

BAB 2

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Salah satu usaha dalam rangka meningkatkan pemasaran produk yang ada pada Koperasi Sejahtera kabupaten Batang secara langsung juga akan meningkatkan pendapatan pada koperasi itu sendiri. Menggunakan dan juga memanfaatkan perkembangan teknologi informasi sekarang ini sudah menjadi sebuah kebutuhan yang sangat penting, kemudahan-kemudahan yang didapat sangat membantu dalam semua aspek yang ada. Pada penelitian sebelumnya yang berkaitan dengan pemasaran produk secara *online* adalah yang dilakukan Wendiansyah Febrianto, Sasongko dan Sri Wahyuni dengan judul Pemasaran Secara *Online* Produk Batik Pada Rumah Batik Rolla Jember dengan hasil pemasaran secara online meningkatkan pendapatan dan juga menjadi sarana yang mudah dan murah dalam mengembangkan usaha. [1]

Penelitian lainnya yang dilakukan oleh Hendri pada tahun 2012 dengan judul Pemanfaatan *Content Management System* (CMS) Sebagai Solusi Dalam Memasarkan Produk Secara *Online* (*Online Marketing*) Untuk UMKM di kota Jambi. Mudah dalam menggunakan implementasi aplikasi *online* yg dapat digunakan oleh siapapun terutama UMKM yang berusaha mengembangkan pemasaran secara *online*. [2]

Dari penelitian terkait diatas dapat dirangkumkan pada tabel dibawah ini :

Tabel 2.1 Penelitian Terkait

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
1.	Wendiansyah Febrianto, Sasongko dan Sri Wahyuni,	Meningkatan pemasaran produk	Deskriptif	pemasaran secara <i>online</i> meningkatkan pendapatan dan juga menjadi sarana yang

No	Nama Peneliti dan Tahun	Masalah	Metode	Hasil
	2013			mudah dan murah dalam mengembangkan usaha
2.	Hendri	Memanfaatkan CMS untuk memasarkan produk secara online	Prototipe	Mudah dalam menggunakan implementasi aplikasi online yg dapat digunakan oleh siapapun terutama UMKM yang berusaha mengembangkan pemasaran secara <i>online</i>

2.2 Konsep Sistem Informasi

2.2.1 Pengertian Sistem

Sistem dapat didefinisikan sebagai sekumpulan elemen yang saling terkait dan terpadu yang dimaksudkan untuk mencapai suatu tujuan. Suatu sistem mempunyai maksud tertentu. Ada yang menyebutkan maksud dari suatu sistem adalah untuk mencapai suatu tujuan (goal) dan ada yang menyebutkan untuk mencapai suatu sasaran (objektif). Goal biasanya dihubungkan dengan ruang lingkup yang lebih luas dan sasaran dalam ruang lingkup yang lebih sempit. Bila merupakan suatu sistem utama, seperti misalnya sistem bisnis, maka istilah goal lebih tepat diterapkan.

Untuk sistem akuntansi atau sistem-sistem lainnya yang merupakan bagian atau subsistem dari sistem bisnis, maka istilah objektif yang lebih tepat. Jadi

tergantung dari ruang lingkup darimana memandang sistem tersebut. Seringkali tujuan (goal) dan sasaran (objektif) digunakan bergantian dan tidak dibedakan. [3]

2.2.2 Pengertian Informasi

Informasi dapat didefinisikan sebagai data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang. Sumber dari informasi adalah data. Data seringkali disebut sebagai bahan mentah informasi. Melalui suatu proses transformasi, data dibuat menjadi lebih bermakna. [4]

Kualitas informasi tergantung dari 3 hal, yaitu :

1. Akurat

Informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan.

2. Tepat pada waktunya

Informasi yang datang pada penerima tidak boleh terlambat, karena informasi yang sudah usang tidak akan bernilai lagi.

3. Relevan

Informasi tersebut mempunyai manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

2.2.3 Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi yang bersifat manajemen dan kegiatan strategi dari suatu organisasi serta menyediakan pihak-pihak luar tertentu dan laporan-laporan yang diperlukan. [3]

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsioperasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. [4]

2.2.4 Pengertian Sistem Informasi Manajemen (SIM)

Sistem informasi manajemen merupakan penerapan sistem informasi di dalam organisasi untuk mendukung informasi-informasi yang dibutuhkan oleh semua tingkat manajemen. SIM dapat didefinisikan sebagai suatu sistem yang melakukan fungsi-fungsi untuk menyediakan semua informasi yang mempengaruhi semua informasi organisasi.

SIM biasanya menghasilkan informasi untuk memantau kinerja, memelihara kondisi, dan menyediakan informasi untuk operasi organisasi. [5]

2.3 Pengembangan Sistem Informasi

2.3.1 Perlunya Pengembangan Sistem Informasi

Pengembangan sistem informasi dapat diartikan sebagai menyusun suatu sistem baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem lama perlu diperbaiki atau diganti disebabkan karena beberapa hal, yakni hal-hal berikut ini:

1. Adanya permasalahan-permasalahan yang timbul di sistem yang lama. Permasalahn yang timbul berupa :
 - a. Ketidakterbacaan, misal kecurangan-kecurangan, tidak efisiennya operasi, tidak ditaatinya kebijaksanaan manajemen yang telah ditetapkan.
 - b. Pertumbuhan organisasi
Pertumbuhan organisasi diantaranya adalah kebutuhan informasi yang semakin luas, volume pengolahan data yang semakin meningkat, perubahan prinsip akuntansi yang baru. Karena adanya perubahan ini, maka menyebabkan sistem yang lama tidak efektif lagi, sehingga sistem yang lama sudah tidak memenuhi lagi semua kebutuhan informasi yang dibutuhkan oleh manajemen.
2. Untuk meraih kesuksesan
Teknologi informasi berkembang dengan sangat cepatnya. Organisasi mulai merasakan bahwa teknologi informasi ini perlu digunakan untuk meningkatkan penyediaan informasi sehingga dapat mendukung dalam proses pengambilan keputusan yang akan dilakukan oleh pihak manajemen. Dalam

keadaan pasar bersaing, kecepatan informasi atau efisiensi waktu sangat menentukan berhasil atau tidaknya strategi dan rencana-rencana yang telah disusun untuk meraih kesempatan-kesempatan yang ada.

3. Adanya instruksi-instruksi

Adanya instruksi-instruksi, misal dari pimpinan atau dari pihak luar (peraturan pemerintah) dapat mempengaruhi terjadinya pengembangan sistem yang baru.[3]

2.3.2 Siklus Hidup Pengembangan Sistem

Siklus hidup dari suatu sistem merupakan suatu bentuk yang digunakan untuk menggambarkan tahapan utama dan langkah-langkah di dalam tahapan tersebut dalam pengembangannya.

Tahapan-tahapan dari siklus hidup pengembangan sistem adalah :

1. Perencanaan sistem (*systems planning*)
2. Analisis sistem (*systems analysis*)
3. Desain sistem (*systems design*)
4. Seleksi sistem (*systems selection*)
5. Implementasi sistem (*systems implementation*)
6. Perawatan sistem (*systems maintenance*)

Namun secara umum dari tahapan-tahapan diatas, yang merupakan tahapan inti dari pengembangan sistem adalah analisis, desain, dan implementasi sistem. [3]

2.3.3 Analisis Sistem

2.3.3.1 Pengertian Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi, dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya. [6]

2.3.3.2 Tahap-tahap Analisis Sistem

1. Identifikasi Masalah

Mengidentifikasi masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah dapat didefinisikan sebagai suatu pernyataan yang diinginkan untuk dipecahkan. Masalah inilah yang menyebabkan sasaran dari sistem tidak tercapai. Tugas-tugas yang harus dilakukan dalam tahap ini adalah :

- a. Mengidentifikasi penyebab masalah.
- b. Mengidentifikasi titik keputusan.
- c. Mengidentifikasi personil-personil kunci

2. Memahami kinerja dari sistem yang ada

Langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang ada beroperasi. Sejumlah data perlu dikumpulkan dalam penelitian terinci ini. Pada langkah ini, ada beberapa tugas yang perlu dilakukan, yaitu :

- a. Menentukan jenis penelitian.
- b. Merencanakan jadwal penelitian.
- c. Membuat penugasan penelitian.
- d. Membuat agenda wawancara
- e. Mengumpulkan hasil penelitian

3. Menganalisis hasil penelitian

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang dilakukan. Menganalisis hasil penelitian sering sulit dilakukan oleh analis sistem yang masih baru. Pengalaman menunjukkan bahwa banyak analis sistem baru mencoba memecahkan masalah tanpa menganalisisnya.

4. Membuat laporan hasil analisis

Laporan hasil penelitian diserahkan kepada steering committee yang nantinya akan diteruskan ke manajemen. Tujuan utama dari penyerahan laporan ini kepada manajemen adalah :

- a. Pelaporan bahwa analisis telah selesai dilakukan.

- b. Meluruskan kesalah-pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analis sistem tapi tidak sesuai menurut manajemen.
- c. Meminta pendapat dan saran-saran dari pihak manajemen.
- d. Meminta persetujuan kepada pihak manajemen untuk melakukan tindakan selanjutnya.

2.4 UML (*Unified Modeling Language*) Sistem

Unified Modelling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk memvisualisasi, merancang dan mendokumentasikan sistem perangkat lunak. Dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi perangkat lunak, dimana aplikasi tersebut dapat berjalan pada perangkat keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa pemrograman yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam perangkat lunak.

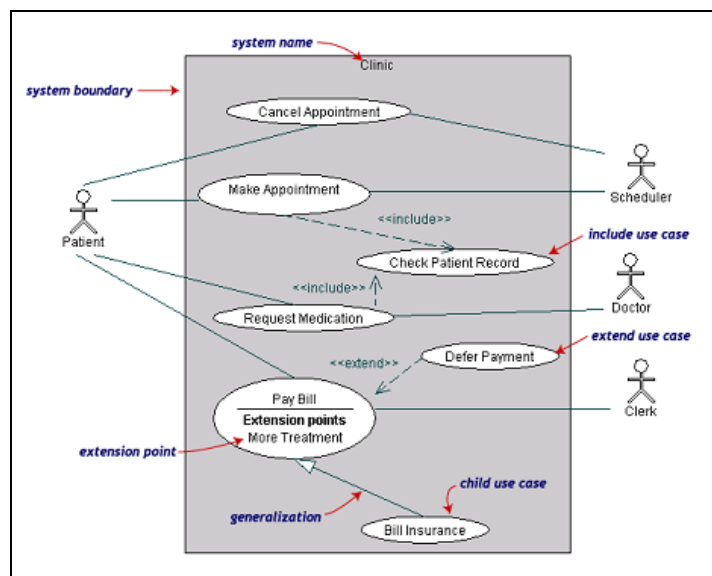
UML sendiri terdiri atas pengelompokan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model. UML, mempunyai 8 diagram, yaitu:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include*

dieksekusi secara normal. Sebuah *use case* dapat dapat juga di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat *dihindari* dengan cara menarik keluar fungsionalitas yang common. Sebuah *use case* juga dapat meng-*extend* *use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.



Gambar 2.1: Contoh use case diagram

2. Class Diagram

Class diagram memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan mereka. *Class diagram* bersifat statis, menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan.

Class diagram mempunyai 3 macam *relationships* (hubungan), yaitu:

a. Association

Suatu hubungan antara bagian dari dua kelas. Terjadi association antara dua kelas jika salah satu bagian dari kelas mengetahui yang lainnya dalam melakukan suatu kegiatan. Di dalam diagram, sebuah association adalah penghubung yang menghubungkan dua kelas.

b. Aggregation

Suatu *association* dimana salah satu kelasnya merupakan bagian dari suatu kumpulan. Aggregation memiliki titik pusat yang mencakup keseluruhan bagian.

c. Generalization

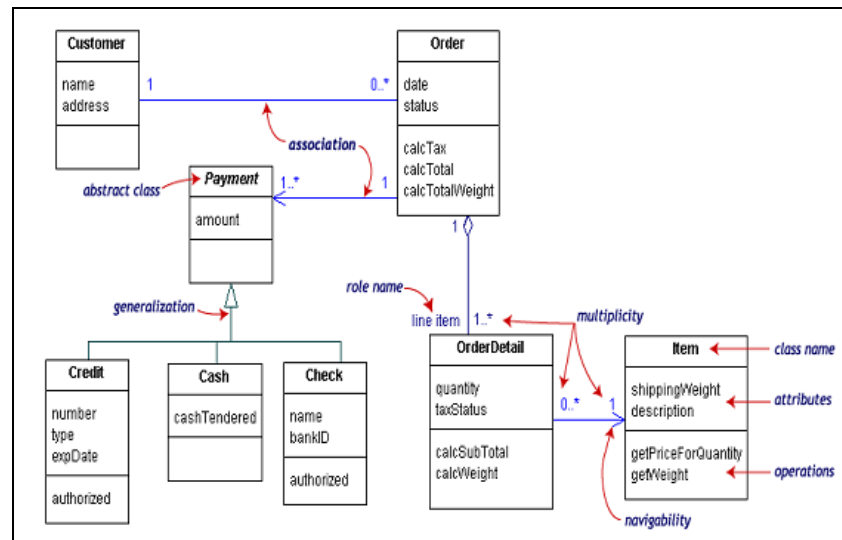
Suatu hubungan turunan dengan mengasumsikan satu kelas merupakan suatu *superClass* (kelas super) dari kelas yang lain. *Generalization* memiliki tingkatan yang berpusat pada *superClass*.

Panah *navigability* (pengatur alur arah) dalam suatu *association* menggambarkan arah mana *association* dapat ditransfer atau disusun. Panah ini juga menjelaskan siapa “memiliki” implementasi dari *association*. *Association* tanpa arah panah merupakan bidirectional (bolak-balik) *Multiplicity* dari suatu titik *association* adalah angka kemungkinan bagian dari hubungan kelas dengan single *instance* (bagian) pada titik yang lain. *Multiplicity* berupa *single number* (angka tunggal) atau *range number* (angka batasan).

Tabel 2.2: Multiplicity

Multiplicities	Keterangan
0..1	Nol atau satu bagian. Notasi <i>n..m</i> menerangkan n sampai m bagian.
0..* atau *	Tak hingga pada jangkauan bagian (termasuk kosong)
1	Tepat satu bagian
1..*	Sedikitnya hanya satu bagian

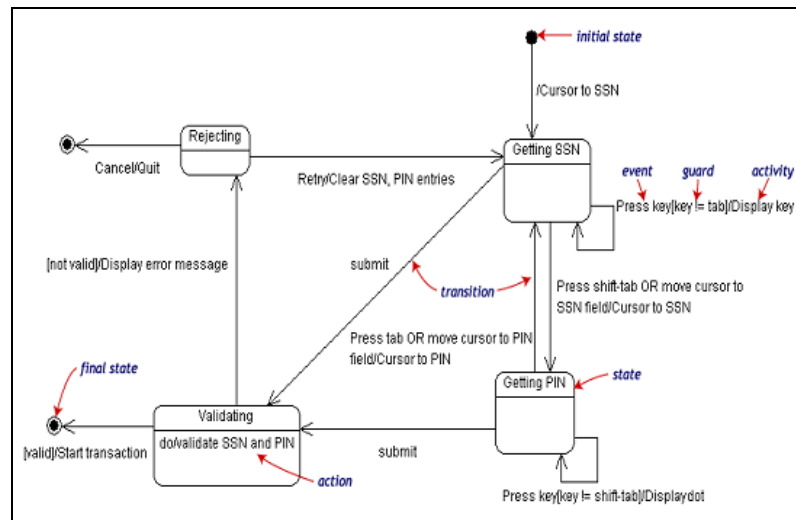
Setiap class diagram memiliki *Class* (kelas), *Association*, dan *Multiplicity*. Sedangkan *navigability* (alur arah) dan *role* (kegiatan) merupakan suatu tambahan yang tidak diharuskan.



Gambar 2.2: Contoh class diagram

3. Statechart Diagram

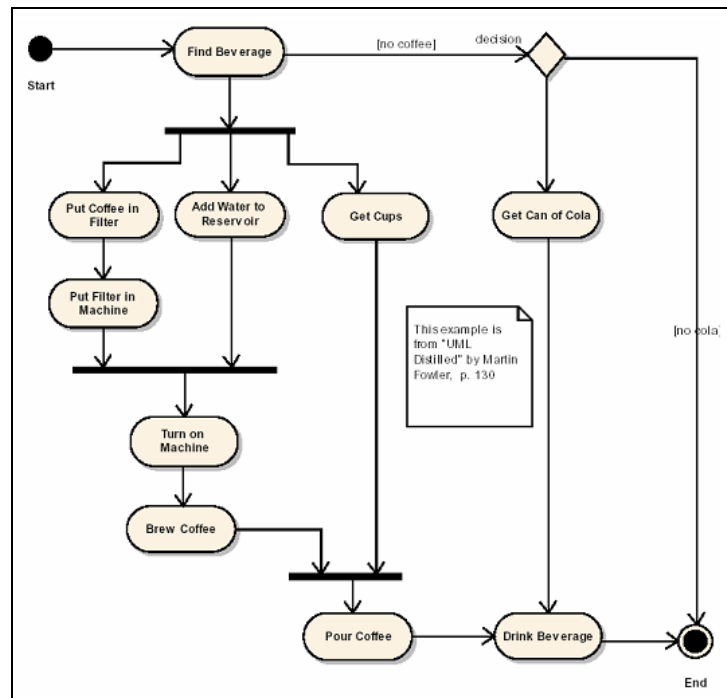
Statechart diagram menggambarkan transisi dan perubahan keadaan (dari suatu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari stimuli yang diterima. Pada umumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*). Dalam UML, *statechart* digambarkan berbentuk segi empat dengan sudut membulat dan *memiliki* nama sesuai kondisi saat itu. Transisi antar *state* umumnya memiliki kondisi *guard* yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. *Action* yang dilakukan sebagai akibat dari *event* tertentu dituliskan dengan diawali garis miring.



Gambar 2.3: Contoh statechart diagram

4. Activity Diagram

Activity Diagram memodelkan alur kerja (*workflow*) sebuah proses bisnis dan urutan aktifitas dalam suatu proses. diagram ini sangat mirip dengan sebuah flowchart karena kita dapat memodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari satu aktivitas kedalam keadaan sesaat (*state*). Seringkali bermanfaat bila kita membuat sebuah *activity diagram* terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan. *Activity diagram* juga sangat berguna ketika ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi.

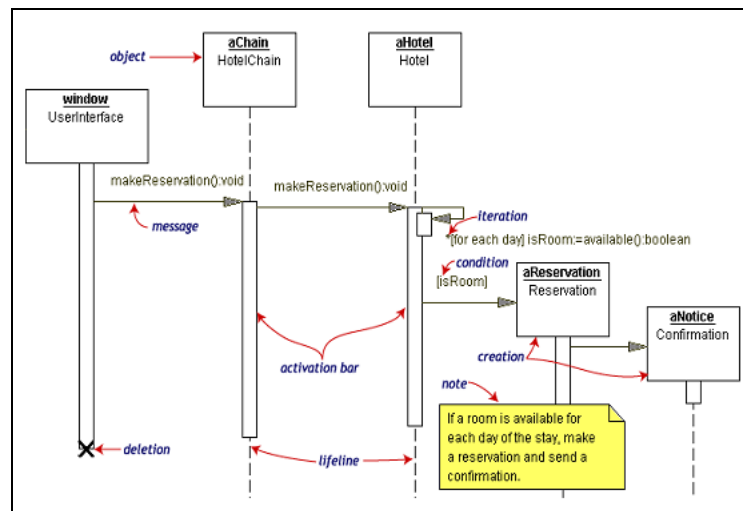


Gambar 2.4: Contoh activity diagram

5. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan sekitar sistem berupa *message* yang digambarkan terhadap waktu. Sequence diagram terdiri dari dimensi vertikal (waktu) dan dimensi horizontal (objek-objek terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respons dari sebuah *event* untuk menghasilkan *output* tertentu.

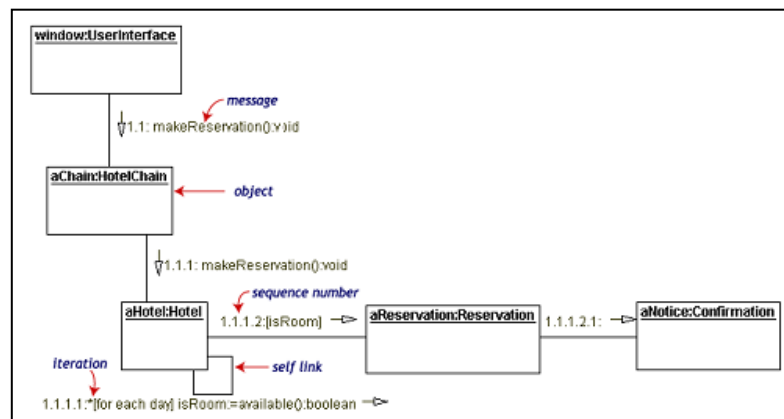
Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara *internal* dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline vertikal*. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi atau metoda dari class. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message*.



Gambar 2.5: Contoh *sequence diagram*

6. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. *Messages* dari level yang sama memiliki prefiks yang sama.

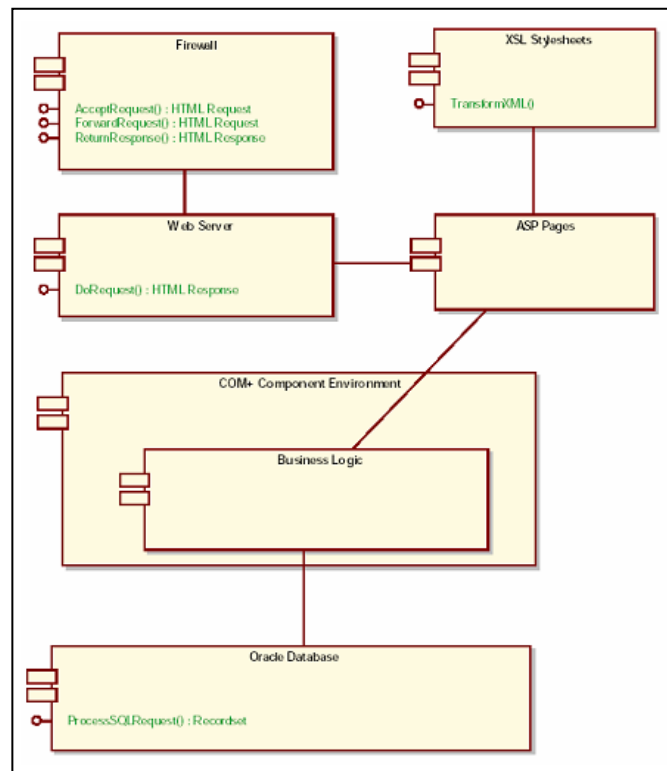


Gambar 2.6: Contoh *collaboration diagram*

7. Component Diagram

Component Diagram menggambarkan alokasi semua kelas dan objek ke dalam komponen-komponen dalam desain fisik sistem software. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponen-komponen

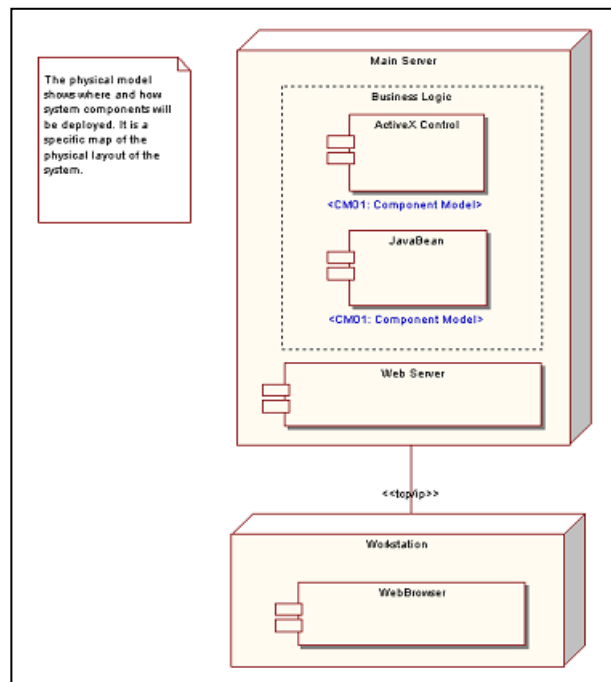
software, seperti *source code*, *binary code*, dan komponen tereksekusi (*executable components*).



Gambar 2.7: Contoh *component diagram*

8. *Deployment Diagram*

Deployment Diagram menggambarkan detail bagaimana komponen dikembangkan dalam infrastruktur sistem, di mana komponen akan terletak, bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik.

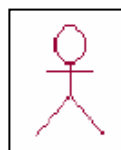


Gambar 2.8: Contoh *deployment diagram*

UML menyediakan beberapa notasi dan artifak standar yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. Artifak di dalam UML didefinisikan sebagai informasi dalam berbagai bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat lunak. Beberapa notasi yang digunakan dalam analisis dan desain berorientasi objek adalah sebagai berikut:

1. *Actor*

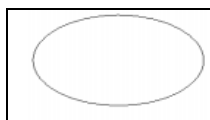
Actor menggambarkan segala pengguna software aplikasi (*user*). *Actor* memberikan suatu gambaran jelas tentang apa yang harus dikerjakan software aplikasi. Sebuah *actor* mungkin seorang manusia, suatu *device*, *hardware* atau sistem informasi lainnya.



Gambar 2.9: Notasi *actor*

2. *Use case*

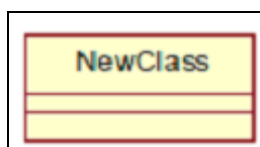
Use case menjelaskan urutan kegiatan yang dilakukan *actor* dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan, namun *use case* hanya menjelaskan apa yang dilakukan oleh *actor* dan sistem bukan bagaimana actor dan sistem melakukan kegiatan tersebut.



Gambar 2.10: Notasi *use case*

3. *Class*

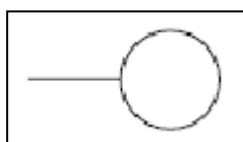
Class merupakan pembentuk utama dari sistem berorientasi objek, karena *class* menunjukkan kumpulan objek yang memiliki atribut dan operasi yang sama. *Class* digunakan untuk mengimplementasikan interface. *Class* digunakan untuk menggambarkan elemen-elemen dari sistem yang sedang dibangun. *Class* bisa untuk merepresetasikan baik perangkat lunak maupun perangkat keras, baik konsep maupun benda nyata.



Gambar 2.11: Notasi *class*

4. *Interface*

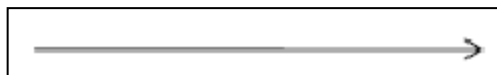
Interface merupakan kumpulan operasi tanpa implementasi dari suatu *class*. Implementasi operasi dalam *interface* dijabarkan oleh operasi didalam *class*. Oleh karena itu keberadaan *interface* selalu disertai oleh *class* yang mengimplementasikan operasinya. *Interface* ini merupakan salah satu cara mewujudkan prinsip enkapsulasi dalam objek.



Gambar 2.12: Notasi *interface*

5. *Interaction*

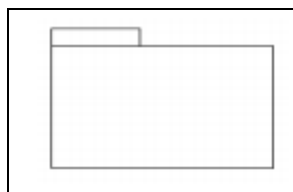
Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar objek maupun hubungan antar objek. Biasanya *interaction* ini dilengkapi juga teks bernama *operation signature* yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.



Gambar 2.13: Notasi *interaction*

6. *Package*

Package adalah kontainer atau wadah konseptual yang digunakan untuk mengelompokkan elemen-elemen dari sistem yang sedang dibangun, sehingga bisa dibuat model yang lebih sederhana. Tujuannya adalah untuk mempermudah penglihatan (*visibility*) dari model yang sedang dibangun.



Gambar 2.14: Notasi *package*

7. *Dependency*

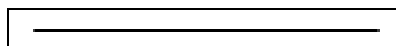
Dependency merupakan relasi yang menunjukkan bahwa perubahan pada salah satu elemen memberi pengaruh pada elemen lain. Elemen yang ada di bagian tanda panah adalah elemen yang bergantung pada elemen yang ada di bagian tanpa panah.



Gambar 2.15: Notasi *Dependency*

8. Association

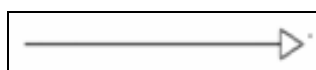
Association menggambarkan navigasi antar class (*Navigation*), berapa banyak objek lain yang bisa berhubungan dengan satu objek (*multiplicity antar class*), dan apakah suatu class menjadi bagian dari class lainnya (*Aggregation*)



Gambar 2.16: Notasi association

9. Generalization

Generalization menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik. Dengan *generalization*, *class* yang lebih spesifik (*subclass*) akan menurunkan atribut atau operasi dari class yang lebih umum (*superclass*) atau *subclass is superclass*. Dengan menggunakan notasi *generalization* ini konsep inheritance dari prinsip hirarki dimodelkan



Gambar 2.17: Notasi generalization

10. Realization

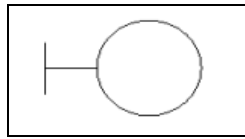
Realization menunjukkan hubungan bahwa elemen yang ada di bagian tanpa panah akan merealisasikan apa yang dinyatakan oleh elemen yang ada di bagian dengan panah



Gambar 2.18: Notasi realization

11. Boundry Class

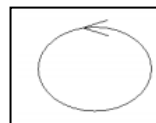
Boundary class adalah *class* yang menghubungkan user dengan sistem. Oleh karena itu sering disebut sebagai user *interface class*, dan untuk selanjutnya *user interface class* sering disamakan dengan *form* yang digunakan sebagai *interface* antara sistem dengan *user*



Gambar 2.19: Notasi *boundry class*

12. Control Class

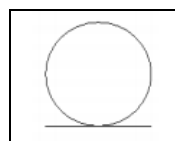
Control class adalah *class* yang mengkoordinasikan aktivitas dalam sistem. *Class* ini *menghubungkan boundary class* dengan *entity class*. Jadi mengkoordinasikan *entity class* mana yang perlu dikunjungi, kapan dan apa yang ingin didapatkan dari *entity class* tersebut.



Gambar 2.20: Notasi *control class*

13. Entity Class

Entity class adalah *class* yang berhubungan dengan data atau informasi yang digunakan oleh sistem. *Entity class* ini adalah *class* yang menyimpan dan mengolah data.



Gambar 2.21: Notasi *entity class*

2.5 Rekayasa Web (*Web Engineering*)

Perkembangan teknologi internet yang pesat mengakibatkan ketergantungan masyarakat pada sistem dan aplikasi yang menggunakan antarmuka web dalam menjalankan rutinitas interaksi online. Oleh karena itu para pengembang web membutuhkan suatu metoda, suatu bidang keilmuan dan proses yang dapat diduplikasi, alat-alat pengembang web yang baik dan panduan-panduan dalam

proses pengembangan web yang baik. *Web engineering* (rekayasa web) adalah suatu proses yang digunakan untuk menciptakan suatu sistem aplikasi berbasis web dengan menggunakan ilmu rekayasa, prinsip-prinsip manajemen dan pendekatan sistematis sehingga dapat diperoleh sistem dan aplikasi web dengan kualitas tinggi.

2.5.1 Alur Kerja Rekayasa Web

Bertolak belakang dengan persepsi dari beberapa pengembang perangkat lunak dan ahli-ahli dalam bidang rekayasa perangkat lunak (*software engineering professional*), rekayasa web tidaklah sama dengan rekayasa perangkat lunak walaupun keduanya melibatkan pemrograman dan pengembangan perangkat lunak. Walaupun rekayasa web banyak mengadopsi prinsip-prinsip rekayasa perangkat lunak, rekayasa web memiliki banyak pendekatan, metoda, alat bantu, teknik dan panduan yang memenuhi persyaratan pembuatan sistem berbasis web. Pengembangan sistem berbasis web berbeda dengan pengembangan perangkat lunak konvensional, dimana pengembangan sistem berbasis web lebih banyak menghadapi tantangan. Pengembangan web adalah gabungan dari print publishing dan pengembangan perangkat lunak, diantara marketing dan perhitungan dan diantara seni dan teknologi.

2.5.2 Formulasi (*Formulation*)

Kegiatan yang berfungsi untuk merumuskan tujuan dan ukuran dari aplikasi berbasis web serta menentukan batasannya sistem. Tujuan yang ingin dicapai bisa dibedakan menjadi dua kategori, yaitu:

1. Tujuan yang bersifat informatif
Menyediakan suatu informasi tertentu kepada pengguna, berupa teks, grafik, audio, dan video.
2. Tujuan yang bersifat fungsional
Kemampuan untuk melakukan suatu fungsi yang dibutuhkan pengguna, misal dengan menggunakan aplikasi tersebut seorang dosen dapat memperoleh nilai akhir dan statistik nilai mahasiswa dari data-data ujian, tugas, kuis yang dia input ke dalam aplikasi.

2.5.3 Perencanaan (*Planning*)

Kegiatan yang digunakan untuk menghitung estimasi biaya proyek pembuatan aplikasi berbasis web ini, estimasi jumlah pengembang, estimasi waktu pengembangan, evaluasi resiko pengembangan proyek, dan mendefinisikan jadwal pengembangan untuk versi selanjutnya (jika diperlukan).

2.5.4 Analisis (*Analysis*)

Kegiatan untuk menentukan persyaratan – persyaratan teknik dan mengidentifikasi informasi yang akan ditampilkan pada aplikasi berbasis web. Analisis yang digunakan pada rekayasa web dilakukan dari empat sisi, yaitu :

1. Analisis isi informasi

Mengidentifikasi isi yang akan ditampilkan pada aplikasi berbasis web ini. Isi informasi dapat berupa teks, grafik, audio, maupun video.

2. Analisis interaksi

Analisis yang menunjukkan hubungan antara web dengan pengguna.

3. Analisis fungsional

Analisis tentang proses bagaimana aplikasi berbasis web ini akan menampilkan informasi kepada pengguna.

4. Analisis konfigurasi

Konfigurasi yang digunakan pada aplikasi berbasis web, internet, intranet, atau extranet. Selain itu, analisis ini juga meliputi relasi database dengan web jika diperlukan.

2.5.5 Implementasi (*Page Generation*) dan Pengujian (*Testing*)

Suatu kegiatan untuk mewujudkan desain menjadi suatu web site. Teknologi yang digunakan tergantung dengan kebutuhan yang telah dirumuskan pada tahap analisis. Pengujian dilakukan setelah implementasi selesai dilaksanakan. Pengujian meliputi beberapa parameter yang akan menentukan standar aplikasi berbasis web yang telah dibuat. Tahap pengujian adalah suatu proses untuk menguji aplikasi berbasis web yang telah selesai dibuat.

Hal ini bertujuan untuk menemukan kesalahan dan kemudian memperbaikinya. Pengembang suatu aplikasi berbasis web mendapat tantangan besar untuk melakukan pengujian karena karakter aplikasi ini yang beroperasi pada jaringan

dengan berbagai macam pengguna, berbagai macam sistem operasi, perangkat keras, browser, protokol komunikasi, dll. Ada beberapa pendekatan yang digunakan untuk melakukan pengujian, yaitu :

1. Pengujian fungsional dan operasional (fungsional and operational testing), bertujuan untuk menguji masukan dan keluaran dari aplikasi ini. Hasil keluaran aplikasi bergantung dari teknologi yang digunakan, baik itu bahasa pemrograman maupun bahasa skrip yang digunakan.
2. Pengujian navigasi (navigation testing), digunakan untuk melihat kesesuaian antara desain navigasi dengan navigasi yang ada di aplikasi. Navigasi berhubungan dengan link-link yang terdapat didalam aplikasi.
3. Pengujian konfigurasi (configuration testing), dilakukan pada sistem operasi, browser, sistem perangkat keras dan perangkat lunak pendukung. Pengujian ini dilakukan untuk menentukan batas toleransi kebutuhan aplikasi akan perangkat lunak dan perangkat keras pendukungnya.
4. Pengujian keamanan dan performansi (security and performance testing), dilakukan untuk melihat tingkat keamanan aplikasi dengan cara menguji aspek-aspek yang dapat menimbulkan gangguan keamanan aplikasi maupun server. Keamanan aplikasi sangat bergantung pada teknologi pengembangan *website*, konfigurasi server yang digunakan dan kelakuan sistem. Pengujian performansi dapat dilakukan bersamaan dengan pengujian keamanan aplikasi, karena keamanan aplikasi berbasis web juga tergantung dari performansi server dan aplikasi tersebut.

2.5.6 Evaluasi oleh Konsumen (*Customer Evaluation*)

Suatu kegiatan akhir dari siklus proses rekayasa web, akan menentukan apakah web yang telah selesai dibuat tersebut sesuai dengan yang mereka inginkan. Apabila aplikasi berbasis web ini belum sesuai dengan kehendak mereka, maka proses rekayasa web akan terus dilakukan dan dimulai lagi dari tahap formulasi untuk versi berikutnya.

2.6 Metode Pengembangan Sistem

Proses pengembangan sistem adalah satu set aktivitas, metode, praktik terbaik, siap dikirimkan dan peralatan terotomatisasi yang akan digunakan oleh para pengembang sistem untuk mengembangkan dan memelihara sistem informasi dan perangkat lunak. [6]

2.6.1 *Rapid Application Development (RAD)*

Rapid Application Development (RAD) merupakan sebuah strategi yang menekankan kecepatan pengembangan melalui keterlibatan pengguna yang ekstensif dalam konstruksi, cepat, berulang dan bertambah serangkaian prototype bekerja pada sebuah sistem yang pada akhirnya berkembang ke dalam sistem final. [7]

2.6.2 Fase-fase (RAD)



Gambar 2.22 : Siklus (Fase-fase) RAD. [7]

Metode-metode RAD mempunyai 3 (tiga) tahapan utama, yaitu :

1. Perencanaan Syarat-syarat

Pada tahap ini, user dan analyst melakukan semacam pertemuan untuk melakukan identifikasi tujuan dari aplikasi atau sistem dan melakukan identifikasi kebutuhan informasi untuk mencapai tujuan. Pada tiap tahap ini hal terpenting adalah adanya keterlibatan dari kedua belah pihak bukan hanya sekedar persetujuan proposal yang sudah dibuat. Untuk lebih jauh lagi, keterlibatan user bukan hanya dari satu tingkatan pada suatu organisasi,

melainkan beberapa tingkatan organisasi, sehingga informasi yang dibutuhkan untuk masing-masing user dapat terpenuhi dengan baik.

2. Workshop Design

Pada tahap ini adalah melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara user dan analyst. Untuk tahap ini maka keaktifan user yang terlibat sangat menentukan untuk mencapai tujuan, karena user bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain.

3. Implementasi

Setelah desari dari sistem yang akan dibuat sudah disetujui baik itu oleh user dan analyst, maka pada tahap ini programmer mengembangkan desain menjadi suatu program. Setelah program selesai dengan baik itu sebagaimana maupun secara keseluruhan, maka dilakukan proses pengujian terhadap program tersebut apakah terdapat kesalahan atau tidak sebelum diaplikasikan pada suatu organisasi.

2.6.3 Kekurangan dan kelebihan RAD

Metode pengembangan sistem RAD relatif lebih sesuai dengan rencana pengembangan aplikasi yang tidak memiliki ruang lingkup yang besar dan akan dikembangkan oleh tim yang kecil. Namun, RAD pun memiliki kelebihan dan kekurangannya sebagai sebuah metodologi pengembangan aplikasi. Berikut ini adalah kelebihan metodologi RAD: [8]

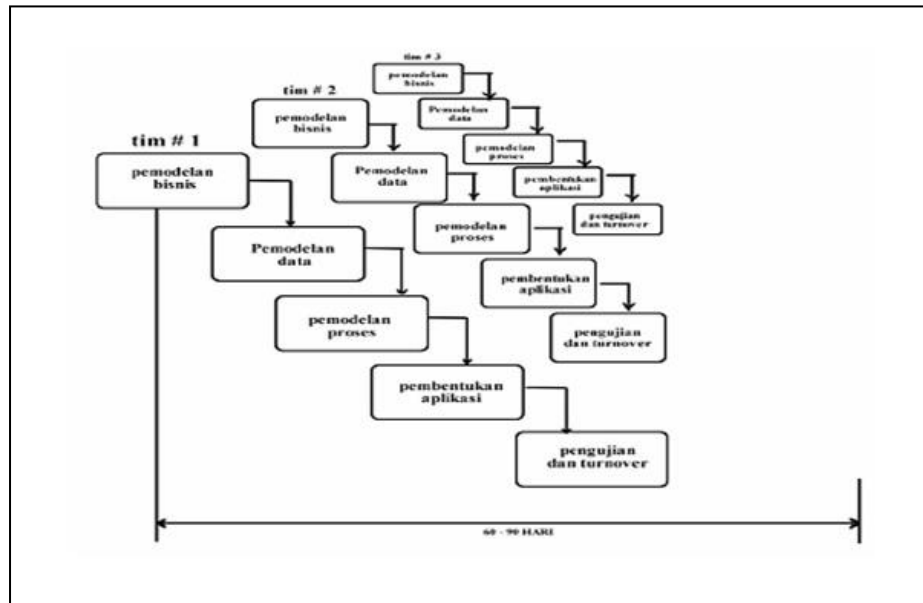
1. Penghematan waktu dalam keseluruhan fase projek dapat dicapai.
2. RAD mengurangi seluruh kebutuhan yang berkaitan dengan biaya projek dan sumberdaya manusia.
3. RAD sangat membantu pengembangan aplikasi yang berfokus pada waktu penyelesaian projek.

4. Perubahan desain sistem dapat lebih berpengaruh dengan cepat dibandingkan dengan pendekatan SDLC tradisional.
5. Sudut pandang user disajikan dalam sistem akhir baik melalui fungsi-fungsi sistem atau antarmuka pengguna.
6. RAD menciptakan rasa kepemilikan yang kuat di antara seluruh pemangku kebijakan proyek.

Sedangkan, mengacu pada pendapat Kendall (2018), maka dapat diketahui bahwa kekurangan penerapan metode RAD adalah sebagai berikut:

- a. Dengan metode RAD, penganalisis berusaha mempercepat proyek dengan terburu-buru.
- b. Kelemahan yang berkaitan dengan waktu dan perhatian terhadap detail. Aplikasi dapat diselesaikan secara lebih cepat, tetapi tidak mampu mengarahkan penekanan terhadap permasalahan-permasalahan perusahaan yang seharusnya diarahkan.
- c. RAD menyulitkan *programmer* yang tidak berpengalaman menggunakan prangkat ini di mana *programmer* dan *analyst* dituntut untuk menguasai kemampuan-kemampuan baru sementara pada saat yang sama mereka harus bekerja mengembangkan sistem.

2.6.4 Tahap – Tahap Rekayasa Software Dalam RAD Model



Gambar 2.23 : RAD Model

Model RAD menekankan pada tahap-tahap berikut :

1. Business modeling

Pada tahap ini, aliran informasi (*information flow*) pada fungsi-fungsi bisnis dimodelkan untuk mengetahui informasi apa yang mengendalikan proses bisnis, informasi apa yang dihasilkan, siapa yang membuat informasi itu, kemana saja informasi mengalir, dan siapa yang mengolahnya.

2. Data modeling

Aliran informasi yang didefinisikan dari *business modeling*, disaring lagi agar bisa dijadikan bagian-bagian dari objek data yang dibutuhkan untuk mendukung bisnis tersebut. Karakteristik (atribut) setiap objek ditentukan beserta relasi antar objeknya.

3. Process modelling

Objek-objek data yang didefinisikan sebelumnya diubah agar bisa menghasilkan aliran informasi untuk diimplementasikan menjadi fungsi bisnis. Pengolahan deskripsi dibuat untuk menambah, merubah, menghapus, atau mengambil kembali objek data.

4. Application generation

RAD bekerja dengan menggunakan *fourth generation techniques*(4GT). Sehingga pada tahap ini sangat jarang digunakan pemrograman konvensional menggunakan bahasa pemrograman generasi ketiga (*third generation programming languages*), tetapi lebih ditekankan pada reuse komponen-komponen (jika ada) atau membuat komponen baru (jika perlu). Dalam semua kasus, alat bantu untuk otomatisasi digunakan untuk memfasilitasi pembuatan perangkat lunak.

5. Testing and turnover

Karena menekankan pada penggunaan kembali komponen yang telah ada (reuse), sebagian komponen-komponen tersebut sudah diuji sebelumnya. Sehingga mengurangi waktu testing secara keseluruhan. Kecuali untuk komponen-komponen baru.

2.7 Sistem Informasi Pemasaran

Sistem informasi pemasaran adalah sistem yang terdiri dari orang, peralatan dan prosedur untuk mengumpulkan, memilih, menganalisa, mengevaluasi dan mendistribusikan informasi yang dibutuhkan, tepat waktu dan akurat bagi para pengambil keputusan pemasaran. Sedangkan pengertian yang lain sistem informasi pemasaran adalah suatu sistem berbasis komputer yang bekerja sama dengan sistem informasi fungsional lain untuk mendukung manajemen perusahaan dalam menyelesaikan masalah yang berhubungan langsung dengan pemasaran produk perusahaan. [9]

2.8 Konsep Pemasaran

Kegiatan-kegiatan pemasaran tampil dalam semua bentuk. Satu lukisan tradisional tentang kegiatan pemasaran adalah dari segi bauran pemasaran, yang telah didefinisikan sebagai seperangkat alat pemasaran yang digunakan untuk mencapai tujuan pemasarannya. [8]

Manajer pemasaran memiliki beragam sumber daya untuk dikaryakan, tujuannya adalah mengembangkan strategi yang menerapkan sumber daya ini bagi

pemasaran barang, jasa dan gagasan perusahaan. Strategi pemasaran terdiri dari campuran unsure-unsur yang dinamakan bauran pemasaran (marketing mix) : Produk, harga, tempat dan promosi.[9]

2.9 Metode dan Cara Pemasaran Secara Online

Persaingan pasar di dunia *Online* kini begitu besar, sebenarnya ada 4 Pilar Kunci Sukses dalam pemasaran *Online*, yaitu:

1. *Product*

Sehebat apapun Anda sebagai *online* marketer bila Anda mempunyai produk yang buruk maka Anda tidak akan berhasil malahan akan menjadi boomerang bagi nama baik Anda sendiri. Dari sisi Produk kita harus memperhatikan 4 hal, yaitu:

- a. *Quality*, kita harus memiliki barang yang berkualitas baik tentunya Anda harus berani memberikan garansi terhadap barang tersebut, bila Anda berani memberikan Garansi maka orang akan lebih percaya untuk membeli produk Anda.
- b. *USP(Unique Selling Product)*, produk yang Anda jual harus memiliki keunikan, kelebihan atau *added value* dibanding produk yang lain, sehingga Anda dapat memberikan alasan mengapa orang harus membeli produk Anda.
- c. *Reasonable Price*, Produk yang Anda jual harus mempunyai harga yang masuk akal, artinya harga yang Anda jual sesuai dengan nilai yang terkandung dari barang tersebut, tidak melulu harus murah melainkan sesuai, bahkan untuk barang yang memiliki target market kelas atas jika Anda menjual terlalu murah malahan mereka curiga.
- d. *Availability of Goods*, Anda harus memiliki ketersediaan barang yang memadai sehingga bila ada pemesan mereka tidak harus menunggu terlalu lama untuk mendapatkan produk tersebut, menurut survey umumnya paling lama seseorang sabar menunggu pesanan barang yang dipesan lewat online adalah dalam 1 minggu.

2. *Service*

Sebaik apapun barang Anda dan sebaik apapun promosi Anda jika Anda tidak dapat memberikan pelayanan yang baik maka hanya dalam waktu singkat usaha Anda akan mati, 4 hal utama yang harus diperhatikan dalam pelayanan yaitu:

- a. *Quick Response*, Anda harus melayani calon pelanggan Anda dengan cepat, tentu sangat memakan biaya jika Anda harus melayani calon pembeli Anda satu persatu lewat telepon untuk itu di website Anda diperlukan fasilitas chatting online dan contact form bagi mereka yang ingin mengetahui lebih lanjut atau berniat mau membeli produk Anda.
- b. *Delivery Service*, Anda harus menggunakan layanan pengiriman yang terbaik dan terpercaya bagi para pembeli Anda, jangan sampai barang yang Anda kirim rusak apalagi hilang didalam perjalanan. Sekali Anda mengecewakan calon pembeli Anda, maka dengan sangat mudah mereka akan menyebarkannya lewat berbagai media online sehingga membuat nama Anda jatuh.
- c. *Payment System*, Anda harus memberikan kemudahan bagi calon pembeli Anda dalam melakukan pembayaran, tentunya Anda dapat menggunakan berbagai fasilitas pembayaran yang ada seperti Pay Pal, dan lain-lain.
- d. *After Sales Service*, Anda memberikan layanan purna jual yang baik kepada para pelanggan Anda. Tentunya jika produk yang mereka beli rusak, sudah seharusnya Anda mau untuk memperbaiki ataupun mengganti produk tersebut. Sehingga mereka puas terhadap pelayanan Anda, dan bagi mereka yang puas tentunya Anda dapat meminta testimoni agar Anda dapat memasangnya di *website* Anda, dengan adanya testimony akan meningkatkan kepercayaan bagi calon pembeli Anda.

3. *Promotion Strategy*

Anda harus menggunakan *Strategy* pemasaran yang tepat dalam memasarkan produk Anda, didalam pemasaran *online* ada 5 *Strategy* yang paling baik untuk dilakukan yaitu:

- a. *Web 2.0*, Sebuah *website* dapat menjual sesuatu jika memiliki pengunjung yang tinggi kita tahu bahwa sebuah *website* jika ingin memiliki *traffic* tinggi harus menggunakan sistem web 2.0 dimana disediakan fasilitas agar para pengunjung dapat melakukan interaksi didalam *website* kita.
- b. *Search Engine Optimization*, Jika seseorang ingin mencari atau bahkan membeli sesuatu maka mereka akan mencarinya di mesin pencarian seperti Google, Yahoo, ataupun MSN, untuk itu kita harus mengoptimalkan *website* kita agar alamat *website* kita dapat menempati peringkat yang baik didalam mesin pencarian.
- c. *Social Media Optimization*, Anda dapat memanfaatkan Social Media yang ada di dunia online seperti: *facebook*, *twitter*, *youtube* untuk membangun sebuah komunitas yang bertemakan produk brand Anda. Dengan demikian akan membuat produk Anda dikenal di masyarakat online.
- d. *Affiliate Program*, Anda dapat memberikan sebuah hak bagi para pelanggan Anda untuk turut menjual produk kita dan mereka mendapatkan sebuah kode link dimana jika ada pembelian yang terjadi berasal dari link tersebut maka secara otomatis kita akan memberikan komisi kepada mereka tergantung kesepakatan kita dengan mereka.
- e. *PPC Campaign*, PPC singkatan dari Pay Per Click artinya iklan yang dihitung biayanya perklik. Anda dapat menentukan *budget* maksimal untuk bulanan *PPC Campaign* Anda.

4. *Maintaining Relationships*

Anda harus dapat menjaga hubungan baik dengan pelanggan Anda, sehingga mereka akan membeli kembali kepada Anda, Ada 4 cara untuk menjaga hubungan baik dengan mereka:

- a. *Member System*, Bagi mereka yang telah membeli produk Anda maka Anda dapat memberikan mereka fasilitas sebagai member, Anda dapat memberikan sebuah kartu member sebagai penghargaan bagi mereka dan sebuah *Account Member* dimana mereka dapat mengakses halaman khusus didalam *website* Anda. Dimana kita memperlakukan mereka secara special dengan memberikan diskon khusus pada hari-hari special bagi mereka, misalnya: ulang tahun mereka, ulang tahun pernikahan, dan lain-lain.
- b. *Email Marketing/ Newsletter*, Anda dapat memaintain pelanggan Anda dengan mengirimkan email secara berkala yang isinya mengenai sesuatu hal yang berguna misalnya pengetahuan tentang produk Anda ataupun informasi tips & trik yang berhubungan dengan produk Anda.
- c. *Special Promo*, Anda harus mengadakan special promo dan sangat baik jika Anda memanfaatkan momen yang sedang berlangsung misalnya: lebaran, natal, tahun baru , hari kasih sayang (valentine), halloween dan lain-lain.
- d. *Reward Point*, Anda dapat memberikan point bagi setiap pembelian yang dilakukan pelanggan Anda, dimana jika mereka mendapat point dalam jumlah tertentu mereka dapat tukarkan dengan hadiah.

5. *E-Marketing, E-Commerce, E-Bussines*

Dalam dunia marketing non tradisional, kita seringkali menemukan beberapa istilah-istilah yang membingungkan dan seringkali saling tumpang tindih. Oleh karena itu dalam artikel ini saya mencoba untuk menjelaskan apa itu e-Marketing dan perbedaannya dengan internet-Marketing. e-Marketing atau electronic-Marketing adalah bagain dari *e-Business* yang memanfaatkan medium elektronik untuk melakukan aktivitas marketing dalam upaya mencapai tujuan marketing. Ada berbagai bentuk *e-Marketing* seperti internet-Marketing, interactive-Marketing dan mobile-Marketing. e-Business dapat diartikan sebagai pemanfaatan medium elektronik dalam kegiatan bisnis sehari-

hari. Ada beberapa tingkatan dalam pemanfaatan medium elektronik, misalnya ada perusahaan yang sepenuhnya bergantung pada medium elektronik, sementara perusahaan yang lain hanya memanfaatkan untuk tugas-tugas tertentu saja.

a. Perbedaan antara e-Business, e-Commerce and e-Marketing

E-Business adalah kesatuan yang sangat luas melibatkan suatu sistem kompleks yang memanfaatkan medium elektronik untuk melakukan atau membantu aktivitas bisnis tertentu. e-Commerce paling tepat di definisikan dalam konteks transaksi. Sebagai contoh transaksi elektronik dari uang, informasi atau hiburan masuk dalam kategori e-Commerce. Secara teknis, e-Commerce adalah bagian dari e-Business. Seperti sudah di jelaskan di atas, e-Marketing adalah bagian dari e-Business yang melibatkan medium elektronik untuk mencapai tujuan Marketing. e-Marketing berada pada level strategik sebagai pelengkap dari marketing dan strategi bisnis tradisional. Selain itu masih ada satu istilah lagi yaitu interactive-Marketing, dan secara umum dapat diartikan sebagai bagian dari e-Marketing yang melibatkan suatu level interaksi.

b. Keuntungan dari *e-Marketing*

Ada beberapa keuntungan *e-Marketing* di bandingkan dengan tradisional marketing:

1. Pengurangan biaya melalui otomatisasi dan penggunaan media elektronik.
2. Respon yang lebih cepat baik untuk end user maupun bagi marketer.
3. Adanya kemampuan untuk pengukuran dan pengumpulan data Personalisasi.
4. Memungkinkan adanya interaksi.

c. Kelemahan dari *e-Marketing*

Selain keunggulan di atas, e-Marketing juga mempunyai beberapa kelemahan :

1. Ketergantungan pada teknologi.
2. Isu keamanan dan privasi.

3. Adanya biaya pemeliharaan karena teknologi yang terus berkembang.
4. Akses teknologi yang belum merata terutama di Indonesia.
5. Transparansi harga sehingga mengakibatkan meningkatnya kompetisi harga.
6. Kompetisi global.

Selain itu, berdasarkan *survey* ada beberapa kendala bagi perusahaan untuk menerapkan *e-Marketing*, diantaranya : kurangnya kemampuan untuk mengukur impact dari program *e-Marketing*, kemampuan internal yang terbatas untuk menjalankannya dan kesulitan untuk meyakinkan manajer senior.

2.10 Koperasi

Koperasi adalah badan usaha yang beranggotakan orang atau badan hukum yang berlandaskan pada asas kekeluargaan dan demokrasi ekonomi. Kegiatan usaha koperasi merupakan penjabaran dari UUD 1945 pasal 33 ayat (1). Dengan adanya penjelasan UUD 1945 Pasal 33 ayat (1) koperasi berkedudukan sebagai soko guru perekonomian nasional dan sebagai bagian yang tidak terpisahkan dalam sistem perekonomian nasional. [11]

Sebagai salah satu pelaku ekonomi, koperasi merupakan organisasi ekonomi yang berusaha menggerakkan potensi sumber daya ekonomi demi memajukan kesejahteraan anggota. Karena sumber daya ekonomi tersebut terbatas, dan dalam mengembangkan koperasi harus mengutamakan kepentingan anggota, maka koperasi harus mampu bekerja seefisien mungkin dan mengikuti prinsip-prinsip koperasi dan kaidah-kaidah ekonomi.

Koperasi adalah suatu kumpulan orang – orang untuk bekerja sama demi kesejahteraan bersama. Koperasi Indonesia adalah organisasi ekonomi rakyat yang berwatak social dan beranggotakan orang – orang, badan - badan hukum koperasi yang merupakan tata susunan ekonomi sebagai usaha bersama berdasar atas asas kekeluargaan.[12]