



BAB:03

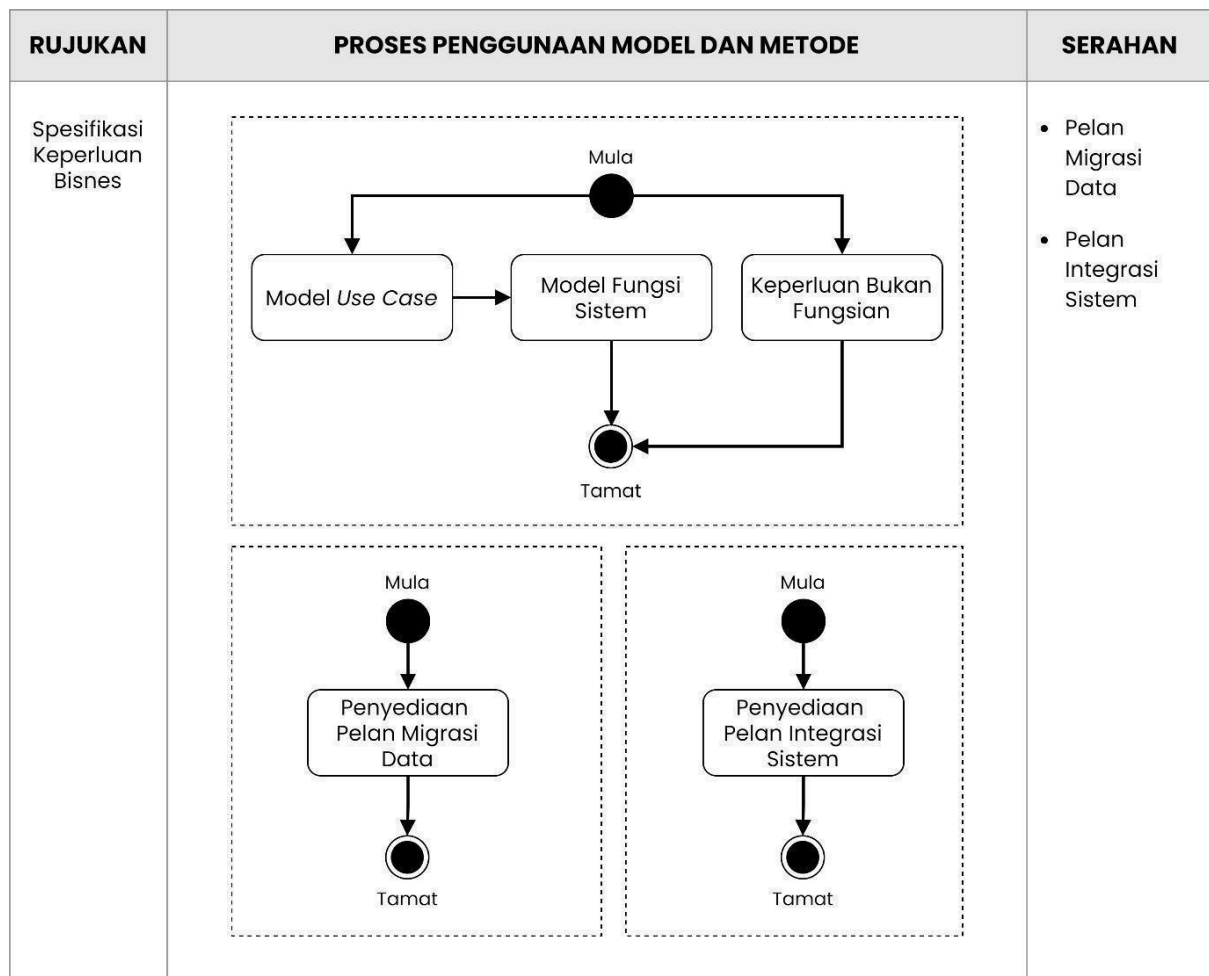
FASA ANALISIS

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 3.1: Gambaran Keseluruhan Fasa 2 – Analisis

3.2 PENGENALAN

Fasa analisis sistem merangkumi aktiviti-aktiviti menganalisis keperluan bisnes yang telah dikenal pasti dalam fasa permulaan projek dan menterjemahkannya kepada keperluan sistem yang jelas dan terperinci sebagai asas rujukan bagi aktiviti-aktiviti pada fasa berikutnya. Keperluan sistem merujuk kepada keterangan terperinci berkenaan aspek fungsian, perkhidmatan dan kekangan operasi bagi sistem aplikasi. Tambahan pula, Keperluan ini bertujuan untuk menetapkan dengan jelas perkara yang perlu dibangunkan dan dilaksanakan dalam sistem aplikasi. Keperluan sistem aplikasi pada kebiasaannya diklasifikasikan kepada dua jenis keperluan, iaitu:

Keperluan Fungsian	Keperluan fungsian menyatakan secara terperinci apakah yang perlu dilakukan oleh satu-satu sistem yang merangkumi bukan sahaja fungsi-fungsi utama tetapi juga fungsi sokongan sistem. Ia juga menerangkan keperluan fungsi yang perlu disediakan oleh sistem bagi menyokong proses dan memenuhi keperluan bisnes yang telah dikenal pasti dalam fasa sebelumnya.
Keperluan Bukan Fungsian	Keperluan bukan fungsian adalah keperluan yang menetapkan kekangan atau had terhadap perkhidmatan dan fungsi sistem aplikasi termasuk kekangan dari segi masa, proses pembangunan, dan pematuhan kepada standard tertentu. Keperluan ini lazimnya merujuk kepada sistem aplikasi secara keseluruhan dan bukannya harus kepada satu-satu fungsi spesifik. Ia juga menentukan kriteria atau ciri-ciri yang digunakan untuk mengukur prestasi, kebolehgunaan, keselamatan, dan kebolehselenggaraan satu-satu sistem yang dibangunkan.

Penghasilan keperluan sistem perlu melalui kajian yang terperinci ke atas maklumat keperluan bisnes yang telah diperolehi. Pelaksanaan kajian terperinci ini melibatkan lapan (8) aktiviti seperti berikut:

- a) Pemodelan Use Case
- b) Pemodelan Fungsi Sistem
- c) Penentuan Keperluan Bukan Fungsian
- d) Pemodelan Keperluan Data (ERD)
- e) Penentuan Teknologi
- f) Penyediaan Spesifikasi Keperluan Sistem (SRS)
- g) Penyediaan Pelan Migrasi Data
- h) Penyediaan Pelan Integrasi Data

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

Dokumen Serahan kepada Fasa Analisis adalah seperti berikut:

- a) D03 Dokumen Spesifikasi Keperluan Sistem (SRS)
- b) D05 Dokumen Pelan Migrasi
- c) D07 Dokumen Pelan Integrasi.

Model dan aktiviti yang berkaitan dengan Pemodelan *Use Case*, Fungsi Sistem, Keperluan Bukan Fungsian dan Pemodelan Keperluan Data (ERD) akan dikumpul dan direkodkan dalam D03 Dokumen Spesifikasi Keperluan Sistem (SRS) .

3.3 PENGLIBATAN PEMEGANG TARUH

Pemegang taruh utama yang perlu terlibat di dalam analisis keperluan sistem adalah 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 secara lebih terperinci adalah seperti berikut:

- a) **Juruanalisis Sistem** yang bertanggungjawab menjalankan analisis keperluan sistem dan menyediakan model serta dokumen berkaitan bagi menyokong fasa reka bentuk dan pembangunan sistem.
- b) **Juruanalisis Bisnes** yang berperanan sebagai rujukan bagi memastikan keperluan bisnes yang telah dikenalpasti dalam fasa sebelumnya dirujuk dan diterjemahkan dengan tepat ke dalam keperluan sistem.
- c) **Subject Matter Expert (SME)** atau **Pemilik Sistem** yang berperanan untuk menyemak dan mengesahkan keperluan fungsian dan bukan fungsian sistem yang telah dikenal pasti agar selaras dengan keperluan organisasi.
- d) **Pengurus Projek** berperanan untuk memastikan penyelarasan, pemantauan kemajuan dan pematuhan kepada skop serta jadual projek.

3.4 FAKTOR KEJAYAAN

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

- a) **D02 Spesifikasi Keperluan Bisnes** adalah lengkap dan perlu mendapatkan pengesahan pemegang-pemegang taruh yang terlibat, khususnya SME atau/dan Pemilik Proses.
- b) Pasukan Juruanalisis Sistem mempunyai kompetensi dan kemahiran yang secukupnya untuk menterjemahkan keperluan bisnes kepada keperluan sistem.
- c) **D05 Pelan Migrasi Data** dan **D07 Pelan Integrasi Sistem** disemak dan mendapat pengesahan pemegang-pemegang taruh yang ditetapkan oleh organisasi.

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

3.5.1 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 fungsi bisnes.

Pemodelan *Use Case* terdiri daripada set berikut:

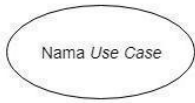
- Koleksi aktor yang berinteraksi dengan sistem;
- Koleksi interaksi antara *Use Case* dalam sistem; dan
- Lain-lain komponen yang menjelaskan gambarajah *Use Case*.

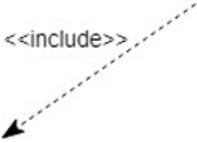
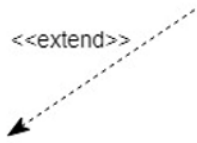
3.5.2 OBJEKTIF

- 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 fungsi bisnes.

Notasi

Jadual 3.1: Notasi Rajah *Use Case*

Elemen	Keterangan
<i>Actor</i>  Nama Aktor	- Digunakan bagi mewakili peranan yang dimainkan oleh entiti luar seperti pengguna atau sistem luaran yang berinteraksi dengan sistem. - Setiap aktor perlu dinyatakan nama peranan.
<i>Use Case</i>  Nama <i>Use Case</i>	- 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.

Elemen	Keterangan
Association 	Digunakan bagi mewakili interaksi aktor dengan <i>Use Case</i>.
Boundary 	Digunakan bagi menetapkan sempadan sistem/modul yang menakrifkan skop sistem/modul yang akan dibangunkan.
Include 	- 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.
Extend 	- 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>
Inheritance / Generalization 	- 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.
Comment 	Digunakan bagi mewakili nota ringkas yang ingin dimasukkan di dalam aliran proses sebagai tambahan nota kepada gambarajah.
Artifak 	Bagi mewakili hubungan kepada gambarajah atau dokumen yang lain. Ia boleh dihubungkan dengan kebergantungan kepada mana-mana elemen lain pada gambarajah.

Elemen	Keterangan
ID Use Case UC - SS - SBS - MM - SBM - 99	- Digunakan sebagai konvensyen nama dan nombor bagi setiap Rajah Use Case berpandukan kepada Model Fungsional Sistem yang dibangunkan. - Keterangan Kandungan ID: - UC = Ringkasan Nama Teknik - SS = Ringkasan Nama Sistem - SBS = Ringkasan Nama Subsystem - MM = Ringkasan Nama Modul - SBM = Ringkasan Nama Submodul *99 = Nombor Rujukan Use Case * hanya diperlukan bagi pelabelan notasi Use Case sahaja.

Langkah-langkah

Langkah 1 : Terjemahkan Peranan (PFD) Kepada Aktor

- Pilih salah satu Rajah Aliran Proses Bisnes (PFD) yang didokumenkan dalam BRS yang dihasilkan dari fasa Pemodelan Proses Bisnes[F1.4].
- 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 Pegawai Tadbir. Contoh penterjemahan peranan kepada aktor adalah seperti jadual berikut:

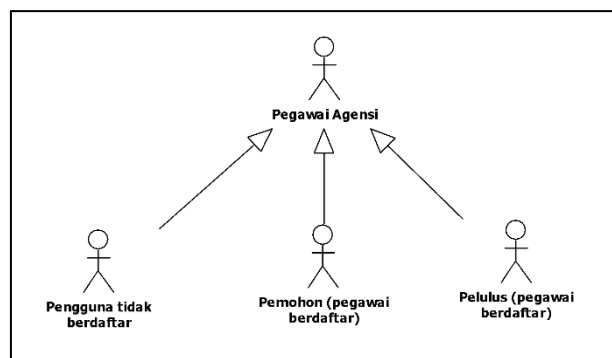
Jadual 3.2 : Contoh Jadual Terjemahan Peranan kepada Aktor

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:
- Aktor terletak di luar sempadan sistem/modul.
 - Aktor boleh memainkan peranan sebagai pelaksana kepada *Use Case* dan juga penerima hasil daripada tindakan *Use Case* dalam berinteraksi dalam system.
- d) Laksanakan pengkhususan Aktor. Multi-Aktor boleh dikumpulkan (*inherit behaviour*) bagi menjelaskan peranan aktor dalam Rajah *Use Case*.

Contoh:

Pegawai agensi boleh berperanan sebagai pengguna biasa (pegawai tidak berdaftar, pemohon (pegawai berdaftar) dan pelulus (pegawai berdaftar).



Rajah 3.2: Contoh Penggunaan *Inherit Behavior* (Aktor)

Langkah 2 : Terjemahkan Aktiviti Kepada *Use Case*

- Analisis aktiviti-aktiviti yang terdapat dalam PFD yang dipilih.
- 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 diambil kira dalam mengenalpasti *Use Case*. Ciri-ciri yang perlu diambil kira dalam melakar *Use Case* adalah:
 - Use Case* perlulah lengkap, bermakna dan memberi hasil kepada pengguna.
 - Elakkan daripada penstrukturan *Use Case* dalam peringkat tinggi kerana ini *Use Case* sukar difahami.
 - Use Case* boleh dipecahkan berdasarkan kepada kompleksiti fungsinya dan juga penggunaan fungsi secara berulang.
 - 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 3.3: Keterangan Label Use Case

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

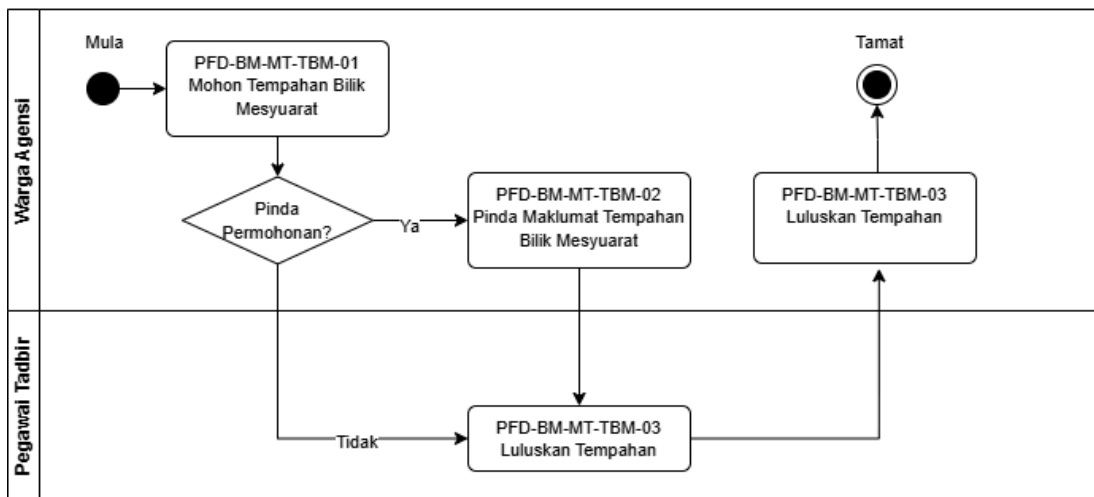
Langkah 3 : Sediakan Hubungan-hubungan Di Antara Aktor Dengan Use Case

- Lukiskan Aktor yang diterjemahkan dalam langkah 1 menggunakan notasi *Actor*.
- Lukiskan *Use Case* yang dikenalpasti dari langkah 2 menggunakan notasi *Use Case* dan labelkan setiap *Use Case*.
- Lakarkan sempadan sistem/modul merangkum semua *Use Case* yang dilukiskan menggunakan notasi *Boundary*. Manakala *Actor* perlu berada di luar sempadan.
- Lakarkan hubungan antara *Actor* dan *Use Case*. Hubungan ini menunjukkan interaksi aktor kepada *Use Case*. Dalam hubungan ini notasi *association* sahaja digunakan.
- Sediakan keterangan setiap *use case* yang dilakarkan.

Contoh:

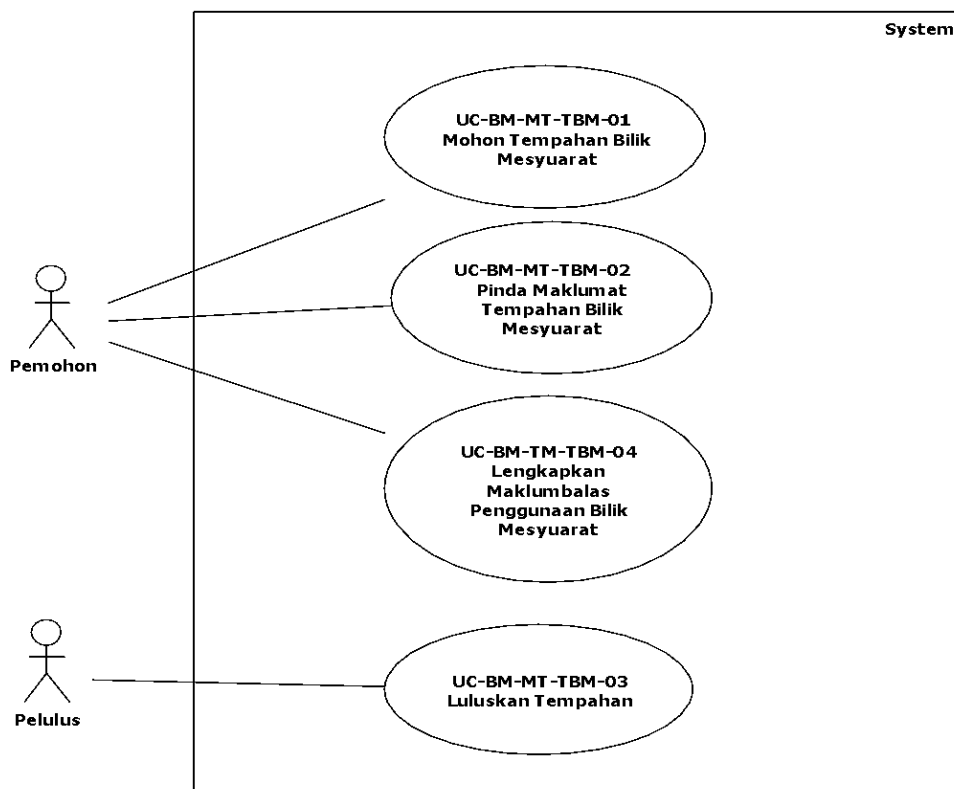
Menterjemahkan Proses Tempahan Bilik Mesyuarat (PFD-BM-MT-TBM) kepada **Rajah Use Case UC-BM-MT-TBM**. Berikut adalah Rajah Aliran Proses yang telah dibangunkan dalam **Pemodelan Proses Bisnes [F1.4]** bagi proses PFD-BM-MT-TBM Tempah Bilik Mesyuarat.

PFD-BM-MT-TBM Tempah Bilik Mesyuarat



Rajah 3.3 : Rajah Aliran Kerja PFD-BM-MT-TBM Tempah Bilik Mesyuarat

Rajah Use Case - Modul Tempahan Bilik Mesyuarat



Rajah 3.4 : Contoh Rajah Use Case - Hubungan Aktor Dengan Use Case

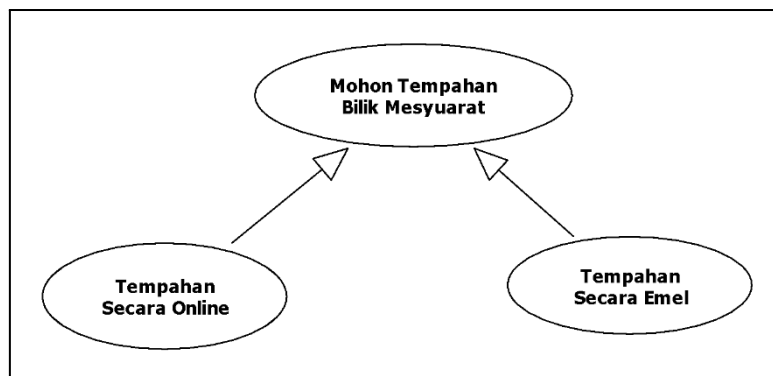
Jadual 3.4 : Contoh Keterangan Use Case

Label	Nama Use Case	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 : Perincikan 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 dipecahkan 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 melaksanakan pengkhususan kepada Use Case utama. Hubungan ini hanya untuk mengumpulkan Use Case yang berkaitan.

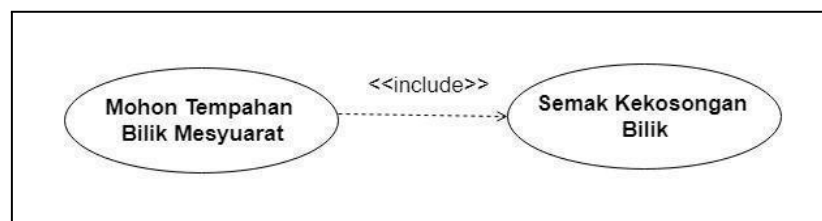
Contoh: Tempahan Bilik Mesyuarat boleh dilakukan dengan dua cara, iaitu secara online yang disediakan atau tempahan boleh dilakukan secara emel kepada Pegawai Tadbir tempahan.



Rajah 3.5 : 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.

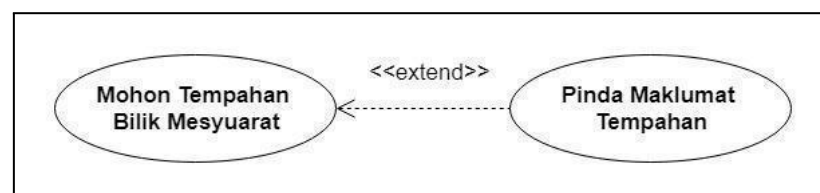
Contoh: Use Case Mohon Tempahan Bilik Mesyuarat perlu memanggil fungsi Semak kekosongan Bilik bagi melengkapkan proses.



Rajah 3.6 : Contoh Hubungan Use Case (Include)

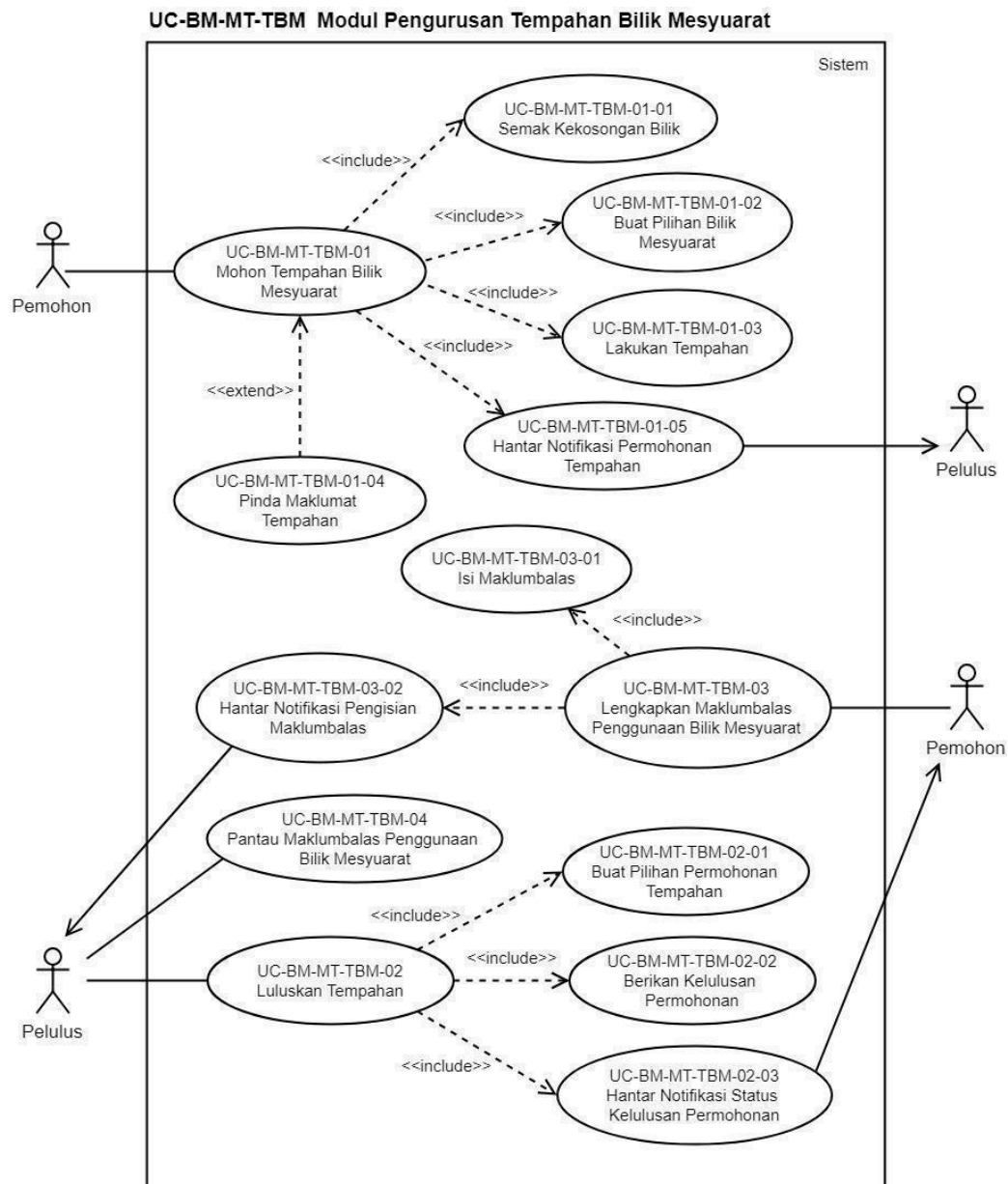
- iii) <Extend> – Use Case yang memberi nilai tambah kepada Use Case yang lain. Ianya digunakan bagi mentakrifkan sesuatu Use Case menggunakan fungsi dalam Use Case yang lain dengan syarat-syarat tertentu.

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



Rajah 3.7 : 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 3.8 : Contoh Rajah *Use Case* Modul Pengurusan Tempahan Bilik Mesyuarat

Langkah 5 : Sediakan Senario *Use Case* (*Use Case Scenario*)

- Sediakan penerangan ringkas *Use Case*.
- Tentukan langkah-langkah aktiviti *Use Case*
- Kenal pasti Pasca Syarat *Use Case*
- Nyatakan Proses Alternatif *Use Case*
- Sedia dan lengkapkan Templat Senario *Use Case*

f) Dokumentenkan Senario *Use Case*

Langkah 6 : Dapatkan Pengesahan SME

- a) Semak Rajah *Use Case* yang dihasilkan dengan SME. Semakan perlulah mengambil kira aspek-aspek berikut;
- i) Lengkap – mengambil kira 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:

Pasukan Pembangunan	Fahami fungsi sistem yang akan dibangunkan.
SME	Fahami kaitan proses bisnes dengan fungsi yang dibangunkan.

Langkah 7 : Dokumentenkan Rajah *Use Case*

Dokumenkan hasil Pemodelan *Use Case* (Rajah *Use Case*, keterangan *Use Case* dan Senario *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]

3.6.1 KETERANGAN

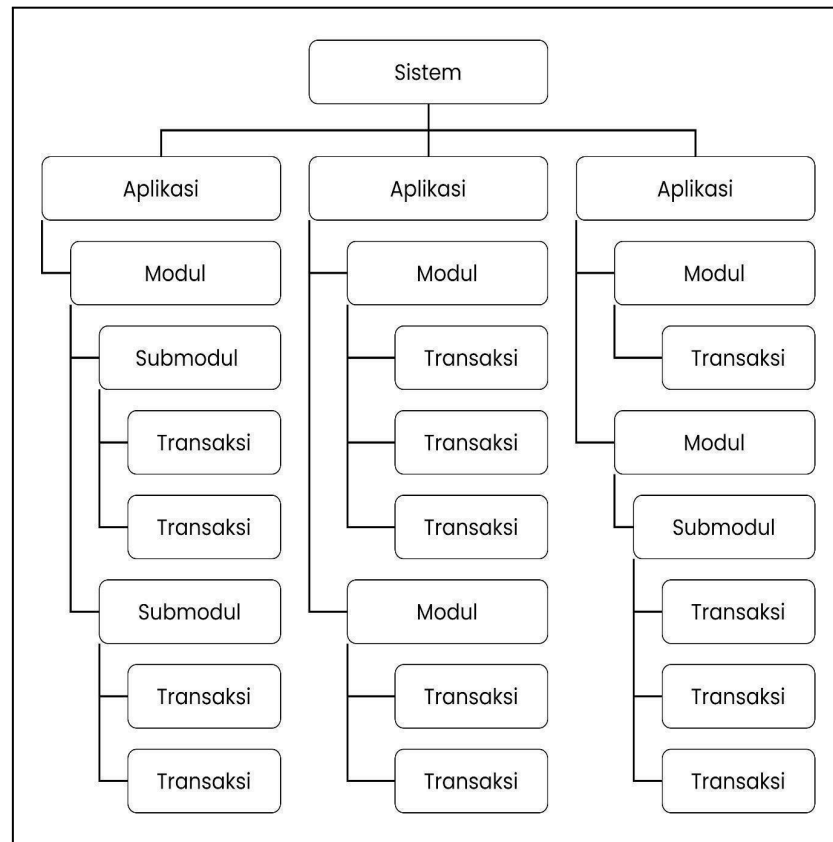
Pemodelan Fungsi Sistem adalah teknik untuk mengenal pasti dan menstruktur fungsi-fungsi di dalam sistem yang ingin dibangunkan. Teknik ini adalah berpandukan kepada rajah-rajah *use case* yang telah disediakan dalam **Pemodelan Use Case [F2.1]** dan rajah hierarki bisnes di dalam **Pemodelan Fungsi Bisnes [F1.3]**. Berbeza dengan rajah hierarki fungsi bisnes, rajah hierarki fungsi sistem ini disediakan bertujuan untuk mereka bentuk struktur fungsi ataupun lebih mudah untuk difahami sebagai 'menu' di dalam sistem yang akan dibangunkan.

Hasil pemodelan ini dipaparkan dalam bentuk rajah hierarki fungsi sistem. Fungsi-fungsi sistem di dalam rajah ini secara amnya boleh terdiri daripada lima (5) tahap, iaitu:

Jadual 3.5: Tahap-tahap Fungsi Sistem

Tahap	Penerangan
Sistem	Merupakan fungsi sistem utama yang ingin dibangunkan. Perlu diingati bahawa fungsi sistem pada tahap ini hanya boleh terdiri dari satu komponen sahaja bagi satu-satu sistem aplikasi.
Aplikasi (Subsistem)	Pecahan fungsi sistem yang merangkumi pelbagai jenis aplikasi bagi menyokong kepada operasi keseluruhan satu-satu sistem. Contoh fungsi pada tahap ini adalah seperti aplikasi web, aplikasi mudah alih, portal dan mana-mana komponen lain yang berkaitan.
Modul	Pecahan fungsi aplikasi yang merangkumi proses-proses tertentu dalam satu-satu aplikasi. Contoh fungsi pada tahap modul adalah seperti Modul Pengurusan Pengguna dan Modul Pengurusan Tempahan.
Submodul	Pecahan fungsi modul yang merangkumi proses-proses tertentu dalam satu-satu modul. Contoh fungsi pada tahap submodul adalah seperti Submodul Profil Pengguna, Submodul Tempahan dan Submodul Maklum Balas Tempahan.
Transaksi	Aktiviti atau proses terkecil dalam modul atau submodul yang terdiri daripada aktiviti-aktiviti untuk melaksanakan satu proses/subproses atau tugas tertentu dan spesifik. Contoh fungsi pada tahap transaksi adalah seperti Daftar Akaun Pengguna, Mohon Tempahan, Lulus Tempahan dan Mengisi Maklumbalas.

Penyediaan bagi tahap-tahap fungsi sistem secara amnya boleh distrukturkan seperti rajah berikut:



Rajah 3.9: Variasi Penstrukturan Tahap-tahap Fungsi Sistem

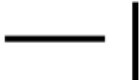
Penentuan bilangan tahap fungsi sistem adalah bergantung kepada saiz dan kompleksiti sistem aplikasi yang ingin dibangunkan. Struktur dan penentuan tahap fungsi sistem perlu direka bentuk secara iteratif dan disesuaikan dengan objektif skop bisnes. Pendekatan ini adalah bagi memastikan sistem aplikasi mampu memenuhi keperluan bisnes secara holistik dan pada masa yang sama juga berkebolehan untuk mengadaptasikan strukturnya kepada sebarang perubahan di masa hadapan.

3.6.2 Objektif

Menyediakan rajah hierarki fungsi sistem yang memaparkan struktur fungsi sistem yang ingin dibangunkan.

Notasi

Jadual 3.6 : Notasi Fungsi Bisnes

Elemen	Keterangan														
Fungsi Sistem 	- 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. - Bagi mana-mana modul/submodule/fungsi transaksi yang menggunakan Emerging Technology seperti AI, Blockchain dan sebagainya – Warnakan Notasi Fungsi Sistem dengan warna yang berbeza dengan fungsi sistem yang biasa.														
Penghubung 	Penghubung di antara notasi Fungsi-fungsi Sistem menggunakan garis lurus mendatar atau menegak.														
ID Fungsi Sistem SF - SS - SBS - MM - SBM - 99 - 88	- Digunakan sebagai konvensyen nama dan nombor bagi setiap Rajah Hierarki Fungsian Sistem yang dibangunkan. - Keterangan Kandungan ID: <table border="1" data-bbox="574 1178 1329 1467"> <tr> <td>SF</td><td>Ringkasan Nama Teknik</td></tr> <tr> <td>SS</td><td>Ringkasan Nama Sistem</td></tr> <tr> <td>*SBS</td><td>Ringkasan Nama Subsistem</td></tr> <tr> <td>*MM</td><td>Ringkasan Nama Modul</td></tr> <tr> <td>*SBM</td><td>Ringkasan Nama Submodul</td></tr> <tr> <td>*99</td><td>Nombor Rujukan Transaksi</td></tr> <tr> <td>*88</td><td>Nombor Rujukan Subtransaksi</td></tr> </table> * hanya diperlukan sekiranya tahap hierarki Fungsi Sistem adalah merupakan subsistem, modul, submodul, transaksi atau subtransaksi.	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
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														

Langkah-langkah

Langkah 1 : Rujuk Rajah Hierarki Fungsi Bisnes dan *Use Case*

- a) Penyediaan rajah hierarki fungsi sistem ini boleh merujuk kepada rajah hierarki fungsi bisnes yang telah dibangunkan dalam **Pemodelan Fungsi Bisnes [F1.3]**. Ini kerana terdapat persamaan yang ketara antara kedua-dua rajah hierarki tersebut meskipun susun atur fungsinya adalah berlainan. Merujuk kepada rajah *use case* yang telah dibangunkan dalam **Pemodelan Use Case [F2.1]**, perbezaan yang berkemungkinan akan berlaku (tertakluk kepada keperluan bagi satu-satu projek) kepada rajah hierarki fungsi sistem berbanding dengan rajah hierarki sebelumnya adalah disebabkan oleh:
- i) Fungsi/subfungsi digabungkan kerana mempunyai proses bisnes dan pengguna yang sama. Contohnya seperti fungsi permohonan cuti rehat, cuti separuh gaji dan cuti tanpa gaji boleh dikumpulkan di bawah fungsi permohonan cuti.
 - ii) Fungsi/subfungsi diasing bagi mengurangkan kompleksiti dalam menavigasi sistem dan menjadikannya lebih intuitif kepada pengguna, contohnya seperti fungsi pembelian makanan (seperti aplikasi Grab dan Foodpanda) diasingkan kepada fungsi pembelian makanan restoran dan makanan pasar raya walaupun prosesnya adalah sama.
 - iii) Kedudukan fungsi/subfungsi bertukar atau berpindah ke lokasi yang lain bertujuan untuk mendapatkan reka bentuk sistem yang lebih teratur, konsisten dan berkesan. Perkara ini boleh berlaku dengan mengumpulkan satu-satu fungsi/subfungsi di bawah fungsi yang lain, ataupun dengan meletakkanya di bawah satu fungsi yang baharu dalam rajah hierarki sistem.
 - iv) Fungsi/Subfungsi diperincikan lagi hingga ke tahap transaksi (bersamaan dengan tahap aktiviti dalam **Pemodelan Proses Bisnes [F1.4]**).
 - v) Pertambahan fungsi/subfungsi sistem yang tidak berkaitan secara terus dengan fungsi bisnes, contohnya seperti mewujudkan fungsi Aplikasi Web dan Aplikasi Portal, ataupun Modul Profil Pengguna dan Modul Pentadbiran Sistem.

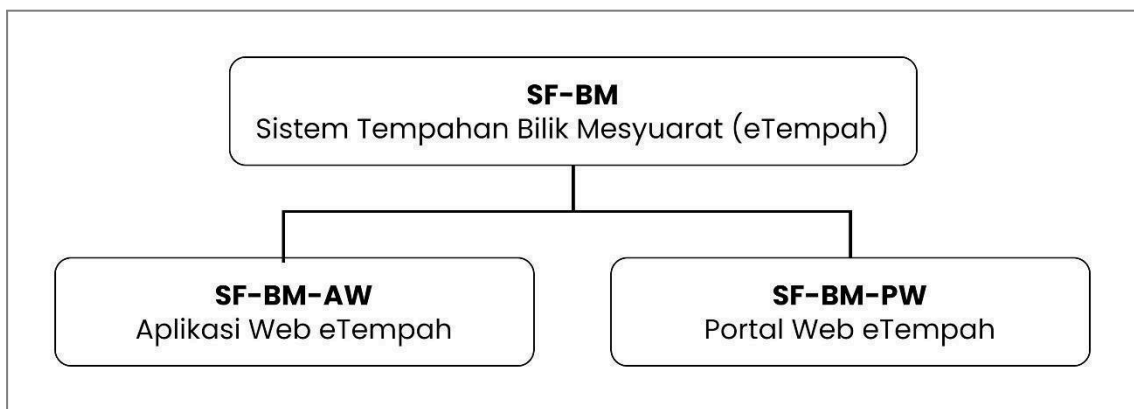
Langkah 2 : Sediakan Fungsi pada Tahap Sistem dan Aplikasi

- a) Penyediaan rajah hierarki dimulakan dengan menyertakan komponen utama iaitu fungsi pada tahap sistem. Penyediaan bagi setiap fungsi sistem adalah dengan menggunakan notasi seperti yang telah dinyatakan dalam **Jadual 3.6** serta ID dan nama fungsi sistem yang sesuai. Contoh fungsi pada tahap sistem adalah seperti berikut:



Rajah 3.10: Contoh Fungsi pada Tahap Sistem bagi Sistem eTempah

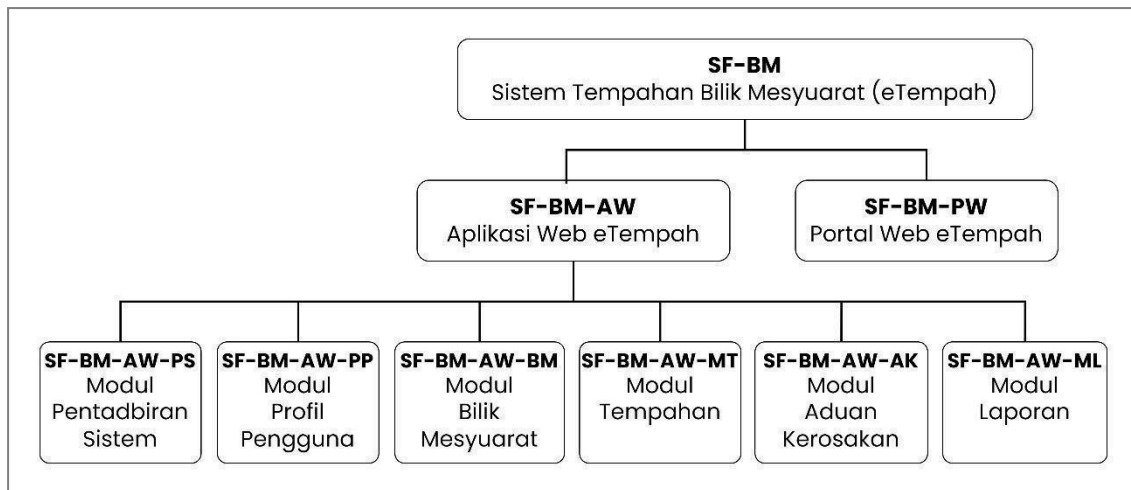
- b) Setelah komponen untuk nama sistem dimasukkan, komponen seterusnya yang akan disediakan adalah fungsi-fungsi yang berada pada tahap aplikasi. Penyediaan fungsi pada tahap aplikasi ini tertakluk kepada projek-projek yang mengandungi pelbagai aplikasi di bawahnya. Sekiranya satu-satu projek pembangunan sistem yang dilaksanakan hanya mengandungi satu aplikasi tunggal sahaja, fungsi pada tahap aplikasi ini boleh digabungkan bersekali dengan fungsi pada tahap sistem. Fungsi-fungsi pada tahap aplikasi perlu dihubungkan dengan fungsi sebelumnya melalui notasi 'penghubung'. Berikut adalah contoh bagi fungsi pada tahap aplikasi bagi sistem eTempah berserta notasi-notasi yang terlibat.



Rajah 3.11: Contoh Fungsi pada Tahap Aplikasi bagi Sistem eTempah

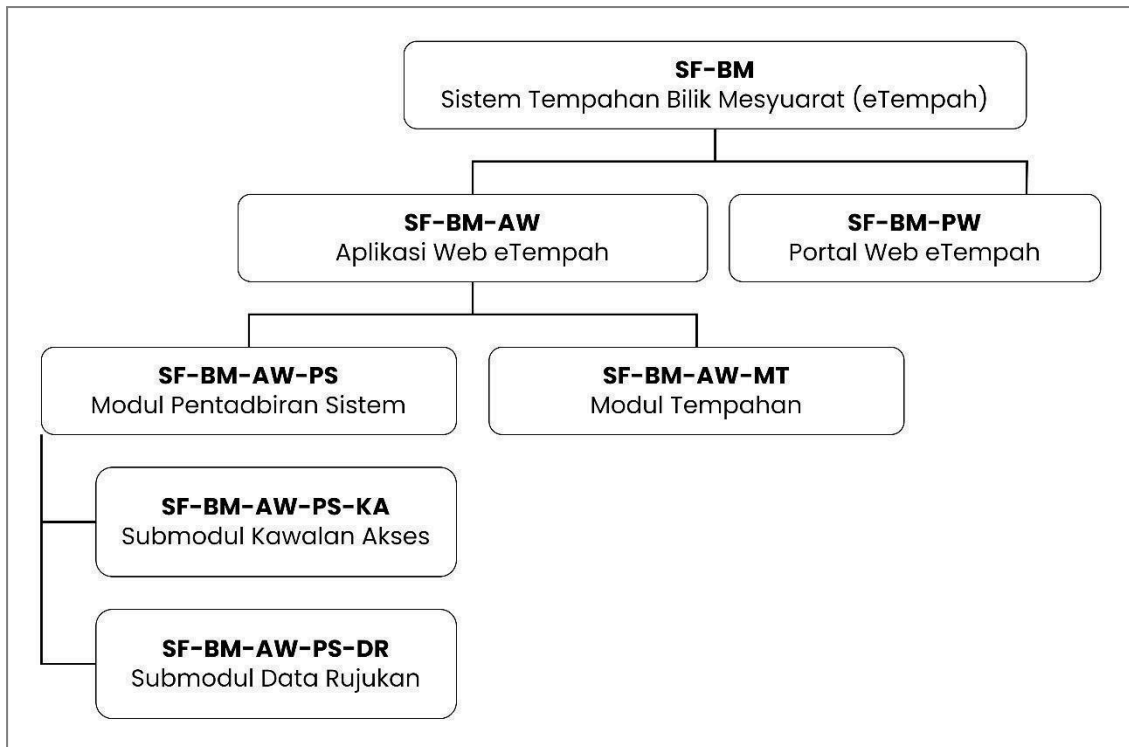
Langkah 3 : Sertakan Fungsi pada Tahap Modul dan Submodul

- a) Penyediaan fungsi sistem pada tahap modul adalah dengan merangkumkan proses-proses yang berkaitan di antara satu sama lain di bawah satu kumpulan. Penentuan fungsi pada tahap modul boleh juga dirujuk kepada rajah hierarki fungsi bisnes yang telah dibangunkan sebelum ini. Contoh fungsi pada tahap modul bagi sistem (aplikasi web) eTempah berserta notasi-notasi yang terlibat adalah seperti Rajah 3.12 di bawah:



Rajah 3.12: Contoh Fungsi pada Tahap Modul bagi Sistem (Aplikasi Web) eTempah

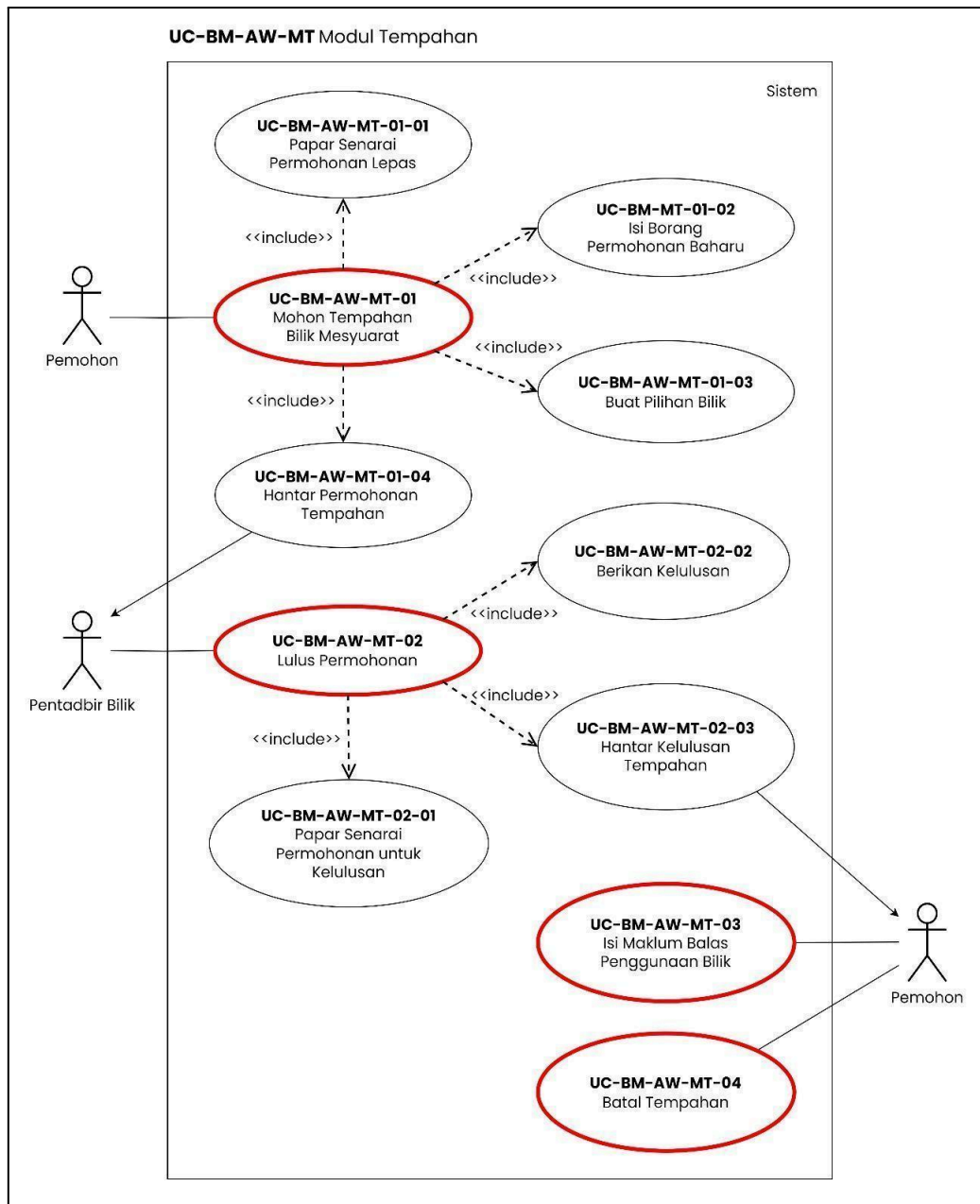
- b) Keperluan untuk menyediakan fungsi sistem pada tahap submodul adalah bergantung kepada saiz dan kompleksiti sistem. Sekiranya saiz sistem atau aplikasi yang ingin dibangunkan meliputi skop fungsi yang luas serta proses yang pelbagai, penggunaan fungsi submodul adalah lebih sesuai untuk dipertimbangkan. Contoh yang sesuai yang boleh diperhatikan dalam sektor awam adalah *Human Resource Management Information System (HRMIS)* yang mengandungi kedua-dua lapisan fungsi bertahap modul dan submodul kerana saiznya yang besar dan berkompleksiti tinggi. Sebaliknya, sekiranya saiz bagi satu-satu sistem atau aplikasi adalah kecil dan sederhana, fungsi sistem pada tahap submodul berkemungkinan tidak begitu sesuai untuk digunakan (tertakluk kepada keperluan projek). Contoh fungsi pada tahap modul bagi sistem (aplikasi web) eTempah, bagi Modul Pentadbiran Sistem dan Modul Tempahan (tiada submodul), beserta notasi-notasi yang terlibat adalah seperti berikut:



Rajah 3.13: Contoh Fungsi pada Tahap Submodul bagi Modul Mengurus Pengguna Sistem (Aplikasi Web) eTempah

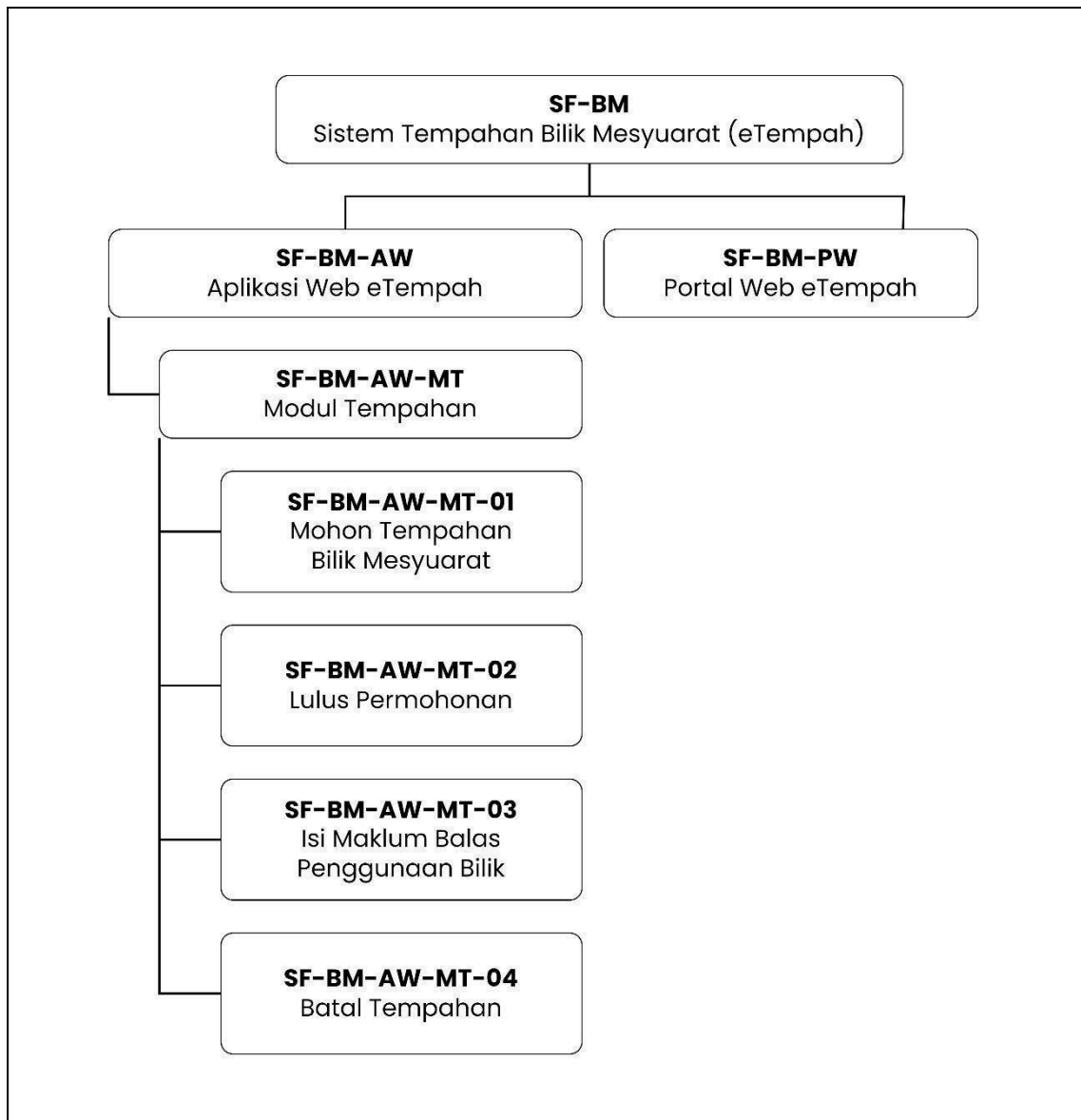
Langkah 4 : Sediakan Fungsi pada Tahap Transaksi

- a) Fungsi sistem pada tahap transaksi adalah merupakan fungsi asas atau terkecil yang akan disertakan dalam rajah hierarki fungsi sistem. Untuk menyediakan fungsi sistem pada tahap transaksi, kenal pasti setiap *use case* utama yang berinteraksi secara langsung dan terus dengan aktor-aktor yang terlibat. Sebarang *use case* yang dihubungkan melalui notasi *association* <<include>> dan <<extend>> adalah tidak boleh dianggap sebagai *use case* utama atau fungsi sistem pada tahap transaksi, dan tidak perlu disertakan di dalam rajah hierarki ini.
- b) Berdasarkan kepada contoh seperti di Rajah 3.14 di bawah, terdapat empat (4) *use case* utama di bawah Modul Tempahan (*use case* yang bertanda merah) yang telah dikenalpasti untuk dijadikan sebagai fungsi sistem pada tahap transaksi, iaitu:
 - i) **UC-BM-AW-MT-01** Mohon Tempahan Bilik Mesyuarat
 - ii) **UC-BM-AW-MT-02** Lulus Permohonan
 - iii) **UC-BM-AW-MT-03** Isi Maklum Balas Penggunaan Bilik
 - iv) **UC-BM-AW-MT-04** Batal Tempahan



Rajah 3.14: Contoh Rajah Use Case bagi Modul Tempahan Sistem (Aplikasi Web) eTempah

- c) Lukiskan fungsi sistem pada tahap transaksi mengikut *use case* utama yang telah dikenal pasti. Berikut adalah contoh fungsi sistem pada tahap transaksi di bawah Modul Tempahan bagi Sistem (Aplikasi Web) eTempah:



Rajah 3.15: Contoh Fungsi pada Tahap Transaksi bagi Modul Tempahan Sistem (Aplikasi Web) eTempah

- d) Lukiskan juga fungsi sistem pada tahap transaksi bagi modul dan/atau submodul lain dalam rajah hierarki ini. Sekiranya fungsi/subfungsi terlalu banyak untuk disatukan dalam satu rajah, pecahkan rajah-rajah hierarki fungsi sistem mengikut modul dan submodul sistem secara bersesuaian.

Langkah 5 : Sediakan Rajah Hierarki Fungsi Sistem secara Iteratif

Penyediaan rajah hierarki 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 ataupun perubahan semasa reka bentuk sistem.

Langkah 6 : Dokumentasikan Model Fungsi Sistem

Dokumen dan masukkan hasil Pemodelan Fungsi Sistem, yaitu rajah hierarki fungsi sistem, yang telah dibangun ke dalam D03- Spesifikasi Keperluan Sistem (SRS).

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]

3.7.1 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 ERD 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.

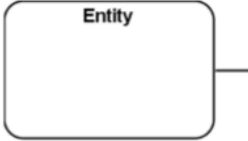
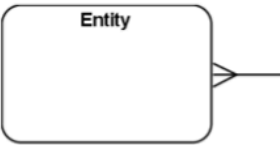
3.7.2 OBJEKTIF

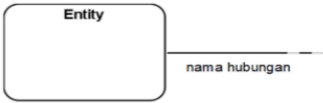
- o Menyediakan model maklumat yang tepat berdasarkan keperluan di sesebuah organisasi. Model maklumat ini boleh dijadikan rangka kerja/panduan untuk menaiktaraf aplikasi sedia ada, atau pembangunan sistem aplikasi yang baharu di organisasi berkenaan.
- o Menyediakan model yang tidak bersandar dengan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

Notasi

Pemodelan Maklumat (ERD) Konseptual adalah berdasarkan notasi Richard Barker:

Jadual 3.7: Notasi Rajah Hubungan Data

Elemen	Keterangan
Entiti 	- 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>).
Kardinaliti : Tahap hubungan 'satu dan hanya satu' 	Kardinaliti 'berkaki satu' menunjukkan tahap hubungan (<i>degree of relationship</i>) 'satu dan hanya satu' dengan entiti lain.
Kardinaliti : Tahap hubungan 'satu atau lebih' 	Kardinaliti 'berkaki tiga' (<i>crowsfoot</i>) menunjukkan tahap hubungan (<i>degree of relationship</i>) 'satu atau lebih' dengan entiti lain.
Modaliti : hubungan mandatori 	Modaliti mandatori menunjukkan sifat kebergantungan (<i>dependence</i>) bagi hubungan antara dua entiti.
Modaliti : hubungan pilihan 	Modaliti pilihan menunjukkan sifat tiada kebergantungan (<i>independence</i>) bagi hubungan antara dua entiti.

Elemen	Keterangan
Nama hubungan 	Nama hubungan yang menunjukkan hubung kait antara dua entiti.
Atribut 	- Atribut adalah ciri-ciri yang menggambarkan sesebuah entiti. Contoh atribut adalah id, nama, keterangan, tarikh dan lain-lain. - Jenis-jenis atribut adalah seperti berikut: - Pengenal Unik/ <i>Unique Identifier</i> (UID) Primer diwakili dengan notasi # - Pengenal Unik (UID) Sekunder diwakili dengan notasi U - 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 °

Langkah-langkah

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

- Kenal pasti dan senaraikan calon entiti dan atribut berdasarkan fungsi hierarki bisnes dan proses pemodelan fungsi bisnes (rujuk Apendiks 2 Definisi Aktiviti Fungsi Bisnes). Aktiviti ini perlu dilaksanakan secara komprehensif, terperinci dan iteratif dengan mengambil kira proses keseluruhan termasuk kekerapan data diakses.
- Berdasarkan contoh cabutan sebahagian daripada kes bisnes dalam proses pemodelan fungsi bisnes (rujuk Apendiks 2 Definisi Aktiviti Fungsi Bisnes), perkataan yang bergaris merupakan perkataan sebagai calon entiti dan atribut yang boleh digunakan dalam pemodelan maklumat. Maklumat yang dikenal pasti diterjemahkan kepada elemen pemodelan maklumat.

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.”

- c) Pola capaian merujuk kepada bagaimana operasi CRUD dilaksanakan keatas data.
- d) Perkara 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 3.8: 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	<i>Create (C):</i> i) Pengguna perlu diwujudkan terlebih dahulu sebelum boleh membuat sebarang tempahan aset ii) Pengguna mesti didaftarkan bawah satu bahagian
Peraturan Bisnes	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

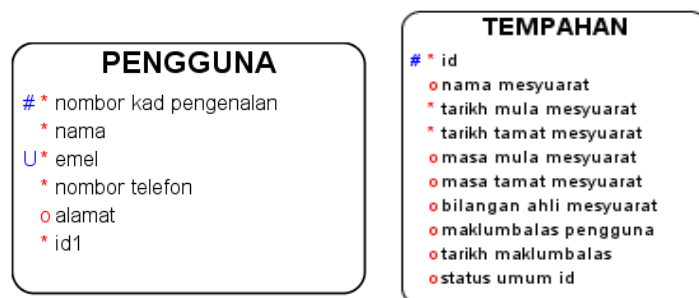
- a) Lukiskan setiap entiti yang telah dikenalpasti. Rujuk notasi seperti di **Jadual 1** untuk membina model maklumat. Contoh entiti adalah seperti rajah berikut.



Rajah 3.16 : 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).



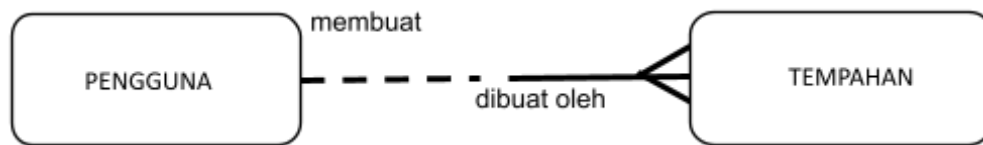
Rajah 3.17 : Contoh Entiti Dan Senarai Atribut

- vi) Dalam satu entiti, satu atau lebih atribut boleh dijadikan Pengenal Unik 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, kenal pasti 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 3.18 : Contoh Nama Hubungan Entiti

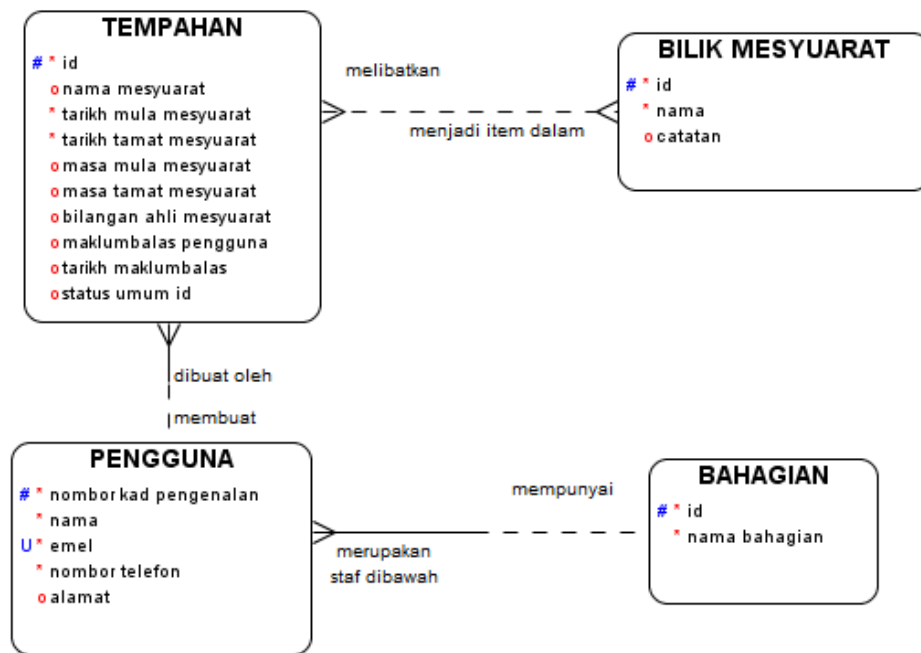
- c) Peraturan bagi membaca hubungan antara entiti adalah seperti berikut:

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:

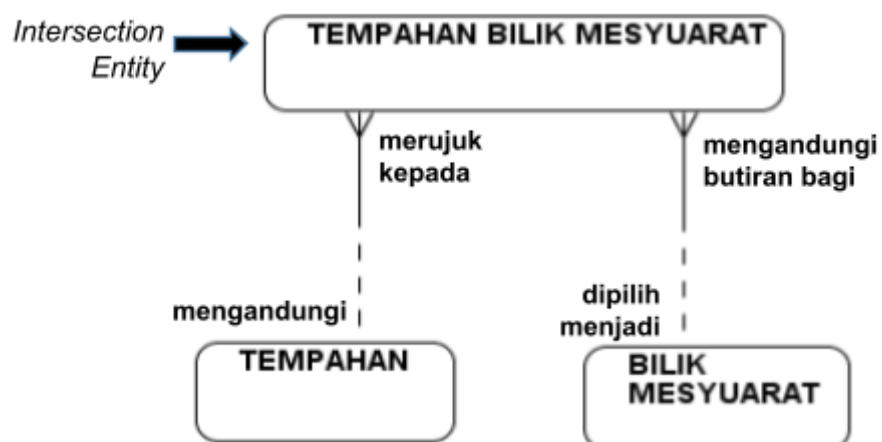
- i) Kardinaliti hubungan menunjukkan setiap PENGGUNA membuat satu atau lebih TEMPAHAN.
 - ii) Dari segi modaliti, hubungan adalah pilihan. Ini bermaksud rekod pengguna boleh wujud tanpa membuat sebarang tempahan bilik mesyuarat, dan beberapa tempahan bilik mesyuarat boleh dibuat oleh pengguna.
 - iii) Bacaan adalah 'setiap PENGGUNA mungkin membuat satu atau lebih TEMPAHAN'.
- d) 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 3.19 : Contoh Rajah Hubungan Entiti (ERD)

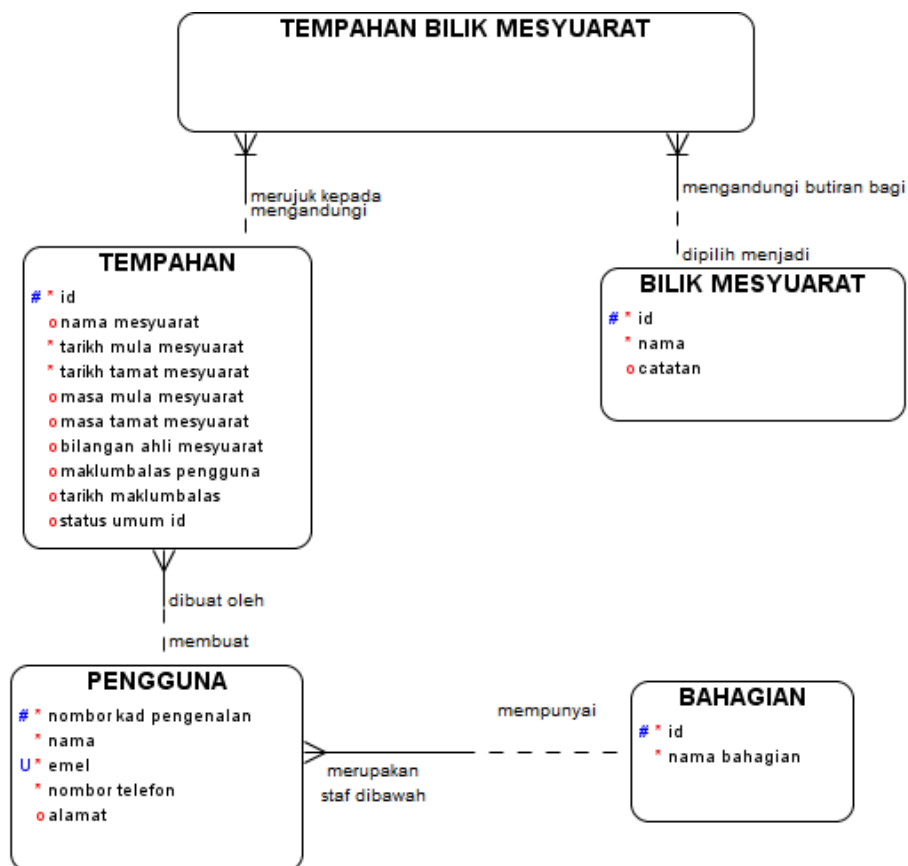
Langkah 4: Perkemaskan Model Maklumat

- 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) bagi memastikan semua maklumat yang diperlukan tidak tercicir. Lengkapkan ERD tersebut dengan menambah entiti-entiti yang baru (jika ada).
- Wujudkan *Intersection Entity* bagi menyelesaikan hubungan banyak-ke-banyak (*many-to-many*) di antara dua entiti sekiranya perlu. Contoh adalah seperti rajah dibawah.



Rajah 3.20 : 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 3.21: ERD yang telah diperkemas

Langkah 5: Dokumentasikan Model Maklumat Konseptual

Dokumentasikan semua output yang dihasilkan sebagai hasil serahan bagi keperluan maklumat ke dalam **D03 Spesifikasi Keperluan Sistem**.

3.8 PENENTUAN KEPERLUAN BUKAN FUNGSIAN [F2.4]

3.8.1 KETERANGAN

Keperluan bukan fungsian, adalah keperluan yang tidak berkaitan secara langsung dengan servis atau fungsi yang tawarkan 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.

3.8.2 OBJEKTIF

- i. Memahami kepentingan keperluan bukan fungsian.
- ii. 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 Templat Definisi Aktiviti 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 berkemungkinan 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 3.22 : Klasifikasi Keperluan Bukan Fungsian

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

Jadual 3.9 : 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 3.10 : 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 3.11 : Keterangan Medan-medan Templat Ciri-ciri Sistem Aplikasi

Medan	Keterangan						
ID	ID merupakan pengenalan kepada setiap ciri-ciri kualiti atau keperluan bukan fungsian yang dikenalpasti. Contoh format ID adalah seperti NF-XX-99, di mana: <table border="1"> <tr> <td>NF</td><td>Bukan Fungsian</td></tr> <tr> <td>XX</td><td>Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran</td></tr> <tr> <td>99</td><td>Siri nombor</td></tr> </table>	NF	Bukan Fungsian	XX	Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran	99	Siri nombor
NF	Bukan Fungsian						
XX	Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran						
99	Siri nombor						
Ciri-ciri Kualiti	Ciri-ciri kualiti perlu dinyatakan mengikut kategori bukan fungsian masing-masing. 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	Catatkan keterangan ciri-ciri kualiti atau keperluan pengguna yang dipetik dari Definisi Fungsi Bisnes.						

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

Jadual 3.12 : Senarai Definisi Ciri-ciri Kualiti Sistem Aplikasi

Ciri	Sub Ciri	Definisi
Functional suitability	i) Functional completeness ii) Functional correctness iii) Functional appropriateness	Sejauh mana sistem menyediakan fungsi yang memenuhi kehendak yang nyata dan tersirat apabila digunakan dalam keadaan tertentu. i) Sejauh mana set fungsi meliputi semua tugas-tugas yang ditetapkan dan objektif pengguna ii) Sejauh mana sistem menyediakan keputusan yang betul dan tepat iii) Sejauh mana fungsi memudahkan pencapaian tugas dan objektif yang dinyatakan
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>i) Confidentiality</i> <i>ii) Integrity</i> <i>iii) Non-repudiation</i> <i>iv) Accountability</i> <i>v) Authenticity</i>	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>i) Modularity</i> <i>ii) Reusability</i> <i>iii) Analysability</i> <i>iv) Modifiability</i> <i>v) Testability</i>	Tahap keberkesanan dan kecekapan penyelenggara sistem yang dipertanggungjawab.
Portability	<i>i) Adaptability</i> <i>ii) Installability</i> <i>iii) Replaceability</i>	Tahap keberkesanan dan kecekapan dengan mana sistem, produk atau komponen boleh dipindahkan dari satu perkakasan, perisian atau persekitaran operasi yang lain.

Langkah 4 : Dokumentkan Keperluan Bukan Fungsian

Dokumentkan hasil analisis bukan fungsian ke dalam D03 Spesifikasi Keperluan Sistem (SRS) di bawah seksyen keperluan bukan fungsian

Rujukan

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

3.9 PENENTUAN TEKNOLOGI [F2.5]

3.9.1 KETERANGAN

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 dan kaji :

- 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 berbeza. Matriks alternatif boleh disediakan seperti dalam contoh di bawah. Tahap/Pemberat dan skor-skor dirangkumi bersekali bagi mewujudkan kad skor yang mengenal pasti 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 3.13 : Nilai Skor Skema Penilaian Teknikal

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

Jadual 3.14 : Matriks Alternatif Bagi Penentuan Bahasa Pengaturcaraan

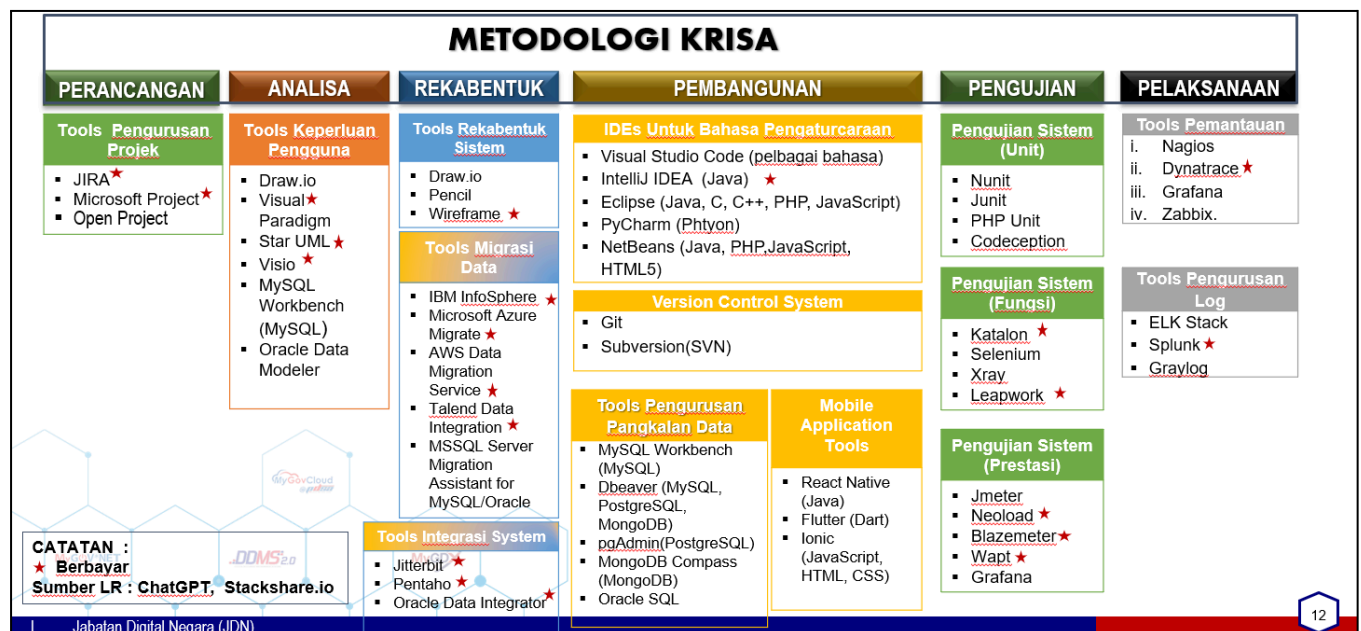
No.	Kriteria	Bahasa Pengaturcaraan		
		Java	C#	Phyton
		Nilai Skor		
1.	Pengetahuan, pengalaman dan kemahiran pasukan pembangun	8	6	1
2.	Ketersediaan (<i>availability</i>) pengaturcara	9	5	1
3.	Ketersediaan <i>Integrated Development Environment</i> (IDE) dan <i>tools</i> di pasaran	8	8	6
4.	Kemudahan 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.	Trend semasa	9	6	4
JUMLAH		83	72	59

Kajian melalui Sistem Repositori Aplikasi Sektor Awam (RASA), mendapati Maklumat Teknologi Aplikasi yang sering digunakan oleh agensi sektor awam di dalam pembangunan sistem adalah seperti Rajah 3.32

Bil	Bahasa Pengaturcaraan	Web Application Framework	Pangkalan Data	Operating System	Web Server
1	PHP	Laravel, CodeIgniter, Yii, Zend, CakePHP	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Apache, Nginx
2	Java	SmartGWT	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, Informix, TIBERO, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Tomcat, WildFly, JBoss, Nginx
3	C#	ASP.NET	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB, Microsoft Access	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux	IIS
4	JavaScript	Angular, Vue	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Nginx
5	LotusScript	(Tiada rangka kerja spesifik)	Domino DB	Windows Server, Other Linux, Other Unix	
6	C++	(Jarang digunakan untuk web)	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Apache, Nginx
7	Python	Django	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Nginx, Apache
8	Dart	AngularDart	MySQL, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Nginx, Apache
9	Formula Language	(Tiada rangka kerja spesifik)	Domino DB	Windows Server, Other Linux, Other Unix	
10	TypeScript	Angular, Vue	MySQL, Microsoft SQL Server, Oracle, MariaDB, PostgreSQL, MongoDB	Windows Server, CentOS Linux, Ubuntu Linux, Red Hat Linux, Other Linux, Solaris	Nginx, Apache

Rajah 3.23. : Trend Penggunaan Maklumat Teknologi Aplikasi Sektor Awam

Melalui padanan maklumat di dalam Rajah 3.23, *tools* yang popular dalam pembangunan sistem aplikasi masa kini mengikut aktiviti Metodologi KRISA adalah seperti Rajah 3.24;



Rajah 3.24. : Tools yang Popular dalam Pembangunan Sistem Aplikasi Sektor Awam mengikut Metodologi KRISA

3.10 PENYEDIAAN SPESIFIKASI KEPERLUAN SISTEM [F2.6]

3.10.1 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 dimana pengisian kandungan-kandungannya adalah seperti langkah-langkah di bawah.

3.10.2 OBJEKTIF

- a) Menyediakan Model Fungsi Sistem hasil daripada langkah-langkah yang dilaksanakan di dalam Pemodelan Fungsi Sistem [F2.2].
- b) Menyediakan Model Maklumat, Definisi Entiti dan Definisi Atribut dengan merujuk kepada langkah-langkah di bawah Pemodelan Keperluan Data [F2.3].
- c) Menyediakan Model dan Definisi Proses Sistem dengan merujuk kepada langkah-langkah di bawah Pemodelan Proses Sistem [F2.4].
- d) Menyertakan maklumat yang telah dikenal pasti di dalam Penentuan Keperluan Bukan Fungsian [F2.5].
- e) 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 3.15 : 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 : 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 6 : 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 7 : 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)

3.11 PENYEDIAAN PELAN MIGRASI DATA [F2.7]

Migrasi data merupakan proses memindahkan data daripada sumber asal (sama ada daripada format excel, *hard copies* atau sebagainya) atau pangkalan data sumber (sistem legasi/lama) ke destinasi pangkalan data baharu.

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 di dalam sistem baharu selaras dengan keperluan bisnes.

Empat (4) aktiviti utama dalam migrasi data iaitu:

- a) Kenal Pasti keperluan Migrasi Data di Fasa Permulaan;
- b) Penyediaan Pelan Migrasi Data di Fasa Analisis;
- c) Reka bentuk Migrasi Data di Fasa Rekabentuk; dan
- d) Pelaksanaan Migrasi Data di Fasa Pelaksanaan

3.11.1 KETERANGAN

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

Pelan Migrasi Data 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.

3.11.2 OBJEKTIF

Menghasilkan Pelan Migrasi Data sebagai panduan dan rujukan yang menetapkan strategi, kaedah dan jadual pelaksanaan dalam menjayakan pelaksanaan migrasi data.

Langkah-Langkah

Langkah 1 : Kenal pasti data yang terlibat

Keperluan Migrasi Data telah direkodkan dalam Spesifikasi Keperluan Bisnes semasa Fasa Permulaan Projek. Data yang terlibat dalam pelaksanaan migrasi perlu dikenal pasti dengan lebih terperinci bagi membolehkan strategi dan kaedah pelaksanaan migrasi dapat dirancang dengan teliti.

Berdasarkan Keperluan Migrasi Data dan Templat Definisi Aktiviti Fungsi Bisnes dalam Spesifikasi Keperluan Bisnes, lengkapkan maklumat-maklumat berikut.

Jadual 3.16: Data yang terlibat dalam migrasi

Perkara	Keterangan
ID Fungsi Bisnes	ID fungsi bisnes dari rajah Hierarki Fungsi Bisnes dalam Dokumen Keperluan Bisnes
Nama Fungsi Bisnes	Nama fungsi bisnes dari rajah Hierarki Fungsi Bisnes dalam Dokumen Keperluan Bisnes
Data Migrasi yang diperlukan	Senarai data yang perlu dimigrasikan berdasarkan keterangan dalam Definisi Aktiviti Fungsi Bisnes dalam Dokumen Keperluan Bisnes. Perincian sekiranya berkaitan.
Kategori Data	Kategori data sama ada data terbuka atau terperingkat daripada sistem legasi. Maklumat ini diperlukan sebagai input kepada strategi dan kaedah migrasi.
Bila data perlu dimigrasikan	Nyatakan bilakah data perlu dimigrasikan dan maklumat ini perlu diambil kira dalam penyediaan jadual pelaksanaan.
Sumber Data	Nama agensi, sistem dan pangkalan data atau fail sumber untuk migrasi data
Jenis pangkalan data dan versi	Maklumat persekitaran teknologi yang digunakan seperti jenis pangkalan data (DBMS) dan versi.
Jumlah rekod dalam pangkalan data	Jumlah rekod dalam pangkalan data atau fail luaran yang hendak dimigrasikan
Bilangan table	Bilangan table dalam pangkalan data yang terlibat dalam migrasi data
Isu kualiti data dan kompleksiti data	Nyatakan isu kualiti data dalam pangkalan data sedia ada (data tidak standard, pertindihan data, perlukan <i>conversion data</i>) dan kompleksiti data (compatibility issues, data calculation/formula, data encryption and security measures, data relationships and dependencies, data structure and format)

Langkah 2 : Kenal Pasti Strategi Migrasi Data

Strategi migrasi data boleh ditentukan berdasarkan pendekatan pembangunan sistem. Terdapat dua jenis strategi migrasi data yang biasa digunakan iaitu sama ada secara sekali sahaja atau secara berfasa.

a) Migrasi data secara sekali sahaja

Sekiranya pembangunan sistem dilaksanakan adalah secara sekali sahaja, strategi migrasi data adalah secara *one-off*.

b) Migrasi data secara berfasa

Sekiranya pembangunan sistem adalah secara berperingkat, pelaksanaan migrasi data adalah secara berfasa. Contoh sekiranya pembangunan sistem secara berperingkat adalah mengikut modul, maka strategi migrasi data adalah berfasa berdasarkan modul-modul yang telah dibangunkan. Perlu diambil kira juga sekiranya perlu melaksanakan sinkronisasi data daripada sistem lama (legasi) bagi mengekalkan pengemaskinian data terkini.

Pemilihan strategi juga bergantung kepada kategori data seperti tahap kekerapan data dikemaskini. Sekiranya kekerapan adalah rendah seperti data adalah statik dan jarang berubah seperti data rujukan, boleh dipindahkan terlebih dahulu. Sekiranya kekerapan data dikemaskini adalah tinggi seperti data transaksi, boleh dipindahkan kemudian berdasarkan keperluan.

Langkah 3 : Kenal Pasti Aktiviti dalam Fasa Migrasi Data

Aktiviti migrasi data adalah proses-proses yang terlibat semasa melaksanakan migrasi data iaitu dalam fasa pra-migrasi data, fasa pelaksanaan migrasi data dan fasa selepas migrasi data. Proses utama yang terlibat dalam setiap fasa migrasi data adalah seperti berikut.

Jadual 3.17 : Penerangan Fasa Migrasi Data

Fasa Migrasi Data	Proses dan aktiviti yang terlibat
Pra-migrasi data	a. Pengumpulan data : proses mengumpul data daripada sumber-sumber data iaitu meliputi format data yang ditetapkan b. Pemprofilan data : proses meneliti dan menganalisis data bagi menggambarkan kualiti data secara keseluruhan c. Pembersihan data : proses menyelesaikan masalah data (ralat, tidak konsisten, duplikasi, hilang) bagi meningkatkan kualiti data d. Transformasi data : menukarkan format data mengikut keperluan pangkalan data baharu e. Pemetaan data : data dipadankan mengikut struktur pangkalan data baharu f. Persampelan data : menggunakan data sebenar (jumlah yang bersesuaian) yang dijadikan sebagai sampel bagi tujuan pembangunan dan pengujian proses migrasi g. Penyediaan skrip migrasi data
Pelaksanaan migrasi data	a. Ekstrak data dari pangkalan data sumber dan melalui proses pembersihan dan transformasi data menggunakan alatan

	(tools)
	b. Muatkan data yang telah dipetakan ke dalam pangkalan data baharu
Selepas migrasi data	a. Pengujian integriti dan pengesahan data b. Penyediaan Laporan Migrasi Data

Jelaskan setiap proses yang terlibat bagi kaedah migrasi data mengikut fasa migrasi dalam pelan ini.

Langkah 4 : Kenal Pasti Kaedah Pengujian Data dan Pengesahan Data Yang Di migrasi

a) Kaedah Pengujian Data

Pengujian data dijalankan bagi mendapatkan ketepatan bilangan data yang berjaya dimigrasi. Jelaskan kaedah bagi pengujian data yang dimigrasi seperti pengujian bilangan baris data.

b) Kaedah Pengesahan Data

Pengesahan data dijalankan bagi memastikan bahawa data yang dipindahkan adalah betul dan tepat. Jelaskan kaedah pengesahan data yang sesuai bagi memastikan data daripada sistem lama (legasi) dapat digunakan dalam sistem baharu selaras dengan keperluan bisnes.

Langkah 5 : Kenal Pasti Strategi Pembetulan Data

Pembetulan data boleh dilaksanakan pada dua keadaan iaitu semasa pembangunan skrip migrasi data menggunakan persampelan data dan selepas pelaksanaan migrasi data.

a) Semasa pembangunan migrasi data

Pembangunan skrip migrasi data melibatkan pembersihan data, transformasi dan pemetaan data. Skrip migrasi ini akan diuji menggunakan persampelan data. Sebarang ralat data yang ditemui seperti antaranya data hilang, tidak sah dan bertindih akan dibuat tindakan pembetulan. Tindakan ini merupakan input kepada pembangunan skrip migrasi data dan tindakan pembetulan ini adalah secara berulang sehinggalah skrip migrasi yang dibangunkan adalah sempurna dan optimum.

b) Selepas pelaksanaan migrasi data

Sebarang ralat data yang dikenal pasti selepas pelaksanaan migrasi data hendaklah dibuat tindakan pembedahan. Strategi pelaksanaan pembedahan data dilakukan sama ada secara manual atau menggunakan *tools* perlu dijelaskan dalam pelan migrasi sebagai rujukan kepada tindakan yang perlu dibuat.

Jelaskan strategi pembedahan data yang akan dilaksanakan dalam pelan migrasi data.

Langkah 6 : Kenal Pasti Persekitaran Migrasi Data

Jelaskan persekitaran yang digunakan sepanjang pelaksanaan migrasi data. Persekitaran yang digunakan adalah termasuk *tools*, server dan keperluan teknikal lain yang digunakan dalam pembangunan skrip migrasi, pengujian dan produksi iaitu merangkumi perkakasan dan perisian. Gambarkan persekitaran yang digunakan bagi pelaksanaan migrasi data.

Langkah 7 : Kenal Pasti Andaian dan Risiko

Jelaskan andaian dan risiko dalam pelaksanaan migrasi data.

- a) Andaian adalah keadaan dijangka berlaku yang memberi kesan positif atau negatif sepanjang pelaksanaan migrasi data.
- b) Risiko adalah keadaan yang memberi kesan ke atas objektif migrasi data iaitu berpunca daripada risiko dalaman dan luaran. Risiko dalaman adalah keadaan yang boleh dikawal oleh pasukan projek. Manakala risiko luaran adalah keadaan yang berlaku di luar kawalan pasukan projek.

Jelaskan mitigasi yang akan dilaksanakan berdasarkan risiko yang dikenal pasti.

Langkah 8 : Kenal Pasti Pasukan Projek

Kenal pasti ahli pasukan yang terlibat dalam pelaksanaan migrasi data dan peranan setiap ahli tersebut. Jelaskan pasukan projek dan peranan dalam bentuk jadual atau carta.

Langkah 9 : Sediakan Jadual Pelaksanaan

Sediakan jadual pelaksanaan migrasi data yang dicadangkan dalam bentuk carta gantt atau mana-mana carta yang bersesuaian. Jadual pelaksanaan hendaklah selaras dengan jadual pelaksanaan pembangunan sistem.

Langkah 10 : Bangunkan Pelan Migrasi Data

Bangunkan Pelan Migrasi Data dengan merujuk kepada langkah-langkah seperti di atas. Pelan Migrasi Data dibangunkan perlu mengandungi **sekurang-kurangnya** perkara seperti berikut :

Jadual 3.18: Isi Kandungan Pelan Migrasi Data

Tajuk	Isi Kandungan
Tujuan Dokumen	Penerangan tujuan dokumen dihasilkan sebagai panduan pelaksanaan migrasi data bagi sistem legasi kepada sistem baharu. Nama sistem perlu dinyatakan dengan jelas.
Latar Belakang	Pengenalan ringkas berkaitan fungsi sistem dan impak pelaksanaan migrasi data kepada bisnes. Jelaskan impak sekiranya migrasi tidak dilaksanakan mengikut jadual kepada bisnes agensi. Adakah akan mengganggu aliran proses bisnes?
Objektif Migrasi	Penerangan objektif migrasi data dilaksanakan.
Skop Migrasi	Penerangan bagi senarai data migrasi yang terlibat.
Strategi Migrasi Data	Penjelasan tentang strategi pelaksanaan migrasi data seperti yang diterangkan di Langkah 2.
Kaedah Migrasi Data	Penjelasan tentang kaedah migrasi data bagi semua proses yang terlibat dalam pelaksanaan migrasi data seperti yang diterangkan di Langkah 3.
Kaedah Pengujian dan Pengesahan Data	Penjelasan tentang strategi pengujian data yang dimigrasi dan kaedah pengesahan data yang telah dimigrasi seperti yang diterangkan di Langkah 4.
Strategi Pembetulan Data	Penjelasan tentang strategi pembetulan data sepanjang migrasi data dilaksanakan seperti yang diterangkan di Langkah 5.
Persekitaran Migrasi Data	Penjelasan tentang persekitaran yang digunakan bagi pelaksanaan migrasi data.
Andaian dan Risiko	Penjelasan tentang andaian dan risiko dalam pelaksanaan migrasi data.
Pasukan Projek	Struktur organisasi bagi pasukan projek dan menyertakan nama-nama pegawai terlibat serta peranan.

Tajuk	Isi Kandungan
Jadual Pelaksanaan	Penerangan jadual pelaksanaan migrasi data yang dicadangkan dalam bentuk <i>gant chart</i> atau mana-mana carta yang bersesuaian. Ia mengandungi aktiviti migrasi bermula daripada analisis keperluan sehingga penyediaan laporan. Jadual pelaksanaan juga perlu mengambil kira aspek seperti berikut : - Susunan keutamaan data yang ingin dipindahkan; - Tempoh downtime yang dibenarkan untuk sistem semasa, sekiranya melibatkan sistem yang sedang beroperasi; - Menggambarkan pendekatan pelaksanaan migrasi data; dan - Selaras dengan jadual pembangunan sistem.

Pelan Migrasi Data perlu dikemaskini dari semasa ke semasa mengikut keperluan yang telah dipersetujui. Rujuk format **Pelan Migrasi Data**.

Langkah 12: 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. Oracle White Paper (2011). Successful Data Migration. <http://www.oracle.com/technetwork/middleware/oedq/successful-data-migration-wp-1555708.pdf>
2. Credesoft White Paper. Eight key steps which help ensure a successful data migration project: A white paper for inspection management professionals.
3. SAGA Group (2012). Methods of Data Migration.
4. Astera Whitepaper: Data Profiling: The First Step in the Data Migration Process

3.12 PENYEDIAAN PELAN INTEGRASI SISTEM [F2.8]

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 digunakan semasa melaksanakan tugas secara bersepadu dan *seamless*. Ini dapat meningkatkan produktiviti penyampaian perkhidmatan digital dan memudahkan organisasi mencapai matlamatnya. Dalam pada itu, integrasi sistem juga dapat meningkatkan ketepatan dan kebolehpercayaan data.

Tiga (3) aktiviti utama integrasi sistem iaitu:

- a) Kenal pasti keperluan Integrasi Sistem di Fasa Permulaan;
- b) Penyediaan Pelan Integrasi Sistem di Fasa Analisis; dan
- c) Reka Bentuk Integrasi Sistem di Fasa Reka Bentuk.

3.12.1 KETERANGAN

Keperluan integrasi diambil kira semasa proses mengenal pasti proses bisnes sebagaimana diterangkan dalam **Pemodelan Proses Bisnes [F1.4]**. Analisis keperluan integrasi sistem perlu dilaksanakan bagi menentukan skop kerja integrasi sistem. Pelan integrasi dibangunkan sebagai panduan dan rujukan bagi keseluruhan pelaksanaan integrasi sistem. Pelan integrasi sistem menggariskan kaedah, strategi dan jadual pelaksanaan integrasi yang perlu dipatuhi.

3.12.2 OBJEKTIF

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

Langkah-Langkah

Langkah 1: Kenal pasti skop yang terlibat

Keperluan awal integrasi sistem telah dimasukkan dalam Spesifikasi Keperluan Bisnes semasa Fasa Permulaan Projek. Sistem luaran yang terlibat dalam pelaksanaan integrasi iaitu data keluar (inbound) dan data masuk (outbound) perlu dikenal pasti dengan lebih terperinci bagi membolehkan strategi dan kaedah dapat dirancang dengan teliti. Keperluan integrasi sistem yang lebih terperinci adalah sama ada untuk semakan atau penyatuan data dan pendekatan sama ada secara *real time* atau berkala. Kenal pasti juga sekiranya terdapat komponen-komponen yang terlibat dalam integrasi sistem.

Berdasarkan keperluan awal integrasi sistem dalam Spesifikasi Keperluan Bisnes, lengkapkan maklumat-maklumat berikut.

Jadual 3.19: Keperluan Awal Integrasi Sistem

Tajuk	Isi Kandungan
ID Fungsi Bisnes	ID fungsi bisnes dari rajah Hierarki Fungsi Bisnes dalam Dokumen Keperluan Bisnes
Nama Fungsi Bisnes	Nama fungsi bisnes dari rajah Hierarki Fungsi Bisnes dalam Dokumen Keperluan Bisnes
ID Aktiviti Bisnes	ID aktiviti bisnes dari rajah aliran proses dalam Dokumen Keperluan Bisnes
Nama Aktiviti Bisnes	Nama aktiviti bisnes dari rajah aliran proses dalam Dokumen Keperluan Bisnes
Nama Sistem Luaran/Agensi/Komponen	Nama sistem luaran atau agensi yang terlibat dalam integrasi data
Inbound / Outbound	Data masuk (Inbound) atau data keluar (outbound)
Data / Maklumat	Data yang terlibat dalam integrasi
Klasifikasi Data/Maklumat	Klasifikasi data (terbuka, terperingkat)
Tujuan Integrasi	Nyatakan tujuan integrasi data dilaksanakan
Integrasi Sedia Ada? (Ya/Tidak)	Nyatakan sekiranya terdapat integrasi sebelum ini yang berkaitan
Kaedah Integrasi (<i>real time</i> / <i>batch</i> / lain-lain kaedah - nyatakan)	Nyatakan kaedah integrasi sedia ada sekiranya ada. Sekiranya tiada, nyatakan kaedah integrasi yang dicadangkan.

Kajian dan perbincangan perlu melibatkan antara pemilik proses, pemilik sistem dan pasukan pembangun integrasi.

Langkah 2: Kenal pasti strategi pelaksanaan integrasi sistem

Strategi pelaksanaan integrasi sistem adalah menerangkan perancangan aktiviti yang diadakan bagi keperluan lanjut integrasi sistem.

- a) Menerangkan strategi yang dilakukan untuk mengenal pasti integrasi sistem yang diperlukan dan pelaksanaan integrasi tersebut seperti berikut.
 - i) Nyatakan perancangan aktiviti yang perlu dilaksanakan untuk mengenal pasti keperluan integrasi contoh seperti mendapat keperluan awal integrasi semasa menyediakan Dokumen Keperluan Bisnes.
 - ii) Nyatakan strategi yang digunakan dalam mendapatkan keperluan integrasi contoh seperti mengenal pasti integrasi berdasarkan modul dan medium yang akan digunakan seperti mesyuarat, perbincangan atau bengkel.
 - iii) Pelaksanaan strategi hendaklah selari dengan akta, peraturan, pekeliling dan garis panduan perkongsian data yang berkuatkuasa.
- b) Strategi pelaksanaan integrasi sistem hendaklah dikenal pasti semasa penyediaan keperluan bisnes di Fasa Permulaan Projek.
- c) Strategi pelaksanaan pembangunan integrasi sistem sama ada dilaksanakan bersekali dengan pembangunan sistem utama ataupun selepas sistem utama siap.

Langkah 3: Kenal pasti kaedah pengujian dan pengesahan data yang diintegrasikan

a) Ujian integriti data

Ujian integriti data dilakukan adalah untuk memastikan data kekal konsisten dan tepat selepas proses integrasi. Pendekatan yang digunakan dalam Ujian Integriti Data adalah dengan memeriksa bahawa data tidak rosak semasa proses integrasi dan memastikan bahawa data yang diintegrasikan adalah padan dan diselaraskan dengan betul. Ujian ini boleh dilaksanakan bersekali dengan pengujian modul.

b) Ujian dan pengesahan data secara menyeluruh

Ujian data secara menyeluruh dilakukan untuk memastikan keseluruhan aliran data yang diintegrasikan adalah tepat hingga proses akhir. Objektif ujian ini adalah untuk memastikan data bergerak lancar antara sistem yang diintegrasikan tanpa ralat dan ianya akan melibatkan pengesahan data pada setiap langkah untuk memastikan tiada masalah berlaku sepanjang proses. Ujian ini boleh dilaksanakan bersekali dengan pengujian modul.

Langkah 4: Kenal Pasti Persekitaran Sistem Yang Diintegrasikan

Jelaskan persekitaran yang digunakan sepanjang pelaksanaan integrasi sistem serta ujian dan pengesahan integrasi. Persekitaran yang digunakan adalah termasuk *tools*, *server* dan keperluan teknikal lain yang digunakan dalam pembangunan integrasi, pengujian dan produksi iaitu merangkumi perkakasan dan perisian. Gambarkan persekitaran yang digunakan bagi pelaksanaan integrasi sistem.

Langkah 5 : Kenal Pasti Andaian dan Risiko

Jelaskan andaian dan risiko dalam pelaksanaan integrasi sistem.

- a) Andaian adalah keadaan dijangka berlaku yang memberi kesan positif atau negatif sepanjang pelaksanaan integrasi sistem.
- b) Risiko adalah keadaan yang memberi kesan ke atas objektif integrasi sistem iaitu berpunca daripada risiko dalaman dan luaran. Risiko dalaman adalah keadaan yang boleh dikawal oleh pasukan projek. Manakala risiko luaran adalah keadaan yang berlaku di luar kawalan pasukan projek.

Jelaskan mitigasi yang akan dilaksanakan berdasarkan risiko yang dikenal pasti.

Langkah 6: Kenal Pasti Pasukan Projek

Kenal pasti ahli pasukan yang terlibat dalam pelaksanaan integrasi sistem dan peranan setiap ahli tersebut. Jelaskan pasukan projek dan peranan dalam bentuk jadual atau carta.

Langkah 7: Sediakan Jadual Pelaksanaan

Sediakan jadual pelaksanaan integrasi yang dicadangkan dalam bentuk carta gantt atau mana-mana carta yang bersesuaian. Jadual pelaksanaan hendaklah selaras dengan jadual pelaksanaan pembangunan sistem.

Langkah 8: Bangunkan Pelan Integrasi

Pelan Integrasi mengandungi sekurang-kurangnya perkara seperti berikut:

Jadual 3.20: Isi Kandungan Pelan Integrasi Sistem

Kandungan	Keterangan
Tujuan	Terangkan tujuan dokumen ini dihasilkan.
Objektif	Terangkan objektif pengintegrasian sistem ini
Skop kerja integrasi	Terangkan skop kerja integrasi yang terlibat
Strategi pelaksanaan integrasi	Strategi pelaksanaan integrasi data (rujuk Langkah 2) yang merangkumi perkara berikut. a. strategi untuk mengenal pasti integrasi data yang diperlukan b. strategi pelaksanaan integrasi data
Kaedah pengujian dan pengesahan data	Terangkan kaedah pengujian dan pengesahan data yang diintegrasikan. Rujuk Langkah 3.
Persekitaran pelaksanaan integrasi	Nyatakan persekitaran yang digunakan untuk pelaksanaan integrasi termasuk persekitaran untuk pengujian dan pengesahan data. Rujuk Langkah 4.
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, reka bentuk, 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.
Lampiran / Rujukan	Sertakan minit mesyuarat atau laporan berkaitan dapatan sesi libat urus dengan agensi yang terlibat dengan integrasi data.

Pelan Integrasi Sistem perlu dikemaskini dari semasa ke semasa mengikut keperluan yang telah dipersetujui. Pelaksanaan integrasi akan dilaksanakan berdasarkan pelan yang dihasilkan. Rujuk format **D07 Pelan Integrasi Sistem**.

Langkah 9: Sahkan Pelan Integrasi

Pelan Integrasi yang didokumenkan perlu dibentang dan mendapat pengesahan pemilik sistem bagi memastikan aktiviti integrasi mendapat sokongan dan kerja sama.

Rujukan

1. Gregor Hohpe & Bobby Woolf (2003). Enterprise Integration Patterns. Addison-Wesley. ISBN: 0321200683.
2. MSC CFI Architecture – System Integration Architecture Version 3.0 (2004).
3. <https://rivery.io/data-learning-center/data-integration-techniques-and-strategies/>
4. Principle of Data Integration (2014) Alon Halevy, AnHai Doan, and Zachary Ives.