




BAB:03

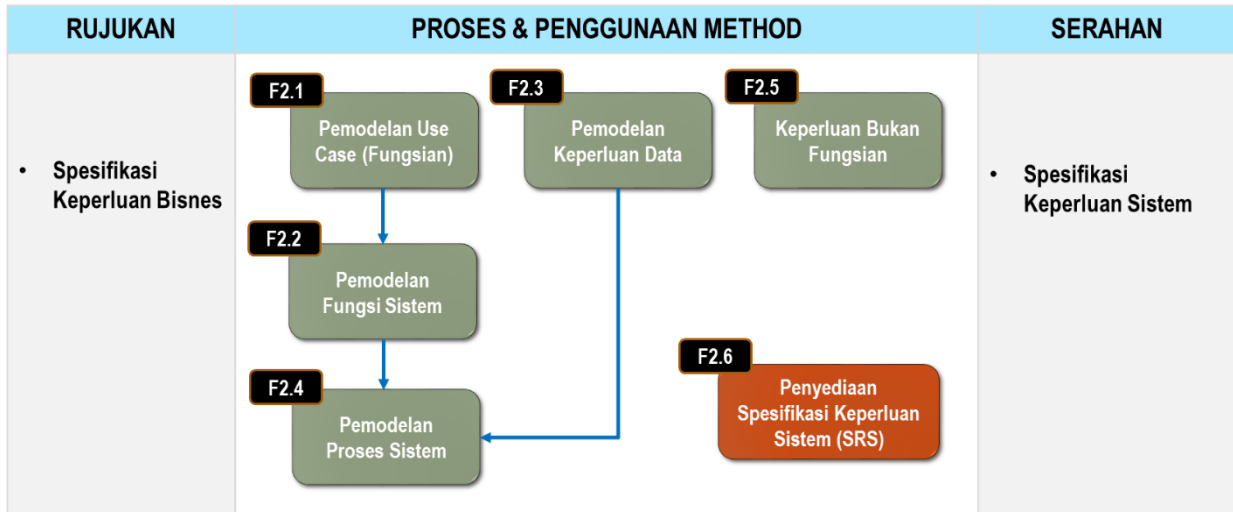
FASA ANALISIS

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 2 - Analisis

3.2 Pengenalan

Fasa analisis keperluan sistem merangkumi aktiviti-aktiviti analisis terhadap keperluan bisnes yang dikenali 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. Ia 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:

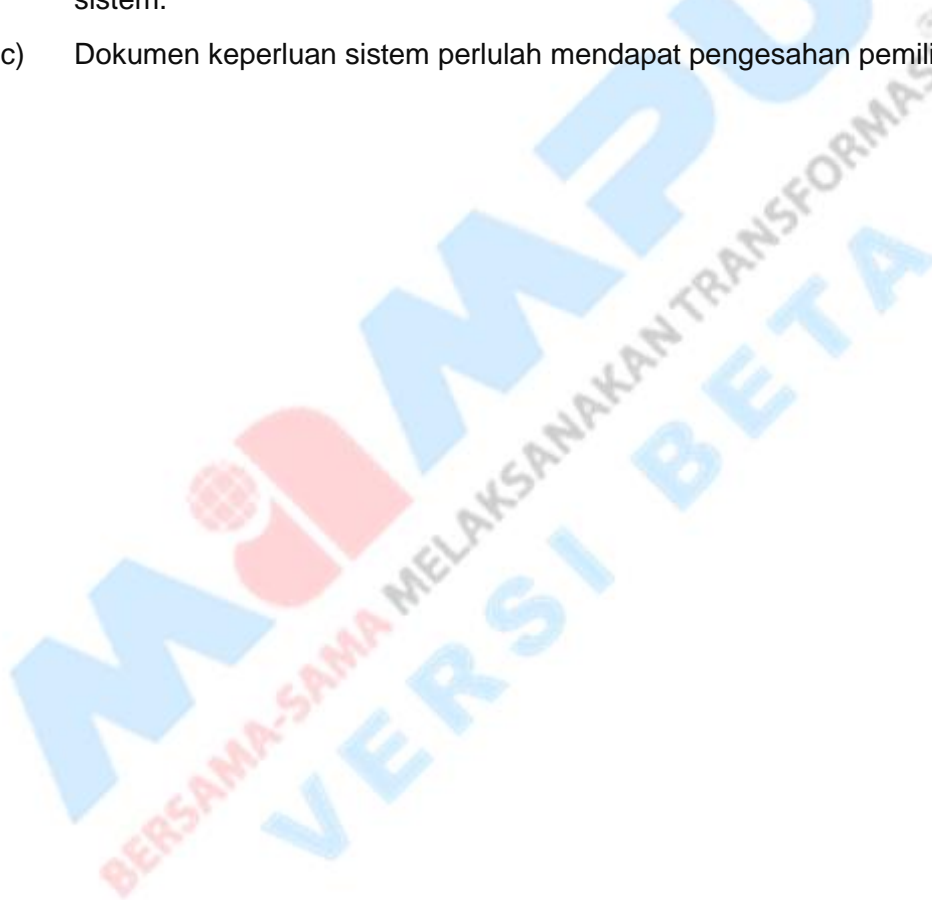
- a) Juruanalisis Sistem

- 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 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*.

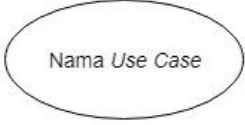


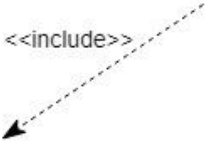
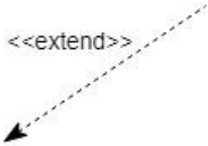

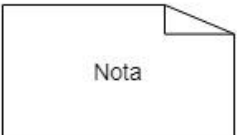
Objektif

- 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 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: 100px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> <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 Use Case</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 Use Case 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 Use Case <p>* hanya diperlukan bagi penglabelan notasi Use Case sahaja.</p>

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:

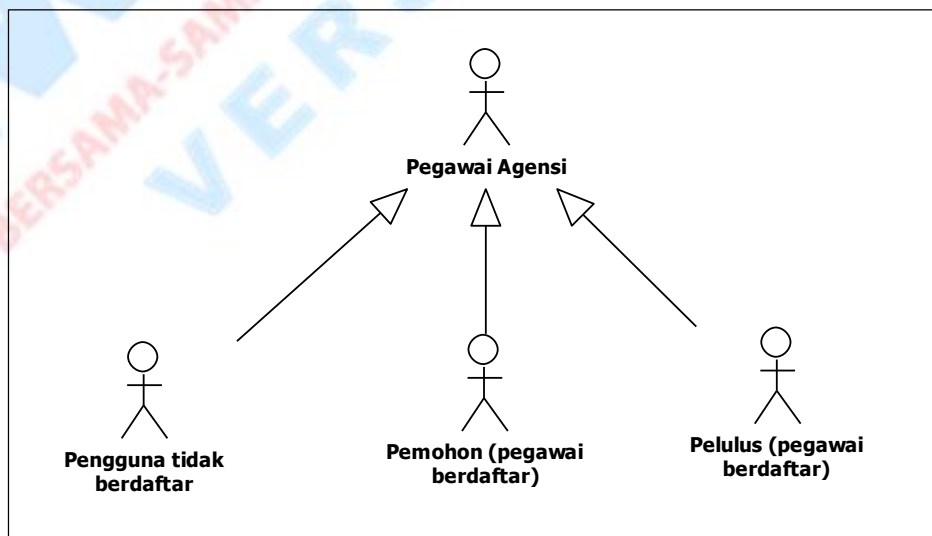
Jadual 17 : Contoh Jadual Terjemahan Peranan kepada Aktor

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:
- Aktor terletak di luar sempadan sistem/modul.
 - Aktor boleh memainkan peranan sebagai 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

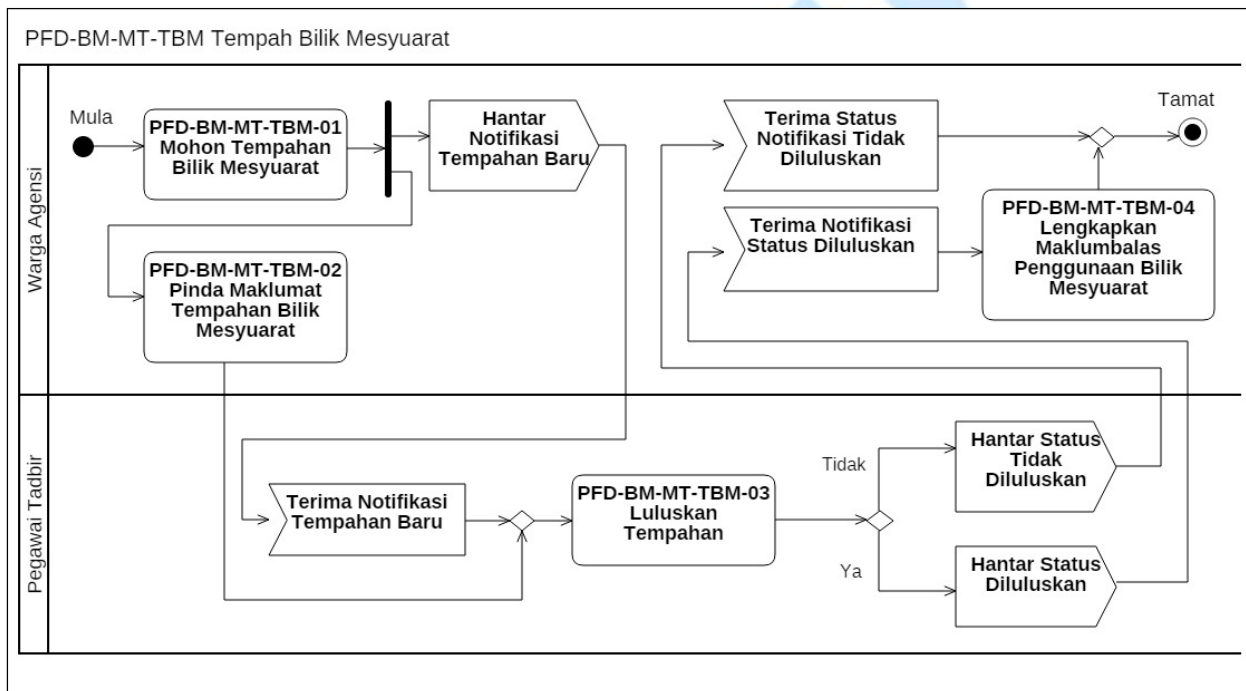
Langkah 3 : Sediakan Hubungan-hubungan Di Antara Aktor Dengan Use Case

- 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.

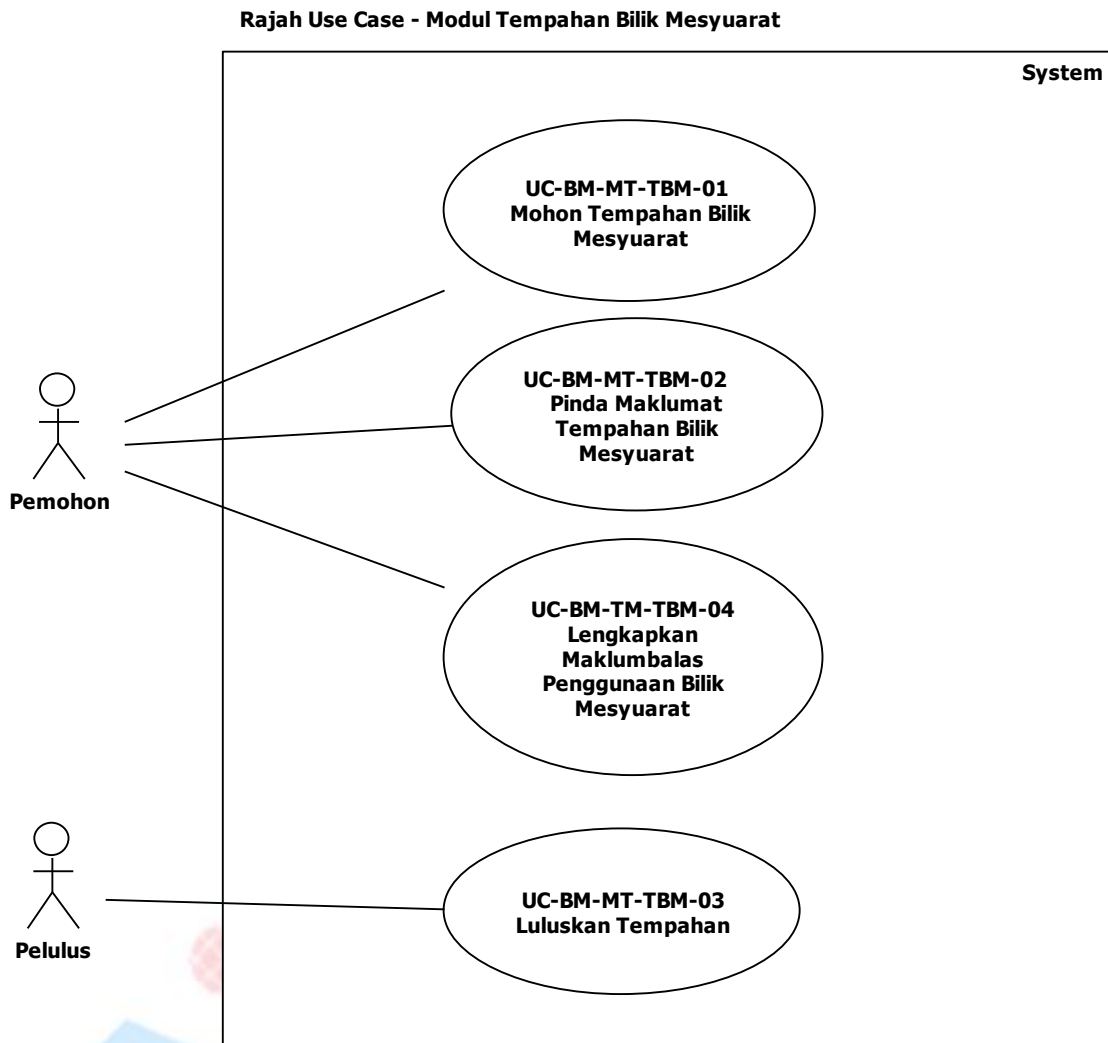
- d) Lakarkan hubungan antara Actor dan *Use Case*. Hubungan ini menunjukkan interaksi aktor kepada *Use Case*. Dalam hubungan ini notasi *association* sahaja digunakan.
- e) Sediakan keterangan setiap use case yang dilakarkan.

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

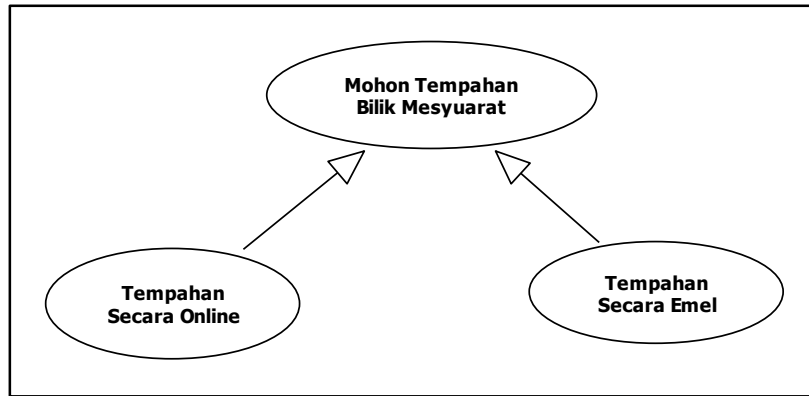
UC-BM-MT-TBM-03	Luluskan Tempahan	Pelulus menyemak dan meluluskan permohonan tempahan yang dilakukan oleh pemohon
UC-BM-MT-TBM-04	Lengkapkan maklumbalas Penggunaan Bilik Mesyuarat	Selepas penggunaan bilik pemohon diminta untuk melengkapkan borang maklumbalas penggunaan Bilik

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:

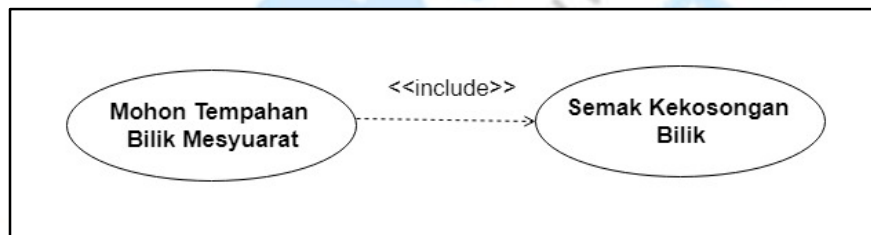
Tempahan Bilik Mesyuarat boleh dilakukan dengan 2 cara iaitu secara online yang disediakan atau tempahan boleh dilakukan secara emel kepada Pegawai Tadbir tempahan.



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.

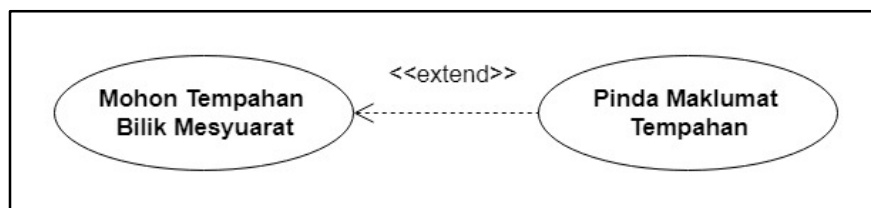
Contoh: *Use Case* Mohon Tempahan Bilik Mesyuarat perlu memanggil fungsi Semak kekosongan Bilik bagi melengkapkan proses.



Rajah 36 : Contoh Hubungan *Use Case* (Include)

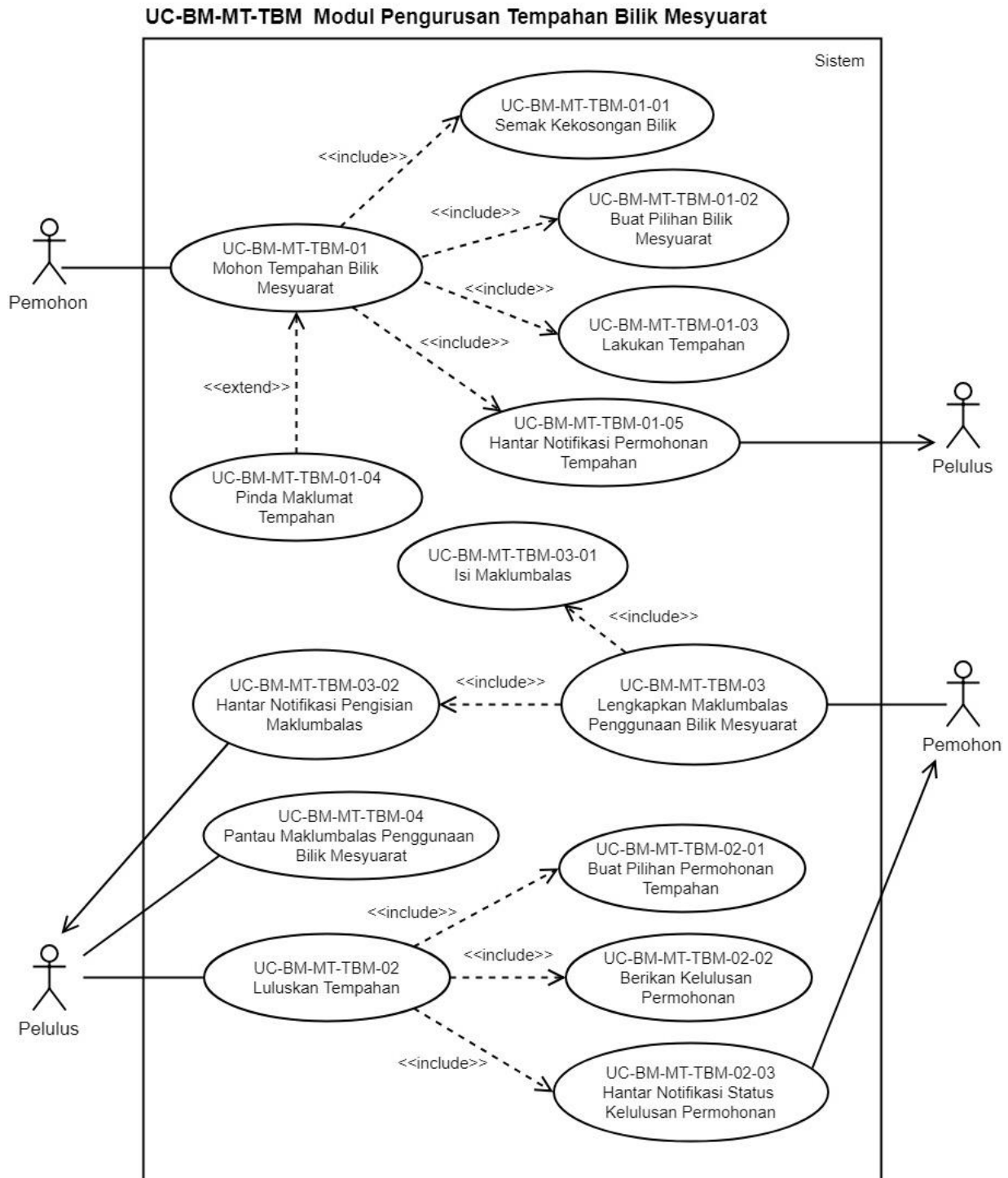
- iii) <Extend> – *Use Case* yang memberi nilai tambah kepada *Use Case* yang lain. Ianya digunakan bagi mentakrifkan sesuatu *Use Case* menggunakan fungsi dalam *Use Case* yang lain dengan syarat-syarat tertentu.

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 : Dokumentkan Rajah *Use Case*

Dokumentkan 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]**.


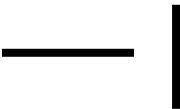
Hasil Pemodelan Fungsi Sistem direpresentasi dalam bentuk Rajah Hierarki Fungsian Sistem. Rajah ini memaparkan struktur sistem di mana tahap hierarkinya bermula dari sistem itu sendiri dan diikuti oleh subsistem-subsistem, modul-modul, submodul-submodul dan transaksi-transaksi yang berkaitan dengannya. 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 Pemaparan 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

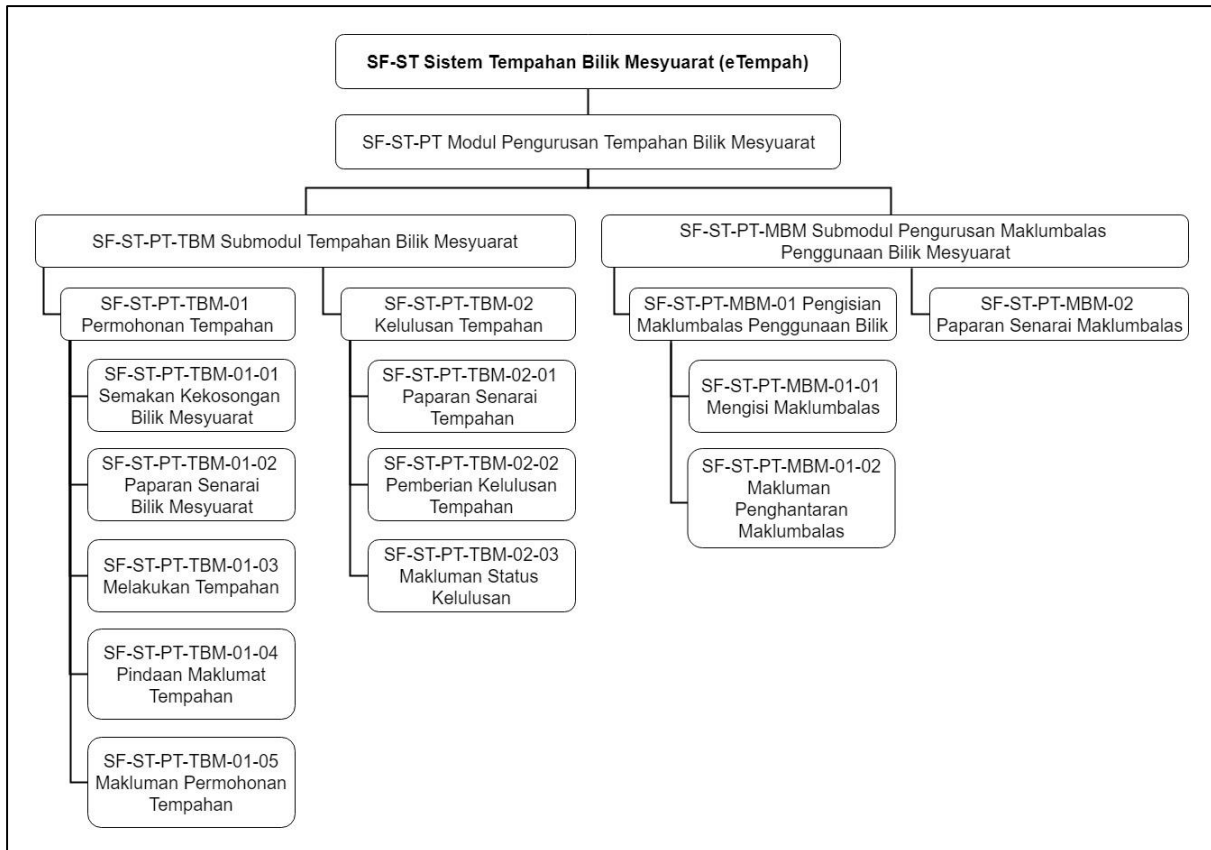
Transaksi Am merupakan komponen Fungsi Sistem yang diterjemahkan dari *Use Case* yang berhubung bukan sahaja dengan satu *Use Case* induk malah juga dengan *Use Case* induk yang lain di dalam modul/submodul yang berbeza. Kesemua Transaksi Am yang telah dikenalpasti akan dikumpulkan di bawah satu modul 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 kebiasaannya 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

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 >> Paparasi 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 >> Paparasi 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


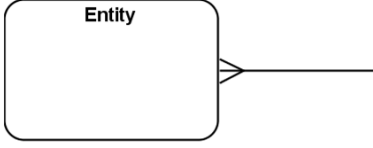



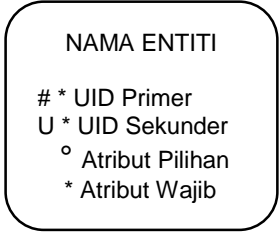
- Menyediakan model maklumat yang tepat berdasarkan keperluan di sesebuah organisasi. Model maklumat ini boleh dijadikan rangka kerja/panduan untuk menaiktaraf aplikasi sedia ada, atau pembangunan sistem aplikasi yang baharu di organisasi berkenaan.
- Menyediakan model yang tidak bersandar dengan mana-mana 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*).

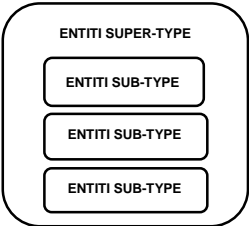
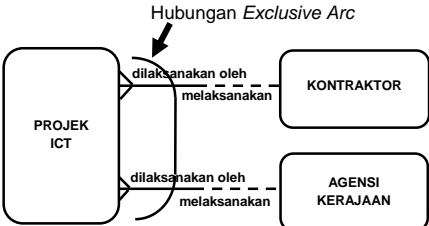

Notasi

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/ <i>Unique Identifier</i> (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 / 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 Aktivii Fungsi 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 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.

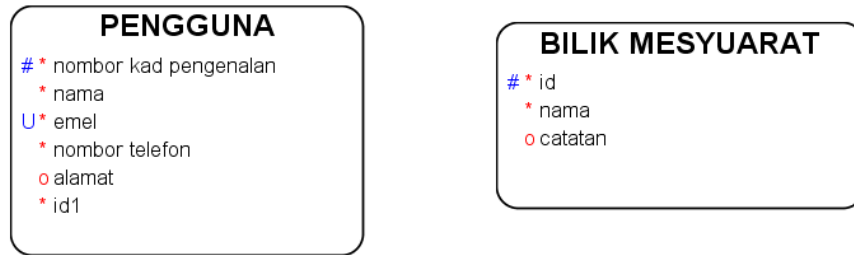
Langkah 2 : Bina Model Maklumat

- 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).

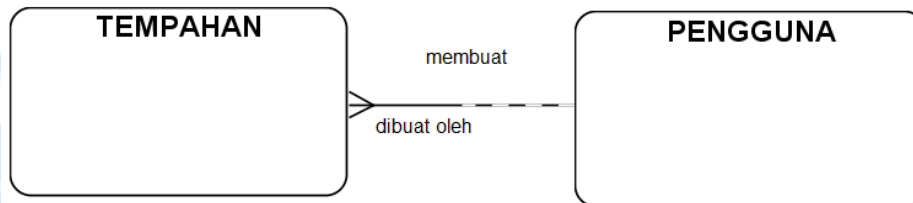


Rajah 41 : Contoh Entiti Dan Senarai Atribut

- vi) Dalam satu entiti, satu atau lebih atribut boleh dijadikan Pengenal Unik 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:

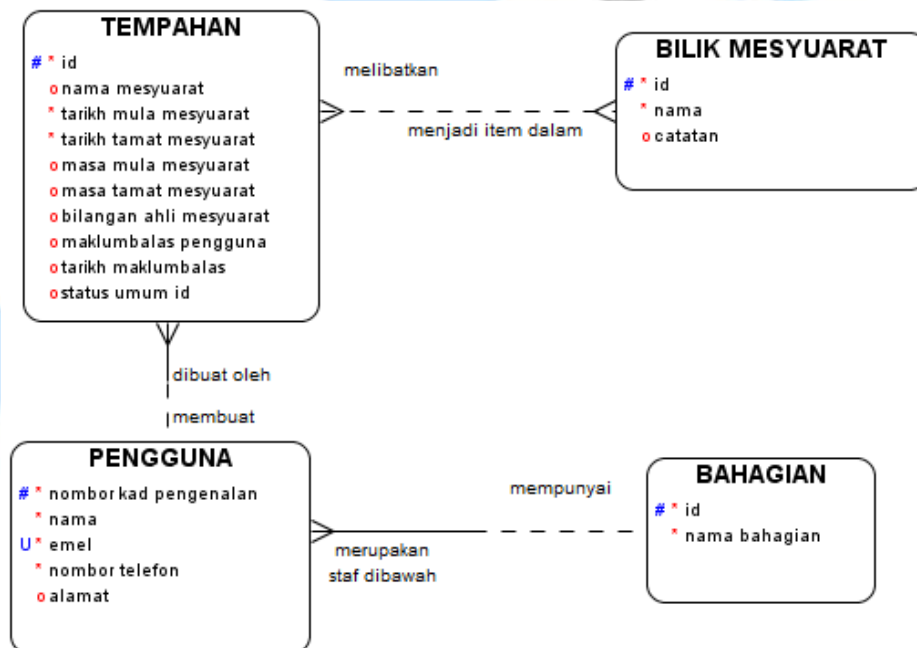
- i) Kardinaliti hubungan menunjukkan setiap TEMPAHAN dibuat oleh satu, dan hanya satu PENGGUNA. Ini bermaksud pada satu-satu tarikh dan masa tertentu, tempahan aset hanya boleh dibuat oleh seorang pengguna sahaja.
- ii) Dari segi modaliti, hubungan adalah mandatori (mesti) iaitu rekod TEMPAHAN tidak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

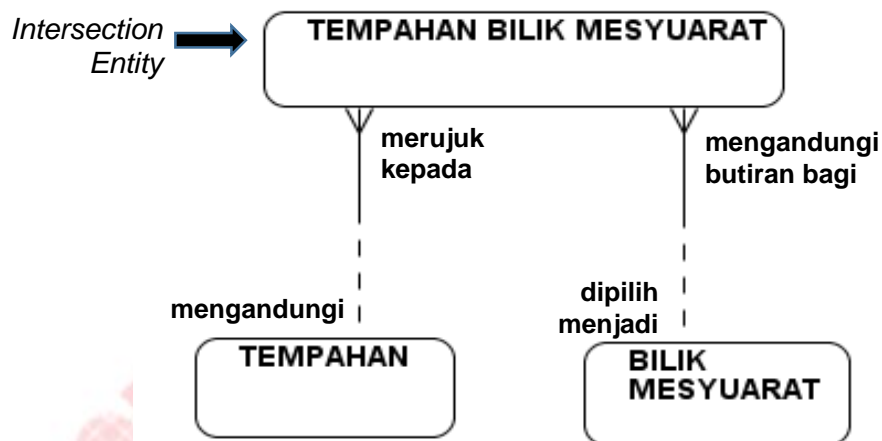
Contoh ERD yang dihasilkan adalah seperti yang di bawah.



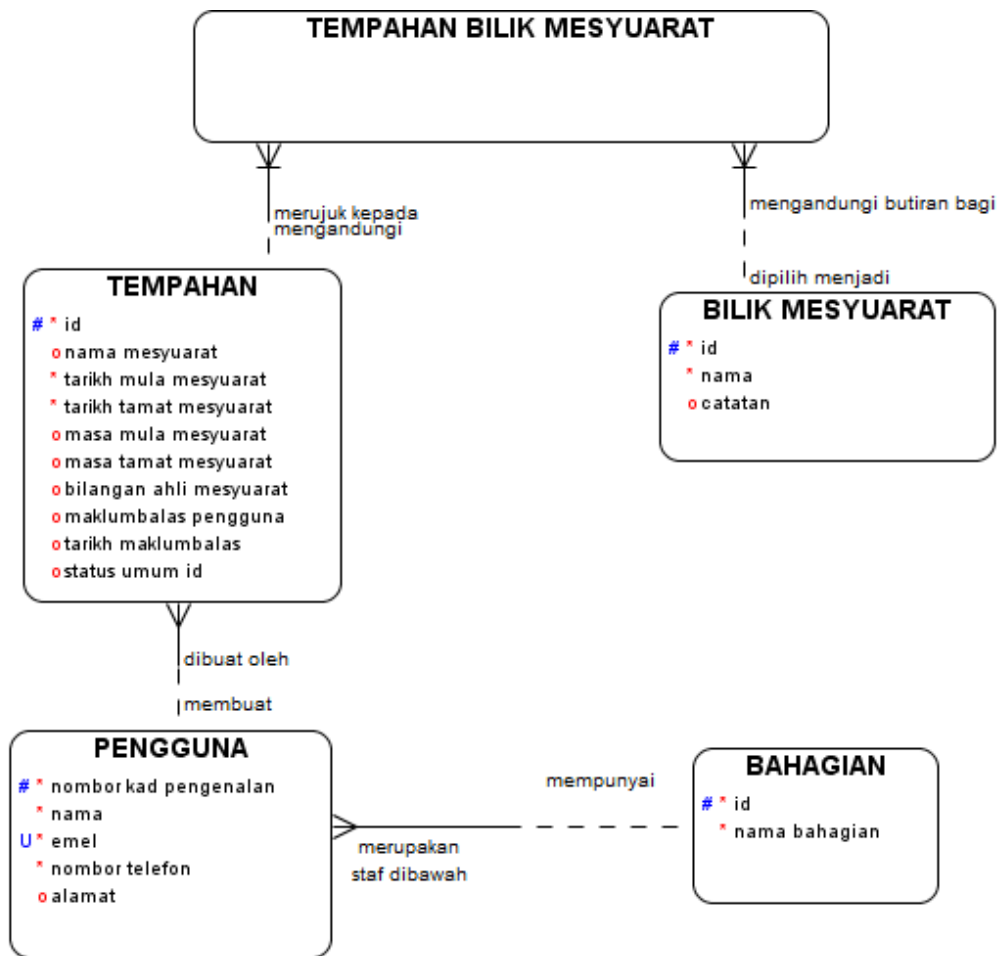
Rajah 43 : Contoh Rajah Hubungan Entiti (ERD)

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.

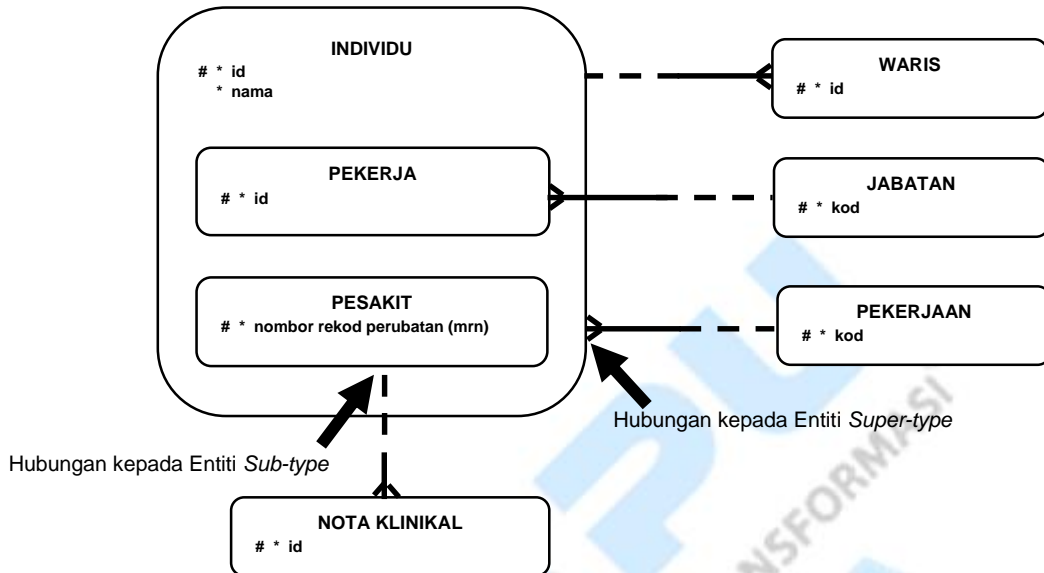
Rajah 44 : Contoh *Intersection Entity*

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*.

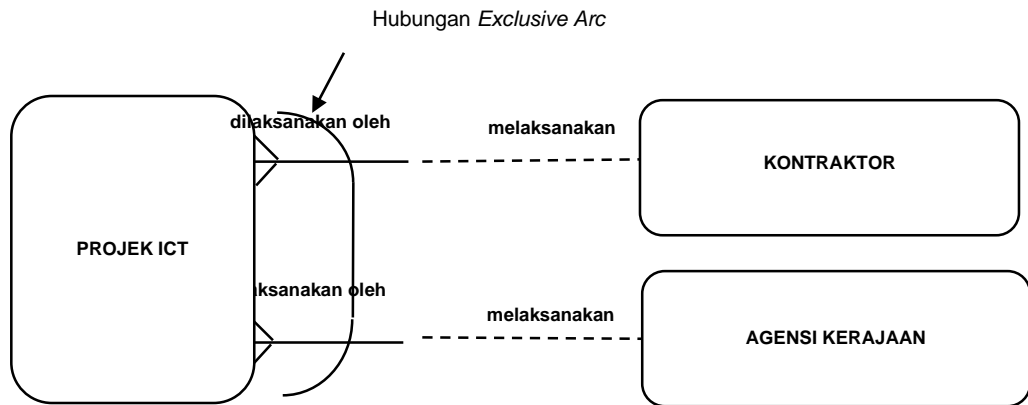
Contoh hubungan entiti lain kepada Entiti *Super-type* atau *Sub-type* adalah seperti rajah dibawah.



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:

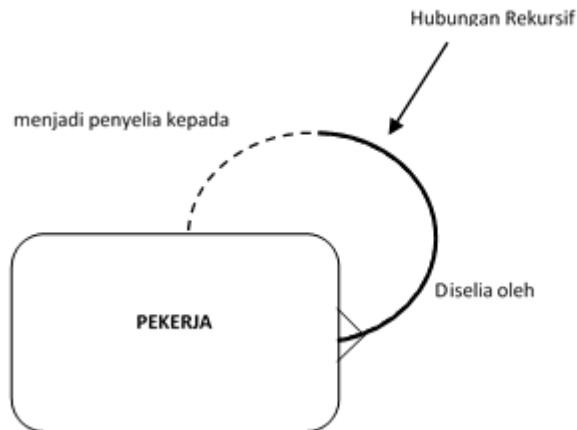
Setiap PROJEK ICT mesti dilaksanakan oleh satu, dan hanya satu KONTRAKTOR atau dilaksanakan oleh satu, dan hanya satu AGENSI KERAJAAN

- 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 iaitu:

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


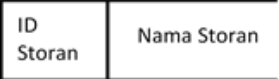
Objektif

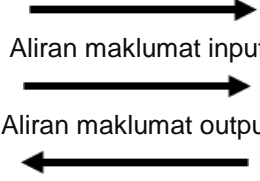
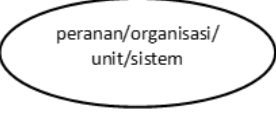
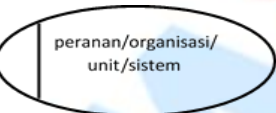
- Menghasilkan model proses sistem yang menerangkan pemprosesan yang akan dilaksanakan oleh sistem
- Mengenal pasti sumber data dan destinasi maklumat yang diperlukan oleh proses
- Menetapkan sempadan proses dan sistem yang dibangunkan

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> <table border="1" data-bbox="204 282 496 360"> <tr> <td>ID Storan</td> <td>Nama Storan</td> </tr> </table>	ID Storan	Nama Storan	<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
ID Storan	Nama Storan		
<p>Aliran Maklumat</p> <p>ID Aliran Maklumat Nama Aliran Maklumat</p>  <p>Aliran maklumat input</p> <p>Aliran maklumat output</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:</p> <p>DFD = Ringkasan Nama Teknik SS = Ringkasan Nama Sistem SBS = Ringkasan Nama Subsystem MM = Ringkasan Nama Modul SBM = Ringkasan Nama Submodul</p>		

	<p>*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

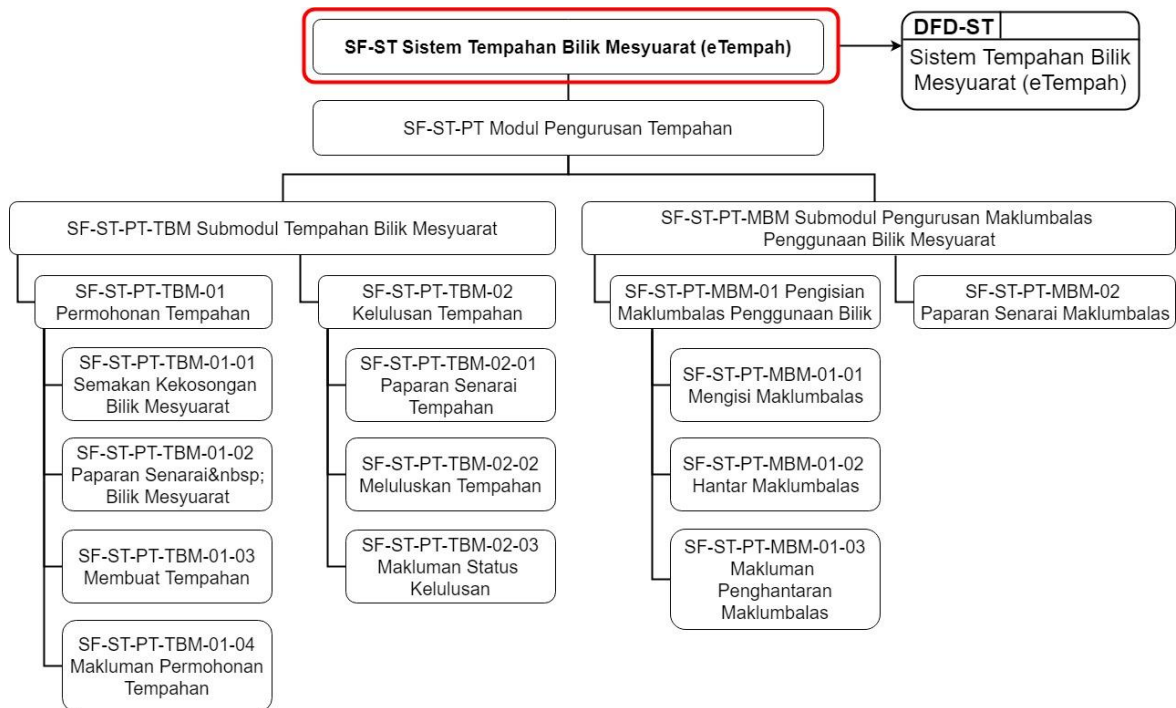
Langkah 1 : Sediakan Rajah Konteks

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.



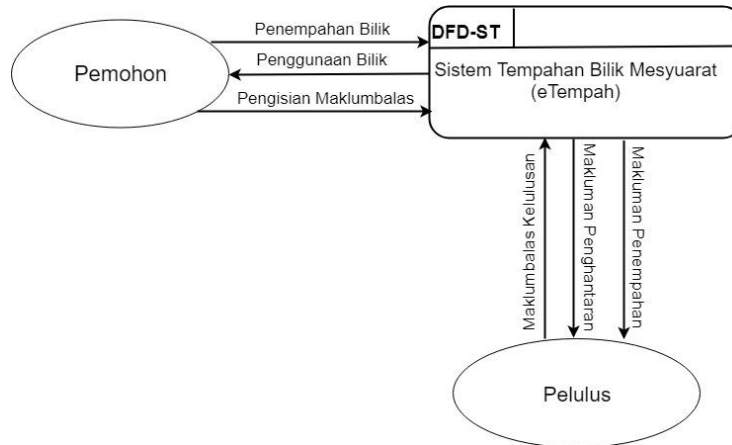
Rajah 50 : Contoh Pemetaan Hierarki Fungsi Sistem kepada Notasi Fungsi

- 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

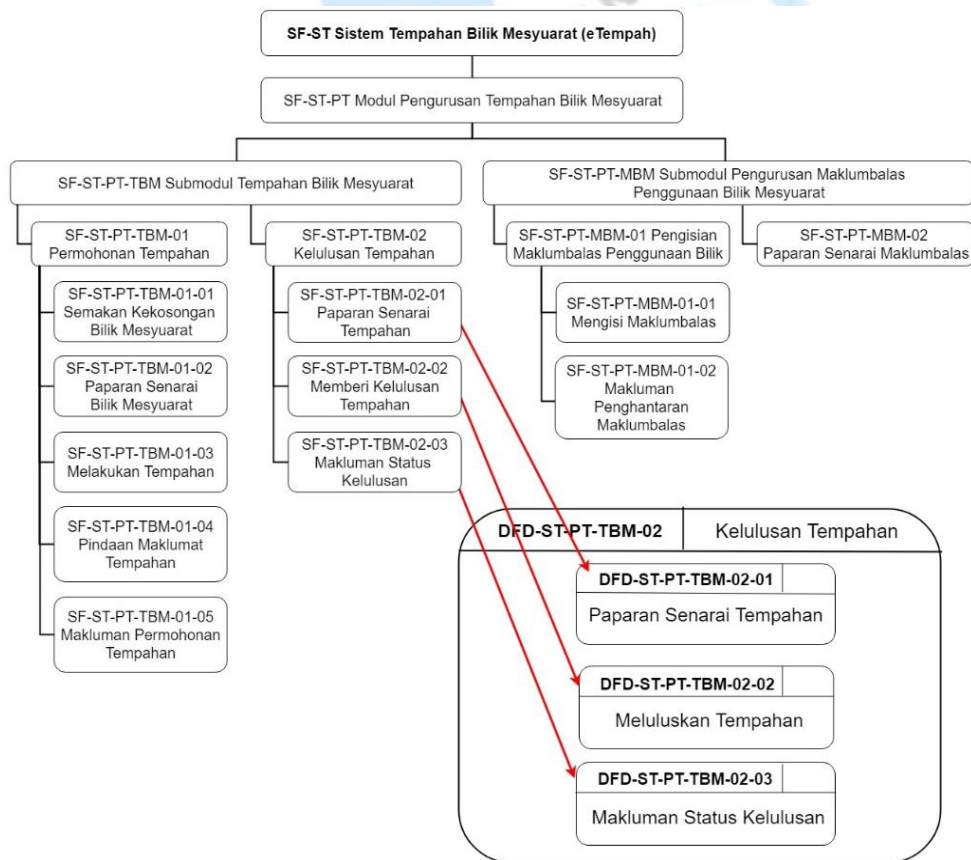
- d) Hubungkan antara entiti luaran dan fungsi melalui penggunaan notasi aliran maklumat. Hubungan di antara entiti luaran dan fungsi serta input dan output yang terlibat boleh dianalisis daripada keterangan di dalam Definisi Fungsi Bisnes yang dihasilkan melalui Pemodelan Proses Bisnes [F1.4]. Contoh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



Rajah 52 : Contoh Rajah Konteks DFD

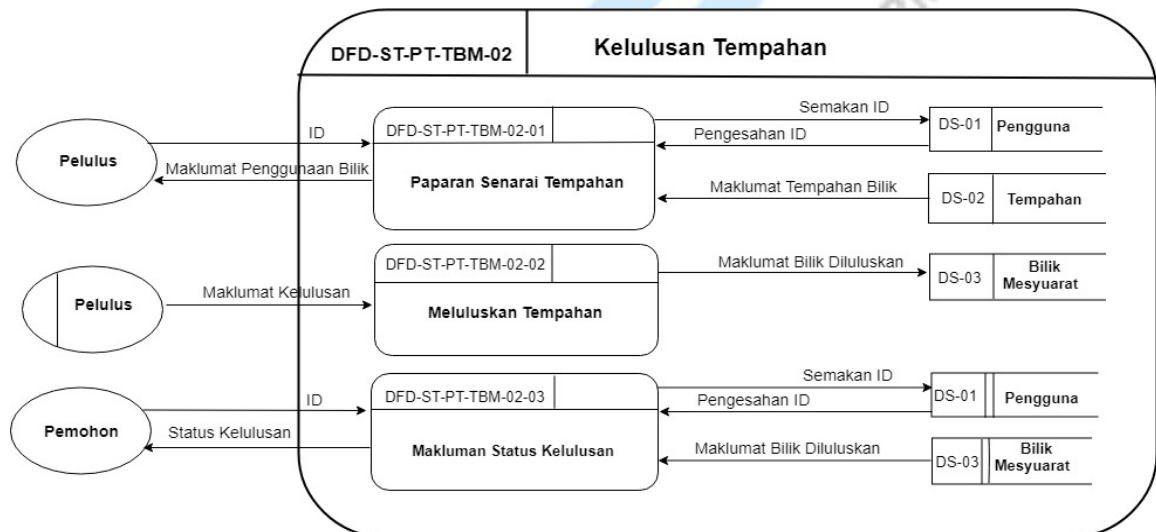
Langkah 2 : Menyediakan DFD Aras n

- 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

- a) 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 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.

Objektif

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

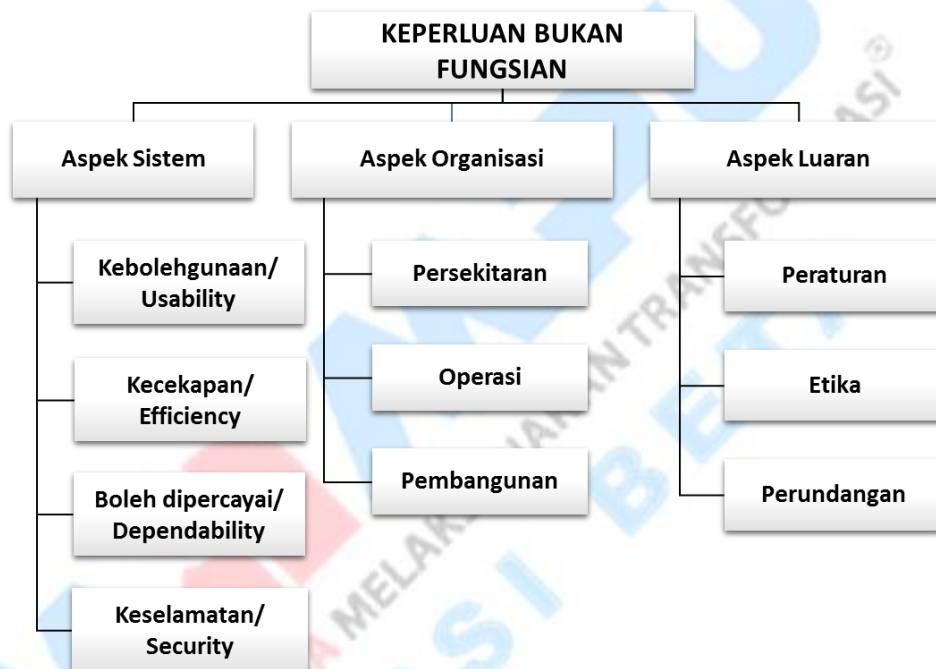
Langkah-langkah

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="456 477 1249 786"> <tr> <td data-bbox="456 477 576 535">BF</td> <td data-bbox="576 477 1249 535">Bukan Fungsian</td> </tr> <tr> <td data-bbox="456 535 576 723">XX</td> <td data-bbox="576 535 1249 723"> Klasifikasi bukan fungsian: AS – Aspek Teknikal AO – Aspek Organisasi AL – Aspek Luaran </td> </tr> <tr> <td data-bbox="456 723 576 786">99</td> <td data-bbox="576 723 1249 786">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"> <li data-bbox="469 925 1390 1077">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>. <li data-bbox="469 1111 1390 1182">ii) Aspek Organisasi Keperluan yang meliputi persekitaran, operasi dan pembangunan <li data-bbox="469 1216 1390 1288">iii) Aspek Luaran Keperluan yang meliputi peraturan, akta dan perundangan 						
Catatan	<p>Catatkan 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 Reka bentuk, 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)