

## BAB II TINJAUAN PUSTAKA

### 2.1 Konsep Sistem Informasi

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut. Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif. Suatu sistem dapat terdiri dari beberapa bagian-bagian sistem atau subsistem. Sebagai contoh, sistem komputer dapat terdiri subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat berisi subsistem-subsistem yang lainnya atau terdiri dari komponen-komponen pendukung sistem itu sendiri. Subsistem perangkat keras dapat terdiri dari alat masukan, alat pemroses, alat keluaran, dan media penyimpanan. Subsistem-subsistem yang ada saling berinteraksi dan saling berhubungan membentuk suatu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai. Interaksi dari subsistem-subsistem tersebut terjadi demikian rupa sehingga dicapai suatu kesatuan yang terpadu dan terintegrasi.

Model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

a. Komponen Sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup sistem tersebut.

d. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer dan “data” adalah *signal input* untuk diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi.

g. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

Informasi adalah data yang diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan keputusan berulang sederhana sampai keputusan strategis jangka panjang. Fungsi utama informasi adalah menambah pengetahuan,

informasi yang disampaikan kepada pemakai mungkin merupakan hasil data yang sudah diolah menjadi sebuah keputusan. Informasi dapat dikelompokkan menjadi 3 bagian, yaitu :

- a. Informasi Strategi, informasi ini digunakan untuk mengambil keputusan jangka panjang, yang mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.
- b. Informasi Taktis, informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.
- c. Informasi Teknis, informasi ini digunakan untuk keperluan operasional sehari-hari seperti informasi persediaan stok, retur penjualan, dan laporan kas harian.

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [2].

## 2.2 Basis Data

Basis data terdiri atas 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul, sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

Sebagai satu kesatuan istilah, Basis Data (*Database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah
2. Kumpulan data yang saling berhubungan yang disimpan secara bersamaan sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Di dalam sebuah *disk*, basis data dapat diciptakan dan dapat pula ditiadakan. Di dalam sebuah *disk*, juga dapat menempatkan beberapa basis data. Sementara dalam sebuah basis data, dapat menempatkan satu atau lebih tabel. Pada tabel inilah sesungguhnya data disimpan dan ditempatkan. Setiap basis data umumnya dibuat untuk mewakili sebuah semesta data

yang spesifik. Karena itu, operasi-operasi dasar yang dapat dilakukan berkenaan dengan basis data dapat meliputi :

1. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru.
2. Penghapusan basis data (*drop database*), yang identik dengan perusakan lemari arsip (sekaligus beserta isinya, jika ada).
3. Pembuatan tabel baru ke suatu basis data (*create table*), yang identik dengan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.
4. Penghapusan tabel dari suatu basis data (*drop table*), yang identik dengan perusakan map arsip lama yang ada di sebuah lemari arsip.
5. Penambahan/pengisian data baru ke sebuah tabel di sebuah basis data (*insert*), yang identik dengan penambahan lembaran arsip ke sebuah map arsip.
6. Pengambilan data dari sebuah tabel (*query*), yang identik dengan pencarian lembaran arsip dari sebuah map arsip.
7. Pengubahan data dari sebuah tabel (*update*), yang identik dengan perbaikan isi lembaran arsip yang ada di sebuah map arsip.
8. Penghapusan data dari sebuah tabel (*delete*), yang identik dengan penghapusan sebuah lembaran arsip yang ada di sebuah map arsip [3].

### 2.3 Metodologi RAD (Rapid Application Development)

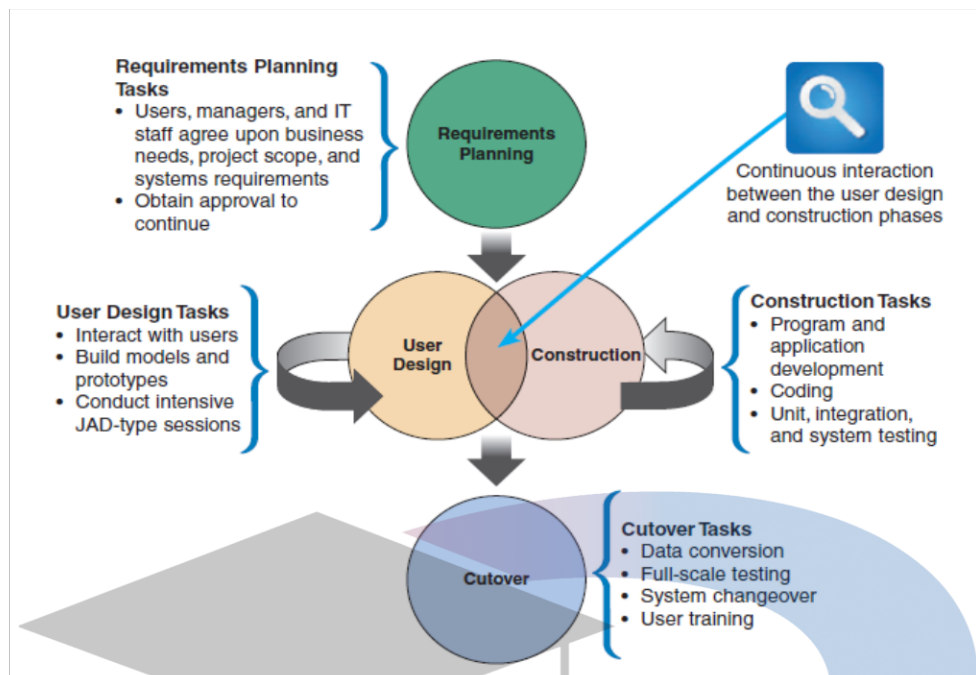
Rapid Application Development (RAD) adalah teknik berbasis tim yang mempercepat pengembangan sistem informasi dan menghasilkan sistem informasi yang berfungsi. Seperti JAD, RAD menggunakan pendekatan kelompok tetapi berjalan lebih jauh. Sedangkan produk akhir JAD adalah model persyaratan, produk akhir RAD adalah sistem informasi baru. RAD adalah metodologi lengkap, dengan siklus hidup empat *fase* yang sejajar dengan fase SDLC tradisional. Perusahaan menggunakan RAD untuk mengurangi biaya dan waktu pengembangan dan meningkatkan kemungkinan keberhasilan RAD sangat bergantung pada *prototype* dan keterlibatan pengguna. Proses RAD memungkinkan pengguna untuk memeriksa model kerja sedini mungkin, menentukan apakah itu memenuhi kebutuhan mereka, dan menyarankan perubahan yang diperlukan. Berdasarkan input pengguna, *prototype* dimodifikasi dan proses interaktif berlanjut sampai sistem benar-benar dikembangkan dan pengguna merasa puas. Tim proyek menggunakan alat kasus untuk membangun *prototype* dan membuat aliran dokumentasi berkelanjutan [4].

### 2.3.1 Fase dan Aktivitas RAD :

Model RAD terdiri dari empat fase, yaitu :

- a. Perencanaan Persyaratan : Tahap perencanaan kebutuhan menggabungkan elemen dari perencanaan sistem dan fase analisis sistem SDLC. Pengguna, manajer, dan anggota staf TI mendiskusikan dan menyetujui kebutuhan bisnis, ruang lingkup proyek, batasan, dan persyaratan sistem. Fase perencanaan kebutuhan berakhir ketika tim setuju pada isu-isu kunci dan mendapatkan otorisasi manajemen untuk melanjutkan.
- b. Desain Pengguna : Selama fase desain pengguna, pengguna berinteraksi dengan analis sistem dan mengembangkan model dan *prototype* yang mewakili semua proses sistem, *input*, dan *output*. Kelompok RAD atau subkelompok biasanya menggunakan kombinasi teknik JAD dan alat KASUS untuk menerjemahkan kebutuhan pengguna ke dalam model kerja. Desain pengguna adalah proses interaktif dan berkelanjutan yang memungkinkan pengguna untuk memahami, memodifikasi, dan akhirnya menyetujui model kerja dari sistem yang memenuhi kebutuhan mereka.
- c. Kontruksi : Fase konstruksi berfokus pada program dan aplikasi tugas pengembangan yang serupa dengan SDLC. Namun, di RAD, pengguna terus berpartisipasi dan masih dapat menyarankan perubahan atau peningkatan karena layar atau laporan aktual dikembangkan.
- d. *Cutover* : Fase *cutover* menyerupai tugas akhir dalam implementasi SDLC fase, termasuk konversi data, pengujian, peralihan ke sistem baru, dan pengguna latihan. Dibandingkan dengan metode tradisional, seluruh proses dikompresi. Sebagai Hasilnya, sistem baru dibangun, dikirimkan, dan ditempatkan dalam operasi lebih cepat [4].

UNIVERSITAS  
MIKROSKIL



Gambar 2. 1 Fase metode RAD

### 2.3.2 Tujuan RAD

Tujuan utama dari semua pendekatan RAD adalah untuk memotong waktu dan biaya pengembangan dengan melibatkan pengguna di setiap fase pengembangan sistem. Karena ini adalah proses yang berkelanjutan, RAD memungkinkan tim pengembangan untuk melakukan modifikasi yang diperlukan dengan cepat, seiring dengan perkembangan desain. Pada saat anggaran perusahaan ketat, sangat penting untuk membatasi biaya perubahan yang biasanya terjadi dalam jadwal pengembangan yang panjang dan berlarut-larut. Selain keterlibatan pengguna, tim RAD yang sukses harus memiliki sumber daya, keterampilan, dan dukungan manajemen teknologi informasi. Karena ini adalah proses dinamis yang digerakkan oleh pengguna, RAD sangat berharga ketika perusahaan membutuhkan sistem informasi untuk mendukung fungsi bisnis baru. Dengan mendapatkan input pengguna dari awal, RAD juga membantu tim pengembangan merancang sistem yang membutuhkan antarmuka pengguna yang sangat interaktif atau kompleks [4].

### 2.3.3 Keuntungan dan Kerugian RAD

Adapun keuntungan dan kerugian yang dimiliki metode RAD ialah :

- Keuntungan utamanya adalah bahwa sistem dapat dikembangkan lebih cepat dengan penghemat biaya yang signifikan.

- b. Kerugiannya adalah bahwa RAD menekankan pada mekanisme sistem itu sendiri dan tidak menekankan kebutuhan bisnis strategis perusahaan. Resikonya adalah bahwa sistem mungkin bekerja dengan baik dalam jangka pendek, tetapi perusahaan dalam jangka panjang tujuan untuk sistem mungkin tidak terpenuhi. Kerugian potensial lainnya adalah bahwa siklus waktu yang dipercepat memungkinkan lebih sedikit waktu untuk mengembangkan kualitas, konsistensi, dan standar desain. RAD bisa menjadi alternatif yang menarik, jika sebuah organisasi memahami resiko yang mungkin terjadi [4].

## 2.4 Teknik Pengembangan Sistem

### 2.4.1 Use Case

*Use case diagram* adalah *diagram* yang menggambarkan interaksi antara sistem internal dengan sistem eksternal pengguna. Dengan kata lain, secara garis besar menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem. *Use case* adalah urutan langkah-langkah yang secara tindakan saling terkait baik terotomatisasi maupun secara manual untuk tujuan melengkapinya satu tugas bisnis tunggal. *Actor* adalah segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi. *Association* adalah hubungan antar pelaku / *actor* dengan *use case* dimana terjadi interaksi diantara mereka. *Extension use case* adalah *use case* yang lebih kompleks untuk menyederhanakan masalah orisinal dan arena itu memperluas fungsinya. *Abstract use case* adalah *use case* yang mengurangi redundansi antara dua atau lebih *use case* dengan menggabungkan langkah-langkah yang biasa ditemukan pada *use case* tersebut.

Pemodelan *use case* merupakan proses pemodelan fungsi-fungsi sistem dalam konteks peristiwa-peristiwa bisnis, siapa yang mengawalinya dan bagaimana sistem itu merespon hal tersebut. Ada dua alat utama yang digunakan saat menyajikan pemodelan *use case*. Pertama dengan *use case narrative* dan yang kedua adalah *use case diagram*.


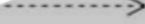








*Use case narrative* adalah deskripsi tekstual kegiatan bisnis dan menjelaskan bagaimana pengguna akan berinteraksi dengan sistem untuk menyelesaikan suatu tugas, sedangkan *use case diagram* adalah diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal dan pelaku.

Pemodelan *use case* memberikan manfaat :

1. Menyediakan *tool* untuk menggambarkan persyaratan fungsional.
2. Membantu menyusun ulang lingkup sistem menjadi bagian-bagian yang dapat dikelola.

3. Menyediakan alat komunikasi dengan para *stakeholder* yang berhubungan dengan fungsionalitas sistem.
4. Memberikan cara bagaimana mengidentifikasi, menetapkan, mengontrol dan mengelola kegiatan pengembangan sistem [5].

Berikut ini adalah notasi *Use Case* :

SIMBOL	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya ( <i>sinergi</i> ).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Gambar 2. 2 Notasi *Use Case Diagram*

Selama pemodelan persyaratan, analisis sistem dan pengguna bekerja bersama untuk mendokumentasikan persyaratan dan memodelkan fungsi sistem. Diagram *use case* secara visual menggambarkan interaksi antara pengguna dan sistem informasi. Dalam diagram *use case*, pengguna menjadi aktor, dengan peran spesifik yang menggambarkan bagaimana



berinteraksi dengan sistem. Analisis sistem dapat menggambarkan diagram *use case freehand* atau menggunakan alat CASE yang mengintegrasikan kasus penggunaan ke dalam desain sistem secara keseluruhan.

<b>Name of Use Case:</b>	Credit card validation process
<b>Actor:</b>	Customer
<b>Description:</b>	Describes the credit card validation process
<b>Successful Completion:</b>	<ol style="list-style-type: none"> <li>1. Customer clicks the input selector and enters credit card number and expiration date</li> <li>2. System verifies card</li> <li>3. System sends authorization message</li> </ol>
<b>Alternative:</b>	<ol style="list-style-type: none"> <li>1. Customer clicks the input selector and enters credit card number and expiration date</li> <li>2. System rejects card</li> <li>3. System sends rejection message</li> </ol>
<b>Precondition:</b>	Customer has selected at least one item and has proceeded to checkout area
<b>Postcondition:</b>	Credit card information has been validated Customer can continue with order
<b>Assumptions:</b>	None

Gambar 2.3 Contoh Narasi Use Case

Sebagai contoh, gambar 2.3 menunjukkan tabel narasi yang mendokumentasikan kasus penggunaan validasi kartu kredit [4].

## 2.4.2 Prototyping

*Prototyping* menghasilkan versi awal yang cepat dibangun dari sistem informasi yang diusulkan, yang disebut *prototype*. *Prototyping*, yang melibatkan repetitif rangkaian analisis, desain, pemodelan dan pengujian adalah teknik umum yang dapat digunakan untuk mendesain apa pun dari rumah baru ke jaringan komputer. *Prototyping* memungkinkan pengguna untuk memeriksa suatu model yang secara akurat mewakili sistem *input*, *ouput*, *interface*, dan proses. Pengguna dapat “menguji-drive” model di lingkungan bebas resiko dan menyetujui atau meminta perubahan. Dalam beberapa situasi, *prototype* berkembang menjadi versi final dari sistem informasi. Dalam kasus lain, *prototype* hanya ditujukan untuk memvalidasi persyaratan pengguna dan dibuang sesudahnya.

Analisis sistem juga menggunakan *prototype* untuk memverifikasi persyaratan pengguna, setelah *prototype* dibuang dan implementasi berlanjut. Pendekatan ini disebut desain *prototyping*, atau *prototyping* sekali pakai. Dalam hal ini, tujuan *prototyping* lebih terbatas, tetapi tidak kurang penting. Produk akhir dari *prototype* desain adalah model yang disetujui oleh pengguna yang mendokumentasikan dan memberi tolak ukur fitur sistem yang sudah jadi. Desain *prototyping* memungkinkan untuk menangkap *input* dan persetujuan

pengguna sambil terus mengembangkan sistem dalam kerangka SDLC. Analisis sistem biasanya menggunakan desain *prototype* ketika mereka membangun *output*, *input* dan antarmuka pengguna [4].

Manfaat *Prototyping* sebagai berikut :

- a. Pengguna dan pengembang sistem dapat menghindari kesalahpahaman.
- b. Pengembang sistem dapat membuat spesifikasi yang akurat untuk sistem selesai berdasarkan *prototype*.
- c. Manajer dapat mengevaluasi model kerja secara lebih efektif daripada spesifikasi kertas.
- d. Analisis sistem dapat menggunakan *prototype* untuk mengembangkan prosedur pengujian dan pelatihan sebelum sistem selesai tersedia.
- e. Pembuatan *prototype* mengurangi resiko dan potensi eksposur keuangan yang terjadi ketika sistem yang selesai gagal mendukung kebutuhan bisnis.

Meskipun kebanyakan analisis sistem percaya bahwa keuntungan dari *prototyping* jauh lebih besar daripada kerugian, masalah potensial berikut harus dipertimbangkan:

- a. Laju perkembangan yang cepat dapat menciptakan masalah kualitas, yang tidak ditemukan sampai sistem selesai beroperasi.
- b. Persyaratan sistem lainnya, seperti keandalan dan pemeliharaan, tidak bisa diuji secara memadai menggunakan *prototype*.
- c. Dalam sistem yang sangat kompleks, *prototype* dapat menjadi berat dan sulit untuk dikelola.

Seorang klien atau pengguna mungkin ingin mengadopsi *prototype* dengan sedikit atau tanpa perubahan, keliru berpikir bahwa *prototype* akan memenuhi kebutuhan mereka meskipun mungkin memerlukan penyesuaian lebih lanjut, yang mengarah ke peningkatan biaya pemeliharaan kemudian di SDLC [4].

## 2.5 Website

*Website* adalah bagian dari internet yang mendukung multimedia dan terdiri dari kumpulan dokumen yang ditautkan. Untuk mendukung multimedia, *website* bergantung pada *Hypertext transfer protocol* (HTTP), yang merupakan seperangkat aturan untuk bertukar teks, grafik, suara, video dan file multimedia lainnya. Dokumen yang ditautkan, atau halaman informasi, di *website* dikenal sebagai halaman situs web. Karena situs web mendukung teks, grafik, suara dan video, halaman situs web dapat menyertakan elemen multimedia ini. Situs

web adalah kumpulan halaman situs web terkait yang dibuat dan dikelola oleh individu, perusahaan, lembaga pendidikan, atau organisasi lain.

Situs internet, juga dikenal sebagai situs web, adalah situs yang umumnya tersedia untuk umum, individu, kelompok, perusahaan, dan lembaga pendidikan menggunakan situs web untuk berbagai tujuan. Perusahaan menggunakan situs web untuk mengiklankan atau menjual produk dan layanan mereka di seluruh dunia, serta memberikan dukungan teknis dan produk untuk pelanggan mereka. Banyak situs web perusahaan juga mendukung *e-commerce*, yang merupakan pembelian dan penjualan barang dan jasa di internet. Dengan menggunakan teknologi *e-commerce*, situs-situs *website* ini memungkinkan pelanggan untuk menelusuri kubah produk, toko perbandingan, dan memesan produk secara *online* [4].

## 2.6 Pengembangan Front End

### 2.6.1 HTML (HyperText Markup Language)

HTML mendefinisikan struktur dan gaya dokumen, termasuk tajuk utama, posisi grafik, tabel, dan pemformatan teks. Sejak diperkenalkan, *browser* utama telah terus menambahkan fitur ke HTML untuk memungkinkan *programmer* untuk lebih lanjut memperbaiki tata letak halaman mereka. Namun, beberapa peningkatan *browser* hanya dapat berfungsi di *browser* satu perusahaan. Setiap kali membangun situs *e-commerce*, harus berhati-hati bahwa halaman dapat dilihat oleh *browser* utama, bahkan versi *browser* yang ketinggalan zaman. Halaman web HTML dapat di-cret dengan editor teks apa pun, seperti Notepad atau Wordpad, menggunakan Ms word (cukup simpan dokumen kata sebagai halaman web), atau salah satu dari beberapa alat pengembangan halaman web seperti Ms expression Web atau adobe Dreamweaver CC [6].

Beberapa fitur yang tersedia pada HTML5 antara lain sebagai berikut [7]:

- a. *Canvas*, memungkinkan pembuatan gambar dalam kanvas. Jadi, gambar tidak lagi diambil dari gambar utuh, melainkan bisa disusun sendiri.
- b. *Header*, berguna untuk menyatakan suatu judul, yang bisa diisi dengan logo dan nama perusahaan.
- c. *Footer*, merupakan kebalikan dari header. Sebagai catatan kaki, elemen ini berguna untuk menaruh informasi di bagian bawah halaman web.
- d. *Time*, elemen ini berguna untuk menyajikan informasi tentang waktu.
- e. *Audio*, memungkinkan penyajian *player* untuk memutar suara.
- f. *Video*, memungkinkan *player* untuk memainkan film.

Secara prinsip, fitur pada HTML dapat dikelompokkan ke dalam [7]:

- a. Struktur halaman.
- b. Presentasi visual.
- c. Peranti penyaji gambar.
- d. Pendukung media.
- e. Peningkatan koneksi dengan JavaScript.

## 2.6.2 Bootstrap

Bootstrap adalah paket aplikasi yang siap dipakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, Bootstrap adalah template desain web dengan fitur plus. Bootstrap diciptakan untuk mempermudah proses desain web bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup dengan bermodalkan pengetahuan dasar mengenai HTML dan CSS, sudah dapat menggunakan Bootstrap.

Paket Bootstrap berisi sekumpulan file CSS, font, dan JavaScript yang siap diintegrasikan ke sebuah dokumen HTML. Dokumen HTML yang dihasilkan pun secara dinamis akan tampil dalam *layout* yang disesuaikan dengan ukuran layar piranti pengunjung. Daya tarik Bootstrap terletak pada kemudahan dan kepraktisan penggunaannya. Tata *layout* dan pewarnaannya juga bersih, simpel, indah, dan berkesan modern. Selain itu, mengingat akses web via piranti *mobile* semakin meningkat tajam, maka pendekatan desain web *responsive* dan *mobile-friendly* yang ditawarkan Bootstrap pun juga menjadi solusi praktis dan mudah bagi para pengembang situs web.

Bootstrap merupakan proyek *open source* dan tentunya pemakaiannya bersifat gratis. Saat ini, Bootstrap menjadi aplikasi paling populer di situs layanan hosting software *open source* GitHub.com. Jutaan *website* telah menggunakan Bootstrap sebagai framework desainnya. Situs pemantau teknologi web Builtwith.com mencatat, lebih dari 6 juta *website* aktif menggunakan Bootstrap. Bootstrap awalnya bernama *Twitter Blueprint*. Tool ini mulai dikembangkan pada tahun 2010 oleh tim desainer dan pengembang di Twitter, yaitu Mark Otto dan Jacob Thornton. Setelah digunakan sebagai *framework* internal di situs Twitter, setahun kemudian tool ini dirilis ke publik sebagai proyek *open source* menggunakan nama Bootstrap. Karena latar belakang sejarah pengembangannya, Bootstrap hingga kini masih sering disebut dengan *Twitter Bootstrap* [8].

## 2.7 Pengembangan Back End

### 2.7.1 PHP (PHP Hypertext Preprocessor)

PHP singkatan dari PHP *Hypertext Preprocessor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan web dapat dibuat dinamis sehingga *maintenance* situs web tersebut menjadi lebih mudah dan efisien. PHP merupakan *software Open-Source* yang diserbarkan dan dilisensikan secara gratis serta dapat di-*download* secara bebas dari situs resminya <http://www.php.net>.

PHP diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *open source*. Oleh karena itu, ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis, kemudian menambah kemampuan PHP 1.0 dan meluncurkan PHP 2.0. PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa *script* sejenis. PHP difokuskan pada pembuatan *script server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh CGI (*Common Gateway Interface*), seperti mengumpulkan data dari *form*, menghasilkan isi halaman web dinamis, dan kemampuan mengirim serta menerima cookies, bahkan lebih daripada kemampuan CGI.

PHP dapat digunakan pada semua sistem operasi, antara lain Linux, Unix (termasuk variannya HP-UX, Solaris, dan OpenBSD), Microsoft Windows, Mac OS X, RISC OS. PHP juga mendukung banyak *web server*, seperti Apache, Microsoft Internet Information Server (IIS), Personal Web Server (PWS), Netscape and iPlanet servers, masih banyak lagi lainnya, bahkan PHP dapat bekerja sebagai suatu CGI processor. PHP tidak terbatas pada hasil keluaran HTML. PHP juga memiliki kemampuan untuk mengolah keluaran gambar, file PDF, dan *movies Flash*. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya [9].

### 2.7.2 Laravel

Laravel adalah sebuah kerangka kerja *open source* yang diciptakan oleh Taylor Otwell. Laravel merupakan *framework bundle*, migrasi dan artisan CLI (*Command Line Interface*) yang menawarkan seperangkat alat dan arsitektur aplikasi yang menggabungkan banyak fitur terbaik dari kerangka kerja seperti *Codeigniter*, *Yii*, *ASP.NET MVC*, *Ruby on Rails*, *Sinatra* dan *lain-lain*. Laravel memiliki seperangkat fitur yang akan meningkatkan kecepatan pengembangan situs web. Alur kerja laravel menggunakan MVC, MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti manipulasi *data*, *controller* dan *user interface*.

Ada 5 konsep arsitektur pada framework laravel yang mempunyai masing-masing fungsi diantaranya :

1. *Routes* merupakan sebagai pemberi akses pada setiap *request* sesuai alur yang telah di tentukan. Didalam *routes* memiliki 4 instruksi standar, yakni:
  - a. *Get* : berfungsi untuk memanggil *request*.
  - b. *Put* : berfungsi untuk mengambil data sesuai *request*.
  - c. *Post* : berfungsi untuk menambahkan data sesuai *request*.
  - d. *Delete* : berfungsi untuk menghapus data sesuai *request*.
2. *Controller* merupakan bagian yang menjadi penghubung antara *model* dan *view*. *Controller* memiliki perintah-perintah yang berfungsi untuk memproses bagaimana data ditampilkan dari *Model* ke *View* atau sebaliknya. Struktur *controller* pada penulisan kode program di laravel yakni :
  - a. *Index* : Berfungsi untuk menampilkan data keseluruhan.
  - b. *Create* : Berfungsi untuk memanggil form yang berisikan kolom inputan.
  - c. *Store* : Berfungsi untuk menyimpan data ke dalam *table*.
  - d. *Show* : Berfungsi untuk menampilkan data sesuai Id.
  - e. *Edit* : Memanggil data sesuai Id yang berisikan form inputan untuk proses *update*.
  - f. *Update* : Berfungsi untuk mengupdate data pada *table*.
  - g. *Delete* : Berfungsi untuk menghapus data sesuai Id.
3. *Model* merupakan sekumpulan data yang memiliki fungsi-fungsi untuk mengelola suatu *table* pada sebuah *database*. Struktur pemodelan data pada laravel yakni memiliki fungsi yang terdiri dari *table*, *primaryKey* dan *fillable*. Dimana ketiga fungsi tersebut harus di *protected*. Pada bagian *table* harus diisi dengan nama *table* yang sesuai pada *database*, di bagian *primaryKey* harus diisi sesuai *primary key* pada *table* tersebut dan pada bagian *fillable* diisi dengan bagian-bagian yang mencakup dalam *table* tersebut.
4. *View* merupakan file yang berisi kode html (*HyperText Markup Language*) yang berfungsi untuk menampilkan suatu data ke dalam *browser*. *Format view* pada laravel harus menggunakan istilah blade, contohnya seperti: *view.blade.php*.
5. *Migrations* merupakan proses perancangan suatu *table*, dalam hal ini *migrations* berfungsi sebagai *blueprint database* atau dapat diistilahkan sebagai penyedia sistem kontrol untuk skema *database*.

*Framework* laravel memiliki keunggulan tersendiri yang menjadikannya lebih baik dari pada *framework* lainnya, berikut ini merupakan kelebihan dari laravel yaitu, *performance* lebih cepat, *reload* data lebih stabil, memiliki keamanan data, menggunakan fitur canggih

seperti *blade* menggunakan konsep HMVC (*Hierarchical Model View Controller*), tersedianya *library-library* yang sudah siap untuk digunakan dan adanya fitur pengelolaan *migrations* untuk pembuatan skema *table* pada *database* [10].

## 2.8 E-Commerce

*E-Commerce* merupakan aktivitas pembelian dan penjualan melalui jaringan internet di mana pembeli dan penjual tidak bertemu secara langsung, melainkan berkomunikasi melalui media internet. *E-Commerce* memiliki berbagai macam jenis transaksi dalam menerapkan sistemnya. Jenis-jenis transaksi *E-Commerce* di antaranya sebagai berikut :

1. *Collaborative Commerce* yaitu kerjasama secara elektronik antara rekan bisnis. Kerjasama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (*supply chain*).
2. *Business-to-Business* (B2B) tipe ini meliputi transaksi antarorganisasi yang dilakukan di *electronic market*. *Business to Business* memiliki karakteristik:
  - a. *Trading Partners* yang sudah diketahui dan umumnya memiliki hubungan yang cukup lama. Informasi hanya dipertukarkan dengan partner tersebut. Dikarenakan sudah mengenal rekan komunikasi, jenis informasi yang dikirimkan dapat disusun sesuai dengan kebutuhan dan kepercayaan (*trust*).
  - b. Pertukaran data (*data exchange*) berlangsung berulang-ulang dan secara berkala, misalnya setiap hari, dengan format data yang sudah disepakati bersama. Dengan kata lain, servis yang digunakan sudah tertentu. Hal ini memudahkan pertukaran data untuk dua entri yang menggunakan standar yang sama.
  - c. Salah satu pelaku dapat melakukan inisiatif untuk mengirimkan data, tidak harus menunggu partnernya.
  - d. Model yang umum digunakan adalah *peer-to-peer*, dimana *processing intelligence* dapat didistribusikan pada kedua pelaku bisnis.
3. *Business-to-Consumer* (B2C) yaitu penjual adalah suatu organisasi dan pembeli adalah individu. B2C memiliki karakteristik sebagai berikut:
  - a. Terbuka untuk umum, dimana informasi disebarkan ke umum.
  - b. Servis yang diberikan bersifat umum.
  - c. Servis diberikan berdasarkan permohonan, konsumen melakukan inisiatif dan produser harus siap memberikan respon sesuai dengan permohonan.

4. *Consumer-to-Business (C2B)* yaitu konsumen memberitahukan kebutuhan atas suatu produk atau jasa tertentu, dan para pemasok bersaing untuk menyediakan produk atau jasa tersebut ke konsumen.
5. *Customer-to-Customer (C2C)* yaitu konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dalam C2C seorang menjual produk atau jasa ke orang lain. Dapat juga disebut sebagai pelanggan ke pelanggan, yaitu orang yang menjual produk dan jasa ke satu sama lain [6].

### 2.8.1 E-Retailing

Retail *online* mungkin adalah sektor *e-commerce* yang paling terkenal. Selama masa lalu, sektor ini telah mengalami pertumbuhan eksplosif dan kegagalan spektakuler. Banyak dari perusahaan-perusahaan *online* murni-bermain awal yang memelopori pasar ritel gagal. Pengusaha dan investor mereka secara serius salah menilai faktor-faktor yang diperlukan untuk berhasil di pasar tersebut. Tetapi orang yang selamat dari periode awal ini muncul lebih kuat, dan bersama dengan pedagang *offline* umum dan pedagang khusus, serta perusahaan baru, ruang *e-tail* tumbuh sangat cepat dan meningkatkan jangkauan dan ukurannya.

Visi *E-commerce* ritel, pada tahun-tahun awal *e-commerce* secara harfiah ribuan pengecer berbasis situs web wirausaha tertarik ke pasar untuk barang-barang ritel, hanya karena itu adalah salah satu peluang pasar terbesar dalam perekonomian AS. Banyak pengusaha awalnya percaya itu mudah untuk memasuki pasar ritel. Para penulis awal meramalkan bahwa industri ritel akan mengalami revolusi, secara harfiah "diledakkan menjadi potongan-potongan" seperti diwakili oleh dua konsultan dalam sebuah buku sekolah Harvard Business yang terkenal (Evans dan Wurster, 2000). Dasar dari revolusi ini adalah empat kali lipat. Pertama, karena internet sangat mengurangi biaya pencarian, konsumen akan menggunakan web untuk mencari produk dengan biaya terendah. Beberapa hasil akan mengikuti. Konsumen akan semakin hanyut ke situs web untuk berbelanja dan membeli, dan hanya ritel *online* berbiaya rendah, layanan tinggi, berkualitas dan biaya-didorong tidak didorong oleh nilai atau merek yang dirasakan, keduanya merupakan faktor non-rasional.

Kedua, diasumsikan bahwa biaya memasuki pasar ritel *online* jauh lebih sedikit daripada yang dibutuhkan untuk mendirikan toko fisik, dan bahwa pedagang *online* secara inheren lebih efisien dalam pemasaran dan memenuhi pesanan toko offline. Biaya pendirian Web yang kuat dianggap sangat kecil dibandingkan dengan biaya gudang, pusat pemenuhan, dan toko fisik. Tidak akan ada kesulitan membangun entri pesanan canggih, *shopping cart*, dan sistem pemenuhan karena teknologi ini sudah dikenal, dan biaya teknologi turun 50%



setiap tahun. Bahkan biaya memperoleh konsumen dianggap jauh lebih rendah karena mesin pencari yang hampir bisa langsung menghubungkan pelanggan ke vendor *online*.

Ketiga, ketika harga jatuh, pedagang fisik *offline* fisik tradisional akan dipaksa keluar dari bisnis. Perusahaan-perusahaan wirausaha baru seperti Amazon akan menggantikan toko-toko tradisional. Diperkirakan bahwa jika pedagang *online* tumbuh sangat cepat, mereka akan memiliki keuntungan penggerak pertama dan mengunci perusahaan tradisional yang lebih tua yang terlalu lambat untuk memasuki pasar *online*.

Keempat, di beberapa industri seperti elektronik, pakaian, dan konten digital - pasar akan disintermediasi sebagai produsen atau distributor mereka masuk untuk membangun hubungan langsung dengan pelanggan, menghancurkan perantara ritel atau perantara. Dalam skenario ini, saluran ritel tradisional seperti toko fisik, panitera penjualan, dan tenaga penjualan akan digantikan oleh saluran tunggal yang dominan seperti situs web.

Banyak diprediksi, di sisi lain semacam *hypermediation* berdasarkan konsep film virtual di mana pengecer *online* akan mendapatkan keuntungan atas pedagang offline yang didirikan dengan melambungkan nama merek *online* yang menarik jutaan pelanggan, dan *outsourcing* pergudangan yang mahal dan pemenuhan pesanan fungsi-konsep asli Amazon dan Drugstore.com. Ternyata, pelanggan tidak terutama digerakkan oleh harga saat berbelanja di internet tetapi mempertimbangkan nama merek, kepercayaan, keandalan, waktu pengiriman, kenyamanan, kemudahan penggunaan, dan di atas semua "pengalaman," setidaknya sama pentingnya sebagai harga (Brynjolfsson, Dick, dan Smith, 2004).

Namun, internet telah menciptakan tempat yang sepenuhnya baru untuk perusahaan omni-channel (yang menjual produk melalui berbagai saluran dan mengintegrasikan toko fisik mereka dengan situs web dan *platform* mereka), dan dalam beberapa kasus, *Internet* telah mendukung pengembangan dari pedagang *online* hanya main-main, baik merchandiser umum maupun pengecer khusus. Seperti yang diperkirakan, ritel *online* memang menjadi saluran ritel yang paling cepat berkembang dan paling dinamis dalam pengertian inovasi saluran. Web telah menciptakan pasar baru bagi jutaan pelanggan untuk berbelanja dengan nyaman. *Internet* dan Web terus menyediakan peluang baru untuk perusahaan yang sepenuhnya baru menggunakan model bisnis baru dan produk *online* baru - seperti Blue Nile, seperti yang dijelaskan sebelumnya. Saluran *online* dapat bertentangan dengan saluran lain milik pedagang, seperti tenaga penjualan langsung, toko fisik, dan pesanan pos, tetapi konflik multichannel ini dapat dikelola dan diubah menjadi kekuatan. Retail *online* memiliki keuntungan dan tantangan, adapun keuntungan dan tantangan tersebut ialah [6]:

1. Keuntungan :
  - a. Menurunkan biaya rantai pasokan dengan menggabungkan permintaan pada daya beli tunggal dan meningkat.
  - b. Turunkan biaya distribusi menggunakan situs web daripada toko fisik.
  - c. Kemampuan untuk menjangkau dan melayani kelompok pelanggan yang terdistribusi secara geografis jauh lebih besar.
  - d. Kemampuan untuk bereaksi dengan cepat terhadap selera dan permintaan pelanggan.
  - e. Kemampuan untuk mengubah harga hampir seketika.
  - f. Kemampuan untuk secara cepat mengubah presentasi visual barang.
  - g. Menghindari biaya pemasaran langsung dari katalog dan surat fisik.
  - h. Peningkatan peluang untuk personalisasi, kustomisasi.
  - i. Kemampuan untuk meningkatkan informasi dan pengetahuan yang disampaikan kepada pelanggan.
  - j. Kemampuan untuk menurunkan biaya transaksi pasar keseluruhan konsumen.
2. Tantangan:
  - a. Kekhawatiran konsumen tentang keamanan transaksi.
  - b. Keprihatinan konsumen tentang privasi informasi pribadi yang diberikan kepada situs web.
  - c. Keterlambatan pengiriman barang jika dibandingkan dengan belanja toko.
  - d. Ketidaknyamanan yang terkait dengan pengembalian barang yang rusak atau ditukarkan.
  - e. Mengatasi kurangnya kepercayaan pelanggan terhadap nama merek *online*.
  - f. Biaya tambahan untuk fotografi *online*, video, dan presentasi animasi.
  - g. Biaya pemasaran *online* untuk pencarian, *e-mail*, dan *display*.
  - h. Menambahkan kerumitan pada penawaran produk dan layanan pelanggan.
  - i. Informasi pelanggan yang lebih besar dapat diterjemahkan ke dalam persaingan harga dan keuntungan yang lebih rendah.

## 2.9 Pemasaran Digital

Sebenarnya, tidak ada perbedaan antara pemasaran "tradisional" dan pemasaran digital. Hanya saja adalah satu dan sama, terpisah dari digital yang khusus untuk sebuah media. Akhirnya, tujuan dari semua jenis pemasaran adalah untuk mempertahankan dan mengembangkan basis pelanggan dan merangsang penjualan di masa depan. Alat komunikasi

digital berkontribusi terhadap koneksi dan membangun hubungan jangka panjang dengan pelanggan.

Pemasaran digital sangat kuat dalam dua cara mendasar. Pertama, penonton dapat disegmentasikan dengan sangat tepat, bahkan sampai ke faktor-faktor seperti lokasi saat ini dan interaksi merek terbaru, yang berarti bahwa pesan dapat (dan harus) dipersonalisasi dan disesuaikan khusus untuk mereka. Kedua, lingkup digital hampir sepenuhnya terukur. Setiap menit dan setiap klik oleh konsumen dapat dipertanggungjawabkan. Dalam digital, dapat melihat dengan tepat bagaimana berbagai kampanye ditampilkan, saluran mana yang paling bermanfaat dan di mana upaya yang paling difokuskan. Secara kumulatif, akses ke data yang mengukur seluruh pengalaman pelanggan harus mengarah pada pengambilan keputusan berdasarkan data. Pemasaran digital membantu menciptakan permintaan konsumen dengan menggunakan kekuatan situs web yang saling berhubungan dan interaktif. Ini memungkinkan pertukaran mata uang tetapi lebih dari itu, memungkinkan pertukaran perhatian untuk nilai.

Digital memungkinkan peluang yang lebih besar untuk interaksi dan keterlibatan konsumen daripada yang mungkin di masa lalu, jadi penting untuk mempertimbangkan cara-cara di mana merek dapat menciptakan pengalaman interaktif bagi konsumen, bukan hanya mesra luas. Fakta bahwa pemasaran digital sangat empiris adalah salah satu kekuatan utamanya. Hampir semuanya dapat diukur: mulai dari perilaku, tindakan, dan jalur tindakan, elemen tertentu secara *online*. Ini berarti bahwa ahli strategi pemasaran digital harus terus mengukur dan beradaptasi untuk memastikan ROI tertinggi. Dibangun ke dalam strategi apa pun harus menjadi kerangka pengujian dan kemampuan untuk tetap fleksibel dan dinamis dalam media yang bergeser dan berubah sesuai perilaku pengguna [11].

### 2.9.1 CRM (Customer Relationship Management)

CRM (*Customer Relationship Management*) pengelolaan hubungan pelanggan, sudah ada sejak orang pertama mulai menjual barang. Penjaga toko pertama yang berhenti untuk mengobrol dengan pelanggannya, yang mengenal mereka dengan nama dan mungkin memberikan "freebie" kecil untuk terus menggunakan jasanya, sedang berlatih bentuk pemasaran hubungan pelanggan dengan membuat pelanggan merasa istimewa. Dia juga mungkin melihat dampak yang luar biasa pada garis bawahnya. Ini membantu pelanggan dilayani langsung oleh pemilik bisnis.

CRM adalah pendekatan yang berfokus pada pelanggan untuk bisnis yang didasarkan pada pengembangan hubungan jangka panjang yang bermakna. CRM bukan tentang keuntungan langsung. Ini tentang nilai seumur hidup pelanggan, pembelian yang akan

mereka lakukan di masa depan, kata-kata yang positif dari mulut ke mulut yang akan mereka hasilkan secara langsung dan setia mereka akan menunjukkan merek yang dimiliki. CRM yang efektif memungkinkan bisnis untuk berkolaborasi dengan pelanggan untuk menginformasikan keseluruhan awal bisnis, mendorong proses bisnis, mendukung pengembangan merek dan memaksimalkan ROI.

Ada dua pendekatan untuk CRM, yaitu :

1. Dapat menerapkannya sebagai pendekatan untuk strategi komunikasi menggunakan kontak yang dipersonalisasi dan disegmentasikan.
2. Sebagai strategi bisnis inti seperti program loyalitas yang digunakan dapat memilih tergantung pada ukuran dan tujuan bisnis [11].

Dari titik pandang arsitektur, kerangka CRM secara keseluruhan dapat diklasifikasikan ke dalam tiga komponen utama yaitu :

1. *Operational CRM*, merupakan pengelolaan secara otomatisasi dari proses bisnis secara terintegrasi dan horizontal termasuk *customer touch-points* dan integrasi *front-back office*. Salah satu penerapan CRM yang termasuk kategori operational CRM adalah dalam bentuk aplikasi situs web, melalui situs web suatu perusahaan dapat memberikan pelayanan kepada pelanggan.
2. *Analytical CRM*, merupakan analisis data yang diperoleh dari operasional CRM dengan memanfaatkan *tools* dan *software* untuk mendapatkan pemahaman yang lebih baik tentang perilaku pelanggan atau kelompok pelanggan.
3. *Collaborative CRM*, merupakan seperangkat aplikasi dari pelayanan kolaborasi termasuk *e-mail*, *e-communities*, publikasi personal, dan alat lainnya yang sejenis yang dirancang untuk memfasilitasi interaksi antara pelanggan dengan perusahaan.

## 2.9.2 SEO (Search Engine Optimization)

Dengan jutaan orang melakukan miliaran pencarian setiap hari untuk menemukan konten di-*internet*, google sendiri memproses lebih dari 40.000 pencarian per detik (Internet Live Stats, 2017). Masuk akal bahwa pemasar ingin produk mereka ditemukan *online*. Mesin pencari, saluran di mana pencarian ini terjadi, menggunakan algoritma yang dijaga ketat untuk menentukan hasil yang ditampilkan. Menentukan faktor apa yang diperhitungkan oleh algoritma ini telah menyebabkan praktik yang berkembang dikenal sebagai pengoptimalan mesin telusur (SEO). SEO adalah praktik mengoptimalkan situs web untuk mencapai ranking tertinggi pada halaman hasil mesin pencari (SERPs). Seseorang yang mempraktekkan SEO secara profesional dikenal sebagai SEO (*search engine optimization*).

Google menggunakan sekitar 200 faktor berbeda dalam algoritmanya untuk menentukan relevansi dan ranking (Dean, 2016). Tak satu pun dari mesin pencari utama mengungkapkan elemen yang mereka gunakan untuk memberi peringkat halaman, tetapi ada banyak praktisi SEO yang menghabiskan waktu menganalisis aplikasi paten untuk mencoba menentukan apa ini. SEO dapat dibagi menjadi dua yaitu, SEO topi putih dan SEO topi hitam, dengan beberapa pemakai topi abu-abu di antaranya. *Black hat* SEO mengacu pada mencoba permainan mesin *search*. SEO ini menggunakan cara yang meragukan untuk mencapai ranking tinggi dan situs web mereka kadang-kadang masuk daftar hitam oleh mesin pencari. *White hat* SEO, di sisi lain, mengacu pada bekerja dalam parameter yang ditetapkan oleh mesin pencari untuk mengoptimalkan situs web yang paling sesuai dengan kebutuhan mereka, jadi SEO topi putih harus memastikan bahwa pengguna dapat menemukan apa yang mereka cari [11].

Tabel 2. 1 Istilah dan Konsep utama SEO

Istilah	Defenisi
Semua teks	"alt" untuk tag HTML IMG. Ini digunakan dalam HTML untuk mengaitkan bidang teks ke gambar pada halaman web, biasanya dengan fungsi deskriptif, memberi tahu mesin telusur atau pengguna tentang apa gambar itu dan menampilkan teks dalam contoh di mana gambar tidak dapat memuat. Juga disebut semua tag. Teks jangkar adalah teks yang dapat diklik yang terlihat dalam tautan.
Optimasi <i>App Store</i> (ASO)	Proses mengoptimalkan aplikasi seluler dan web untuk toko web tertentu di mana mereka didistribusikan.
<i>Backlink</i>	Semua link dari halaman di domain eksternal menunjuk ke halaman di domain sendiri. Setiap tautan dari domain eksternal ke halaman tertentu dikenal sebagai <i>inbound / backlink</i> . Jumlah backlink mempengaruhi ranking, jadi semakin banyak backlink lebih baik, dapatkan tautan.
Canonical	Dalam SEO, canonical mengacu pada URL defenitif. Versi kanonikal adalah versi definitif.
Nama Domain	Nama yang mudah dibaca digunakan untuk mengidentifikasi alamat IP server yang membedakannya dari sistem lain di

	World wide Web : dengan nama domain redandyellow.co.za.
<i>Flash</i>	Teknologi yang digunakan untuk menampilkan video dan animasi di situs web. Dapat berupa <i>bandwith</i> yang berat dan tidak ramah untuk mencari laba-laba tambahan.
<i>Heading tag</i>	Heading tags (H1, H2, H3, dll) adalah elemen standar yang digunakan untuk menentukan judul dan subpos pada halaman web. Angka ini menunjukkan pentingnya sehingga tag H1 dilihat oleh spider sebagai lebih penting daripada tag H3. kata kunci target dalam tag H sangat penting untuk SEO yang efektif.
Halaman Depan	Halaman pertama dari situs web apa pun. Laman beranda memberi pengguna sekilas tentang situs apa yang mirip dengan indeks dalam buku, atau halaman konten di majalah.
<i>HyperText Markup Language (HTML)</i>	Tag HTML tertentu digunakan untuk menyusun informasi dan fitur dalam halaman web.
Hyperlink	Tautan dalam dokumen elektronik yang memungkinkan setelah mengkliknya, untuk mengikuti tautan ke halaman web yang relevan.
Internal Link	Suatu hyperlink di situs web yang menunjuk dari satu halaman ke halaman lain di domain situs web yang sama.
<i>Internet Protocol (IP)</i>	Alamat protokol internet adalah nomor unik yang digunakan untuk mewakili setiap komputer dalam satu jaringan.
Frekuensi kata kunci	Frekuensi kata kunci atau kata kunci di situs web.
Frasa kunci	Dua atau lebih kata yang digabungkan untuk membentuk permintaan pencarian sering disebut sebagai kata kunci. Akan lebih baik jika mengoptimalkan untuk frasa daripada kata tunggal.
Kata kunci rangking	Dimana kata kunci atau frase yang ditargetkan oleh peringkat SEO di hasil mesin pencari. Jika istilah yang ditargetkan tidak muncul pada tiga halaman pertama, mulailah khawatir.
Halaman arahan	URL yang disematkan di halaman web. Jika mengklik tautan, akan dibawa ke halaman itu.
Tautan	URL yang disematkan di halaman web. Jika mengklik tautan,

	akan dibawa ke halaman itu
Tautan balt	Teknik untuk membuat konten yang dirancang khusus untuk menarik dari halaman web lain.
<i>Meta tag</i>	Tag yang memberi tahu mesin search spider persis seperti apa halaman web itu. Sangat penting bahwa meta tag dioptimalkan untuk kata kunci yang ditargetkan. Meta tags adalah singkatan dari judul meta, deskripsi dan kata kunci.
Pengarah	Ketika seorang pengguna mengklik suatu tautan dari satu situs dari satu situs ke situs lain, situs yang ditinggalkan oleh pengguna adalah pengarah. Kebanyakan browser mencatat URL pengarah dalam string pengarah. Informasi ini sangat penting dalam menentukan kueri mana yang digunakan untuk menemukan situs tertentu.
Reboots.txt	Sebuah file yang ditulis dan disimpan dalam direktori root dari situs web yang membatasi spider mesin telusur dari pengindeksan laman tertentu dari situs web.
Mesin pencari spider	Program yang menjelajahi web, mengikuti tautan dan membangun indeks mesin telusur.
<i>Universal Resource Locator (URL)</i>	Alamat web yang unik untuk setiap halaman di internet.
Kegunaan	Ukuran seberapa mudah bagi pengguna untuk menyelesaikan tugas yang diinginkan. Situs dengan biaya kegunaan yang sangat baik jauh lebih baik daripada yang sulit digunakan.
XML sitemap	Panduan yang digunakan mesin telusur untuk membantu mereka mengindeks situs web, yang menunjukkan berapa banyak halaman yang ada, seberapa sering mereka diperbarui dan seberapa penting mereka.

Mesin pencari harus membantu pengguna menemukan apa yang mereka cari. Untuk memastikannya, mereka mencantumkan hasil terbaik terlebih dahulu, mencari sinyal yang memiliki popularitas, wewenang, relevansi, kepercayaan dan kepentingan. SEO, juga disebut organik dari optimasi alami, melibatkan mengoptimalkan situs web untuk mencapai ranking tinggi di *search engine* untuk kata kunci tertentu yang dipilih. Umumnya, teknik yang

digunakan untuk mengoptimalkan pada satu mesin telusur juga akan membantu upaya di yang lain.

SEO dapat dibagi menjadi dua strategi utama :

1. Optimasi *on-page*, yang dicapai dengan mengubah kode HTML, konten dan struktur situs web, membuatnya lebih mudah diakses oleh mesin pencari dan dengan perluasan, lebih mudah ditemukan oleh pengguna [11].

Dalam riwayat optimasi mesin pencari, peringkat sepotong konten dalam hasil mesin pencari biasanya turun ke dua pendorong utama yaitu, relevansi dan otoritas yang dibangun melalui konten di situs web. Saat mengoptimalkan konten, fokuskan salinan pada kata kunci spesifik yang cocok dengan apa yang orang cari secara *online*. Aturan pertama SEO *on-page* adalah untuk memikirkan apa yang mungkin dicari oleh target pengguna dan pastikan kata kunci tersebut ada di halaman. Ini meningkatkan kemungkinan menjangkau pengguna tersebut ketika mengakses ke Google, Bing atau mesin pencari lainnya. *On-page* SEO pada dasarnya adalah tentang dua hal :

- a. Memilih kata kunci terbaik yang menjadi dasar setiap halaman Anda
- b. Memperjelas mesin pencari bahwa halaman Anda berputar di sekitar kata kunci tersebut

*On-Page Optimization : the old way*, algoritma mesin pencari memeringkat halaman web berdasarkan banyak faktor. Premis dasarnya adalah bahwa halaman akan muncul di hasil mesin pencari karena situs web memiliki halaman yang menyebutkan istilah yang dicari. Tentu saja, banyak situs web menggunakan kata kunci yang sama di halaman yang sama. Ratusan alat berserakan di seluruh web yang memungkinkan mengukur kepadatan kata kunci suatu halaman. Kata kunci ini akan diisi di berbagai tempat, termasuk :

- a. URL
- b. Judul halaman
- c. Tag deskripsi meta
- d. Judul pada halaman

*On-Page Optimization: the new way*, pemasar masih harus menggunakan kata kunci. Tetapi hanya membuangnya di setiap aspek halaman, pikirkan nilai yang diinginkan untuk setiap halaman, dan kata kunci mana yang cocok dengan nilai itu. Dan bagaimana mengulangi kata yang sama berulang kali, dimana harus menggunakan serangkaian kata kunci terkait yang beragam untuk membantu memberi peringkat untuk berbagai kata kunci yang panjang [12].



2. Optimasi *off-page*, umumnya difokuskan pada membangun link ke situs web dan mencakup kegiatan seperti media sosial dan digital PR (*public relations*). SEO adalah cara yang sangat efektif untuk menghasilkan bisnis baru ke situs yang merupakan sebuah proses berkesinambungan dan cara berpikir tentang bagaimana mesin pencari melihat situs web [11].

*Off-page* SEO telah lama ditentukan oleh kuantitas, kualitas, dan relevansi tautan ke situs web yang membentuk otoritas SEO dan pada akhirnya memengaruhi peringkat hasil pencarian. Otoritas secara historis didirikan sebagai situs web lain yang terhubung dengan membangun portofolio tautan. Masalahnya, bagaimanapun, adalah bahwa otoritas hampir seluruhnya ditentukan oleh tautan yang adil. Pada dasarnya, SEO *off-page* dulu merupakan kata yang disukai untuk "mendapatkan lebih banyak tautan," tetapi sekarang harus fokus pada mendapatkan tautan melalui banyak saluran.

Otoritas situs hanya sebagian didasarkan pada analisis situs yang menautkan ke. SEO di luar halaman lebih dari sekadar pembuatan tautan (pembuatan tautan sendiri akan dibahas lebih banyak di bab selanjutnya). Cara mendiversifikasi otoritas untuk situs web adalah melalui percakapan yang dilakukan orang tentang merek dan referensi yang dibuat.

a. *Link Building*

Pembuatan tautan pada dasarnya adalah praktik membangun tautan masuk untuk membantu memberikan otoritas situs web. Upaya modern dari tautan "menghasilkan" dibahas lebih banyak di bab selanjutnya.

b. Konten yang menarik

Cara terbaik untuk mendapatkan tautan yang relevan dari situs lain ke situs adalah membuat konten unik dan relevan yang dapat dengan cepat mendapatkan popularitas online, terutama dengan target. Semakin bermanfaat, relevan, dan menarik konten, semakin besar peluang orang lain menemukan bahwa konten itu berharga dan terhubung dengannya.

c. *Co-Marketing*

*Co-marketing* mendefinisikan upaya banyak promosi silang, jaringan dan pembangunan komunitas. Ini adalah kemitraan antara dua atau lebih perusahaan di mana kedua perusahaan bersama-sama memasarkan konten, produk, atau layanan satu sama lain.

d. Media sosial

Media sosial tidak selalu di radar sebagai bagian dari SEO di luar halaman. Ketika pertama kali muncul ke permukaan itu sering digunakan secara tidak tepat. Pengguna atau bisnis pada dasarnya dianggap spam jaringan sosial dan pengikut bukannya menempatkan pemikiran nyata dan bekerja di belakangnya. Beberapa tanggapan, media sosial hanyalah satu

cara lagi untuk mendapatkan tautan. Media sosial dalam beberapa hal adalah forum baru atau komentar blog dengan kesempatan untuk melakukan percakapan nyata.

e. PR (*public relations*)

PR terencana dan dioptimalkan dengan baik dapat menjadi luar biasa untuk upaya SEO. PR harus mempromosikan konten yang didorong oleh dialog yang tulus dan etis untuk meningkatkan hasil mesin pencari organik suatu merek. Penjangkauan *public relations* yang efektif harus mencakup lebih dari saluran publikasi tradisional. Strategi harus mencakup penjangkauan ke *blogger*, *influencer* industri, dan pemimpin media sosial [12].

