

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Kajian pustaka yang digunakan pada penelitian ini merujuk pada penelitian sebelumnya yang berjudul "*Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*" yang dilakukan oleh Somayeh Salehi pada tahun 2015. Latar belakang dari penelitian tersebut adalah untuk mengeksplorasi dampak strategi e-CRM pada loyalitas pelanggan sebuah bank. Hal ini didasari dengan adanya fakta bahwa loyalitas pelanggan merupakan sebuah hal penting dalam sebuah organisasi komersil atau penyedia layanan, selain itu terdapatnya perubahan dan perkembangan teknologi yang semakin baik. Penelitian ini adalah penelitian *descriptive* dimana menggunakan metode *correlation survey*, dimana penelitian dilakukan dengan memberikan kuisisioner secara acak kepada Esfahan Branch's of Sepah Bank, Iran. Hasil dari penelitian tersebut adalah strategi CRM khususnya e-CRM merupakan sebuah strategi kuat yang muncul di lingkungan yang kompetitif seperti dunia perbankan. Implementasi e-CRM oleh bank akan mengarah pada peningkatan kepuasan pelanggan dan kepuasan pelanggan merupakan faktor penentu dalam loyalitas pelanggan.

Daris Arifin dan Johan Setiawan melakukan penelitian yang berjudul "Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana" pada tahun 2005. Latar belakang dari penelitian ini adalah PT Essensilindo Perdana yang merupakan distributor biskuit khong guan dan monde merasa perlu memiliki sebuah e-CRM untuk meningkatkan loyalitas dan *profitable customer* yang dimiliki. Penelitian dilakukan dengan menggali informasi dari bagian pemasaran dan penjualan mengenai apa saja yang dibutuhkan oleh pelanggan. Secara keseluruhan penelitian bersifat analisi perancangan, dimana hasil penelitian hanya sampai pada tahapan rancangan sistem. Terdapat 3 fitur utama yang direkomendasikan oleh peneliti yaitu fitur buku tamu dan tanya jawab, belanja online, dan rekomendasi produk. Setelah dibuat sebuah rancangan sistem, rancangan tersebut dievaluasi yang tujuannya untuk mengetahui kepuasan calon pengguna terhadap sistem yang akan dibuat. Evaluasi dilakukan dengan metode kuisisioner yang di dalamnya terdapat 3 komponen uji yaitu tampilan sistem, kelengkapan fitur dan kemudahan memperoleh informasi. Hasil dari penelitian tersebut adalah e-CRM mampu menjadi sebuah solusi tepat untuk memenuhi kebutuhan pelanggan, fitur – fitur yang ditawarkan dapat mempermudah pelanggan mendapatkan informasi sehingga meningkatkan efisiensi dan loyalitas.

Sebuah penelitian yang berjudul "*Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers*" oleh Edy Paryitno dan Novita Amylia pada tahun 2017. Penelitian dilakukan pada sebuah layanan kesehatan hewan yaitu UPT Poliklinik Hewan Yogyakarta. Tujuan dari penelitian ini adalah untuk mengetahui dampak positif penerapan aplikasi CRM pada sebuah poliklinik hewan. Dalam penelitian ini terbagi menjadi 3 tahap utama yaitu analisis

kebutuhan, perancangan, dan implementasi. Setelah tahapan implementasi dari sistem, akan dilakukan evaluasi terhadap penerapan CRM. Hasil akhir dari penelitian tersebut adalah CRM secara menyeluruh membantu dalam layanan, terdapat perbaikan waktu tunggu sebesar 82%, meningkatkan layanan sebesar 50%.

Ghaniyu Zulfata melakukan penelitian yang berjudul "*Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM*" yang dilakukan oleh pada tahun 2017. Penelitian berdasar pada kebutuhan organisasi untuk mempertahankan pelanggannya ditengah persaingan yang ketat. Tujuan dari penelitian ini adalah untuk mengembangkan sebuah aplikasi yang berdasar pada strategi CRM dengan menggunakan *framework of Dynamic CRM*. *Dynamic CRM* digunakan pada tahapan analisis kebutuhan fungsional. Pengujian sistem yang telah dibuat meliputi pengujian *white-box*, *black-box*, *compability* dan *user acceptance testing*. Hasil dari penelitian tersebut adalah semua kebutuhan sistem dapat dideskripsikan sesuai dengan 3 fase dalam *dynamic CRM* sehingga menghasilkan sebuah aplikasi dengan tingkat penerimaan pengguna mencapai 81,9%.

Pada penelitian lainnya yang berjudul "*Analisis dan Implementasi Perancangan Metode Rational Unified Process pada layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi*" oleh Muhammad Taufik pada tahun 2015. Bank Mandiri yang merupakan bank salah satu bank terbesar di Indonesia masih menerapkan sebuah sistem yang bersifat manual untuk layanan *save deposit box* (SDB). Layanan SDB memiliki peran yang cukup vital karena dapat dijadikan sebagai tolak ukur pertumbuhan dana nasabah. Pengembangan menggunakan metode RUP dipilih untuk menciptakan sebuah sistem yang sesuai dengan kebutuhan. Dalam pengembangan sistem hanya menerapkan 3 dari total 4 fase RUP dan pengujian sistem menggunakan metode *product matric*. Hasil dari penelitian tersebut adalah metode RUP menciptakan sistem yang sesuai dengan kebutuhan dan berkualitas dengan pengujian yang baik.

Ardhani Dwi Kinasih dalam penelitian yang berjudul "*Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)*" tahun 2017. Landasan dari penelitian ini adalah mengembangkan sebuah sistem evaluasi pembelajaran PAUD untuk menyelaraskan dengan tuntutan kurikulum 2013. Dalam pengembangan sistem tersebut peneliti menggunakan metode *Rational Unified Process (RUP)* karena dapat mengakomodasi perubahan kebutuhan namun tidak berakibat pada penundaan waktu proyek. Metode RUP diterapkan secara menyeluruh meliputi keempat fasenya. Pengujian sistem menggunakan beberapa metode, kebutuhan fungsional menggunakan *unit testing* dan *integration testing* sedangkan *compatibility testing* untuk kebutuhan non fungsional. Hasil dari penelitian ini adalah RUP membantu pengembangan sistem yang baik ditandai dengan hasil pengujian *cyclomatic complexity* kurang dari 10, skor *SUS* sebesar 84,9 dan *compatibility* yang menunjukkan sistem dapat berjalan pada semua perangkat *Android* diatas versi 2.3.

Table 2.1 Kajian Pustaka

No	Judul Penelitian	Tujuan Penelitian	Metode	Kesimpulan
1	Somayeh Salehi, <i>Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)</i> . IEEE 2015	Mengetahui dampak strategi e-CRM terhadap loyalitas pelanggan pada organisasi perbankan.	<i>Correlation Survey, Customer Relationship Management</i>	Implementasi e-CRM oleh bank akan mengarah pada peningkatan kepuasan pelanggan dan loyalitas pelanggan.
2	Daris Arifin, "Rancangan Aplikasi e-CRM pada PT Essensiindo Perdana". <i>Soft Computing, Intelligent Systems and Information Technology</i> 2005.	Melakukan analisis dan perancangan aplikasi e-CRM berbasis website untuk meningkatkan loyalitas dan <i>profitable customer</i> .	<i>Customer Relationship Management</i>	Kemudahan yang diberikan oleh aplikasi e-CRM untuk mendapatkan informasi memberikan efisiensi dan layanan terhadap pelanggan.
3	Edy Paryitno dan Novita Amylia, "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers". IEEE. 2017	Mengetahui dampak positif penerapan strategi CRM pada layanan poliklinik hewan.	<i>Customer Relationship Management</i>	Penerapan CRM memberikan dampak berupa perbaikan waktu tunggu sebesar 82% dan meningkatkan layanan sebesar 50%.
4	Ghaniyu Zulfata, "Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM". J-PTIUK 2017	Melakukan analisis dan pengembangan sebuah sistem e-CRM untuk menghadapi persaingan dan mempertahankan pelanggan.	<i>Customer Relationship Management, framework of Dynamic CRM, Compatibility</i>	Pengembangan sistem e-CRM dapat diterima oleh calon penggunaanya dengan nilai sebesar 81,9%

No	Judul Penelitian	Tujuan Penelitian	Metode	Kesimpulan
			<i>Testing, User Acceptance Testing</i>	
5	Muhammad Taufik, "Analisis dan Implementasi Perancangan Metode Rational Unified Process pada Layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi". e-Proceeding of Engineering telkomuniversity.ac.id. 2015	Melakukan analisis dan pengembangan sistem save deposit box (SDB) menggunakan metode Rational Unified Process (RUP) pada Bank Mandiri.	<i>Rational Unified Process (RUP), Product Matrix</i>	Penggunaan metode RUP dapat membantu dalam pengembangan sistem yang sesuai dengan kebutuhan dan memiliki kualitas yang baik.
6	Ardhani Dwi Kinasih, "Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)". J-PTIHK. 2017	Melakukan analisis dan pengembangan sistem evaluasi pembelajaran PAUD menggunakan metode Rational Unified Process (RUP).	<i>Rational Unified Process (RUP), Unit Testing, Integration Testing, Compatibility Testing</i>	Metode RUP dapat mengakomodasi perubahan yang ada pada proses pengembangan sistem sehingga dapat menghasilkan sebuah sistem yang sesuai dengan kebutuhan.

Berdasarkan penelitian – penelitian diatas terdapat keterkaitan dengan penelitian yang akan dilakukan. Penelitian dengan judul *Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*, Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana, dan *Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers* memiliki keterkaitan dimana penelitian yang akan dilakukan menggunakan strategi CRM untuk meningkatkan layanan dan loyalitas pelanggan. Sedangkan penelitian yang berjudul “Pengembangan *Electronic Customer Relationship Management (E-CRM)* Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM” memiliki korelasi yaitu penggunaan metode pengujian UAT untuk menguji sistem dan strategi CRM yang akan diterapkan. Dua penelitian akhir yang berjudul “Analisis dan Implementasi Perancangan Metode *Rational Unified Process* pada layanan SDB dan Metode Pengujian *Product Metric* pada Bank Mandiri Cabang Palu Sam Ratulangi” dan “Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)” memiliki keterkaitan pada penggunaan metode RUP untuk mengembangkan sistem dengan baik.

2.2 Koperasi

Koperasi berasal dari kata latin “*coopere*” atau dalam bahasa inggris “*cooperation*”. Sehingga secara etimologi koperasi mempunyai arti “kerja sama” atau sebuah kegiatan yang dilakukan secara bersama – sama dan mempunyai tujuan yang sama. Koperasi juga dapat diartikan sebagai suatu perkumpulan yang beranggotakan perseorangan atau badan hukum yang memberikan kebebasan kepada anggotanya untuk masuk atau keluar. Setiap anggota bekerja sama secara kekeluargaan menjalankan usaha untuk memperbaiki kesejahteraan jasmaniah anggotanya (Sitio & Tamba, 2001).

Secara internasional oleh *ILO (International Labour Organization)* mendefinisikan koperasi sebagai perkumpulan orang yang secara suka rela bergabung untuk bekerja sama dalam mendapatkan keuntungan dan menanggung resiko bersama. Dari definisi tersebut terdapat 6 elemen dasar dari koperasi yaitu perkumpulan, keanggotaan sukarela, adanya tujuan ekonomi, suatu organisasi bisnis, keadilan dan keseimbangan.

Bapak koperasi Indonesia yaitu Moh. Hatta mendefinisikan koperasi sebagai usaha bersama untuk memperbaiki perekonomian berdasarkan semangat tolong menolong. Indonesia sendiri mengatur organisasi koperasi dalam UU no. 25 tahun 1992 tentang Perkoperasian. Dalam undang – undang tersebut terdapat tujuan koperasi secara umum yaitu memajukan kesejahteraan anggota dan masyarakat pada umumnya serta ikut memanjudkan perekonomian nasional.

2.3 KPRI “GURU” Kecamatan Sumbermanjing Wetan

KPRI atau Koperasi Pegawai Republik Indonesia sebelumnya bernama Koperasi Pegawai Negeri (KPN) sudah menjadi wadah bagi pegawai negeri khususnya bagi guru di Kecamatan Sumbermanjing Wetan sejak tahun 1968.

Koperasi KPRI “GURU” Sumbermanjing Wetan mempunyai area yang luas terdiri dari 63 instansi pendidikan yang tergabung di dalamnya dengan jumlah anggota lebih dari 426 orang. Cakupan wilayah KPRI seluas 27.218,49 Ha dan terbagi menjadi 14 kecamatan dengan kondisi wilayah yang berbukit.

KPRI mempunyai mempunyai beberapa jenis unit antara lain unit simpan pinjam, unit toko dan unit transportasi. Unit simpan pinjam merupakan unit yang menyediakan layanan bagi anggota berupa jasa simpan atau tabung dan jasa pinjam atau kredit. Unit toko merupakan salah satu usaha KPRI yaitu menyediakan peralatan tulis sedangkan unit transportasi memberikan jasa penyewaan kendaraan. Berdasarkan pada asas keterbukaan dan kekeluargaan yang diterapkan oleh koperasi maka segala informasi mengenai koperasi dapat didapatkan oleh seluruh anggota. Termasuk dalam penetapan pengurus, setiap tahunnya diadakan pemilihan yang pencalonannya dari anggota dan dipilih oleh anggota.

2.4 Simpan Pinjam

Kegiatan simpan pinjam dilakukan oleh unit simpan pinjam yang didirikan dengan tujuan untuk meningkatkan kesejahteraan anggota koperasi, serta memiliki peran aktif dalam upaya meningkatkan kualitas kehidupan ekonomi (Sitio & Tamba, 2001). Usaha simpan pinjam merupakan salah jenis usaha yang telah dikenal secara luas oleh anggota koperasi dan masyarakat di Indonesia. Usaha ini adalah salah satu usaha lembaga keuangan non bank dilakukan untuk menghimpun dana dan menyalurkannya dari dan untuk anggotanya.

Pada jenis usaha ini dua faktor utamanya adalah simpanan (tabungan) dan pinjaman (piutang) itu sendiri. Pinjaman merupakan semua jenis pinjaman yang harus dibayar kembali bersama bunganya oleh peminjam sesuai dengan perjanjian yang telah disepakati. Pinjaman juga dapat didefinisikan sebagai bentuk kepercayaan memberikan atau menitipkan baik berupa barang, jasa, atau uang dari suatu pihak ke pihak lainnya, dimana pihak yang menerima harus mengembalikan dan memberikan balas jasa atas hal tersebut (Sitio & Tamba, 2001).

Sedangkan simpan atau tabungan adalah sebagian pendapatan yang diterima seseorang tidak semua digunakan untuk konsumsi, melainkan sebagian akan disimpan sebagai simpanan (Sitio & Tamba, 2001). Dalam koperasi simpan juga dapat berarti sebagai iuran yang telah disepakati. Terdapat berbagai jenis simpanan dalam sebuah koperasi yang memiliki fungsi yang berbeda-beda. Jenis – jenis simpanan dalam koperasi simpan pinjam antara lain simpanan pokok, wajib, berjangka dan lain sebagainya yang dapat ditentukan berdasarkan pada kesepakatan antar anggotanya.

2.5 Customer Relationship Management (CRM)

Customer Relationship Management (CRM) merupakan proses pengelolaan semua aspek interaksi yang dimiliki oleh perusahaan dengan

pelanggannya. Aspek – aspek tersebut meliputi prospeksi, penjualan dan layanan. CRM membantu untuk memberikan wawasan dan memperbaiki hubungan perusahaan dengan menggabungkan semua perspektif (Buttle, 2009).

CRM juga dapat di artikan sebagai sebuah strategi pendekatan untuk mengidentifikasi, mengakusisi dan mempertahankan pelanggan. Secara tidak langsung CRM dapat membantu untuk memaksimalkan setiap interaksi pelanggan. Hal ini dikarenakan CRM merupakan sebuah strategi bisnis inti yang mengintegrasikan proses dan fungsi internal dan eksternal untuk menciptakan nilai bagi pelanggan berdasarkan pada data berkualitas tinggi yang dimungkinkan oleh teknologi informasi.

Konsekuensi dari perkembangan dunia bisnis adalah setiap organisasi harus bersaing baik di pasar domestik maupun global. Cara untuk memenangkan persaingan, bisa dilakukan dengan cara mempererat hubungan antara pemangku kepentingan, pelanggan, distributor, karyawan dan lingkungan sekitar lainnya. Perusahaan harus merubah orientasi transaksi atau hubungan jangka pendek menjadi membangun hubungan jangka panjang (Payne, 2005).

2.5.1 Fase *Customer Relationship Management* (CRM)

Berdasarkan strategi untuk mengelola pelanggan, CRM dibagi menjadi tiga fase, yaitu: (Kalakota & Robinson, 2001)

1. *Acquire* (Mendapatkan pelanggan)

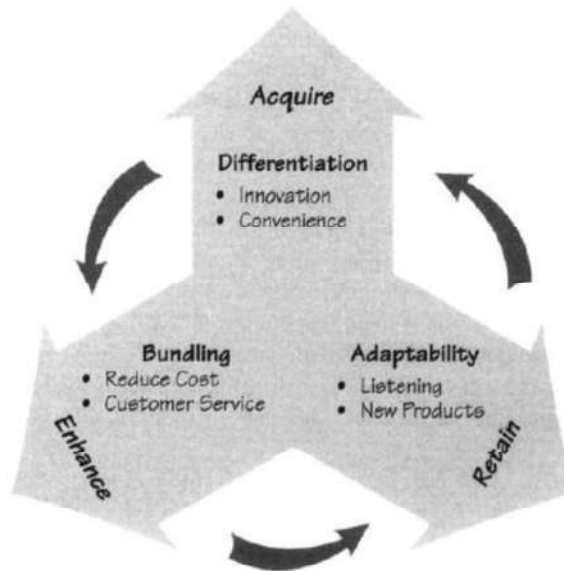
Fase awal dalam hubungan dengan pelanggan adalah mendapatkan kepercayaan pelanggan itu sendiri, untuk mendapatkan pelanggan baru organisasi harus mempromosikan produk atau jasa yang inovatif. Selain itu organisasi harus memberikan gambaran bagaimana mereka memperlakukan pelanggan dengan baik sehingga dapat menarik hati calon pelanggannya. Faktor penting lainnya adalah menawarkan produk atau jasa yang lebih unggul dari kompetitor.

2. *Enhance* (Meningkatkan keuntungan dari pelanggan yang sudah ada)

Organisasi dituntut untuk mampu menciptakan hubungan yang erat dengan para pelanggan. Pada fase ini organisasi harus mendengarkan berbagai keluhan dan meningkatkan kualitas pelayanan dari pelanggan yang telah ada. Selain itu organisasi juga dapat menerapkan strategi *cross-selling* atau *up-selling* yang dapat meningkatkan pendapatan dan mengurangi biaya (*reduce cost*).

3. *Retain* (Mempertahankan pelanggan)

Mempertahankan pelanggan membutuhkan pengertian yang baik tentang apa yang dibutuhkan oleh pelanggan. Pada fase ini fokus utama organisasi ada pada pelayanan, organisasi dapat memenuhi apa yang dibutuhkan oleh pelanggan. Menerima segala keluhan atau saran dari pelanggan agar dijadikan sebagai bahan evaluasi untuk meningkatkan kualitas pelayanan yang ada serta memberikan kemudahan – kemudahan untuk memanjakan pelanggan.



Gambar 2.1 Fase Customer Relationship Management

Sumber : Kalakota & Robinson (2001)

2.5.2 Jenis Customer Relationship Management (CRM)

Terdapat empat tipe CRM yang mendukung kerjanya strategi CRM yaitu (Buttle, 2009):

a. *Strategic CRM*

Strategic CRM memiliki fokus utama untuk mempertahankan pelanggan yang loyal. Dengan cara memberikan *value* atau nilai yang lebih baik dibandingkan dengan para pesaing. Pendekatan yang sering digunakan pada *strategic CRM* antara lain pendekatan terhadap produk, produksi dan penjualan.

b. *Operasional CRM*

Operasional CRM memiliki fokus utama kepada bagaimana untuk mengotomatisasi dan meningkatkan proses bisnis yang ada untuk mendukung pelayanan terhadap pelanggan. Fungsi aplikasi CRM ini dapat diterapkan pada pemasaran, penjualan maupun layanan. Dalam *operasional CRM* terdapat 3 jenis aplikasi utama yaitu *marketing automation*, *sales force automation*, dan *service automation*.

c. *Analytical CRM*

Analytical CRM memiliki fokus untuk menyimpan dan mengolah data untuk mendapatkan data yang berkaitan dengan peningkatan nilai pelanggan. Dasar dari analytical CRM adalah data dari pelanggan seperti data pembelian, pembayaran dan jasa yang digunakan, selain itu juga dikombinasikan dengan data eksternal perusahaan seperti gaya hidup dan demografi. Tujuan utamanya adalah untuk menganalisa pelanggan mana yang layak untuk dipertahankan.

d. *Collaborative CRM*

Collaborative CRM berfokus dalam penggunaan aplikasi CRM untuk berkomunikasi melampaui batas organisasi. Penggunaan teknologi di seluruh jajaran organisasi ini dengan tujuan untuk mengoptimalkan nilai perusahaan, partner dan pelanggan. Dengan penggunaan *collaborative CRM*, memungkinkan untuk menangani perbedaan jenis aplikasi dipakai oleh pelanggan, seperti *email*, telepon, *fax* dan *website*.

2.6 Operational CRM

Operasional Customer Relationship Management (CRM) lebih terfokus pada otomatisasi cara perusahaan dalam hubungan dengan pelanggannya. Berbagai aplikasi perangkat lunak *Customer Relationship Management (CRM)* memungkinkan untuk mengubah fungsi pemasaran, penjualan, dan pelayanan dapat berjalan secara otomatis. Beberapa aplikasi terpenting yang dapat diterapkan dalam operasional Customer Relationship Management (CRM), yaitu (Buttle, 2009):

a. *Marketing Automation (MA)*

Marketing automation menerapkan teknologi untuk mendukung pemasaran secara otomatis. Proses otomatisasi berdasar pada data pelanggan untuk mengembangkan, melaksanakan dan mengevaluasi komunikasi dan penawaran. Manfaat dari *marketing automation* saat ini ketika banyak sekali *platform* pemasaran yang mampu menarik pelanggan menjadi lebih mudah untuk dilakukan.

b. *Sales Force Automation (SFA)*

Sales force automation menerapkan teknologi untuk membantu perusahaan dalam kegiatan penjualan. Software SPA dapat dikonfigurasi sehingga pada proses penjualan dapat diaplikasikan pada banyak industri atau organisasi. SFA dapat membantu perusahaan karena perusahaan secara otomatis dapat menetapkan prospek dan melacak peluang.

c. *Service Automation (SA)*

Service automation adalah penerap aplikasi teknologi komputer untuk mendukung layanan perusahaan baik layanan langsung maupun layanan pendukung. Penggunaan *service automation* memungkinkan perusahaan untuk mengelola pelayanan kepada pelanggan secara otomatis. Otomatisasi layanan biasanya digunakan untuk mengatur dan menampung keluhan pelanggan yang mempunyai masalah dengan sebuah produk atau layanan.

Pada operasional CRM selain dibagi menjadi 3 jenis di atas terdapat 2 tipe aplikasi yaitu: (Rainer, 2010).

a. *Customer facing application*

Customer facing application merupakan aplikasi operasional CRM yang dibuat untuk mendukung para staf perusahaan atau organisasi dalam

melayani pelanggan. Fungsi dari aplikasi ini lebih bersifat sebagai alat bantu ketika menghadapi atau memberikan pelayanan terhadap pelanggan. Penggunaan tipe aplikasi ini memberikan kemudahan serta meningkatkan kualitas dari layanan yang diberikan.

b. Customer touching application

Customer touching application merupakan aplikasi yang langsung berinteraksi dengan pelanggan. Aplikasi itu dapat bekerja apabila ada permintaan layanan dari pelanggan, otomatis secara periodik, atau jika *event* tertentu dari perusahaan. Sehingga dengan adanya aplikasi ini dapat menggantikan perwakilan atau staf perusahaan untuk berinteraksi dengan pelanggan.

2.6.1 Service Automation (SA)

Service Automation (SA) atau otomatisasi layanan memungkinkan perusahaan untuk mengotomatisasi fungsi layanan perusahaan terhadap pelanggan melalui *call center*, *contact person*, *helpdesk*, layanan berbasis web atau bertatap muka langsung dengan pelanggan di lapangan. *Service Automation (SA)* juga menerapkan teknologi informasi internet khususnya web, melalui web suatu perusahaan dapat memberikan pelayanan sesuai keinginan pelanggan. Manfaat *Service Automation (SA)* bagi perusahaan dan pelanggan yaitu pelayanan kepada pelanggan lebih efektif dan efisien, mengurangi biaya layanan, meningkatkan kualitas layanan, dan meningkatkan produktivitas layanan ke pelanggan.

Adapun macam-macam aktivitas pada modul *Service Automation (SA)* yaitu aktivitas manajemen, agen manajemen, kasus tugas, manajemen kasus, manajemen kontrak, *customer self-service*, *email respon* manajemen, eskalasi, *inbound* komunikasi manajemen, *outbound* komunikasi manajemen, antrian dan *routing*, penjadwalan, *scripting*, analisis layanan, layanan manajemen tingkat, manajemen *resource*, web kolaborasi, dan alur kerja teknik (Buttle, 2009). Sedangkan *Customer Self-Service* adalah layanan berbasis dimana pelanggan dapat melayani dirinya sendiri untuk melakukan transaksi, pemesanan, melaporkan pemesanan secara online dengan fasilitas web yang telah disediakan perusahaan sesuai dengan kebutuhan pelanggan (Adrian, 2005).

2.6.2 Customer-Touching Applications

Customer Touching Applications merupakan salah satu bagian dari CRM. Pada operasional CRM salah satu bagian penting adalah *customer touching applications* karena merupakan sebuah aplikasi yang mampu membantu pemberian layanan secara otomatis. *Customer Touching Applications* sendiri sering disebut sebagian *E-CRM* atau pendekatan CRM yang memanfaatkan teknologi untuk berinteraksi kepada pelanggannya guna memberikan pelayanan yang lebih baik. *Customer Touching Applications* diartikan teknologi atau aplikasi yang mampu membuat pelanggan untuk langsung berinteraksi dengan organisasi. Dengan aplikasi tersebut pelanggan bias membantu kebutuhan dirinya sendiri (Raine, 2010)

2.7 Proses Bisnis

Proses bisnis merupakan sebuah kumpulan aktivitas dimana di dalamnya terdapat satu atau lebih *input* dan menghasilkan sebuah *output* seperti laporan yang nantinya bernilai bagi pelanggan (Monk & Wagner, 2013). Pelanggan atau *customer* dalam hal ini terbagi menjadi 2 yaitu *internal customer* dan *external customer*. *Internal customer* didefinisikan sebagai orang yang membeli atau menggunakan produk akhir, sedangkan *external customer* merupakan orang dari bagian lain dalam sebuah organisasi yang sama.

Dalam sebuah organisasi proses bisnis merupakan inti dari desain organisasi yang berorientasi pada proses. Berbeda dengan struktur organisasi yang membagi sebuah organisasi secara parsial tetapi proses bisnis lebih berkaitan dengan pelaksanaan tugas – tugas serta koordinasi dari beberapa aspek (Becker, et al., 2013). Contoh dari sebuah proses bisnis adalah pemesanan pada pabrik, proses penjualan *retail* dan persetujuan kredit pada bank.

2.7.1 Permodelan proses bisnis *as-is*

Permodelan proses bisnis *as-is* adalah langkah untuk mengumpulkan dan menggambarkan proses bisnis aktual pada sebuah organisasi. Permodelan ini tidak hanya berfungsi untuk menggambarkan proses bisnis yang saat ini, tetapi juga untuk mengetahui masalah atau kekurangan dari proses bisnis sehingga dapat memunculkan gagasan untuk meningkatkan sebuah proses bisnis dalam sebuah organisasi (Becker, et al., 2013).

Pada permodelan proses bisnis *as-is* informasi yang digunakan untuk menggambarkan kondisi aktual dari sebuah proses bisnis sebuah organisasi dapat menggunakan metode observasi atau wawancara. Observasi dilakukan untuk mengetahui secara nyata apa saja yang terjadi dalam sebuah proses bisnis. Sedangkan wawancara dapat dilakukan untuk mengetahui informasi yang lebih detail (Przybyłek, 2011).

2.7.2 Permodelan proses bisnis *to-be*

Permodelan proses bisnis *to-be* merupakan sebuah permodelan untuk menggambarkan hasil *improvement* dari proses bisnis yang ada saat ini (*as-is*). Proses bisnis *to-be* harus dibuat berdasarkan pada proses bisnis *as-is* dan harus disetujui oleh pemangku kepentingan. Dalam penerapannya perubahan proses bisnis *as-is* ke *to-be* dapat dalam jangka waktu yang singkat atau lama (Becker, et al., 2013).

Pada permodelan bisnis *to-be* peran pemangku kepentingan sangat penting hal ini dikarenakan dalam perbaikan proses bisnis pemangku kepentingan dapat mengetahui keadaan dari organisasinya. Terdapat dua langkah utama dalam proses permodelan *to-be*, yang pertama adalah melakukan identifikasi peluang dari perbaikan yang ada. Hal ini untuk mencari proses yang dapat dikurangi atau diotomatisasi. Langkah selanjutnya adalah menentukan mana proses bisnis dasar

dan sejauh mana kita dapat melakukan otomatisasi dari proses yang ada (Przybytek, 2011).

2.8 Business Process Modelling Notation (BPMN)

Business Process Modelling Notation (BPMN) adalah grafis untuk menggambarkan langkah – langkah logis proses bisnis beserta aktor yang terlibat didalamnya. *Business Process Modelling Notation* (BPMN) juga merupakan sebuah standart notasi yang telah disepakati untuk menggambarkan proses bisnis. BPMN bertujuan untuk memberi notasi yang dapat dipahami oleh semua pengguna bisnis, mulai dari analisis bisnis yang membuat rancangan, implementator, serta orang bisnis yang mengelola dan mengontrol proses tersebut (Silver, 2012).


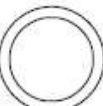
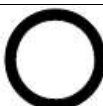
Didalam BPMN terdapat *Business Process Diagram (BPD)* yang berfungsi untuk menggambarkan proses bisnis. BPD pada dasarnya sama dengan *flowchart*, yang dirancang untuk menyajikan urutan dari semua aktifitas yang terjadi selama suatu proses. BPD merupakan diagram yang dirancang untuk membantu analisis dalam merancang, mengendalikan dan mengelola proses.

2.8.1 Flow Objects

1. Event

Event digambarkan dalam bentuk lingkaran yang merupakan suatu hal yang dapat mempengaruhi proses bisnis. Terdapat tiga jenis *event* yaitu *start event*, *intermediate event*, dan *end event* yang dibedakan berdasarkan aliran prosesnya. Dari ketiga jenis *event* ini dapat menangkap (*catch*) atau melempar (*throw*) *event*. (Object Management Group, 2011)

Table 2.2 Tipe Event dalam BPMN


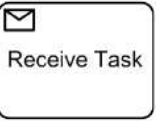


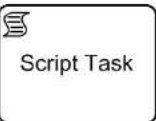
No	Tipe	Diskripsi	Simbol
1	<i>Start</i>	Mendeskripsikan dimana awal dari suatu proses.	
2	<i>Intermediate</i>	Mendeskripsikan dimana sesuatu terjadi diantara awal dan akhir proses. Dimana hal itu akan mempengaruhi alur dari proses, tapi tidak akan memulai atau menghentikan proses tersebut.	
3	<i>End</i>	Mendeskripsikan suatu akhir dari sebuah proses	

Sumber : Object Management Group (2011)

2. Activity

Activity digambarkan dalam bentuk persegi dengan ujung yang membulat. Merupakan sebuah aktivitas yang ada dalam sebuah rangkaian proses bisnis. Dalam BPD activity dibagi menjadi beberapa jenis menurut jenis dari aktifitas itu sendiri. (Object Management Group, 2011)

Table 2.3 Tipe Activity dalam BPMN

No	Tipe	Diskripsi	Simbol
1	<i>Service Task</i>	<i>Service task</i> adalah aktivitas yang dikerjakan oleh perangkat lunak	
2	<i>Receive Task</i>	<i>Receive task</i> adalah sebuah aktivitas menerima pesan dari aktivitas lain. <i>Task</i> ini merupakan alternatif dari catching message event.	
3	<i>User Task</i>	<i>User task</i> dikerjakan oleh manusia. Berbeda dengan manual task setelah user task diselesaikan, engine menunggu hasil dari pekerjaan atau aktivitas yang diselesaikan untuk kemudian diproses.	
4	<i>Manual Task</i>	<i>Task</i> ini dieksekusi oleh manusia secara manual yang tidak mempengaruhi aktivitas yang sudah diselesaikan oleh <i>engine</i> pemrosesan	
5	<i>Script Task</i>	<i>Task</i> ini mengeksekusi <i>script</i> yang sudah didefinisikan dan bisa dijalankan oleh <i>engine</i> pemrosesan. Saat <i>script</i> selesai maka tugas juga selesai.	

Sumber : BPMN 2.0 Handbook Second Edition (2012)

3. Gateway

Gateway digambarkan dengan bentuk berlian dan berfungsi untuk mengontrol alur proses yang berlangsung. *Gateway* mengendalikan

pertemuan dan perpisahan pada sebuah alur proses. Pada gateway banyak *input* atau *output* dapat dipengaruhi oleh jenis *gateway* masing – masing. Pada BPD terdapat beberapa jenis dari yang menunjukkan fungsi sebagai berikut :

Table 2.4 Tipe Gateway dalam BPMN

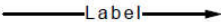
No	Tipe	Diskripsi	Simbol
1	<i>Exclusive</i>	Digunakan untuk membuat jalur dengan mengevaluasi keadaan kemudian mengarahkan kepada satu jalur yang sesuai	 or 
2	<i>Parallel</i>	Digunakan untuk melakukan percabangan proses dimana alur yang ada dijalankan secara bersamaan tanpa melihat proses sebelumnya.	
3	<i>Inclusive</i>	Digunakan ketika terdapat percabangan jalur dimana terdapat kondisi tertentu yang harus dipenuhi sehingga proses dapat diteruskan ke beberapa jalur.	
5	<i>Event-Based</i>	Digunakan ketika terdapat percabangan dimana bukan berdasarkan kondisi apa yang telah dipenuhi melainkan kondisi apa yang telah terjadi.	

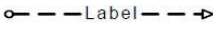
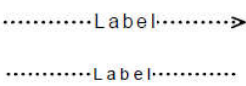
Sumber : www.lucidchart.com

2.8.2 Connection

Connecting object merupakan penghubung antar *flow object* satu dengan yang lainnya atau elemen - elemen lainnya. *Flow object* yang saling terhubung akan membentuk struktur dasar dari BPD. Terdapat tiga jenis *connecting flow* pada BPMN, yaitu *message flow*, *sequence flow* dan *association flow*.

Table 2.5 Tipe Connection dalam BPMN

No	Tipe	Diskripsi	Simbol
1	<i>Sequence flow</i>	Digunakan untuk menunjukkan sebuah urutan kegiatan yang akan dilakukan	



		dapat berupa aliran kondisional maupun <i>default</i>	
2	<i>Message flow</i>	Digunakan untuk menunjukkan pesan yang melintasi " <i>pool</i> " atau batasan organisasi.	
3	<i>Association flow</i>	Digunakan untuk menunjukkan hubungan antara <i>artifact</i> dengan aktifitas.	

Sumber : Object Management Group (2011)

2.8.3 *Swimlanes*

Swimlanes merupakan mekanisme untuk menggambarkan pembagian suatu tanggung jawab dari komponen atau aktor yang terlibat dalam sebuah proses bisnis. Ada yang *swimlanes* berfungsi untuk memperjelas penanggung jawab dari tiap aktifitas yang ada. Dalam BPMN terdapat 2 bentuk atau jenis dari *swimlanes* yaitu *lanes* dan *pool*.

Table 2.6 Tipe *Swimlanes* dalam BPMN

No	Tipe	Diskripsi	Simbol
1	<i>Pool</i>	<i>Pool</i> mempresentasikan tanggung jawab untuk aktivitas secara garis besar. <i>Pool</i> biasanya digunakan untuk menggambarkan sebuah organisasi.	
2	<i>Lanes</i>	<i>Lanes</i> merupakan sub bagian dari <i>pool</i> . <i>Lane</i> membagi <i>pool</i> menjadi sebuah hirarki sehingga dapat menggambarkan tanggung jawab secara lebih rinci seperti pembagian dalam sebuah organisasi	

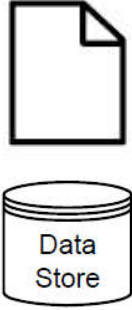
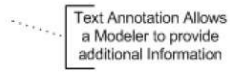

Sumber : Object Management Group (2011)

2.8.4 *Artifact*

Artifact merupakan informasi tambahan untuk memperjelas suatu proses bisnis. Dalam proses bisnis tidak mempengaruhi alur atau aktifitas yang ada. Dalam BPMN terdapat tiga jenis *artifact* yaitu *data object*, *group* dan *annotation*.

Table 2.7 Tipe *Artifact* dalam BPMN

No	Tipe	Diskripsi	Simbol
----	------	-----------	--------

1	<i>Data Object</i>	<i>Data Object</i> memrepresentasikan data berupa dokumen, e-mail atau surat. Selain itu terdapat <i>data store</i> merupakan tempat dimana proses dapat membaca atau menulis data.	
2	<i>Annotation</i>	Merupakan informasi tambahan untuk menunjukkan detail dari sebuah bagian proses bisnis.	
3	<i>Group</i>	Merupakan pengelompokan <i>element</i> dari BPMN secara <i>informal</i> .	

Sumber : Object Management Group (2011)

2.9 Permodelan *Use Case*

Permodelan *use case* merupakan cara yang mudah untuk mendeskripsikan kebutuhan dari sebuah sistem. Permodelan ini dapat menggambarkan perilaku sistem ketika berinteraksi dengan aktor atau dengan sistem lainnya. Tujuan dari adanya permodelan *use case* adalah agar setiap kebutuhan fungsional dari sebuah sistem yang akan dikembangkan lebih mudah dan tepat (Bittner & Spence, 2002).

Bittner dan Spence juga berpendapat bahwa *use case* sangat berpengaruh karena mampu menunjukkan bagaimana sistem dapat memberikan nilai kepada pemangku kepentingan dan mengetahui siapa saja calon pengguna sistem. Banyak pengembangan perangkat lunak yang mendefinisikan secara detail *use case* tanpa mempertimbangkan pemangku kepentingan, tujuan, kebutuhan nyata, dan permasalahan yang ada. Hal semacam itu dapat berujung pada sebuah sistem yang tidak memenuhi atau sesuai dengan apa yang diharapkan. Sehingga terdapat beberapa tahap yang sebaiknya dilakukan dalam melakukan permodelan *use case*.

2.9.1 Identifikasi Tipe Pemangku Kepentingan

Pemangku kepentingan merupakan seseorang yang memperoleh dampak secara materil dari sistem yang akan dikembangkan. Gagasan tersebut mencerminkan fakta bahwa pemangku kepentingan merupakan individu yang secara langsung mendapatkan dampak dari sistem dan secara tidak langsung mempengaruhi sistem yang dikembangkan (Bittner & Spence, 2002). Identifikasi pemangku kepentingan merupakan langkah awal untuk mengetahui siapa saja pemangku kepentingan dari sistem yang akan dikembangkan. Hasil dari

identifikasi tersebut akan dikelompokkan berdasarkan karakteristik dan hubungan kepada sistem.

Table 2.8 Analisis Tipe Pemangku Kepentingan

Tipe Pemangku Kepentingan	Deskripsi	Pemangku Kepentingan
[Tipe pemangku kepentingan]	[Menjelaskan tipe pemangku kepentingan]	[Menyebutkan pemangku kepentingan yang termasuk dalam tipe pemangku kepentingan]

Sumber : Diadaptasi dari Bittner & Spence (2003)

2.9.2 Analisis Masalah

Analisis masalah merupakan sebuah proses pemahaman mengenai masalah sesungguhnya, kebutuhan pengguna, serta bagaimana kebutuhan tersebut dapat dipenuhi (Leffingwell & Widrig, 2003). Tujuan dari proses ini adalah untuk mengetahui pemahaman yang mendalam terhadap permasalahan sebelum pengembangan sistem dilakukan. Dalam analisis masalah identifikasi aktor merupakan hal yang penting, karena aktor yang terlibat dapat mengerti akar dari permasalahan yang ada. Cara terbaik untuk mendefinisikan analisis permasalahan adalah dengan membuat sebuah *problem statement* yang merupakan ringkasan dari permasalahan, dampak, dan solusi dari permasalahan (Bittner & Spence, 2002).

Table 2.9 Analisis Masalah

Masalah	Berisi diskripsi dari permasalahan
Mempengaruhi	Berisi pemangku kepentingan yang terpengaruh dengan adanya permasalahan
Dampak	Berisi dampak masalah terdapat pemangku kepentingan dan proses bisnis
Solusi	Berisi solusi yang ditawarkan untuk menyelesaikan permasalahan

Sumber : Diadaptasi dari Bittner & Spence (2003)

2.9.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Melakukan identifikasi kebutuhan dari pemangku kepentingan dan pengguna merupakan sebuah cara efektif untuk menyelesaikan permasalahan yang kompleks (Bittner & Spence, 2002). Setiap pemangku kepentingan akan memiliki pandangan berbeda mengenai permasalahan dan solusi yang dibutuhkan. Kebutuhan pemangku kepentingan memiliki prioritas untuk memastikan fokus dari pengerjaan proyek hal ini merupakan bagian paling penting dalam sistem, sehingga tim proyek dapat mengetahui bagian yang perlu dikerjakan terlebih dahulu. Salah satu teknik yang digunakan untuk prioritas

kebutuhan adalah aturan *MoSCow*. Aturan *MoSCow* merupakan aturan yang membagi kebutuhan menjadi empat prioritas yaitu (Bittner & Spence, 2002):

- a. *Must Have (Mo)*
Must Have memiliki arti bahwa suatu fitur dalam proyek harus diselesaikan dan dapat bekerja dengan baik ketika proyek selesai. Fitur tersebut penting karena merupakan solusi dari penyelesaian masalah atau fitur utama.
- b. *Should Have (S)*
Should Have memiliki arti bahwa suatu fitur tidak memiliki prioritas utama dalam pengerjaan sistem, tetapi penting dan memiliki nilai lebih bagi pengguna.
- c. *Could Have (Co)*
Could Have memiliki arti bahwa suatu fitur yang dapat meningkatkan pengguna sistem, namun ketika tidak dapat diimplementasikan tidak berpengaruh pada proyek.
- d. *Won't Have This Time (W)*
Won't Have This Time memiliki arti bahwa suatu fitur dapat diimplementasikan setelah pengembangan selesai

Table 2.10 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Kebutuhan	Pemangku Kepentingan	Prioritas	Situasi Saat ini	Solusi yang di tawarkan
Berisi diskripsi dari kebutuhan	Berisi pemangku kepentingan yang berperan	Berisi skala prioritas	Berisi diskripsi dari permasalahan yang ada pada saat ini	Berisi diskripsi dari solusi yang ditawarkan

Sumber : Diadaptasi dari Bittner & Spence (2003)

2.9.4 Identifikasi Fitur

Fitur adalah sebuah kemampuan sistem untuk menyediakan nilai bagi pengguna serta memenuhi kebutuhan pemangku kepentingan dan pengguna. Sebuah set fitur juga dapat menggambarkan ringkasan manfaat yang ditawarkan dari sistem yang dibangun (Bittner & Spence, 2003). Fitur harus dalam lingkup detail yang umum agar semua pemangku kepentingan dapat mengerti namun juga mampu menyediakan informasi bagi tim pengembang untuk membentuk, memvalidasi dan mengelola permodelan *use case*.

Table 2.11 Fitur Sistem

Kode Fitur	Fitur	Diskripsi	Prioritas
Kode setiap fitur	Fitur sistem	Diskripsi dari fitur	Prioritas fitur

Sumber : Bittner & Spence (2003)

2.9.5 Identifikasi Kebutuhan Fungsional dan Kebutuhan Non Fungsional

Kebutuhan fungsional merupakan layanan yang harus dimiliki sistem, kebutuhan fungsional juga merupakan pernyataan yang menyatakan bagaimana sistem merespon *input* tertentu dan bagaimana sistem tersebut berjalan dalam sebuah kondisi tertentu (Sommerville, 2011). Sebuah kebutuhan fungsional yang baik harus lebih spesifik dalam menggambarkan fungsi – fungsi sistem mencakup *input* dan *output* yang ada pada sistem tersebut.

Kebutuhan non fungsional merupakan sebuah kebutuhan yang tidak secara spesifik disampaikan oleh sistem kepada penggunaannya. Contoh dari sebuah kebutuhan nonfungsional antara lain *performance*, *security* atau *availability* (Sommerville, 2011). Namun kebutuhan non fungsional sangat berpengaruh pada sistem, terkadang jika kebutuhan non fungsional tersebut tidak terpenuhi maka sistem tidak dapat berjalan. Implementasi kebutuhan non fungsional lebih sulit karena bersifat menyeluruh dan terkadang saling berkaitan.

2.10 Unified Modeling Language (UML)


Unified Modeling Language (UML) merupakan sebuah notasi bagi *developer* atau pengembang untuk menggambarkan skenario dan merumuskan arsitektur sehingga mudah untuk dipahami. UML memiliki sifat yang sederhana namun ekspresif sehingga mempermudah pemeriksaan terdapat konsistensi dan kebenaran dari skenario yang dibuat. Saat ini *UML* merupakan bahasa pemodelan utama untuk menganalisa, menentukan dan merancang sistem perangkat lunak (Booch, 1998).



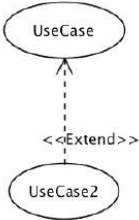
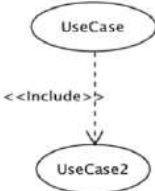
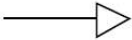
UML bahasa pemodelan terpadu yang berisikan notasi untuk permodelan dan pengembangan sistem berorientasi objek. UML menyediakan teknologi untuk mendukung pengembangan rekayasa perangkat lunak akan tetapi tidak menyediakan kerangka kerja proses untuk manajemen tim. *Unified Modelling Language (UML)* memiliki berbagai jenis diagram diantaranya *use case diagram*, *activity diagram*, *class diagram*, dan *sequence diagram* (Pressman, 2010).

2.10.1 Use Case Diagram

Use Case Diagram adalah permodelan dari kelakuan (*behavior*) sistem informasi yang akan dikembangkan. *Use case* mendiskripsikan suatu interaksi antara *actor* dengan sistem. Secara singkat *use case* digunakan untuk mengetahui fungsi – fungsi apa saja yang ada pada sebuah sistem dan siapa saja yang berhak menggunakan fungsi tersebut (Sukanto dan Shalahuddin, 2014).

Table 2.12 Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1		<i>Use case</i> fungsional yang disediakan oleh sistem dengan penamaan menggunakan kalimat kerja.

2	Actor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem yang dikembangkan. Penamaan <i>actor</i> menggunakan kata benda.
3	Association 	Komunikasi antara <i>actor</i> dan <i>use case</i> .
4	Extend 	<i>Extend</i> merupakan sebuah hubungan dengan menambahkan sebuah <i>use case</i> tambahan kepada sebuah <i>use case</i> . Dimana <i>use case</i> utama dapat berdiri sendiri tanpa <i>use case</i> tambahan dilakukan.
5	Include 	<i>Include</i> merupakan sebuah hubungan dengan menambahkan sebuah <i>use case</i> tambahan yang wajib untuk dijalankan ketika <i>use case</i> utama dilakukan.
6	Generalisasi 	Merupakan notasi hubungan antar aktor yang menghubungkan aktor yang lebih spesifik dengan aktor yang umum.

Sumber: Diadaptasi dari Object Management Group (2005)

Pada *use case diagram* tidak menyediakan detail informasi mengenai sebuah *use case* tertentu. Sehingga dalam sebuah *use case diagram* harus didukung dengan sebuah permodelan yang memiliki spesifikasi *use case*, dimana berisi penjelasan bagaimana aktor dan sistem berkolaborasi untuk mencapai tujuan yang diharapkan dari tiap *use case* tersebut.

Table 2.13 Skenario *Use Case*

<i>Use Case Name</i>	
<i>Brief Description</i>	Berisi penjelasan singkat dari sebuah <i>use case</i> .
<i>Actor</i>	Berisi aktor yang terlibat dalam <i>use case</i> .
<i>Pre-conditions</i>	Berisi satu atau lebih kondisi yang harus dipenuhi sebelum <i>use case</i> dapat dijalankan.



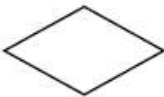


<i>Basic flow of Events</i>	Berisi alur normal yang dilakukan oleh <i>use case</i> .
<i>Alternative Flows</i>	Berisi alur opsional ketika sistem tidak berjalan sesuai dengan alur semestinya.
<i>Post-conditions</i>	Berisi satu atau lebih keadaan yang terjadi ketika <i>use case</i> telah berakhir

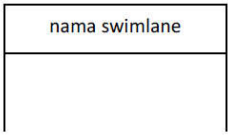
Sumber: Diadaptasi dari Bittner & Spence (2002)

2.10.2 Activity Diagram

Activity diagram menggambarkan sebuah aliran kerja atau *workflow* dari sebuah sistem. Sehingga *activity diagram* dapat menggambarkan aliran kerja dari sebuah aktifitas di dalam sistem dan siapa yang melakukan hal tersebut (Sukanto dan Shalahuddin, 2014).

Table 2.14 Simbol Activity Diagram

No	Simbol	Deskripsi
1	Status awal 	Status awal dari sebuah aktivitas sistem.
2	Aktivitas 	Aktifitas yang dilakukan didalam sistem yang biasanya diawali dengan kata kerja.
3	Percabangan/ <i>Decision</i> 	<i>Asosiasi</i> dimana jika terdapat percabangan pilihan 2 atau lebih aktivitas.
4	Penggabungan / Join 	<i>Asosiasi</i> penggabungan dari hasil perpecahan percabangan.
5	Status akhir 	Sebuah akhir atau hasil dari sebuah aktifitas dalam sebuah sistem.





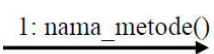
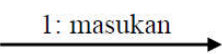
6	Swimlane 	Pemisah atau pembagi organisasi bisnis yang bertanggung jawab kepada aktifitas yang ada.
---	------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------

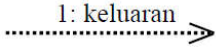
Sumber: Sukamto dan Shalahuddin (2014)

2.10.3 Sequence Diagram

Sequence diagram adalah sebuah notasi untuk menggambarkan interaksi objek pada *use case* dengan mendeskripsikan waktu hidup serta pesan yang dikirim antar objek. Untuk membuat *sequence diagram* hal terpenting adalah mengetahui objek – objek apa yang terlibat beserta metode – metode yang dimiliki (Sukamto dan Shalahuddin, 2014).

Table 2.15 Simbol dalam *Sequence Diagram*

No	Simbol	Deskripsi
1	Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem dikembangkan
2	Objek 	Menyatakan objek yang berinteraksi.
3	Waktu aktif 	Menyatakan sebuah objek dalam keadaan aktif dan berinteraksi
4	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek lain, digambarkan dengan arah panah mengarah pada objek yang dibuat.
5	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri.
6	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan <i>input</i> ke objek lainnya, digambarkan dengan arah panah mengarah pada objek yang menerima pesan.

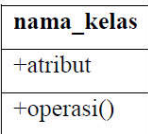
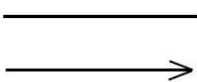
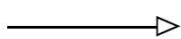

7	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian (<i>return</i>).
---	---------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------

Sumber: Sukamto & Shalahudin (2014)

2.10.4 Class Diagram

Class diagram berfungsi untuk menunjukkan keberadaan dan hubungan setiap *class* yang membentuk arsitektur sistem (Booch, et al., 1998). Terdapat dua elemen utama dalam pembentukan *class diagram* yaitu *class* dan *relationships*. Sukamto dan Shalahudin (2014) mengartikan *class diagram* sebagai notasi untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

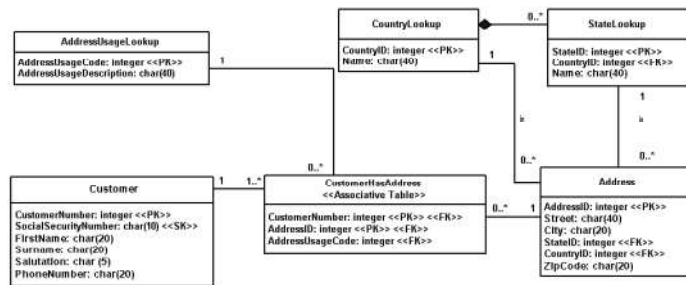
Table 2.16 Simbol Class Diagram

No	Simbol	Deskripsi
1	Kelas 	Kelas pada struktur sistem. Terdiri dari nama kelas, atribut dan operasi yang ada pada kelas tersebut.
3	Asosiasi / Association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi spesialisasi (umum khusus).
6	Aggregation 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

Sumber: Sukamto & Shalahudin (2014)

2.11 Physical Data Model

Physical data model (PDM) digunakan untuk merancang skema internal *database*. PDM menggambarkan tabel data, kolom data, dan hubungan antar tabel (www.agiledata.org). Dalam sebuah pengembangan sistem PDM sangat berguna karena merupakan acuan dalam pembuatan data base dari sistem tersebut.

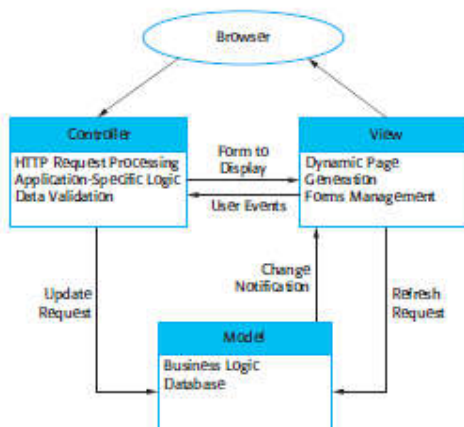


Gambar 2.2 Contoh *Physical Data Model*

Sumber: www.agiledata.org

2.12 Model-View-Controller

Model-View-Controller (MVC) merupakan salah satu pola arsitektur dimana membagi sebuah sistem menjadi 3 komponen logis yang saling berkaitan (Sommerville, 2011). Salah satu kelebihan dari pola MVC adalah independen sehingga mempermudah *custom* pada data yang ditampilkan. Dalam bukunya Sommerville menjelaskan bahwa ketiga komponen tersebut mempunyai fungsi yang berberda-beda.



Gambar 2.3 Skema *Model View Controller*

Sumber: Sommerville (2011)

Tiga komponen utama dalam MVC adalah *Model*, *View* dan *Controller*. Komponen *model* berfungsi untuk mengelola data pada sistem dan operasi terkait data tersebut. Didalam *model* terdapat logika bisnis atau fungsi untuk mengelola *data base*. Pada komponen *view* peran utamanya adalah menampilkan informasi kepada pengguna sistem. Setiap informasi yang ditampilkan pada komponen *view* dipengaruhi oleh data yang diperoleh dari komponen *model*. Sedangkan komponen *controller* memiliki peran untuk mengelola interaksi pengguna pada komponen *view*. Secara tidak langsung komponen *controller* merupakan jembatan antara presentasi dan data base.

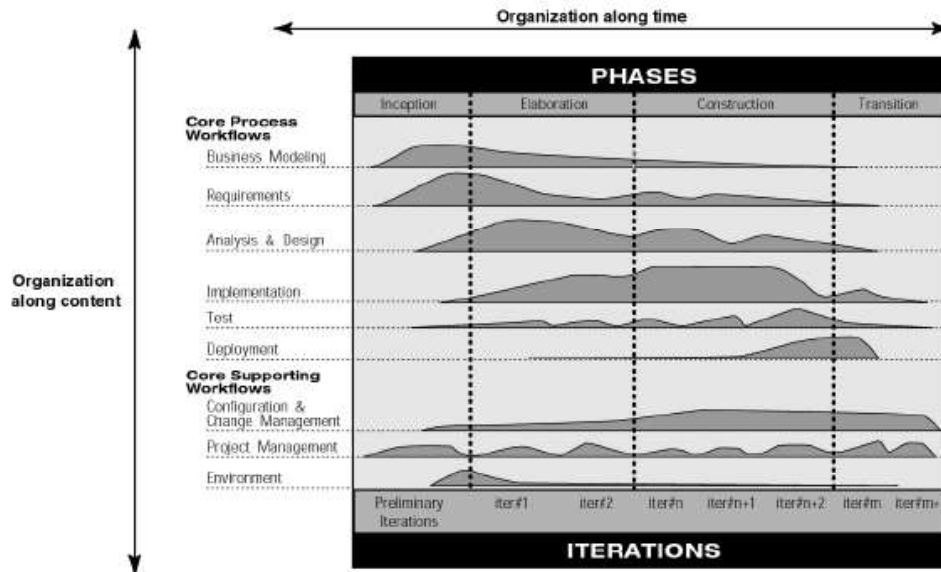
2.13 Rational Unified Process (RUP)

Rational Unified Process adalah sebuah metode dalam pengembangan perangkat lunak dengan menggunakan pendekatan *iterative* (perulangan). Metode ini mengusung pengembangan perangkat lunak dengan membagi menjadi beberapa fase. RUP menerapkan konsep *object-oriented* yang pada pengembangannya berfokus pada *Unified Model Language (UML)* (Jalote, 2008). *Iterative* dalam metode RUP merupakan pembeda dengan metode pengembangan lainnya, berbeda dengan *waterfall* dimana iteratif bersifat keseluruhan metode RUP mengakomodasi iteratif pada tiap fasenya. Selain itu pada RUP memisahkan tahapan dan aktifitas, dimana pada tiap fase terdapat aktifitas utama dan sub aktifitas dari fase lainnya.

Tujuan dari *Rational Unified Process (RUP)* adalah menciptakan sebuah perangkat lunak berkualitas tinggi yang memenuhi kebutuhan *end user* dalam jadwal dan anggaran yang dapat diprediksi. RUP yang bersifat *iterative* dibagi menjadi beberapa fase – fase yang mempunyai fokus tersendiri menekankan pada pembangunan sistem berdasarkan pemahaman secara keseluruhan tentang bagaimana sistem tersebut akan digunakan. Penggunaan *use case* dan *scenario* digunakan untuk menyelaraskan antara alur suatu proses dari kebutuhan yang didapat. Dalam segi pengembangan yang berfokus pada kebutuhan pengguna maka RUP merupakan salah satu metode pengembangan terbaik saat ini (Booch et al, 1998). Metode RUP memiliki karakteristik yang dapat membedakan dengan metode pengembangan lainnya antara lain :

1. *Architecture-Centric* yang berarti arsitektur sistem dijadikan sebagai patokan penentuan konsep, konstruksi, pengelolaan dan perbaikan sistem selama pengembangan berlangsung.
2. *Use Case driven* yang berarti penggunaan *use case* sebagai patokan terhadap perilaku sistem yang akan dikembangkan.
3. *Object-Oriented Techniques* yang berarti pengembangan berdasar pada konsep objek dan kelas yang saling berhubungan dan menggunakan UML sebagai notasi umumnya.
4. *Configurable Process* yang berarti metode ini dapat disesuaikan berdasarkan proyek yang dikembangkan.
5. *Quality Control And Risk Management* yang berarti metode ini dapat memberikan kontrol terhadap sistem yang dikembangkan dan resiko yang ada karena pada metode RUP semua kegiatan melibatkan pemangku kepentingan.

2.13.1 Fase – Fase pada Metode RUP



Gambar 2.4 Fase – Fase pada Metode RUP

Sumber : IBM (1998)

Fase – fase pada *Rational Unified Process (RUP)* digambarkan pada sumbu horizontal yang berkaitan dengan waktu. Pembagian tiap fase berdasarkan pada fokus aktifitas utama yang dikerjakan pada tiap fasenya. Akan tetapi RUP dapat mengakomodasi sub aktifitas seperti yang digambarkan pada gambar 2.4. Berikut merupakan fase – fase yang ada pada metode *Rational Unified Process (RUP)* (IBM,1998) :

1. Insepsi

Tahapan ini merupakan tahapan dimana terdapat penetapan permasalahan, *business case*, dan batasan ruang lingkup proyek. Untuk dapat melakukan identifikasi terhadap semua permasalahan diatas perlu dilakukan analisi terhadap seluruh aspek yang berkaitan dengan sistem yang akan dikembangkan. Hasil dari fase *inception* adalah :

- Gambaran umum mengenai sistem meliputi kebutuhan sistem, fitur utama dan kendala utama.
- Use case telah dibuat (10% - 20%).
- Mendefinisikan kemungkinan masalah
- Mendefinisikan resiko.
- Melakukan perancangan proyek meliputi fase dan literasi yang terjadi.

- Pemodelan bisnis.

2. Elaborasi

Tahap ini lebih fokus kepada analisis dan perancangan sistem yang akan dikembangkan berdasarkan permodelan *use case* pada tahap sebelumnya. Fase elaborasi mengakomodasi perubahan kebutuhan dari fase sebelumnya berdasarkan perubahan kebutuhan dari pemangku kepentingan. Dalam fase ini harus memiliki satu atau dua iterasi, sehingga pada akhir fase sudah didapatkan sebuah kebutuhan dan rancangan sistem yang stabil. Hasil dari fase elaborasi antara lain :

- Pemodelan *use case* telah mencapai 80% selesai.
- Mendefinisikan arsitektur sistem.
- Mendefinisikan rancangan pengembangan sistem secara menyeluruh.

3. Konstruksi

Tahapan ini adalah tahapan dilakukannya pembuatan sistem secara keseluruhan. Fokus utamanya adalah pengembangan dan perbaikan rancangan sistem berdasar pada rancangan yang didefinisikan pada fase sebelumnya. Pada fase ini permodelan bisnis dan kebutuhan harus dalam kondisi stabil. Aktivitas pengujian juga terfokus pada fase ini. Hasil dari fase konstruksi antara lain:

- Produk perangkat lunak harus sudah dapat digunakan.
- Panduan pengguna.
- Deskripsi versi dari produk perangkat lunak.

4. Transisi

Pada tahapan akhir dalam metode RUP ini sistem yang dikembangkan dapat diberikan kepada pengguna akhir. Hal tersebut dapat memberikan *feedback* sehingga dapat mengetahui kekurangan yang nantinya akan diselesaikan pada versi berikutnya. Dapat diartikan pada tahapan ini mengharuskan sistem telah selesai dan mendapatkan tingkat penerimaan tinggi dari pengguna. Hasil dari fase transisi adalah :

- *Beta tesing* untuk mengetahui validasi sistem terhadap penggunaanya.
- Peralihan dari sistem sebelumnya pada sistem yang baru.
- Penerapan data sesungguhnya.
- Pelatihan dan pengelolaan sistem.

2.13.2 Aktivitas dalam Metode RUP

Terdapat sembilan aktifitas yang ada pada metode *Rational Unified Process (RUP)*. Aktifitas - aktifitas tersebut dikelompokkan menjadi 2 kategori utama yang berdasarkan dari jenis aktifitasnya. Sehingga terdapat 6 aktifitas yang tergabung pada *core process workflow* dimana didalamnya merupakan aktifitas yang berkaitan dengan pengembangan sistem. Sedangkan 3 aktifitas lainnya tergolong dalam *core supporting workflow* yang lebih berkaitan dengan manajemen dari proyek tersebut. Berikut merupakan aktifitas – aktifitas yang terdapat pada metode *Rational Unified Process (IBM,1998)*:

1. Pemodelan bisnis.

Melakukan pemodelan bisnis bertujuan untuk memahami lebih dalam mengenai proses bisnis yang berkaitan dengan sistem yang akan dibuat. Sehingga terdapat pemahaman yang sama pada semua pemangku kepentingan. Namun tidak semua proyek menerapkan aktifitas ini.

2. Kebutuhan

Aktifitas ini bertujuan untuk menggambarkan apa saja yang dapat dilakukan oleh sistem. Tahapan ini terdapat identifikasi terhadap aktor dan *use case*. Sehingga dapat diketahui interaksi antara aktor dan sistem.

3. Analisis dan Perancangan

Tujuan aktifitas ini adalah untuk menggambarkan bagaimana sistem akan dibuat pada tahap implementasi. Perancang terdiri dari desain struktur *class* dan tampilan yang nantinya akan diterapkan. Perancangan berdasar pada arsitektur utama dan desain perancangan merupakan fokus utama iterasi.

4. Implementasi

Implementasi dilakukan untuk merealisasikan perancangan sistem yang telah dibuat menjadi sebuah sistem nyata.

5. Pengujian

Pengujian bertujuan untuk melakukan validasi terhadap integrasi yang ada pada sistem. Selain itu juga dilakukan validasi terhadap semua kebutuhan yang telah di definisikan pada awal pengembangan sistem. Secara garis besar pengujian dilakukan terhadap 3 ukuran kualitas *functionality*, *application performance* dan *system performance*.

6. *Deployment*

Deployment merupakan aktifitas untuk merilis produk yang telah dibangun sebelumnya.

7. Konfigurasi dan manajemen perubahan

Aktifitas ini bertujuan untuk mengontrol berbagai artefak yang didefinisikan oleh tim pengembang secara keseluruhan.

8. Manajemen Proyek

Aktivitas ini merupakan aktivitas pendukung mengenai penyelarasan antara tujuan utama, manajemen resiko, dan manajemen terhadap kendala yang ada.

9. *Environment*

Aktivitas ini merupakan aktivitas pendukung yang berkaitan dengan konfigurasi proses dalam proyek selain itu juga berfokus untuk membuat pedoman *guideline* dalam proyek.

2.14 Pengujian

Pengujian perangkat lunak merupakan sebuah tahapan untuk bahwa perangkat lunak yang telah dikembangkan dapat berjalan sesuai harapan dan untuk mengetahui cacat (*defects*) (Sommerville, 2011). Dalam pengujian perangkat lunak terdapat berbagai jenis pengujian dengan fokus yang berbeda – beda. Secara garis besar dibagi menjadi 2 jenis yaitu *development testing* dan *user testing*.

2.14.1 *Black Box Testing*

Pengujian *black box* juga dikenal sebagai *behavioral testing* merupakan pengujian perangkat lunak yang berfokus pada persyaratan fungsional. Pengujian *black box* dilakukan dengan cara menentukan set *input* yang nantinya akan diproses oleh sistem dengan sepenuhnya menjalankan semua persyaratan fungsional. Pengujian ini tidak memperdulikan struktur dari sistem, melainkan berfokus pada domain informasi (Pressman, 2010).

Pengujian *black box* berusaha untuk menemukan kesalahan dalam beberapa katagori sebagai berikut :

1. Fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan performa.
5. Kesalahan inisialisasi dan kesalahan terminasi.

2.14.2 *Compatibility Testing*

Compatibility tesing merupakan pengujian sebuah *web* dengan menjalankan pada berbagai *host* yang berbeda. Tujuannya adalah untuk menemukan kesalahan yang khusus pada konfigurasi *host* (Pressman, 2010). Pengujian ini pada dasarnya dilakukan untuk mengetahui performa dari sistem pada sisi *client* atau pengguna yang memiliki berbagai jenis cara untuk mengakses sistem tersebut.

Konfigurasi dari pengujian ini dapat meliputi penggunaan sistem operasi, jenis perangkat, aplikasi peramba atau kecepatan yang berbeda. Salah satu

software yang dapat membantu dalam *compatibility testing* adalah *sortsite*. *Software* tersebut dapat melakukan serangkaian pengujian dengan cara memverituaikan berbagai konfigurasi.

2.14.3 User Acceptance Testing

User Acceptance Testing (UAT) merupakan pengujian tingkat penerimaan sistem berdasarkan harapan *user* dari sistem tersebut. Pemangku kepentingan harus memastikan bahwa sistem yang telah dibangun dapat memenuhi kebutuhan dan harapan dari pengguna akhir agar sistem tersebut diterima atau digunakan. Pengujian UAT harus dapat diukur sehingga dapat diketahui sejauh mana sistem diterima oleh *end user* (Kshirasagar, 2008).

Pada pengujian *User Acceptance Testing* terdapat beberapa kriteria yang membantu dalam menilai penerimaan sistem. Namun dalam penerapannya tidak semua kriteria harus diuji pada sistem karena akan mengakibatkan terlalu banyak dan umumnya hasil pengujian. Tim pengembang dapat memilih bagian – bagian penting yang dapat mewakili penerimaan sistem. Berikut merupakan kriteria – kriteria yang ada pada pengujian UAT.

Table 2.17 Kriteria User Acceptance Testing

Kriteria	Diskripsi
<i>Functional Correctness and Completeness</i>	Berfokus pada apakah sistem tersebut telah sesuai harapan dengan mengacu pada fitur yang telah didefinisikan.
<i>Accuracy</i>	Berfokus apakah perhitungan atau hasil dari sistem sesuai dengan nilai seharusnya.
<i>Data Integrity</i>	Berfokus pada performa sistem dalam mengolah data.
<i>Data Conversion</i>	Berfokus pada performa sistem dalam menjaga data pada saat terjadi perubahan jenis atau bentuk.
<i>Backup and Recovery</i>	Berfokus pada pengcadangan dan pemulihan data
<i>Competitive Edge</i>	Berfokus pada keunggulan dibandingkan model yang lain.
<i>Usability</i>	Berfokus pada kemudahan dalam penggunaan sistem.
<i>Performance</i>	Berfokus pada kinerja sistem.

Sumber: Diadaptasi dari Kshirasagar (2008)

Dalam proses *User Acceptance Testing* terdapat beberapa tahap yang harus dilakukan (Sommerville, 2011):

1. *Define acceptance criteria*

Tahapan awal dalam UAT adalah mendefinisikan kriteria apa yang akan dijadikan pedoman dalam pengujian.

2. *Plan acceptance testing*

Tahap perencanaan mengenai biaya, waktu dan partisipan dalam pengujian yang akan dilakukan.

3. *Derive acceptance tests*

Pada tahap ini terdapat perencanaan lebih mendetail mengenai pengujian sehingga pada akhirnya dapat mewakili pengujian pada seluruh kebutuhan fungsional dan non fungsional.

4. *Run acceptance tests*

Tahapan dimana pengujian dilakukan dan melibatkan calon pengguna. Idealnya sistem dijalankan dalam lingkungan sebenarnya namun hal tersebut mungkin dapat mengganggu atau membutuhkan sumber daya yang besar.

5. *Negotiate test results*

Tahapan dimana hasil dari pengujian diolah sedemikian rupa sehingga dapat diambil sebuah nilai besar penerimaan terhadap sistem.

6. *Reject/accept system*

Merupakan tahap akhir dimana pemangku kepentingan akan memberikan keputusan apakah sistem dapat dijalankan atau harus dilakukan perbaikan.