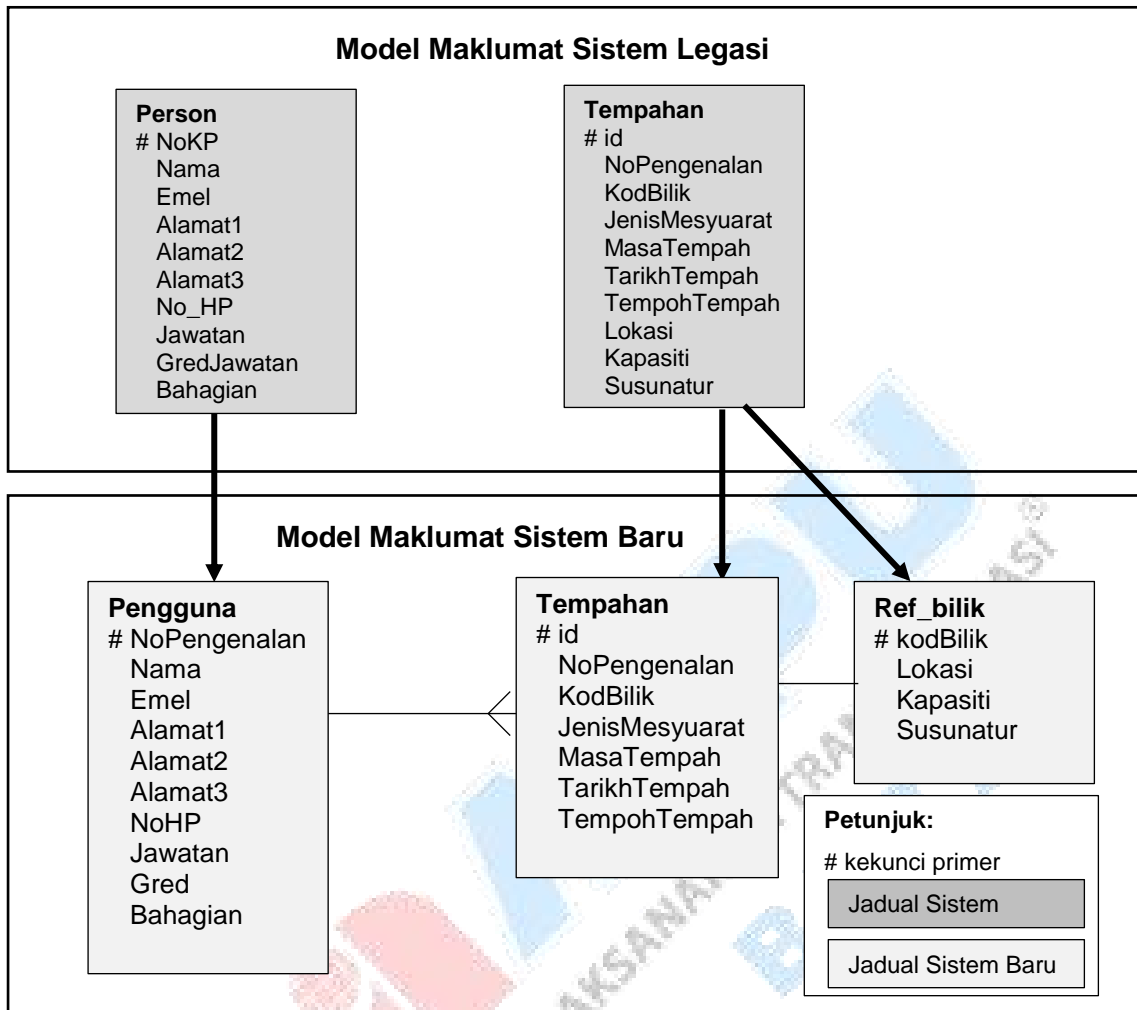
Proses reka bentuk dan pemetaan akan memastikan data yang dipindahkan dipetakan dari sumber asal ke destinasi baharu.

LANGKAH-LANGKAH

Langkah 1 : Pemetaan Jadual

Langkah pertama ini akan mengenal pasti bagaimana data daripada pangkalan data sistem legasi dipetakan kepada pangkalan data sistem baharu. Proses ini melibatkan pengenalpastian jadual yang terlibat pada kedua-dua pangkalan data sistem legasi dan sistem baharu. Sumber rujukan bagi proses ini adalah Model Maklumat Logikal bagi pangkalan data sistem yang terlibat.

Contoh pengenalpastian dan pemetaan jadual yang dijalankan dalam proses migrasi adalah seperti berikut.



Rajah 73 : Contoh Pemetaan Jadual antara Pangkalan Data Sistem Legasi dan Sistem Baharu

Langkah 2 : Pemetaan Medan Data

- Kenal pasti elemen data berdasarkan kepada Skema Logikal Pangkalan Data bagi sistem legasi serta sistem baharu.
- Bangunkan peraturan bisnes dalam mengendalikan kesemua proses ini. Sebagai contoh:
 - Alamat lokasi aset disimpan dalam 3 baris berlainan pada pangkalan data sumber iaitu Address1, Address2 dan Address3. Tentukan sama ada alamat lokasi aset di destinasi baharu akan mengikut format sumber atau disimpan dalam 1 baris sahaja pada *field* "Alamat".
 - Tinggi dan lebar aset direkodkan dalam format sentimeter pada pangkalan data sistem baharu. Oleh yang demikian, data dalam format inci pada pangkalan data sistem legasi perlu ditukar kepada sentimeter.

- iii) Kod aset pada pangkalan data sistem baharu akan diselaraskan menggunakan kod standard pada *Data Dictionary* Sektor Awam (DDSA).

c) **Lengkapkan** Apendiks 8a Templat Pemetaan Data Migrasi.

Contoh pengisian pemetaan data :

Jadual 43 : Pemetaan Medan Data antara Sistem Legasi dan Sistem Baharu

Bil.	MAKLUMAT PANGKALAN DATA SUMBER						MAKLUMAT PANGKALAN DATA DESTINASI						Catatan
	Penyedia (SISTEM SUMBER MANUSIA)						Penerima (SISTEM TEMPAHAN BILIK MESYUARAT)						
	Jadual	Medan	Jenis	P/F	Mandatori/Tidak	DDSA	Jadual	Medan	Keterangan	Jenis	P/F	Mandatori/Tidak	
1	PERSON	EPD_NAMA	Varchar (100)		Mandatori		PENGGUNA	nama	Nama pegawai	Varchar (100)		Mandatori	Tidak
2	PERSON	EPD_NOKP	Varchar (16)	P	Mandatori	Ya	PENGGUNA	no_pengenalan	No. Kad Pengenalan	Varchar (16)	P	Mandatori	Tidak
3	PERSON	EMEL	Varchar (50)		Tidak		PENGGUNA	emel	Alamat emel pegawai	Varchar (50)		Tidak	Tidak

Nota:

- i) P – Kekunci Primer
 ii) F – Kekunci Asing
 iii) DDSA – *Data Dictionary* Sektor Awam

Langkah 3 : Sediakan Keperluan Pemetaan Kod

Proses pemetaan kod dijalankan sekiranya ada keperluan untuk menyesuaikan dan menyelaraskan kod (atau ID) bagi atribut daripada sumber data dengan destinasi baharu data. Proses ini biasanya dijalankan ke atas jadual rujukan.

Jadual 44 : Contoh Pemetaan Kod

Kod Data Asal (sumber asal)	Penerangan data	Kod Data Baharu (destinasi baharu)
MJKP	Mesyuarat Jawatankuasa Pemandu	M01
MJKT	Mesyuarat Jawatankuasa Teknikal	M02
MBHGN	Mesyuarat Bahagian	M03

Aktiviti yang terlibat dalam proses keperluan pemetaan kod adalah seperti berikut.

- a) Kenal pasti kod yang berubah.

Kod yang berubah adalah seperti berikut:

MJKP -> M01
 MJKT -> M02
 MBHGN -> M03

- b) Kenal pasti jadual dan medan yang menyimpan maklumat kod tersebut. Contoh jadual yang dikenalpasti adalah jadual sumber asal: ref_Mesyuarat dan jadual destinasi baru: ruj_JenisMesyuarat (jadual baharu) seperti jadual dibawah.

Jadual 45 : Contoh jadual yang menyimpan maklumat kod

Jadual dan Medan Asal (Sumber Asal)	Jadual dan Medan Baru (Destinasi Baru)
Jadual: ref_mesyuarat	Jadual: ruj_jenisMesyuarat
Medan: kod, nama	Medan: kod, keterangan

Langkah 4 : Sediakan Keperluan Pemetaan Rekod (Data) Disebabkan Perubahan Kod/ID

- a) Adakalanya perubahan kod yang berlaku menyebabkan perubahan ke atas data dalam rekod maklumat. Perubahan yang berlaku ke atas data memerlukan pemetaan data berdasarkan perubahan kod (atau ID). Proses pemetaan data dijalankan bagi menyesuaikan dan menyelaraskan rekod data berdasarkan kod data baharu. Contoh bagi keperluan pemetaan rekod (data) disebabkan perubahan kod/ID adalah seperti berikut.

Dalam sistem lama, rekod skim dan perjawatan adalah dalam jadual sumber asal iaitu Jadual : ref_skimJwtn seperti dibawah.

Jadual 46: jadual sumber asal ref_skimJwtn

Kod	Keterangan
ICT1	Skim jawatan ICT untuk gred 1
ICT2	Skim jawatan ICT untuk gred 2
ICT3	Skim jawatan ICT untuk gred 3

Manakala dalam sistem baru pula, terdapat perubahan dalam kod skim jawatan ICT. Rekod skim dan gred perjawatan yang baru (dalam sistem baru) adalah dalam jadual destinasi baru iaitu Jadual : ref_skimGredJwtn seperti dibawah.

Jadual 47: Jadual destinasi baru ref_skimGredJwtn

Kod	Keterangan
X41	Skim jawatan ICT untuk gred 41
X44	Skim jawatan ICT untuk gred 44
X48	Skim jawatan ICT untuk gred 48
X52	Skim jawatan ICT untuk gred 52
X54	Skim jawatan ICT untuk gred 54
XX60	Skim jawatan ICT untuk gred 60
XX61	Skim jawatan ICT untuk gred 61
XX62	Skim jawatan ICT untuk gred 62

Kedua-dua jadual di atas adalah contoh jadual yang menyimpan rekod skim dan perjawatan bagi sumber asal dan jadual destinasi baru yang telah dikenalpasti.

Berikut adalah sebahagian daripada medan yang terdapat dalam jadual sumber asal Jadual : pekerja. Jadual ini menunjukkan rekod pekerja yang mengandungi maklumat rekod peribadi iaitu termasuklah skim dan gred jawatan serta gaji. Medan 'gred' menggunakan gred skim perjawatan yang lama. Jadual dibawah adalah contoh jadual sumber asal iaitu Jadual : pekerja yang telah dikenalpasti.

Jadual 48: Jadual sumber asal 'pekerja'

idPekerja	nama	tarikh_mula_bekerja	gaji	gred
0000001	Ali bin Abu	01 April 1985	12070.00	ICT3
0000002	Aminah binti Aziz	15 Januari 1988	10800.00	ICT3
0000003	Latifah binti Mazlan	01 Mac 2005	6500.00	ICT2
0000004	Suriya binti Abdullah	01 Mei 2010	3500.00	ICT1
0000005	Hassan bin Abu	01 Jun 2017	2100.00	ICT1

Jadual berikut adalah maklumat yang mengandungi perubahan gred perjawatan bagi skim ICT yang terlibat berdasarkan gaji dan tempoh perkhidmatan.

Jadual 49: Contoh Maklumat Perubahan Gred Perjawatan

Skim/Gred Lama (Sumber Asal)	Tempoh Perkhidmatan	Gaji (RM)	Skim/Gred Baru (Sumber Destinasi)
ICT1	Kurang 5 tahun	1,500.00 hingga 3,000.00	X41
ICT1	5 hingga 8 tahun	3,000.01 hingga 5,000.00	X44
ICT2	8 hingga 12 tahun	5,000.01 hingga 7,000.00	X48
ICT2	12 hingga 15 tahun	7,000.01 hingga 9,000.00	X52
ICT2	15 hingga 20 tahun	9,000.01 hingga 12,000.00	X54
ICT3	20 hingga 25 tahun	Lebih 12,000.00	XX60
ICT3	25 hingga 30 tahun	Lebih 12,000.00	XX61
ICT3	30 hingga 35 tahun	Lebih 12,000.00	XX62

b) Berdasarkan contoh di atas, berikut adalah langkah-langkah yang perlu diambil bagi melaksanakan pemetaan rekod (data) disebabkan perubahan kod/ID:

- i) Kenal pasti data atau rekod yang perlu dikemaskini disebabkan perubahan kod tersebut. Data atau rekod yang perlu dikemaskini disebabkan perubahan kod adalah data gred bagi setiap pekerja. Dalam pangkalan data sumber asal, rekod pekerja disimpan dalam jadual 'pekerja' dan medan yang akan mengalami perubahan kod adalah medan 'gred'.

Kenalpasti jadual dan medan yang terlibat dalam proses pemetaan data. Jadual yang terlibat dalam proses pemetaan data adalah jadual sumber asal iaitu Jadual: ref_skimJwtn dan Jadual: pekerja seperti yang telah dinyatakan di atas.

- ii) Sediakan peraturan pemetaan data bagi rekod yang terlibat seperti dibawah. Contoh pemetaan bagi kod baru (medan 'gred') untuk rekod kakitangan di jadual destinasi baru iaitu Jadual : PERSONEL. Peraturan yang digunakan adalah seperti yang terdapat dalam jadual perubahan gred perjawatan bagi skim ICT berdasarkan gaji dan tempoh perkhidmatan. Berdasarkan rajah diatas, peraturan yang terlibat dalam jadual pekerja adalah dalam medan 'tarikh_mula_bekerja' dan medan 'gaji'.

Contoh pengiraan bagi formula pemetaan data (andaian *current* tarikh adalah 15 Oktober 2018) bagi idPekerja '000001' adalah seperti berikut.

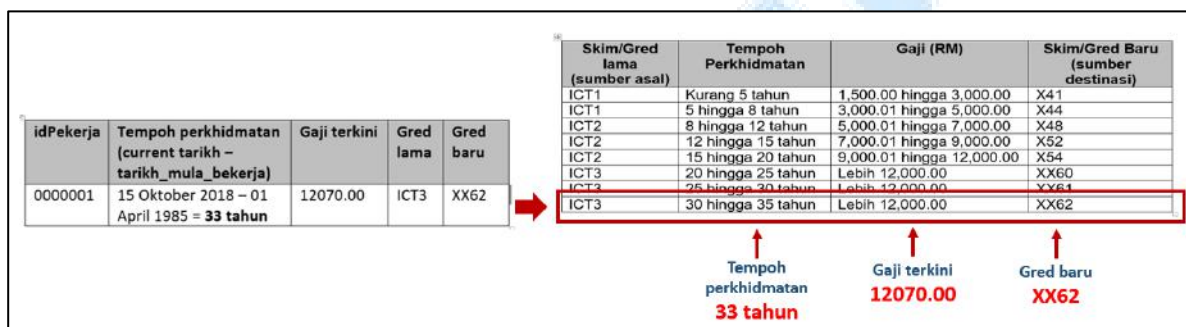
$$\begin{aligned}
 \text{Tempoh perkhidmatan} &= \text{current tarikh} - \text{tarikh_mula_bekerja} \\
 &= 15 \text{ Oktober } 2018 - 01 \text{ April } 1985 \\
 &= \mathbf{33 \text{ tahun}}
 \end{aligned}$$

Contoh Jadual bagi idPekerja '0000001' adalah seperti berikut.

Jadual 50: Contoh Jadual Yang Menunjukkan Rekod idPekerja '0000001'

idPekerja	Tempoh perkhidmatan (current tarikh – tarikh_mula_bekerja)	Gaji terkini	Gred lama
0000001	15 Oktober 2018 – 01 April 1985 = 33 tahun	12070.00	ICT3

Contoh pemetaan data bagi idPekerja '0000001' berdasarkan jadual Pekerja dan maklumat perubahan gred perjawatan adalah seperti rajah berikut.



Rajah 74 : Contoh pemetaan data

Pemetaan bagi rekod yang menunjukkan maklumat gred lama dan gred baru adalah seperti jadual di bawah.

Jadual 51 : Contoh Pemetaan Rekod Yang Menunjukkan Maklumat Perubahan Gred

idPekerja	Tempoh perkhidmatan (current tarikh – tarikh_mula_bekerja)	Gaji terkini	Gred lama	Gred baru
0000001	15 Oktober 2018 – 01 April 1985 = 33 tahun	12070.00	ICT3	XX62
0000002	15 Oktober 2018 – 15 Januari 1990 = 28 tahun	10800.00	ICT3	X54
0000003	15 Oktober 2018 – 01 Mac 2005 = 13 tahun	6500.00	ICT2	X48
0000004	15 Oktober 2018 – 01 Mei 2009 = 9 tahun	3500.00	ICT1	X44
0000005	15 Oktober 2018 – 01 Jun 2017 = 1 tahun	2100.00	ICT1	X41

Dalam pangkalan data destinasi, rekod pekerja disimpan dalam jadual pesonel dan medan yang perlu dikemaskini ialah 'gred'. Jadual dibawah (jadual: pesonel) adalah contoh rekod pesonel bagi jadual pesonel di destinasi baru. Medan 'gred'

menggunakan skim gred perjawatan yang baru. Formula perubahan adalah berdasarkan jadual perubahan gred perjawatan bagi skim ICT berdasarkan gaji dan tempoh perkhidmatan seperti di atas. Contoh data yang terlibat dalam perubahan dalam kod skim jawatan ICT adalah di jadual destinasi baru Jadual : Personel seperti jadual di bawah.

Jadual 52 : Contoh Jadual Destinasi Baru Pesonel

idPesonel	namaPesonel	tarikhMulaKhidmat	gaji	gred
0000001	Ali bin Abu	01 April 1985	12070.00	XX62
0000002	Aminah binti Aziz	15 Januari 1990	10800.00	X54
0000003	Latifah binti Mazlan	01 Mac 2005	6500.00	X48
0000004	Suriya binti Abdullah	01 Mei 2009	3500.00	X44
0000005	Hassan bin Abu	01 Jun 2017	2100.00	X41

- c) Lengkapkan Apendiks 8b) Templat Peraturan Pemetaan Data bagi rekod yang terlibat. Contoh pengisian peraturan pemetaan data seperti jadual dibawah.

Jadual 53 : Contoh Pengisian Peraturan Pemetaan Data

Kod Data Asal (Sumber Asal)	Keterangan Data	Peraturan /Nota	Kod Data Baharu (Destinasi Baru)
ICT1	Kod Skim	tempohPerkhidmatan = currentdate – PERSONEL.tarikhMulaKhidmat; gred = X41 If PERSONEL.kodGredSkim = ICT1 && PERSONEL.gaji is between 1500.00 and 3000.00 && tempohPerkhidmatan < 5 years;	X41
ICT1	Kod Skim	tempohPerkhidmatan = currentdate – PERSONEL.tarikhMulaKhidmat; gred = X44 If PERSONEL.kodGredSkim = ICT1 && PERSONEL.gaji is between 3500.01 and 5000.00;	X44

Langkah 5 : Sediakan Spesifikasi Migrasi Data

- a) Spesifikasi Migrasi Data perlu dibangunkan dan dijadikan rujukan dalam pelaksanaan migrasi data bersama-sama Pelan Migrasi Data. Spesifikasi ini akan mendokumentasikan langkah 1 hingga 3 yang telah diterangkan di atas iaitu mengandungi perkara seperti berikut:

Jadual 54 : Isi Kandungan Spesifikasi Migrasi Data

Tajuk	Isi Kandungan
Tujuan Dokumen	Penerangan tujuan dokumen dihasilkan adalah untuk merekodkan maklumat bagi rekabentuk migrasi data bagi tujuan memindahkan data daripada pangkalan data sumber ke destinasi pangkalan data baharu.
Maklumat Sistem	Maklumat sistem dan pangkalan data yang terlibat bagi tujuan migrasi data daripada sumber asal ke destinasi baharu direkodkan. Ini bertujuan bagi memudahkan penyelarasan dan konfigurasi sistem untuk tujuan migrasi data dijalankan.
Pemetaan Jadual	Penerangan dan rajah pemetaan jadual yang terlibat pada kedua-dua pangkalan data sistem legasi dan sistem baharu.
Peraturan Bisnes	Senarai peraturan bisnes yang perlu dipatuhi dalam proses migrasi data yang telah ditetapkan untuk menyelaraskan format data yang dipindahkan.
Pemetaan Data	Penerangan dan jadual yang mengandungi perincian data yang perlu dipindahkan perlu dikenal pasti berdasarkan kepada Skema Logikal Pangkalan Data bagi sistem legasi serta sistem baharu.
Pemetaan Kod	Penerangan dan jadual yang mengandungi perincian kod bagi atribut yang perlu diselaraskan antara pangkalan data sistem legasi dan sistem baharu.

- b) Rujuk format dokumen D06 Spesifikasi Migrasi Data. Dokumen ini juga boleh dijadikan sebagai lampiran dalam D05 Pelan Migrasi Data.
- c) Langkah seterusnya iaitu proses Pelaksanaan Migrasi Data [F6.1] yang mana terdiri daripada aktiviti Pelaksanaan, Pengujian dan Penyediaan Laporan Migrasi data.

Langkah 6 : Sahkan Spesifikasi Migrasi Data

D06 Spesifikasi Migrasi Data yang didokumenkan perlu dibentang dan mendapat pengesahan pemilik sistem bagi memastikan kesahihan dan spesifikasi yang dihasilkan memenuhi keperluan migrasi data.

RUJUKAN

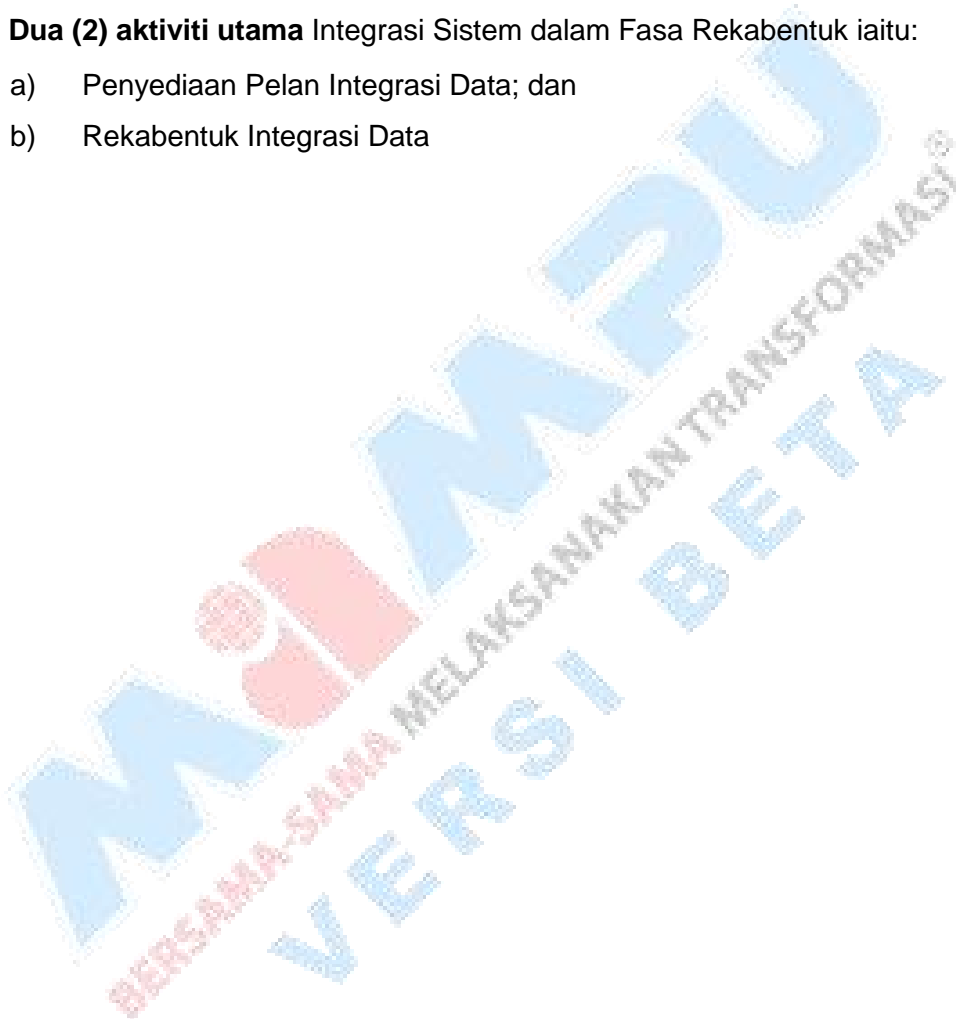
1. Pelan Migrasi Data Sistem eRoses
2. Oracle White Paper (2011). Successful Data Migration. <http://www.oracle.com/technetwork/middleware/oedq/successful-data-migration-wp-1555708.pdf>
3. Credosoft White Paper. Eight key steps which help ensure a successful data migration project: A white paper for inspection management professionals. <http://credosoft.com/wp/wp-content/uploads/2014/01/Eight-key-steps-which-help-ensure-a-successfu-data-migration-project.pdf>
4. SAGA Group (2012). Methods of Data Migration.

4.12 INTEGRASI SISTEM

Terdapat pelbagai sistem aplikasi dibangunkan untuk menyokong fungsi sesebuah organisasi. Kewujudan pelbagai sistem aplikasi yang beroperasi secara silo menyebabkan pertindihan fungsi dan duplikasi data. Integrasi sistem dilaksanakan untuk membolehkan sistem-sistem aplikasi yang berasingan dapat melaksanakan tugas secara bersepadu dan *seamless*. Ini dapat meningkatkan produktiviti pekerja dan memudahkan organisasi mencapai matlamatnya. Selain itu, integrasi sistem juga dapat meningkatkan ketepatan dan kebolehpercayaan data.

Dua (2) aktiviti utama Integrasi Sistem dalam Fasa Rekabentuk iaitu:

- a) Penyediaan Pelan Integrasi Data; dan
- b) Rekabentuk Integrasi Data



4.12.1 PENYEDIAAN PELAN INTEGRASI SISTEM [F3.9]

KETERANGAN

Pelan integrasi dibangunkan sebagai panduan dan rujukan bagi keseluruhan pelaksanaan integrasi sistem. Pelan integrasi menggariskan kaedah, strategi dan jadual pelaksanaan integrasi yang perlu dipatuhi.

OBJEKTIF

Menghasilkan Pelan Integrasi berfungsi sebagai rujukan yang menetapkan kaedah dan strategi yang akan digunakan serta jangkamasa yang diperlukan semasa pelaksanaan integrasi.

LANGKAH-LANGKAH

Langkah 1 : Analisis Keperluan

Analisis Keperluan Integrasi perlu dilaksanakan untuk tujuan penyediaan Pelan Integrasi. Kajian dan perbincangan perlu melibatkan antara pemilik proses, pemilik sistem dan pasukan pembangun integrasi. Berikut adalah perkara-perkara yang perlu diambil kira untuk tujuan penyediaan Pelan Integrasi:

Jadual 55 : Analisa Keperluan Integrasi Sistem

Perkara	Keterangan
Apakah tujuan pengintegrasian? Apakah fungsi bisnes yang perlu disokong?	Fungsi bisnes yang ingin disokong melalui pengintegrasian sistem.
Apakah skop integrasi?	Sistem yang terlibat dan komponen sistem yang terlibat. Nyatakan juga integrasi yang melibatkan sistem luaran milik agensi luar.
Apakah strategi pelaksanaan integrasi?	Strategi yang akan dilaksanakan sama ada bersekali dengan pembangunan sistem utama ataupun selepas sistem utama siap. Strategi pelaksanaan integrasi perlu diselaraskan dengan D01 Pelan Pembangunan Sistem merangkumi fasa pembangunan, pengujian dan pelaksanaan.
Apakah keperluan perisian dan perkakasan untuk pembangunan integrasi?	Perisian dan perkakasan yang diperlukan untuk pembangunan integrasi.

Siapa yang terlibat dalam proses integrasi sistem?	Sumber manusia dan pihak berkepentingan yang akan terlibat dan peranan mereka dalam pembangunan, pengujian dan pelaksanaan integrasi
Apakah aktiviti yang terlibat?	Jadual masa bagi setiap aktiviti yang terlibat.

Langkah 2 : Sediakan Pelan Integrasi

Pelan Integrasi dihasilkan selepas analisa keperluan selesai dijalankan dan mengandungi sekurang-kurangnya perkara seperti berikut:

Jadual 56 : Isi Kandungan Pelan Integrasi Sistem

Kandungan	Keterangan
Objektif	Terangkan objektif pengintegrasian sistem ini
Skop kerja integrasi	Terangkan skop kerja integrasi yang terlibat
Pendekatan dan strategi	Pendekatan dan strategi yang digunakan untuk melaksanakan integrasi - Terangkan kaedah dan strategi yang dilakukan untuk mengenalpasti integrasi yang diperlukan dan pelaksanaan integrasi tersebut.
Kaedah integrasi, <i>tools</i> dan persekitaran	Terangkan kaedah integrasi dan <i>tools</i> yang terlibat dalam integrasi tersebut. Nyatakan juga persekitaran yang digunakan untuk pengujian integrasi tersebut
Tugas dan tanggungjawab	Terangkan tugas dan tanggungjawab pasukan integrasi dan pihak berkepentingan. Sertakan juga carta organisasi (jika berkaitan).
Jadual Pelaksanaan	Nyatakan tempoh masa yang diperlukan untuk setiap aktiviti yang dirancang. Aktiviti melibatkan fasa kajian, rekabentuk, pembangunan, pengujian dan pelaksanaan integrasi.
Andaian dan Risiko	Terangkan andaian yang dibuat sepanjang pelaksanaan integrasi dan potensi halangan yang akan memberi impak kepada pelaksanaan integrasi tersebut.

Pelaksanaan integrasi akan dilaksanakan berdasarkan pelan yang dihasilkan. Rujuk format **D07 Pelan Integrasi Data**.

Langkah 3 : Sahkan Pelan Integrasi

Pelan Integrasi yang didokumenkan perlu dibenteng dan mendapat pengesahan kesemua pemilik sistem yang terlibat dengan integrasi bagi memastikan aktiviti integrasi mendapat sokongan dan kerjasama.

4.12.2 REKABENTUK INTEGRASI SISTEM [F3.10]

KETERANGAN

Integrasi sistem dilaksanakan bagi membolehkan sistem-sistem aplikasi yang berasingan bertukar maklumat secara automatik dan *seamless*. Ini kerana terdapat pelaksanaan proses bisnes yang memerlukan fungsi-fungsi daripada sistem aplikasi berlainan berinteraksi antara satu sama lain.

Service Orientation Architecture (SOA) merupakan kaedah integrasi yang menggunakan servis untuk berinteraksi antara sistem aplikasi. Servis merupakan logik fungsian yang dibangunkan mewakili proses bisnes. Ia boleh dikemaskini tanpa mengganggu servis lain dan ini menggalakkan pengintegrasian dilaksanakan secara *loosely coupled*. Selain itu, servis yang dibangunkan tidak terikat dengan platform, bahasa pengaturcaraan, sistem pengoperasian dan persekitaran sistem aplikasi. Ini kerana ia berkomunikasi menggunakan format data yang sama.

OBJEKTIF

Menghasilkan spesifikasi integrasi bagi pelaksanaan keperluan proses bisnes yang merentasi fungsi bisnes agensi atau unit bisnes dalam agensi.

LANGKAH-LANGKAH

Langkah 1 : Kenal Pasti Keperluan Integrasi

Kenal pasti kebergantungan sistem yang sedang dibangunkan dengan sistem-sistem lain. Maklumat tersebut boleh diperolehi daripada hasil kajian keperluan sistem yang terdapat D03 Spesifikasi Keperluan Sistem. Pemodelan *Use Case* dan Rajah Konteks DFD turut digunakan dalam dokumen tersebut untuk menggambarkan skop interaksi sistem dengan pengguna atau sistem luaran secara menyeluruh.

Contoh Kajian Kes:

Rajah Konteks Sistem Pengurusan Tempahan Bilik Mesyuarat, **DFD-BM** menggambarkan secara keseluruhan hubungan sistem dengan entiti luaran. Berdasarkan Rajah Konteks tersebut, Sistem Pengurusan Tempahan Bilik Mesyuarat berintegrasi dengan Sistem Sumber Manusia dan Sistem Senggara Aset. Maka langkah seterusnya akan tertumpu kepada proses yang berlaku dan maklumat yang dikongsi antara sistem tersebut.

Langkah 2 : Kenalpasti Servis Integrasi Yang Diperlukan

- a) Senaraikan keperluan integrasi yang diperlukan. Nyatakan secara ringkas keperluan integrasi tersebut. Maklumat terperinci berkaitan servis tersebut boleh diperolehi daripada pemodelan *Use Case* dan *Data Flow Diagram* (DFD) dalam **D03 Spesifikasi Keperluan Sistem** serta perincian Rajah Aliran Proses Bisnes (PFD) dalam **D02 Spesifikasi Keperluan Bisnes**.

- b) Maklumat yang perlu ada bagi keperluan servis integrasi adalah seperti di **Apendiks 9a Maklumat Servis Integrasi**. Jadual dibawah menunjukkan keterangan templat maklumat servis integrasi.

Jadual 57 : Keterangan Templat Maklumat Servis Integrasi

Nama Label	Keterangan
Rujukan Fungsi	Label dan nama Rajah Aliran Proses Bisnes yang dirujuk
Rujukan Aktiviti	Label dan nama <i>Use Case</i> yang dirujuk
Nama sistem sumber	Sistem yang membekalkan maklumat
Pemilik maklumat	Bahagian yang bertanggungjawab atas maklumat tersebut
Keterangan maklumat yang dihantar	Penerangan ringkas mengenai maklumat yang dihantar ke sistem lain
Tujuan penggunaan maklumat	Penerangan ringkas mengenai tujuan maklumat tersebut dihantar ke sistem lain

Contoh Kajian Kes:

Penguraian **DFD-BM-MA** menunjukkan bahawa integrasi berlaku semasa menguruskan aduan kerosakan. Perincian aktiviti berkaitan integrasi tersebut boleh diperolehi daripada Definisi Fungsi Bisnes pada Pemodelan Proses Bisnes (PFD), **PFD-BM-MA** dan Rajah *Use Case UC-BM-MA*. Senaraikan maklumat yang diperlukan seperti jadual berikut:

Jadual 58 : Contoh Maklumat Aktiviti yang Memerlukan Integrasi

Bil	Rujukan Fungsi	Rujukan Aktiviti	Nama Sistem Sumber	Pemilik Maklumat	Keterangan Maklumat yang Dihantar	Tujuan Penggunaan Maklumat
1	PFD-BM-MA-01 Mengurus Aduan Kerosakan	UC-BM-MA-01-001 Sediakan Aduan Kerosakan Baru	Sistem Pengurusan Tempahan Bilik Mesyuarat	Unit Sumber Manusia	Aduan baru bagi kerosakan bilik mesyuarat	Pengemaskinian status aset (bilik mesyuarat)
2	PFD-BM-MA-01 Mengurus Aduan Kerosakan	UC-BM-MA-01-002 Kemaskini Status Kesediaan Bilik Mesyuarat	Sistem Senggara Aset	Unit Pentadbiran	Aduan kerosakan yang telah selesai	Pengemaskinian status kesediaan bilik mesyuarat

Berdasarkan jadual di atas, terdapat dua servis integrasi yang perlu dibangunkan iaitu:

- i) Sistem Pengurusan Tempahan Bilik Mesyuarat akan menghantar maklumat aduan bagi kerosakan bilik mesyuarat.
- ii) Sistem Senggara Aset akan menghantar maklumat aduan kerosakan yang telah selesai.

Langkah 3 : Muktamadkan Format Pertukaran Data (*Data Exchange Format*)

- a) Berdasarkan senarai integrasi yang telah dikenalpasti, berikan nama servis untuk setiap integrasi tersebut dan pendekatan kaedah integrasi yang digunakan. Terangkan format pertukaran data yang akan digunakan. Rujuk Apendiks 9b) Format Pertukaran Data. Keterangan bagi templat Format Pertukaran Data adalah seperti di jadual dibawah.

Jadual 59 : Keterangan Templat Format Pertukaran Data

Nama Label	Keterangan
Nama Servis	Nama servis integrasi yang ingin dibangunkan.
Keterangan	Penerangan mengenai servis integrasi yang dibangunkan.
Kaedah Integrasi	Kaedah integrasi yang digunakan dan format pertukaran data yang digunakan.
Url <i>Web Service</i>	Alamat url servis integrasi tersebut. Nyatakan url yang digunakan semasa pembangunan, pengujian dan pelaksanaan (jika berkaitan).
<i>Request</i>	Kaedah permohonan maklumat.
<i>Respond</i>	Kaedah maklumbalas.
Data yang terlibat	Penerangan struktur data yang terlibat dalam integrasi.

Contoh Kajian Kes:

Berdasarkan contoh kajian kes, jadual dibawah menunjukkan keterangan servis integrasi dan data yang terlibat dalam integrasi tersebut.

Jadual 60 : Contoh Penerangan Servis Integrasi dan Data yang Terlibat

Nama Servis	Aduan Kerosakan																																																						
Keterangan	Menghantar maklumat aduan kerosakan baru daripada Sistem Pengurusan Tempahan Bilik Mesyuarat																																																						
Kaedah Integrasi	RESTful service Format: JSON																																																						
Url Web Service	Testing: testing.company.com.my/sptbm/rest/integration Production: www.sptbm.my/rest/integration																																																						
Request	Sistem luar perlu hantar permintaan kepada link berikut: <url web service>/aduan Arahan: GET																																																						
Respond	<pre>{ "result" : { "aduan" : [{ "no_aduan" : , "tajuk_aduan" : , "keterangan_aduan" : , "nama_bilik_mesy" : , "status_bilik_mesy" : , "tarikh_daftar_aduan" : , "jenis_kerosakan" : , "status_penyelenggaraan" : , "tarikh_hantar_data" : }] } }</pre>																																																						
Data yang terlibat	<table border="1"> <thead> <tr> <th>Nama</th> <th>Jenis</th> <th>Saiz</th> <th>Nullable</th> <th>Rules</th> </tr> </thead> <tbody> <tr> <td>no_aduan</td> <td>varchar</td> <td>15</td> <td>N</td> <td></td> </tr> <tr> <td>tajuk_aduan</td> <td>varchar</td> <td>100</td> <td>N</td> <td></td> </tr> <tr> <td>keterangan_aduan</td> <td>varchar</td> <td>255</td> <td>Y</td> <td></td> </tr> <tr> <td>nama_bilik_mesy</td> <td>varchar</td> <td>100</td> <td>N</td> <td></td> </tr> <tr> <td>status_bilik_mesy</td> <td>int</td> <td>2</td> <td>N</td> <td></td> </tr> <tr> <td>tarikh_daftar_aduan</td> <td>date</td> <td></td> <td>N</td> <td>dd/MM/yyyy</td> </tr> <tr> <td>jenis_kerosakan</td> <td>varchar</td> <td>100</td> <td>N</td> <td></td> </tr> <tr> <td>status_penyelenggaraan</td> <td>int</td> <td>2</td> <td>N</td> <td></td> </tr> <tr> <td>tarikh_hantar_data</td> <td>datetime</td> <td></td> <td>N</td> <td>dd/MM/yyyy HH:mm:ss</td> </tr> </tbody> </table>					Nama	Jenis	Saiz	Nullable	Rules	no_aduan	varchar	15	N		tajuk_aduan	varchar	100	N		keterangan_aduan	varchar	255	Y		nama_bilik_mesy	varchar	100	N		status_bilik_mesy	int	2	N		tarikh_daftar_aduan	date		N	dd/MM/yyyy	jenis_kerosakan	varchar	100	N		status_penyelenggaraan	int	2	N		tarikh_hantar_data	datetime		N	dd/MM/yyyy HH:mm:ss
Nama	Jenis	Saiz	Nullable	Rules																																																			
no_aduan	varchar	15	N																																																				
tajuk_aduan	varchar	100	N																																																				
keterangan_aduan	varchar	255	Y																																																				
nama_bilik_mesy	varchar	100	N																																																				
status_bilik_mesy	int	2	N																																																				
tarikh_daftar_aduan	date		N	dd/MM/yyyy																																																			
jenis_kerosakan	varchar	100	N																																																				
status_penyelenggaraan	int	2	N																																																				
tarikh_hantar_data	datetime		N	dd/MM/yyyy HH:mm:ss																																																			

Jadual di atas menerangkan servis integrasi yang akan dibangunkan untuk membolehkan Sistem Pengurusan Tempahan Bilik Mesyuarat menghantar maklumat aduan kerosakan bilik mesyuarat ke sistem luar (Sistem Senggara Aset).

Langkah 4 : Sediakan Pemetaan Data (*Data Mapping*)

- a) Kenalpasti struktur data yang diperlukan oleh sistem dan struktur data yang dihantar oleh sistem sumber. Petakan struktur data tersebut. Apendiks 9c) Pemetaan Struktur Data. Keterangan bagi templat pemetaan struktur data adalah seperti di jadual dibawah.

Jadual 61 : Keterangan Templat Pemetaan Struktur Data

Nama Label	Keterangan
Nama Servis	Nama servis integrasi.
Nama Sistem Penerima	Sistem yang memohon dan menerima data tersebut melalui servis integrasi.
Nama Sistem Pemilik	Sistem yang menyedia dan menghantar data tersebut kepada pemohon melalui servis integrasi.

Contoh Kajian Kes:

Berdasarkan contoh kajian kes, jadual dibawah menunjukkan contoh pemetaan struktur data yang terlibat.

Jadual 62 : Contoh Pemetaan Struktur Data

Nama Servis							
Aduan Kerosakan							
Nama Sistem Penerima				Nama Sistem Pemilik			
Sistem Selenggara Aset				Sistem Pengurusan Tempahan Bilik Mesyuarat			
Nama Medan	Jenis	Saiz	Keterangan	Nama Medan	Jenis	Saiz	Keterangan
no_siri	varchar	12		no_aduan	string	12	
tajuk	varchar	255		tajuk_aduan	string	100	
keterangan	text			keterangan_aduan	string	255	
nama_bilik_mesyuarat	varchar	255		nama_bilik_mesy	string	100	
status_bilik_mesyuarat	int	2		status_bilik_mesy	number	2	
tarikh_daftar	date		dd/MM/yyyy	tarikh_daftar_aduan	string		dd/MM/yyyy
jenis_kerosakan	varchar	255		jenis_kerosakan	string	100	

status_ penyelenggaraan	int	2		status_ penyelenggaraan	number	2	
tarikh_terima_data	datetime		dd/MM/yyyy HH:mm:ss	tarikh_hantar_data	string		dd/MM/yyyy HH:mm:ss

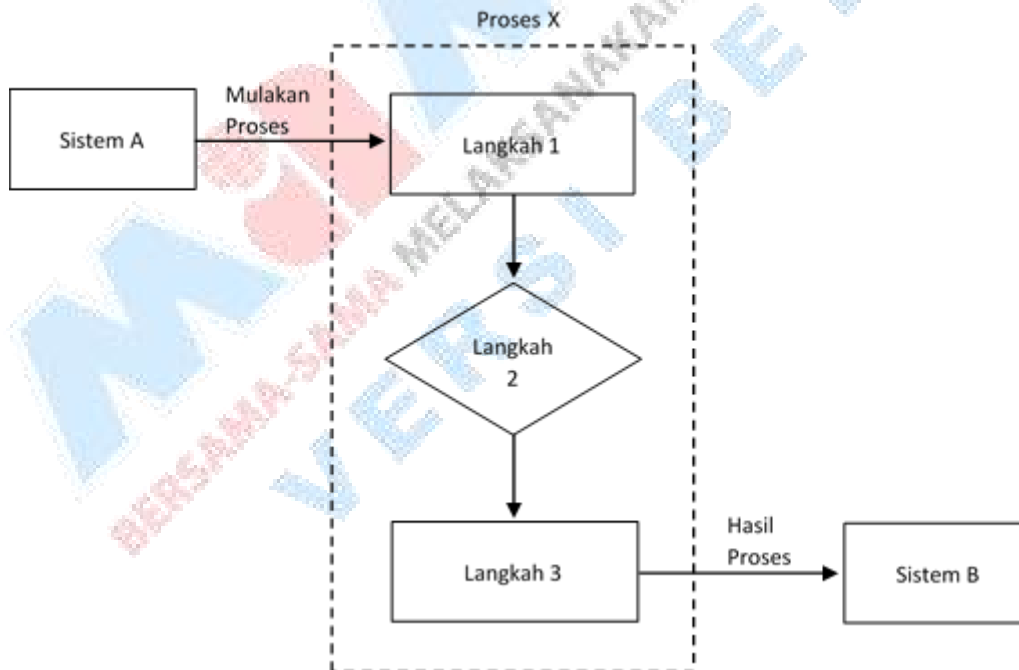
Jadual di atas menunjukkan pemetaan data aduan kerosakan bilik mesyuarat antara Sistem Senggara Aset dengan Sistem Pengurusan Tempahan Bilik Mesyuarat. Ini untuk memastikan data yang diterima daripada Sistem Pengurusan Tempahan Bilik Mesyuarat memenuhi keperluan Sistem Senggara Aset.

Langkah 5 : Sediakan Peraturan Integrasi Data

Kenalpasti aliran proses yang terlibat semasa pertukaran data berlaku. Nyatakan peraturan yang dilaksanakan semasa pertukaran data tersebut seperti logik integrasi dan transformasi data. Terdapat 2 jenis proses integrasi iaitu:

a) Proses Khusus (*Specialized Processes*)

Proses Khusus merupakan pengintegrasian untuk melaksanakan proses yang khusus untuk sistem tertentu. Ini bermakna hanya ada satu output sahaja yang dihasilkan oleh proses tersebut untuk diproses oleh sistem tertentu.

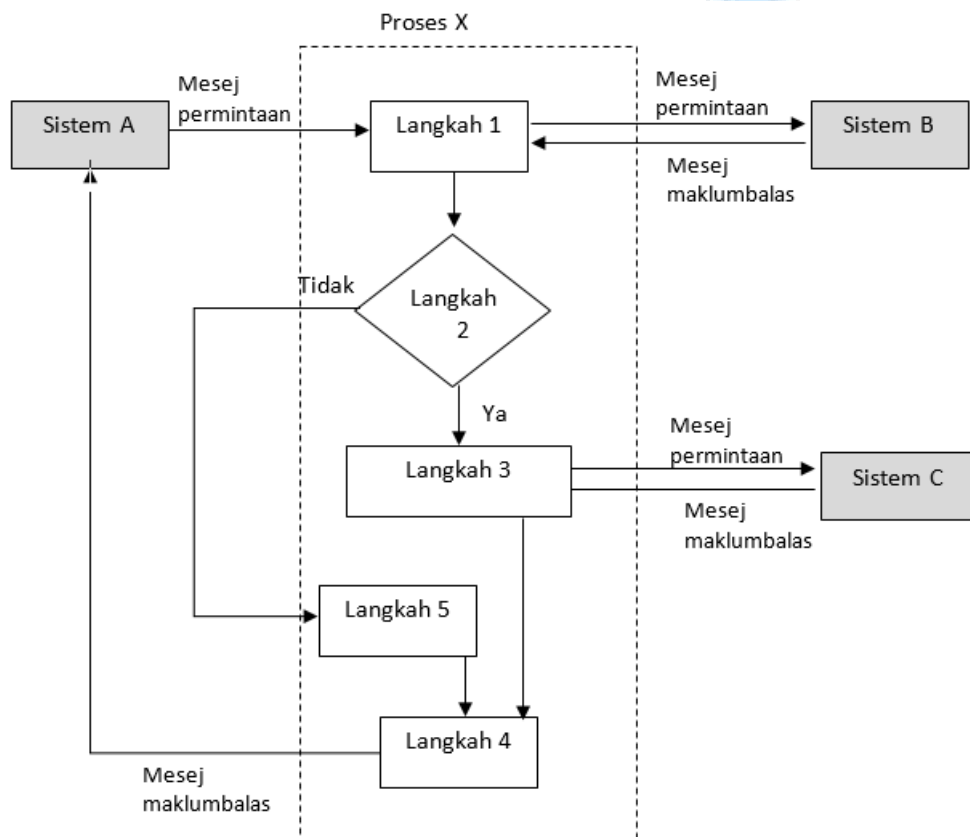


Rajah 75 : Proses Khusus

Berdasarkan rajah di atas, Sistem A akan memulakan Proses X yang terdapat pada server integrasi. Output yang dihasilkan oleh Proses X akan dihantar kepada Sistem B untuk diproses. Proses X ini diwujudkan untuk menyokong Sistem B agar sistem lain boleh melaksanakan prosedur tersebut tanpa perlu mengetahui senarai proses yang perlu dilakukan.

b) Proses Berbilang Langkah (Multistep Process)

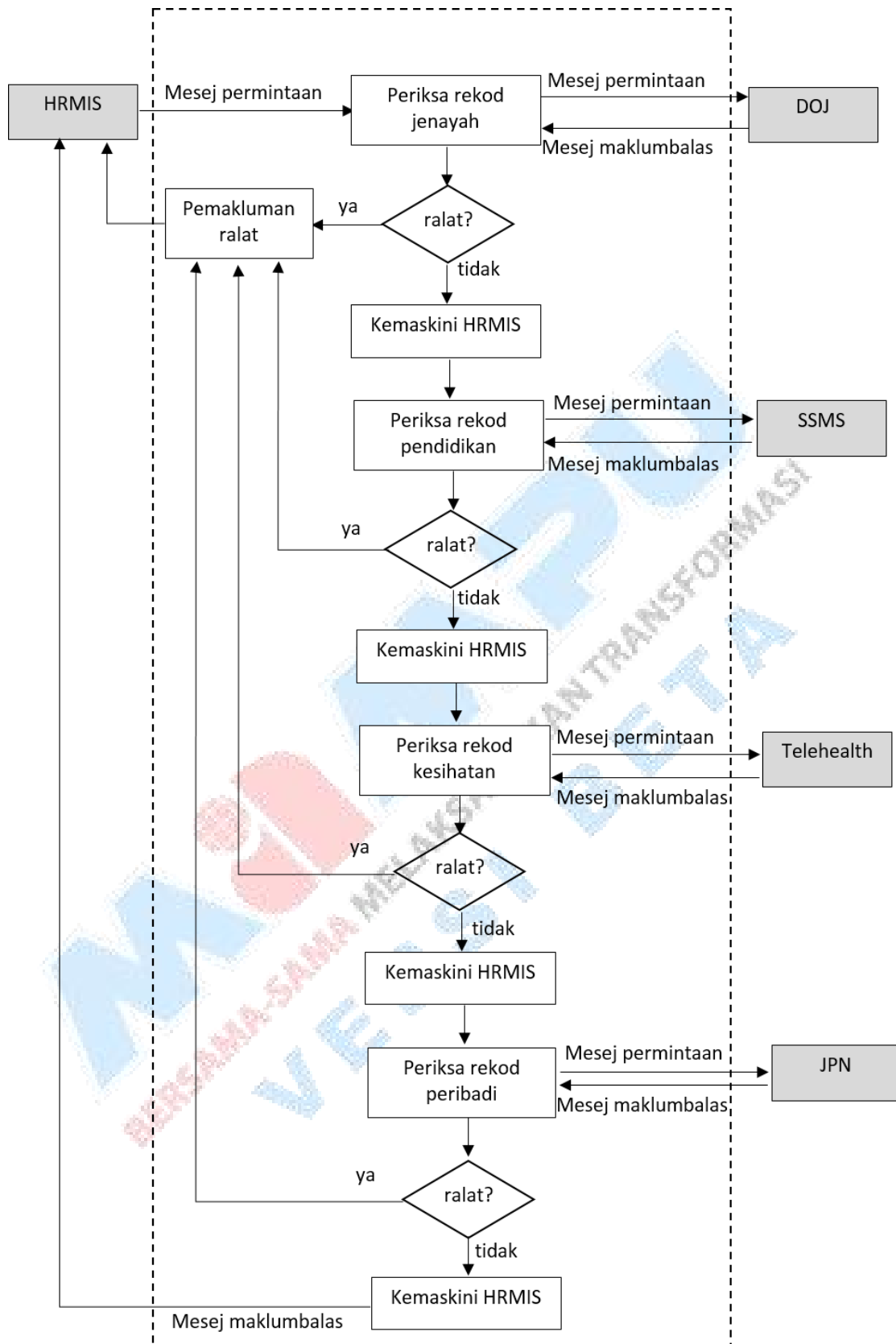
Proses Berbilang Langkah melibatkan pengintegrasian lebih dari satu sistem untuk melaksanakan proses tertentu. Aliran proses yang diwujudkan dalam server integrasi akan mengendalikan interaksi antara sistem tersebut. Semua transaksi antara proses dan sistem tersebut perlu mematuhi kekangan transaksi *atomic*. Ini bermakna sekiranya salah satu transaksi gagal, maka proses tersebut akan gagal. Maka semua perubahan yang telah berlaku kepada data perlu kembali kepada asal seolah-olah proses tersebut tidak berlaku. Mekanisma pengurusan pengecualian (*exception handling mechanism*) perlu diwujudkan untuk melaksanakan pembalikan (*rollback*) tersebut.



Rajah 76 : Proses Berbilang Langkah

Rajah di atas menunjukkan Sistem A perlu berintegrasi dengan Sistem B dan Sistem C untuk melaksanakan Proses X. Mesej permintaan digunakan untuk meminta menggunakan servis atau data yang dikongsi oleh sistem lain manakala mesej maklumbalas ialah hasil proses servis atau data yang diminta oleh mesej permintaan tersebut.

Contoh Senario Proses Integrasi:

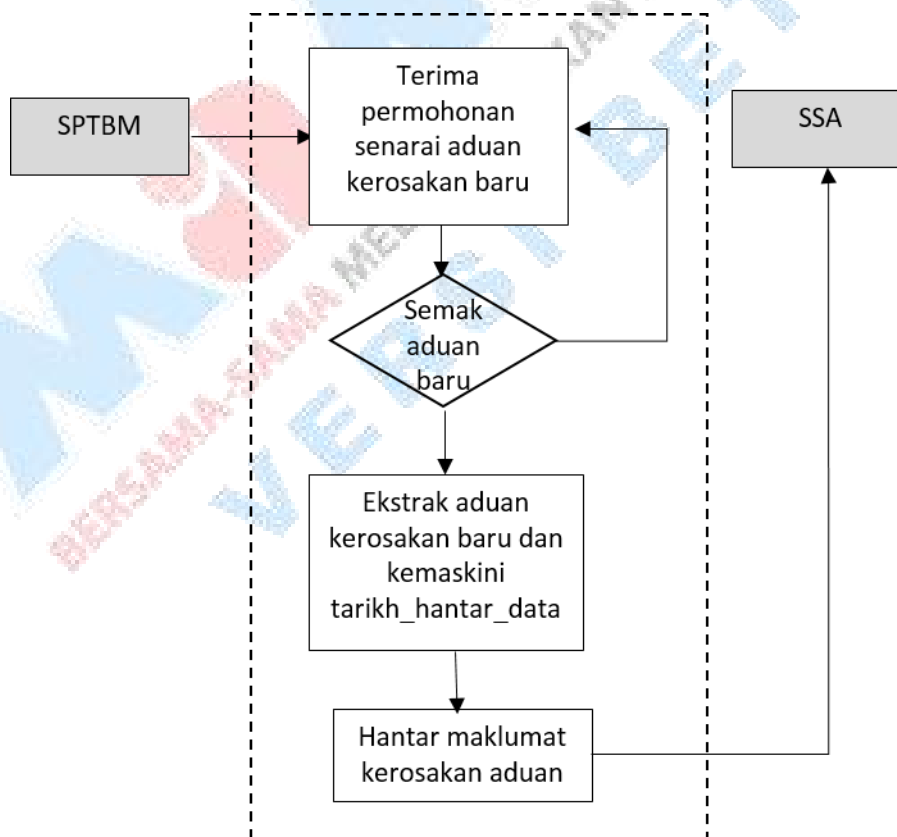


Rajah 77 : Contoh Senario Proses Integrasi

Berdasarkan proses di atas, Sistem Maklumat Rekod Pekerja (HRMIS) akan menghantar maklumat pekerja kepada proses integrasi untuk memeriksa rekod jenayah daripada Sistem Maklumat Kehakiman (DOJ), rekod pendidikan daripada Sistem Maklumat Pendidikan (SSMS), rekod kesihatan daripada Sistem Maklumat Kesihatan (Telehealth) dan rekod peribadi daripada Sistem Maklumat Pendaftaran (JPN). Jika semua rekod memenuhi kriteria, proses pengemaskinian akan dilakukan kepada sistem Telehealth dan HRMIS. Jika berlaku kegagalan semasa transaksi, mekanisme pengurusan pengecualian (exception handling mechanism) akan memainkan peranan untuk melaksanakan pembalikan (rollback) proses tersebut. Segala perubahan yang telah berlaku kepada sistem yang terlibat akan dibatalkan dan proses tersebut perlu diulang semula.

Contoh Kajian Kes:

Sistem Senggara Aset (SSA) akan memohon maklumat aduan kerosakan daripada Sistem Pengurusan Tempahan Bilik Mesyuarat (SPTBM). Sistem Pengurusan Tempahan Bilik Mesyuarat akan mendapatkan maklumat aduan kerosakan baru dan menghantar maklumat tersebut bersama tarikh_hantar_data yang telah dikemaskini. Sistem Senggara Aset akan menerima maklumat tersebut dan menyimpannya untuk tindakan seterusnya.



Rajah 78 : Contoh Proses Integrasi Servis Aduan Kerosakan

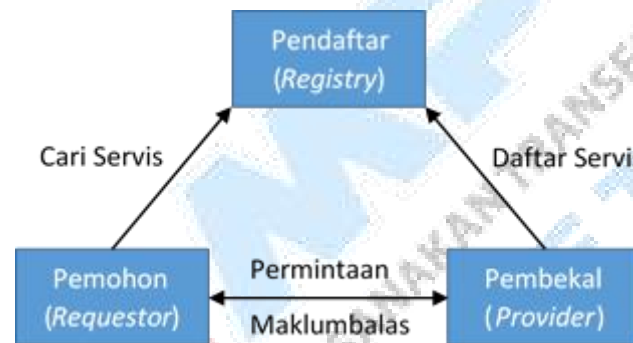
Langkah 6 : Sediakan Reka Bentuk Arkitektur Integrasi

Reka bentuk arkitektur integrasi adalah bergantung kepada cara dan peraturan integrasi tersebut dilakukan. Penerangan di bawah menunjukkan kaedah integrasi menggunakan SOAP, *RESTful* dan *messaging*.

a) Arkitektur SOAP

SOAP bermaksud *Simple Object Access Protocol*. Terdapat tiga peranan utama dalam arkitektur *web service*:

- i) Pembekal (*Provider*) - mewujudkan servis dan menjadikannya tersedia untuk sistem aplikasi yang ingin menggunakannya.
- ii) Pemohon (*Requestor*) - menggunakan servis sedia ada dengan menghantar permintaan kepada pembekal servis.
- iii) Pendaftar (*Registry*) – menyimpan maklumat servis yang disediakan oleh pembekal servis.

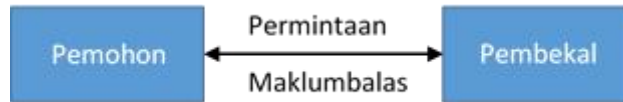


Rajah 79 : Arkitektur SOAP

Pembekal servis akan memaklumkan kepada pendaftar tentang servis yang disediakan dan cara menggunakannya. Pemohon akan mencari servis yang dikehendaki dalam pendaftar. Setelah mendapat maklumat mengenai servis yang diperlukan, pemohon akan menghantar permintaan kepada pembekal servis tersebut. Pembekal akan memberi maklumbalas berdasarkan jenis permintaan tersebut.

b) Arkitektur *RESTful*

8. REST bermaksud *REpresentational State Transfer*. Ia mempunyai ciri-ciri berikut:
 - i) Setiap sumber seperti maklumat boleh diakses melalui URL.
 - ii) Kata arahan GET, POST, PUT atau DELETE akan digunakan semasa mengakses sumber tersebut.



Rajah 80 : Arkitektur *RESTful*

Arkitektur *RESTful* berkonsepkan *client-server*. Permintaan akan dilakukan oleh klien/pemohon dengan cara menghantar URL sumber yang diperlukan dan kata arahan. Pelayan/pembekal servis akan memberi maklumbalas berdasarkan URL dan kata arahan permintaan tersebut.

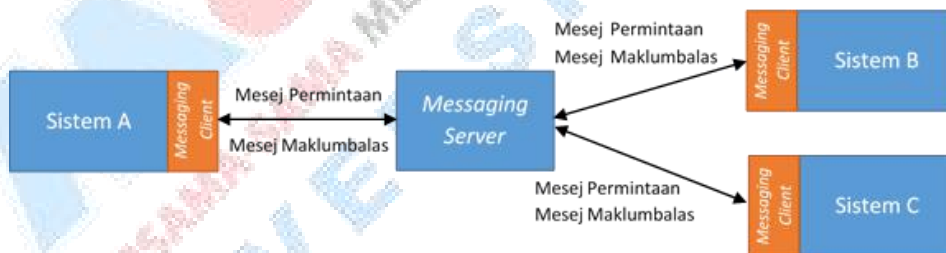
c) Arkitektur *Messaging*

Penghantaran Mesej (*Messaging*) membolehkan setiap sistem aplikasi berkomunikasi menggunakan mesej melalui saluran mesej (*message channel*) yang sama. Mesej yang dihantar oleh sistem aplikasi akan diuruskan oleh sistem penghantaran mesej (*messaging system*). Perkongsian data dan arahan adalah menggunakan mesej.



Rajah 81 : Arkitektur *Messaging*

Setiap sistem aplikasi akan dipasang dengan *messaging client* untuk membolehkannya menghantar dan menerima mesej. *Messaging server* bertindak sebagai orang tengah yang menguruskan penghantaran mesej.



Rajah 82 : Arkitektur *Messaging* untuk Pengintegrasian Berbilang Server

Arkitektur *messaging* memudahkan pengintegrasian yang melibatkan proses berbilang langkah (*multistep step*) kerana ia menyokong pengintegrasian berbilang sistem. Proses integrasi tersebut akan dibangunkan dalam *messaging server*.

Langkah 7 : Dokumentasikan Spesifikasi Integrasi Data

Kompilkan hasil langkah-langkah yang telah dilaksanakan ke dalam dokumen **D08 Spesifikasi Integrasi Data**.

Langkah 8 : Dapatkan Pengesahan Pengguna

Spesifikasi Integrasi Data yang didokumentasikan perlu dibentangkan dan mendapat pengesahan pemilik sistem bagi memastikan kesahihan dan spesifikasi yang dihasilkan memenuhi keperluan integrasi data.

RUJUKAN

1. Katalog Servis Pasukan Perunding ICT Sektor Awam (2013)
2. Dokumen SRDS Projek Perkongsian Maklumat Perkhidmatan Perguruan – Edu-XChange (2011)
3. Gregor Hohpe & Bobby Woolf (2003). Enterprise Integration Patterns. Addison-Wesley. ISBN: 0321200683
4. MSC CFI Architecture – System Integration Architecture Version 3.0 (2004)
5. Dokumen uCustoms Penang Port Sdn Bhd (PPSB) Integration Specification Issue 1.0 (2015)



BAB:05

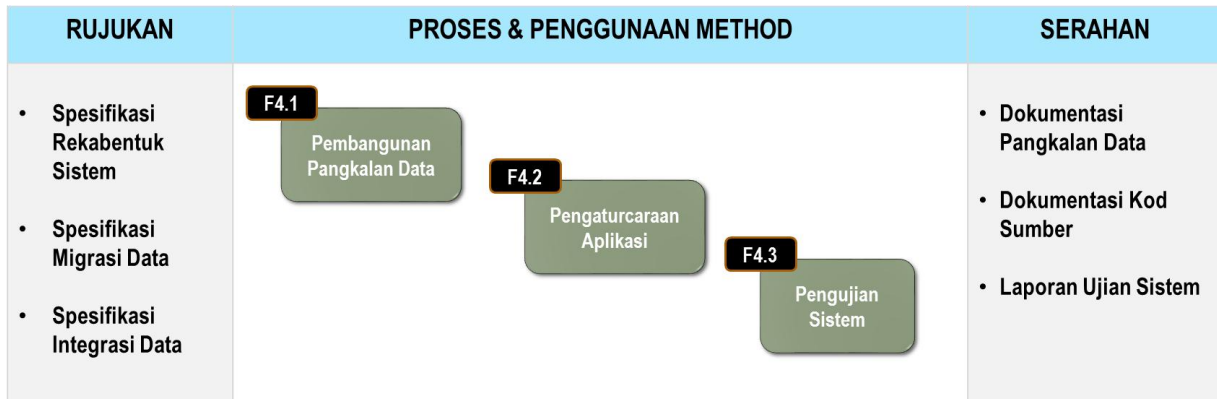
FASA PEMBANGUNAN

Bab ini menerangkan aktiviti-aktiviti pembangunan sistem aplikasi serta amalan terbaik dalam pengaturcaraan, pembangunan pangkalan data dan pengujian yang perlu dilaksanakan ke atas sistem aplikasi.



5 PEMBANGUNAN

5.1 GAMBARAN KESELURUHAN



Rajah 83 : Gambaran Keseluruhan Fasa IV - Pembangunan

5.2 PENGENALAN

Fasa pembangunan merupakan fasa yang sangat penting dalam Kitar Hayat Pembangunan Sistem (SDLC), pada fasa inilah bermulanya pembangunan sistem yang sebenar, fasa sebelum ini merupakan asas kepada pembangunan sistem. Manakala fasa seterusnya adalah untuk memastikan sistem memenuhi kehendak pengguna. Pada fasa ini kebanyakan fungsi-fungsi atau modul-modul sistem yang akan dibangunkan telah dimuktamatkan.

Matlamat utama fasa ini ialah untuk menterjemahkan atau merealisasikan SDS yang dihasilkan dalam fasa rekabentuk kepada kod aturcara dalam bahasa pengaturcaraan yang telah dipilih, memasang dan seterusnya melaksanakan pengujian sistem. Pengujian dilaksanakan agar sistem yang dibangunkan bebas dari sebarang ralat, dapat berfungsi sepenuhnya dan berjaya memenuhi keperluan sebenar pembangunan. Ini dapat memastikan sistem memenuhi kriteria-kriteria kualiti yang telah ditetapkan bagi sesebuah perisian serta meningkatkan tahap keyakinan pengguna ke atas perisian yang dibangunkan.

Fasa Pembangunan Sistem melibatkan 3 aktiviti utama iaitu:

- Pembangunan Pangkalan Data
- Pengaturcaraan Aplikasi
- Ujian Sistem

Dokumen Rujukan kepada Fasa Pembangunan adalah seperti berikut:

- a) **D04 Spesifikasi Rekabentuk Sistem**
- b) **D06 Spesifikasi Migrasi Data**
- c) **D08 Spesifikasi Integrasi Sistem**

Dokumen Serahan kepada Fasa Pembangunan adalah seperti berikut:

- a) **D09 Dokumentasi kod Pangkalan Data**
- b) **D10 Dokumentasi Kod Sumber**
- c) **D11 Laporan Ujian Sistem**

5.3 PENGLIBATAN PEMEGANG TARUH

Pemegang taruh utama yang patut terlibat dalam fasa pembangunan adalah Pengaturcara Program, Penguji Perisian, Pemilik prosidur/proses semasa dan Pengguna. Pengaturcara program perlulah berkeupayaan untuk menterjemahkan keperluan rekabentuk sistem kepada kod program dan logik pembangunan untuk menghasilkan sistem yang berkualiti dan memenuhi kehendak pelanggan bagi memastikan sistem yang dibangunkan bertepatan dengan keperluan. Manakala Penguji Sistem pula perlulah berkeupayaan membangunkan pelan pengujian dan skrip ujian berdasarkan SRS dan SDS yang diberikan.

Cadangan penglibatan kategori pemegang taruh adalah seperti berikut:

a) **Pengurus Projek**

Memantau kemajuan pembangunan supaya mengikut perancangan, mengenalpasti risiko semasa pembangunan, berkomunikasi antara pembangunan dengan SME.

b) **Pengaturcara Program**

Menulis kod aturcara untuk menghasilkan sistem aplikasi

c) **Penguji Sistem**

Menguji sistem bagi memastikan sistem aplikasi yang dibangunkan berkualiti dan sedia untuk dilancarkan.

d) **Pemilik Prosidur/Proses Semasa dan Pengguna**

Melaksanakan ujian penerimaan pengguna.

5.4 FAKTOR KEJAYAAN

Untuk memastikan aktiviti berjaya dilaksanakan, berikut adalah faktor kejayaan utama yang perlu dipertimbangkan sebelum dan semasa aktiviti dilaksanakan:

- a) Spesifikasi Rekabentuk Sistem (SDS) yang didokumenkan adalah lengkap dan memenuhi kehendak pengguna.
- b) Pasukan pengaturcara program berupaya menterjemahkan SDS kepada kod aturcara dan logik pemrograman, mempunyai kemahiran dan kepakaran dalam bahasa pengaturcaraan yang dipilih dan SQL (*skill coding dan SQL query*).
- c) Bilangan pengaturcara yang mencukupi dan bersesuaian dengan masa pembangunan.
- d) Kelengkapan tools dan persekitaran pembangunan yang sempurna.



5.5 PEMBANGUNAN PANGKALAN DATA [F4.1]

KETERANGAN

Pembangunan Pangkalan Data merupakan proses mewujudkan pangkalan data fizikal berdasarkan reka bentuk pangkalan data logikal dan arkitektur pangkalan data. Model data fizikal menggambarkan bagaimana penstoran data dan capaian data dilakukan. Model ini juga turut menggambarkan jadual, medan yang terdapat dalam jadual serta spesifikasi bagi storan secara fizikal yang juga boleh merangkumi pengagihan data dan mekanisma capaian. Ciri-ciri data model fizikal adalah seperti berikut:-

- a) Spesifikasi semua jadual dan medan;
- b) Kekunci Asing digunakan bagi mengenalpasti hubungan antara jadual;
- c) Data model fizikal adalah berbeza mengikut RDBMS.

Pembangunan Pangkalan Data dilaksanakan oleh Pentadbir Pangkalan Data atau lebih dikenali sebagai *Database Administrator (DBA)*. Terdapat beberapa bahasa yang boleh digunakan untuk membangunkan pangkalan data, walau bagaimanapun buku panduan ini hanya mengkhususkan kepada penggunaan *Structured Query Language (SQL)*. SQL merupakan bahasa khusus yang digunakan untuk pembangunan pangkalan data dan mengurus data-data yang berada dalam Sistem Pengurusan Pangkalan Data Hubungan (*Relational Database Management System-RDBMS*). SQL terdiri daripada:

- a) Bahasa Kawalan Data (*Data Control Language-DCL*) seperti *GRANT, REVOKE*;
- b) Bahasa Definisi Data (*Data Definition Language-DDL*) seperti *CREATE, DROP, ALTER, RENAME, TRUNCATE*; dan
- c) Bahasa Manipulasi Data (*Data Manipulation Language-DML*) seperti *SELECT, INSERT, UPDATE, DELETE*.

OBJEKTIF

Membangunkan pangkalan data fizikal untuk tujuan pembangunan dan pengujian sistem

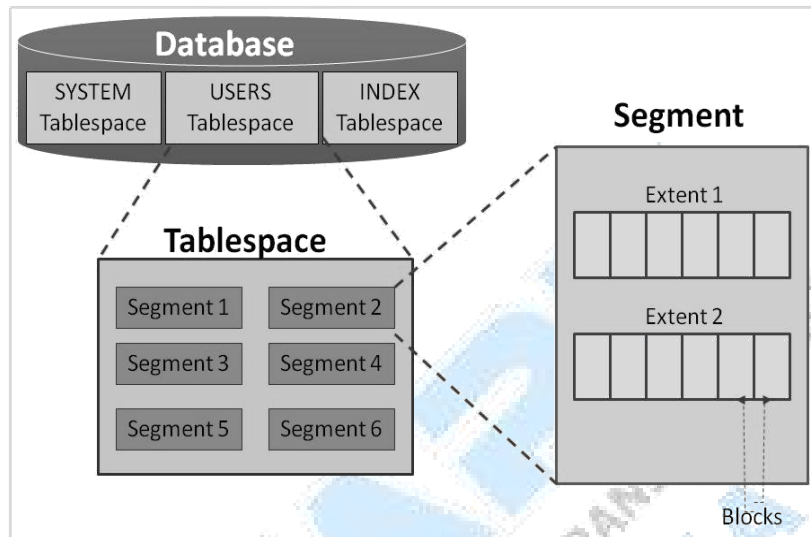
LANGKAH-LANGKAH

Langkah 1 : Pemasangan (*Install*) Perisian RDBMS

Bagi membangunkan pangkalan data fizikal, perisian RDBMS yang dipilih perlu dipasang terlebih dahulu. Sekiranya MySQL yang dipilih, perisian tersebut boleh dimuat turun daripada laman web MySQL dan rujuk dokumentasi pemasangan langkah demi langkah daripada laman web yang sama.

Langkah 2 : Peruntukan Ruang Jadual (*Tablespace*)

- a) Struktur storan pangkalan data terdiri daripada struktur storan fizikal dan struktur storan logikal. Struktur fizikal terdiri fail-fail seperti *datafiles*, *redo log files* dan *control files*. Manakala struktur logikal pula terdiri daripada beberapa *tablespace* iaitu ruang sebenar bagi menyimpan beberapa *datafile*.



Rajah 84 : Struktur Storannya Secara Logikal Dalam Pangkalan Data

Berdasarkan rajah di atas, *tablespace* adalah storan logikal yang mengandungi beberapa *segment*. *Segment* adalah objek pangkalan data yang terdiri daripada jadual-jadual dan *index*. Satu *segment* mengandungi beberapa set *extent* yang diperuntukkan bagi objek pangkalan data secara spesifik seperti jadual. Sebagai contoh, jadual pengguna disimpan dalam data *segment* yang tersendiri manakala *index* bagi jadual pengguna disimpan dalam *index segment* itu sendiri. Namun begitu, semua *extent* yang diperuntukkan dalam satu *segment* adalah disimpan dalam *tablespace* yang sama. Sebagai contoh, satu *segment* disimpan dalam pengguna01.dbf manakala *segment* yang satu lagi adalah dalam pengguna02.dbf. Setiap *extent* terdiri daripada beberapa data *block* yang diperuntukkan untuk menyimpan data yang spesifik. Satu data *block* adalah ruang cakera yang spesifik kepada jumlah bait (*byte*). Sebagai contoh, satu data *block* adalah ruang cakera fizikal berjumlah 2KB. Peruntukan bagi 24KB dalam satu *extent*, sebanyak 12 data *block* diperlukan. Data *block* adalah unit terkecil dalam storan pangkalan data yang diperuntukkan untuk menyimpan *datafile* secara fizikal.

- b) Ruang jadual adalah lokasi storan bagi data sebenar yang terdapat dalam objek pangkalan data. Ruang jadual hanya memperuntukkan lokasi storan bagi pangkalan data. Dengan menggunakan ruang jadual, DBA boleh mengawal bagaimana reka bentuk penstoran dilakukan semasa proses pembangunan pangkalan data. Fungsi penggunaan ruang jadual adalah untuk mengoptimumkan prestasi pangkalan data iaitu seperti pengasingan lokasi atau jenis cakera bagi penyimpanan jenis pengaksesan ke atas data. Iaitu seperti data yang kerap diindeks atau diakses perlu disimpan dalam cakera yang lebih stabil seperti *solid-state drive* (SDD) manakala data yang mengandungi data atau fail arkib disimpan dalam cakera yang biasa seperti *standard hard drive* (HDD).

- c) Arahan yang digunakan untuk mencipta ruang jadual bagi Oracle, DB2, Informix, PostgreSQL, MySQL adalah seperti berikut.

```
CREATE TABLESPACE <nama ruang jadual>;
```

Manakala SQL pula arahan adalah seperti berikut.

```
FILEGROUP <nama ruang jadual>;
```

Langkah 3 : Ciptakan Pangkalan Data (Create A Database)

- a) Pengguna yang mempunyai capaian *root* dibenarkan untuk mencipta pangkalan data. Arahan yang digunakan untuk mencipta pangkalan data ialah:

```
CREATE DATABASE <nama pangkalan data>;
```

Contoh arahan untuk mencipta pangkalan data *pdata1* adalah seperti berikut:

```
mysql> CREATE DATABASE pdata1
```

- b) Bagi melihat senarai pangkalan data yang telah dicipta, arahan yang digunakan adalah seperti berikut.

```
SHOW DATABASES;
```

Berikut adalah paparan bagi arahan tersebut.

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| pdata1 |
+-----+
4 rows in set (0.01 sec)
```

Paparan di atas menunjukkan bahawa terdapat empat (4) pangkalan data yang terdapat dalam hos pelayan (*server host*).

Langkah 4 : Wujudkan Jadual (Create Table)

- a) Jadual diwujudkan setelah pangkalan data siap dibangunkan. Jadual adalah terdiri daripada baris dan lajur yang mengandungi rekod maklumat dalam pangkalan data. Dalam reka bentuk pangkalan data logikal (sila rujuk **Reka Bentuk Pangkalan Data Logikal [F3.3]**) Entiti adalah merujuk kepada jadual (*table*) bagi pangkalan data fizikal. Manakala atribut pula adalah medan (*field*). Jadual berikut adalah pepadanan secara teknologi antara reka bentuk pangkalan data logikal dan fizikal.

Jadual 63 : Pemadanan Istilah antara reka bentuk logikal dan reka bentuk fizikal pangkalan data

Istilah dalam reka bentuk logikal	Istilah dalam reka bentuk fizikal
Entiti	Jadual (<i>table</i>)
Atribut	Medan (<i>column</i> atau <i>field</i>)
<i>Primary UID</i>	<i>Primary Key</i>
<i>Secondary UID</i>	<i>Unique Key</i>
<i>Relationship</i>	<i>Foreign Key</i>

b) Arahan bagi mencipta jadual adalah seperti berikut.

```
CREATE TABLE <nama jadual> (<nama medan> <jenis medan> NOT NULL
PRIMARY KEY <AUTO_INCREMENT sekiranya jenis medan adalah INT>,
<nama medan> <jenis medan>;
```

- i) Untuk mengelakkan data tidak diisi ciri-ciri *NOT NULL* diberikan kepada medan tersebut.
- ii) Ciri medan *AUTO_INCREMENT* adalah bagi menambahkan satu nilai seterusnya dalam medan dengan mengambil kira bahawa jenis medan adalah *INT*
- iii) Medan yang ditakrif sebagai *PRIMARY KEY* adalah menunjukkan bahawa medan tersebut adalah kunci utama dalam jadual. *PRIMARY KEY* boleh ditakrifkan kepada satu atau lebih medan.

Contoh arahan bagi mencipta jadual aset adalah seperti berikut.

```
CREATE TABLE aset (id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,
nama VARCHAR(20),
catatan VARCHAR(50);
```

Langkah 5 : Wujudkan *VIEW* (*Create VIEW*)

- a) *VIEW* adalah merujuk kepada jadual maya yang dihasilkan melalui arahan *SQL*. Jadual adalah terdiri daripada baris dan lajur yang mengandungi rekod maklumat berdasarkan arahan *SQL* yang telah dilaksanakan.
- b) Dalam satu pangkalan data, *view* dan jadual berkongsi ruang jadual. Namun begitu, *view* dan jadual tidak boleh mempunyai nama yang sama.
- c) Beberapa fungsi *SQL* boleh dimasukkan ke dalam arahan seperti *WHERE* dan arahan *JOIN* daripada beberapa jadual lain kepada jadual maya yang mana fungsi arahan tersebut dipaparkan sebagai satu jadual. Arahan bagi mencipta *view* adalah seperti berikut.

```
CREATE VIEW <nama_view> AS
SELECT <medan1>, <medan2>, ...
FROM <nama_jadual>
WHERE condition;
```


Contoh arahan bagi mencipta view bagi tempahan bilik oleh pengguna adalah seperti berikut.

```
CREATE VIEW penggunaTempahBilik AS
SELECT pengguna.nama, bilik_mesy.namabilik, bilik_mesy.lokasi
FROM pengguna, bilik_mesy
WHERE pengguna.nokp == bilik_mesy.nokp;
```

Langkah 6 : Wujudkan *INDEX* (Create *INDEX*)

- a) Kebiasaannya data disimpan tidak mengikut urutan. Data baru yang dimasukkan tidak disimpan mengikut susunan berdasarkan data yang dimasukkan terdahulu. Oleh yang demikian, tempoh untuk menemui data berdasarkan arahan adalah kurang pantas berbanding kemasukkan data. Dengan itu, *index* perlu dilakukan bagi membolehkan data ditemui dengan lebih cepat.
- b) Arahan *INDEX* digunakan untuk medan-medan tertentu dalam sesuatu jadual bagi mempercepatkan carian data. Lokasi sesuatu data lebih pantas ditemui berbanding carian satu-persatu baris yang terdapat dalam jadual jika tidak menggunakan *INDEX*. Arahan bagi mencipta *index* adalah seperti berikut.

```
create index <nama index> on <nama jadual> (<nama medan>);
```

Contoh arahan bagi mencipta indeks no kad pengenalan pengguna adalah seperti berikut.

```
create index nokp_idx on pengguna (nokp);
```

Langkah 7 : Memuat Masuk (Load) Data Ke Dalam Pangkalan Data

Sekiranya terdapat data daripada pangkalan data lama, data tersebut boleh dimuat masuk ke dalam pangkalan data yang baru dibina. Terdapat dua cara untuk memuat masuk data yang sedia ada ke dalam pangkalan data iaitu:

- a) Menggunakan Penyataan *INSERT*

Penyataan ini adalah untuk memasukkan rekod ke dalam jadual. Sintaks bagi penyataan *INSERT* adalah seperti berikut.

```
INSERT INTO <nama jadual> (<nama medan>)
VALUES (data1),
       (data2);
```

Contoh penyataan untuk memasukkan rekod ke dalam jadual adalah seperti berikut.

```
INSERT INTO pengguna (nama, emel, nombor_telefon, alamat,
BAHAGIAN_id)
VALUES ('Sanem Can Divit', 'sanem@erkenci.com.my', '03-88723038',
'No.1, Jalan Albatross, 62300 Putrajaya', 'BPI');
```

b) Menggunakan Penyataan *LOAD DATA*

Penyataan *LOAD DATA* membolehkan data banyak yang terdapat dalam fail teks dimasukkan ke dalam pangkalan dengan data menggunakan satu arahan sahaja. Sintaks bagi penyataan *LOAD DATA* adalah seperti berikut.

```
LOAD DATA INFILE <nama fail teks.txt> INTO TABLE <nama
pangkalan.nama jadual>;
```

Contoh penyataan untuk memasukkan rekod ke dalam jadual adalah seperti berikut.

```
LOAD DATA INFILE pengguna.txt INTO TABLE dbtempahan.pengguna;
```

Langkah 8 : Wujudkan Pengguna Dan Kawalan Capaian

- a) Pengguna diwujudkan untuk mengakses sesuatu pangkalan data. Dengan itu, pengguna perlu mempunyai capaian *root* untuk melaksanakan aktiviti-aktiviti pembangunan pangkalan data dan juga mencipta pengguna. Berikut adalah arahan yang digunakan untuk mencipta pengguna.

```
CREATE USER '<nama pengguna>'@ '<nama hos>' IDENTIFIED BY '<kata laluan>';
```

Contoh arahan untuk mencipta pengguna nama1 adalah seperti berikut:

```
mysql> CREATE USER 'nama1'@'localhost' IDENTIFIED BY
'katalaluan';
```

- b) Pentadbir pangkalan data (*Database Administrator – DBA*) boleh memberi kebenaran pengguna untuk mencapai beberapa akses melalui arahan *GRANT* seperti berikut:

```
GRANT ALL PRIVILEGES ON database.table TO 'user'@'localhost';
```

Arahan di atas memberikan semua kawalan akses '*GRANT ALL PRIVILEGES*' bagi semua pangkalan data dan jadual '*database.table*' kepada pengguna '*user*'.

- c) Berikut adalah senarai arahan kebenaran yang digunakan mengikut kesesuaian capaian.
- i) *ALL PRIVILEGES* – membenarkan semua aktiviti
 - ii) *CREATE* – membenarkan pengguna mencipta pangkalan data dan jadual
 - iii) *DROP* – membenarkan pengguna menghapus pangkalan data dan jadual
 - iv) *DELETE* – membenarkan pengguna menghapuskan baris rekod dalam jadual

- v) *INSERT* – membenarkan pengguna menambah baris rekod dalam jadual
 - vi) *SELECT* – membenarkan pengguna membaca rekod dalam pangkalan data
 - vii) *UPDATE* – membenarkan pengguna mengemaskini rekod dalam jadual
- d) Pentadbir pangkalan data (*Database Administrator* – DBA) boleh memberi menarik semula kebenaran pengguna melalui arahan *REVOKE* seperti berikut:

```
REVOKE ALL PRIVILEGES ON database.table TO 'user'@'localhost';
```

Arahan di atas menarik balik semua kebenaran kawalan akses '*REVOKE ALL PRIVILEGES*' bagi semua pangkalan data dan jadual '*database.table*' kepada pengguna '*user*'.

Langkah 9 : Dokumentasikan Pangkalan Data

Dokumentasikan maklumat pangkalan data fizikal yang dibangunkan ke dalam **D09 Dokumen Pangkalan Data**. Dokumentasi mengikut susunan berikut:

- a) Ringkasan maklumat pangkalan data fizikal.
- b) Skrip yang mengandungi arahan-arahan *SQL*

RUJUKAN

1. Oracle Database SQL Language Reference, 11g Release 2 (11.2)
2. MySQL 5.7 Reference Manual

5.6 PENGATURCARAAN APLIKASI [F4.2]

KETERANGAN

Pengaturcaraan aplikasi adalah proses untuk menulis kod aturcara bagi menghasilkan satu sistem aplikasi. Kod ini menentukan tindakan yang perlu dilaksanakan oleh sistem. Ia ditulis oleh pengaturcara program menggunakan bahasa pengaturcaraan tertentu. Terdapat pelbagai bahasa pengaturcaraan dan *tools* yang boleh digunakan.

Fasa ini sangat penting kerana ia mempengaruhi fasa seterusnya iaitu fasa pengujian dan penyelenggaraan. Satu kod yang ditulis dengan baik mampu mengurangkan kerja-kerja pada kedua-dua fasa tersebut. Sehubungan itu, tumpuan harus diberikan semasa fasa pembangunan ini supaya dapat menghasilkan kod aturcara yang berkualiti dan memenuhi keperluan pengguna.

Terdapat beberapa teknik dalam pengaturcaraan:

a) Teknik Pengaturcaraan Tidak Berstruktur (*Unstructured Programming*):

Kod ditulis tanpa berstruktur, semua arahan ditulis dalam fungsi main().

b) Pengaturcaraan Berstruktur (*Structure Programming*)

Pengaturcaraan berstruktur merupakan pengaturcaraan bertatacara. Kod yang besar dipecahkan kepada kaedah-kaedah pendek (juga dikenali sebagai fungsi atau tatacara) yang lebih kecil agar mudah difahami.

Pengaturcaraan berstruktur biasanya dikaitkan dengan rekabentuk yang menggunakan pendekatan atas-bawah. Dengan pendekatan ini, pengaturcara memetakan struktur yang besar dalam aturcara kepada bentuk operasi kecil, seterusnya melaksanakannya dan menguji operasi-operasi kecil tersebut, dan akhirnya menggabungkan kepada keseluruhan aturcara.

c) Pengaturcaraan Berorientasikan Objek (*Object-Oriented Programming*)

Pengaturcaraan berorientasikan objek merupakan aturcara komputer yang terdiri daripada sekumpulan unit-unit atau objek. Setiap objek berupaya untuk menerima dan menghantar mesej (pesanan) kepada objek lain. Dengan cara ini, mesej dapat ditangani oleh sebahagian daripada kod. Konsep Asas pengaturcaraan berorientasikan objek adalah seperti berikut:

- i) **Kelas** (*class*) — sebuah kelas mentakrifkan ciri-ciri abstrak bagi sesuatu benda. Ini termasuklah sifat-sifat yang ada padanya dan peranannya.
- ii) **Objek** (*object*) — *instance* bagi suatu kelas.
- iii) **Kaedah** (*method*) — kebolehan bagi sebuah objek.

- iv) **Pewarisan** (*inheritance*) — lazimnya sebuah kelas boleh memiliki "subkelas" yang mengkhususkan kelas tersebut. Semua subkelas ini akan "mewarisi" segala sifat yang ada pada kelasnya.
 - v) **Pengkapsulan** (*encapsulation*) — mengasingkan pelaksanaan (implementasi) daripada antaramuka.
 - vi) **Pengabstrakan** (*abstraction*) — mengurangkan struktur data dan operasi kepada jenis data yang mudah dan hanya mengandungi *properties* yang penting untuk tujuan tertentu.
 - vii) **Polimorfisme** (*polymorphism*) — menggunakan nama yang sama untuk memulakan operasi yang berlainan pada objek yang menggunakan jenis data berbeza.
- d) **Pengaturcaraan Web (Web Programming)**
Pengaturcaraan web merujuk kepada penulisan, *markup* dan pengekodan yang terlibat dalam pembangunan web, ia termasuk kandungan web, pelanggan web, skrip pelayan dan keselamatan rangkaian. Bahasa pengaturcaraan yang biasa digunakan untuk pengaturcaraan web adalah XML, HTML, JavaScript, Perl, PHP dan lain-lain.
- e) **Stored Procedure**
Satu set arahan komputer yang disimpan dalam pangkalan data untuk membuat capaian data dari pangkalan data. *Store procedure* boleh menyimpan logik pengaturcaraan yang pada asalnya dilaksanakan oleh kod aturcara. *Stored procedure* dapat menjimatkan masa dan *memory* di mana proses yang kompleks dan memerlukan pelaksanaan beberapa kenyataan SQL dilaksanakan oleh *store procedure*.

OBJEKTIF

- o Menulis kod dalam struktur yang tersusun dan ringkas supaya mudah untuk dibaca dan difahami, diubahsuai / ditambahbaik dan diselenggara.
- o Menukarkan dokumen SDS kepada kod aturcara dan membuat ujian unit ke atas kod yang dibangunkan. Seterusnya menghasilkan produk iaitu Sistem Aplikasi yang memenuhi kehendak pengguna.

LANGKAH-LANGKAH

Langkah 1: Sediakan Keperluan Pra Pembangunan

- a) Menyediakan keperluan teknikal seperti pendekatan, teknik, *tools*, bahasa pengaturcaraan, pangkalan data dan lain-lain keperluan teknikal yang berkaitan sebelum aktiviti pembangunan dilaksanakan seperti dalam **Penyediaan Pelan Pembangunan Sistem [F1.1] - Langkah 4: Proses Teknikal**.

- b) Pasukan pembangunan perlu memahami dengan mendalam Spesifikasi keperluan sistem dan Spesifikasi Rekabentuk Sistem.

Langkah 2 : Pertimbang Dan Laksanakan Amalan Terbaik Dalam Pengaturcaraan

- a) Amalan baik dalam pengaturcaraan (*Best Programming Practise*) adalah satu set peraturan tidak rasmi yang dihasilkan daripada pengalaman dan pembelajaran oleh komuniti pembangun sistem dari semasa ke semasa. Setiap bahasa pengaturcaraan mempunyai set peraturan yang tersendiri. Teknik dan amalan pengaturcaraan yang baik dapat meningkatkan kualiti dan prestasi perisian yang dihasilkan. Berikut adalah cadangan-cadangan umum praktis pengaturcaraan yang baik:

i) Penggunaan Konvensyen Ulasan (*Commenting Convention*)

Konvensyen ulasan adalah meletakkan ulasan / komen dalam bahasa yang difahami oleh pembaca dalam kod aturcara. Ia menerangkan secara ringkas tentang apa yang kod laksanakan. Komen-komen ini sangat penting supaya pengaturcara dapat menulis perkara-perkara rumit/penting yang telah dilaksanakan dalam kod tersebut. Melalui ulasan / komen, pengaturcara boleh mengetahui kegunaan kod tanpa perlu membaca keseluruhan kod. Berikut adalah contoh maklumat yang direkodkan sebelum sesuatu kod ditulis (*format/syntax* komen ini bergantung kepada Bahasa pengaturcaraan yang digunakan):

- Bahagian atas setiap fail kod sumber (*File Header Comment*)
 - Pengaturcara asal,
 - Tarikh
 - Tujuan
 - Algoritma yang digunakan (merujuk kepada SDS mana)
 - Senarai pengubahsuaian kepada kod aturcara
- Fungsi / Method
 - Purpose of method
 - Argument descriptions
 - Result descriptions
 - Exceptions thrown

Contoh catatan dalam fungsi:

```
/**
 * This method has no use, and it just illustrative. <br>
 * Although it does suggest adding the two parameters together ! <br>
 *
 * @param first The first number to be added
 * @param second The second number to be added
 * @return The sum of the two parameters
```

```

* @throws BadException if something goes very wrong !
*/
public int sumNumbers(int first, int second) throws BadException{
    . . .

```

- *In line*

Kod yang rumit dan kurang jelas, diletakkan komen pada bahagian atas sebelum kod atau sebaris dengan kod yang ditulis, untuk memberi penerangan mudah berkaitan kod yang ditulis.

- *Classes and Interfaces*

- Nama Kelas
- Keterangan berkaitan kelas dan tujuannya
- Versi Semakan
- Nama pengaturcara asal
- Version

Contoh penggunaan *classes* dan *interfaces*:

```

/**
 * MyClass <br>
 *
 * This class is merely for illustrative purposes. <br>
 *
 * Revision History:<br>
 * 1.1 – Added javadoc headers <br>
 * 1.0 - Original release<br>
 *
 * @author T.D.Bishop
 * @version 1.1, 19/04/2000
 */
public class MyClass {
    . . .

```

- Pembolehubah (*Variables*)

Ulasan untuk pembolehubah sepatutnya ringkas sahaja, menerangkan secara ringkas apa kegunaannya.

ii) Penggunaan Konvensyen Pemformatan (*Formatting Convention*)

Pemformatan menjadikan kod yang dihasilkan tersusun, seragam, mudah dibaca dan dicetak. Penggunaan format haruslah konsisten sepanjang kod ditulis. Berikut adalah antara format-format yang di syorkan:

- *Indentation dan Layout*

- Tetapkan saiz yang standard untuk inden. Secara global saiz inden ditetapkan kepada 4 ruang (*4 spaces*) untuk setiap peringkat.
- Baris kod tidak terlalu panjang, elakkan menggunakan *horizontal scroll*. Pecahkan kod-kod yang panjang (*Break up long lines*) kepada yang lebih pendek dan pastikan ia dipecahkan pada titik yang bersesuaian supaya mudah dibaca dan dicetak – maksimum 100 aksara untuk setiap baris.
- *Bracketing* – selaraskan susunan penggunaan '{' dan '}' supaya selari, atau menggunakan *slanting style*, di mana '{' adalah dihujung kod manakala '}' selari dengan permulaan kod.

Contoh penggunaan bracketing:

```
class MyExample {
    public void method(){
        ...
    }
}
```

- Penulisan HTML – tetapkan format untuk *tags* dan *attributes*, seperti menggunakan huruf besar untuk semua *tags* dan huruf kecil untuk *attributes*.
- Penulisan pernyataan SQL – gunakan huruf besar untuk kata kunci (SELECT, DELETE, UPDATE, WHERE, ORDER BY) dan huruf kecil elemen pangkalan data seperti nama *tables*, *columns* dan *views*.
- Susun setiap klausa SQL yang utama pada baris yang berasingan supaya kenyataan ini adalah lebih mudah untuk dibaca dan dipinda.

Penggunaan format haruslah konsisten dalam setiap kod yang ditulis. Penggunaan gaya pengaturcaraan yang bercampur akan menyukarkan proses penyelenggaraan atau penambahbaikan kod.

iii) Penggunaan Konvensyen Penamaan (*Naming Convention*)

Konvensyen penamaan adalah satu set peraturan untuk menentukan pemilihan nama bagi pembolehubah, fungsi, kelas dan lain-lain entiti dalam kod sumber termasuk nama fail dan folder yang terlibat.

- Gunakan nama yang memberi makna, mudah difahami dan sesuai dengan tujuannya. Contoh fungsi Cetak(), melaksanakan fungsi untuk mencetak.
- Penggunaan nama yang baik mampu menggambarkan kandungan/ tujuan entiti tersebut (*Self Describing*) dan mudah dicari.
- Elakkan penggunaan nama-nama yang hanya berbeza pada aksara, contoh cetak dan Cetak tetapi berbeza tujuan.

- Elakkan menggunakan nama yang boleh mengelirukan, contohnya x. Tidak memberi sebarang makna.

Jadual 64 : Contoh Konvensyen Penamaan

Jenis	Konvensyen Penamaan	Contoh
<i>Classes</i>	<ul style="list-style-type: none"> • UpperCamelCase: bermula dengan huruf besar dan pada permulaan setiap perkataan baru 	DaftarAset(); PermohonanAset(); CleverClassName();
<i>Methods</i>	<ul style="list-style-type: none"> • lowerCamelCase: bermula dengan huruf kecil dan huruf besar pada permulaan setiap perkataan baru 	main(); kiraJumlah(); aUsefulMethod();
<i>Attribute / Variables</i>	<ul style="list-style-type: none"> • lowerCamelCase: bermula dengan huruf kecil dan huruf besar pada permulaan setiap perkataan baru • nama-nama pembolehubah tidak harus bermula dengan garis bawah (_) atau tanda dolar (\$) walaupun dibenarkan • nama pembolehubah sepatutnya pendek tetapi memberi makna • penggunaan 1 aksara sebagai pembolehubah perlu dielakkan kecuali pada pembolehubah sementara, untuk kegunaan loop / for yang digunakan sebagai pembolehubah bilangan sahaja. Kebiasaan pembolehubah sementara menggunakan aksara i,j,k dan m untuk int dan c,d dan e untuk char. 	Pembolehubah sementara: int i; char c; myVariable; aClassAttribute; float jumlahMarkah;
<i>Constants / Magic Number</i>	<ul style="list-style-type: none"> • UpperCase • Dipisahkan dengan '_' setiap perkataan 	int MAX_PESERTA = 10;

iv) Konvensyen Pengaturcaraan (*Programming Convention*)

Konvensyen pengaturcaraan menjadikan kod mudah dibaca dan membenarkan pasukan memahami kod yang baru dengan lebih cepat.

Kod Sumber

- Kod yang ditulis perlu mudah dan jelas - elakkan penulisan kod yang panjang, contohnya melebihi 20 baris. Amalkan penggunaan pengaturcaraan berstruktur / bertatacara.

- Elakkan penggunaan nilai *hard-coded / magic numbers* – kod sumber seharusnya tidak menggunakan *hard-coded* untuk merujuk kepada mana-mana parameter seperti *paths, file, host*, alamat IP, URLs, *ports* dan lain-lain. Penggunaannya dalam kod menyebabkan ia sukar untuk dikenal pasti jika perlu diubah. Sebaiknya wujudkan pembolehubah dan ia dikonfigurasi dengan baik.

```
private final int ST = 2;
private final int E = 3;
private final int S = 1;
```

- Mewujudkan *Program Versioning*
- Mewujudkan fungsi / method yang boleh dikongsi
- Direktori simpanan - Simpan fail dalam *folder* secara teratur untuk mrmudahkan carian
- Security - validation, code hard (SQL injection)
- Menyediakan mesej ralat yang menggambarkan ralat sebenar, agar mudah untuk mengesan punca ralat.
- Elakkan penggunaan pernyataan bersarang yang melebihi 3 peringkat (deeply nested control statements).

Pengaturcaraan Pangkalan Data

- Elakkan penggunaan SELECT *, Biasakan menulis dengan jelas kolom-kolum yang dikehendaki.
- Gunakan *stored procedure* untuk menggantikan pernyataan SQL dalam kod sumber bagi meningkatkan prestasi capaian data.
- Melakukan *data validation* pada client semasa data entry. Ini dapat mengelakkan capaian ke pangkalan data berulang kali dengan data yang tidak sah.

Langkah 2 : Bangunkan Sistem

- Menyediakan persekitaran pembangunan (*development environment*):
 - Membina dan memasang perisian, perkakasan (*server*), rangkaian, *tools* dan lain-lain peralatan yang berkaitan.
 - Menguji persekitaran pembangunan supaya memenuhi keperluan pembangunan.
- Mewujudkan dokumentasi *coding standard* (Rujuk Dokumentasi Kod Sumber)
- Memulakan penulisan kod aturcara (Rujuk Amalan Baik dalam Pengaturcaraan)
- Melaksanakan ujian unit bagi setiap komponen sistem dan mendokumenkan keputusan dalam Laporan Ujian (Keterangan lanjut dalam perenggan **Ujian Sistem [F4.3]**)

Langkah 3 : Menyediakan Dokumentasi Kod Sumber

- a) Dokumentasi kod sumber adalah dokumen yang mengandungi senarai kod aturcara yang meliputi struktur direktori dan hierarki fail serta garis panduan yang mengesyorkan perkara berkaitan gaya pengaturcaraan (*style*), konvensyen penamaan, *indentation*, ulasan / komen, pengistiharan pembolehkan, pernyataan SQL dan lain-lain yang perlu dipatuhi oleh semua pengaturcara. **Rujuk D10 Dokumen Kod Sumber.**
- b) Tujuan pengwujudan dokumen ini adalah untuk mewujudkan habit / kebiasaan yang baik dalam gaya penulisan kod dan mewujudkan keseragaman kepada semua pengaturcara.
- c) Kod aturcara yang telah dibangunkan adalah sangat penting untuk didokumentasikan. Ia termasuk fail README yang menyimpan maklumat umum seperti tujuan sistem, URL kepada kod sumber utama (*main source code*) dan lain-lain maklumat asas berkaitan sistem. Fail ini menjadi rujukan pertama kepada pembaca kod.
- d) Pengaturcara sangat digalakkan untuk mengikuti garis panduan ini supaya:
 - i) meningkatkan kebolehbacaan dan kefahaman ke atas kod sumber mereka.
 - ii) memudahkan semakan, penambahbaikan dan penyelenggaraan ke atas perisian.
 - iii) proses pemindahan teknologi (*Transfer Of Technology*) kepada ahli pasukan lain / baru menjadi mudah dan lebih cepat.
 - iv) memudahkan pasukan pengaturcara membuat carian kepada fungsi-fungsi atau kelas dalam kod yang ditulis, seterusnya menggalakkan penggunaan semua kod sedia ada (*reusability*) dan
 - v) sebagai rujukan pada masa depan.
- e) Salah satu cabaran dalam mendokumentasikan kod sumber adalah memastikan bahawa ulasan / komen dikemaskini selari dengan kod aturcara yang ditulis. Kekangan masa menyebabkan pengaturcara hanya fokus kepada kod yang tulis dan sangat sedikit masa diberikan kepada penulisan dokumen kod sumber.
- f) Menjana dokumen secara automatik (*Documentation Generator*). Terdapat *tool* yang membolehkan dokumen ini dijana secara automatik dari kod sumber yang ditulis. Ia dikenali sebagai *documentation generator*.
 - i) *Documentation generator* dijana dari kod sumber seperti ulasan (*comment*) yang ditulis oleh pengaturcara, ini menjadikan dokumen mudah dikemaskini dari masa ke masa selaras dengan kod yang ditulis.
 - ii) Setiap Bahasa pengaturcaraan mempunyai *documentation generator* sendiri. Contoh: Doxygen, NDoc, javadoc, EiffelStudio, Sandcastle, ROBODoc, POD, TwinText, atau Universal Report dan lain-lain.

RUJUKAN

1. General Software Development Standard and Guidelines Version 3.5 - Science Infusion Software Engineering Process Group (SISEPG).
2. Java Language Coding Standard - Sun Microsystem



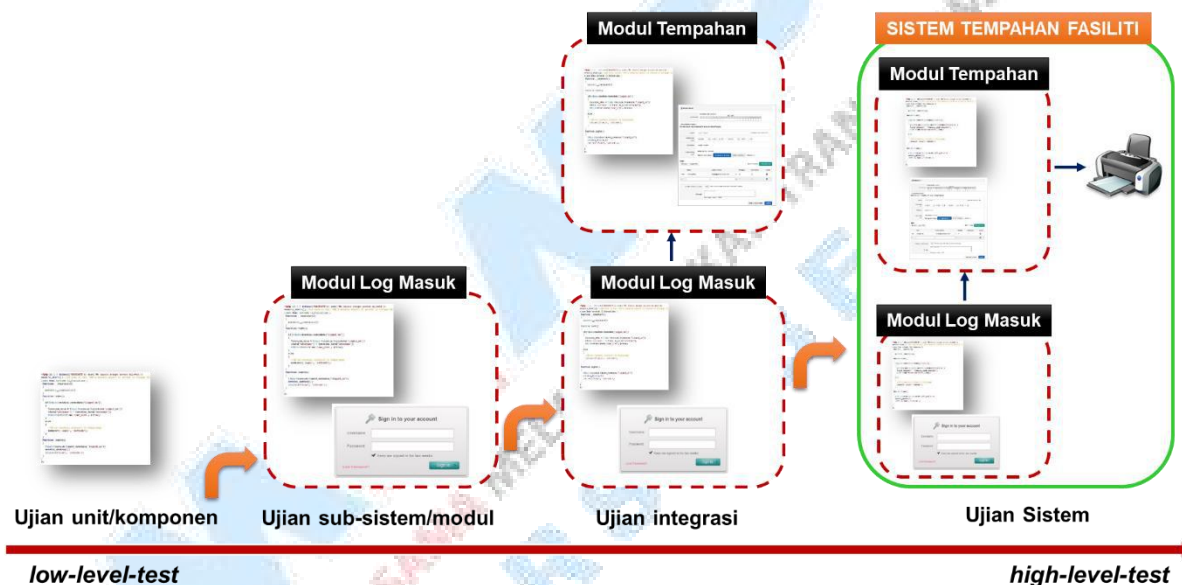
5.7 PENGUJIAN SISTEM [F4.3]

KETERANGAN

Pengujian Sistem merupakan aktiviti verifikasi yang dilakukan terhadap komponen atau sistem (*test object*) untuk memastikan ia dibangunkan berdasarkan kepada **spesifikasi keperluan dan reka bentuk sistem**. Pengujian sistem merangkumi pelbagai peringkat ujian sebelum sistem diuji secara komprehensif di dalam fasa pengujian penerimaan. Semasa pengujian ini dilaksanakan, ralat yang dikesan akan diperbetulkan dan unit/komponen/modul yang berkaitan akan diuji semula sehingga ralat berjaya diperbaiki.

Jenis-jenis pengujian yang dijalankan adalah **pengujian keperluan fungsian, pengujian keperluan bukan fungsian (kualiti)** serta **verifikasi terhadap ralat yang telah dibaiki**. Pengujian ini akan **dilaksanakan oleh Pasukan Pembangun Sistem**.

Peringkat-peringkat ujian yang dilaksanakan mengikut turutan adalah seperti rajah di bawah.



Rajah 85 : Peringkat Pengujian Sistem

Terdapat dua jenis pendekatan pengujian yang digunakan semasa melaksanakan pengujian ke atas sistem aplikasi iaitu *big bang testing* dan *incremental testing*. *Big bang testing* merupakan pengujian yang dilaksanakan terhadap sistem aplikasi yang telah lengkap dibangunkan, manakala *incremental testing* merupakan satu bentuk ujian modul di mana modul yang akan diuji digabungkan dengan modul yang sudah diuji.

Pendekatan bagi aktiviti pengujian ini dikenali sebagai *incremental testing* iaitu **pengujian dilakukan secara berperingkat** bermula daripada pengujian unit/komponen terkecil sistem aplikasi seperti fungsi, kelas, prosedur dan antara muka (**Ujian unit/komponen**); diikuti dengan ujian modul (**Ujian sub-sistem/modul**); seterusnya menguji dua atau lebih modul/sistem/elemen perkakasan yang disepadukan (**Ujian integrasi**); dan akhirnya semua modul yang terlibat diuji secara menyeluruh (**Ujian sistem**).

Pendekatan ini menerangkan bahawa **ujian tahap rendah (*low-level-test*)** perlu dilaksanakan terlebih dahulu bertujuan untuk mengesahkan bahawa ujian segmen kod sumber telah dilaksanakan dengan betul sebelum ke peringkat ujian berikutnya iaitu, **ujian tahap tinggi (*high-level test*)** yang bertujuan mengesahkan fungsi-fungsi utama sistem aplikasi.

Bagi setiap peringkat pengujian, terdapat empat (4) elemen yang perlu ditetapkan sebelum pelaksanaan ujian iaitu:

- a) *Entry Criteria* boleh merujuk kepada dokumen, status/ aktiviti serta tahap pencapaian atau pengukuran yang menjadi pra-syarat untuk melaksanakan sesuatu peringkat pengujian.
- b) Aktor merujuk kepada individu/kumpulan yang terlibat dengan sesuatu ujian.
- c) Aktiviti ialah aktiviti yang perlu dijalankan semasa ujian.
- d) *Exit Criteria* pula merujuk kepada dokumen, status/aktiviti serta tahap pencapaian atau pengukuran yang menjadi syarat untuk menamatkan sesuatu peringkat pengujian

OBJEKTIF

Menilai kualiti keseluruhan sistem selepas pembangunan bagi memastikan sistem aplikasi yang dibangunkan sedia untuk diuji di peringkat pengujian penerimaan pengguna.

LANGKAH-LANGKAH

Langkah 1 : Laksana Ujian Unit/Komponen

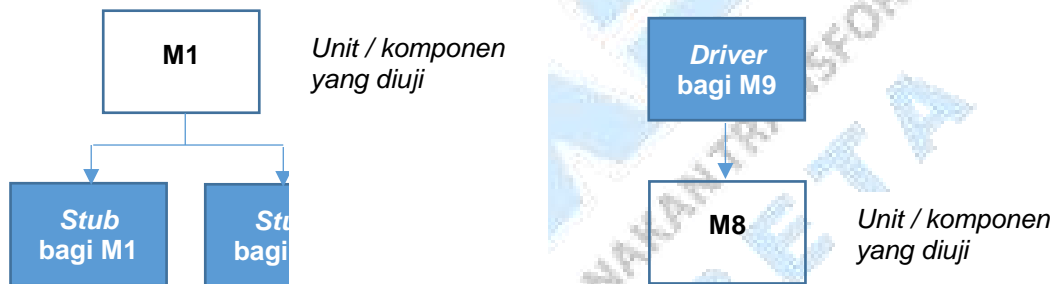
- a) Tetapkan elemen *Entry Criteria*, aktor, aktiviti dan *Exit Criteria* bagi Ujian Unit/ Komponen seperti jadual di bawah:

Jadual 65 : *Entry Criteria* dan *Exit Criteria* bagi Ujian Unit/Komponen

Entry Criteria	i) Dokumen SRS dan SDS telah disahkan ii) Kod sumber bagi unit/komponen telah selesai dibangunkan/ telah diperbaiki bagi tujuan <i>re-test</i> iii) Unit/komponen untuk diuji serta <i>stubs/drivers</i> yang diperlukan telah disediakan dalam persekitaran pembangunan
Aktor	Pasukan Pembangun Sistem
Aktiviti	i) Pasukan Pembangun Sistem melaksanakan ujian unit/komponen ii) Pembetulan kod sumber dilakukan bagi ujian yang gagal. iii) Pengujian semula dilaksanakan bagi ujian yang gagal

Exit Criteria	<p>i) Senarai semak hasil ujian menunjukkan unit/komponen berjaya melaksanakan fungsian atau bukan fungsian seperti yang telah ditetapkan</p> <p>ii) Ralat telah berjaya dibaiki</p>
----------------------	--

- b) Kenalpasti unit/komponen adalah bahagian terkecil di dalam sistem aplikasi yang boleh diuji (*testable*) seperti fungsi, kelas, prosedur dan antara muka.
- c) Sediakan *stubs* dan *drivers* yang diperlukan di dalam ujian unit/ komponen bagi menggantikan unit/ komponen/ modul yang perlu semasa berinteraksi dengan unit/komponen yang diuji. *Stubs* adalah simulasi (*dummy module*) bagi menggantikan unit/komponen selepas (*subordinate/ lower level*) unit/komponen yang diuji. *Drivers* pula adalah simulasi bagi menggantikan unit/komponen sebelum (*upper level*) unit/komponen yang diuji. Rajah di bawah menunjukkan gambaran penggunaan *stubs* dan *drivers* di dalam ujian unit/komponen.



Rajah 86 : Ujian Komponen - Stubs dan Driv

- d) Laksana ujian unit/komponen bagi memastikan kod program memenuhi fungsi unit/komponen dan mengenal pasti ralat di dalam unit/komponen berkenaan. Ujian ini juga hendaklah merangkumi ujian fungsian dan ujian bukan fungsian.
- e) Rujuk asas ujian (*test basis*) yang berkaitan iaitu spesifikasi keperluan unit/komponen, dokumen reka bentuk terperinci seperti SRS atau SDS dan kod sumber program semasa melaksanakan ujian bagi memastikan ianya memenuhi keperluan dan reka bentuk unit/komponen yang telah ditetapkan.
- f) Guna kombinasi teknik pengujian yang bersesuaian seperti teknik *white-box testing* dan juga *black box testing*. *White-box testing* merupakan teknik pengujian yang terperinci yang dilakukan terhadap logik dalam dan struktur kod. Teknik ini memerlukan penguji mempunyai pengetahuan penuh tentang kod sumber. Manakala *black box testing* pula merupakan teknik pengujian berdasarkan kepada spesifikasi sistem aplikasi. Pengujian ini dilaksanakan dengan memasukkan *input* dan *output* akan diperiksa sama ada memenuhi fungsi yang telah ditetapkan atau tidak.

- g) Uji pengendalian ralat (*error handling*) sekiranya input yang tidak sah diberikan.
- h) Baiki ralat yang ditemui dan uji semula unit/komponen berkenaan. Pengujian akan dilaksanakan sehingga unit/ komponen memenuhi *Exit Criteria* yang telah ditetapkan untuk ke peringkat ujian seterusnya iaitu Ujian Sub-Sistem/Modul.

Langkah 2 : Laksana Ujian Sub-Sistem/Modul

- a) Tetapkan elemen *Entry Criteria*, aktor, aktiviti dan *Exit Criteria* bagi Ujian Sub-Sistem/Modul seperti jadual di bawah:

Jadual 66 : *Entry Criteria* dan *Exit Criteria* bagi Ujian Sub-Sistem/Modul

<i>Entry Criteria</i>	<ul style="list-style-type: none"> i) Ujian unit/komponen telah dilaksanakan dengan sempurna ii) <i>Defects/bugs</i> yang dilaporkan dalam ujian unit/komponen telah diperbaiki dan diuji semula iii) <i>Test script</i> dan <i>test data</i> bagi ujian sub-sistem/modul telah disediakan oleh Pasukan Pembangun Sistem
Aktor	Pasukan Pembangun Sistem
Aktiviti	<ul style="list-style-type: none"> i) Pasukan Pembangun Sistem melaksanakan ujian sub-sistem/modul ii) Pembedulan dilakukan bagi ujian sub-sistem/modul yang gagal iii) Pengujian semula dilaksanakan bagi ujian yang gagal
<i>Exit Criteria</i>	<ul style="list-style-type: none"> i) <i>Test script</i> telah diuji ii) Senarai semak hasil ujian menunjukkan sub-sistem/modul berjaya melaksanakan fungsian atau bukan fungsian seperti yang telah ditetapkan iii) Ralat telah berjaya dibaiki

- b) Sediakan *test data*, bangunkan *test script* berdasarkan pertimbangan di bawah bagi memastikan ujian sub-sistem/modul dilaksanakan dengan berkesan:
- i) *Module Interface Test*: bertujuan untuk menguji maklumat yang masuk dan keluar daripada modul;
 - ii) *Local data structures*: Struktur data tempatan diperiksa untuk memastikan data yang disimpan secara sementara dapat mengekalkan integritinya semasa pelaksanaan algoritma.
 - iii) *Boundary conditions*: bertujuan untuk memastikan modul beroperasi dengan betul di sempadan yang ditetapkan (*to limit or restrict processing*).

- iv) *Independent paths*: bertujuan untuk menguji semua *independent paths* yang melalui struktur kawalan bagi memastikan bahawa semua kenyataan dalam modul telah dilaksanakan sekurang-kurangnya sekali.
 - v) *Error handling paths*: untuk memastikan ralat ditangani dengan betul dan *error handling paths* yang dikenalpasti dapat digunakan selepas melepasi beberapa siri ujian.
- c) Laksanakan ujian sub-sistem/modul untuk menguji interaksi di antara unit/komponen dalam modul bagi memastikan ianya dapat berfungsi secara kolektif bagi fungsi, kelas, prosedur dan antara muka yang terlibat di dalam sesebuah modul.
- d) Uji setiap modul yang terdapat dalam sistem aplikasi secara berasingan sebelum ianya diuji secara bersepadu di peringkat ujian integrasi.

Langkah 3 : Laksana Ujian Integrasi

- a) Tetapkan elemen *Entry Criteria*, aktor, aktiviti dan *Exit Criteria* bagi Ujian Integrasi seperti jadual di bawah:

Jadual 67 : *Entry Criteria* dan *Exit Criteria* bag Ujian Integrasi

<i>Entry Criteria</i>	<ul style="list-style-type: none"> i) Ujian sub-sistem/modul telah dilaksanakan dengan sempurna ii) <i>Defects/bugs</i> yang dilaporkan dalam ujian sub-sistem/modul telah diperbaiki dan diuji semula iii) Sistem/antara muka sistem luar telah sedia diintegrasikan dengan sistem yang diuji (<i>SUT</i>). iv) <i>Test script</i> dan <i>test data</i> bagi ujian integrasi telah disediakan oleh Pasukan Pembangun Sistem v) Penguji Integrasi telah diberikan penerangan
Aktor	<ul style="list-style-type: none"> i) Pasukan Pembangun Sistem ii) Penguji Integrasi
Aktiviti	<ul style="list-style-type: none"> i) Penguji integrasi melaksanakan ujian integrasi ii) Penguji integrasi merekodkan hasil ujian iii) Pembedetulan dilakukan bagi ujian integrasi yang gagal iv) Pengujian semula dilaksanakan bagi ujian yang gagal
<i>Exit Criteria</i>	<ul style="list-style-type: none"> i) <i>Test script</i> telah diuji ii) Senarai semak hasil ujian ujian menunjukkan integrasi berjaya melaksanakan fungsian atau bukan fungsian seperti yang telah ditetapkan iii) Ralat telah berjaya dibaiki

- b) Sediakan *test data*, bangun *test script* bagi memastikan ujian sub-sistem/modul dilaksanakan dengan berkesan. Objek yang akan dinilai semasa ujian integrasi adalah sub-sistem, pangkalan data, infrastruktur, antara muka serta konfigurasi sistem. Bagi tujuan ini, Pasukan Pembangun Sistem hendaklah mempunyai kefahaman yang jelas terhadap reka bentuk dan keperluan integrasi sebelum ujian ini dilaksanakan.
- c) Laksanakan ujian integrasi yang merangkumi ujian fungsian dan ujian bukan fungsian bagi menguji perkara berikut:
- i) interaksi di antara sub-sistem/modul dalam sistem aplikasi;
 - ii) interaksi dalaman di antara sistem operasi, fail sistem dan API perkakasan sistem; DAN
 - iii) integrasi di antara sistem yang dibangunkan dengan sistem luaran/antara muka luaran (sekiranya ada).
- d) Rujuk asas ujian (test basis) yang berkaitan iaitu reka bentuk dan arkitektur sistem serta dokumen D02 Spesifikasi Keperluan Bisnes.
- e) Laksanakan ujian integrasi dalam beberapa sesi mengikut keperluan integrasi yang wujud dalam sistem aplikasi sehingga memenuhi *Exit Criteria* yang telah ditetapkan untuk ke peringkat ujian seterusnya iaitu Ujian Sistem.

Langkah 4 : Laksana Ujian Sistem

- a) Tetapkan elemen *Entry Criteria*, aktor, aktiviti dan *Exit Criteria* bagi Ujian Sistem seperti jadual di bawah:

Jadual 68 : Entry Criteria dan Exit Criteria bag Ujian Sistem

Entry Criteria	<ol style="list-style-type: none"> i) Ujian integrasi telah dilaksanakan dengan sempurna ii) <i>Defects/bugs</i> yang dilaporkan dalam ujian integrasi telah diperbaiki. Tiada lagi <i>defects/bugs</i> bagi: <ul style="list-style-type: none"> • <i>Priority High</i> atau <i>Medium</i>; dan • <i>Severity Blocking, Critical, dan Major</i> iii) <i>SUT</i> telah disediakan di dalam persekitaran pengujian. iv) <i>Test script</i> dan <i>test data</i> bagi ujian sistem telah disediakan oleh Pasukan Pembangun Sistem
Aktor	Pasukan Pembangun Sistem
Aktiviti	<ol style="list-style-type: none"> i) Pasukan Pembangun Sistem melaksanakan ujian sistem ii) Pasukan Pembangun Sistem merekodkan hasil ujian

	iii) Pembetulan dilakukan bagi ujian yang gagal iv) Pengujian semula dilaksanakan bagi ujian yang gagal
Exit Criteria	i) <i>Test script</i> telah diuji ii) Hasil ujian menunjukkan sistem berjaya melaksanakan fungsian atau bukan fungsian seperti yang telah ditetapkan. iii) Sistem telah berjaya memenuhi keperluan bisnes dan keperluan fungsian iv) Ralat telah berjaya dibaiki

- b) Sediakan *test data*, bangunkan *test script* bagi memastikan ujian sistem dilaksanakan dengan berkesan.
- c) Laksanakan ujian sistem untuk menentukan keseluruhan sistem aplikasi telah memenuhi spesifikasi yang ditetapkan sebelum ke fasa pengujian penerimaan. Walaubagaimanapun, ujian ini sebaik-baiknya dilaksanakan di dalam persekitaran yang hampir sama dengan persekitaran sebenar (*staging/pre-production*).
- d) Rujuk dokumen spesifikasi keperluan, proses bisnes, *Use Case*, laporan analisis risiko, interaksi sistem dengan sistem operasi dan *system resources* semasa melaksanakan ujian sistem.
- e) Laksanakan ujian sistem sehingga memenuhi *Exit Criteria* yang telah ditetapkan.

Langkah 5 : Sediakan Laporan Ujian Sistem

- a) Sediakan Laporan Ujian Sistem sebagai pengesahan aktiviti Pengujian Sistem telah dilaksanakan sepenuhnya. Laporan Ujian Sistem menentukan tahap kesiapan sistem dan merupakan *Entry Criteria* kepada Ujian Penerimaan Pengguna. Format Laporan Ujian Sistem adalah seperti D11 Laporan Ujian Sistem.
- b) Laporan Ujian Sistem hendaklah **mempunyai sekurang-kurangnya elemen** berikut:

Jadual 69 : Format Laporan Ujian Sistem

Bil.	Item	Keterangan
1	Pengenalan	Penerangan berkaitan tujuan pelaporan dan skop pelaporan. Skop pelaporan boleh berdasarkan sistem atau modul (sekiranya sistem adalah berskala besar dan mempunyai kompleksiti yang tinggi).

2	Aktiviti Pengujian	<p>Serahan</p> <p>Menerangkan bilangan modul/submodul yang terlibat, status lulus dan gagal, serta keterangan yang berkaitan dengannya.</p> <table border="1" data-bbox="528 365 1380 674"> <thead> <tr> <th>Bil</th> <th>Modul & Sub Modul</th> <th>Status Pengujian (Lulus/Gagal/KIV)</th> <th>Keterangan</th> </tr> </thead> <tbody> <tr> <td rowspan="3">1</td> <td>Modul Personel</td> <td>Lulus</td> <td>Selesai</td> </tr> <tr> <td>Modul Gaji</td> <td>KIV</td> <td>2 modul masih KIV untuk pengujian</td> </tr> <tr> <td>Modul Pelaporan</td> <td>Lulus</td> <td>Selesai</td> </tr> </tbody> </table> <p>Rumusan Serahan</p> <p>Ringkasan kepada aktiviti pengujian sistem yang telah dilakukan berdasarkan sistem atau modul yang diuji.</p> <p>Sekiranya rumusan adalah diperingkat sistem, maka bilangan yang digunakan adalah bilangan modul. Bagi peringkat modul, bilangan yang digunakan adalah bilangan submodul.</p> <p>Contoh:</p> <table border="1" data-bbox="528 1115 1380 1720"> <thead> <tr> <th>Bil</th> <th>Modul</th> <th>Bil Lulus</th> <th>Bil Gagal/KIV</th> <th>Keterangan</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Modul Personel</td> <td>5</td> <td>0</td> <td>Selesai</td> </tr> <tr> <td>2</td> <td>Modul Gaji</td> <td>5</td> <td>2</td> <td>2 submodul KIV kerana terdapat isu <i>compliance</i></td> </tr> <tr> <td>3</td> <td>Modul Pelaporan</td> <td>5</td> <td>1</td> <td>Submodul <i>Business Intelligence</i> tidak diuji untuk <i>unstructured data</i></td> </tr> <tr> <td>4</td> <td>Jumlah</td> <td>15</td> <td>3</td> <td></td> </tr> <tr> <td>5</td> <td>Peratus Lulus/Gagal/KIV</td> <td>83.33</td> <td>16.67</td> <td></td> </tr> </tbody> </table>	Bil	Modul & Sub Modul	Status Pengujian (Lulus/Gagal/KIV)	Keterangan	1	Modul Personel	Lulus	Selesai	Modul Gaji	KIV	2 modul masih KIV untuk pengujian	Modul Pelaporan	Lulus	Selesai	Bil	Modul	Bil Lulus	Bil Gagal/KIV	Keterangan	1	Modul Personel	5	0	Selesai	2	Modul Gaji	5	2	2 submodul KIV kerana terdapat isu <i>compliance</i>	3	Modul Pelaporan	5	1	Submodul <i>Business Intelligence</i> tidak diuji untuk <i>unstructured data</i>	4	Jumlah	15	3		5	Peratus Lulus/Gagal/KIV	83.33	16.67	
Bil	Modul & Sub Modul	Status Pengujian (Lulus/Gagal/KIV)	Keterangan																																											
1	Modul Personel	Lulus	Selesai																																											
	Modul Gaji	KIV	2 modul masih KIV untuk pengujian																																											
	Modul Pelaporan	Lulus	Selesai																																											
Bil	Modul	Bil Lulus	Bil Gagal/KIV	Keterangan																																										
1	Modul Personel	5	0	Selesai																																										
2	Modul Gaji	5	2	2 submodul KIV kerana terdapat isu <i>compliance</i>																																										
3	Modul Pelaporan	5	1	Submodul <i>Business Intelligence</i> tidak diuji untuk <i>unstructured data</i>																																										
4	Jumlah	15	3																																											
5	Peratus Lulus/Gagal/KIV	83.33	16.67																																											
3	Dokumen Sokongan	<p>Menyatakan dokumen sokongan yang dirujuk berkaitan dengan Laporan Ujian Sistem yang boleh digunakan oleh pemilik sistem untuk mengesahkan pelaksanaan Ujian Sistem.</p> <p>Contoh:</p> <p>Senarai nama penguji, modul/sub-modul yang terlibat dan tarikh selesai pengujian.</p>																																												

RUJUKAN

1. Software Engineering: A Practitioner's Approach Eighth Edition, Roger S. Pressman, Ph.D, Bruce R. Maxim, Ph.D, Mc Graw Hill Education.
2. Software Quality Assurance From theory to implementation, G. Daniel
3. Certified Tester Foundation Level Syllabus Version 2018
4. Certified Tester Advanced Level Syllabus Test Analyst Version 2012






BAB:06

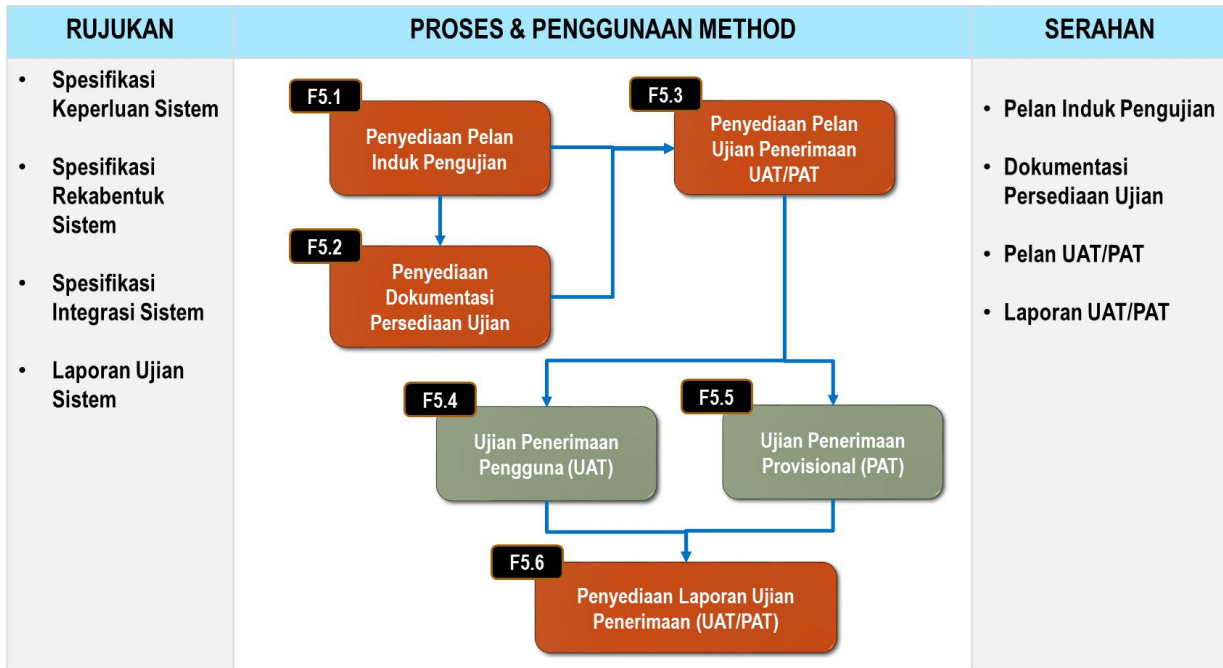
FASA PENGUJIAN PENERIMAAN

Bab ini menerangkan berkenaan dengan aktiviti-aktiviti pengujian penerimaan sistem aplikasi yang merangkumi penyediaan pelan-pelan pengujian, Ujian Penerimaan Pengguna (UAT), Ujian Penerimaan Provisional (PAT), dokumentasi persediaan ujian dan Laporan Ujian Penerimaan.



6 FASA PENGUJIAN PENERIMAAN

6.1 GAMBARAN KESELURUHAN



Rajah 87 : Gambaran Keseluruhan Fasa V – Pengujian Penerimaan

6.2 PENGENALAN

Pengujian penerimaan sistem aplikasi merupakan penentu keyakinan bahawa sistem yang dibangunkan telah memenuhi keperluan bisnes dan keperluan pengguna yang ditetapkan. Pengujian juga bertujuan untuk memastikan bahawa fungsian sistem yang dibangunkan sedia digunakan. Sebarang kesilapan yang ditemui perlu ditambahbaik, diperbaiki dan diuji semula.

Pengujian hendaklah dirancang dengan teliti berdasarkan tempoh, kos dan keperluan sumber yang disediakan agar sistem yang dibangunkan mencapai tahap kualiti yang ditetapkan. Hasil keputusan ujian yang diperolehi melalui pelaksanaan pengujian yang teratur dan pematuhan kepada amalan terbaik akan menjadi pertimbangan bagi pemegang taruh dalam memutuskan pelaksanaan sistem.

Pengujian akan dilaksanakan dengan melibatkan pengguna sistem termasuk SME, pemilik proses dan pengguna akhir. Ekspektasi pengguna terhadap sistem akan dinilai melalui dua (2) aktiviti utama iaitu:

- Ujian Penerimaan Pengguna (UAT), ianya menilai dari aspek fungsian; dan

- b) Ujian Penerimaan Provisional (PAT), ianya menilai dari aspek fungsian dan bukan fungsian

Dokumen rujukan untuk Fasa Pengujian Penerimaan adalah seperti berikut:

- a) D03 Spesifikasi Keperluan Sistem.
- b) D04 Spesifikasi Rekabentuk Sistem.
- c) D08 Spesifikasi Integrasi Sistem.
- d) D11 Laporan Ujian Sistem.

Dokumen serahan untuk Fasa Pengujian Penerimaan adalah seperti berikut:

- a) D12 Pelan Induk Pengujian (*Master Test Plan*).
- b) D13 Pelan Ujian Penerimaan Pengguna (UAT) / Ujian Penerimaan Pengguna Provisional (PAT).
- c) D14 Laporan Ujian Penerimaan (UAT & PAT)

6.3 PENGLIBATAN PEMEGANG TARUH

Pemegang taruh utama yang patut terlibat dalam fasa pengujian adalah Pengurus / Ketua Ujian, Penguji Sistem Aplikasi (termasuk Penguji Integrasi, Penguji Migrasi sekiranya berkaitan), Pasukan Pembangun Sistem Aplikasi, SME, Pemilik Proses dan Pengguna Akhir.

Cadangan penglibatan kategori pemegang taruh adalah seperti berikut:

- a) Pengurus Ujian – bertanggungjawab sepenuhnya mengurus dan menyediakan laporan bagi aktiviti pengujian penerimaan.
- b) Ketua Ujian – bertanggungjawab mengendalikan pasukan pengujian.
- c) Pasukan Penguji – menyediakan dokumentasi berkaitan pengujian serta mengendali atau melaksanakan aktiviti pengujian.
- d) Pasukan Pembangun Sistem Aplikasi – menjalankan tindakan pembetulan ke atas sistem aplikasi berdasarkan laporan ujian yang dikeluarkan oleh Penguji Sistem Aplikasi.
- e) SME dan Pemilik Proses – membantu pelaksanaan aktiviti pengujian penerimaan. Pemilik proses juga boleh terlibat di dalam aktiviti pengujian penerimaan.
- f) Pengguna Akhir – terlibat dalam pengujian dan memberi maklum balas terhadap ujian penerimaan yang dilaksanakan.

6.4 FAKTOR KEJAYAAN

Untuk memastikan aktiviti dalam fasa pengujian berjaya dilaksanakan, faktor-faktor yang perlu diberi perhatian sebelum dan semasa aktiviti pengujian dilaksanakan adalah seperti berikut:

- a) Spesifikasi Keperluan Bisnes, Spesifikasi Keperluan Sistem dan Spesifikasi Rekabentuk Sistem yang didokumenkan adalah lengkap dan telah dipersetujui oleh pengguna.
- b) Ujian sistem (ujian unit/ komponen, ujian sub-sistem/ modul dan ujian integrasi sistem) telah dijalankan dengan sempurna. Sistem bebas daripada ralat dengan tahap severity tinggi yang boleh menggagalkan fungsi utamanya.
- c) Pengurus / Ketua Ujian berpengalaman serta berkelayakan dalam merancang dan mengendalikan ujian.
- d) Persediaan terperinci bagi ujian penerimaan seperti jadual pelaksanaan ujian, undangan penguji serta kesediaan persekitaran pengujian.
- e) Komitmen daripada semua peringkat pemegang taruh diperlukan semasa pengujian dilaksanakan.

Selain daripada faktor di atas, penggunaan *tools* yang bersesuaian di dalam pengurusan, pelaksanaan dan kawalan pengujian juga memberi nilai tambah di dalam kejayaan pengujian.

6.4 PENYEDIAAN PELAN INDUK PENGUJIAN [F5.1]

KETERANGAN

Pelan Induk Pengujian (*Master Test Plan*) adalah pelan utama yang perlu disediakan sebagai rujukan perancangan dan pengurusan aktiviti pengujian sistem secara menyeluruh. Pelan ini mengandungi strategi pengujian, rekabentuk pengujian, jenis pengujian, senarai serahan ujian dan jadual pelaksanaan pengujian. Pelan Induk Pengujian hendaklah dibangunkan sebelum fasa pengujian.

OBJEKTIF

Menghasilkan Pelan Induk Pengujian bagi merancang dan mengurus aktiviti pengujian sistem secara menyeluruh.

LANGKAH-LANGKAH

Langkah 1 : Terangkan Pengenalan Projek

- a) Pelan Induk Pengujian mempunyai pengenalan projek yang akan memberi gambaran mengenai projek secara keseluruhannya. Kerangka pengenalan projek adalah:
 - i) Tujuan projek.
 - ii) Skop projek.
 - iii) Senarai Pemegang Taruh (Stakeholder).
 - iv) Sumber rujukan.
 - v) Glosari/ definisi berkaitan dengan projek.
- b) Sebahagian contoh pengenalan projek ini boleh dirujuk di Langkah 1, Penyediaan Pelan Pembangunan Sistem [F1.1].

Langkah 2 : Terangkan Konteks Pengujian

- a) Konteks ujian mengandungi perincian skop dan item yang akan diuji serta kekangan yang dijangka dihadapi semasa aktiviti ujian. Kerangka bagi konteks ujian adalah seperti berikut:

- i) **Item ujian**

Item yang akan diuji perlu disenaraikan seperti contoh berikut :

Modul-modul yang terlibat pengujian ini adalah :

Modul 1 : Pendaftaran Pengguna

Modul 2 : Tempahan bilik mesyuarat

Modul 3 : Laporan

Modul 4 : Pentadbir sistem

Modul 5 : ...

ii) **Skop ujian**

Skop ujian bertujuan untuk menetapkan perimeter ujian termasuk :

- ciri-ciri (fungsi dan kualiti/ bukan fungsi) yang akan diuji ke atas item ujian
- ciri-ciri yang tidak akan diuji ke atas item ujian

iii) **Kekangan**

Kekangan adalah sekatan atau halangan yang dijangka akan berlaku semasa pengujian dilaksanakan. Kekangan ini boleh mengganggu perjalanan proses pengujian dan berpotensi untuk memberi risiko ke atas sistem aplikasi. Kekangan dengan kebarangkalian yang tinggi untuk berlaku perlu dinyatakan di dalam dokumen ini.

Langkah 3 : Nyatakan Komunikasi Pengujian

- a) Komunikasi ujian akan menjelaskan peranan, tindakan dan medium komunikasi bagi sesuatu situasi yang berlaku di dalam aktiviti pengujian. Komunikasi ujian boleh dinyatakan dalam bentuk jadual seperti contoh berikut :

Jadual 70 : Komunikasi Pengujian

SITUASI	PERANAN	TINDAKAN	MEDIUM KOMUNIKASI
Semakan dan pengesahan Pelan Ujian UAT	Pengurus Ujian / Penguji / Pemilik Sistem	Semakan melalui sesi <i>walkthrough</i>	<ul style="list-style-type: none"> • Mesyuarat / Perbincangan • Emel • Surat • Telefon
Penemuan ralat dalam ujian	Penguji	Melaporkan ralat	<ul style="list-style-type: none"> • Borang laporan insiden • <i>Testing tool / system</i>
	Pasukan pembangun sistem	Memperbaiki ralat yang dilaporkan	Kemaskini status ralat di dalam : <ul style="list-style-type: none"> • Borang laporan insiden • <i>Testing tool / system</i>

Langkah 4 : Lengkapkan Daftar Risiko (*Risk Register*) Ujian

Rujuk item (b) dan (c), Langkah 3, **Penyediaan Pelan Pembangunan Sistem [F1.1]** untuk panduan.

Langkah 5 : Terangkan Strategi Ujian

Strategi ujian menjelaskan bagaimana aktiviti pengujian akan dilaksanakan. Strategi ujian diperincikan di dalam kerangka berikut:

a) Peringkat Ujian

Sub-proses ujian menerangkan tentang peringkat pengujian serta jenis-jenis ujian yang terlibat seperti Jadual di bawah :

Jadual 71 : Contoh Peringkat dan Jenis-jenis Ujian

PERINGKAT PENGUJIAN	JENIS UJIAN
Ujian penerimaan pengguna (UAT)	<ul style="list-style-type: none"> • Ujian fungsian
Ujian migrasi sistem (jika ada)	<ul style="list-style-type: none"> • Ujian migrasi
Ujian integrasi sistem (jika ada)	<ul style="list-style-type: none"> • Ujian integrasi
Ujian penerimaan sementara (PAT)	<ul style="list-style-type: none"> • Ujian fungsian • Ujian prestasi • Ujian bebanan • Ujian tekanan • Ujian keselamatan
Ujian penerimaan Akhir (FAT)	<ul style="list-style-type: none"> • Pemantauan pelaksanaan sistem di dalam persekitaran sebenar bagi tempoh tertentu

b) Senarai Dokumen Serahan Ujian

Dokumen serahan ujian adalah sebarang dokumen atau laporan yang dihasilkan dan perlu diserahkan kepada pemilik sistem semasa fasa pengujian. Berikut adalah contoh dokumen serahan ujian :

- i) Dokumentasi Persediaan Ujian (rujuk kepada [F5.2])
- ii) Pelan Ujian UAT/PAT
- iii) Laporan Ujian Penerimaan UAT/PAT
- iv) Laporan Penamatan Ujian

c) Entry Criteria Dan Exit Criteria

Entry criteria boleh merujuk kepada dokumen, status/ aktiviti serta tahap pencapaian atau pengukuran yang menjadi pra-syarat untuk melaksanakan sesuatu peringkat pengujian. *Exit criteria* pula merujuk kepada dokumen, status/ aktiviti serta tahap pencapaian atau pengukuran yang menjadi syarat untuk menamatkan sesuatu peringkat pengujian

Entry dan *Exit Criteria* boleh diperincikan di dalam Pelan Induk Pengujian atau Pelan Ujian UAT/PAT mengikut kesesuaian. *Entry criteria* dan *Exit criteria* yang ditetapkan hendaklah dibincang dan dipersetujui bersama oleh pemilik projek dan pengurus / ketua ujian.

d) Metrik Pengukuran

Senaraikan metrik pengukuran yang digunakan semasa pengujian.

Contoh metrik pengukuran :

- i) Bilangan dan peratus kes ujian yang gagal
- ii) Bilangan dan peratus kes ujian yang lulus
- iii) Purata masa tindak balas

e) Data Ujian / Simulasi

Data ujian yang akan digunakan perlu dijelaskan sama ada menggunakan data ujian/ simulasi atau data sebenar. Data ujian ini perlu dipersetujui oleh pemilik sistem aplikasi dan pasukan penguji sebelum digunakan.

f) Persekitaran Pengujian

Persekitaran pengujian perlu dinyatakan sama ada persekitaran pembangunan, persekitaran *staging* atau persekitaran produksi . Persekitaran pengujian ini merangkumi infrastruktur (contoh : rangkaian), perkakasan (contoh : pelayan web, pelayan pangkalan data, storan, sistem operasi, pengimbas, mesin pencetak), perisian (contoh : DBMS, JAVA, Oracle) dan *Tools* (contoh : JIRA, Selenium, JMeter) yang digunakan semasa pengujian.

g) Penetapan Tahap Severity

Severity bermaksud implikasi terhadap kegagalan fungsian atau penemuan ralat semasa ujian. Tahap *severity* boleh dijadikan sebagai salah satu faktor dalam menentukan keutamaan (priority) tindakan pembetulan. Merujuk kepada contoh di bawah, tahap *severity* boleh dikategorikan secara numerik atau pernyataan kelas.

Jadual 72 : Contoh Tahap Severity Hasil Ujian

Tahap Severity	Keterangan
1 / Tinggi	Ralat kritikal yang menyebabkan kegagalan fungsi sistem seperti kehilangan data atau kegagalan fungsi utama (<i>blocker/showstopper</i>).
2 / Sederhana	Ralat yang kritikal tetapi tidak melibatkan kehilangan data dan kegagalan fungsi sistem seperti output yang tidak sepadan atau laporan yang tidak tepat.
3 / Rendah	Ralat yang tidak menjejaskan fungsi sistem seperti kesilapan ejaan dan label.

Langkah 6 : Tetapkan Jadual Perancangan Pengujian

Jadual perancangan yang terperinci bagi aktiviti pengujian perlu disediakan. Perancangan perlulah mengambilkira kebergantungan dan tarikh sasaran serahan. Contoh jadual perancangan pengujian seperti berikut :

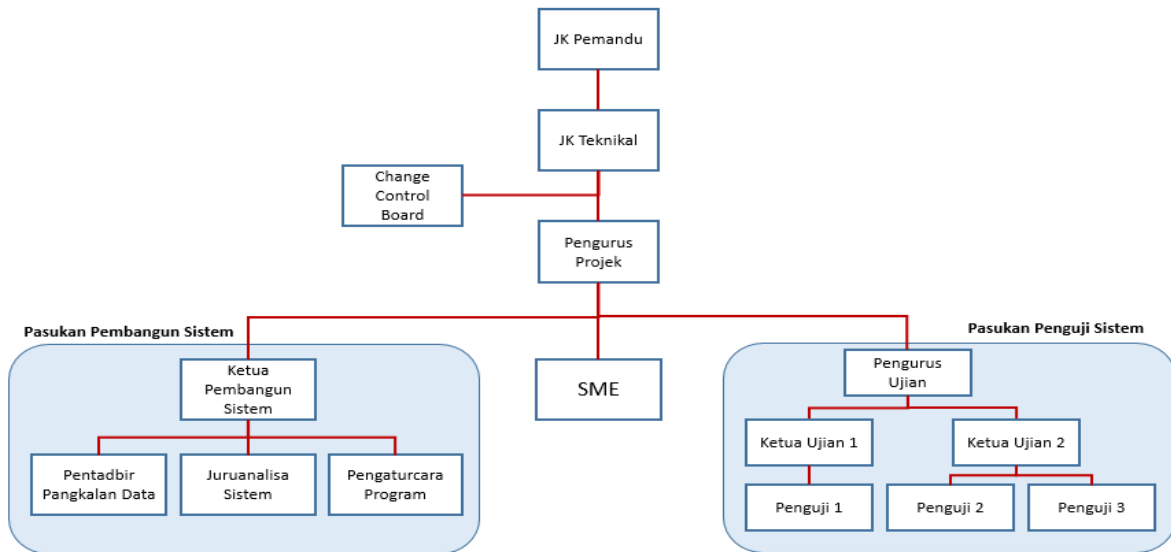
Jadual 73 : Contoh Jadual Perancangan Pengujian

BIL.	AKTIVITI	MAC				APR				MEI				JUN			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1.	UAT	■	■	■	■												
2.	Pembetulan ralat UAT					■	■	■									
3.	Ujian semula UAT								■	■							
4.	PAT										■	■	■	■	■		

Langkah 7 : Lengkapkan Struktur Perjawatan Pengujian

Struktur organisasi pasukan pengujian digambarkan dengan menggunakan sama ada carta organisasi, rajah matriks atau notasi-notasi lain. Struktur organisasi yang disediakan juga perlu menyertakan pasukan pengurusan projek, pasukan pembangunan sistem, pasukan teknikal (keselamatan) serta tadbir urus projek (Jawatankuasa Pemandu Projek dan Jawatankuasa Teknikal Projek) bagi menggambarkan bidang kuasa, tanggungjawab dan komunikasi di dalam projek.

Contoh struktur organisasi adalah sebagaimana dibawah:



Rajah 88 : Contoh Struktur Pasukan Pengujian

Jadual 74 : Contoh Peranan dan Tanggungjawab dalam Pengujian

PERANAN	TANGGUNGJAWAB
Pengurus Projek	<ul style="list-style-type: none"> Merancang, menyelaras dan melaksanakan keseluruhan pembangunan projek. Memastikan serahan sistem diterima dan disahkan oleh pemilik sistem.
Pengurus Ujian	<ul style="list-style-type: none"> Merancang, menyelaras dan melaksanakan aktiviti pengujian sistem. Menyediakan serahan sistem kepada pengurus projek.
Ketua Ujian	<ul style="list-style-type: none"> Mengenal pasti jenis ujian dan aktiviti pengujian yang akan dilaksanakan. Menyemak kes ujian, prosedur ujian dan senario ujian sebelum ujian dilaksanakan
Penguji	<ul style="list-style-type: none"> Melaksanakan senario ujian seperti yang telah dirancang Melaporkan ralat yang ditemui semasa ujian Menguji semula ralat yang telah diperbaiki
Pembangun Sistem	<ul style="list-style-type: none"> Memperbaiki ralat yang ditemui ketika pengujian dan melaksanakan semula pengujian unit/komponen

RUJUKAN

1. ISO/IEC 29119-3:2013 Software and systems engineering -- Software testing -- Part 3: Test documentation.

6.5 PENYEDIAAN DOKUMENTASI PERSEDIAAN UJIAN [F5.2]

KETERANGAN

Dokumentasi persediaan ujian adalah dokumen yang mengandungi maklumat terperinci bagi memulakan aktiviti-aktiviti pengujian yang dirancang di dalam Pelan Induk Pengujian dan merupakan lampiran bagi Pelan Ujian UAT dan PAT. Dokumen yang perlu disediakan sebelum sesuatu ujian dilaksanakan adalah seperti berikut:

- a) Senario Ujian (*Test Scenario*)
- b) Kes Ujian (*Test Case*)
- c) Prosedur/ Skrip Ujian (*Test Procedure/ Script*)
- d) Data Ujian (*Test Data*)
- e) *Traceability Matrix*

OBJEKTIF

- Memastikan skop pengujian menepati D03 Spesifikasi Keperluan Sistem dan D04 Spesifikasi Rekabentuk Sistem aplikasi.
- Memastikan pelaksanaan ujian secara sistematik, teratur dan menyeluruh.

LANGKAH-LANGKAH

Langkah 1: Tentukan Senario Ujian

a) Keterangan Ringkas

Tujuan menentukan Senario Ujian adalah untuk memastikan setiap fungsi diuji secara *end-to-end* bagi menjamin sistem aplikasi memenuhi keperluan proses bisnes. Senario ujian dihasilkan berdasarkan analisis terhadap dokumen SRS dan SDS. Setiap senario ujian perlu merujuk kepada sekurang-kurangnya satu fungsi bisnes.

b) Langkah

- i) Rujuk Rajah *Use Case* yang telah dibangunkan di dalam Pemodelan *Use Case* [F2.1] dan Senario *Use Case* yang telah disediakan di dalam Rekabentuk Transaksi Sistem [F3.5].
- ii) Kenalpasti aliran transaksi yang ingin diuji sebagai satu senario ujian. Satu senario ujian boleh mewakili satu aktiviti atau gabungan aktiviti yang dapat melengkapkan sesuatu fungsi bisnes.

Contoh Senario:

- i. Mohon Tempahan Bilik Mesyuarat – merangkumi aktiviti memasukkan profil pengguna untuk memulakan tempahan, menyemak kekosongan bilik, membuat pilihan bilik serta menghantar tempahan
 - ii. Pinda Maklumat Tempahan – hanya melibatkan aktiviti meminda maklumat tempahan
- iii) Senaraikan semua senario ujian yang telah dikenalpasti. Senario ini akan dijadikan sebagai senario utama.
 - iv) Kenalpasti senario alternatif iaitu senario negatif atau senario lain yang mungkin (jika ada) bagi *Use Case* atau senario utama tersebut berdasarkan maklumat yang dinyatakan di dalam SRS.

Contoh:

Senario bagi *Use Case* Mohon Tempahan Bilik Mesyuarat

Senario Utama : Permohonan Tempahan Bilik Mesyuarat Secara *Online*

Senario Alternatif: i. Permohonan Tempahan Bilik Mesyuarat Melalui Emel

ii. Permohonan Tempahan Bilik Mesyuarat Yang Telah Ditempah

- v) Senario ujian yang dikenalpasti perlu disemak dengan SME bagi memastikan semua fungsi dan keperluan diuji sewajarnya.
- vi) Senarai senario ujian yang telah dipersetujui diisikan ke dalam Templat Senario Ujian seperti di Apendiks 10a) Templat Senario Ujian. Bagi setiap senario ujian, lengkapkan maklumat-maklumat seperti berikut:

Jadual 75 : Isi Kandungan Templat Senario Ujian

Item	Keterangan
ID Use Case	ID <i>Use Case</i> yang dirujuk daripada dokumen SRS
Keterangan Use Case	Keterangan mengenai <i>Use Case</i> .
ID Senario Ujian	Nombor unik bagi senario ujian
Keterangan Senario Ujian	Keterangan mengenai senario.

Contoh Templat Senario Ujian yang telah diisi adalah seperti di bawah:

Jadual 76 : Contoh Pengisian Templat Senario Ujian (Memohon Tempahan Bilik Mesyuarat)

ID Use Case	UC-BM-MT-TBM-01
Keterangan Use Case	Mohon Tempahan Bilik Mesyuarat
ID Senario Ujian	Keterangan Senario Ujian
Senario utama	
SR-BM-MT-TBM-01-1	Memohon tempahan bilik mesyuarat secara <i>online</i>
Senario alternatif	
SR-BM-MT-TBM-01-2	Memohon tempahan bilik mesyuarat melalui e-mel
SR-BM-MT-TBM-01-3	Memohon tempahan bilik mesyuarat ke atas bilik yang telah ditempah tetapi belum diluluskan
SR-BM-MT-TBM-01-4	Memohon tempahan bilik dan membuat pindaan ke atas maklumat tempahan tersebut.

Jadual 77 : Contoh Pengisian Templat Senario Ujian (Melengkapkan maklumbalas penggunaan bilik)

ID Use Case	UC-BM-MT-TBM-03
Keterangan Use Case	Lengkapkan Maklumbalas Penggunaan Bilik Mesyuarat
ID Senario Ujian	Keterangan Senario Ujian
Senario utama	
SR-BM-MT-TBM-01-03-1	Melengkapkan maklumbalas penggunaan bilik mesyuarat

Bagi *Use Case* Lengkapkan Maklumbalas Penggunaan Bilik Mesyuarat, hanya satu senario ujian dikenalpasti kerana hanya pemohon yang telah diluluskan tempahan akan dipaparkan butang Isi Maklumbalas Penggunaan bagi pengisian maklumbalas penggunaan bilik mesyuarat.

Langkah 2: Sediakan Kes Ujian

a) Keterangan Ringkas

Kes ujian adalah satu set pra-syarat, input/ langkah-langkah ujian dan hasil jangkaan, yang dibangunkan untuk menguji sama ada sistem aplikasi memenuhi keperluan yang ditetapkan serta berfungsi dengan betul. Kes ujian disediakan berdasarkan fungsi asas yang terdapat di dalam senario ujian yang telah dikenalpasti. Kes ujian perlu menerangkan secara terperinci bagaimana untuk menguji fungsi asas tersebut serta mengambil kira situasi positif dan negatif.

Terdapat dua (2) pendekatan dalam menghasilkan kes ujian iaitu:

i) *High-level Test Case*

High-level Test Case ialah kes ujian yang mengandungi langkah-langkah umum untuk menguji sesuatu fungsi asas. Kes ujian sebegini sesuai dihasilkan sekiranya:

- Pengguna mempunyai pengetahuan tinggi terhadap proses bisnes serta kaedah navigasi di dalam sistem aplikasi yang dibangunkan; atau
- Kes ujian yang dibangunkan tidak memerlukan perincian bagi melaksanakan ujian (kurang kritikal).

Kes ujian *high-level* tidak menyatakan secara khusus sebarang nilai di dalam input atau jangkaan hasil ujian contohnya:

- Pengguna memasukkan tarikh tempahan bilik mesyuarat.
- Sistem akan memaparkan ralat jika tarikh tempahan yang dimasukkan telah lepas atau tempoh tempahan melebihi had yang dibenarkan.

ii) *Low-level Test Case*

Low-level Test Case ialah adalah kes ujian yang mengandungi langkah-langkah terperinci untuk menguji sesuatu fungsi asas. Kes ujian sebegini sesuai dihasilkan sekiranya:

- Pengguna tidak mempunyai pengetahuan tinggi terhadap aplikasi yang dibangunkan (pengguna tidak pernah atau tidak biasa menggunakan aplikasi tersebut); atau
- Kes ujian yang dibangunkan bersifat kritikal dan perlu diuji secara terperinci.

Kes ujian *low-level* perlu menyatakan secara khusus nilai bagi input dan jangkaan hasil ujian contohnya:

- Pengguna memasukkan 1 April 2017 di dalam ruangan tarikh mula guna dan 10 April 2017 di dalam ruangan tarikh tamat guna untuk membuat tempahan bilik mesyuarat. Seterusnya pengguna klik butang Hantar.

- Sistem akan memaparkan mesej ralat “Tempoh tempahan melebihi had 3 hari yang dibenarkan”. Pengguna klik butang Tutup untuk menutup mesej ralat.

b) Langkah

- Rujuk kepada Antaramuka Pengguna dan Senario *Use Case* yang terdapat di dalam D04 Spesifikasi Rekabentuk Sistem.
- Kenalpasti pendekatan yang ingin digunakan untuk menghasilkan kes ujian.
- Berdasarkan Rekabentuk Antaramuka Pengguna [F3.4] dan Rekabentuk Transaksi Sistem [F3.5], kenalpasti kes-kes ujian yang perlu disediakan serta langkah-langkah ujian yang diperlukan.
- Bangunkan kes ujian dengan mengisikan Templat Kes Ujian seperti di Apendiks 10b) Templat Kes Ujian dengan maklumat-maklumat seperti berikut:

Jadual 78 : Isi Kandungan Templat Kes Ujian

Item	Keterangan
ID Kes Ujian	Nombor unik bagi kes ujian.
Nama Kes Ujian	Nama bagi kes ujian.
Keterangan Kes Ujian	Keterangan ringkas mengenai kes ujian.
Rujukan ID Keperluan	ID <i>Use Case</i> yang dirujuk daripada dokumen SRS.
Pra-Syarat	Pra-syarat bagi membolehkan kes ujian dilaksanakan.
Input/ Langkah-Langkah Ujian	Turutan langkah-langkah dalam melaksanakan kes ujian.
Jangkaan Hasil	Jangkaan hasil bagi kes ujian.
Hasil Sebenar	Hasil sebenar yang diperolehi setelah kes ujian dilaksanakan.
Status	Keputusan/ status kes ujian sama ada Lulus atau Gagal.

Contoh Templat Kes Ujian yang telah diisi adalah seperti di bawah:

Jadual 79 : Contoh Pengisian Templat Kes Ujian (Log Masuk Positif)

ID KES UJIAN	TC-BM-LM-01-1		
NAMA KES UJIAN	Log Masuk (Positif)		
KETERANGAN KES UJIAN	Pegguna log masuk ke dalam sistem menggunakan ID pengguna (nombor kad pengenalan) dan kata laluan yang sah.		
RUJUKAN ID KEPERLUAN	UC-BM-LM-01		
PRA-SYARAT	Pegguna telah berdaftar di dalam sistem.		
INPUT / LANGKAH-LANGKAH UJIAN	JANGKAAN HASIL	HASIL SEBENAR	STATUS (LULUS/ GAGAL)
<ol style="list-style-type: none"> 1. Masuk ke dalam sistem melalui <alamat URL>. 2. Masukkan nombor kad pengenalan yang sah dalam ruangan ID Pengguna. 3. Masukkan kata laluan yang sah dalam ruangan Kata Laluan. 4. Klik butang Hantar. 	<p>Laman utama sistem akan dipaparkan.</p> <p>Laman <i>dashboard</i> pengguna akan dipaparkan.</p>		

Jadual 80 : Contoh Pengisian Templat Kes Ujian (Log Masuk Negatif)

ID KES UJIAN	TC-BM-LM-01-2		
NAMA KES UJIAN	Log Masuk (Negatif)		
KETERANGAN KES UJIAN	Pengguna log masuk ke dalam sistem menggunakan ID pengguna (nombor kad pengenalan) yang sah tetapi kata laluan yang salah.		
RUJUKAN ID KEPERLUAN	UC-BM-LM-01		
PRA-SYARAT	Pengguna telah berdaftar di dalam sistem.		
INPUT / LANGKAH-LANGKAH UJIAN	JANGKAAN HASIL	HASIL SEBENAR	STATUS (LULUS/ GAGAL)
<ol style="list-style-type: none"> Masuk ke dalam sistem melalui <alamat URL>. Masukkan nombor kad pengenalan yang sah dalam ruangan ID Pengguna. Masukkan kata laluan yang salah dalam ruangan Kata Laluan. Klik butang Hantar. 	<p>Laman utama sistem akan dipaparkan.</p> <p>Sistem akan memaparkan mesej <i>pop-up</i> "Kata Laluan yang anda masukkan adalah salah."</p>		

Jadual 81 : Contoh Pengisian Templat Kes Ujian (Tempahan Bilik Mesyuarat)

ID KES UJIAN	TC- BM-MT-TBM-01-1		
NAMA KES UJIAN	Membuat Tempahan Bilik Mesyuarat		
KETERANGAN KES UJIAN	Pengguna membuat tempahan bilik mesyuarat yang kosong.		
RUJUKAN ID KEPERLUAN	UC-BM-MT-TBM-01		
PRA-SYARAT	<ul style="list-style-type: none"> • Pengguna yang sah berjaya log masuk ke dalam sistem. • Bilik mesyuarat yang ingin ditempah telah dipastikan kekosongannya. 		
INPUT / LANGKAH-LANGKAH UJIAN	JANGKAAN HASIL	HASIL SEBENAR	STATUS (LULUS/ GAGAL)
<ol style="list-style-type: none"> 1. Pilih menu Tempahan Bilik Mesyuarat. 2. Pilih tarikh bagi Tarikh Mula dan Tarikh Tamat bagi carian kekosongan bilik. 3. Klik butang Semak. 	<ul style="list-style-type: none"> • Antara muka Permohonan Tempahan Bilik Mesyuarat dipaparkan. • Secara <i>default</i> nama dan emel login dipaparkan di ruangan Butiran Pegawai untuk dihubungi. • Secara <i>default</i>, sistem akan akan memaparkan tarikh semasa sebagai Tarikh Mula dan Tarikh Tamat. <p>Sistem akan memaparkan senarai bilik yang berstatus kosong pada tarikh yang dipilih. Butiran yang dipaparkan adalah nama bilik, kapasiti bilik dan status bilik.</p>		

4. Pilih dengan <i>tick</i> pada ruangan di sebelah Nama Bilik yang ingin ditempah.	Sistem akan memaparkan mesej <i>pop-up</i> "Permohonan anda telah dihantar kepada pegawai tadbir untuk kelulusan."		
5. Klik butang Hantar.			

Semak Kes Ujian yang dihasilkan dengan Ketua Ujian bagi memastikan setiap fungsi asas yang terlibat dilaksanakan pengujian yang sesuai dan terperinci.

Langkah 3: Sediakan Prosedur/ Skrip Ujian

a) Keterangan Ringkas

Prosedur/ skrip ujian adalah turutan kes ujian (*test cases*) yang mengandungi arahan terperinci untuk menguji senario ujian. Prosedur/ skrip ujian merujuk kepada pelaksanaan kes-kes ujian yang terlibat secara manual atau menggunakan peralatan ujian (*testing tool*).

b) Langkah

Sediakan prosedur/ skrip ujian dengan mengisi Templat Prosedur/ Skrip Ujian seperti di Apendiks 10c Templat Prosedur/ Skrip Ujian dengan maklumat-maklumat seperti berikut:

Jadual 82 : Isi Kandungan Templat Prosedur/ Skrip Ujian

Item	Keterangan
ID Prosedur/Skrip Ujian	Unik ID bagi prosedur/ skrip ujian.
ID Senario Ujian	ID Senario Ujian yang dirujuk bagi penghasilan prosedur/ skrip ujian.
Keterangan Senario Ujian	Keterangan ringkas mengenai senario ujian.
Objektif	Objektif prosedur/ skrip ujian dibangunkan.
Langkah Permulaan	Langkah-langkah yang perlu dijalankan untuk pelaksanaan set kes ujian di dalam prosedur tersebut.
Kaitan Dengan Prosedur Lain	Kaitan/ kebergantungan (<i>dependencies</i>) prosedur kepada prosedur yang lain.
Log Ujian	Maklumat tarikh, tandatangan personel yang bertanggungjawab dalam pelaksanaan ujian, item yang diuji dan status ujian bagi pengujian yang dilaksanakan.
Prosedur Ujian	Set kes ujian yang terlibat mengikut turutan beserta maklumat hasil dan keputusan ujian.

Contoh Templat Prosedur/ Skrip Ujian yang telah diisi adalah seperti di bawah:

Jadual 83 : Contoh Pengisian Templat Prosedur/ Skrip Ujian

ID Prosedur/ Skrip Ujian	PR-BM-MT-TBM-02			
ID Senario Ujian	SR-BM-MT-TBM-01-4			
Keterangan Senario Ujian	Memohon tempahan bilik dan membuat pindaan ke atas maklumat tempahan tersebut.			
Objektif	Bilik mesyuarat dapat ditempah dan maklumat tempahan berjaya dipinda.			
Langkah Permulaan :				
<ol style="list-style-type: none"> 1. Bilik mesyuarat telah didaftarkan ke dalam sistem. 2. Terdapat bilik mesyuarat yang kosong pada tarikh yang ingin ditempah. 3. Pengguna log masuk dengan ID pengguna dan kata laluan yang sah. 				
Kaitan dengan prosedur lain : PR- BM-MT-TBM-01				
LOG UJIAN				
Tarikh	Tandatangan	Item Ujian	Status Ujian (Lulus / Gagal)	
Komen :				
PROSEDUR UJIAN				
ID Kes Ujian	Nama Kes Ujian	Jangkaan Hasil	Hasil Sebenar/ Catatan	Keputusan Ujian
TC-BM- LM-01-1	Log Masuk (Positif)	Pengguna dipaparkan laman utama tempahan bilik mesyuarat.		
TC-BM- MT-TBM- 01-1	Membuat Tempahan Bilik Mesyuarat	Pengguna dipaparkan makluman bahawa tempahan bilik mesyuarat telah dihantar kepada pegawai tadbir untuk kelulusan.		
TC-BM- MT-TBM- 01-04-1	Meminda Maklumat Tempahan Bilik Mesyuarat	Pengguna dipaparkan makluman bahawa maklumat tempahan telah dikemaskini.		

Langkah 4: Sediakan Data Ujian

a) Keterangan Ringkas

Data Ujian ialah data yang dicipta atau dipilih bagi dijadikan input untuk melaksanakan pengujian ke atas kes ujian. Data ujian boleh dijana dengan:

- i) Menyediakan secara manual;
- ii) Menyalin data daripada persekitaran produksi (*production*) ke persekitaran pengujian;
- iii) Menyalin data ujian daripada sistem aplikasi terdahulu (*legacy*); atau
- iv) Menggunakan Alat Penjanaan Data Ujian Automatik (*Automated Test Data Generation Tools*).

b) Langkah

Sebagai persediaan untuk melaksanakan pengujian, sediakan data ujian bagi setiap kes ujian dalam dua kategori iaitu positif dan negatif. Data ujian positif akan menguji fungsi sistem aplikasi untuk menjana hasil seperti jangkaan. Manakala data ujian negatif adalah bagi menguji keupayaan sistem aplikasi mengendalikan input luar kebiasaan, ekstrim atau yang tidak dijangka.

Jadual 84 : Kategori Data Ujian

KATEGORI	INPUT	OUTPUT
POSITIF	Data ujian yang betul.	Sistem memberi respon terhadap data input dengan mengeluarkan output yang dijangkakan.
NEGATIF	Data ujian yang salah (contoh: format data yang salah, data di luar julat, data yang tidak wujud dalam sistem).	Sistem memaparkan mesej ralat yang bersesuaian terhadap data input.

Penyediaan data ujian yang mencukupi dan merangkumi pelbagai situasi adalah amat penting bagi menghasilkan sistem aplikasi yang berkualiti.

Langkah 5: Sediakan *Traceability Matrix*

a) Keterangan Ringkas

Traceability Matrix disediakan bagi tujuan menjejaki sesuatu keperluan (*requirement*) dan hubungan sepanjang kitar hayat pembangunan sistem (SDLC). Ia berupaya memastikan:

- i) semua keperluan sistem telah dibangunkan;
- ii) keperluan atau fungsi yang dibangunkan telah dilaksanakan pengujian; dan
- iii) semua pindaan ke atas aktiviti yang berkaitan dilaksanakan.

Kelebihan penggunaan *Traceability Matrix* adalah seperti berikut:

- i) Memastikan keseluruhan keperluan diuji dan berjaya dipenuhi.
- ii) Mengenal pasti keperluan yang tidak dinyatakan atau tidak konsisten.
- iii) Mengenal pasti ralat dan status pengujian berorientasikan keperluan bisnes.
- iv) Membantu mengenal pasti dan menganggarkan implikasi pembetulan ralat semasa pengujian atau pengurusan perubahan.

b) Langkah

Sediakan *traceability matrix* dengan mengisi Templat *Traceability Matrix* seperti di Apendiks 10d Templat *Traceability Matrix* dengan maklumat-maklumat seperti berikut:

Jadual 85 : Isi Kandungan Templat *Traceability Matrix*

Item	Keterangan
ID Senario Ujian	Unik ID bagi senario ujian.
ID Use Case	ID <i>Use Case</i> yang dirujuk daripada dokumen SRS
ID Kes Ujian	Unik ID bagi kes ujian yang merujuk kepada keperluan.
Keterangan Kes Ujian	Keterangan ringkas bagi kes ujian.

Contoh Templat *Traceability Matrix* yang telah diisi adalah seperti di bawah:

Jadual 86 : Contoh Pengisian Templat *Traceability Matrix*

ID Senario Ujian	ID Use Case	ID Kes Ujian	Keterangan Kes Ujian
SR-BM-MP-DP-01-1	UC-BM-MP-DP-01	TC-BM-MP-DP-01-1	Mendaftar akaun baharu.
SR-BM-MT-TBM-02	UC-BM-LM-01	TC-BM-LM-01-1	Pengguna log masuk ke dalam sistem menggunakan ID pengguna (nombor kad pengenalan) dan kata laluan yang sah.

	UC-BM-MT-TBM-01	TC-BM-MT-TBM-01-1	Pengguna membuat tempahan bilik mesyuarat yang kosong.
	UC-BM-MT-TBM-01-04	TC-BM-MT-TBM-01-04-1	Pengguna membuat pindaan maklumat tempahan bilik mesyuarat.

RUJUKAN

1. ISO/IEEE/IEC 29119-3: Software and systems engineering - Software testing - Test documentation
2. Foundations of Software Testing- ISTQB by Rex Black, Erik Van Veenendaal, Dorothy Graham
3. Test documentation by The Quest, winner of Test Design Competition SOFTEC Asia 2017
4. "Software Test Documents - Test Plan, Test Scenario, Test Case, Traceability Matrix" - Jaiganesh Periyasamy, Senior Test Engineer, <http://jobsandnewstoday.blogspot.my/2013/05/software-test-documents-testplan-testscenario-testcase.html>
5. "What's the difference between a Test Case and a Test Scenario?" - Jake Bartlett, Software Tester, <https://blog.testlodge.com/whats-the-difference-between-test-case-and-test-scenario/>
6. "Tips and Tricks to Generate Test Data", <https://www.guru99.com/software-testing-test-data.html>
7. "How to Review SRS Document and Create Test Scenarios", <http://www.softwaretestinghelp.com/rview-srs-document-and-create-test-scenarios-software-testing-training-course-day-2/>

6.6 PENYEDIAAN PELAN UJIAN (UAT & PAT) [F5.3]

KETERANGAN

Pelan Ujian UAT dan Pelan Ujian PAT mengandungi strategi pengujian, rekabentuk pengujian, jenis pengujian dan serahan ujian yang khusus bagi ujian di peringkat UAT atau PAT. Pelan Ujian UAT dan PAT ini perlu dihasilkan secara berasingan. Sebahagian perancangan bagi aktiviti UAT dan PAT berkemungkinan tidak dinyatakan secara terperinci di dalam Pelan Induk Pengujian yang dihasilkan lebih awal.

Apabila Pelan Ujian UAT dan PAT dibangunkan, perlu ada rujukan silang terhadap Pelan Induk Pengujian untuk menjamin kesinambungan aktiviti pengujian. Jika terdapat sebarang perubahan terhadap Pelan Induk Pengujian, ia perlu direkodkan dengan jelas di dalam Pindaan Dokumen.

OBJEKTIF

Menghasilkan Pelan Ujian bagi merancang dan mengurus aktiviti Ujian Penerimaan Pengguna (UAT) atau Ujian Penerimaan Sementara (PAT).

LANGKAH-LANGKAH

Langkah 1 : Sediakan Pengenalan Projek

Struktur dan kandungan pengenalan projek bagi Pelan UAT/PAT adalah sama seperti di dalam Pelan Induk Pengujian. Ia boleh dirujuk dan disalin dari Langkah 1, **Penyediaan Pelan Induk Pengujian [F5.1]**.

Langkah 2 : Lengkapkan Konteks Ujian

Struktur konteks ujian bagi Pelan UAT/PAT adalah sama seperti di dalam Pelan Induk Pengujian. Walau bagaimanapun, kandungan konteks ujian lebih terperinci dan spesifik bagi pengujian UAT/PAT.

a) Item ujian

Rujuk item (a), Langkah 2, Penyediaan Pelan Induk Pengujian [F5.1] untuk panduan melengkapkan item ujian UAT/PAT.

b) Skop ujian

Rujuk item (b), Langkah 2, Penyediaan Pelan Induk Pengujian [F5.1] untuk panduan melengkapkan skop ujian UAT/PAT.

Contoh bagi penentuan skop ujian UAT adalah seperti berikut :

Skop ujian adalah untuk menguji fungsi-fungsi modul yang telah dinyatakan di dalam Item Ujian (a).

Contoh bagi penentuan skop ujian PAT:

Ujian PAT yang akan dilaksanakan merangkumi skop berikut :

- i) Ujian prestasi – untuk mengenal pasti bagaimana sistem bertindak di bawah beban kerja dan kekangan tertentu untuk melaksanakan sesuatu fungsi.
- ii) Ujian bebanan – untuk memastikan tindak balas sistem apabila berada di bawah beban yang berbeza (rendah, sederhana, tinggi)

c) Kekangan

Kekangan adalah sekatan atau halangan yang dijangka akan berlaku semasa pengujian dilaksanakan. Bagi mendapatkan penerangan lebih terperinci berkenaan kekangan, rujuk item (c), Langkah 2, Pelan Induk Pengujian [F5.1] untuk panduan.

Contoh kekangan di dalam Pelan Ujian PAT adalah seperti berikut :

Ujian Prestasi di dalam PAT tidak dapat dilaksanakan jika UAT belum selesai sepenuhnya.

Langkah 3 : Lengkapkan Komunikasi Pengujian

Rujuk Langkah 3, **Pelan Induk Pengujian [F5.1]** untuk panduan.

Langkah 4 : Lengkapkan Daftar Risiko (*Risk Register*) Pengujian

Rujuk item (b) dan (c), Langkah 3, **Pelan Induk Pengujian [F1.1]** untuk panduan.

Langkah 5 : Lengkapkan Strategi Ujian

Struktur strategi ujian bagi Pelan UAT/PAT adalah sama seperti di dalam Pelan Induk Pengujian. Walau bagaimanapun, kandungan strategi ujian lebih terperinci dan spesifik bagi pengujian UAT/PAT.

a) Peringkat ujian

Sub-proses ujian menerangkan tentang jenis-jenis ujian yang terlibat dan strategi pelaksanaannya.

Berikut adalah contoh strategi ujian bagi ujian bebanan:

Ujian bebanan bagi Sistem Tempahan Mesyuarat diukur berdasarkan bilangan pengguna seperti berikut :

- i) 200 pengguna per saat (rendah)
- ii) 400 pengguna per saat (sederhana)
- iii) 1000 pengguna per saat (tinggi)

Bilangan pengguna maya yang akan digunakan untuk simulasi bebanan ke atas sistem adalah seperti berikut :

Jadual 87 : Contoh Strategi Ujian Bagi Ujian Bebanan

Bilangan Pengguna Maya (Virtual)	Pattern	Jangkaan Purata Masa Tindak Balas
50 (beban rendah)	Step	≤ 7 saat/transaksi
60 (beban sederhana)	Step	≤ 7 saat/transaksi
100 (beban tinggi)	Step	≤ 7 saat/transaksi

Contoh strategi ujian bagi ujian prestasi :

Ujian ini bertujuan untuk menentukan bagaimana prestasi sistem di bawah *workload* dan kekangan tertentu dapat mencapai purata masa tindak balas selama 10 saat.

Metrik pengukuran berikut akan digunakan dalam ujian prestasi tersebut:

Jadual 88 : Contoh Strategi Ujian Bagi Ujian Prestasi

Transaksi	Kriteria	Pengukuran
Melengkapkan borang permohonan	<i>Concurrent users</i>	<i>Number of virtual users</i>
	<i>Average response time</i>	<i>Second (s)</i>
	<i>Average page time</i>	<i>Second (s)</i>
	<i>Average transaction time</i>	<i>Second (s)</i>
	<i>Hits per second</i>	<i>Hits/sec</i>
	<i>Page per second</i>	<i>Page/sec</i>
	<i>Transaction per second</i>	<i>Transaction/sec</i>
	<i>Status code success</i>	<i>Percentage (%)</i>

Contoh strategi ujian bagi ujian tekanan:

Berikut adalah bilangan pengguna maya yang akan digunakan untuk simulasi bebanan ke atas sistem :

Jadual 89 : Contoh Strategi Ujian Bagi Ujian Tekanan

<i>Pattern</i>	<i>Ramp-Up Rate</i>	<i>Jangkaan Purata Masa Tindak Balas</i>
500	<i>Step</i>	1 VU/1sec
100	<i>Step</i>	100 VU/1sec
200	<i>Step</i>	200 VU/1sec

Pada masa yang sama, *system resources* akan dipantau seperti berikut:

Jadual 90 : Contoh Strategi Ujian Bagi Pemantauan System Resources

<i>System Resource</i>	<i>Criteria</i>	<i>Measurement</i>
<i>Processor</i>	<i>Utilization</i>	<i>CPU Utilization Percentage</i>
<i>Memory</i>	<i>Utilization</i>	<i>Memory Utilization Percentage</i>

Contoh strategi ujian bagi ujian keselamatan:

Ujian keselamatan bertujuan untuk mengenalpasti kelemahan terhadap ancaman keselamatan sistem. Ujian keselamatan ini dilaksanakan berdasarkan sepuluh (10) senarai teratas ancaman yang disenaraikan *Open Web Application Security Project (OWASP) 2017* seperti berikut :

Bagi mengenalpasti kelemahan atau ancaman keselamatan terhadap sistem, ujian keselamatan dijalankan ke atas sistem aplikasi berdasarkan sepuluh (10) ancaman yang disenaraikan *Open Web Application Security Project (OWASP) 2013* seperti berikut :

- i) *SQL Injection* – data yang tidak sah disuntik ke dalam sistem untuk membuka ruang kepada capaian data di dalam pangkalan data sistem. Contoh ancaman: *SQL, OS dan LDAP injection*.
- ii) *Broken Authentication and Session Management* – kelemahan fungsi aplikasi berkaitan dengan proses pengesahan (*authentication*) dan pengurusan *session* membolehkan penyerang memanipulasi kata kunci dan token *session* untuk memanipulasi akses pengguna sedia ada.

- iii) *Sensitive Data Exposure* – sistem aplikasi hendaklah melindungi data sensitif seperti maklumat kad kredit, nombor akaun pengguna, nombor ID pengguna (seperti nombor passport, nombor kad pengenalan) serta maklumat akses pengguna agar tidak dimanipulasi oleh pihak tidak bertanggungjawab.
- iv) *XML External Entities (XXE)* – Kelemahan konfigurasi ke atas pemproses XML membuka ruang kepada serangan seperti serangan *denial of service* (DOS).
- v) *Broken Access Control* – kelemahan kawalan dan penguatkuasaan terhadap aktiviti yang dibenarkan ke atas pengguna yang sah membuka ruang kepada capaian maklumat yang sulit dan terperingkat.
- vi) *Security Misconfiguration* – sesuatu sistem yang baik perlu dikonfigurasi di peringkat aplikasi, kerangka, pelayan aplikasi, pelayan web, pelayan pangkalan data dan platform serta tidak menggunakan konfigurasi *default* semata-mata. Selain itu, perisian juga perlu sentiasa dikemaskini.
- vii) *Cross-Site Scripting (XSS)* – penyerang akan melaksanakan skrip di dalam browser mangsa dan seterusnya berupaya memanipulasi *user sessions*, mengubah laman web sedia ada atau memesongkan pengguna kepada laman web lain.
- viii) *Insecure Deserialization* - penyerang berupaya membuat serangan bersiri ke atas kod aturcara secara *remote*.
- ix) *Using Components with Known Vulnerabilities* – eksploitasi ke atas komponen seperti *libraries*, kerangka dan modul perisian boleh menyebabkan kehilangan data yang serius atau manipulasi ke atas pelayan.
- x) *Insufficient Logging and Monitoring* – kelemahan kawalan ke atas log masuk sistem membolehkan penyerang memecah masuk dan merosakkan data di dalam sistem.

Ujian keselamatan ini biasanya dilaksanakan oleh pihak ketiga.

b) Serahan ujian

Sila rujuk item (b), Langkah 5, Pelan Induk Pengujian [F5.1] untuk panduan.

c) *Entry Criteria* dan *Exit Criteria*

Struktur *Entry criteria* dan *Exit criteria* bagi Pelan UAT/PAT adalah sama seperti di dalam Pelan Induk Pengujian. Walau bagaimanapun, kandungan *Entry criteria* dan *Exit criteria* lebih terperinci dan spesifik bagi pengujian UAT/PAT.

Contoh bagi *entry criteria* dan *exit criteria* untuk Pelan Ujian UAT/PAT adalah seperti berikut:

Jadual 91 : Contoh *Entry Criteria* Dan *Exit Criteria* Untuk Pelan Ujian UAT/PAT

Jenis Ujian	Contoh <i>Entry criteria</i>	Contoh <i>Exit criteria</i>
UAT	<ul style="list-style-type: none"> • Pelan Induk Pengujian telah diterima dan disahkan oleh pemilik projek • Dokumen SRS telah diterima dan disahkan oleh pemilik projek • Persekitaran pengujian telah disediakan • Data ujian telah disediakan • Instalasi dan konfigurasi sistem yang diuji (SUT) telah selesai dilaksanakan 	<ul style="list-style-type: none"> • 100% <i>test scenario</i> telah diuji • 95% <i>test case</i> telah lulus dalam ujian. • 100% <i>test coverage</i> telah dicapai. • Semua ralat dengan tahap <i>severity</i> 1 dan 2 telah diperbaiki, diuji semula dan ditutup. • Laporan Penamatan UAT telah diserahkan dan disahkan oleh pemilik projek.
PAT	<ul style="list-style-type: none"> • UAT telah selesai dan berjaya dilaksanakan. 	<ul style="list-style-type: none"> • Semua ralat dengan tahap <i>severity</i> 1 dan 2 telah diperbaiki, diuji semula dan ditutup • Laporan Penamatan PAT telah diserahkan dan disahkan oleh pemilik projek.

d) Metrik Pengukuran

Metrik pengukuran yang digunakan semasa pengujian akan disenaraikan. Contoh metrik pengukuran untuk Ujian Prestasi adalah seperti berikut:

Jadual 92 : Contoh Metrik Pengukuran

KRITERIA	PENGUKURAN
<i>Concurrent users</i>	<i>Number of virtual users</i>
<i>Average response time</i>	<i>Second (s)</i>
<i>Average page time</i>	<i>Second (s)</i>
<i>Average transaction time</i>	<i>Second (s)</i>
<i>Hits per second</i>	<i>Hits/sec</i>
<i>Page per second</i>	<i>Page/sec</i>
<i>Transaction per second</i>	<i>Transaction/sec</i>

e) Data Ujian

Rujuk item (e), Langkah 5, Pelan Induk Pengujian [F5.1] untuk panduan.

f) Persekitaran ujian

Rujuk item (f), Langkah 5, Pelan Induk Pengujian [F5.1] untuk panduan.

g) Penetapan Tahap Severity

Rujuk item (g), Langkah 5, Pelan Induk Pengujian [F5.1] untuk panduan.

Langkah 6 : Lengkapkan Jadual Perancangan Pengujian

Rujuk Langkah 6, **Pelan Induk Pengujian [F5.1]** untuk panduan

Langkah 7 : Lengkapkan Struktur Perjawatan Pengujian

Rujuk Langkah 7, **Pelan Induk Pengujian [F5.1]** untuk panduan.

Langkah 8 : Lengkapkan Lampiran Bagi Pelan Pengujian UAT / PAT

Dokumen berkaitan pelaksanaan ujian UAT dan PAT seperti senario ujian, prosedur ujian, kes ujian dan *traceability matrix* seperti yang diterangkan di **Dokumentasi Persediaan Ujian [F5.2]** akan dilampirkan bersama Pelan Pengujian UAT dan PAT ini. Dokumen yang dilampirkan boleh dikemaskini dari semasa ke semasa.

RUJUKAN

1. ISO/IEC 29119-3:2013 : Software Testing — Part 2 : Test Processes

6.7 UJIAN PENERIMAAN PENGGUNA (UAT) [F5.4]

KETERANGAN

Ujian Penerimaan Pengguna (UAT) perisian atau aplikasi diuji pada persekitaran pengujian (*Staging* atau *Test Environment*). Keberkesanan integrasi dan data migrasi juga akan diuji semasa UAT. Senario pengujian dan skrip ujian akan dibangunkan agar setiap keperluan dapat diuji secara menyeluruh.

Bagi memastikan keberkesanan UAT, sesi *Walkthrough* bersama pengguna sistem perlu dilaksanakan untuk menilai kesesuaian senario ujian. Selain daripada itu, *smoke test/sanity test* juga perlu dilaksanakan oleh pasukan pembangun untuk memastikan fungsi-fungsi utama sistem berjalan dengan lancar sebelum UAT sebenar dilaksanakan.

Perancangan UAT sangat penting dalam memastikan :

- a) Semua peranan pengguna dapat diuji;
- b) Semua keperluan sistem dapat diuji; dan
- c) Semua senario ujian dapat diuji bagi mencapai 100% liputan ujian (*test coverage*)

UAT boleh dilaksanakan dalam beberapa kitaran mengikut perancangan di dalam Pelan Ujian UAT atau berdasarkan keperluan pengguna (bergantung kepada hasil ujian). Hasil ujian akan direkodkan dan dijadikan rujukan bagi aktiviti pembetulan ralat dan pengujian semula. Setelah UAT tamat, Laporan UAT akan dikeluarkan sebagai rujukan dan *entry criteria* untuk PAT.

OBJEKTIF

- Mengesahkan bahawa sistem yang telah dibangunkan memenuhi keperluan fungsian yang telah digariskan oleh pengguna.
- Mengesahkan bahawa sistem bebas daripada ralat (*high severity bugs*) yang boleh menjejaskan operasi sistem.
- Mendapatkan keyakinan pengguna untuk menggunakan sistem di dalam persekitaran sebenar.

LANGKAH-LANGKAH

Langkah 1 : Tentukan *Entry Criteria* dan *Exit Criteria* Bagi Ujian Penerimaan Pengguna (*User Acceptance Test - UAT*)

- a) ***Entry Criteria* dan *Exit Criteria* UAT** perlu ditetapkan dan dipersetujui terlebih dahulu di antara pembangun sistem, pasukan penguji, pemilik sistem dan pemegang taruh yang berkenaan.
- b) Contoh bagi ***Entry Criteria* UAT** :

- i) Ujian Sistem telah dilaksanakan dengan sempurna.
 - ii) Ralat/Pepijat yang dilaporkan dalam ujian unit telah diperbaiki, disahkan oleh wakil pengguna dan tiada lagi Ralat/Pepijat:
 - Prioriti Tinggi atau Sederhana dan
 - *Severity* Tinggi, Sederhana, dan Rendah
 - iii) Pelan UAT telah disahkan oleh Pasukan Projek.
 - iv) *Staging* atau *Test Environment* telah disediakan.
 - v) *System Under Test* (SUT) telah disediakan dalam *Staging* atau *Test Environment*.
 - vi) Senario Ujian telah disediakan oleh Pasukan Pembangun Sistem dan telah disahkan oleh Pasukan Projek.
 - vii) Skrip Ujian telah disediakan oleh Pasukan Pembangun Sistem dan telah disahkan oleh Pasukan Projek.
 - viii) Data Ujian telah disediakan dan dimasukkan ke dalam pangkalan data SUT.
 - ix) Pasukan Projek telah menerima keputusan Ujian Sistem dan/atau memberi kebenaran untuk memulakan UAT.
 - x) Pasukan penguji UAT telah diberikan penerangan.
- c) Contoh bagi **Exit Criteria UAT**:
- i) 100% Senario Ujian telah diuji
 - ii) 95% Kes Ujian telah lulus dalam ujian.
 - iii) 100% Liputan Ujian telah dicapai.
 - iv) Semua ralat dengan tahap severity 1 dan 2 telah diperbaiki, diuji semula dan ditutup.
 - v) Laporan Penamatan UAT telah diserahkan dan disahkan oleh pemilik projek.

Langkah 2 : Laksanakan UAT

- a) UAT melibatkan pemegang taruh seperti berikut:
 - i) SME (Pemilik Modul)
 - ii) Pasukan Projek
 - iii) Wakil Pengguna
 - iv) Wakil pasukan operasi

- b) Aktiviti-aktiviti yang dilaksanakan semasa UAT adalah :

- i) Pengujian perisian atau aplikasi pada persekitaran *Staging* atau *Test Environment*.
- ii) Pengujian fungsian berdasarkan senario sebenar pengguna melaksanakan kerja menggunakan aplikasi tersebut.
- iii) Melaksanakan sekurang-kurangnya satu (1) UAT lengkap.
- iv) Mendapatkan persetujuan bertulis Pasukan Projek untuk meneruskan fasa pengujian berikutnya.
- v) Melaksanakan *Defect Classification*.

Langkah 3 : Sediakan Laporan Ujian Penerimaan Pengguna (UAT)

Setelah pelaksanaan UAT disempurnakan, Laporan Ujian Penerimaan Pengguna perlu disediakan. **Rujuk Laporan Ujian Penerimaan (UAT & PAT) [F5.6].**

RUJUKAN

1. ISO/IEC 29119-3:2013: *Software and systems engineering- Software testing - Part 3: Test Documentation*

6.8 UJIAN PENERIMAAN PROVISIONAL (PAT) [F5.5]

KETERANGAN

Ujian Penerimaan Provisional (PAT) merangkumi ujian fungsian dan bukan fungsian ke atas sistem aplikasi yang telah dibangunkan. Ia adalah ujian akhir sebelum sistem aplikasi dilancarkan (*go live*). Objektif PAT adalah untuk menilai keupayaan sistem beroperasi di dalam persekitaran yang sebenar.

Bagi menguji fungsian sistem di dalam PAT, ia akan dibuat secara selektif. Hanya sebahagian fungsi akan dipilih untuk diuji kerana keseluruhan fungsi telah diuji semasa UAT. Pemilihan fungsi untuk diuji boleh dibuat berdasarkan fungsi kritikal atau fungsi (modul) mengikut lokasi PAT yang telah dipilih.

Ujian bukan fungsian di dalam PAT merangkumi ujian prestasi, ujian bebanan, ujian tekanan dan juga ujian keselamatan.

OBJEKTIF

- Memastikan bahawa sistem yang telah dibangunkan memenuhi keperluan fungsian dan bukan fungsian yang telah ditetapkan pengguna.
- Memastikan bahawa sistem bebas daripada ralat kritikal (*high severity bugs*) yang boleh menjejaskan operasi sistem.
- Mendapatkan keyakinan pengguna untuk menggunakan sistem di dalam persekitaran sebenar.

LANGKAH-LANGKAH

Langkah 1 : Tentukan *Entry Criteria* dan *Exit Criteria* Ujian Penerimaan Provisional (*Provisional Acceptance Test - PAT*)

- a) *Entry Criteria* dan *Exit Criteria* PAT perlu ditetapkan dan dipersetujui terlebih dahulu di antara pembangun sistem, pasukan penguji, pemilik sistem dan pemegang taruh yang berkenaan.
- b) Contoh ***Entry Criteria*** PAT :
 - i) UAT telah dilaksanakan dengan sempurna.
 - ii) Ralat yang dilaporkan dalam UAT telah diperbaiki dan disahkan oleh wakil pengguna dan tiada lagi Ralat/Pepijat
 - Prioriti Tinggi atau Sederhana dan
 - *Severity* Tinggi, Sederhana, dan Rendah
 - iii) Tempoh PAT atau *Pilot Run* telah ditetapkan (sekurang-kurangnya untuk satu (1) bulan mengikut keperluan projek).

- iv) Pengguna *Pilot* telah dikenalpasti.
- v) Lokasi *Pilot* telah dikenalpasti.
- vi) Strategi untuk PAT atau *Pilot Run* telah diluluskan oleh Jawatankuasa Pemandu Projek.
- vii) *Pre-production Environment* atau *Production Environment* telah disediakan.
- viii) Migrasi data telah dilaksanakan.
- ix) Manual Sistem telah disiapkan dan diluluskan untuk kegunaan PAT.
- x) Pengguna telah diberikan penerangan dan latihan menggunakan sistem.
- xi) Semua sistem terlibat telah sedia.
- xii) SUT telah dilancarkan di semua lokasi PAT.

Contoh **Exit Criteria PAT**:

- i) Semua ralat dengan tahap *severity* 1 dan 2 telah diperbaiki, diuji semula dan ditutup.
- ii) Laporan Penamatan PAT telah diserahkan dan disahkan oleh pemilik projek.

Langkah 2 : Laksanakan PAT

- a) PAT melibatkan pemegang taruh seperti berikut:
 - i) Pembangun Sistem
 - ii) Pasukan Projek
 - iii) SME (Pemilik Modul)
 - iv) Pengguna *Pilot*
 - v) Pasukan Operasi

- b) Aktiviti-aktiviti yang dilaksanakan semasa PAT adalah :
 - i) Pembangun sistem melakukan instalasi SUT di *Production Environment*, lengkap dengan data yang telah dimigrasi dari sistem legasi dan/atau proses manual.
 - ii) Pembangun sistem menyediakan pasukan pemantau PAT yang dilengkapi dengan peralatan komunikasi.
 - iii) Memastikan ujian bukan fungsian (*Non Functional Test*) dilaksanakan :
 - Ujian prestasi (*Performance Test*) – untuk mengenal pasti bagaimana sistem bertindak di bawah beban kerja dan kekangan tertentu untuk melaksanakan sesuatu fungsi.

- Ujian bebanan (*Load Test*) – untuk memastikan tindak balas sistem apabila berada di bawah beban yang berbeza (rendah, sederhana, tinggi)
 - Ujian tekanan (*Stress Test*) – untuk menentukan tindak balas sistem apabila diberi tekanan tertentu (melangkaui beban yang boleh ditampung oleh sistem).
 - Ujian keselamatan (*Security Test*) – untuk memastikan tahap kerentanan (*vulnerability*) sistem terhadap ancaman keselamatan tertentu
- iv) PAT dilakukan sekurang-kurangnya untuk tempoh satu (1) bulan mengikut keperluan projek:
- Satu (1) lokasi sekiranya semua pengguna berada di lokasi yang sama, atau
 - Menetapkan lokasi-lokasi yang berlainan sekiranya melibatkan pengguna luar
- v) Pembangun sistem melakukan *deployment* dan melancarkan SUT di setiap lokasi.
- vi) Pembangun sistem menempatkan *Deployment Engineer* di lokasi untuk suatu tempoh yang dipersetujui (contohnya dua (2) hari) bagi memantau dan melaporkan status.
- vii) Pembangun sistem memantau, menyediakan dan mengemukakan laporan status PAT kepada agensi bagi tujuan semakan dan penerimaan PAT.
- viii) Pasukan Operasi memantau panggilan/laporan ke *Help Desk* untuk masalah yang dilaporkan oleh pengguna, dan melaporkan kepada Pasukan Projek.
- ix) Pembangun sistem melakukan siasatan dan membuat pembaikan terhadap ralat yang dilaporkan oleh pengguna.
- x) Pembangun sistem melakukan tampalan (*patch*) terhadap *software* atau konfigurasi terhadap SUT, dalam tempoh penyelenggaraan (*Maintenance Window*) terhadap SUT

Langkah 3 : Sediakan Laporan Ujian Penerimaan Provisional (PAT)

Setelah pelaksanaan PAT disempurnakan, Laporan Ujian PAT perlu disediakan. Rujuk **Laporan Ujian Penerimaan (UAT & PAT) [F5.6]**.

RUJUKAN

1. ISO/IEC 29119-3:2013: *Software And Systems Engineering- Software Testing - Part 3: Test Documentation*.

6.9 LAPORAN UJIAN PENERIMAAN (UAT & PAT) [F5.6]

KETERANGAN

Laporan Ujian Penerimaan (UAT & PAT) adalah laporan yang perlu dikeluarkan setelah pelaksanaan Ujian Penerimaan Pengguna (UAT) atau Ujian Penerimaan Provisional (PAT) disempurnakan. Objektif laporan ini dikeluarkan adalah untuk melapor keupayaan sistem bagi persediaan pelaksanaan. Laporan Ujian Penerimaan adalah merujuk kepada ISO/IEC/IEEE 29119. Format Laporan Ujian Penerimaan (PAT) adalah merujuk kepada **D14 Laporan Ujian Penerimaan Pengguna**

OBJEKTIF

Melaporkan keupayaan sistem untuk sesuatu sesi Ujian Penerimaan telah dilakukan.

LANGKAH-LANGKAH

Langkah 1 : Terangkan Maklumat Spesifik Dokumen

Maklumat spesifik dokumen perlu mengandungi maklumat asas Pelan Ujian UAT/PAT seperti nombor ID, versi, tarikh dihasilkan dan tarikh dikemaskini, organisasi, pengesahan serta rekod pindaan dokumen. Rujuk **Penyediaan Dokumentasi Persediaan Ujian [F5.2] Langkah 2**.

Langkah 2 : Nyatakan Ringkasan Ujian Yang Dijalankan

Penerangan ringkas terhadap ujian UAT/PAT yang telah dijalankan. Rujuk **Penyediaan Pelan Ujian UAT/PAT [F5.4] Langkah 2(b) Skop Ujian**.

Langkah 3 : Nyatakan Perubahan Dari Pelan Ujian

Penerangan perubahan yang berlaku semasa pelaksanaan pengujian berbanding DENGAN Pelan Ujian (UAT/PAT).

Contoh:

Maklumat Perubahan :	Penambahan fungsi pada modul Bayaran
Sebab perubahan dilaksanakan :	Arahan Kementerian Kewangan
Maklumat Test Case yang terlibat :	Test ID : TC001
Dokumen sokongan	<i>Pekeliling Kewangan yang berkaitan Change Request Incident Reports</i>

Langkah 4 : Jelaskan Kriteria Penamatan Ujian PAT

Ringkasan situasi yang membolehkan penamatan ujian. Rujuk **Ujian Penerimaan Provisional (PAT) [F5.5] Langkah 1** untuk panduan.

Contoh :

EXIT CRITERIA	
100% test scenario telah diuji	✓
95% test case telah lulus dalam ujian	✓
100% test coverage telah dicapai	✓
Semua ralat dengan tahap severity 1 dan 2 telah diperbaiki, diuji semula dan ditutup.	✓
Laporan Penamatan UAT telah diserahkan dan disahkan oleh pemilik projek	✓

Langkah 5 : Nyatakan Faktor Yang Menghalang Kemajuan

Penerangan faktor-faktor yang menghalang atau melewatkan kemajuan ujian.

Contoh :

Ujian migrasi lewat dilaksanakan kerana saiz data migrasi yang besar dan perlu tapisan semula (*cleansing*).

Langkah 6 : Nyatakan Pengukuran Hasil Ujian

Seksyen ini menerangkan dapatan hasil ujian.

Contoh :

MODUL	KES UJIAN YANG LULUS	KES UJIAN YANG GAGAL		KES UJIAN YANG TIDAK SELESAI	JUMLAH	KADAR KEJAYAAN (%)	LIPUTAN KES UJIAN (%)
		TAHAP SEVERITY	BIL				
Modul A	20	1	1	0	25	80%	100%
		2	1				
		3	3				
Modul B	15	1	0	3	20	88%	85%
		2	0				
		3	2				
Modul C	13	1	0	0	18	72%	100%
		2	0				
		3	5				

Penerangan risiko baru, perubahan risiko dan risiko sedia ada (yang belum dapat diselesaikan) setelah ujian dilaksanakan. **Rujuk Penyediaan Pelan Induk Pengujian [F5.1], Langkah 5.**

Langkah 7 : Nyatakan Risiko

Penerangan risiko baru, perubahan risiko dan risiko sedia ada (yang belum dapat diselesaikan) setelah ujian dilaksanakan. **Rujuk Penyediaan Pelan Induk Pengujian [F5.1], Langkah 5.**

Langkah 8 : Nyatakan Serahan Ujian

Senarai serahan ujian yang perlu dihantar. **Rujuk Penyediaan Pelan Induk Pengujian [F5.1], Langkah 6(b).**

Langkah 9 : Nyatakan Aset Ujian Yang Boleh Diguna Semula

Senarai spesifikasi ujian, persekitaran ujian dan data ujian yang boleh digunakan semula bagi kitaran ujian seterusnya.

Langkah 10 : Nyatakan Lesson Learned

Pengajaran yang boleh dijadikan sumber rujukan dan diperbaiki dalam kitaran ujian akan datang.

RUJUKAN


1. ISO/IEC 29119-3:2013: *Software And Systems Engineering - Software Testing - Part 3: Test Documentation*



BAB:07

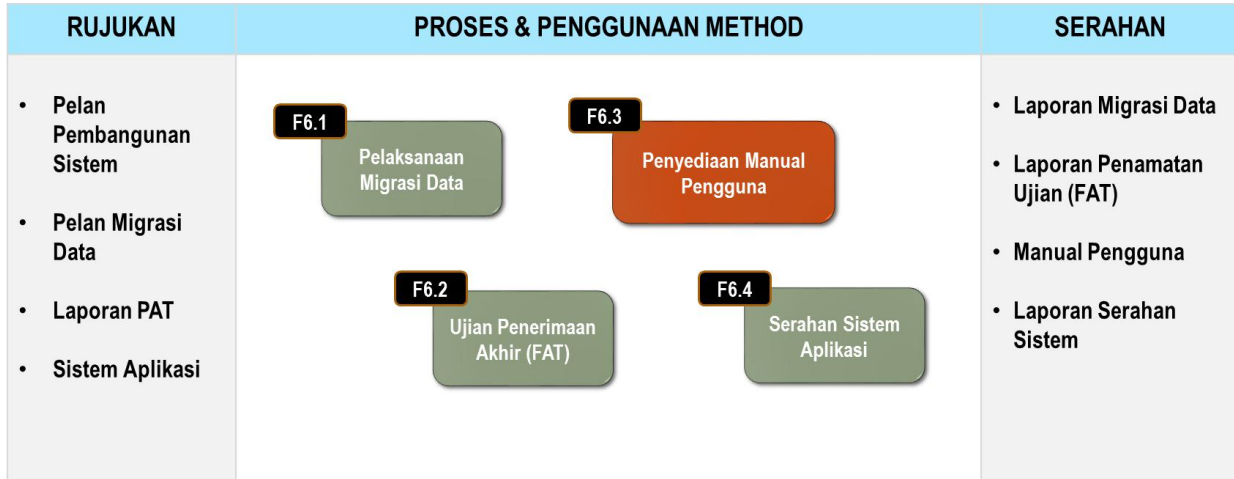
FASA PELAKSANAAN

Bab ini menerangkan aktiviti-aktiviti pelaksanaan sistem yang melibatkan migrasi data dari sistem yang lama kepada sistem yang baru, Ujian Penerimaan Akhir(FAT), penyediaan Manual Pengguna dan penyempurnaan serahan akhir.



7 FASA PELAKSANAAN

7.1 GAMBARAN KESELURUHAN



Rajah 89 : Gambaran Keseluruhan Fasa VI – Pelaksanaan

7.2 PENGENALAN

Tujuan pelaksanaan sistem adalah untuk memastikan sistem yang dibangunkan dapat berfungsi dengan lancar mengikut spesifikasi yang telah di minta oleh pengguna. Fasa pelaksanaan merupakan fasa yang melibatkan aktiviti-aktiviti peralihan daripada sistem yang lama kepada sistem yang baru. Ia akan melibatkan perubahan dari cara kerja manual kepada cara baru yang berasaskan kepada sistem berkomputer. Proses peralihan boleh dilaksanakan ke atas sistem semasa dengan membuat penambahbaikan mengikut keperluan yang baru. Semasa fasa pelaksanaan, isu-isu yang melibatkan bisnes, teknikal dan orang awam hendaklah dikenalpasti dan diambil tindakan.

Aktiviti-aktiviti dalam Fasa Pelaksanaan adalah:

- Pelaksanaan Migrasi Data
- Ujian Penerimaan Akhir
- Penyediaan Manual Pengguna
- Serahan Sistem Aplikasi

Sebelum membuat perancangan untuk pelaksanaan sistem, seharusnya terlebih dahulu mengenalpasti tahap kerumitan yang terlibat. Jika pelaksanaan yang akan dijalankan adalah mudah, maka ia tidak rumit untuk dilaksanakan. Walau bagaimanapun, kebanyakan projek mempunyai jadual pelaksanaan bagi memastikan sistem yang dibangunkan dapat dilaksanakan dengan jayanya.

Salah satu amalan terbaik dalam kitaran hayat projek adalah dengan menyediakan perancangan awal. Jika sistem yang dibangunkan mempunyai skop yang besar, maka organisasi perlulah menyediakan **Dokumen Perancangan Pelaksanaan Sistem**. Dokumen ini akan menerangkan tentang pendekatan secara menyeluruh bagi pelaksanaan, skop, andaian, risiko dan lain-lain.

Dokumen Rujukan kepada Fasa Pelaksanaan adalah seperti berikut:

- a) **D01 Pelan Pembangunan Sistem**
- b) **D05 Pelan Migrasi Data**
- c) **D14 Laporan UAT & Laporan PAT**

Dokumen Serahan kepada Fasa Pelaksanaan adalah seperti berikut:

- a) **D15 Laporan Migrasi Data**
- b) **D16 Laporan Penamatan Ujian**
- c) **D17 Manual Pengguna**
- d) **D18 Laporan Serahan Sistem**

7.3 PENGLIBATAN PEMEGANG TARUH

Pemegang taruh memainkan peranan yang penting dalam memastikan kejayaan pelaksanaan projek ICT di sesebuah organisasi. Oleh yang demikian, pengurusan projek ICT perlu melibatkan pelbagai pihak bagi mewujudkan suatu pengurusan projek yang mampan. Secara spesifiknya, pemegang taruh adalah orang-orang perseorangan dan organisasi yang terlibat secara aktif dalam projek atau pihak yang mempunyai kepentingan yang terlibat secara positif atau negatif dan boleh memberi kesan kepada pelaksanaan projek atau penyiapan projek dengan jayanya. Keberkesanan penglibatan pemegang taruh adalah salah satu penentu utama kepada kejayaan penyiapan projek ICT di sektor awam. Ini adalah kerana penglibatan pemegang taruh yang berkesan akan mempengaruhi prestasi dari aspek pengurusan kos, masa, kualiti, skop dan risiko projek ICT.

Lazimnya, pemegang taruh yang akan terlibat di dalam pelaksanaan projek ICT adalah:

- a) **Pengurus projek** - individu yang bertanggungjawab menguruskan projek.
- b) **Pelanggan** - orang perseorangan atau organisasi yang akan menggunakan produk projek.
- c) **Organisasi Pelaksana** - pekerja-pekerja yang terlibat secara langsung dalam melaksanakan kerja-kerja untuk projek
- d) **Pembiaya** - orang perseorangan atau kumpulan dalam organisasi pelaksana yang menyediakan sumber-sumber kewangan, samada dalam bentuk tunai atau barangan untuk projek.

7.4 FAKTOR KEJAYAAN

Dalam memastikan sesebuah projek ICT dapat dilaksanakan dengan jayanya dan mencapai objektif yang telah ditetapkan, sesebuah organisasi hendaklah mempertimbangkan beberapa faktor kejayaan kritikal. Antaranya ialah:

a) **Komitmen Daripada Pihak Pengurusan Atasan**

Komitmen daripada pihak pengurusan atasan merupakan faktor yang paling penting dalam memastikan ia dapat dibangunkan dan dilaksanakan dengan jaya di sesebuah organisasi. Pengurusan atasan hendaklah mengambil maklum tentang perjalanan projek dari peringkat awal sehinggalah projek tamat. Status kemajuan projek hendaklah dimaklumkan kepada pihak pengurusan tertinggi melalui mesyuarat yang dibentuk mengikut struktur tadbir urus yang telah dipersetujui.

b) **Komitmen Daripada Pihak Pengguna (*Subject Matter Expert*)**

Komitmen daripada pihak pengguna juga turut menjadi penyumbang kepada kejayaan pembangunan dan pelaksanaan projek ICT. Input dan maklum balas daripada pengguna adalah amatlah penting kerana tanpa input yang mencukupi sistem tidak akan dapat memenuhi spesifikasi yang ditetapkan. Oleh itu, komitmen dan kerjasama daripada pengguna amat diperlukan agar aliran proses sistem dapat berjalan dengan lancar. Kerjasama ini akan dapat dijalankan melalui sesi mesyuarat dan perbincangan di antara pengguna dan pegawai IT/pembangun sistem bagi memastikan setiap keperluan daripada pengguna akan ditransformasikan kepada sistem.

c) **Keperluan dan Spesifikasi Yang Jelas**

Mengenalpasti keperluan dan spesifikasi dengan jelas di awal projek adalah sangat penting. Kadang-kadang pengguna sendiri tidak memahami masalah yang mereka hadapi dan mereka meminta fungsi yang mereka tidak perlukan. Oleh itu, menjadi tanggungjawab pengurus projek atau penganalisa sistem untuk menganalisa masalah pengguna dan mendokumenkan satu senarai keperluan dan spesifikasi sistem yang jelas. Keperluan dan spesifikasi yang jelas mampu mengelakkan terlalu banyak perubahan semasa pembangunan, penyediaan jadual yang realistik dan membangun sistem yang tepat mengikut keperluan. Hampir kesemua proses pembangunan bergantung kepada kajian keperluan dan spesifikasi di awal proses pembangunan.

d) **Objektif dan Matlamat Yang Jelas**

Matlamat dan objektif penting kerana ianya akan menjadi hala tuju dan panduan dalam pembangunan sistem. Penetapan matlamat dan objektif hendaklah dinyatakan dengan jelas dan juga realistik. Projek yang berjaya adalah projek yang mencapai semua matlamat dan objektifnya dalam masa yang ditetapkan.

e) Jadual Yang Realistik

Penyediaan jadual yang realistik juga penting dalam memastikan projek disiapkan dalam jangka masa dan keperluan yang telah dirancang. Kesilapan selalunya berlaku apabila pengurus projek menggalas tanggungjawab seorang diri menyediakan jadual tanpa melibatkan ahli pasukan projek yang lain. Seharusnya, kesemua ahli pasukan projek hendaklah sama-sama berbincang memberi cadangan mengenai masa yang mereka perlukan.

f) Sumber Manusia

Sumber manusia juga merupakan faktor yang perlu di ambil kira dalam menentukan kejayaan sesebuah projek ICT. Oleh itu, setiap projek ICT yang akan dibangunkan perlu mempunyai struktur tadbir urus yang terdiri daripada Jawatankuasa Pemandu, Jawatankuasa Teknikal dan Pasukan Projek. Setiap jawatankuasa yang dilantik perlu dinyatakan peranan yang jelas agar pengurusan projek dapat dilaksanakan mengikut perancangan yang telah ditetapkan.

g) Kemahiran dan Keupayaan Personel ICT

Dalam mewujudkan tadbir urus pengurusan projek, kemahiran dan pengetahuan personel ICT adalah perlu diberi pertimbangan bagi memastikan projek dapat dilaksanakan mengikut masa dan kos yang ditetapkan serta dapat memenuhi kehendak pengguna.

h) Pengurusan Projek yang Cepak

Pengurus projek yang cekap dan pemilihan metodologi pengurusan projek yang tepat mampu meningkatkan peluang kejayaan projek. Pemilihan metodologi pengurusan projek hendaklah dilakukan dengan berhati-hati kerana setiap metodologi yang digunakan bergantung kepada kesesuaian projek yang dibangunkan. Sebagai contoh, metodologi tradisional pengurusan projek *waterfall* sesuai digunakan untuk projek yang mampu menjangka dan menyatakan keperluan projek dengan hanya sedikit ataupun tiada langsung perubahan. Manakala metodologi *agile* memberi kelebihan kepada keupayaan untuk menyesuaikan diri kepada perubahan yang berlaku semasa pembangunan projek.

7.5 PELAKSANAAN MIGRASI DATA [F6.1]

KETERANGAN

Pelaksanaan Migrasi Data melibatkan proses pemindahan data dilakukan mengikut **D05 Pelan Migrasi Data** dan **D06 Spesifikasi Migrasi Data** yang telah disediakan dalam Fasa Rekabentuk. Langkah-langkah yang terlibat adalah seperti berikut:

- a) Pembangunan Migrasi Data;
- b) Pengujian Sampel;
- c) Pelaksanaan Migrasi Data;
- d) Pengujian dan Verifikasi Data; dan
- e) Penyediaan Laporan.

OBJEKTIF

- Memindahkan data daripada sumber asal ke destinasi baharu; dan
- Memastikan data yang dipindahkan adalah berintegriti dan berkualiti.

LANGKAH-LANGKAH

Langkah 1 : Laksanakan Pembangunan Migrasi Data

- a) Pembangunan migrasi data melibatkan proses mengekstrak data daripada sumber asal dan menjalankan proses transformasi data (bagi membersihkan data dan memastikan integriti data) sebelum dipindahkan (*load*) ke destinasi baharu. Proses ini perlu dilakukan mengikut *data mapping template* pada Spesifikasi Migrasi Data yang telah disediakan.
- b) Kaedah yang biasa digunakan bagi pembangunan migrasi data adalah seperti:
 - i) Penghasilan skrip untuk migrasi data; dan
 - ii) Penggunaan *tools* untuk migrasi data;
- c) Contoh kaedah penghasilan skrip migrasi adalah melalui Notepad++, SQL scripting, UNIX/Windows command dan *bash scripting*. Manakala sebahagian contoh kaedah penggunaan *tools* untuk migrasi data pula ialah melalui perisian adalah seperti:
 - i) Navicat bagi pangkalan data MySQL, MariaDB, Oracle, PostgreSQL dan lain-lain;
 - ii) Aqua Data Studio bagi Oracle, DB2, MS SQL dan lain-lain;
 - iii) Entity Framework – Code First Migration; dan
 - iv) Oracle Data Pump bagi pangkalan data Oracle.

- d) Pada peringkat pembangunan migrasi data, data dipindahkan daripada pangkalan data sumber ke pangkalan data baharu di persekitaran *staging*/pembangunan melalui kaedah yang telah dikenal pasti. Data yang hendak dipindahkan perlu dibuat salinan pendua atau *backup* terlebih dahulu sebagai persediaan untuk sebarang risiko.

Langkah 2 : Lakukan Pengujian Sampel

- a) Ujian sampel dijalankan setelah data dipindahkan daripada pangkalan data sumber ke pangkalan data baharu di persekitaran *staging*/pembangunan.
- b) Pengujian boleh dijalankan melalui kaedah *scripting*, *SQL command* atau menggunakan fungsi pengujian yang disediakan pada *tools* yang digunakan. Perbandingan jumlah data yang dipindahkan dilakukan semasa pengujian ini.
- c) Proses verifikasi data juga boleh dilakukan mengikut kaedah yang akan diterangkan lanjut pada Langkah 4.
- d) Lakukan pembaikan pada *script/command* atau *tools* digunakan jika berlaku *error* atau ketidaktepatan data yang dipindahkan.
- e) Setelah pengujian di persekitaran *staging*/pembangunan berjaya, kenal pasti dan *backup* kesemua data yang perlu dimigrasi sekali lagi sebelum proses seterusnya.

Langkah 3 : Laksanakan Migrasi Data

- a) Pelaksanaan migrasi data melibatkan perpindahan data daripada pangkalan data sumber ke pangkalan data baharu di persekitaran *production*. Data akan dipindahkan menggunakan kaedah yang sama pada Langkah 1 iaitu sama ada melalui *scripting* atau menggunakan *tools*. Sebelum migrasi data dijalankan, pastikan data yang ingin dipindahkan telah dibuat salinan pendua atau *backup*. Aktiviti seperti berikut dijalankan semasa pelaksanaan migrasi data :
 - i) Pastikan ketersediaan pangkalan data baharu di persekitaran *production*;
 - ii) *Run* skrip migrasi atau gunakan *tools* untuk *load* data ke destinasi baharu pada persekitaran *production*;
 - iii) Dapatkan statistik data yang berjaya dan gagal dimigrasi; dan
- b) Lakukan verifikasi data bagi data yang telah dipindahkan daripada pangkalan data sumber ke pangkalan data baharu pada *server production*;

Langkah 4 : Lakukan Pengujian dan Verifikasi Data

- a) Fasa Pengujian dan Verifikasi pula melibatkan pengujian dan verifikasi bagi data yang telah dipindahkan. Pengujian ini dilakukan oleh pasukan pengujian yang telah dikenal pasti iaitu yang terdiri daripada pegawai agensi bersama-sama pasukan pembekal sekiranya dijalankan secara *outsourc*e.
- b) Proses pengujian biasanya dijalankan untuk menguji ketepatan bilangan data. Manakala proses verifikasi pula dijalankan untuk mengesahkan kesahihan dan integriti data. Selain itu, jaminan kualiti data (*data quality assurance*) yang berterusan perlu dijalankan untuk mengelakkan pertindihan data dan memastikan kesahihan data. Proses pengujian dan verifikasi data yang biasa dijalankan adalah seperti berikut:
 - i) Pengujian bilangan data dipindah dalam setiap table : menggunakan *SQL statement (Select Count)* atau *tools* tertentu; dan
 - ii) Pensampelan statistik untuk verifikasi nilai data: mendapatkan saiz rekod (sampel) yang bersesuaian. Pensampelan ini mengambil kira saiz populasi, *confidence level (margin of error, ±X %)* dan *confidence interval* (peratus keyakinan kesahihan data dalam lingkungan *margin of error* pilihan).

Contoh pengiraan sampel data yang perlu diuji adalah seperti berikut :

Jadual 93 : Contoh Pengiraan Sampel Data Bagi Ujian Migrasi Data

JUMLAH POPULASI = 10,000	
CONFIDENCE INTERVAL = 95%	
CONFIDENCE LEVEL	SAIZ SAMPEL YANG DIPERLUKAN
±2%	1,936
±3%	964
±5%	370

Sumber rujukan : <http://www.surveysystem.com/sscalc.htm#one>

Langkah 4 : Sediakan Laporan Migrasi Data

Di akhir proses migrasi data, satu Laporan Migrasi yang mengandungi maklumat berkaitan proses migrasi yang telah dijalankan akan dihasilkan.

- a) Laporan Migrasi Data disediakan selepas semua aktiviti migrasi termasuk pengujian selesai dijalankan. Laporan akan dijadikan makluman kepada pemegang taruh yang terlibat mengenai status pelaksanaan migrasi dan boleh dijadikan sebagai panduan untuk aktiviti migrasi yang akan datang.

b) Kandungan Laporan adalah terdiri daripada:

Jadual 94 : Isi Kandungan Laporan Migrasi Data

PERKARA	PENERANGAN
i. Jadual pelaksanaan sebenar	Jadual pelaksanaan yang sebenar
ii. Status Migrasi	Nyatakan status migrasi data daripada sumber data ke destinasi baharu – <i>complete/pending/etc.</i>
iii. Sumber Data	Nyatakan nama pangkalan data dan <i>table</i> sumber
iv. Destinasi Baharu Data	Nyatakan nama pangkalan data dan <i>table</i> destinasi baharu
v. Jumlah baris dalam <i>table</i> sumber	Nyatakan jumlah baris dalam <i>table</i> sumber
vi. Jumlah baris yang berjaya dimigrasi	Nyatakan jumlah baris (<i>row</i>) yang berjaya dimigrasi bagi setiap <i>table</i>
vii. Ratio	Peratusan baris (<i>row</i>) yang berjaya dimigrasi
viii. Perincian	Perincian maklumat sekiranya terdapat migrasi data yang gagal. Sebab kegagalan dan tindakan pembetulan perlu dinyatakan

Rujuk **D15 Laporan Migrasi Data** sebagai panduan.

RUJUKAN

1. Pelan Migrasi Data Sistem eRoses
2. Oracle White Paper (2011). Successful Data Migration. <http://www.oracle.com/technetwork/middleware/oedq/successful-data-migration-wp-1555708.pdf>
3. Credosoft White Paper. Eight key steps which help ensure a successful data migration project: A white paper for inspection management professionals. <http://credosoft.com/wp/wp-content/uploads/2014/01/Eight-key-steps-which-help-ensure-a-successfu-data-migration-project.pdf>
4. SAGA Group (2012). Methods of Data Migration.

7.6 UJIAN PENERIMAAN AKHIR [F6.2]

KETERANGAN

Ujian Penerimaan Akhir (FAT) merupakan proses formal untuk mendapatkan persetujuan dari Agensi bagi membolehkan sistem yang diuji memasuki fasa produksi dan juga Tempoh Jaminan (Warranty Period).

OBJEKTIF

- Mengesahkan bahawa sistem yang telah dibangunkan memenuhi keperluan yang telah digariskan pengguna
- Mengesahkan bahawa sistem bebas daripada ralat (high severity bugs) yang boleh mengganggu operasi sistem
- Mendapatkan keyakinan pengguna terhadap keseluruhan sistem

LANGKAH-LANGKAH

Langkah 1 : Tentukan *Entry Criteria* dan *Exit Criteria* Ujian Penerimaan Akhir (*Final Acceptance Test - FAT*)

Entry Criteria dan *Exit Criteria* FAT ditetapkan dan dipersetujui terlebih dahulu di antara pembangun sistem, pasukan penguji, pemilik sistem dan pemegang taruh yang berkenaan.

Jadual 95 : Syarat Masuk dan Keluar Ujian Penerimaan akhir

Entry Criteria	<ul style="list-style-type: none"> i) PAT telah dilaksanakan dengan sempurna. ii) <i>Defects/Bugs</i> yang dilaporkan dalam PAT telah diperbaiki dan disahkan oleh wakil pengguna dan tiada lagi <i>Defects/Bugs</i> <ul style="list-style-type: none"> • Prioriti Tinggi atau Sederhana dan • <i>Severity Blocking, Critical</i> dan <i>Major</i> iii) Latihan Pengguna telah disempurnakan. iv) Keputusan PAT telah diterima, dan/atau SUT telah diterima oleh Agensi melalui keputusan Jawatankuasa Pemandu Projek. v) Migrasi data telah dilakukan oleh <i>Development Team</i> dari sistem legasi dan/atau proses manual. vi) <i>Production Environment</i> telah disediakan. vii) Manual Sistem telah disiapkan dan diluluskan.
Exit Criteria	i. Pembangunan sistem telah tamat secara rasmi dan sistem memasuki fasa Tempoh Jaminan (<i>Warranty Period</i>).

Langkah 2 : Laksanakan FAT

FAT melibatkan aktor dan aktiviti-aktiviti berikut :

Jadual 96 : Aktor dan Aktiviti Ujian Penerimaan akhir

Aktor	i) Pembangun Sistem ii) Pasukan Projek iii) SME (Pemilik Modul) iv) Pengguna Sistem v) Pasukan Operasi
Aktiviti	i) <i>Development Team</i> melancarkan sistem di semua lokasi. ii) <i>Development Team</i> mendapatkan keputusan penerimaan sistem dari Jawatankuasa Pemandu Projek.

Langkah 3 : Sediakan Laporan Penamatan Ujian (*Test Completion Report*)

- Laporan Penamatan Ujian mengandungi ringkasan hasil ujian yang dilaksanakan. Laporan Penamatan Ujian boleh disediakan bagi sesuatu peringkat ujian atau bagi keseluruhan projek.
- Merujuk kepada ISO/IEC/IEEE 29119, Laporan Penamatan Ujian mempunyai elemen-elemen berikut:

Jadual 97 : Isi Kandungan Laporan Penamatan Ujian

Elemen	Keterangan						
Maklumat spesifik dokumen	Mengandungi maklumat asas seperti nombor ID, versi, tarikh dihasilkan dan tarikh dikemaskini, organisasi, pengesahan serta rekod pindaan dokumen						
Objektif Ujian	Penerangan mengenai objektif ujian yang dijalankan						
Skop Ujian	Skop ujian bertujuan untuk menetapkan perimeter ujian termasuk ciri-ciri (fungsian dan kualiti/ bukan fungsian) yang akan diuji ke atas item ujian dan ciri-ciri yang tidak akan diuji ke atas item ujian						
Butiran Ujian	Seksyen ini menerangkan butiran tempoh Ujian Penamatan akan dilaksanakan dan peserta yang akan terlibat. Contoh: <table border="1" data-bbox="486 1865 1358 2004"> <tbody> <tr> <td>Tarikh Mula FAT</td> <td>4hb Ogos 2018</td> </tr> <tr> <td>Tarikh Akhir FAT</td> <td>4hb September 2018</td> </tr> <tr> <td>Peserta FAT</td> <td>Pengguna Sistem Tempahan Bilik</td> </tr> </tbody> </table>	Tarikh Mula FAT	4hb Ogos 2018	Tarikh Akhir FAT	4hb September 2018	Peserta FAT	Pengguna Sistem Tempahan Bilik
Tarikh Mula FAT	4hb Ogos 2018						
Tarikh Akhir FAT	4hb September 2018						
Peserta FAT	Pengguna Sistem Tempahan Bilik						

Keperluan Persekitaran	<p>Keperluan persekitaran merujuk kepada keperluan perkakasan dan perisian yang digunakan semasa sesi Ujian Penamatan.</p> <p>Contoh:</p> <table border="1" data-bbox="472 320 1362 465"> <thead> <tr> <th>Bil.</th> <th>Keperluan Persekitaran</th> <th>Keterangan</th> </tr> </thead> <tbody> <tr> <td>1.</td> <td><i>Production Environment</i></td> <td>URL: https://www.tempahan.gov.my Pelayar: Chrome / Mozilla</td> </tr> </tbody> </table>	Bil.	Keperluan Persekitaran	Keterangan	1.	<i>Production Environment</i>	URL: https://www.tempahan.gov.my Pelayar: Chrome / Mozilla																																			
Bil.	Keperluan Persekitaran	Keterangan																																								
1.	<i>Production Environment</i>	URL: https://www.tempahan.gov.my Pelayar: Chrome / Mozilla																																								
Hasil Ujian	<p>Seksyen ini menerangkan dapatan hasil ujian seperti berikut:</p> <p>i) Isu-isu FAT yang dilaporkan</p> <p>ii) Kategori Isu-isu FAT (<i>Functionality, Usability or Operational</i>)</p> <p>iii) Status Isu-isu FAT</p> <p>iv) Ulasan Keputusan Ujian</p> <p>Contoh</p> <table border="1" data-bbox="472 763 1294 1167"> <thead> <tr> <th rowspan="2">Sumber</th> <th rowspan="2">Status</th> <th colspan="3">Severity</th> <th rowspan="2">Jumlah Isu</th> </tr> <tr> <th>High</th> <th>Medium</th> <th>Low</th> </tr> </thead> <tbody> <tr> <td rowspan="2">Pasukan Projek</td> <td><i>Open</i></td> <td>0</td> <td>2</td> <td>0</td> <td>2</td> </tr> <tr> <td><i>Closed</i></td> <td>0</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <th rowspan="2">Sumber</th> <th rowspan="2">Status</th> <th colspan="3">Type</th> <th>-</th> </tr> <tr> <th>Request</th> <th>Incident</th> <th></th> <th>-</th> </tr> <tr> <td rowspan="2">Pasukan Operasi</td> <td><i>Open</i></td> <td>1</td> <td>0</td> <td></td> <td>1</td> </tr> <tr> <td><i>Closed</i></td> <td>0</td> <td>1</td> <td></td> <td>1</td> </tr> </tbody> </table>	Sumber	Status	Severity			Jumlah Isu	High	Medium	Low	Pasukan Projek	<i>Open</i>	0	2	0	2	<i>Closed</i>	0	1	0	1	Sumber	Status	Type			-	Request	Incident		-	Pasukan Operasi	<i>Open</i>	1	0		1	<i>Closed</i>	0	1		1
Sumber	Status			Severity				Jumlah Isu																																		
		High	Medium	Low																																						
Pasukan Projek	<i>Open</i>	0	2	0	2																																					
	<i>Closed</i>	0	1	0	1																																					
Sumber	Status	Type			-																																					
		Request	Incident		-																																					
Pasukan Operasi	<i>Open</i>	1	0		1																																					
	<i>Closed</i>	0	1		1																																					
Rumusan	<p>Mengandungi penerangan rumusan hasil daripada ujian FAT.</p> <p>Contoh:</p> <p>i. Tiada isu yang dibangkitkan dalam tempoh FAT.</p> <p>ii. Laporan FAT ini merangkumi semua skop seperti yang telah ditetapkan.</p> <p>iii. FAT ini telah diperaku dan diterima oleh Kerajaan.</p>																																									

Templat **Laporan Penamatan Ujian** seperti dalam **D16 Laporan Penamatan Ujian**.

Langkah 4 : Sediakan Sijil Penerimaan Akhir

Sijil Penerimaan Akhir atau *Final Acceptance Certificate* (FAC) akan dikeluarkan oleh pemilik sistem kepada pembekal. Sijil ini menunjukkan bahawa pemilik telah bersetuju menerima sistem yang dibangunkan dan diinstalasi di persekitaran produksi. Tarikh sijil FAC ini

menandakan tamatnya pembangunan dan pengujian sistem dan bermulanya tempoh jaminan. Rujuk kepada **Apendiks 11 Contoh Sijil Penerimaan Akhir** .

RUJUKAN

1. ISO/IEC 29119-3:2013: *Software And Systems Engineering - Software Testing - Part 3: Test Documentation.*



7.7 PENYEDIAAN MANUAL PENGGUNA [F6.3]

KETERANGAN

Dokumentasi Manual Pengguna (MP) adalah dokumen yang bertindak sebagai panduan dan rujukan untuk pengguna memahami dan menggunakan aplikasi yang siap dibangunkan. Ianya mengandungi maklumat mengenai langkah-langkah penggunaan dan pengoperasian aplikasi secara terperinci.

Apabila berlaku sebarang perubahan atau penambahbaikan ke atas aplikasi, dokumen manual pengguna yang sedia ada perlulah kemaskini. Segala perubahan yang dilakukan pada dokumen manual pengguna mestilah direkodkan dan versi dokumen juga perlu dikemaskini. Rujuk kepada **D17 Manual Pengguna Sistem** untuk melihat format dokumen Manual Pengguna Sistem di mana pengisian kandungan-kandungannya adalah seperti langkah-langkah di bawah.

OBJEKTIF

- Membantu pengguna sistem untuk memahami proses kerja dan menggunakan setiap modul di dalam sistem dengan lancar dan sempurna
- Mewujudkan kesinambungan dalam penggunaan dan pengoperasian sistem aplikasi
- Memudahkan latihan pengguna dan latihan operasi dilaksanakan kepada pegawai-pegawai yang bertanggungjawab ke atas sistem aplikasi dari semasa ke semasa

LANGKAH-LANGKAH

Langkah 1 : Sediakan Pengenalan Bagi Sistem Yang Akan Dilaksanakan

Bahagian ini mengandungi perkara-perkara seperti berikut:

a) Tujuan dan Skop

Seksyen ini menerangkan tentang tujuan dan skop sistem aplikasi yang akan dilaksanakan.

b) Organisasi Manual

Seksyen ini menerangkan tentang bagaimana manual pengguna di susun untuk kemudahan pengguna.

c) Maklumat Untuk Dihubungi

Seksyen ini menerangkan tentang maklumat organisasi dan kakitangan yang boleh dihubungi dalam membantu pengguna berkaitan dengan penggunaan sistem. Jika kemudahan khidmat bantuan (*helpdesk*) di sesebuah organisasi adalah disediakan, maka ia perlu dijelaskan di bahagian ini.

d) Rujukan Projek

Seksyen ini menerangkan tentang bibliografi rujukan projek utama dan serahan yang telah dihasilkan sepanjang tempoh projek.

e) Fungsi Utama Sistem

Seksyen ini menjelaskan mengenai perspektif bisnes dan tanggungjawab pengguna yang akan disokong oleh sistem. Fungsi-fungsi bisnes perlu dijelaskan supaya pengguna dapat memahami objektif utama sistem aplikasi itu dibangunkan.

Langkah 2 : Sediakan Gambaran Keseluruhan Sistem

Bahagian ini memberikan gambaran keseluruhan ringkas mengenai sistem dan keupayaannya yang merangkumi perkara-perkara seperti berikut:

a) Tujuan

Seksyen ini menerangkan tentang tujuan sistem aplikasi dibangun dan dilaksanakan.

b) Keterangan Sistem

Seksyen ini memberikan gambaran keseluruhan mengenai keupayaan sistem, fungsi dan operasi serta fungsi peringkat tinggi yang akan dijalankan oleh sistem. Penggunaan gambarajah dan jadual boleh digunakan jika ia bersesuaian.

Langkah 3 : Sediakan Keterangan Fungsi Sistem

Bahagian ini menerangkan tentang setiap fungsi yang ada di dalam sistem. Ia akan menggambarkan berkenaan dengan konvensyen yang digunakan dalam subseksyen yang berkaitan. Bab ini merangkumi perkara-perkara seperti berikut :

a) Senarai Fungsi Sistem

Seksyen ini menyenaraikan nama bagi setiap fungsi yang ada di dalam sistem.

b) Perincian Keterangan Bagi Fungsi Sistem

Seksyen ini menyediakan keterangan secara terperinci bagi setiap fungsi sistem seperti berikut :

- i) Tujuan dan kegunaan fungsi sistem
- ii) Pengawalan fungsi sistem, jika berkenaan
- iii) Pilihan pelaksanaan yang berkaitan dengan fungsi sistem
- iv) Keterangan input fungsi
- v) Keterangan output dan hasil yang diharapkan
- vi) Hubungan dengan fungsi sistem yang lain
- vii) Ringkasan operasi fungsi sistem

Langkah 4 : Sediakan Arahan Penggunaan Sistem

Bahagian ini menyediakan arahan terperinci langkah demi langkah bagi kaedah pengoperasian sistem.

a) Log Masuk Sistem

Seksyen ini menerangkan berkaitan prosedur untuk log masuk sistem dan kawalan paparan sistem, seperti skrin menu utama sistem. Prosedur pengawalan hendaklah menerangkan bagaimana untuk menentukan mod yang diperlukan operasi dan menetapkan apa-apa parameter permulaan yang diperlukan untuk operasi. Prosedur pemasangan perisian perlu dimasukkan jika perisian diedarkan pada disket dan perlu dimuat turun terlebih dahulu sebelum digunakan.

b) Proses Pengoperasian Sistem

Seksyen ini menerangkan berkaitan prosedur untuk melaksanakan operasi perisian di mana tindak balas daripada pengguna diperlukan.

c) Penamatan dan Pengoperasian Semula Sistem

Seksyen ini menerangkan tentang prosedur untuk penamatan operasi sistem secara normal dan tidak berjadual bagi sistem. Di samping itu, ia juga perlu menentukan cara untuk memulakan semula sistem selepas operasi sistem ditamatkan.

Langkah 5 : Sediakan Maklumat Pengendalian Ralat

Bahagian ini menerangkan berkaitan mesej ralat dan kemudahan bantuan yang disediakan kepada pengguna. Ia merangkumi perkara-perkara seperti berikut:

- a) Menyatakan mesej ralat dan kemudahan bantuan. Maklumat tambahan dan subseksyen boleh ditambah seperti yang diperlukan. Masukkan senarai semua mesej ralat seperti berikut:
 - i) Mana-mana kod ralat angka yang berkaitan dengan mesej ralat
 - ii) Keterangan makna mesej ralat
 - iii) Perbincangan bagaimana untuk menyelesaikan ralat Bantuan Helpdesk
- b) Menerangkan apa-apa perisian bantuan atau apa-apa kemudahan khidmat bantuan atau kontraktor bantuan yang pengguna boleh hubungi untuk menyelesaikan ralat. Nombor telefon meja bantuan hendaklah dimasukkan.

RUJUKAN

1. <http://www.arbowebforest.com/android/ArboWebForestUserManual.pdf>
2. https://www.utdallas.edu/~chung/CS4351/.../TeamInitech/Technical_Manual_v2.docx

7.8 SERAHAN SISTEM APLIKASI [F6.4]

KETERANGAN

Aktiviti serahan sistem adalah aktiviti terakhir dalam fasa pelaksanaan sistem. Serahan Sistem Aplikasi dilaksanakan oleh pasukan pembangun kepada pemilik sistem. Ianya sering dilaksanakan secara rasmi menerusi sesi *sign off* projek.

Laporan Serahan sistem perlu disediakan oleh Pasukan Pembangun Sistem Aplikasi dan ianya sebagai dokumen akhir semasa sesi serahan sistem dari pasukan.

Serahan sistem aplikasi akan dilaksanakan setelah mendapat kelulusan Jawatankuasa Pemandu Projek/ Jawatankuasa Pemandu ICT Agensi ke atas Laporan Penamatan Ujian dan semua dokumen serahan telah dihasilkan.

OBJEKTIF

- Memberi tanggungjawab kepada pemilik projek untuk memiliki sistem aplikasi sebagai aset dalam menyokong proses/tugas organisasi.
- Penyerahan secara formal sistem aplikasi yang dibangunkan kepada Pemilik Sistem dan Pasukan Operasi untuk meneruskan pelaksanaan dan penyelenggaraan sistem aplikasi

LANGKAH-LANGKAH

Langkah 1 : Pastikan Semua Serahan Projek Telah Diselesaikan

- a) Pastikan semua serahan dokumen yang dipersetujui dalam Pelan Pembangunan Sistem dihasilkan dengan lengkap dan komprehensif oleh Pasukan Pembangun.

Contoh serahan projek pembangunan Sistem Aplikasi adalah seperti dalam **Jadual 2 - Keperluan Minima Dokumentasi Projek Pembangunan Sistem** mengikut fasa pembangunan sistem.

- b) Semak Laporan Penamatan Ujian.
c) Pakejkan sistem aplikasi yang telah lengkap selepas tempoh FAT.

Langkah 2 : Sediakan Laporan Serahan Sistem Aplikasi

a) Tujuan

Seksyen ini menerangkan tujuan dokumen serahan ini disediakan iaitu dengan menyatakan dokumen-dokumen serahan yang akan disampaikan dan sebagai perakuan penerimaan sistem.

b) Overview

Seksyen ini menerangkan ringkasan maklumat berkaitan komitmen awal projek pembangunan sistem antara pasukan projek dan pemilik sistem melalui **D01 Pelan Pembangunan Sistem** atau Piagam Projek yang dipersetujui. Maklumat termasuklah:

- i) Latarbelakang projek
- ii) Skop Projek
- iii) Serahan Projek
- iv) Jadual Pelaksanaan
- v) Keperluan sumber

c) Pencapaian dan Serahan

Seksyen ini menerangkan pencapaian projek pembangunan sistem aplikasi berdasarkan jadual perancangan pelaksanaan yang dipersetujui. Pencapaian termasuklah:

- i) Pencapaian aktiviti projek;
- ii) Pencapaian pembangunan Sistem; dan
- iii) Pencapaian persediaan dokumentasi serahan sistem.

d) Pendekatan Pembangunan

Seksyen ini menerangkan pendekatan pembangunan yang telah diambil sehingga kejayaan pembangunan sistem aplikasi mengikut keperluan pengguna. Seringkali pendekatan yang dinyatakan dalam **D01 Pelan Pembangunan Sistem** adalah berubah sekiranya perubahan dalam strategi pelaksanaan dan perubahan dalam struktur pasukan projek.

e) Penjimatan

Menyatakan penjimatan yang telah berlaku sepanjang pembangunan sistem aplikasi dilaksanakan. Penjimatan dalam aspek:

- i) Kos Peruntukan dan perbelanjaan
- ii) Sumber Manusia
- iii) Penggunaan *tools* dan teknik

f) Faedah Sistem Aplikasi

Seksyen ini menerangkan faedah jangkamasa pendek dan jangkamasa panjang kepada perkhidmatan penyampaian agensi dengan adanya atau laksananya sistem aplikasi yang dibangunkan menyokong dalam proses bisnes agensi.

g) Penerimaan Sistem

Seksyen ini adalah mengandungi perakuan serahan dan penerimaan serahan dokumen dan sistem aplikasi yang dibangunkan oleh pasukan projek/pembangun sistem kepada pemilik sistem

Perakuan dilakukan oleh sekurang-kurangnya dua (2) pihak:

- i) Pengurus/Pengarah Projek
- ii) Pemilik Projek/Sistem

Rujuk kepada **D18 Laporan Serahan Sistem**.

Langkah 3 : Aturkan Sesi Penyerahan Sistem Aplikasi

Sesi penyerahan sistem aplikasi mengambilkira perkara berikut:

- a) Penglibatan Pasukan Pembangun, Pemilik Sistem dan wakil SME.
- b) Penyerahan bersama-sama dokumen serahan yang dipersetujui dan dalam D01 Pelan Pembangunan Sistem.
- c) Dilakukan dalam sesi *Sign-Off* antara Pasukan Pembangun dan Pemilik Sistem



BAB:08

PENGIRAAN SAIZ APLIKASI

Bab ini menerangkan penggunaan kaedah *Function Point Analysis* (FPA) untuk menentukan saiz sistem aplikasi dalam fasa-fasa Permulaan Projek, Analisis dan Pelaksanaan.



8 PENGIRAAN SAIZ SISTEM APLIKASI

8.1 PENGENALAN KEPADA *FUNCTION POINTS*

Pada pertengahan tahun 1970an, Allan Albrecht daripada syarikat IBM memperkenalkan *Functional Point Analysis* (FPA) sebagai kaedah penentuan saiz sistem aplikasi yang dibangunkan. FPA yang telah diperkenalkan oleh Allan Albrecht ini kemudiannya ditambahbaik oleh IFPUG. Berbeza dengan kaedah pengiraan berasaskan bilangan baris kod sumber dan kaedah-kaedah lain, kaedah FPA mengukur saiz sistem aplikasi berdasarkan kefungsiannya yang diminta dan diterima oleh pengguna dan kaedah ini bebas dari faktor-faktor teknikal dan teknologi. Oleh yang demikian, pengiraan saiz sistem aplikasi kekal konsisten walau pun sistem aplikasi yang dibangunkan menggunakan platform dan teknologi yang berlainan. FPA menyediakan metrik pengiraan saiz sistem aplikasi yang spesifik dan lebih tepat dengan margin ralat +/- 10%. Metrik atau unit pengukuran saiz fungsi perisian atau sistem aplikasi dinamakan *function points* (FP). Jadi 1FP bermaksud saiz kefungsiannya sistem aplikasi yang dikira ialah satu FP. FP merupakan unit pengukuran saiz sistem aplikasi seperti mana meter atau kaki persegi untuk mengukur saiz bangunan, jam untuk mengukur masa, kilometer atau batu untuk mengukur jarak, celsius untuk mengukur suhu atau byte untuk mengukur saiz storan. Secara ringkasnya, kaedah pengukuran saiz fungsi sistem aplikasi IFPUG dikenali sebagai *Function Points Analysis* (FPA) dan unit saiz fungsinya dinamakan *function points* (FP).

8.2 PENGIRAAN SAIZ FUNGSIAN SISTEM APLIKASI

8.2.1 Definisi

Kaedah pengiraan saiz fungsi sistem aplikasi yang ditetapkan oleh IFPUG mengukur saiz ke atas dua kategori fungsi sistem iaitu fungsi transaksi dan fungsi data. Fungsi transaksi merujuk kepada fungsi-fungsi transaksi asas yang melaksanakan proses menyimpan, mengemaskini, menghapus dan mempamir data logikal. Fungsi data merujuk kepada data logikal yang telah disimpan dan tersedia untuk dikemaskini dan dicapai. Setiap fungsi ini mempunyai komponen-komponen yang digunakan dalam pengiraan saiz fungsi sistem. Penentuan klasifikasi setiap komponen ini perlu menurut peraturan-peraturan dan panduan yang telah ditetapkan oleh IFPUG. Buku panduan ini tidak menerangkan secara terperinci kaedah, peraturan dan panduan yang digunakan dalam FPA (*Function Points Analysis*). Panduan FPA yang spesifik boleh didapati dengan merujuk kepada literatur berkaitan FPA dan panduan yang dikeluarkan oleh IFPUG. Walau bagaimanapun sebagai panduan asas, di bawah diberikan definisi ringkas setiap komponen fungsi tersebut sebagaimana ditakrifkan oleh IFPUG.

8.2.1.1 Komponen Fungsi

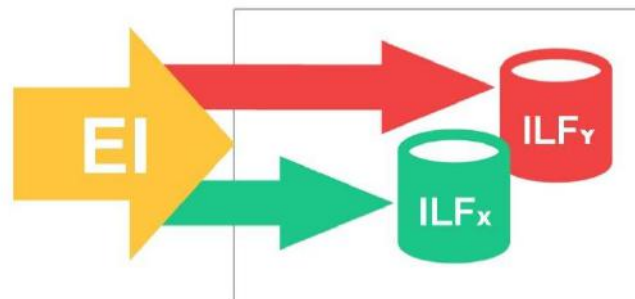
a) **Internal Logical Files (ILF's)**

Berasaskan definisi ini, ILF ialah fail-fail, jadual-jadual (*table*) dalam pangkalan data atau kumpulan-kumpulan data yang dimiliki dan diselenggara oleh sistem aplikasi yang dibangun.

b) **External Interface Files (EIF's)**

Berasaskan definisi ini, EIF ialah fail-fail, jadual-jadual (*table*) dalam pangkalan data atau kumpulan-kumpulan data yang dimiliki dan diselenggara oleh sistem aplikasi lain tetapi dirujuk oleh sistem yang dibangun.

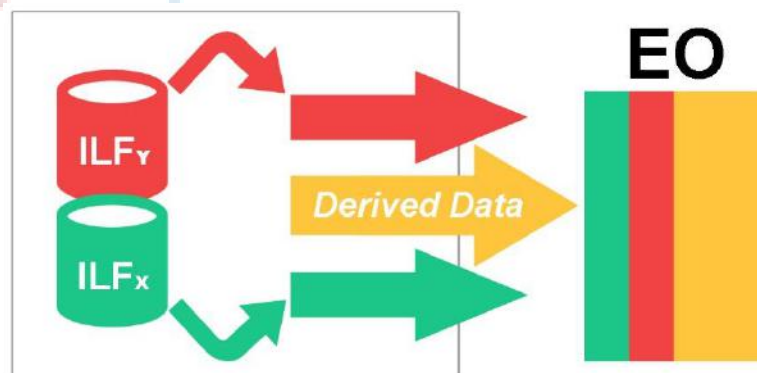
c) **External Inputs (EI)**



Rajah 90 : Komponen Fungsi EI

Berasaskan definisi ini, EI ialah fungsi transaksi asas yang menyimpan, mengemaskini dan/atau menghapus data dalam ILF. Fungsi transaksi asas ini mungkin dalam bentuk skrin input atau antaramuka dengan sistem atau peralatan lain.

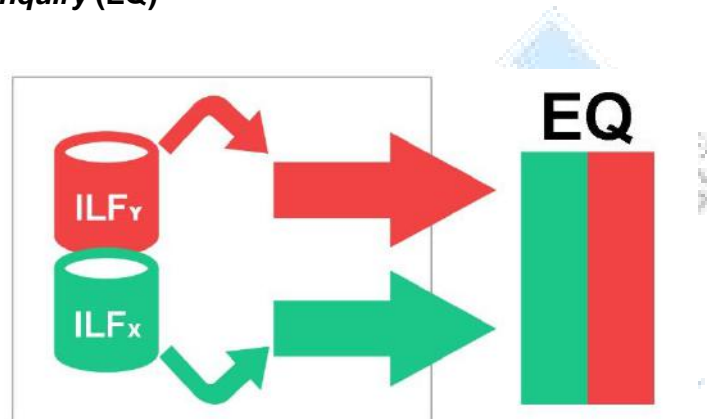
d) **External Outputs (EO)**



Rajah 91 : Komponen Fungsi EO

Berasaskan definisi ini, EO ialah fungsi transaksi asas yang mencapai data daripada ILF, memproses dan menghantar data ke luar sempadan sistem. EO juga mungkin melibatkan kemasukan, pengemaskinian dan penghapusan data dalam ILF. Fungsi transaksi asas ini mungkin menghasilkan laporan atau output kepada sistem atau peralatan lain. Gambarajah di atas menunjukkan EO dengan 2 ILF dan data yang dihasilkan melalui logik pemrosesan (*derived data*).

e) **External Inquiry (EQ)**



Rajah 92 : Komponen Fungsi EQ

Berasaskan definisi ini, EQ ialah fungsi transaksi asas yang mencapai data daripada ILF dan menghantar data ke luar sempadan sistem. EO tidak melibatkan kemasukan, pengemaskinian dan penghapusan data dalam ILF. Fungsi transaksi asas ini mungkin menghasilkan laporan atau output kepada sistem atau peralatan lain. Gambarajah di atas menunjukkan EO dengan 2 IFL dan tanpa data yang dihasilkan melalui logik pemrosesan.

f) **Files Type Reference (FTR)**

File type reference (FTR) ialah objek atau jenis fail yang dirujuk oleh transaksi di dalam EI, EO dan EQ. FTR adalah terdiri sama ada ILF atau EIF.

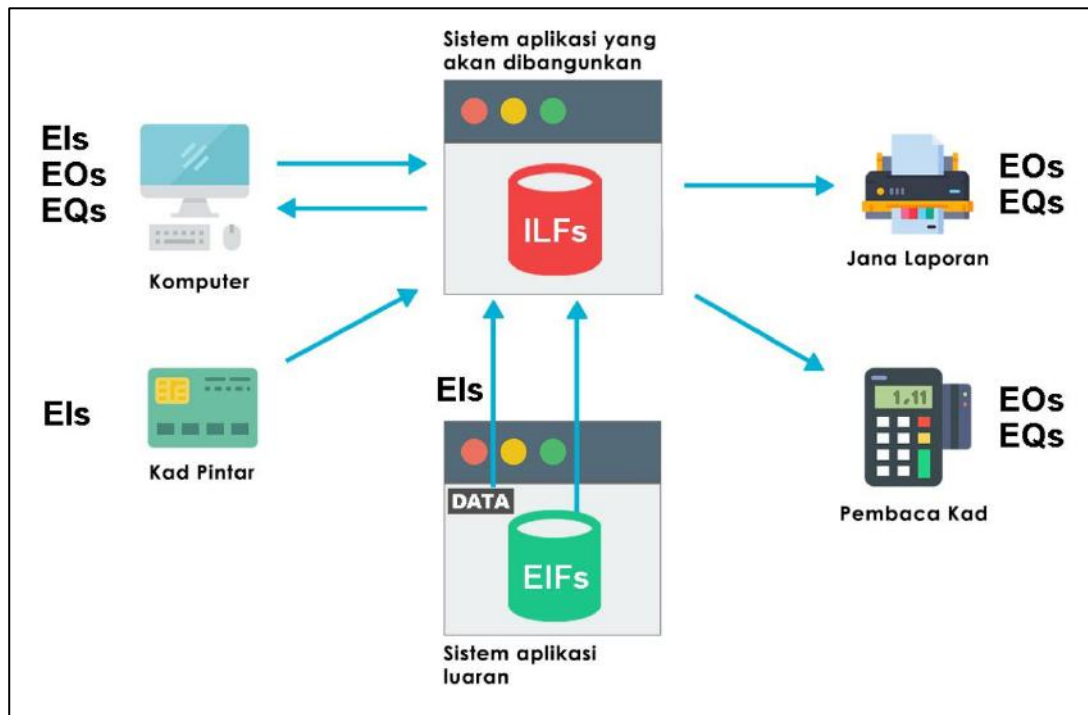
g) **Data Element Types (DET)**

Data element types (DET) adalah merupakan *field* yang dinamik, *user recognizable* dan unik (*non-repetitive*). Data elemen boleh terdiri daripada bentuk kuantitatif seperti nombor, dan bentuk kualitatif seperti teks, foto, video dan audio.

h) **Record Element Type (RET)**

Record Element Type (RET) adalah merupakan sub kumpulan kepada elemen data yang terkandung di dalam satu-satu ILF atau EIF.

Tahap kompleksiti EI, EO dan EQ ditentukan oleh bilangan *Files Type Reference* (FTR) dan bilangan *Data Element Types* (DET). Dalam konteks pengurusan pangkalan data, FTR ialah *table* dan DET ialah *field* atau *column*. Bagi ILF dan EIF pula tahap kompleksitinya ditentukan oleh bilangan RET and bilangan DET. Dalam model ERD, RET ialah entiti dan DET ialah atribut kepada entiti. Kaedah dan peraturan penentuan ILF, EIF, EI, EO, EQ, FTR, RET dan DET hendaklah dirujuk dalam panduan yang dikeluarkan oleh IFPUG. Secara keseluruhannya, gambarajah di bawah menunjukkan bagaimana komponen-komponen tersebut diklasifikasikan.



Rajah 93 : Gambaran Komponen-Komponen Fungsi Di Dalam Sistem

8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan

a) Jadual Fungsi Transaksi

Jadual Matriks Kompleksiti dan Jadual Penterjemahan bagi Fungsi Transaksi adalah seperti di bawah:

Jadual 98 : Matriks Kompleksiti EI

	1 - 4 DET	5 - 15 DET	16 atau lebih DET
0 - 1 FTR	Rendah	Rendah	Sederhana
2 FTR	Rendah	Sederhana	Tinggi
3 atau lebih FTR	Sederhana	Tinggi	Tinggi

Jadual 99 : Matriks Kompleksiti EO dan EQ

	1 - 5 DET	6 - 19 DET	20 atau lebih DET
0 - 1 FTR	Rendah	Rendah	Sederhana
2 - 3 FTR	Rendah	Sederhana	Tinggi
4 atau lebih FTR	Sederhana	Tinggi	Tinggi

Jadual 100 : Penterjemahan Saiz EI dan EQ

Tahap Kompleksiti Fungsi Transaksi	Function Points
Rendah	3
Sederhana	4
Tinggi	6

Jadual 101 : Penterjemahan Saiz EO

Tahap Kompleksiti Fungsi Transaksi	Function Points
Rendah	4
Sederhana	5
Tinggi	7

b) Jadual Fungsi Data

Jadual Matriks Kompleksiti dan Jadual Penterjemahan bagi Fungsi Data adalah seperti di bawah:

Jadual 102 : Matriks Kompleksiti ILF dan EIF

	1 - 19 DET	20 - 50 DET	51 atau lebih DET
1 RET	Rendah	Rendah	Sederhana
2 - 5 RET	Rendah	Sederhana	Tinggi
6 atau lebih RET	Sederhana	Tinggi	Tinggi

Jadual 103 : Penterjemahan Saiz ILF

Tahap Kompleksiti Fungsi Data	Function Points
Rendah	7
Sederhana	10
Tinggi	15

Jadual 104 : Penterjemahan Saiz EIF

Tahap Kompleksiti Fungsi Data	Function Points
Rendah	5
Sederhana	7
Tinggi	10

8.2.1.3 Empat Belas Ciri-ciri Am Sistem (GSC)

Pengiraan saiz ini berasaskan kepada kompleksiti keperluan fungsian sistem yang diterima oleh pengguna. Faktor-faktor dari aspek keperluan teknikal dan kualiti yang mempengaruhi kompleksiti sistem juga perlu diambil kira. IFPUG menetapkan 14 faktor teknikal dan kualiti seperti dalam jadual di bawah.

Jadual 105 : 14 Ciri-ciri Am Sistem (GSC)

Bil.	Faktor	Keterangan
1	Komunikasi Data	Berapa banyak fasiliti komunikasi yang ada untuk membantu pemindahan atau pertukaran maklumat dengan sistem aplikasi?
2	Pemprosesan Data Teragih	Bagaimana data teragih (<i>distributed data</i>) dan fungsi pemprosesan dikendalikan?
3	Prestasi	Adakah pengguna memerlukan maklumat berkenaan masa tindakbalas dan daya pemprosesan (<i>throughput</i>)?
4	Konfigurasi yang Kerap Digunakan	Berapa kerap platform perkakasan sedia ada akan digunakan untuk melaksanakan sistem aplikasi pada masa akan datang?
5	Kadar Transaksi	Berapa kerap transaksi dilaksanakan dalam masa sehari, seminggu, sebulan dan sebagainya?
6	Kemasukan Data Dalam Talian	Apakah peratusan maklumat yang direkodkan secara dalam talian?
7	Efisiensi Pengguna	Adakah aplikasi direkabentuk berdasarkan efisiensi pengguna?

8	Pengemaskinian Dalam Talian	Berapa banyak ILF yang dikemaskini melalui transaksi dalam talian?
9	Pemprosesan yang Kompleks	Adakah sistem aplikasi yang akan dibangunkan mengandungi logikal dan pemprosesan matematik yang kompleks?
10	<i>Reusability</i>	Adakah aplikasi dibangunkan bertujuan untuk memenuhi keperluan seseorang pengguna atau ia mengambil kira juga keperluan pengguna-pengguna yang lain?
11	<i>Installation Ease</i>	Berapa sukar proses instalasi yang akan dilaksanakan?
12	<i>Operational Ease</i>	Apakah tahap keberkesanan dan automasi bagi prosedur-prosedur <i>start-up</i> , <i>back-up</i> dan pemulihan?
13	Lokasi	Adakah sistem aplikasi direkabentuk, dibangun dan menyokong kepada pemasangan di pelbagai lokasi dan organisasi?
14	Perubahan Fasiliti	Adakah sistem aplikasi direkabentuk, dibangun dan menyokong kepada perubahan fasiliti?

Faktor-faktor yang dinyatakan di atas merupakan empat belas (14) ciri-ciri am sistem (GSC) yang ditetapkan oleh IFPUG untuk mengukur kefungsiian teknikal dan kualiti sistem. Setiap karakteristik ini mempengaruhi kompleksiti sistem dan tahap pengaruhnya ditetapkan mengikut 6 skil ukuran iaitu 0 (*Not present/no influence*), 1 (*Incidental influence*), 2 (*moderate influence*), 3 (*average influence*) dan 4 (*significant influence*) dan 5 (*strong influence throughout*). Keterangan terperinci faktor-faktor tersebut dan peraturan bagi menetapkan tahap kesan dan pengaruh setiap faktor ke atas kompleksiti sistem perlu dirujuk dalam panduan pengiraan yang disediakan oleh IFPUG.

8.2.2 Pengiraan *Value Adjustment Factor (VAF)*

Pengaruh empat belas (14) ciri-ciri am sistem (GSC) digunakan untuk mendapatkan *Value Adjustment Factor (VAF)* dengan menggunakan formula IFPUG seperti berikut:

$$\text{VAF} = 0.65 + [(\sum C_i) / 100]$$

$i = 1$ to 14 mewakili bilangan item GSC

C_i = kadar pengaruh setiap item GSC

\sum = jumlah kadar pengaruh semua 14 item GSC

8.2.3 Pengiraan *Unadjusted Function Points (UFP)*

Berpandukan kaedah pengiraan dan jadual kompleksiti yang telah diterangkan di atas, pengiraan UFP boleh dikira dengan menggunakan jadual pengiraan dan contoh di bawah.

Jadual 106 : Formula Pengiraan UFP

Komponen Fungsi	Jumlah Fungsi Transaksi / Fungsi Data	Kompleksiti			Saiz (FP)
		Rendah	Sederhana	Tinggi	
Inputs (EI)	$EI_n = \sum EI_r + \sum EI_s + \sum EI_t$	$\sum EI_r \times 3$	$\sum EI_s \times 4$	$\sum EI_t \times 6$	FP1
Outputs (EO)	$EO_n = \sum EO_r + \sum EO_s + \sum EO_t$	$\sum EO_r \times 4$	$\sum EO_s \times 5$	$\sum EO_t \times 7$	FP2
Queries (EQ)	$EQ_n = \sum EQ_r + \sum EQ_s + \sum EQ_t$	$\sum EQ_r \times 3$	$\sum EQ_s \times 4$	$\sum EQ_t \times 6$	FP3
Internal Logical Files (ILF)	$ILF_n = \sum ILF_r + \sum ILF_s + \sum ILF_t$	$\sum ILF_r \times 7$	$\sum ILF_s \times 10$	$\sum ILF_t \times 15$	FP4
External Interface Files (EIF)	$EIF_n = \sum EIF_r + \sum EIF_s + \sum EIF_t$	$\sum EIF_r \times 5$	$\sum EIF_s \times 7$	$\sum EIF_t \times 10$	FP5
Jumlah Fungsi	$EI_n + EO_n + EQ_n + ILF_n + EIF_n$				
Jumlah <i>Unadjusted Function Points (UFP)</i>		$FP1 + FP2 + FP3 + FP4 + FP5$			

8.2.4 Pengiraan *Adjusted Function Points (AFP)*

Kompleksiti sesebuah sistem aplikasi tidak hanya bergantung komponen-komponen sistem yang digunakan dalam pengiraan UFP tetapi juga bergantung kepada faktor-

faktor lain. Bagi mendapatkan saiz sistem yang lebih realistik, faktor-faktor lain seperti *end-user efficiency*, *reusability*, aspek prestasi dan lain-lain juga perlu diambil kira. Faktor-faktor ini digunakan untuk mendapatkan nilai *value adjusted factor* (VAF). Saiz sistem aplikasi yang realistik atau *adjusted function points* (AFP) dapat dikira menggunakan formula di bawah:

$$\text{AFP} = \text{UFP} \times \text{VAF}$$

Oleh kerana faktor-faktor yang digunakan untuk mengira VAF belum dapat dikenalpasti di peringkat kajian keperluan bisnes, maka nilai VAF boleh diambil antara 0.65 hingga 1.00 bagi sistem yang kecil dan 1.01 hingga 1.35 bagi sistem sederhana dan besar bergantung kepada tahap kompleksiti sistem tersebut. Saiz sistem samada kecil atau besar boleh diandaikan berpandukan kepada UFP iaitu sistem bersaiz kecil sekiranya kurang daripada 100FP, bersaiz sederhana bagi sistem aplikasi bersaiz antara 100FP hingga 999FP dan bersaiz besar bagi sistem aplikasi bersaiz 1000FP ke atas.

Jadual 107 : Formula Pengiraan AFP

Komponen Fungsi	Jumlah Fungsi Transaksi / Fungsi Data	Kompleksiti			Saiz (FP)
		Rendah	Sederhana	Tinggi	
Inputs (EI)	$EI_n = \sum EI_r + \sum EI_s + \sum EI_t$	$\sum EI_r \times 3$	$\sum EI_s \times 4$	$\sum EI_t \times 6$	FP1
Outputs (EO)	$EO_n = \sum EO_r + \sum EO_s + \sum EO_t$	$\sum EO_r \times 4$	$\sum EO_s \times 5$	$\sum EO_t \times 7$	FP2
Queries (EQ)	$EQ_n = \sum EQ_r + \sum EQ_s + \sum EQ_t$	$\sum EQ_r \times 3$	$\sum EQ_s \times 4$	$\sum EQ_t \times 6$	FP3
Internal Logical Files (ILF)	$ILF_n = \sum ILF_r + \sum ILF_s + \sum ILF_t$	$\sum ILF_r \times 7$	$\sum ILF_s \times 10$	$\sum ILF_t \times 15$	FP4
External Interface Files (EIF)	$EIF_n = \sum EIF_r + \sum EIF_s + \sum EIF_t$	$\sum EIF_r \times 5$	$\sum EIF_s \times 7$	$\sum EIF_t \times 10$	FP5
Jumlah Fungsi	$EI_n + EO_n + EQ_n + ILF_n + EIF_n$				
Jumlah Unadjusted Function Points (UFP)		$FP1 + FP2 + FP3 + FP4 + FP5$			
Value Adjusted Factor (VAF)		0.65 ≤ VAF ≤ 1.00 atau 1.01 ≤ VAF ≤ 1.35			
Jumlah Adjusted Function Points (AFP)		UFP x VAF			

8.2.5 Pengiraan Anggaran *Effort* (man hours) dan Kos Pembangunan Sistem

Selepas jumlah AFP didapati, anggaran *effort*, saiz sumber manusia dan kos pembangunan sistem boleh dilakukan. Anggaran ini boleh dibuat berdasarkan kepada standard kadar produktiviti (FP per masa), bilangan kategori sumber manusia per FP dan kos per FP. Anggaran ini lebih tepat sekiranya standard yang digunakan ialah berdasarkan pengalaman pasukan pembangunan agensi. Sekiranya tiada standard di peringkat agensi atau sektor awam, standard antarabangsa atau negara-negara lain yang bersesuaian boleh digunakan. Mengikut *International Software Benchmarking Standard Group* (ISBSG), kadar produktiviti adalah antara 8 hingga 11 *man-hours* per FP, kos pembangunan per FP mengikut negara adalah USD125/FP (India), USD180/FP (Thailand) dan USD185/FP (Indonesia). Menurut kajian Capers Jones pada tahun 2017, kadar produktiviti pembangunan sistem di Malaysia ialah 11.73 FP sebulan dan 15 *man-hours* per FP. Standard ini atau mana-mana standard yang terkini boleh digunakan sebagai panduan dalam membuat anggaran keperluan sumber yang diperlukan dalam proses pembangunan sistem aplikasi. Berdasarkan kepada standard ini, pengiraan *effort* dan masa pembangunan adalah seperti berikut:

$$\begin{aligned} \text{Effort Pembangunan Sistem} &= (\text{Adjusted Function Points} \times \text{Kadar Produktiviti} \\ &\quad \text{man-hours di Malaysia}) \\ &= (\text{AFP} \times 15 \text{ man-hours}) \end{aligned}$$

$$\begin{aligned} \text{Masa Pembangunan Sistem} &= \text{Adjusted Function Points} / \\ &\quad \text{Kadar Produktiviti FP Sebulan di Malaysia} \\ &= \text{AFP} / 11.73 \text{ FP} \end{aligned}$$

* 1 Hari Bekerja = 8 Jam Bekerja, 1 Bulan Bekerja = 22 Hari Bekerja

Pengiraan kos pembangunan boleh dilakukan sama ada merujuk kepada kos pembangunan per FP berdasarkan kepada *International Software Benchmarking Standard Group* (ISBSG); atau dengan berpandukan kepada kos pasaran semasa industri bagi bilangan *mandays* yang telah diperolehi dari nilai FP.

Bagi pengiraan kos pembangunan per FP berdasarkan kepada *International Software Benchmarking Standard Group* (ISBSG), kadar kos pembangunan per FP boleh dirujuk kepada kadar bagi negara-negara serantau asia negara seperti Indonesia dan Thailand. Rujukan ini dilakukan oleh kerana maklumat kos pembangunan per FP bagi Malaysia pada ketika masih belum lagi diterbitkan oleh ISBSG. Formula pengiraan kos pembangunan sistem mengikut kadar ISBSG adalah seperti berikut:

$$\begin{aligned} \text{Kos Pembangunan Sistem} &= \text{Adjusted Function Points} \times \text{Kos} \\ \text{(ISBSG)} &\quad \text{Pembangunan Per FP Indonesia} \times \text{Nilai} \\ &\quad \text{Pertukaran USD kepada Ringgit Semasa} \\ &= \text{AFP} \times \text{USD185.00} \times \text{RM4.00} \end{aligned}$$

* Kos pembangunan per FP USD185.00 adalah berdasarkan kepada kos pembangunan per FP negara Indonesia

Bagi pengiraan kos pembangunan mengikut *mandays*, kos bagi setiap *mandays* yang diperolehi dengan menggunakan kaedah FP boleh berpandukan sama ada kepada kos pasaran semasa industri atau kos *mandays* yang ditentukan oleh agensi. Formula pengiraan kos pembangunan sistem mengikut kadar *mandays* adalah seperti berikut:

Kos Pembangunan Sistem (Mandays) = *Effort* Pembangunan Sistem x Kos Per *Mandays*



8.3 LANGKAH-LANGKAH PENGIRAAN SAIZ FUNGSIAN SISTEM APLIKASI DI FASA PERMULAAN PROJEK

Pengiraan saiz sistem aplikasi di Fasa Permulaan Projek adalah merupakan pengiraan awal saiz bagi membantu satu-satu organisasi membuat anggaran sumber, masa serta perancangan lain yang diperlukan bagi pembangunan sistem aplikasi terlibat. Penentuan dan pengiraan saiz sistem aplikasi di Fasa Permulaan Projek akan merujuk kepada model-model yang dibangunkan di dalam fasa ini, iaitu Rajah Hierarki Bisnes - **Pemodelan Fungsi Bisnes [F1.3]**, Rajah Aliran Proses dan Definisi Aktiviti Fungsi Bisnes - **Pemodelan Proses Bisnes [F1.4]**. Langkah-langkah pengiraan saiz sistem aplikasi di Fasa Permulaan Projek adalah seperti berikut:

a) Kenalpasti Kompleksiti Komponen Fungsi bagi Fungsi Data

i) Tentukan bilangan ILF / EIF

Penentuan bilangan ILF/EIF adalah berdasarkan kepada bilangan kumpulan maklumat yang telah disenaraikan di dalam Definisi Aktiviti Fungsi Bisnes. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan ILF/EIF di dalam Fasa Permulaan Projek:

- Setiap kumpulan maklumat boleh dianggap mempunyai hubungan *independent* di antara satu sama lain dan setiapnya mempunyai ILF/EIF yang berasingan. Namun begitu, andaian hubungan ini tidak terpakai sekiranya terdapat hubungan di antara kumpulan-kumpulan maklumat berkenaan telah jelas dikenalpasti.
- Individu yang melaksanakan penentuan ILF/EIF perlu juga melakukan andaian untuk merangkumkan bersekali kumpulan-kumpulan maklumat tambahan, seperti jadual rujukan (*look up table*), sekiranya maklumat tersebut tidak disenaraikan di dalam Definisi Aktiviti Fungsi Bisnes.
- Penentuan dan pengiraan bilangan ILF/EIF bagi kumpulan maklumat yang sama tidak boleh dilakukan secara berulang kali.

ii) Tentukan bilangan RET

Penentuan bilangan RET adalah berdasarkan kepada bilangan kumpulan-kumpulan maklumat yang berhubung kait di dalam satu-satu ILF/EIF. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan RET di dalam Fasa Permulaan Projek:

- Pengiraan bilangan RET boleh dilakukan dengan menganggap bahawa setiap ILF/EIF bagi kumpulan-kumpulan maklumat yang terlibat hanya mempunyai bilangan RET sebanyak 1 sahaja. Anggaran ini dibuat oleh kerana hubungan di antara kumpulan-kumpulan maklumat masih belum lagi dapat dikenalpasti secara terperinci pada fasa ini, melainkan individu yang melakukan pengiraan

adalah serba mahir dalam bidang pemodelan keperluan maklumat serta mampu untuk melakukan andaian kepada struktur pangkalan data yang akan dibangunkan pada fasa-fasa berikutnya.

- Penentuan dan pengiraan bilangan RET bagi kumpulan maklumat yang sama tidak boleh dilakukan berulang kali walaupun kumpulan maklumat tersebut berhubung kait di bawah ILF/EIF yang berbeza.

iii) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Data adalah berdasarkan kepada bilangan data di bawah satu-satu ILF/EIF dengan merujuk kepada kumpulan maklumat yang telah dikenalpasti di dalam Definisi Aktiviti Fungsi Bisnes. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan DET di dalam Fasa Permulaan Projek:

- Senaraikan dan kira bilangan data-data yang terkandung di bawah satu-satu kumpulan maklumat termasuk kumpulan maklumat tambahan yang telah diandaikan sebelum ini. Setiap data yang telah dikenalpasti mempunyai nilai DET sebanyak 1 sahaja.
- Data-data yang dianggap sebagai kunci primer (*primary key*) bagi setiap kumpulan maklumat tidak akan ditentukan sebagai DET sekiranya data tersebut hanya merupakan artifak teknikal di mana ia tidak memberi makna kepada bisnes dan bukan terdiri dari maklumat yang boleh difahami oleh pengguna (*non user-recognizable*). Sekiranya data-data kunci primer tersebut digunakan sebagai kunci sekunder di dalam kumpulan maklumat yang lain, data kunci sekunder tersebut akan dianggap sebagai DET.
- Penentuan dan pengiraan bilangan DET bagi data yang sama tidak boleh dilakukan berulang kali melainkan data yang sama juga diletakkan di dalam kumpulan maklumat yang berbeza.

iv) Tentukan kompleksiti setiap ILF / EIF

Berdasarkan kepada bilangan ILF/EIF, FTR dan DET yang telah ditentukan, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Data seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap komponen fungsi yang telah dikenalpasti.

v) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama kumpulan maklumat, data yang terlibat, bilangan ILF/EIF, FTR, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 108 : Penentuan Tahap Kompleksiti Fungsi Data

Kumpulan Maklumat	Data	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

b) Kenalpasti kompleksiti komponen fungsi bagi Fungsi Transaksi

i) Kenalpasti Fungsi Transaksi Asas

Berdasarkan kepada Rajah Hierarki Fungsi yang telah disediakan di dalam Pemodelan Fungsi Bisnes [F.1.2], kenalpasti setiap Fungsi Transaksi Asas (*Elementary Function*) yang terlibat dengan bisnes dan sistem yang ingin dibangunkan.

ii) Tentukan bilangan EI, EO dan EQ

Penentuan bilangan komponen fungsi EI, EO dan EQ berdasarkan kepada interaksi pengguna dengan aktiviti-aktiviti bisnes di dalam Rajah Aliran Proses dan Definisi Aktiviti Fungsi Bisnes. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan EI, EO dan EQ di dalam Fasa Permulaan Projek:

- Bagi setiap aktiviti bisnes yang telah dikenalpasti di dalam Rajah Aliran Proses, tentukan sama ada aktiviti-aktiviti tersebut adalah terdiri daripada komponen fungsi EI, EO atau/dan EQ.
- Setiap aktiviti bisnes di dalam Rajah Aliran Proses boleh mengandungi lebih dari satu jenis komponen fungsi EI, EO atau EQ.
- Jenis komponen fungsi yang sama tidak boleh bertindih dan dikira berulang kali di bawah satu-satu aktiviti bisnes yang sama.

iii) Tentukan bilangan FTR

Penentuan bilangan FTR adalah berpandukan kepada bilangan dan hubungan di antara kumpulan-kumpulan maklumat yang terlibat di dalam transaksi aktiviti bisnes bagi setiap komponen fungsi EI, EO dan EQ yang telah dikenalpasti. Rujuk kepada Definisi Aktiviti Fungsi Bisnes dan juga senarai ILF/EIF yang telah ditentukan di dalam Fungsi Data untuk menentukan bilangan kumpulan maklumat yang terlibat serta hubungannya di antara satu sama lain. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Permulaan Projek:

- Berdasarkan kepada senarai ILF/EIF di dalam Fungsi Data, kenalpasti bilangan dan hubungan di antara kumpulan-kumpulan maklumat yang berinteraksi dengan aktiviti bisnes di bawah satu-satu komponen fungsi EI, EO atau EQ.
- Jumlah bilangan komponen fungsi ILF/EIF yang terlibat dengan aktiviti bisnes berkenaan adalah merupakan bilangan FTR yang akan diperolehi.

iv) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Transaksi adalah bergantung kepada bilangan data yang digunakan di dalam transaksi setiap komponen fungsi EI, EO atau EQ bagi satu-satu aktiviti bisnes. Rujuk kepada ruangan Pengguna Maklumat di dalam Definisi Aktiviti Fungsi Bisnes untuk mengetahui maklumat/data yang terlibat bagi setiap aktiviti bisnes terlibat. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Permulaan Projek:

- Bagi komponen fungsi EI, kirakan bilangan DET yang hanya melibatkan data yang dimasukkan, dikemaskini atau/dan dihapuskan.
- Bagi komponen fungsi EO, kirakan bilangan DET yang hanya melibatkan data data yang dipaparkan melalui pengiraan atau logik pengaturcaraan.
- Bagi komponen fungsi EQ, kirakan bilangan DET yang hanya melibatkan data-data yang dipaparkan secara terus tanpa melalui pengiraan atau logik pengaturcaraan.
- Berbeza dengan pengiraan bilangan DET di dalam Fungsi Data, penentuan dan pengiraan bilangan DET bagi Fungsi Transaksi boleh dilakukan berulang kali bagi data-data yang sama di bawah fungsi atau aktiviti bisnes yang berbeza.

v) Tentukan kompleksiti setiap EI, EO dan EQ

Berdasarkan kepada bilangan EI, EO, EQ, FTR dan DET yang telah ditentukan, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Transaksi seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap komponen fungsi yang telah dikenalpasti.

vi) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama aktiviti, kumpulan maklumat dan data yang terlibat, bilangan EI, EO, EQ, RET, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 109 : Penentuan Tahap Kompleksiti Fungsi Transaksi

Nama Aktiviti	Maklumat	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

c) Kirakan *Value Adjustment Function* (VAF)

- i) Tentukan dahulu kadar pengaruh bagi empat belas (14) Ciri-Ciri Am Sistem (GSC) seperti yang disenarai dan diperjelaskan di dalam 8.2.1.3 Empat Belas Ciri-ciri Am Sistem (GSC). Tambahkan skor-skor yang telah diberikan kepada ciri-ciri am berkenaan untuk mendapatkan nilai ΣCi .
- ii) Dengan berpandukan kepada formula pengiraan seperti yang diterangkan di dalam 8.2.2 Pengiraan *Value Adjustment Factor*, masukkan nilai ΣCi yang telah diperolehi ke dalam formula di bawah untuk mendapatkan nilai VAF.

$$VAF = 0.65 + [(\Sigma Ci) / 100]$$

d) Kirakan *Unadjusted Function Points* (UFP)

- i) Berdasarkan maklumat kompleksiti yang telah diperolehi melalui penentuan berdasarkan Fungsi Transaksi dan Fungsi Data, rujuk kepada Jadual Penterjemahan Saiz seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk mendapatkan nilai *Function Points* (FP) bagi setiap komponen fungsi yang telah dikenal pasti.
- ii) Gunakan jadual serta formula pengiraan seperti di dalam 8.2.3 Pengiraan *Unadjusted Function Points* untuk mendapatkan nilai UFP.

e) Kirakan *Adjusted Function Points* (AFP)

Berpandukan kepada nilai **VAF** dan **UFP**, gunakan formula pengiraan seperti di **8.2.4 Pengiraan *Adjusted Function Points* (AFP)** untuk mendapatkan nilai AFP.

f) Kirakan Anggaran *Effort*, Masa dan Kos Pembangunan

Berikutan daripada nilai AFP yang telah diperolehi, nilai tersebut akan digunakan untuk melakukan pengiraan anggaran *effort*, masa dan kos pembangunan keseluruhan sistem aplikasi ataupun fungsi-fungsi di dalamnya. Rujuk 8.2.5 Pengiraan Anggaran *Effort* dan Kos Pembangunan Sistem untuk mendapatkan keterangan lanjut berkenaan dengan formula-formula pengiraan *effort*, masa dan kos.



8.4 LANGKAH-LANGKAH PENGIRAAN SAIZ FUNGSIAN SISTEM APLIKASI DI FASA ANALISIS

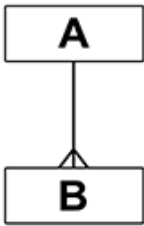
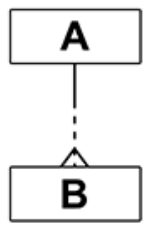
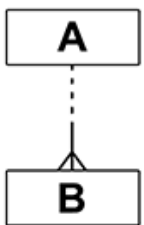
Pengiraan saiz sistem aplikasi perlu dilakukan semula di dalam Fasa Analisis untuk mendapatkan nilai pengiraan yang lebih tepat dan jitu berbanding dengan pengiraan sebelumnya. Berbeza dengan pengiraan di dalam Fasa Permulaan Projek, pengiraan saiz sistem aplikasi di Fasa Analisis akan merujuk kepada model-model yang dibangunkan di dalam fasa ini, iaitu Rajah Aliran Data (DFD) - **Pemodelan Proses Sistem [F2.3]**, Rajah Hubungan Entiti (ERD) dan Definisi Aliran Data - **Pemodelan Keperluan Data [F2.2]**. Langkah-langkah pengiraan saiz sistem aplikasi di Fasa Analisis adalah seperti berikut:

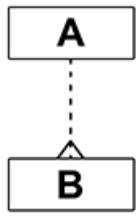
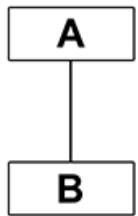
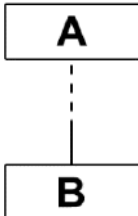
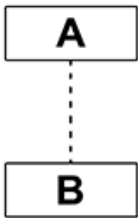
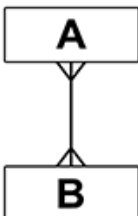
a) Tentukan Kompleksiti Berdasarkan Fungsi Data

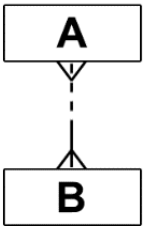
i) **Rujuk dan patuhi peraturan penentuan komponen fungsi**

Penentuan dan pengiraan komponen fungsi ILF, EIF, RET dan DET bagi Fungsi Data perlulah mematuhi kepada peraturan-peraturan hubungan di antara entiti-entiti seperti yang disenaraikan di dalam jadual di bawah:

Jadual 110 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data

Rajah ERD	Jenis Hubungan	Peraturan	Bil. Komponen Fungsi
	1:N	Sekiranya B adalah <i>dependent</i> kepada A	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
		Sekiranya B adalah <i>independent</i> dari A	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	(1):N	Sekiranya A adalah <i>dependent</i> kepada B	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
		Sekiranya A adalah <i>independent</i> dari B	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	1:(N)	Sekiranya B adalah <i>dependent</i> kepada A	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
		Sekiranya B adalah <i>independent</i> dari A	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan

Rajah ERD	Jenis Hubungan	Peraturan	Bil. Komponen Fungsi
	(1):(N)	A dan B adalah <i>independent</i> di antara satu sama lain.	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	1:1	A dan B adalah <i>dependent</i> di antara satu sama lain.	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
	1:(1)	Sekiranya B adalah <i>dependent</i> kepada A	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
		Sekiranya B adalah <i>independent</i> dari A	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	(1):(1)	A dan B adalah <i>independent</i> di antara satu sama lain.	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	N:M	Sekiranya B adalah <i>dependent</i> kepada A	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B
		Sekiranya B adalah <i>independent</i> dari A	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	N:(M)	Sekiranya B adalah <i>dependent</i> kepada A	1 ILF/EIF, 2 RET dan jumlahkan DET bagi A dan B

Rajah ERD	Jenis Hubungan	Peraturan	Bil. Komponen Fungsi
		Sekiranya B adalah <i>independent</i> dari A	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan
	(N):(M)	A dan B adalah <i>independent</i> di antara satu sama lain.	2 ILF/EIF, setiapnya mempunyai RET dan DET yang berasingan

ii) Tentukan bilangan ILF / EIF

Penentuan bilangan ILF/EIF adalah berdasarkan kepada bilangan entiti yang telah disediakan di dalam ERD. Penentuan bilangan ILF dan EIF juga adalah bergantung kepada hubungan modaliti dan kardinaliti di antara satu entiti dengan entiti yang lain di dalam ERD yang telah dibangunkan. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan ILF/EIF di dalam Fasa Analisis:

- Rujuk kepada DFD untuk menentukan sama ada satu-satu entiti itu terdiri daripada ILF sekiranya ia merupakan entiti yang diselenggara di dalam sistem aplikasi yang akan dibangunkan, ataupun EIF sekiranya entiti tersebut ditarik dari sistem luar.
- Kenalpasti hubungan di antara entiti-entiti yang terlibat sama ada ia bersifat *dependent* atau *independent* dengan merujuk kepada ERD yang telah disediakan.
- Tentu dan kirakan bilangan ILF/EIF berpandukan kepada peraturan-peraturan yang disenaraikan di dalam Jadual 104 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data.
- Penentuan dan pengiraan bilangan ILF/EIF bagi entiti yang sama tidak boleh dilakukan secara berulang kali.

iii) Tentukan bilangan RET

Penentuan bilangan RET adalah berdasarkan kepada bilangan entiti yang berhubung kait di dalam satu-satu ILF/EIF. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan RET di dalam Fasa Analisis:

- Berikutan dari penentuan bilangan ILF/EIF dan hubungan di antara entiti pada langkah yang sebelum, kirakan bilangan entiti yang berhubung kait dengan satu-satu entiti yang lain dengan merujuk kepada ERD yang telah disediakan.
- Berdasarkan kepada Jadual 104 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data, kirakan bilangan RET secara bersekali bagi entiti-entiti yang mempunyai hubungan yang bersifat *dependent* di antara satu sama lain. Sekiranya entiti-entiti tersebut mempunyai hubungan bersifat *independent*, kirakan bilangan RET secara berasingan bagi setiap ILF/EIF yang telah dikenalpasti.
- Penentuan dan pengiraan bilangan RET bagi entiti yang sama tidak boleh dilakukan berulang kali walaupun entiti tersebut berhubung kait di bawah ILF/EIF yang berbeza.

iv) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Data adalah berdasarkan kepada bilangan atribut di bawah satu-satu ILF/EIF dengan merujuk kepada entiti yang telah dikenalpasti di dalam ERD. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan DET di dalam Fasa Analisa:

- Senaraikan dan kira bilangan atribut yang terkandung di bawah satu-satu entiti. Setiap data yang telah dikenalpasti mempunyai nilai DET sebanyak 1 sahaja.
- Sama seperti di dalam penentuan di dalam Fasa Permlulaan Projek, atribut yang disetkan sebagai kunci primer (*primary key*) bagi setiap entiti tidak akan ditentukan sebagai DET sekiranya atribut tersebut hanya merupakan artifak teknikal di mana ia tidak memberi makna kepada bisnes dan bukan terdiri dari maklumat yang boleh difahami oleh pengguna (*non user-recognizable*). Sekiranya atribut kunci primer tersebut digunakan sebagai kunci sekunder di dalam entiti yang lain, atribut kunci sekunder tersebut akan dianggap sebagai DET.
- Pengiraan bilangan DET juga bergantung kepada hubungan di antara entiti-entiti. Merujuk kepada Jadual 104 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data, bilangan DET akan digabungkan bersekali bagi entiti-entiti yang berbeza sekiranya hubungan di antaranya adalah *dependent* di antara satu sama lain. Manakala pula, bilangan DET perlu dikira secara berasingan sekiranya hubungan di antara entiti-entiti yang berbeza adalah bersifat *independent*.
- Penentuan dan pengiraan bilangan DET bagi atribut yang sama tidak boleh dilakukan berulang kali melainkan atribut yang sama ditempatkan juga di dalam entiti yang berbeza.

v) Tentukan kompleksiti setiap ILF / EIF

Berdasarkan maklumat EIF/ILF, FTR dan DET, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Data seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap entiti yang telah dikenalpasti.

vi) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama entiti, atribut yang terlibat, bilangan ILF/EIF, FTR, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 111 : Penentuan Tahap Kompleksiti Fungsi Data

Entiti	Atribut	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

b) Tentukan Kompleksiti Berdasarkan Fungsi Transaksi**i) Kenalpasti penguraian DFD yang terendah**

Berpandukan kepada rajah-rajah DFD yang telah dibangunkan di dalam Pemodelan Proses Sistem [F.2.3], kenalpasti setiap penguraian DFD yang berada pada aras yang terendah. Penentuan kompleksiti berdasarkan Fungsi Transaksi akan dilakukan kepada setiap penguraian DFD yang terendah sahaja tanpa perlu mengambil kira rajah DFD pada aras-aras yang lain.

ii) Kenalpasti bilangan EI, EO dan EQ

Penentuan bilangan komponen fungsi EI, EO dan EQ berdasarkan kepada aliran data yang menghubungkan di antara entiti dengan fungsi bisnes dan storan data di dalam DFD. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan EI, EO dan EQ di dalam Fasa Analisis:

- Bagi setiap aliran data yang telah dikenalpasti di dalam DFD serta penerangannya di dalam Definisi Aliran Data, tentukan sama ada

aktiviti-aktiviti tersebut adalah terdiri daripada komponen fungsi EI, EO atau/dan EQ. Penentuan komponen fungsi ini boleh dikenalpasti berdasarkan kepada arah pergerakan aliran-aliran data sama ada yang bermula dari entiti, fungsi bisnes dan berakhir storan data ataupun bermula dan tamat pada arah sebaliknya.

- Setiap aliran data di dalam Rajah Aliran Proses tidak semestinya boleh mengandungi lebih dari satu jenis komponen fungsi EI, EO atau EQ. Bagi arah aliran data yang bermula dari entiti, aliran data tersebut hanya terdiri dari komponen fungsi EI sahaja. Manakala bagi arah aliran data yang bermula dari storan data pula, komponen fungsinya boleh terdiri sama ada dari EO, EQ atau kedua-duanya sekali.
- Jenis komponen fungsi yang sama tidak boleh bertindih dan dikira berulang kali di bawah satu-satu aliran data yang sama.

iii) Tentukan bilangan FTR

Penentuan bilangan FTR adalah berpandukan kepada bilangan dan hubungan di antara entiti yang terlibat di dalam aliran data bagi setiap komponen fungsi EI, EO dan EQ yang telah dikenalpasti. Rujuk kepada senarai ILF/EIF yang telah ditentukan di dalam Fungsi Data untuk menentukan bilangan entiti yang terlibat serta hubungannya di antara satu sama lain. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Analisis:

- Berdasarkan kepada senarai ILF/EIF di dalam Fungsi Data, kenalpasti bilangan dan hubungan di antara entiti pangkalan data yang berinteraksi dengan fungsi bisnes di bawah satu-satu komponen fungsi EI, EO atau EQ.
- Jumlah bilangan komponen fungsi ILF/EIF yang terlibat dengan satu-satu fungsi bisnes adalah merupakan bilangan FTR yang akan diperolehi.

iv) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Transaksi adalah bergantung kepada bilangan atribut yang terlibat di dalam setiap komponen fungsi EI, EO atau EQ bagi satu-satu aliran data. Dalam masa yang sama juga, rujuk kepada Definisi Aliran Data untuk mengetahui atribut yang terlibat pada aliran data berkenaan. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Analisis:

- Bagi komponen fungsi EI, kirakan bilangan DET yang hanya melibatkan data yang dimasukkan, dikemaskini atau/dan dihapuskan.

- Bagi komponen fungsi EO, kirakan bilangan DET yang hanya melibatkan data data yang dipaparkan melalui pengiraan atau logik pengaturcaraan.
- Bagi komponen fungsi EQ, kirakan bilangan DET yang hanya melibatkan data-data yang dipaparkan secara terus tanpa melalui pengiraan atau logik pengaturcaraan.
- Berbeza dengan pengiraan bilangan DET di dalam Fungsi Data, penentuan dan pengiraan bilangan DET bagi Fungsi Transaksi boleh dilakukan berulang kali bagi atribut yang sama di dalam aliran data yang berbeza.

v) Tentukan kompleksiti setiap EI, EO dan EQ

Berdasarkan kepada bilangan EI, EO, EQ, FTR dan DET yang telah ditentukan, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Transaksi seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap komponen fungsi yang telah dikenalpasti.

vi) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama aktiviti, kumpulan maklumat dan data yang terlibat, bilangan EI, EO, EQ, RET, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 112 : Penentuan Tahap Kompleksiti Fungsi Transaksi

Nama Aktiviti	Maklumat	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

c) Kirakan Value Adjustment Function (VAF)

Semak dan kemaskini kadar pengaruh bagi empat belas (14) GSC yang telah diperolehi di dalam pengiraan saiz pada fasa sebelumnya. Pengemaskinian kadar pengaruh dilakukan berdasarkan kepada maklumat dan informasi projek terkini

yang telah diperolehi. Kaedah-kaedah pengiran VAF adalah sama seperti yang telah dijelaskan di dalam Fasa Analisis, iaitu:

- i) Tentukan kadar pengaruh bagi empat belas (14) Ciri-Ciri Am Sistem (GSC) seperti yang disenarai dan diperjelaskan di dalam 8.2.1.3 Empat Belas Ciri-ciri Am Sistem (GSC). Tambahkan skor-skor yang telah diberikan kepada ciri-ciri am berkenaan untuk mendapatkan nilai ΣCi .
- ii) Dengan berpandukan kepada formula pengiraan seperti yang diterangkan di dalam 8.2.2 Pengiraan *Value Adjustment Factor*, masukkan nilai ΣCi yang telah diperolehi ke dalam formula di bawah untuk mendapatkan nilai VAF.

$$VAF = 0.65 + [(\Sigma Ci) / 100]$$

d) Kirakan *Unadjusted Function Points* (UFP)

- i) Berdasarkan maklumat kompleksiti yang telah diperolehi melalui penentuan berdasarkan Fungsi Transaksi dan Fungsi Data, rujuk kepada Jadual Penterjemahan Saiz seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk mendapatkan nilai *Function Points* (FP) bagi setiap komponen fungsi yang telah dikenal pasti.
- ii) Gunakan jadual serta formula pengiraan seperti di dalam 8.2.3 Pengiraan *Unadjusted Function Points* untuk mendapatkan nilai UFP.

e) Kirakan *Adjusted Function Points* (AFP)

Berpandukan kepada nilai **VAF** dan **UFP**, gunakan formula pengiraan seperti di **8.2.4 Pengiraan *Adjusted Function Points* (AFP)** untuk mendapatkan nilai AFP.

f) Kirakan Anggaran *Effort*, Masa dan Kos Pembangunan

Berikutan daripada nilai AFP yang telah diperolehi, nilai tersebut akan digunakan untuk melakukan pengiraan anggaran *effort*, masa dan kos pembangunan keseluruhan sistem aplikasi ataupun fungsi-fungsi di dalamnya. Rujuk 8.2.5 Pengiraan Anggaran *Effort* dan Kos Pembangunan Sistem untuk mendapatkan keterangan lanjut berkenaan dengan formula-formula pengiraan *effort*, masa dan kos.

8.5 LANGKAH-LANGKAH PENGIRAAN SAIZ FUNGSIAN SISTEM APLIKASI DI FASA PELAKSANAAN

Berbeza dengan pengiraan di dalam fasa-fasa sebelumnya, pengiraan saiz sistem aplikasi di Fasa Pelaksanaan akan merujuk kepada model-model yang dibangunkan di dalam fasa ini, iaitu Model Maklumat Logikal – **Rekabentuk Pangkalan Data [F3.4]**, Spesifikasi Integrasi Data – **Rekabentuk Integrasi Sistem [F3.10]**, skrin-skrin antaramuka pengguna dan Jadual Pemetaan Data - **Pemodelan Antaramuka Pengguna [F3.5]**. Langkah-langkah pengiraan saiz sistem aplikasi di Fasa Pelaksanaan adalah seperti berikut:

a) Tentukan Kompleksiti Berdasarkan Fungsi Data

i) Rujuk dan patuhi peraturan penentuan komponen fungsi

Penentuan dan pengiraan komponen fungsi ILF, EIF, RET dan DET bagi Fungsi Data perlulah mematuhi kepada peraturan-peraturan hubungan di antara entiti-entiti atau adual-jadual pangkalan data seperti yang disenaraikan di dalam Jadual Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data.

ii) Tentukan bilangan ILF/EIF

Penentuan bilangan ILF/EIF adalah berdasarkan kepada bilangan jadual yang telah disediakan di dalam Model Maklumat Logikal dan juga senarai data di dalam Spesifikasi Integrasi Data. Penentuan bilangan ILF dan EIF adalah bergantung kepada hubungan modaliti dan kardinaliti di antara satu entiti dengan entiti yang lain. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan ILF/EIF di dalam Fasa Pelaksanaan:

- Rujuk kepada Model Maklumat Logikal untuk menentukan entiti-entiti yang terdiri dari komponen fungsi ILF, dan Spesifikasi Integrasi Data untuk menentukan komponen fungsi EIF.
- Kenalpasti hubungan di antara jadual-jadual yang terlibat sama ada ia bersifat *dependent* atau *independent* dengan merujuk kepada Model Maklumat Logikal dan Spesifikasi Integrasi Data yang telah disediakan.
- Tentu dan kirakan bilangan ILF/EIF berpandukan kepada peraturan-peraturan yang disenaraikan di dalam Jadual Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data.
- Penentuan dan pengiraan bilangan ILF/EIF bagi jadual yang sama tidak boleh dilakukan secara berulang kali.

iii) Tentukan bilangan RET

Penentuan bilangan RET adalah berdasarkan kepada bilangan jadual yang berhubung kait di dalam satu-satu ILF/EIF. Berikut adalah kaedah-kaedah bagi menentu dan mengirakan bilangan RET di dalam Fasa Pelaksanaan:

- Berikutan dari penentuan bilangan ILF/EIF dan hubungan di antara jadual pada langkah yang sebelum, kirakan bilangan jadual yang berhubung kait dengan satu-satu jadual yang lain dengan merujuk kepada Model Maklumat Logikal dan Spesifikasi Integrasi Data yang telah disediakan.
- Berdasarkan kepada Jadual Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data, kirakan bilangan RET secara bersekali bagi jadual-jadual yang mempunyai hubungan yang bersifat *dependent* di antara satu sama lain. Sekiranya jadual-jadual tersebut mempunyai hubungan bersifat *independent*, kirakan bilangan RET secara berasingan bagi setiap ILF/EIF yang telah dikenalpasti.
- Penentuan dan pengiraan bilangan RET bagi jadual yang sama tidak boleh dilakukan berulang kali walaupun jadual tersebut berhubung kait di bawah ILF/EIF yang berbeza.

iv) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Data adalah berdasarkan kepada bilangan atribut di bawah satu-satu ILF/EIF dengan merujuk kepada jadual pangkalan data yang telah dikenalpasti di dalam Model Maklumat Logikal dan Spesifikasi Integrasi Data. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan DET di dalam Fasa Pelaksanaan:

- Senaraikan dan kira bilangan atribut yang terkandung di bawah satu-satu jadual. Setiap data yang telah dikenalpasti mempunyai nilai DET sebanyak 1 sahaja.
- Sama seperti di dalam penentuan di dalam fasa-fasa sebelum, atribut yang disetkan sebagai kunci primer (*primary key*) bagi setiap jadual tidak akan ditentukan sebagai DET sekiranya atribut tersebut hanya merupakan artifak teknikal di mana ia tidak memberi makna kepada bisnes dan bukan terdiri dari maklumat yang boleh difahami oleh pengguna (*non user-recognizable*). Sekiranya atribut kunci primer tersebut digunakan sebagai kunci sekunder di dalam entiti yang lain, atribut kunci sekunder tersebut akan dianggap sebagai DET.
- Pengiraan bilangan DET juga bergantung kepada hubungan di antara jadual-jadual. Merujuk kepada Jadual 104 : Peraturan Penentuan Komponen Fungsi Bagi Fungsi Data, bilangan DET akan digabungkan bersekali bagi jadual-jadual yang berbeza sekiranya hubungan di

antaranya adalah *dependent* di antara satu sama lain. Manakala pula, bilangan DET perlu dikira secara berasingan sekiranya hubungan di antara jadual-jadual yang berbeza adalah bersifat *independent*.

- Penentuan dan pengiraan bilangan DET bagi atribut yang sama tidak boleh dilakukan berulang kali melainkan atribut yang sama ditempatkan juga di dalam jadual yang berbeza.

v) Tentukan kompleksiti setiap ILF / EIF

Berdasarkan maklumat EIF/ILF, FTR dan DET, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Data seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap entiti yang telah dikenalpasti.

vii) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama jadual, atribut yang terlibat, bilangan ILF/EIF, FTR, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 113 : Penentuan Tahap Kompleksiti Fungsi Data

Jadual	Atribut	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

b) Tentukan Kompleksiti Berdasarkan Fungsi Transaksi

i) Kenalpasti bilangan EI, EO dan EQ

Penentuan bilangan komponen fungsi EI, EO dan EQ berdasarkan kepada skrin antaramuka pengguna dan Jadual Pemetaan Data yang menghubungkan di antara pengguna dengan sistem aplikasi yang dibangunkan. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan EI, EO dan EQ di dalam Fasa Pelaksanaan:

- Bagi setiap skrin antaramuka pengguna yang telah dibangunkan, tentukan sama ada skrin-skrin tersebut mengandungi komponen-komponen fungsi EI, EO atau/dan EQ.

- Rujuk juga kepada ruangan CRUD di dalam Jadual Pemetaan Data bagi membantu dalam menentukan komponen-komponen fungsi yang terlibat.
- Setiap skrin antaramuka pengguna boleh mengandungi lebih dari satu jenis komponen fungsi sama ada ia merupakan EI, EO atau/dan EQ.
- Jenis komponen fungsi yang sama tidak boleh bertindih dan dikira berulang kali di bawah satu-satu skrin antaramuka.
- Skrin-skrin antaramuka pengguna tambahan, seperti skrin *popup*, kotak mesej (*message box*) dan notifikasi, tidak perlu diambil kira sebagai komponen fungsi EI, EO atau/dan EQ, di mana skrin-skrin tambahan tersebut akan hanya diletakkan di bawah komponen fungsi DET sahaja.

ii) Tentukan bilangan FTR

Penentuan bilangan FTR adalah berpandukan kepada bilangan dan hubungan di antara jadual-jadual pangkalan data yang terlibat di bawah setiap komponen fungsi EI, EO dan EQ seperti yang telah dikenalpasti pada langkah yang sebelumnya. Rujuk kepada senarai ILF/EIF yang telah ditentukan di dalam Fungsi Data serta Jadual Pemetaan Data untuk menentukan bilangan jadual pangkalan data yang terlibat serta hubungannya di antara satu sama lain. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Pelaksanaan:

- Berdasarkan kepada senarai ILF/EIF serta Jadual Pemetaan Data, kenalpasti bilangan dan hubungan di antara jadual pangkalan data yang berinteraksi dengan fungsi bisnes di bawah satu-satu komponen fungsi EI, EO atau EQ.
- Jumlah bilangan komponen fungsi ILF/EIF yang terlibat dengan skrin antaramuka pengguna berkenaan adalah merupakan bilangan FTR yang akan diperolehi.

iii) Tentukan bilangan DET

Penentuan bilangan DET bagi Fungsi Transaksi adalah bergantung kepada bilangan atribut yang terlibat di dalam setiap komponen fungsi EI, EO atau EQ bagi satu-satu skrin antaramuka pengguna. Dalam masa yang sama juga, rujuk kepada Jadual Pemetaan Data untuk mengenalpasti atribut yang diperlukan pada skrin terlibat. Berikut adalah kaedah-kaedah untuk menentu dan mengirakan bilangan FTR di dalam Fasa Pelaksanaan:

- Bagi komponen fungsi EI, kirakan bilangan DET yang hanya melibatkan data yang dimasukkan, dikemaskini atau/dan dihapuskan.

- Bagi komponen fungsi EO, kirakan bilangan DET yang hanya melibatkan data data yang dipaparkan melalui pengiraan atau logik pengaturcaraan.
- Bagi komponen fungsi EQ, kirakan bilangan DET yang hanya melibatkan data-data yang dipaparkan secara terus tanpa melalui pengiraan atau logik pengaturcaraan.
- Berbeza dengan pengiraan bilangan DET di dalam Fungsi Data, penentuan dan pengiraan bilangan DET bagi Fungsi Transaksi boleh dilakukan berulang kali bagi atribut yang sama kecuali atribut yang berada di bawah komponen fungsi dan skrin antarmuka yang sama.

iv) Tentukan kompleksiti setiap EI, EO dan EQ

Berdasarkan kepada bilangan EI, EO, EQ, FTR dan DET yang telah ditentukan, rujuk kepada Jadual Matriks Kompleksiti bagi Fungsi Transaksi seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk menentukan tahap kompleksiti setiap komponen fungsi yang telah dikenalpasti.

v) Lengkapkan Jadual Penentuan Tahap Kompleksiti Fungsi Data

Gunakan jadual di bawah untuk merekodkan nama aktiviti, kumpulan maklumat dan data yang terlibat, bilangan EI, EO, EQ, RET, DET serta kompleksitinya yang telah dikenalpasti di dalam langkah-langkah yang sebelum.

Jadual 114 : Penentuan Tahap Kompleksiti Fungsi Transaksi

Nama Aktiviti	Maklumat	Komponen Fungsi	Bil. RET	Bil. DET	Tahap Kompleksiti

c) Kirakan Value Adjustment Function (VAF)

Semak dan kemaskini kadar pengaruh bagi empat belas (14) GSC yang telah diperolehi di dalam pengiraan pada fasa-fasa sebelumnya. Pengemaskinian kadar pengaruh dilakukan berdasarkan kepada maklumat dan informasi projek

terkini yang telah diperolehi. Kaedah-kaedah pengiraan VAF adalah sama seperti yang telah dijelaskan di dalam fasa-fasa sebelumnya, iaitu:

- i) Tentukan kadar pengaruh bagi empat belas (14) Ciri-Ciri Am Sistem (GSC) seperti yang disenarai dan diperjelaskan di dalam 8.2.1.3 Empat Belas Ciri-ciri Am Sistem (GSC). Tambahkan skor-skor yang telah diberikan kepada ciri-ciri am berkenaan untuk mendapatkan nilai ΣCi .
- ii) Dengan berpandukan kepada formula pengiraan seperti yang diterangkan di dalam 8.2.2 Pengiraan *Value Adjustment Factor*, masukkan nilai ΣCi yang telah diperolehi ke dalam formula di bawah untuk mendapatkan nilai VAF.

$$VAF = 0.65 + [(\Sigma Ci) / 100]$$

d) Kirakan *Unadjusted Function Points* (UFP)

- i. Berdasarkan maklumat kompleksiti yang telah diperolehi melalui penentuan berdasarkan Fungsi Transaksi dan Fungsi Data, rujuk kepada Jadual Penterjemahan Saiz seperti di 8.2.1.2 Jadual Matriks Kompleksiti dan Jadual Penterjemahan untuk mendapatkan nilai *Function Points* (FP) bagi setiap komponen fungsi yang telah dikenal pasti.
- ii. Gunakan jadual serta formula pengiraan seperti di dalam 8.2.3 Pengiraan *Unadjusted Function Points* untuk mendapatkan nilai UFP.

e) Kirakan *Adjusted Function Points* (AFP)

Berpandukan kepada nilai VAF dan UFP, gunakan formula pengiraan seperti di 8.2.4 Pengiraan *Adjusted Function Points* (AFP) untuk mendapatkan nilai AFP.

f) Kirakan Anggaran *Effort*, Masa dan Kos Pembangunan

Berikutan daripada nilai AFP yang telah diperolehi, nilai tersebut akan digunakan untuk melakukan pengiraan anggaran *effort*, masa dan kos pembangunan keseluruhan sistem aplikasi ataupun fungsi-fungsi di dalamnya. Rujuk 8.2.5 Pengiraan Anggaran *Effort* dan Kos Pembangunan Sistem untuk mendapatkan keterangan lanjut berkenaan dengan formula-formula pengiraan *effort*, masa dan kos.

APENDIKS



APENDIKS

Apendiks 1 : Borang Permohonan Pembangunan Sistem

BORANG PERMOHONAN PEMBANGUNAN SISTEM			
Nama Sistem			
Keterangan			
Pemilik Sistem (Bahagian)		Pegawai Bertanggungjawab	
Jenis Permohonan	<input checked="" type="checkbox"/> Pembangunan Baharu <input type="checkbox"/> Peningkatan Sistem <input type="checkbox"/> Penambahbaikan Sistem	Kesegeraan	<input type="checkbox"/> Segera – Keperluan operasi <input type="checkbox"/> Masalah Sistem
Keperluan Bisnes			
Faedah Sistem			
Isu / Kekangan			
Pemohon		Tarikh Permohonan	
Keputusan Permohonan	<input checked="" type="checkbox"/> Tolak <input type="checkbox"/> Pembangunan Dalam <input type="checkbox"/> Pembangunan Outsource		
Pegawai Penilai		Tarikh Keputusan	

Apendiks 2 : Templat Definisi Aktiviti Fungsi Bisnes

DEFINISI AKTIVITI FUNGSI BISNES			
Rujukan Fungsi			
Nama Fungsi			
Rujukan Aktiviti			
Nama Aktiviti			
Keterangan Aktiviti			
Aktor			
Tanggungjawab			
Kekerapan		Unit Kekerapan (Jam/Hari/Bulan)	
Aktiviti Sebelum		Aktiviti Selepas/ Aktiviti Lain	
Kaedah/Operasi (Bagaimana):			
Penggunaan Maklumat:			
Polisi dan Dasar Berkaitan			
Kaedah Alternatif			
Ciri-ciri Kualiti (Keperluan Bukan Fungsian)			
Catatan Tambahan:			

Apendiks 3 : Templat Definisi Kamus Data (Entity & Attribute)

Nama Entiti						
Keterangan Entiti						
Atribut						
Nama	Pilihan (Y/T)	Format	Panjang	Tempat Perpuluhan	Default Value	Keterangan
# – Pengenal Unik (UID) Primer U – Pengenal Unik (UID) Sekunder						
Bil.	Peraturan Bisnes (syarat hubungan dengan entiti yang lain) Masukkan hubungan entiti ini dengan entiti-entiti lain. Jelaskan secara ringkas peraturan bisnes yang berkaitan dengan hubungan tersebut.					
Setiap <Nama Entiti>:-						
Modaliti dan Nama Hubungan		Kardinaliti			Entiti	
Pengenal Unik (UID)						
Bil.	Primer? (Y/T)	Nama Atribut	Hubungan	Ke Nama Entiti		
Kapasiti Maklumat (jumlah rekod yang disimpan)						

Apendiks 4 : Templat Skema Logikal Pangkalan Data

Nama Jadual : PENGGUNA

Nama Medan	Kekunci Primer (P) / Kekunci Unik (U) / Kekunci Asing (F)	Pilihan Null (Y/T)	Format	Panjang	Tempat Perpuluhan	Default Value	Keterangan Medan	Catatan

Apendiks 5 : Templat Definisi Aliran Data

Bil.	Nama Aliran Data	Sumber			ID /Nama	Destinasi			ID/ Nama	Atribut	CRUD
		Entiti Luaran	Fungsi Bisnes	Storan Data		Entiti Luaran	Fungsi Bisnes	Storan Data			
1.											
2.											
3.											

Apendiks 6 : Templat Pemetaan Data (Antaramuka)

Nama Label	Jenis Objek	Nama Jadual	Nama Medan Data	CRUD	Catatan

MAMPU
BERSAMA-SAMA MELAKSANAKAN TRANSFORMASI
VERSI BETA

Apendiks 7 : Templat Senario Use Case

Rujukan Use Case			
Nama Use Case			
Keterangan			
Pra Syarat			
Aktor			
Input	Langkah	Rujukan Rekabentuk Antaramuka	Keperluan Keterangan / Syarat / Kekangan
Pasca Syarat			
Proses Alternatif			

Apendiks 8 : Templat Rekabentuk Migrasi Data

a) Pemetaan Data Migrasi

Bil.	MAKLUMAT PANGKALAN DATA DESTINASI					MAKLUMAT PANGKALAN DATA SUMBER				DDSA	Catatan
	Penerima (NAMA SISTEM)					Penyedia (NAMA SISTEM)					
	Nama Sistem	Table	Medan	Keterangan	Jenis	Nama Sistem	Table	Medan	Jenis		
1.											
2.											
3.											
4.											
5.											

b) Peraturan Pemetaan Data

Kod Data Asal (sumber asal)	Keterangan Data	Peraturan /Nota	Kod Data Baharu (destinasi baru)

Apendiks 9 : Templat Rekabentuk Integrasi**a) Maklumat Servis Integrasi**

Bil.	Rujukan Fungsi	Rujukan Aktiviti	Nama Sistem Sumber	Pemilik Maklumat	Keterangan Maklumat yang dihantar	Tujuan Penggunaan Maklumat

MAMPU
BERSAMA-SAMA MELAKSANAKAN TRANSFORMASI
VERSI BETA

b) Format Pertukaran Data

Nama Servis																										
Keterangan																										
Kaedah Integrasi																										
URL Web Servis																										
Request																										
Respond																										
Data yang terlibat	<table border="1"><thead><tr><th>Nama</th><th>Jenis</th><th>Saiz</th><th>Nullable</th><th>Rules</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></tbody></table>	Nama	Jenis	Saiz	Nullable	Rules																				
Nama	Jenis	Saiz	Nullable	Rules																						

c) Pemetaan Struktur Data

Nama Servis							
<nama servis>							
Nama Sistem Penerima				Nama Sistem Pemilik			
<nama sistem penerima>				<nama sistem pemilik>			
Nama Medan	Jenis	Saiz	Keterangan	Nama Medan	Jenis	Saiz	Keterangan

Apendiks 10 : Templat Persediaan Ujian**a) Templat Senario Ujian**

ID Use Case	
Keterangan Use Case	
ID Senario Ujian	Keterangan Senario Ujian
Senario Utama	
Senario Alternatif	

b) Templat Kes Ujian

ID Kes Ujian			
Nama Kes Ujian			
Keterangan Kes Ujian			
Rujukan ID Keperluan			
Pra-Syarat			
Input / Langkah-langkah Ujian	Jangkaan Hasil	Hasil Sebenar	Status (Lulus/ Gagal)

c) Templat Prosedur / Skrip Ujian

ID Prosedur/ Skrip Ujian				
ID Senario Ujian				
Keterangan Senario Ujian				
Objektif				
Langkah Permulaan :				
Kaitan dengan prosedur lain				
LOG UJIAN				
Tarikh	Tandatangan	Item Ujian	Status Ujian (Lulus/ Gagal)	
Komen :				
PROSEDUR UJIAN				
ID Kes Ujian	Nama Kes Ujian	Jangkaan Hasil	Hasil Sebenar/ Catatan	Keputusan Ujian

d) Templat *Traceability Matrix* Ujian

ID Senario Ujian	ID Use Case	ID Kes Ujian	Keterangan Kes Ujian

Apendiks 11 : Contoh *Final Acceptance Certificate***(FINAL ACCEPTANCE CERTIFICATE – FAC)**

Dengan ini disahkan **Sistem Tempahan Bilik Mesyuarat (eTempah)** telah diuji melalui sesi **Ujian Penerimaan Akhir (FAT)** dan didapati menepati spesifikasi yang telah ditetapkan oleh pihak **Kerajaan**.

Dengan ini disahkan aplikasi eTempah **lulus Ujian Penerimaan Akhir (FAT)**, pada _____.

Untuk pihak Kerajaan:-

1.	_____ Nama dan Jawatan (Pemilik Sistem)	_____ Tandatangan
2.	_____ Nama dan Jawatan (Pembekal Sistem)	_____ Tandatangan

DOKUMEN



RUJUKAN :



D01 DOKUMEN PELAN PEMBANGUNAN SISTEM (PPS)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan. Contoh keterangan dokumen adalah seperti di bawah :

“Dokumen ini menyatakan pelan bagi pengurusan dan pembangunan aplikasi. Ia bertujuan untuk menerangkan secara terperinci perancangan-perancangan yang telah dibangunkan merangkumi serahan projek, pengendalian projek, perancangan proses teknikal seperti pendekatan projek, perkakasan dan perisian yang akan digunakan, dokumen-dokumen yang akan disediakan serta jadual pelaksanaan pembangunan aplikasi.”

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat angka selepas titik perpuluhan sahaja, contohnya, dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. PENGENALAN

1.1 Tujuan Projek

Seksyen ini adalah ruangan untuk menerangkan rasional yang menyebabkan projek pembangunan perlu dilaksanakan seperti keperluan baru, perluasan (roll-out) atau memperbaiki proses/perkhidmatan yang disediakan oleh organisasi. Tujuan projek ini telah dipersetujui oleh pihak pengurusan jabatan/kementerian.

1.2 Skop Projek

Seksyen ini adalah ruangan untuk menjelaskan skop projek yang telah dipersetujui dan akan dijadikan sebagai sempadan (boundary) untuk pelaksanaan pembangunan aplikasi dan baseline untuk mengukur kejayaan projek.

1.3 Serahan Projek

Seksyen ini adalah ruangan untuk menyenaraikan serahan projek mengikut fasa pelaksanaan yang terlibat yang meliputi item nama serahan, tarikh serahan, kuantiti serahan, penyedia, pengesah dan pelulus serahan.

2. PENGENDALIAN PROJEK PEMBANGUNAN APLIKASI

2.1 Model Proses

Seksyen ini adalah ruangan untuk menerangkan turutan fasa bagi keseluruhan tempoh hayat aplikasi (produk) atau lebih dikenali sebagai Kitar Hayat Produk (Product Life Cycle).

2.2 Struktur Organisasi Pasukan

Seksyen ini adalah ruangan untuk menerangkan struktur organisasi pasukan projek dengan menggunakan sama ada carta organisasi, rajah matriks atau notasi-notasi lain yang dapat menggambarkan bidang kuasa, tanggungjawab dan komunikasi di dalam projek.

2.3 Peranan dan Tanggungjawab

Seksyen ini adalah ruangan untuk menyenaraikan pegawai-pegawai atau unit organisasi (unit/seksyen/bahagian) yang bertanggungjawab kepada fungsi-fungsi atau aktiviti-aktiviti utama projek.

Contoh Peranan dan Fungsi

Bil.	Fungsi / Aktiviti Utama	Tanggungjawab
1	Perekayasaan Proses	Bahagian Pengurusan Perubahan, Kementerian Kesihatan Malaysia
2	Penyediaan Dokumen-dokumen Pembangunan Sistem	MAMPU, Jabatan Perdana Menteri
3	Pembangunan Aplikasi	Bahagian Pengurusan Maklumat, Kementerian Kesihatan Malaysia

3. PROSES PENGURUSAN

3.1 Andaian, Kebergantungan dan Kekangan

Seksyen ini adalah ruangan untuk mengenalpasti dan menerangkan andaian, kebergantungan dan kekangan dalam pembangunan perisian. Andaian, kebergantungan dan kekangan dalam pembangunan perisian perlu dikenal pasti dan disenaraikan. Andaian, kebergantungan dan kekangan yang biasa dihadapi bagi setiap projek adalah dari segi peruntukan, perkakasan, sistem sedia ada dan integrasi dengan sistem yang lain yang memberi impak kepada pembangunan sistem baharu.

3.2 Risiko

Seksyen ini adalah ruangan menyenaraikan risiko-risiko projek. Risiko boleh dikategorikan kepada 3 kategori berikut:

- **Risiko Projek** iaitu risiko yang memberi kesan kepada jadual, aktiviti dan sumber projek;
- **Risiko Produk** iaitu risiko yang memberi kesan kualiti perisian yang sedang dibangunkan; dan
- **Risiko Organisasi** iaitu risiko yang memberi kesan kepada organisasi pemilik perisian.

3.3 Tahap Kebarangkalian Risiko dan Tahap Impak

Seksyen ini adalah ruangan untuk mengenalpasti impak risiko kepada projek.

Contoh Borang Penilaian Risiko

Kemungkinan		Tinggi	Sederhana	Rendah
Skala potensi impak	Besar	Peruntukan tidak cukup		
	Sederhana			
	Kecil			

3.4 Pemantauan dan Kawalan

Seksyen ini adalah ruangan untuk mekanisma atau pendekatan untuk memantau dan mengawal kos, jadual pelaksanaan, kualiti sistem dan pelaksanaan sistem sepanjang tempoh projek.

4. PROSES TEKNIKAL

4.1 Pendekatan, Teknik dan Alat Bantu

Seksyen ini adalah ruangan untuk menerangkan keseluruhan gambaran bagi persekitaran pembangunan perisian yang melibatkan persekitaran sistem pengkomputeran.

4.2 Dokumen Aplikasi

Seksyen ini adalah ruangan untuk menyenaraikan dokumen-dokumen pembangunan sistem aplikasi yang perlu disediakan.

4.3 Dokumen Fungsi Sokongan

Seksyen ini adalah ruangan untuk menyenaraikan dokumen-dokumen fungsi sokongan yang berkaitan dengan projek.

5. PAKEJ KERJA, JADUAL DAN PERUNTUKAN

5.1 Pakej Kerja

Seksyen ini adalah ruangan untuk menerangkan aktiviti yang memerlukan kepakaran tertentu bagi menyiapkan sistem yang akan dibangunkan.

Contoh Pakej Kerja Personel

Bil.	Nama Personel	Tempoh Pengalaman Bekerja Dalam Bidang ICT (Bilangan Tahun)	Pengalaman Bekerja Dalam Projek Berkaitan		Bidang Kepakaran	Peranan & Tanggungjawab Dalam Projek
			Ya/Tidak	Bilangan Tahun		
1.						
2.						

5.2 Kebergantungan

Seksyen ini adalah ruangan untuk menerangkan kebergantungan antara satu pakej kerja dengan pakej kerja yang lain atau dengan aktiviti luaran yang lain.

5.3 Sumber

Seksyen ini adalah ruangan untuk menjelaskan perkara yang diperlukan untuk menjalankan aktiviti sepanjang tempoh projek.

5.4 Peruntukan Kos

Seksyen ini adalah ruangan untuk menyenaraikan sumber dan jumlah peruntukan yang diperlukan bagi menjalankan projek.

5.5 Jadual Perancangan

Seksyen ini adalah ruangan untuk menyenaraikan jadual perancangan yang lebih terperinci mengikut aktiviti.

6. KOMPONEN TAMBAHAN

Seksyen ini merupakan ruangan untuk menyenaraikan sebarang komponen tambahan yang berhubung kait dengan projek seperti pelan keselamatan, pelan latihan, pelan perolehan perkakasan dan perisian, pelan-pelan fasiliti, pelan pemasangan, pelan penyelenggaraan dan komponen-komponen lain yang berkaitan.

7. LAMPIRAN

Seksyen ini merupakan ruangan untuk mengepilkan dokumen-dokumen sokongan yang berkaitan dengan pelan pembangunan sistem seperti minit mesyuarat, surat-surat dan sebagainya.

RUJUKAN :



D02 DOKUMEN SPESIFIKASI KEPERLUAN BISNES (BRS)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan ringkas berkenaan dokumen yang disediakan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh jadual semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim**Akronim**

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh jadual bagi senarai akronim adalah seperti berikut :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh jadual bagi senarai definisi adalah seperti berikut :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. PENGENALAN

1.1 Tujuan Bisnes

Seksyen ini adalah ruangan untuk menerangkan latarbelakang, sebab-sebab dan bagaimana sistem yang akan dibangunkan dapat membantu dan menyumbang untuk mencapai objektif bisnes.

1.2 Skop Bisnes

Seksyen ini adalah ruangan untuk menjelaskan penentuan skop bagi domain bisnes organisasi yang terlibat. Sila rujuk kepada langkah 1b di dalam Penyediaan Spesifikasi Keperluan Bisnes [F1.5] untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

1.3 Gambaran Keseluruhan Projek

Seksyen ini adalah ruangan untuk menerangkan struktur organisasi yang berkaitan dengan domain bisnes serta hubungannya dengan entiti luar. Penggunaan rajah adalah digalakkan untuk menerangkan struktur organisasi berkenaan.

1.4 Senarai Pemegang Taruh

Seksyen ini adalah ruangan untuk menyenarai dan menerangkan pemegang-pemegang taruh yang terlibat dengan domain bisnes berkenaan. Sila rujuk kepada langkah 1d di dalam Penyediaan Spesifikasi Keperluan Bisnes [F1.5] untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2. KEPERLUAN PENGURUSAN BISNES

2.1 Matlamat dan Objektif

Seksyen ini adalah ruangan untuk menyenarai dan menerangkan matlamat, objektif dan hasil bisnes yang ingin dicapai melalui pelaksanaan sistem yang akan dibangunkan.

2.2 Arkitektur Bisnes

Seksyen ini adalah ruangan untuk menjelas dan menyediakan rajah Arkitektur Bisnes yang berkaitan dengan sistem yang akan dibangunkan. Sila rujuk kepada langkah 5 di dalam **Penentuan Keperluan Bisnes [F1.2]** dan langkah 2b di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2.3 Arkitektur Maklumat

Seksyen ini adalah ruangan untuk menerangkan Arkitektur Maklumat bagi sistem aplikasi yang akan dibangunkan. Sila rujuk kepada langkah 6 di dalam

Penentuan Keperluan Bisnes [F1.2] dan langkah 2c di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3. KEPERLUAN PENGOPERASIAN BISNES

3.1 Keperluan Fungsi Bisnes

3.1.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi Bisnes. Sila rujuk kepada **Pemodelan Fungsi Bisnes [F1.3]** dan langkah 3a - perkara i di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.1.2 Model Fungsi Bisnes

Seksyen ini adalah ruangan untuk menyediakan Model Fungsi Bisnes yang terdiri daripada Rajah Hirarki Fungsi serta keterangan bagi fungsi-fungsi berkenaan. Sila rujuk kepada langkah-langkah di bawah **Pemodelan Fungsi Bisnes [F1.4]** dan langkah 3a - perkara ii di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.1.3 Senarai Pengguna

Seksyen ini adalah ruangan untuk menyenaraikan senarai pengguna-pengguna yang terlibat secara langsung dengan fungsi bisnes. Sila rujuk kepada langkah 3a - perkara iii di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.2 Keperluan Proses Bisnes

3.2.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Proses. Sila rujuk kepada **Pemodelan Proses Bisnes [F1.4]** dan langkah 3b - perkara i di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** bagi mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.2.2 Model Proses Bisnes

Seksyen ini adalah ruangan untuk menyediakan Model Proses Bisnes yang merangkumi Aliran Proses Bisnes dan Definisi Fungsi Bisnes. Sila rujuk kepada Langkah 1 sehingga langkah 4 di bawah **Pemodelan Proses Bisnes [F1.4]** dan langkah 3b - perkara ii di dalam **Penyediaan Spesifikasi Keperluan Bisnes [F1.5]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini. Sila rujuk

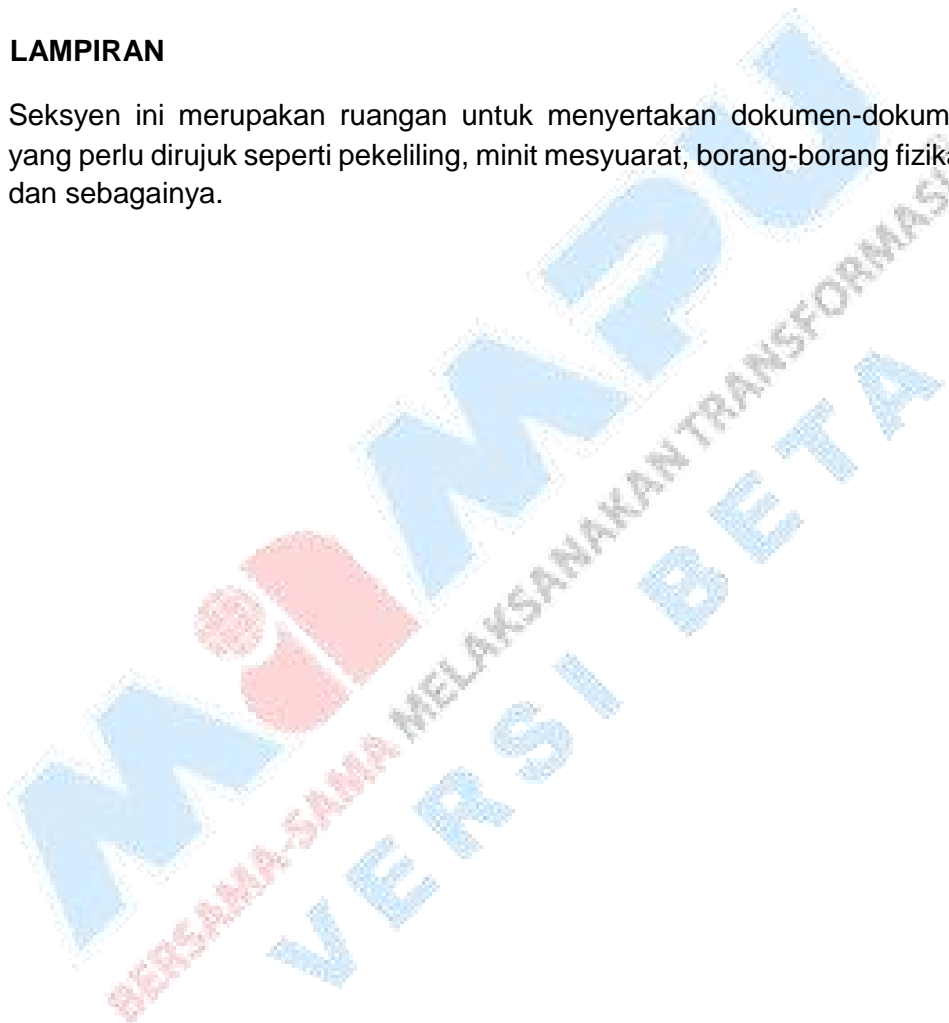
juga kepada contoh format templat bagi penyediaan Definisi Fungsi Bisnes di **Apendiks 2**.

3.3 Pengiraan Saiz Sistem Aplikasi

Seksyen ini adalah ruangan untuk menyediakan Pengiraan Saiz Sistem Aplikasi dengan menggunakan kaedah *Function Points Analysis*. Sila rujuk kepada bab 8 di dalam **Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Permulaan Projek** bagi mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4. LAMPIRAN

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti pekeliling, minit mesyuarat, borang-borang fizikal, surat-surat dan sebagainya.



RUJUKAN :

D03 DOKUMEN SPESIFIKASI KEPERLUAN SISTEM (SRS)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh jadual semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh jadual bagi senarai akronim adalah seperti berikut :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh jadual bagi senarai definisi adalah seperti berikut :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.



1. PENGENALAN

1.1 Tujuan Sistem

Seksyen ini adalah ruangan untuk menerangkan tujuan, objektif dan matlamat sistem aplikasi ini dibangunkan selaras dengan objektif bisnes yang ingin dicapai.

1.2 Skop Sistem

Seksyen ini adalah ruangan untuk menjelaskan penentuan skop sistem aplikasi yang ingin dibangunkan. Sila rujuk kepada langkah 1b di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

1.3 Senarai Aktor Sistem

Seksyen ini adalah ruangan untuk menyenaraikan aktor-aktor sistem yang terlibat serta keterangan fungsinya di dalam sistem aplikasi yang akan dibangunkan. Sila rujuk kepada langkah 1c di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2. PEMODELAN FUNGSI SISTEM

2.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi Sistem. Sila rujuk kepada ruangan notasi di **Pemodelan Fungsi Sistem [F2.2]** dan langkah 2a di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2.2 Rajah Hierarki Fungsian Sistem

Seksyen ini adalah ruangan untuk menyediakan Rajah Hierarki Fungsian Sistem yang merangkumi komponen-komponen seperti sistem, subsistem, fungsi, modul, submodul dan transaksi. Sila rujuk kepada langkah-langkah dalam **Pemodelan Fungsi Sistem [F2.2]** dan langkah 2b di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2.3 Jadual Pepadanan Aktor Dengan Fungsi Sistem

Seksyen ini adalah ruangan untuk menyediakan Jadual Pepadanan. Sila rujuk kepada langkah-langkah dalam **Pemodelan Fungsi Sistem [F2.2]** dan langkah 2c di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3. PEMODELAN *USE CASE*

3.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model *Use Case*. Sila rujuk kepada **Pemodelan *Use Case* [F2.1]** dan langkah 2a - perkara i di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.2 Model *Use Case*

Seksyen ini adalah ruangan untuk menyedia Model *Use Case* yang terdiri daripada Rajah *Use Case* serta keterangan bagi *use case* yang terlibat. Sila rujuk kepada langkah-langkah di bawah **Pemodelan *Use Case* [F2.1]** dan langkah 2a - perkara ii di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4. PEMODELAN MAKLUMAT

4.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Maklumat. Sila rujuk kepada **Pemodelan Keperluan Data [F2.3]** dan langkah 3a - perkara i di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4.2 Model Maklumat

Seksyen ini adalah ruangan untuk menyediakan Model Maklumat yang terdiri daripada Rajah Hubungan Entiti. Sila rujuk kepada langkah 1 sehingga langkah 3 di bawah **Pemodelan Keperluan Data [F2.3]** dan langkah 3a - perkara ii di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4.3 Definisi Kamus Data

Seksyen ini adalah ruangan untuk menyedia dan melengkapkan Definisi Kamus Data bagi setiap entiti dan atribut yang telah disediakan di dalam Rajah Hubungan Entiti. Sila rujuk kepada langkah 4 di bawah **Pemodelan Keperluan Data [F2.3]** dan langkah 3a - perkara iii di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini. Sila rujuk juga kepada contoh format templat di **Apendiks 3 - Templat Definisi Kamus Data**.

5. PEMODELAN PROSES SISTEM

5.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Proses Sistem. Sila rujuk kepada **Pemodelan Proses Sistem [F2.4]** dan langkah 4a - perkara i di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

5.2 Model Proses Sistem

Seksyen ini adalah ruangan untuk menyedia Model Maklumat yang terdiri daripada Rajah Konteks dan Rajah Aliran Data. Sila rujuk kepada langkah 1 sehingga 3 di bawah **Pemodelan Proses Sistem [F2.4]** dan langkah 4a - perkara ii di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

5.3 Definisi Aliran Data

Seksyen ini adalah ruangan untuk menyedia dan melengkapkan Definisi Aliran Data. Sila rujuk kepada langkah 4 di bawah **Pemodelan Proses Sistem [F2.4]** dan langkah 4a – perkara iii di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini. Sila rujuk juga kepada contoh format templat bagi penyediaan Definisi Aliran Data di **Apendiks 5**.

6. PENENTUAN KEPERLUAN BUKAN FUNGSIAN

6.1 Jadual Ciri-ciri Kualiti Sistem

Seksyen ini adalah ruangan untuk menyedia dan melengkapkan jadual keperluan bukan fungsian yang merangkumi tiga aspek utama iaitu aspek sistem, aspek dalaman dan aspek luaran. Sila rujuk kepada langkah-langkah di bawah **Penentuan Keperluan Bukan Fungsian [F2.5]** dan langkah 6 di dalam **Penyediaan Spesifikasi Keperluan Sistem [F2.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

7. PENENTUAN SAIZ SISTEM APLIKASI

Seksyen ini adalah ruangan untuk menyediakan Pengiraan Saiz Sistem Aplikasi dengan menggunakan kaedah *Function Points Analysis*. Sila rujuk kepada bab 8 di dalam **Langkah-langkah Pengiraan Saiz Fungsian Sistem Aplikasi di Fasa Analisis** bagi mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

8. LAMPIRAN

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti format borang fizikal, format laporan dan pelbagai lagi dokumen-dokumen lain yang berkaitan.

RUJUKAN :

D04 DOKUMEN SPESIFIKASI REKABENTUK SISTEM (SDS)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh jadual semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh jadual bagi senarai akronim adalah seperti berikut :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh jadual bagi senarai definisi adalah seperti berikut :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. PENGENALAN

1.1 Tujuan Rekabentuk

Seksyen ini adalah ruangan untuk menerangkan tujuan, objektif dan matlamat yang ingin dicapai di dalam rekabentuk sistem aplikasi selaras dengan objektif bisnes dan keperluan sistem yang ingin dipenuhi.

1.2 Skop Rekabentuk

Seksyen ini adalah ruangan untuk menjelaskan penentuan skop rekabentuk bagi sistem aplikasi yang ingin dibangunkan. Sila rujuk kepada Langkah 1b di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2. REKABENTUK ARKITEKTUR

2.1 Arkitektur Keseluruhan Sistem Aplikasi

Seksyen ini adalah ruangan untuk menyediakan Arkitektur Keseluruhan Sistem Aplikasi yang merangkumi komponen-komponen antaramuka sistem, aplikasi dan pangkalan data. Sila rujuk kepada Langkah 4 di dalam **Rekabentuk Arkitektur [F3.1]** dan Langkah 2a – Perkara i di dalam **4.9 Penyediaan Spesifikasi Rekabentuk Sistem** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

2.2 Arkitektur Aplikasi

Seksyen ini adalah ruangan untuk menyediakan Arkitektur Keseluruhan Sistem Aplikasi yang merangkumi komponen-komponen antaramuka sistem, aplikasi dan pangkalan data. Sila rujuk kepada Langkah 4 di dalam **Rekabentuk Arkitektur [F3.1]** dan Langkah 2a – Perkara i di dalam **4.9 Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3. PEMODELAN FUNGSI SISTEM

3.1 Penggunaan Notasi

Seksyen ini adalah ruangan untuk menyenaraikan notasi-notasi yang akan digunakan untuk menyediakan Model Fungsi Sistem. Sila rujuk kepada ruangan notasi di **Pemodelan Fungsi Sistem [F2.2]** dan langkah 3a di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.2 Rajah Hierarki Fungsian Sistem

Seksyen ini adalah ruangan untuk menyediakan Rajah Hierarki Fungsian Sistem yang merangkumi komponen-komponen Fungsi Sistem pada tahap sistem, subsistem, fungsi, modul, submodul dan transaksi. Sila rujuk kepada langkah-langkah dalam **Pemodelan Fungsi Sistem [F2.2]** dan langkah 3b di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

3.3 Jadual Pepadanan Aktor Dengan Fungsi Sistem

Seksyen ini adalah ruangan untuk menyediakan Jadual Pepadanan. Sila rujuk kepada langkah-langkah dalam **Pemodelan Fungsi Sistem [F2.2]** dan langkah 3c di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4. REKABENTUK FUNGSIAN

4.1 Rekabentuk Antaramuka Pengguna dan Pemetaan Data

Seksyen ini adalah ruangan untuk menyertakan imej-imej antaramuka pengguna dan menyediakan jadual pemetaan data. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Antaramuka Pengguna [F3.5]** dan Langkah 3d di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

4.2 Rekabentuk Transaksi Sistem

Seksyen ini adalah ruangan untuk menyediakan jadual-jadual senario *use case*. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Transaksi Sistem [F3.3]** dan Langkah 3c di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

5. REKABENTUK PANGKALAN DATA

5.1 Rekabentuk Pangkalan Data

Seksyen ini adalah ruangan untuk menyertakan Rajah Hubungan Entiti (ERD) Logikal yang telah dibangunkan. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Pangkalan Data [F3.3]** dan Langkah 4a di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

5.2 Skema Logikal Pangkalan Data

Seksyen ini adalah ruangan untuk menyediakan Skema Logikal Pangkalan Data. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Pangkalan Data [F3.3]** dan Langkah 4b di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

6. REKABENTUK MIGRASI DATA

Seksyen ini adalah ruangan untuk menerangkan secara ringkas berkenaan dengan Rekabentuk Migrasi Data serta rujukan kepada dokumen lain yang terlibat. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Migrasi Data [F3.8]**, dokumen **D05 - Pelan Migrasi Data**, **D06 - Spesifikasi Migrasi Data** dan Langkah 5a di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

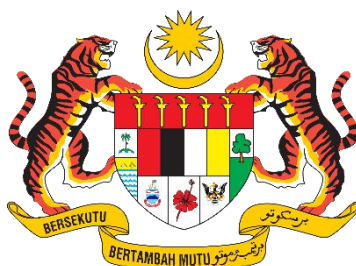
7. REKABENTUK INTEGRASI DATA

Seksyen ini adalah ruangan untuk menerangkan secara ringkas berkenaan dengan Rekabentuk Integrasi Data serta rujukan kepada dokumen lain yang terlibat. Sila rujuk kepada langkah-langkah di dalam **Rekabentuk Integrasi Data [F3.10]**, dokumen **D06 - Pelan Integrasi Data**, **D07 - Spesifikasi Integrasi Data** dan Langkah 5b di dalam **Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]** untuk mendapatkan penjelasan lanjut berkenaan cara-cara untuk melengkapkan seksyen ini.

8. LAMPIRAN

Seksyen ini merupakan ruangan untuk mengepilkkan dokumen-dokumen sokongan yang berkaitan dengan rekabentuk-rekabentuk sistem aplikasi.

RUJUKAN :



D05 DOKUMEN PELAN MIGRASI DATA

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat pindaan yang telah dilakukan kepada dokumen. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan angka selepas titik perpuluhan sahaja, contohnya, dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen.

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. TUJUAN

Perenggan ini menerangkan tujuan dokumen ini dihasilkan.

2. LATAR BELAKANG

Perenggan ini menerangkan secara ringkas latar belakang sistem yang terlibat untuk migrasi data.

3. OBJEKTIF MIGRASI

Perenggan ini menerangkan objektif migrasi data dilaksanakan.

4. SKOP MIGRASI

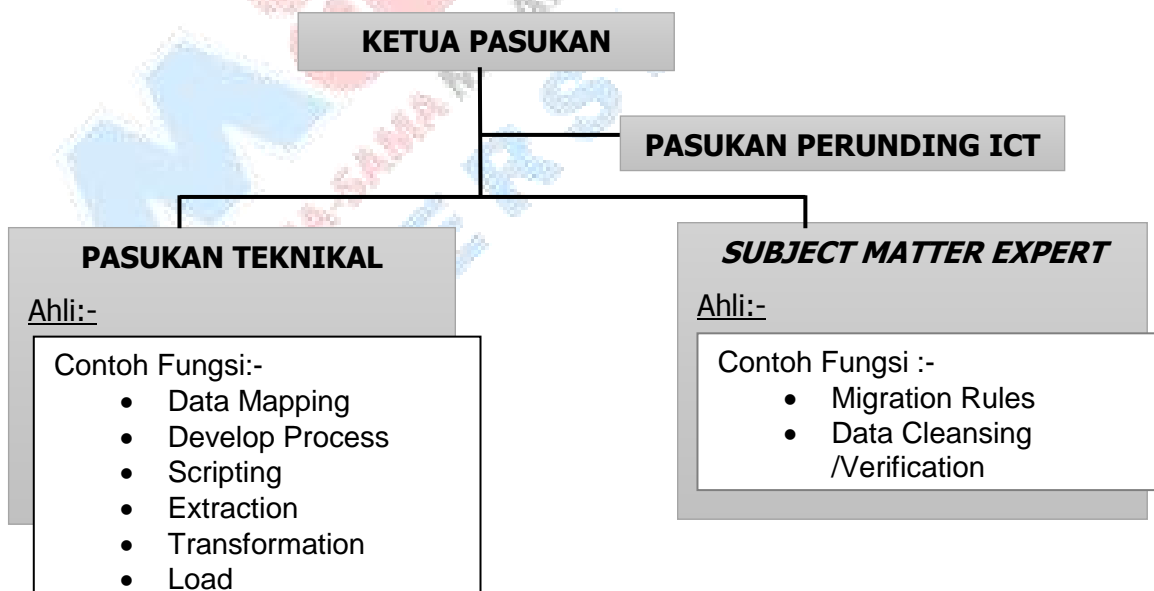
Perenggan ini menerangkan skop data yang akan dimigrasi.

5. PENDEKATAN MIGRASI

Perenggan ini menerangkan pendekatan pelaksanaan migrasi dan kaedah yang dipilih untuk memindahkan data.

6. PASUKAN PROJEK

Perenggan ini menerangkan struktur pasukan projek dan nama-nama pegawai yang terlibat. Contoh adalah seperti di bawah:



7. JADUAL PELAKSANAAN

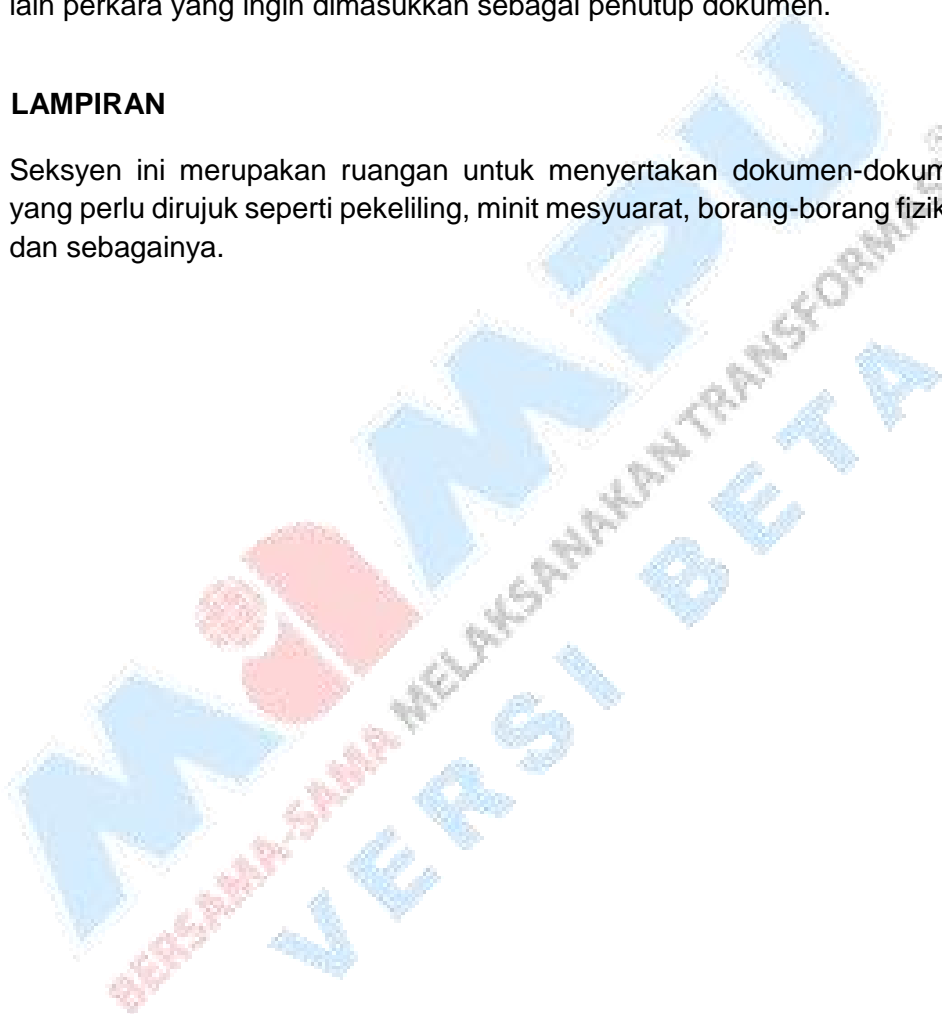
Perenggan ini menerangkan jadual pelaksanaan migrasi data yang menunjukkan semua proses migrasi bermula daripada analisis keperluan sehingga penyediaan laporan.

8. PENUTUP

Perenggan ini menerangkan tentang Faktor Kritikal Kejayaan (*Critical Success Factor*) yang perlu dikenal pasti untuk menjamin kejayaan pelaksanaan migrasi data dan lain-lain perkara yang ingin dimasukkan sebagai penutup dokumen.

9. LAMPIRAN

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti pekeliling, minit mesyuarat, borang-borang fizikal, surat-surat dan sebagainya.



RUJUKAN :

D06 DOKUMEN SPESIFIKASI MIGRASI DATA

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan. Contoh keterangan dokumen adalah seperti di bawah :

“Dokumen ini menyatakan Spesifikasi Migrasi Data yang akan dirujuk semasa fasa pembangunan. Ia bertujuan untuk menerangkan secara terperinci tujuan, maklumat sistem yang terlibat, maklumat data serta rangkaian sistem legasi, pemetaan data, pemetaan kod rujukan dan peraturan bisnes . “

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. TUJUAN DOKUMEN

Perenggan ini menerangkan tujuan dokumen ini dihasilkan.

2. MAKLUMAT SISTEM YANG TERLIBAT

Perenggan ini menerangkan maklumat sistem dan pangkalan data yang terlibat bagi tujuan migrasi data. Contoh adalah seperti di bawah:

Nama Sistem Legasi:

a) Maklumat Sistem Legasi

Bil	Modul terlibat	Data yang terlibat	Fungsi Sistem	Maklumat Sistem & Server									
				Dalam Pembangunan				Penyelenggaraan	Maklumat Server				
				Y	T	Language	SA/ CS/ WB	Inhouse/ Outsource	Nama	OS	IP	Zone (intranet/ DMZ/dll)	

Nota:

SA: Stand Alone

CS: Client-Server

WB: Web based

b) Maklumat Pangkalan Data dan Rangkaian Sistem Legasi

Nama	DBMS	Volume Data	Maklumat Pangkalan Data		Kepatuhan DDSA	Rangkaian		Catatan
			Arkitektur			Jenis		
			Hybrid/ Centralised/ Decentralised			1Gov*Net	Lain-lain (Nyatakan)	

Nota:

DBMS: Database Management System

DDSA: Data Dictionary Sektor Awam

Nama Sistem Baharu:**a) Maklumat Sistem Baharu**

Bil	Modul terlibat	Data yang terlibat	Fungsi Sistem	Maklumat Sistem & Server								
				Dalam Pembangunan				Penyelenggaraan	Maklumat Server			
				Y	T	Language	SA/CS/WB	Inhouse/Outsource	Nama	OS	IP	Zone (intranet/DMZ/dll)

Nota:

SA: Stand Alone

CS: Client-Server

WB: Web based

a) Maklumat Pangkalan Data dan Rangkaian Sistem Baharu

Maklumat Pangkalan Data					Rangkaian		Catatan
Nama	DBMS	Volume Data	Arkitektur	Kepatuhan DDSA	Jenis		
			Hybrid/ Centralised/ Decentralised		1Gov*Net	Lain-lain (Nyatakan)	

Nota:

DBMS: Database Management System

DDSA: Data Dictionary Sektor Awam

3. PEMETAAN JADUAL

Perenggan ini memaparkan rajah pemetaan jadual bagi sistem yang terlibat. Pendekatan pemetaan adalah merujuk kepada **Bab 4.10.2 Langkah 1**.

4. PERATURAN BISNES

Perenggan ini menerangkan peraturan bisnes yang perlu dipatuhi dalam proses migrasi data.

5. PEMETAAN DATA

Perenggan ini memuatkan jadual yang mengandungi perincian data yang perlu dipindahkan. Pendekatan pemetaan adalah merujuk kepada **Bab 4.10.2 Langkah 2**.

6. PEMETAAN KOD

Perenggan ini memuatkan jadual pemetaan kod yang disediakan untuk menyesuaikan dan menyelaraskan kod bagi atribut daripada sumber data dengan destinasi baharu data. Pendekatan pemetaan adalah merujuk kepada **Bab 4.10.2 Langkah 3**.

7. PEMETAAN REKOD (DATA)

Perenggan ini memuatkan jadual pemetaan rekod (data) yang disediakan untuk menyesuaikan dan menyelaraskan rekod data berdasarkan kod data baharu. Pendekatan pemetaan adalah merujuk kepada **Bab 4.10.2 Langkah 4**.

8. LAMPIRAN

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti format borang fizikal, format laporan dan pelbagai lagi dokumen-dokumen lain yang berkaitan.

RUJUKAN :



D07 DOKUMEN PELAN INTEGRASI SISTEM

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. **Keterangan Dokumen**

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan. Contoh keterangan dokumen adalah seperti di bawah :

ii. **Semakan dan Pengesahan Dokumen**

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat pindaan yang telah dilakukan kepada dokumen. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan angka selepas titik perpuluhan sahaja, contohnya, dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

vii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. TUJUAN DOKUMEN

Perenggan ini menerangkan tujuan dokumen ini dihasilkan.

2. OBJEKTIF

Seksyen ini menerangkan objektif pengintegrasian sistem.

3. SKOP KERJA INTEGRASI

Seksyen ini menerangkan skop kerja integrasi yang terlibat

4. PENDEKATAN DAN STRATEGI

Seksyen ini menerangkan kaedah dan strategi yang dilakukan untuk mengenalpasti integrase yang diperlukan dan pelaksanaan integrasi tersebut

5. KAEDAH INTEGRASI, TOOLS DAN PERSEKITARAN

Seksyen ini menerangkan kaedah dan strategi yang dilakukan untuk mengenalpasti integrasi yang diperlukan dan pelaksanaan integrasi tersebut.

6. TUGAS DAN TANGGUNGJAWAB

Seksyen ini menerangkan tugas dan tanggungjawab pasukan integrasi dan pihak berkepentingan

7. JADUAL PELAKSANAAN

Seksyen ini menyatakan tempoh masa yang diperlukan untuk setiap aktiviti yang dirancang.

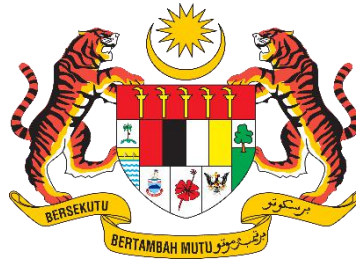
8. ANDAIAN DAN RISIKO

Seksyen menerangkan andaian yang dibuat sepanjang pelaksanaan integrasi dan potensi halangan yang akan memberi impak kepada pelaksanaan integrasi tersebut.

9. PENUTUP

Perenggan ini menerangkan tentang Faktor Kritikal Kejayaan (*Critical Success Factor*) yang perlu dikenal pasti untuk menjamin kejayaan pelaksanaan Integrasi data dan lain-lain perkara yang ingin dimasukkan sebagai penutup dokumen.

RUJUKAN :



D08 DOKUMEN SPESIFIKASI INTEGRASI DATA

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan. Contoh keterangan dokumen adalah seperti di bawah :

“Dokumen ini menyatakan spesifikasi integrasi sistem yang akan dirujuk semasa fasa pembangunan sistem. Ia bertujuan untuk menerangkan secara terperinci tujuan, skop, gambaran keseluruhan, analisis-analisis keperluan sistem, maklumat dan proses bagi perisian yang akan dibangunkan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat pindaan yang telah dilakukan kepada dokumen. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat angka selepas titik perpuluhan sahaja, contohnya, dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. TUJUAN DOKUMEN

Perenggan ini menerangkan tujuan dokumen ini dihasilkan. Senaraikan kumpulan sasaran dokumen ini. Nyatakan andaian, batasan dan kekangan dalam pembangunan integrasi.

2. KEPERLUAN INTEGRASI

Terangkan integasi yang diperlukan oleh sistem. Kenalpasti servis yang diperlukan.

Bil	Rujukan Fungsi	Rujukan Aktiviti	Nama Sistem Sumber	Pemilik Maklumat	Keterangan Maklumat yang dihantar	Tujuan Penggunaan Maklumat
1						
2						

3. KAEDAH INTEGRASI DATA

Penerangan setiap servis integrasi yang dibangunkan.

Nama Servis						
Keterangan						
Kaedah Integrasi						
URL Web Service						
Request						
Respond						
Data yang terlibat	Nama	Jenis	Saiz	Nullable	Rules	

4. PEMETAAN DATA

Lakukan pemetaan data antara sistem yang memohon data dengan data yang diterima.

Sistem yang memohon (<i>request</i>)				Data yang diterima			
Nama Medan	Jenis	Saiz	Keterangan	Nama Medan	Jenis	Saiz	Keterangan

5. PROSES PERTUKARAN DATA

Terangkan aliran proses semasa pertukaran data tersebut. Nyatakan juga sebarang peraturan yang meyebabkan pertukaran data berlaku.

6. REKA BENTUK SENIBINA INTEGRASI

Perincikan reka bentuk senibina integrasi yang terlibat. Ia juga menggambarkan keperluan infrastruktur seperti rangkaian untuk persekitaran pembangunan dan pemngimplementasian serta komponen perisian yang digunakan untuk mewujudkan proses integrasi.

RUJUKAN :

D09 DOKUMENTASI PANGKALAN DATA

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. **Keterangan Dokumen**

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. **Semakan dan Pengesahan Dokumen**

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. **PENGENALAN**

Seksyen ini menerangkan pangkalan data bagi sistem yang dibangunkan. Terangkan secara ringkas latar belakang sistem dan kandungan yang terdapat dalam dokumen ini.

2. **RINGKASAN MAKLUMAT PANGKALAN DATA FIZIKAL YANG DIBANGUNKAN**

Seksyen ini adalah ruang untuk menerangkan maklumat pangkalan data fizikal yang dibangunkan seperti berikut.

- a) Pangkalan data yang digunakan.
- b) Peruntukan ruang jadual (tablespace/filegroup)
- c) Nama pangkalan data
- d) Bilangan table, view
- e) Stored procedure
- f) Maklumat jadual dan medan yang mengandungi INDEX
- g) Senarai pengguna yang mempunyai kawalan akses ke atas pangkalan data/jadual

3. **SKRIP PANGKALAN DATA**

Seksyen ini adalah ruangan untuk skrip pangkalan data yang digunakan untuk membangunkan pangkalan data fizikal.

4. **LAMPIRAN**

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti pekeliling, minit mesyuarat, borang-borang fizikal, surat-surat dan sebagainya.

RUJUKAN :



D10 DOKUMENTASI KOD SUMBER

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan. Contoh keterangan dokumen adalah seperti di bawah :

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim**Akronim**

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. TUJUAN DOKUMEN

Perenggan ini menerangkan tujuan dokumen ini dihasilkan. Senaraikan kumpulan sasar dokumen ini. Nyatakan andaian, batasan dan kekangan dalam menyediakan dokumentasi kod sumber ini.

2. SKOP DOKUMEN

Perenggan ini menyatakan skop bagi penyediaan kod sumber. Berikut adalah contoh maklumat yang boleh dinyatakan:

- i) Nama sistem yang terlibat
- ii) Nama modul yang terlibat
- iii) Nama pasukan pembangunan sistem yang terlibat

3. PIAWAIAN KOD SUMBER

Nyatakan piawaian kod sumber (cara penulisan kod) yang digunakan semasa membangunkan sistem. Pasukan pembangun sistem boleh menyenaraikan kod yang bersesuaian mengikut keperluan. Berikut antara contoh kod sumber yang perlu dipatuhi.

i) File Name

<Nyatakan nama fail yang terlibat>

Contoh:

.java, .php

ii) Class Headers and Declaration

<Isytiharkan Class Header yang mengandungi class name, inheritance, attributes, methods, functionality, visibility, requirement number dan pernyataan yang mengisytiharkan Class tersebut>

Contoh:

```
/******
```

```
* Class name:
```

```
* Inheritance:
```

```
* Attributes:
```

```
* Methods:
```

```
* Functionality:
```

```
* Visibility:
```

```
* From requirement number
```

```
*****/
```

```
public class ClassName
```

iii) Method Headers and Declaration

<Isytiharkan Class Header yang mengandungi method name, inheritance, attributes, precondition, postcondition, functionality, visibility, @param, @return dan requirement numbers>

Contoh:

```

/*****
* Method name:
* Inheritance:
* Attributes:
* Precondition:
* Postcondition:
* Functionality:
* Visibility:
* @param:
* @return:
* From requirement number
*****/

```

iv) Indentation

<Nyatakan kaedah indentation yang digunakan>

Contoh:

Four spaces should be used as the unit of indentation

v) Inline Comments

<Nyatakan kaedah inline comment yang digunakan>

Contoh:

Inline comment should make up 20% of the total lines of code in a program, excluding the header documentation blocks.

vi) Variable Names

<Nyatakan kaedah penulisan variable names yang digunakan>

Contoh:

Variable shall have mnemonic or meaningful names that convey to a casual observer, the intent of its use. Variables shall be initialized prior to its first use.
String tempWord;

vii) Use of Braces

<Nyatakan gaya penggunaan Braces>

Contoh:

Style 1:

```

for (int j = 0 ; j < max_iterations ; ++j)
{
    /* Some work is done here. */
}

```

Style 2:

```
for ( int j = 0 ; j < max_iterations ; ++j )
{
    /* Some work is done here. */
}
```

viii) Line Length

<Nyatakan line length yang digunakan>

Contoh:

Keep the lengths of source code lines at or below 80 characters.

ix) Spacing

<Nyatakan kaedah spacing yang digunakan>

Contoh:

- i. A keyword followed by a parenthesis should be separated by a space.
- ii. A blank space should appear after each comma in an argument list.

```
cost = price + ( price * sales_tax );
fprintf (stdout, "The total cost is %5.2f\n", cost) ;
```

x) Wrapping Lines

<Nyatakan kaedah wrapping lines yang digunakan>

Contoh:

- i. Break after a comma

```
fprintf ( stdout , "\nThere are %d reasons to use standards\n"
        num_reasons ) ;
```

- ii. Break after an operator

```
long int total_apples = num_my_apples + num_his_apples
                        num_her_apples ;
```

xi) Program Statement

<Nyatakan kaedah penulisan program statement yang digunakan>

Contoh:

Program statement should be limited to one per line. Also nested statements should be avoided when possible.

```
number_of_names = names.length ;
b = new JButton [ number_of_names ] ;
```

4. LAMPIRAN

Seksyen ini merupakan ruangan untuk menyertakan dokumen-dokumen sokongan yang perlu dirujuk seperti format borang fizikal, format laporan dan pelbagai lagi dokumen-dokumen lain yang berkaitan.

RUJUKAN :



D11 DOKUMEN LAPORAN UJIAN SISTEM

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

Dengan ini adalah disahkan Ujian Sistem telah selesai dilaksanakan dengan sempurna.

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

1. PENGENALAN

1.1 Tujuan Laporan

Seksyen ini adalah ruangan untuk menerangkan tujuan laporan ini dibangunkan.

1.2 Skop Laporan

Seksyen ini adalah ruangan untuk menyenaraikan modul dan sub modul yang terlibat dalam Ujian sistem merujuk kepada Dokumen SRS.

2. AKTIVITI PENGUJIAN

2.1 Serahan

Seksyen ini adalah ruangan untuk menyenaraikan status laporan ujian sistem yang telah dilaksanakan.

Bil	Modul & Sub Modul	Status Pengujian (Lulus/Gagal)	Keterangan
1.			
2.			

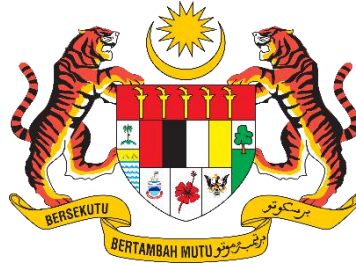
2.2 Rumusan Serahan

Bil	Modul	Bil Lulus	Bil Gagal	Keterangan
1.				
2.				
3.				
4.	Jumlah			
	Peratus Lulus/Gagal			

3. DOKUMEN SOKONGAN

Seksyen ini menyenaraikan dokumen sokongan yang berkaitan dengan laporan ujian yang telah dilaksanakan.

RUJUKAN :



D12 DOKUMEN PELAN INDUK PENGUJIAN

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Kawalan Dokumen

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibatkan perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. PENGENALAN PROJEK

Pengenalan projek memberi gambaran mengenai projek secara keseluruhan. Penjelasan mengenai pengenalan projek boleh dirujuk di Langkah 2, Pelan Induk Pengujian [F5.1]. Pengenalan projek mengandungi sub-topik seperti berikut :

1.1 Tujuan Projek

Seksyen ini menerangkan berkenaan tujuan projek yang dibangunkan.

1.2 Skop Projek

Seksyen ini menerangkan berkenaan skop projek yang dibangunkan.

1.3 Senarai Pemegang Taruh

Seksyen ini menyenaraikan pemegang taruh bagi projek yang dibangunkan.

1.4 Sumber Rujukan

Seksyen ini menyenaraikan sebarang dokumen atau sumber yang dirujuk.

1.5 Glosari

Seksyen ini menerangkan berkenaan glosari yang digunakan di dalam dokumen.

2. KONTEKS PENGUJIAN

2.1 Item ujian

Seksyen ini menerangkan item yang akan diuji.

contoh berikut :

Modul-modul yang terlibat pengujian ini adalah :

Modul 1 : Pendaftaran Pengguna

Modul 2 : Tempahan bilik mesyuarat

Modul 3 : Laporan

Modul 4 : Pentadbir sistem

Modul 5 : ...

2.2 Skop ujian

Seksyen ini menerangkan liputan ujian yang dilaksanakan.

Contoh bagi penentuan skop ujian adalah seperti berikut :

Skop ujian adalah untuk menguji modul yang telah dinyatakan di dalam Item Ujian. Selain itu, ujian juga akan menilai *usability*, *performance efficiency* dan keselamatan modul yang telah dinyatakan di atas.

2.3 Kekangan

Seksyen ini menerangkan kekangan atau halangan yang berkemungkinan berlaku semasa pengujian.

Kekangan adalah sekatan atau halangan yang dijangka akan berlaku semasa pengujian dilaksanakan. Kekangan ini boleh mengganggu perjalanan proses pengujian dan berpotensi untuk memberi risiko ke atas sistem aplikasi. Kekangan dengan kebarangkalian yang tinggi untuk berlaku perlu dinyatakan di dalam dokumen ini.

3. KAEDAH KOMUNIKASI

Seksyen ini menerangkan peranan, tindakan dan medium komunikasi bagi sesuatu situasi yang berlaku di dalam aktiviti pengujian.

Contoh Komunikasi Pengujian

SITUASI	PERANAN	TINDAKAN	MEDIUM KOMUNIKASI
Semakan dan pengesahan Pelan Ujian UAT	Pengurus Ujian / Penguji / Pemilik Sistem	Semakan melalui sesi walkthrough	<ul style="list-style-type: none"> Mesyuarat / Perbincangan Emel Surat Telefon
Penemuan ralat dalam ujian	Penguji	Melaporkan ralat	<ul style="list-style-type: none"> Borang laporan insiden <i>Testing tool / system</i>
	Pasukan pembangun sistem	Memperbaiki ralat yang dilaporkan	Kemaskini status ralat di dalam : <ul style="list-style-type: none"> Borang laporan insiden <i>Testing tool / system</i>

4. DAFTAR RISIKO

Seksyen ini menerangkan kaedah pengurusan risiko yang akan dilaksanakan semasa pengujian.

Skala dan Contoh Risiko Pengujian

Contoh skala yang digunakan :

Rendah	Sederhana rendah	Sederhana	Sederhana tinggi	Tinggi
1	2	3	4	5

Contoh :

RISIKO PRODUK	KEBARANG-KALIAN	IMPAK	TAHAP RISIKO	CADANGAN MITIGASI
Pengiraan yang tidak tepat dalam sistem	2	5	10	Laksanakan ujian unit
Bilangan pembangun sistem yang berkemahiran tidak mencukupi	3	5	15	<ul style="list-style-type: none"> Beri latihan kemahiran kepada pembangun sistem Melantik pembangun sistem tambahan secara kontrak

5. STRATEGI UJIAN

Seksyen ini menerangkan strategi yang digunakan untuk melaksanakan ujian.

- Sub-proses ujian
- Serahan Ujian
- Entry criteria* dan *Exit criteria*
- Pengukuran Metrik
- Data Ujian / Simulasi
- Persekitaran ujian
- Penetapan Tahap *Severity*

Contoh Tahap Severity Hasil Ujian

Tahap Severity			Keterangan
Pilihan 1	Pilihan 2	Pilihan 3	
1	<i>Critical</i>	<i>High</i>	Ralat kritikal yang menyebabkan kegagalan fungsi sistem seperti kehilangan data atau kegagalan fungsi utama (blocker/showstopper).
2	<i>Major</i>	<i>Medium</i>	Ralat yang kritikal tetapi tidak melibatkan kehilangan data dan kegagalan fungsi sistem seperti output yang tidak sepadan atau laporan yang tidak tepat.
3	<i>Minor</i>	<i>Low</i>	Ralat yang tidak menjejaskan fungsi sistem seperti kesilapan ejaan dan label.

6. JADUAL PERANCANGAN UJIAN

Seksyen ini menerangkan perancangan yang terperinci bagi aktiviti pengujian.

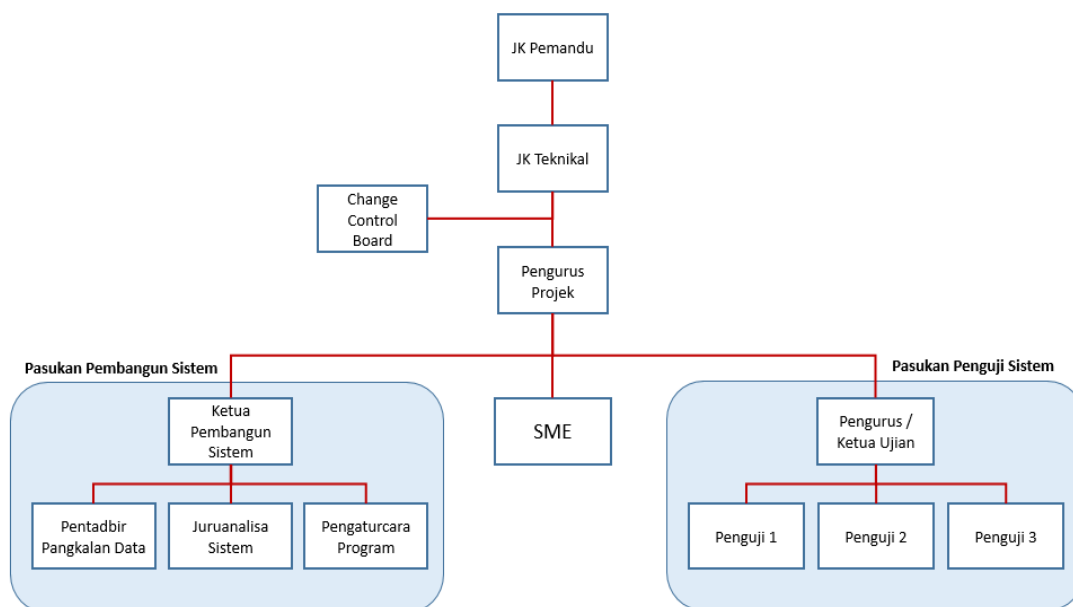
Contoh Perancangan Pengujian

BIL	AKTIVITI	MAC				APR				MEI				JUN			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
5.	UAT	■	■	■	■												
6.	Pembetulan ralat UAT					■	■	■									
7.	Ujian semula UAT								■	■							
8.	PAT										■	■	■	■	■		

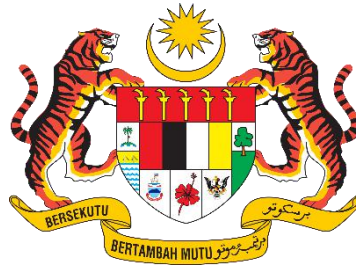
7. STRUKTUR PERJAWATAN

Seksyen ini menerangkan berkenaan struktur organisasi projek termasuk pasukan pengujian.

Contoh struktur organisasi adalah sebagaimana dibawah:



RUJUKAN :



D13 DOKUMEN PELAN UJIAN PENERIMAAN (UAT-PAT)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. **Keterangan Dokumen**

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. **Semakan dan Pengesahan Dokumen**

Seksyen ini adalah ruangan bagi pegawai-pegawai yang bertanggungjawab untuk melakukan semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen ini. Sila sertakan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan. Contoh ruangan semakan dan pengesahan adalah seperti berikut :

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. **Kawalan Dokumen**

Seksyen ini adalah ruangan untuk mencatatkan maklumat-maklumat penyediaan dokumen termasuk maklumat pindaan yang telah dilakukan ke atas dokumen ini. Sila masukkan nombor versi, tarikh, ringkasan pindaan dan nama penyedia di dalam jadual seperti di bawah :

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

Penentuan nombor versi adalah bergantung kepada saiz pindaan kepada dokumen yang telah dilakukan. Sekiranya saiz pindaan yang dilakukan adalah kecil ataupun sederhana, perubahan nombor versi hanya melibat perubahan angka selepas titik perpuluhan sahaja, contohnya, perubahan dari nombor versi 1.2 kepada 1.3. Sekiranya pindaan yang dilakukan adalah besar dan melibatkan perubahan kepada kandungan dokumen, gunakan angka nombor yang seterusnya, contohnya, dari nombor 1.2 kepada 2.0.

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vii. Definisi dan Akronim

Akronim

Sub seksyen ini adalah ruangan untuk menerangkan akronim-akronim yang digunakan di dalam dokumen. Contoh pengisian di ruangan akronim adalah seperti jadual di bawah :

Akronim	Keterangan

Definisi

Sub seksyen ini adalah ruangan untuk menerangkan definisi bagi terma atau istilah yang digunakan di dalam dokumen. Contoh pengisian di ruangan definisi adalah seperti jadual di bawah :

Terma/Istilah	Definisi

viii. Sumber Rujukan

Seksyen ini adalah ruangan untuk menyenaraikan semua sumber-sumber rujukan yang digunakan di dalam penyediaan dokumen ini, contohnya seperti surat pekeliling perkhidmatan, manual prosedur kerja, garis-garis panduan, dokumen-dokumen piawaian ISO/IEC/IEEE dan bahan rujukan lain yang berkaitan.

1. PENGENALAN PROJEK

Pengenalan projek memberi gambaran mengenai projek secara keseluruhan. Penjelasan mengenai pengenalan projek boleh dirujuk di Langkah 1, Pelan Ujian (UAT & PAT) [F5.4]. Pengenalan projek mengandungi sub-topik seperti berikut :

1.1 Tujuan Projek

Seksyen ini menerangkan berkenaan tujuan projek yang dibangunkan.

1.2 Skop Projek

Seksyen ini menerangkan berkenaan skop projek yang dibangunkan.

1.3 Sumber Rujukan

Seksyen ini menerangkan dokumen yang dirujuk bagi melaksanakan pengujian sistem.

1.4 Glosari

Seksyen ini menerangkan berkenaan glosari yang digunakan di dalam dokumen.

2. KONTEKS PENGUJIAN

2.1 Item ujian

Seksyen ini menerangkan item yang akan diuji. Penjelasan mengenai item ujian boleh dirujuk di item (a), Langkah 2, Pelan Ujian (UAT & PAT) [F5.3].

2.2 Skop ujian

Seksyen ini menerangkan liputan ujian yang dilaksanakan. Penjelasan mengenai skop ujian boleh dirujuk di item (b), Langkah 2, Pelan Ujian (UAT & PAT) [F5.3].

2.3 Kekangan

Seksyen ini menerangkan kekangan atau halangan yang berkemungkinan berlaku semasa pengujian. Penjelasan mengenai kekangan ujian boleh dirujuk di item (c), Langkah 2, Pelan Ujian (UAT & PAT) [F5.3].

3. KAEDAH KOMUNIKASI

Seksyen ini menerangkan kaedah komunikasi yang digunakan oleh pasukan pengujian. Penjelasan mengenai kaedah komunikasi boleh dirujuk di Langkah 3, Pelan Ujian (UAT & PAT) [F5.3].

4. DAFTAR RISIKO

Seksyen ini menerangkan kaedah pengurusan risiko yang akan dilaksanakan semasa pengujian. Penjelasan mengenai daftar risiko boleh dirujuk di Langkah 4, Pelan Ujian (UAT & PAT) [F5.3].

5. STRATEGI UJIAN

Seksyen ini menerangkan risiko yang dikenal pasti di dalam projek. Penjelasan mengenai strategi ujian boleh dirujuk di Langkah 5, Pelan Ujian (UAT & PAT) [F5.3].

- a) Pendekatan / strategi ujian
- b) Serahan Ujian
- c) Teknik Rekabentuk Ujian
- d) Kriteria masuk dan kriteria keluar
- e) Metrik
- f) Data Ujian
- g) Persekitaran ujian
- h) Kriteria Penangguhan Dan Penyambungan Semula (Suspension and resumption criteria)

6. JADUAL AKTIVITI PENGUJIAN

Seksyen ini menyenaraikan aktiviti di dalam pengujian dan jadual pengujian. Penjelasan mengenai jadual aktiviti pengujian boleh dirujuk di Langkah 6, Pelan Ujian (UAT & PAT) [F5.3].

Seksyen ini menerangkan berkenaan struktur organisasi di dalam pengujian. Penjelasan mengenai struktur perjawatan boleh dirujuk di Langkah 7, Pelan Ujian (UAT & PAT) [F5.3].

7. LAMPIRAN

Semua dokumentasi ujian seperti senario ujian, prosedur ujian, kes ujian dan traceability matrix dilampirkan di dalam Pelan Ujian UAT/PAT ini.. Penjelasan mengenai struktur perjawatan boleh dirujuk di Langkah 8, Pelan Ujian (UAT & PAT) [F5.3].

RUJUKAN :



D14 LAPORAN UJIAN PENERIMAAN PENGGUNA (UAT/PAT)

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan.

ii. Butiran Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai penyedia, semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

Dengan ini adalah disahkan Ujian Sistem telah selesai dilaksanakan dengan sempurna.

DOKUMEN DISEDIAKAN OLEH

Nama	Jawatan	Tandatangan	Tarikh

DOKUMEN DISEMAK OLEH

Nama	Jawatan	Tandatangan	Tarikh

DOKUMEN DISAHKAN OLEH

Nama	Jawatan	Tandatangan	Tarikh

iii. Pindaan Dokumen

Seksyen ini adalah ruangan untuk menyatakan senarai versi dan perubahan terhadap dokumen.

Versi	Seksyen	Senarai Perubahan	Tarikh	Catatan

iv. Kandungan

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

1. UJIAN YANG DIJALANKAN

Seksyen ini menerangkan Penerangan ringkas terhadap ujian UAT/PAT yang telah dijalankan.

2. PERUBAHAN DARIPADA PELAN PENGUJIAN

Seksyen ini menerangkan perubahan yang berlaku semasa pelaksanaan pengujian berbanding Pelan Pengujian.

3. PENILAIAN PENAMATAN UJIAN

Seksyen ini menerangkan ringkasan situasi yang membolehkan penamatan ujian.

4. FAKTOR YANG MENGHALANG KEMAJUAN

Seksyen ini menerangkan faktor-faktor yang menghalang atau melewatkan kemajuan ujian.

5. PENGUKURAN HASIL UJIAN

Seksyen ini menerangkan dapatan hasil ujian.

6. RISIKO

Seksyen ini menerangkan risiko baru, perubahan risiko dan risiko sedia ada (yang belum dapat diselesaikan) setelah ujian dilaksanakan.

7. SERAHAN UJIAN

Seksyen ini menerangkan semua serahan ujian yang perlu dihantar.

8. ASET UJIAN YANG BOLEH DIGUNAKAN SEMULA

Seksyen ini menyenaraikan spesifikasi ujian, persekitaran ujian dan data ujian yang boleh digunakan semula bagi kitaran ujian seterusnya.

9. LESSON LEARNED

Seksyen ini menerangkan pengajaran yang boleh dirujuk dan diperbaiki dalam kitaran ujian akan datang.

RUJUKAN :



D15 DOKUMEN LAPORAN MIGRASI DATA

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. **Keterangan Dokumen**

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan.

ii. **Semakan dan Pengesahan Dokumen**

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen. Sila masukkan maklumat seperti nama, jawatan, tandatangan dan tarikh semakan atau kelulusan.

Contoh ruangan semakan dan pengesahan adalah seperti berikut :

Dengan ini adalah disahkan Ujian Sistem telah selesai dilaksanakan dengan sempurna.

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. **Kandungan**

Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

iv. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

v. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.



1. PENGENALAN PROJEK

Perenggan ini menerangkan ringkasan pengenalan dan objektif projek migrasi.

2. JADUAL PELAKSANAAN ASAL DAN SEBENAR

Perenggan ini menunjukkan jadual pelaksanaan yang sebenar.

3. STATUS MIGRASI

Perenggan ini menerangkan status migrasi data daripada sumber data ke destinasi baharu.

4. SUMBER DATA

Perenggan ini menerangkan maklumat lengkap server, nama pangkalan data, table sumber dan lain-lain.

5. DESTINASI BAHARU DATA

Perenggan ini menerangkan maklumat server, nama pangkalan data, *table* destinasi baharu dan lain-lain.

6. JUMLAH BARIS DALAM TABLE SUMBER

Perenggan ini menyatakan jumlah baris dalam *table* sumber.

7. JUMLAH BARIS YANG BERJAYA DIMIGRASI

Perenggan ini menyatakan jumlah baris (*row*) yang berjaya dimigrasi bagi setiap *table*.

8. RATIO

Perenggan ini menyatakan peratusan baris (*row*) yang berjaya dimigrasi.

9. PERINCIAN

Perenggan ini menerangkan perincian maklumat sekiranya terdapat migrasi data yang gagal. Sebab kegagalan dan tindakan pembetulan perlu dinyatakan dengan terperinci.

10. LAMPIRAN

Sila lampirkan struktur *table* destinasi baharu berserta kamus data bagi pangkalan data baharu serta lain-lain dokumen yang perlu

RUJUKAN :



D16 DOKUMEN LAPORAN PENAMATAN UJIAN

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. Semakan dan Pengesahan Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen.

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh Semakan

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh Semakan

iii. Pindaan Dokumen

Seksyen ini adalah ruangan untuk menyatakan senarai versi dan perubahan terhadap dokumen.

Versi	Seksyen	Senarai Perubahan	Tarikh	Catatan

iv. Kandungan

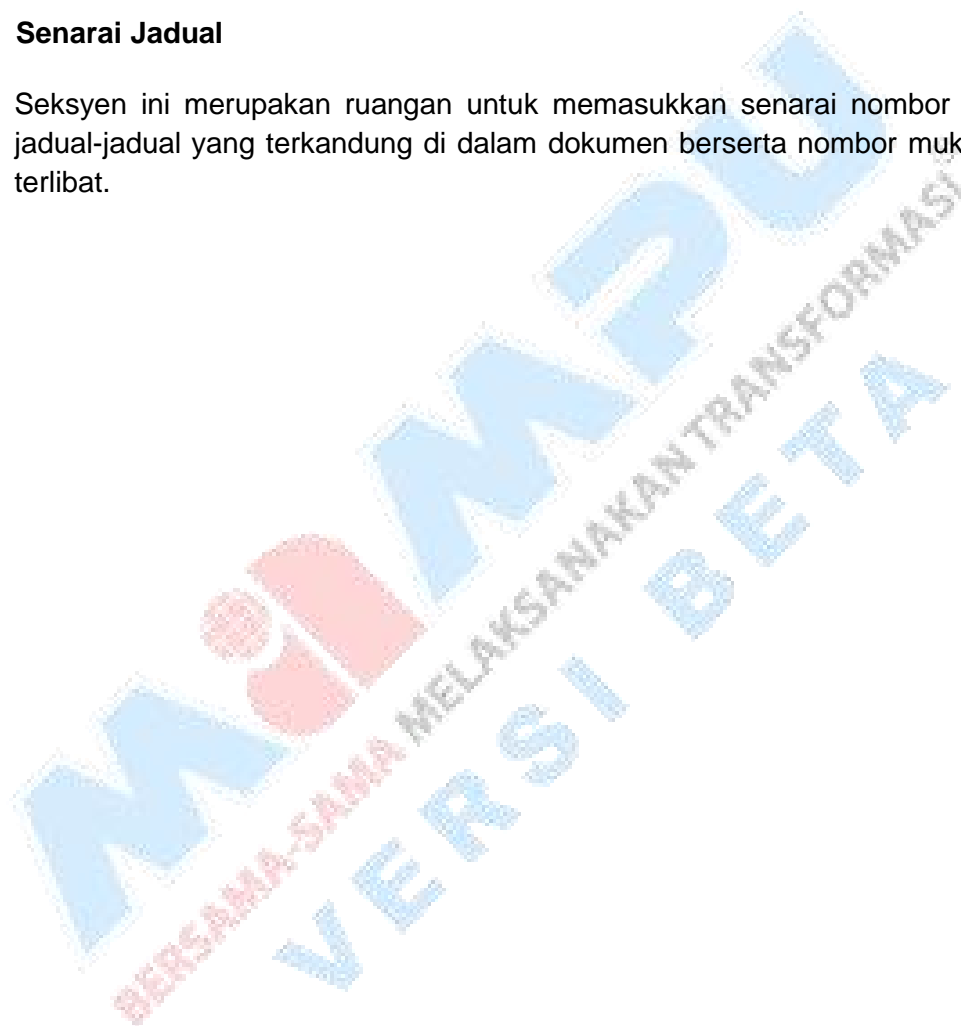
Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

v. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

vi. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.



1. MAKLUMAT SPESIFIK DOKUMEN

Seksyen ini mengandungi maklumat asas seperti nombor ID, versi, tarikh dihasilkan dan tarikh dikemaskini, organisasi, pengesahan serta rekod pindaan dokumen

2. OBJEKTIF UJIAN

Seksyen ini menerangkan mengenai objektif ujian yang dijalankan.

3. SKOP UJIAN

Seksyen ini menerangkan skop ujian bagi menetapkan perimeter ujian termasuk ciri-ciri (fungsi dan kualiti/ bukan fungsi) yang akan diuji ke atas item ujian dan ciri-ciri yang tidak akan diuji ke atas item ujian

4. BUTIRAN UJIAN

Seksyen ini menerangkan butiran tempoh Ujian Penamatan akan dilaksanakan dan peserta yang akan terlibat.

5. KEPERLUAN PERSEKITARAN

Seksyen ini menerangkan Keperluan persekitaran merujuk kepada keperluan perkakasan dan perisian yang digunakan semasa sesi Ujian Penamatan.

6. HASIL UJIAN

Seksyen ini menerangkan dapatan hasil ujian seperti berikut:

- i) Isu-isu FAT yang dilaporkan
- ii) Kategori Isu-isu FAT (Functionality, Usability or Operational)
- iii) Status Isu-isu FAT
- iv) Ulasan Keputusan Ujian

7. RUMUSAN

Seksyen ini Mengandungi penerangan rumusan hasil daripada ujian FAT.

RUJUKAN :

D17 DOKUMEN MANUAL PENGGUNA SISTEM

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Bahagian ini menyatakan secara ringkas berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. Kawalan Dokumen

Bahagian ini adalah ruangan untuk mencatatkan maklumat pindaan yang telah dilakukan kepada dokumen.

KAWALAN DOKUMEN

No. Versi	Tarikh	Ringkasan Pindaan	Penyedia

iii. Kandungan

Bahagian ini merupakan ruangan untuk memasukkan senarai isi kandungan dokumen berserta dengan nombor muka surat yang terlibat.

iv. Senarai Gambarajah

Bahagian ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

v. Senarai Jadual

Bahagian ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

1. PENGENALAN

Manual Pengguna mengandungi semua maklumat penting bagi pengguna untuk menggunakan sepenuhnya sistem maklumat. Manual ini termasuklah penerangan tentang fungsi sistem dan keupayaan, kontijensi dan mod alternatif operasi, dan prosedur langkah demi langkah untuk akses sistem dan kaedah penggunaannya. Gunakan gambarajah jika ada keperluan dalam manual ini. Format manual boleh diubah jika format lain adalah lebih sesuai untuk projek tertentu

1.1 Tujuan dan Skop

Bahagian ini memberikan penerangan mengenai tujuan dan skop Manual Pengguna.

1.2 Organisasi Manual

Bahagian ini menerangkan tentang organisasi Manual Pengguna.

1.3 Maklumat Untuk Dihubungi

Bahagian ini menerangkan tentang kod organisasi dan kakitangan yang boleh membantu pengguna sistem. Jika kemudahan khidmat bantuan (helpdesk) di sesebuah organisasi adalah disediakan, maka ia perlu dijelaskan di bahagian ini.

1.4 Rujukan Projek

Bahagian ini menyediakan bibliografi rujukan projek utama dan serahan yang telah dihasilkan semasa tempoh projek.

1.5 Fungsi Utama Sistem

Bahagian ini menerangkan mengenai perspektif bisnes dan tanggungjawab pengguna yang akan disokong oleh sistem. Ia juga akan menjelaskan mengenai fungsi-fungsi bisnes supaya ia dapat menyokong fungsi-fungsi yang ada di dalam sistem.

1.6 Glosari

Bahagian ini menyediakan glosari semua terma dan singkatan yang digunakan dalam manual ini. Jika glosari mempunyai beberapa halaman atau lebih panjang, ia boleh diletakkan sebagai lampiran.

2. OVERVIEW SISTEM

Bahagian ini memberikan gambaran ringkas mengenai sistem dan keupayaannya.

2.1 Tujuan

Bahagian ini menerangkan tentang tujuan sistem aplikasi dibangun dan dilaksanakan.

2.2 Keterangan Sistem

Bahagian ini memberikan gambaran keseluruhan mengenai keupayaan sistem, fungsi dan operasi serta fungsi peringkat tinggi yang akan dijalankan oleh sistem. Penggunaan gambarajah dan jadual boleh digunakan jika ia bersesuaian.

3. KETERANGAN FUNGSI SISTEM

Bahagian ini menerangkan tentang setiap fungsi yang ada di dalam sistem. Ia akan menggambarkan berkenaan dengan konvensyen yang digunakan dalam subseksyen yang berkaitan. Setiap satu bahagian berikutnya perlu diulangi seberapa kerap yang perlu untuk menerangkan setiap fungsi dalam sistem.

3.1 Senarai Fungsi Sistem

Bahagian ini menyediakan senarai tajuk bagi fungsi-fungsi yang ada di dalam sistem.

3.2 Perincian Keterangan bagi Fungsi Sistem

Bahagian ini menyediakan keterangan bagi setiap fungsi di dalam sistem. Ia merangkumi perkara-perkara seperti berikut :

- a) Tujuan dan kegunaan fungsi
- b) Pengawalan fungsi, jika berkenaan
- c) Pilihan Pelaksanaan yang berkaitan dengan fungsi ini
- d) Keterangan input fungsi
- e) Keterangan output dan hasil yang diharapkan
- f) Hubungan dengan fungsi lain
- g) Ringkasan operasi fungsi

4. ARAHAN PENGGUNAAN SISTEM

Bahagian ini menyediakan arahan terperinci langkah demi langkah bagi kaedah pengoperasian sistem.

4.1 Log Masuk Sistem

Bahagian ini menerangkan berkaitan prosedur untuk log masuk sistem dan kawalan paparan sistem, seperti skrin menu utama sistem. Prosedur pengawalan hendaklah menerangkan tentang kaedah untuk menentukan mod yang diperlukan operasi dan menetapkan apa-apa parameter permulaan yang diperlukan untuk operasi. Prosedur pemasangan perisian perlu dimasukkan jika perisian diedarkan pada disket dan perlu dimuat turun terlebih dahulu sebelum digunakan.

4.2 Proses Pengoperasian Sistem

Bahagian ini menyatakan prosedur untuk melaksanakan operasi sistem di mana ia memerlukan tindak balas daripada pengguna.

4.3 Penamatan dan Pengoperasi Semula Sistem

Bahagian ini menerangkan tentang prosedur untuk penamatan secara normal dan tidak berjadual operasi sistem. Selain itu, ia juga perlu menyatakan cara untuk memulakan semula sistem setelah operasi sistem ditamatkan.

5. PENGENDALIAN RALAT

Bahagian ini perlu menyatakan mesej ralat dan kemudahan bantuan kepada pengguna sistem. Maklumat tambahan dan subseksyen boleh ditambah seperti yang diperlukan. Senaraikan semua mesej ralat seperti berikut:

- a) Mana-mana kod ralat yang berkaitan dengan mesej ralat
- b) Keterangan bagi maksud setiap mesej ralat
- c) Perbincangan bagaimana untuk menyelesaikan ralat

5.1 Bantuan Helpdesk

Bahagian ini menerangkan tentang perisian khidmat bantuan atau kemudahan khidmat bantuan atau kontraktor bantuan yang pengguna boleh hubungi untuk menyelesaikan ralat. Nama pegawai yang bertanggungjawab berserta nombor telefon meja bantuan hendaklah dimasukkan untuk rujukan pengguna sistem.

RUJUKAN :



D18 DOKUMEN LAPORAN SERAHAN SISTEM

NAMA SISTEM

(Sertakan nama modul di bawah nama sistem sekiranya dokumen disediakan secara berasingan bagi setiap modul di bawah sistem yang sama)

NAMA AGENSI	:	
NAMA AGENSI INDUK	:	
TARIKH DOKUMEN	:	
VERSI DOKUMEN	:	

i. Keterangan Dokumen

Seksyen ini adalah ruangan untuk menyatakan secara ringkas keterangan berkenaan dokumen yang disediakan dengan merujuk kepada piawaian antarabangsa yang berkaitan.

ii. Butiran Dokumen

Seksyen ini adalah ruangan bagi pegawai yang terlibat untuk menurunkan tandatangan sebagai semakan dan pengesahan kepada maklumat-maklumat yang terkandung di dalam dokumen.

SEMAKAN DOKUMEN

Disemak Oleh	Jawatan	Tandatangan	Tarikh

PENGESAHAN DOKUMEN

Disahkan Oleh	Jawatan	Tandatangan	Tarikh

iii. Kandungan

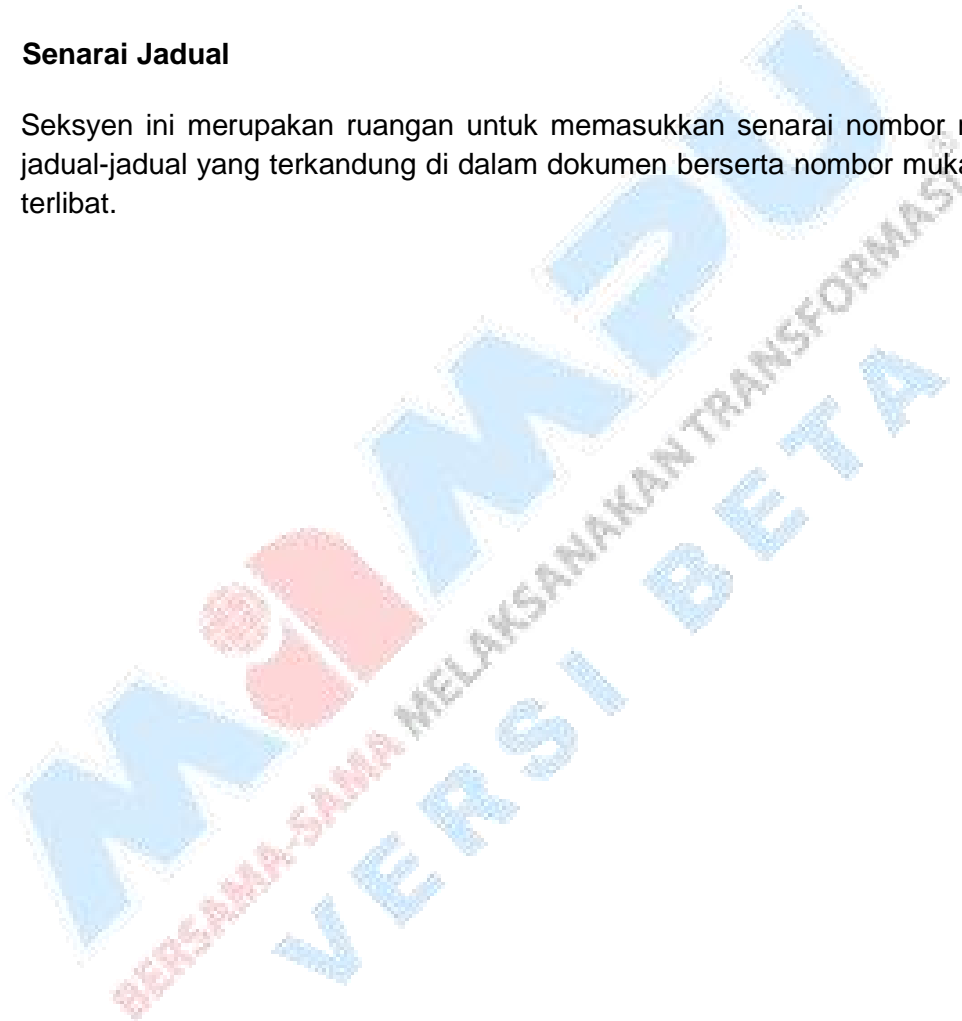
Seksyen ini merupakan ruangan untuk memasukkan maklumat kandungan dokumen berserta nombor muka surat yang terlibat.

iv. Senarai Gambarajah

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi gambarajah-gambarajah yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.

v. Senarai Jadual

Seksyen ini merupakan ruangan untuk memasukkan senarai nombor rujukan bagi jadual-jadual yang terkandung di dalam dokumen berserta nombor muka surat yang terlibat.



1. TUJUAN

Dokumen ini bertujuan melaporkan hasil pelaksanaan Projek Pembangunan Sistem Aplikasi.

2. OVERVIEW PROJEK

Seksyen ini menyediakan sedikit informasi berkaitan komitmen awal projek pembangunan sistem petikan daripada Pelan Pembangunan Sistem ataupun Piagam Projek. Overview sekurang-kurangnya terdiri daripada:

- a) Latar Belakang Projek
- b) Skop Pembangunan Sistem
- c) Serahan
- d) Jadual Pelaksanaan
- e) Keperluan Sumber

3. PENCAPAIAN DAN SERAHAN

Seksyen ini menerangkan pencapaian Projek Pembangunan Sistem Aplikasi mengikut jadual yang telah ditetapkan termasuklah pencapaian aktiviti termasuk bengkel, mesyuarat dan walkthru/taklimat dan senarai serahan yang dihasilkan dan menerangkan berkaitan arkitektur dan fungsi sistem aplikasi yang dibangunkan.

- a) Pencapaian Aktiviti
- b) Senarai Serahan (Deliverables)
- c) Arkitektur dan Fungsi Aplikasi yang dibangunkan

4. PENDEKATAN PEMBANGUNAN

Seksyen ini menerangkan pendekatan pembangunan yang telah dilaksanakan sehingga aplikasi dibangun dan dilaksana dengan jayanya.

5. PENJIMATAN

Seksyen ini menerangkan penjimatan kos dan sumber-sumber lain sepanjang pelaksanaan pembangunan sistem aplikasi.

6. FAEDAH

Seksyen ini menerangkan faedah jangka masa pendek dan jangka masa panjang kepada perkhidmatan penyampaian agensi/kerajaan dengan menggunakan sistem yang dibangunkan.

7. PENERIMAAN SISTEM

Seksyen ini menunjukkan perakuan penerimaan sistem oleh pemilik sistem daripada pasukan projek/pembangun. Contoh pengesahan penerimaan adalah seperti jadual berikut:

TANDATANGAN PENGURUS/PENGARAH PROJEK	TARIKH
TANDATANGAN PEMILIK PROJEK	TARIKH

PENASIHAT

Dr. Azizah Binti Abd. Manan
Kathirrasan K Kupusamy
Noriati Binti Baharum
Noraini Binti Ab Rahim
Hussin Bin. Abu Bakar

PENGHARGAAN

Norazah Binti Ab. Latif
Dr. Mohd Hafeez Bin Osman
Rozaida Binti Mohd Darus
Azizan Bin Torman
Rasyida Binti Zainun

EDITOR

Azih Bin Yusof
Nik Zalbiha Binti Nik Mat
Muhammad Hadri Bin Basri

PENYEMAK

Siti Nurliza Binti Mokhtar
Rohiza Binti Ahmad
Narizan Binti Atan

PENGARANG

	Kandungan	Penulis
1	PENGENALAN KEPADA METODOLOGI KEJURUTERAAN SISTEM APLIKASI	Dr. Azizah Binti Abd. Manan Kathirrasan K Kupusamy Nik Zalbiha Binti Nik Mat Myzatul Akmam Binti Sapaat
2	FASA PERMULAAN PROJEK	
	Penyediaan Pelan Pembangunan Sistem [F1.1]	Nik Zalbiha Binti Nik Mat Abdullah Bin Mohammad
	Penentuan Keperluan Bisnes [F1.2]	Abdullah Bin Mohammad
	Pemodelan Fungsi Bisnes [F1.3]	Abdullah Bin Mohammad
	Pemodelan Proses Bisnes [F1.4]	Muhammad Hadri Bin Basri
	Penyediaan Spesifikasi Keperluan Bisnes [F1.5]	Muhammad Hadri Bin Basri
3	FASA ANALISIS	
	Pemodelan Use Case (Fungsian) [F2.1]	Nik Zalbiha Binti Nik Mat Muhammad Hadri Bin Basri
	Pemodelan Fungsi Sistem [F2.2]	Muhammad Hadri Bin Basri
	Pemodelan Keperluan Data [F2.3]	Rapi'ah Binti Ibrahim Nur Sharmini Alexander
	Pemodelan Proses Sistem [F2.4]	Dr Razatulshima Binti Ghazali
	Penentuan Keperluan Bukan Fungsian [F2.5]	Nik Zalbiha Binti Nik Mat Mohamed Zulkifli Bin Mohamed Nawi
	Penyediaan Spesifikasi Keperluan Sistem [F2.6]	Muhammad Hadri Bin Basri
4	FASA REKABENTUK	
	Rekabentuk Arkitek [F3.1]	Azih Bin Yusof Muhammad Hadri Bin Basri
	Penentuan Teknologi [F3.2]	Muhammad Hadri Bin Basri
	Rekabentuk Pangkalan Data [F3.3]	Rapi'ah Binti Ibrahim Nur Sharmini Alexander
	Rekabentuk Antaramuka Pengguna [F3.4]	Muhammad Hadri Bin Basri

	Kandungan	Penulis
	Rekabentuk Transaksi Sistem [F3.5]	Nik Zalbiha Binti Nik Mat
	Penyediaan Spesifikasi Rekabentuk Sistem [F3.6]	Muhammad Hadri Bin Basri
	Penyediaan Pelan Migrasi Data [F3.7]	Noor Hasliza Binti Mohd Hassan Nur Sharmini Alexander
	Rekabentuk Migrasi Data [F3.8]	Noor Hasliza Binti Mohd Hassan Nur Sharmini Alexander
	Penyediaan Pelan Integrasi Sistem [F3.9]	Mohd Amru Saifullah Bin Alias Nur Sharmini Alexander
	Rekabentuk Integrasi Sistem [F3.10]	Mohd Amru Saifullah Bin Alias Nur Sharmini Alexander
5	FASA PEMBANGUNAN	
	Pembangunan Pangkalan Data [F4.1]	Rapi'ah Binti Ibrahim Nur Sharmini Alexander
	Pengaturcaraan Aplikasi [F4.2]	Rasyida Binti Zainun Asnida Binti Abu Bakar
	Pengujian Sistem [F4.3]	Roshaimieza Binti Mat Adam Myzatul Akmam Binti Sapaat
6	FASA PENGUJIAN PENERIMAAN	
	Penyediaan Pelan Induk Pengujian [F5.1]	Norazah Binti Ab. Latif Myzatul Akmam Binti Sapaat
	Penyediaan Dokumentasi Persediaan Ujian [F5.2]	Asnida Akmal Noor Binti Che Ahmad
	Penyediaan Pelan Ujian (UAT & PAT) [F5.3]	Myzatul Akmam Binti Sapaat
	Ujian Penerimaan Pengguna (UAT) [F5.4]	Sri Lakshmi A/P Kanniah
	Ujian Penerimaan Provisional (PAT) [F5.5]	Sri Lakshmi A/P Kanniah
	Penyediaan Laporan Ujian UAT/PAT [F5.6]	Sri Lakshmi A/P Kanniah
7	FASA PELAKSANAAN	
	Pelaksanaan Migrasi Data [F6.1]	Noor Hasliza Binti Mohd Hassan Nur Sharmini Alexander
	Ujian Penerimaan Akhir [F6.2]	Sri Lakshmi A/P Kanniah
	Penyediaan Manual Pengguna [F6.3]	Malek Riduan Bin Ab. Ghani
	Serahan Sistem Aplikasi [F6.4]	Nik Zalbiha Binti Nik Mat
8	PENGIRAAN SAIZ SISTEM APLIKASI	Azih Bin Yusof Muhammad Hadri Bin Basri

KRISA

PANDUAN KEJURUTERAAN
SISTEM APLIKASI SEKTOR AWAM

UNIT PEMODENAN TADBIRAN & PERANCANGAN PENGURUSAN MALAYSIA
(MAMPU)

JABATAN PERDANA MENTERI

www.mampu.gov.my