BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

2.1.1 Sejarah Perusahaan

Toko Busana Muslim Swarna merupakan perusahaan yang bergerak di bidang penjualan produk busana muslim. Toko ini didirikan pada pertengahan tahun 2009 dengan fokus pada penjualan Jilbab. Toko ini terletak di Lt.1 Bandung Trade Center (BTC), Jl.Dr Djundjunan 143-149 Bandung. Sebagai perusahaan yang bergerak maju, toko busana muslim swarna ini mulai mengembangkan bidang penjualan produknya. Toko ini tidak hanya menjual Jilbab, toko ini mulai menjual gamis. Produk yang ditawarkan bermacam tipe dan warna dengan harga yang bervariasi.

2.1.2 Visi dan Misi

2.1.2.1 Visi

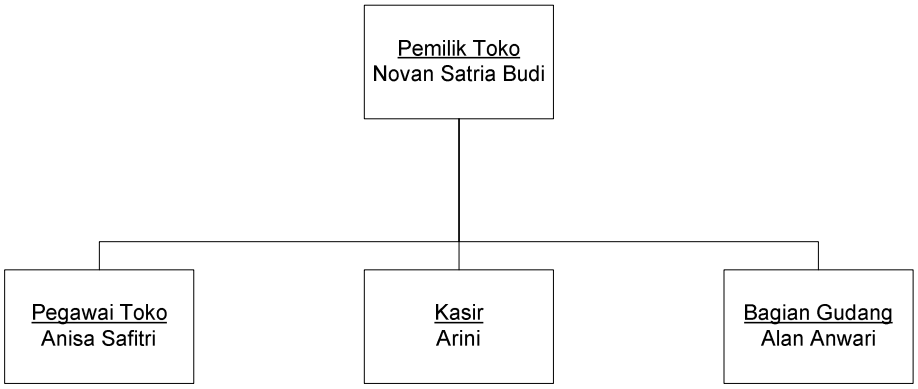
Menjadi toko yang dapat bersaing di pasar produksi dalam bidang penjualan berbagai macam busana muslim dengan menawarkan produk-produk berkualitas dengan harga kompetitif serta dapat memberikan suatu kepuasaan dan kepercayaan terhadap para *customer*.

2.1.2.2 Misi

1. Menyediakan produk yang berkualitas dengan harga yang kompetitif.
2. Mengembangkan inovasi terhadap kualitas produk yang dipasarkan.

2.1.3 Struktur Organisasi

Struktur organisasi pada toko busana muslim swarna dapat dilihat pada gambar 2.1.



Gambar 2.1 Struktur Organisasi Toko Busana Muslim Swarna

2.1.4 Deskripsi Kerja

Berikut ini merupakan deskripsi kerja pada toko busana muslim swarna adalah sebagai berikut:

1. Pemilik toko
    a. Bertanggung jawab atas kelangsungan toko.
    b. Mengontrol pegawai.
    c. Menerima laporan penjualan dan pembelian produk.
2. Pegawai toko
    a. Melayani *customer* yang datang ke toko.
    b. Menjaga kerapihan toko.
3. Kasir
    a. Bertanggung jawab atas keuangan toko.
    b. Mencatat setiap transaksi pembelian oleh *customer*.
    c. Membuat faktur penjualan produk.
4. Bagian gudang
    a. Bertanggung jawab atas pengadaan stok produk.

2.2 Landasan Teori

2.2.1 Kosep Data

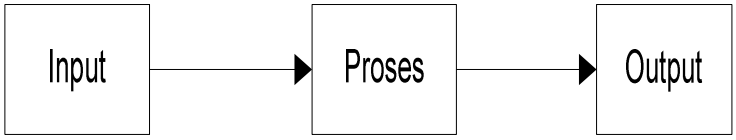
2.2.1.1 Data

Data merupakan fakta-fakta atau kejadian yang berupa angka-angka atau kode-kode tertentu [6]. Data dapat beruapa angka-angka, huruf atau symbol-simbol khusus atau gabungan darinya. Data mentah masih belum bisa bercerita banyak, sehingga perlu diolah lebih lanjut. Pengolahan data (Data Processing) adalah manipulasi dari data ke dalam bentuk yang lebih berguna dan lebih berarti berupa suatu informasi.

2.2.1.2 Informasi

Informasi (*Information*) adalah hasil dari kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian [6]. Jadi pengolahan data elktronik (PDE) atau *electronic data processing* (EDP) adalah manipulasi dan data kedalam bentuk yang lebih berarti berupa suatu informasi dengan menggunakan suatu alat elektronik yaitu komputer.

Suatu proses pengolahan data terdiri dari tiga tahapan dasar yang disebut dengan siklus pengolahan data, yaitu input, proses, dan output.



Gambar 2.2 Proses Pengolahan Data

Input : Tahap ini merupakan proses memasukkan data kedalam proses komputer melalui alat input (*input device*).

Proses : Tahap ini merupakan proses pengolahan dari data yang sudah dilakukan oleh alat pemproses (*processing device*) yang dapat berupa proses menghitung, membangkitkan, mengendalikan, atau mencari di *storage* (media penyimpanan).

OUTPUT : Tahap ini merupakan proses menghasilkan ouput dari hasil pengolahan data ke alat output (*output device*) yaitu berupa informasi.

2.2.1.3 Sistem

Sistem berasal dari bahasa yunani "systema" yang mempunyai arti :

1. Suatu keseluruhan yang tersusun dari sekian banyak bagian.
2. Hubungan yang berlangsung diantara satuan-satuan atau komponen secara teratur.

Terdapat dua kelompok dalam pengertian sistem yaitu pengertian yang menekankan pada prosedur data yang menekankan pada komponen atau elemennya. Dengan pendekatan prosedur sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu [6]. Yang dimaksud prosedur disini adalah urutan-urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa yang harus dikerjakan, siapa yang mengerjakan, kapan dikerjakan dan bagaimana mengerjakannya. Dengan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu [6].

Dengan adanya sistem ini, kita berpandangan bahwa sistem merupakan suatu komponen yang saling berhubungan dan dengan keterkaitannya teresebur dapat menimbulkan suatu yang disebut informasi sebagai output untuk mencapai tujuan.

Karakteristik Sistem yang terdiri dari beberapa karakter yaitu :

1. Komponen-komponen, suatu sistem berisi komponen yang saling berinteraksi atau saling bekerja sama
2. Batas sistem, yaitu pemisah antara sistem denga lingkungan luarnya.
3. Lingkungan luar, yaitu merupakan sistem yang di luar batas sistem ini tetapi lingkungan ini ada yang member pengaruh positif dan negative.
4. Penghubung yaitu, keluaran dari suatu subsistem dapat sebagai penghubung ke subsistem lainnya.

5. Masukan, yaitu masukan yang diberikan pada sistem ada 2 macam yaitu masukan perawatan (sebagai program dalam compiler agar sistem beroperasi) dan masukan sinyal (sebagai data dalam komputer agar menghasilkan).
6. Proses, yatu pengolahan input sehingga menghasilkan output yang diinginkan.
7. Keluaran, yaitu hasil dari operasi sistem ada yang memenuhi sasaran dan ada yang tidak.

Klasifikasi sistem diantaranya :

1. *Transaction Processing Systems* (TPS)

   Sistem informasi kompensasi yang dikembangkan untuk memproses data-data dalam jumlah besar untuk transaksi bisnis rutin seperti daftar gaji dan inventarisasi.

2. *Office Automation Systems* (OAS)

   Mendukung pekerja data yang biasanya tidak menciptakan pengetahuan baru untuk melakukan hanya menganalisis informasi, sedemikian rupa untuk menginformasikan data atau manipulasikannya dengan cara tertentu sebelum membaginya atau menyebarkannya secara keseluruhan dengan organisasi dan kadang-kadang diluar itu.

3. *Knowledge Work Systems* (KWS)

   Mendukung para pekerja profesional seperti ilmuwan, insiyur, dan dokter dengan membantu mereka menciptakan pengetahuan baru dan memungkinkan mereka mengkontribusikannya ke organisasi atau masyrakat.

2.2.2 *E-commerce*

2.2.2.1 Definisi *Electronic Commerce*

E-commerce merupakan kumpulan teknologi, aplikasi dan business proes yang menghubungkan perusahaan, consumer, dan komunitas melalui transaksi elektronik dan pertukaran barang, servis dan informasi secara elektronik [9].

2.2.2.2 Keuntungan *E-commerce*

1) Keuntungan *e-commerce* bagi perusahaan/ toko:

   Dengan penerapan *e-commerce* sebuah perusahaan dapat memiliki sebuah pasar internasional. Bisnis dapat dijalankan tanpa harus terbentur pada batasan daerah/tempat dengan adanya teknologi digital. Selain itu, biaya operasional dapat dihemat. Biaya untuk membuat, memproses, mendistribusikan, menyimpan, dan memperbaiki kembali informasi juga dapat ditekan.

2) Keuntungan *e-commerce* bagi konsumen:

   Sedangkan keuntungan bagi si *customer* atau konsumen dengan penerapan *e-commerce* ini adalah mereka para konsumen dapat berbelanja atau mengolah barbagai transaksi lain dalam 24 jam sepanjang hari, sepanjang tahun di sebagian besar lokasi. Konsumen tidak hanya memiliki sekumpulan produk yang bisa dipilih, namun juga daftar supplier internasional sehingga konsumen memiliki pilihan produk yang lebih banyak. Selain itu konsumen bisa membandingkan harga produk yang akan dibeli dengan mengunjungi berbagai situs yang berbeda atau dengan mengunjungi sebuah website tunggal yang menampilkan berbagai harga dari sejumlah provider.

   Secara ringkas keuntungan *e-commerce* tersebut adalah sebagai berikut:

   a. Bagi konsumen

      bisa membandingkan harga, berbelanja cukup di satu tempat.

   b. Bagi Pengelola bisnis :

      efisiensi, mengurangi human error, menghemat biaya distribusi.

   c. Bagi Manajemen

      Peningkatan pendapatan, loyalitas *customer*.

2.2.2.3 Jenis – jenis *E-commerce*

E-commerce digunakan untuk mendukung kegiatan pembelian dan penjualan, pemasaran produk, jasa, dan informasi melalui internet atau extranet. *E-*

commerce umumnya dikelompokan menjadi dua buah kategori: business-to-business (B2B) dan business-to-consumer (B2C). Pada perkembangan terakhir juga muncul jenis hubungan yang disebut consumer-to-consumer (C2C) dan cosumer-to-business(C2B) [8].

1) *Business To Customer (B2C)*

   Kelompok ini disebut juga transaksi pasar. Pada transaksi pasar, konsumen mempelajari produk yang ditawarkan melalui publikasi elektronik, membelinya dengan *electronic cash* dan *system secure payment*, kemudian minta agar barang dikirimkan. Secara ringkas jenis *e-commerce* ini merupakan *e-commerce* yang melibatkan konsumen dengan *merchant*-nya secara langsung.

2) *Business To Business* (*B2B*)

   Kelompok ini disebut sebagai transaksi antar perusahaan. Perusahaan, pemerintah dan organisasi lainnya bergantung pada komunikasi antar komputer sebagai sarana bisnis yang cepat, ekonomis dan dapat diandalkan. Perusahaan kecil saat ini sudah mulai tertarik dengan keuntungan yang diperoleh menggunakan *B2B* ini. Transaksi pada *B2B* menggunakan *EDI* dan *E-MAIL* untuk pembelian barang dan jasa, informasi dan konsultasi. Selain itu juga digunakan untuk pengiriman dan permintaan bisnis.

3) *Consumer-to-Consumer (C2C)*

   C2C menyatakan model perdagangan yang terjadi antara konsumen dengan konsumen melalui internet.

4) *Consumer-to-Business (C2B)*

   C2B mendukung bisnis yang berbasiskan consumer ke pebisnis. Dalam hal ini, internet dapat digunakan sebagai sarana untuk negosiasi.

2.2.3 *Internet*

Pengertian internet (inter_network) dapat diartikan jaringan komputer luas yang menghubungkan pemakai komputer satu dengan komputer lainnya dan dapat berhubungan dengan komputer dari satu Negara ke Negara di seluruh dunia, dimana

didalamnya terdapat berbagai aneka ragam informasi fasilitas layanan internet browsing atau surfing yaitu kegiatan “berselancar” di internet[2]. Kegiatan ini dapat dianalogikan layaknya berjalan-jalan di mal sambil melihat-lihat ke toko-toko tanpa membeli apapun.

2.2.3.1 Layanan Aplikasi Di Internet

Terdapat banyak sekali layanan aplikasi di internet dan masih terus akan berkembang sejalan dengan perkembangan teknologi informasi, dalam laporan ini hanya akan dibahas beberapa contoh aplilkasi yang banyak digunakan saja, diantaranya :

1. Electronic mail (E-mail)

   Email adalah pengiriman surat elektronik baik berupa teks maupun gabungan dengan gambar yang dikirimkan dari satu alamat email ke alamat email lainnya dijaringan internet. Seperti layaknya surat biasa pada umumnya, email berfungsi untuk mengeirimkan pesan kepada orang lain. Perbedaan email dengan surat biasa adalah email sudah tidak lagi membutuhkan kertas sebagai media untuk menuliskan pesan. Media yang digunakan adalah data digital melaluli jaringan internet.

2. News-USENET

   Digunakan sebagai sarana untuk berdiskusi antar pemakai jaringan internet. Aplikasi ini hampir serupa dengan suatu papan pengumuman, dimana setiap orang dapat mengirim, melihat dan menanggapi suatu berita atau suatu topic diskusi dengan fasilitas yang hampir sama dengan e-mail. Topic diskusi dipisahkan oleh grup, dan pemakai yang berminat dapat melihat isi diskusi pada newsgroup tersebut.

3. File Transfer Protocol

   File Transfer Protocol (FTP) adalah suatu protocol yang berfungsi untuk tukar menukar file dalam suatu network yang menggunakan TCP koneksi bukan UDP. Dua hal yang penting dalam FTP server dan FTP Client. FTP

server adalah suatu server yang menjalankan software yang berfungsi untuk memberikan layanan tukar menukar file dimana server tersebut selalui siap memberikan layanan FTP apabila mendapat permintaan (request) dari FTP client.

FTP client adalah koputer yang merequest koneksi ke FTP server untuk tujuan tukar menukar file. Setelah terhubung dengan FTP server, maka client dapat mendownload, meng-upload, me-rename, men-delete dan lain-lain sesuai dengan permission yang diberikan oleh FTP server.

Tujuan dari FTP server adalah sebagai berikut:

1. Untuk tujuan sharing data
2. Untuk menyediakan indirect atau imlicit remote computer
3. Untuk menyediakan tempat peyimpanan bagi member
4. Untuk menyediakan transfer data yang reliadble dan efisien

4. Remote Login-Telnet

Telnet adalah suatu aplikasi remote login internet yang memungkinkan anda untuk login atau menggunakan komputer yang berbeda pada jaringan secara interaktif. Untuk login dibutuhkan login account pada komputer tujuan, jika anda bukan member maka tidak bisa login ke komputer tersebut. Aplikasi ini biasanya digunakan untuk mengakses komputer berbasis sistem operasi UNIX dari tempat yang berbeda dari servernya.

5. Word Wide WEB

Awalnya informasi dapat dicari pada internet dengan menggunakan fasilitas information service berbasis archie, gopher dan WAIS (Wide Are Information System). Pencarian informasi berdasarkan menu-menu pada sistem tersebut dan output yang dihasilkan berbasis teks. Saat ini dengan teknologi Word Wide Web, dimunggkinkan dengan adanya Hypertext Transfer Protocol (HTTP) yang digunakan untuk mengakses suatu informasi

yang disimpan pada suatu situs web (website). Untuk dapat menggunakan sarana ini, dibutuhkan aplikasi web browser.

6. Universal Resource Locater (URL)

Pada penggunaan Word Wide Web, penunjukan suatu sumber informasi menggunakan metode universal resource locater (URL), yang merupakan konsep penamaan lokasi standar suatu file, direktori, komputer, lokasi komputernya dan metode yang digunakan. URL tidak hanya dapat menunjuk ke suatu file tapi dapat juga menunjuk suatu query, dokumen dalam satu database, atau hasil dari perintah atau yang lainnya. Dengan URL ini didefinisikan lokasi dan metode pengaksesan file tersebut.

7. Hyper Text Transfer Protocol (HTTP)

Web merupakan terobosan baru bagi teknologi sistem informasi yang menghubungkan data dari banyak sumber layanan yang beragam macamnya di internet. Pengguna tinggal mengklik tombol mousenya pada link-link hypertext yang ada untuk melompat ke dokumen diberbagai lokasi di internet. Link-link sendiri bisa mengacu pada dokumen web, server FTP (File Transfer Protocol), e-mail ataupun layanan lainnya.

Server dan browser web berkomunikasi satu sama lain denga protocol yang memang dibuat khusus untuk ini, yaitu HTTP. HTTP bertugas menangani permintaan (request) dari browser untuk mengambil dokumen-dokumen web.

2.2.4 *HTML*

2.2.4.1 Pengantar *Html*

HTML atau *HyperText Markup Language* merupakan file teks atau file ASCII yang berisi instruksi/ script kepada web browser untuk menampilkan suatu tampilan grafis dari sebuah halaman web [1].

Pengenalan Kode *HTML*

```
<HTML>
</HTML>
```

Masing – masing baris diatas disebut *tag*. *Tag* adalah kode yang digunakan untuk me – *mark – up* (memoles) teks *ASCII* menjadi *file HTML*. Setiap *tag* diapit dengan tanda kurung runcing. Ada *tag* pembuka yaitu <HTML> dan ada *tag* penutup yaitu </HTML> yang ditandai dengan tanda *slash* (garing miring) di depan awal tulisannya. *Tag* di atas memberikan kaidah bahwa yang akan ditulis di antara kedua *tag* tersebut adalah isi dari dokumen *HTML*.

Dalam penamaan sebuah dokumen yang akan ditampilkan pada *web browser* maka nama yang digunakan harus diakhiri dengan ektensi *(.html*) atau *(.htm*). Ekstensi dokumen *HTML* awalnya 3 karakter, adalah untuk mengakomodasi sistem penamaan dalam *DOS*.

Pengaturan warna di halaman *HTML* menggunakan mode kombinasi *RGB* (*Red*, *Green*, *Blue*) dimana setiap warna ditampilkan dalam dua digit nilai heksadesimal (0, 1, 2, …F). Setiap bagian dua digit kode menunjukkan banyaknya intensitas dari kombinasi warna merah, hijau dan biru. Dilihat pada tabel 2.1.

Tabel 2.1 Kombinasi warna

Heksadesimal	Warna
#000000	*Black*
#FFFFFF	*White*
#FF0000	*Red*
#00FF00	*Green*
#0000FF	*Blue*
#FF00FF	*Magenta*

2.2.4.2 Definisi elemen

Sebuah dokumen *HTML* disusun oleh beberapa elemen atau lebih dikenal dengan sebuah komponen – komponen dasar. Elemen dapat berupa teks murni, atau bukan teks, atau keduanya. Elemen atau komponen tersebut misal *head*, *body*, paragraf, *list*, dan lain – lain.

Untuk menandai sebuah elemen dalam suatu dokumen *HTML* digunakan *tag*. *Tag HTML* terdiri dari sebuah [(<) (nama *tag* (>)] contoh <h1>, *tag* pada umumnya berpasangan (misalnya <h1> dengan </h1>, tanda / pada *tag* pasangan memberikan tanda bahwa *tag* tersebut merupakan pambatas akhir elemen yang dibuka oleh *tag* awal. Jadi secara umum penulisan sebuah *tag* adalah <nama *tag*>…</nama *tag*> selain itu dalam penamaan *tag* tidak menganut *case sensitive.*

Elemen dasar yang harus dimiliki dalam pembuatan dokumen *HTML* adalah *tag html, tag body* dan *tag head*. Elemen *head* berisi informasi tentang dokumen, sedangkan elemen *body* berisi teks atau elemen lainnya. Lebih jelasnya adalah sebagai berikut:

```
<html>
<head>
...informasi dokumen...
</head>
<body>
...informasi yang di tampilkan pada halaman browser...
</body>
</html>
```

2.2.5 *PHP*

2.2.5.1 Pengantar *PHP*

PHP (*PHP: Hypertext Preprocessor*) adalah bahasa *server – side scripting* yang menyatu dengan *HTML* untuk membuat halaman *web* dinamis [5]. Karena *PHP* merupakan *server – side scripting* maka sintaks dan perintah – perintah *PHP* akan dieksekusi di *server* kemudian hasilnya dikirimkan ke *browser* dalam format *HTML.* Dengan demikian kode program yang ditulis dalam *PHP* tidak akan terlihat

oleh *user* sehingga keamanan halaman *web* lebih terjamin. *PHP* dirancang untuk membentuk halaman *web* yang dinamis, yaitu halaman *web* akan dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*.

PHP termasuk dalam *Open Source Product*, sehingga *source code PHP* dapat diubah dan didistribusikan secara bebas. *PHP* juga dapat berjalan pada berbagai *web server* seperti *IIS* (*Internet Information Server*), *Apache*, *Xitami*. *PHP* juga mampu lintas *platform.* Artinya *PHP* dapat berjalan di banyak sistem operasi yang beredar saat ini, di antaranya: Sistem Operasi *Microsoft Windows* (semua versi), *Linux*, *Mac OS*, *Solaris*. *PHP* dapat dibangun sebagai modul pada *web server Apache* dan sebagai *binary* yang dapat berjalan sebagai *CGI* (*Common Gateway Interface*). *PHP* dapat mengirim *HTTP header*, dapat mengatur *cookies*, mengatur *authentication* dan *redirect users*.

Salah satu keunggulan yang dimiliki oleh *PHP* adalah kemampuannya untuk melakukan koneksi berbagai macam *software* sistem manajemen basis data/*Database Management system* (*DBMS*), sehingga dapat menciptakan suatu halaman *web* yang dinamis. *PHP* mempunyai koneksitas yang baik dengan beberapa *DBMS* antara lain *Oracle*, *Sybase*, *mSQL*, *MySQL*, *Microsoft SQL Server*, *Solid*, *PostgreSQL*, *Adabas*, *FilePro*, *Velocis*, *dBase*, *Unix dbm*, dan tak terkecuali semua *database* ber – *interface ODBC*.

2.2.5.2 Mengenal perintah dasar *PHP*

Script PHP termasuk dalam *HTML – embedded,* artinya kode *PHP* dapat disisipkan pada sebuah halaman *HTML.* Parser *PHP* bekerja membaca *file HTML* sampai ditemukan penanda khusus yang memberitahukan untuk menerjemahkan skrip berikutnya sebagai kode *PHP*. Parser *PHP* akan mengesksekusi semua perintah dalam blok kode *PHP* tersebut. Dengan cara inilah maka kode skrip *PHP* dapat ditempelkan pada dokumen *HTML.* Teks lainnya yang berada di luar blok *PHP* akan dianggap sebagai skrip *HTML* biasa.

Berikut ini contoh sederhana pemakaian bahasa *PHP* yang disisipkan dalam halaman *HTML*:

```
<html>
      <head>
            <title> First PHP Script </title>
      </head>
      <body>
       <?php
        echo “Ini Script PHP pertama”;
       ?>
      </body>
</html>
```

2.2.6 *Cascading Style Sheet* (CSS)

CSS (*Cascading Style Sheet*) adalah mekanisme memisahkan antara isi dan presentasi dokumen *HTML*. *CSS* menambahkan *style* pada halaman, meningkatkan dan memperbaiki penyajian isi halaman. *CSS* Bertindak sebagai lapisan yang mempengaruhi presentasi dari elemen *HTML* yang diberikan. Warna, *font*, *text, size, background*, susunan elemen pada halaman dan seluruh aspek presentasi isi, diatur oleh *CSS*.

2.2.7 *Web Server*

Web server adalah *software* yang menjadi tulang belakang dari *world wide web (www). Web server* menunggu permintaan dari *client* yang menggunakan *browser* seperti *Netscape Navigator*, *Internet Explorer, Modzilla*, dan program *browser* lainnya. Jika ada permintaan dari *browser*, maka *web server* akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke *browser*. Data ini mempunyai format yang standar, disebut dengan format *SGML* (*standar general markup language*). Data yang berupa format ini kemudian akan ditampilkan oleh *browser* sesuai dengan kemampuan *browser* tersebut. Contohnya, bila data yang dikirim berupa gambar, *browser* yang hanya

mampu menampilkan teks (misalnya *lynx*) tidak akan mampu menampilkan gambar tersebut dan jika ada akan menampilkan alternatifnya saja.

Web server, untuk berkomunikasi dengan *client*-nya (*web browser*) mempunyai protokol sendiri, yaitu *HTTP* (*hypertext tarnsfer protocol*). Dengan protokol ini, komunikasi antar *web server* dengan *client*-nya dapat saling dimengerti dan lebih mudah.

2.2.8 *Search Engine Optimization (SEO)*

Istilah *search engine optimization* (*SEO*) pertama kali digunakan pada 26 Juli tahun 1997 oleh sebuah pesan *spam* yang diposting di *Usenet*. Pada masa itu algoritma mesin pencari belum terlalu kompleks sehingga mudah dimanipulasi. Versi awal algoritma pencarian didasarkan sepenuhnya pada informasi yang disediakan oleh *webmaster* melalui *meta tag* pada kode *html* situs *web*. *Meta tag* menyediakan informasi tentang konten yang terkandung pada suatu halaman *web* dengan serangkaian katakunci (*keyword*).

Ada beberapa hal yang akan mendorong *website* berada di *rangking* atas pada *search engine*. Salah satunya dengan memilih nama *domain* yang menggunakan kata kunci yang banyak disukai oleh pengguna. Trik-trik yang lain adalah sebagai berikut:

1. *On Page*: *Title*

   Title adalah judul *website.* Usahakan judul *website* juga sesuai dengan *keyword* yang paling banyak dicari. Judul yang baik adalah judul yang memenuhi beberapa syarat sebagai berikut:

   1) Panjangnya 65 karakter saja. Singkat, padat, jelas, hanya berisi *keyword* yang diinginkan.
   2) Apabila terdiri dari beberapa kata kunci, berikan tanda " | " untuk memisahkan.

3) Sebaiknya jangan gunakan kata hubung seperti “dan, yang, untuk, dan lain-lain”. Kata hubung ini akan diskip oleh *google* sehingga hanya akan menghabiskan tempat.
4) Jangan mengulang kata yang sama lebih dari 3 kali. Penggunaan yang berlebihan dapat menyebabkan *google* melakukan *penalti* karena dianggap melakukan kecurangan atau *SPAM.*

2. *On Page*: *Meta description & meta keyword*

   Meta description adalah uraian singkat mengenai *website* dan akan ditampilkan di hasil pencarian *search engine*. *Meta keyword* bisa diganti dalam *file index.php* atau *index.html*. Kodenya terletak di bagian atas. Beberapa jenis *CMS* memungkinkan anda menggantinya langsung dari halaman manajemen *web* tanpa harus secara manual mengubahnya di *index.html* atau *index.php.*

3. *On Page*: *H1 dan H2*

   H1 dan H2 adalah kode yang merupakan singkatan dari *heading* 1 dan *heading 2*. *Text* apa pun yang diapit kode ini akan berukuran besar. H1 lebih besar dari H2. Contoh dari *text* yang diapit oleh kode tersebut adalah sebagai berikut:

   ini adalah contoh kalimat yang diapit kode H1

   ini adalah contoh kalimat yang diapit kode H2

   Search engine akan menganggap kode yang diapit oleh H1 dan H2 ini sebagai kata kunci yang bobotnya lebih penting dari pada kata kunci yang didapat dalam *content*. Dengan demikian, apabila tidak memberikan judul dalam format H1 dan H2, maka kehilangan kesempatan untuk memberitahu *google* tentang kalimat penting dari sebuah *web*.

4. *On Page*: *Keyword* di dalam halaman *website*

   1) Pakai huruf besar, huruf miring, tanda kutip pada teks *keyword* di halaman *website*.

2) Ketik *keyword* di awal paragraf atau di akhir paragraf terakhir.
3) Pakai *anchor text* pada *link*. Dari pada membuat *link www.swarna.co.id*, lebih baik membuat *link* toko busana muslim swarna, yang ketika diklik mengarah ke *www.swarna.co.id.*

5. *Off page*: *One way link* dan *two way link*

   One way link adalah menaruh sebanyak-banyaknya *link website* di *website* milik orang lain. Hal ini bisa dilakukan dengan memberikan komentar-komentar di *blog* orang lain dengan mencantumkan halaman *web*. Di samping itu, juga bisa melakukan ini dengan bergabung di milis – milis yang bisa diakses oleh umum (bisa masuk ke *mail-archive.com*) dan mencantumkan alamat *web* sebagai *signature email*.

   Two way link adalah menaruh sebanyak-banyaknya *link website* di *website* milik orang lain dengan imbalan orang lain juga taruh linknya di *web*.

2.2.9 *Secure Socket Layer* (*SSL*)

2.2.9.1 Perkembangan *SSL*

Pada awal perkembangan *web,* masalah keamanan telah menjadi perhatian khusus. *National Center for Supercomputing Application* (*NCSA*) merilis *browser web Mosaic* 1.0 yang cukup terkenal pada November 1993. Delapan bulan kemudian, *Netscape Communications* melengkapi rancangan *SSL* versi 1.0 dan diikuti versi – versi selanjutnya dari *SSL*. *SSL* dibangun sedemikian sehingga dapat digunakan untuk hampir semua *browser* dan *server*. *SSL* bertugas memberikan kerahasiaan, otentikasi dan integritas pesan secara aman ke pengguna *web*. Adanya *SSL* dapat diamati dari *URL "https"* yang merupakan *SSL-secured URL* atau dapat diamati dari ikon gembok kecil pada bagian kanan bawah tampilan *windows browser*. Meskipun demikian, untuk mendukung *browsing web* secara aman, *server web* tidak cukup hanya dengan menggunakan *SSL*. *Browser web* juga harus memperoleh sertifikat kunci publik dari organisasi yang dipercaya. Untuk pengguna pada internet publik, organisasi itu biasanya berkuasa atas sertifikat publik.

Beberapa pemilik sertifikat yang terkemuka adalah *AT&T Certificate Services*, *GTE CyberTrust*, *KeyWitness International*, *Microsoft*, *Thawte Consulting* dan *VeriSign*.

2.2.9.2 Pesan *SSL*

Client dan *server* berkomunikasi dengan bertukar pesan. Secara teknis, *SSL* mendefinisikan beberapa level pesan yang dapat dilihat pada Tabel 2.2.

Tabel 2.2 Beberapa Level Pesan pada *SSL*

Message	*Description*
Alert	Memberitahu pihak lain dalam komunikasi tentang kemungkinan adanya penerobosan pada keamanan atau kegagalan komunikasi
ApplicationData	Informasi aktual bahwa kedua pihak saling bertukar pesan yang dienkripsi, diotentikasi dan/atau diverifikasi oleh *SSL*
Certificate	Pesan yang membawa sertifikat kunci publik pengirim
CertificateRequest	*Request* dari *server* kepada *client* untuk memberikan sertifikat kunci publik
CertificateVerify	Pesan dari *client* berisi verifikasi bahwa *client* mengetahui kunci privat yang bersesuaian dengan sertifikat kunci publik *client*
ChangeCipherSpec	Tanda untuk mulai munggunakan layanan keamanan yang telah disetujui sebelumnya, seperti enkripsi
ClientHello	Pesan dari *client* yang memberi informasi layanan keamanan yang diinginkan dan dapat didukung oleh *client*
ClientKeyExchange	Pesan dari *client* yang membawa kunci kriptografi untuk komunikasi
Finished	Indikasi bahwa semua negosiasi awal sudah selesai dan komunikasi yang aman telah dibangun
HelloRequest	Pesan dari *server* kepada *client* untuk memulai atau *restart* proses negosiasi *SSL*
ServerHello	Pesan dari *server* yang menunjukkan layanan keamanan yang akan digunakan untuk komunikasi
ServerHelloDone	Indikasi dari *server* bahwa *server* telah melengkapi semua *request* ke *client* yang

	diperlukan untuk membangun komunikasi
ServerKeyExchage	Pesan dari *server* yang membawa kunci kriptografi untuk komunikasi

2.2.10 *MySQL*

MySQL dikembang oleh sebuah perusahaan Swedia bernama *MySQL AB* yang pada saat itu bernama *TcX DataKonsult AB* sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1979. Awalnya *TcX* membuat *MySQL* dengan tujuan mengembangkan aplikasi *web* untuk klien. *TcX* merupakan perusahaan pengembang *software* dan konsultan *database*. Saat ini *MySQL* sudah diakuisisi oleh *Oracle Corp*.

MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya [7]. Kepopuleran *MySQL* antara lain karena *MySQL* menggunkan *SQL* sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan, kinerja *query* cepat dan mencukupi untuk kebutuhan *database* perusahaan – perusahaan skala menengah kecil. *MySQL* juga bersifat *open source* dan *free* pada berbagai *platform* (kecuali pada *Windows*, yang bersifat *shareware*). *MySQL* didistribusikan dengan lisensi *open source GPL* (*General Public License*) mulai versi 3.23, pada bulan juni 2000.

MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (*PHP dan Perl*). *MySQL* dan *PHP* dianggap sebagai pasangan *software* pengembangan aplikasi *web* yang ideal. *MySQL* lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script PHP*.

2.2.11 *Adobe Dreamweaver*

Sebuah program aplikasi yang menyediakan segala kebutuhan untuk membuat sebuah *web* yang utuh dan mudah dengan adanya *tool – tool* yang sudah terintegrasi dalam program Dreamweaver.

2.2.12 *XAMPP*

XAMPP merupakan aplikasi yang mengintegrasikan beberapa aplikasi utama *web* di dalamnya. Dalam *XAMPP* terdapat instalasi modul *PHP*, *MYSQL*, *Web Server Apache*.

2.2.13 Metode Pembayaran

2.2.13.1 Pembayaran *offline*

1. Pembayaran langsung di toko

   Metode pembayaran langsung di toko merupakan pembayaran secara umum yang telah dilakukan oleh banyak orang, biasanya *customer* datang ke toko untuk melihat produk, apabila ada yang cocok sesuai keinginannya maka *customer* memesan produk tersebut ke kasir atau penjaga toko. Lalu, pembayaran dilakukan secara langsung di toko tersebut kepada kasir dan *customer* dapat membawa langsung produk yang telah dibelinya dengan membawa nota penjualan sebagai tanda bukti pembayaran.

2. Transfer Bank

   Transfer adalah pemindahan dana antar rekening disuatu tempat ke tempat yang lain, baik untuk kepentingan nasabah (debitur/ non debitur) dan atau untuk kepentingan bank itu sendiri.

2.2.13.2 Pembayaran *online*

1. *Paypal*

   a. Sejarah *paypal*

Perusahaan *Paypal* (*Paypal corp*.) seperti yang dikenal sekarang adalah penggabungan antara perusahaan *Confinity* dan *X.com* pada tahun 2000. *Confinity* didirikan pada desember 1998 oleh Peter Thiel dan Max Levchin, awalnya sebagai alat pembayaran *Palm Pilot* dan sebagai perusahaan kriptograpi (ilmu pembacaan sandi, tulisan – tulisan atau angka – angka rahasia). Sedangkan *X.com* didirikan oleh Elon Musk pada maret 1999, yaitu perusahaan penyedia jasa perencanaan keuangan. Kedua perusahaan lokasi pertamanya adalah kantor di 165 University Avenue di Palo Alto, California, rumah dari beberapa pemula di Silicon Valley. *Ebay* melihat perkembangan penggunaan *Paypal* dan menyimpulkan *Paypal* cocok sebagai salah satu alat pembayaran di *ebay*. Pada mei 1999 *ebay* membeli *Billpoint* dan menjadikan *Billpoint* sebagai alat pembayaran utama *ebay* saat itu dan membuat *Billpoint* khusus hanya untuk alat pembayaran di *ebay* saja, karenanya *Paypal* hanya tampil beberapa kali sebagai pilihan alat pembayaran di *ebay* dibandingkan *Billpoint*. Tetapi karena masyarakat lebih menyukai *Paypal* karena mudah dan lebih aman, pada Februari 2000 rata- rata ada 200.000 penawaran barang perhari yang menggunakan *Paypal* sedangkan *Billpoint* hanya 4.000 penawaran. Pada april 2000 lebih dari 1 juta penawaran menggunakan *Paypal*.

b. Pengertian *paypal*

PayPal adalah salah satu alat pembayaran (*Payment procesors*) menggunakan *internet* yang terbanyak digunakan di dunia dan teraman saat ini. Pengguna *internet* dapat membeli barang di situs *e-commerce*, lisensi *software* original, keanggotaan situs, urusan bisnis, mengirim dan menerima donasi atau sumbangan, mengirim uang ke pengguna *PayPal* lain di seluruh dunia dan banyak fungsi lainnya dengan mudah dan otomatis menggunakan *internet*.

Terdapat tiga buah jenis *account* yang disediakan oleh *PayPal* untuk para konsumennya, diantaranya adalah sebagai berikut:

1) *Personal*

Dengan *account* tipe ini, pengguna sudah dapat mengirim dan menerima uang dan melakukan penjualan dengan *ebay*. Pada tipe ini, pengguna bisa menerima pembayaran dari *account Paypal* lain, namun tidak bisa menerima pembayaran dari *credit* atau *debit card*. Tidak ada biaya untuk setiap transaksi yang dilakukan pada tipe ini. Terdapat limit berapa banyak uang yang dapat diterima per bulannya. Tipe personal ini tidak cocok untuk pengguna *PayPal* yang merencanakan untuk berjualan produk dalam jumlah yang besar.

2) *Premier*

Tipe ini hampir sama dengan *account personal*, bedanya pada tipe ini pengguna bisa menerima pembayaran dari *credit card*, *debit card* dan rekening bank. Pengguna juga bisa menggunakan fasilitas *shopping cart* dan fasilitas laporan pembayaran (*payment reporting tool*). *Account premier* cocok digunakan untuk penjual yang ingin menjual produknya secara regular. Untuk pengguna yang memiliki *account personal*, pengguna tersebut dapat melakukan *upgrade* ke *account premier*.

3) *Business*

Tipe ini cocok digunakan untuk bisnis yang berskala besar atau *online store*. Pada tipe ini pengguna diperbolehkan menggunakan laporan dan *tool ebay* tanpa adanya persyaratan mengenai jenis transaksi. Pengguna akan dikenakan biaya dalam menggunakan *account* tipe ini. Jika saat ini pengguna memiliki *account personal* atau *premier*, pengguna dapat melakukan *upgrade* ke *account business*. *PayPal* tidak mengenakan biaya pada pembeli dalam mengirimkan uang ke penjual. *PayPal* akan mengenakan biaya kepada pengguna sebagai penjual (penerima uang) sebesar 1.9% hingga 2.9% dari jumlah uang yang diterima.

c. Keamanan transaksi pada *paypal*

PayPal lebih aman bila dibandingkan dengan alat pembayaran *online* yang lain. Hal ini karena pembeli yang merasa dirugikan atau tidak sesuai deskripsi yang diberitahukan, bisa meminta uang kembali kepada penjual dalam waktu maksimal 45 hari setelah terjadi pembayaran. Jika pembeli menggunakan kartu kredit akan mendapatkan pengembalian uang *chargeback* dari perusahaan kartu kreditnya. *PayPal* juga melindungi penjual dari pengembalian uang atau komplain yang tidak benar dari pembeli sesuai pembuktian.

Perlindungan untuk penjual dirancang untuk melindungi penjual dari klaim pembeli yang mengaku telah mengirim uang yang tidak mempunyai catatan bukti pembayaran dan deskripsi transaksi, setiap pembelian dan pembayaran menggunakan *PayPal* selalu ada catatan bukti pembayarannya di *account PayPal* pengirim dan penerima uang, sebagai bukti jika benar telah terjadi pengiriman uang antara keduanya. Bisa diambil kesimpulan bahwa menggunakan *PayPal* lebih aman dari alat pembayaran *online* lain bagi pengirim dan penerima uang karena ada bukti pembayaran dan catatan deskripsi, serta dapat melakukan komplain yang benar jika terjadi sesuatu atau terhindar dari komplain yang salah.

PayPal sebagai alat pembayaran yang teraman didunia telah mengumumkan kebijakan perlindungan pembeli yang baru dan telah disempurnakan bagi para konsumen di seluruh kawasan Asia Pasifik yang ingin berbelanja *online*.

Kebijakan perlindungan pembeli *PayPal* saat ini berlaku untuk semua pengguna yang terdaftar di Asia Pasifik, termasuk Australia, China, Hongkong, Indonesia, Jepang, Malaysia, Filipina, Singapura dan Thailand. Dengan perlindungan pembeli *Paypal* ini, kini telah menjamin konsumen bahwa informasi keuangan dan transaksi *e-commerce* konsumen terlindungi dengan lebih baik saat membeli barang dari Internet. Dengan menggunakan

Paypal maka pembelian pada transaksi *online* yang baru atau belum dikenal tidak perlu khawatir akan resiko kecurangan, keamanan dan barang tidak dikirim tidak akan terjadi.

Paypal sangat fokus pada keamanan karena salah satu hasil riset *Paypal* dengan ACNielsen menunjukkan bahwa keamanan adalah kekhawatiran utama bagi para pembelanja *online* di Australia, sedangkan privasi informasi dan perlindungan konsumen merupakan dua kekhawatiran utama bagi para pembelanja di Hongkong.

2.2.14 *IP-Dedicated*

IP-Dedicated (*Internet Protocol*) adalah *IP address* yang digunakan secara eksklusif pada satu *account hosting* [2]. *IP Dedicated* biasa digunakan untuk jalur transaksi yang menggunakan *SSL-enncrypted* untuk pengamanan transaksi di *internet*, biasanya digunakan oleh *website* yang memiliki *e-commerce* atau penjualan *online*.

Adapun fungsi dan keuntungan *ip dedicated*:

1. Sebuah *website* dapat diakses langsung dari *IP address* tanpa harus mengetikkan nama *domain*.
2. Menggunakan *IP Dedicated* dapat memaksimalkan *SERP* (*Search Engine Result Page*).
3. Penggunaan *IP Dedicated* dapat dipasangkan dengan *SSL- enncrypted* yang berfungsi menjaga keamanan transaksi melalui *internet* saat menggunakan Kartu Kredit.
4. *Website* lebih cepat diakses.

2.2.15 Metode Untuk *Security*

1. Pengertian Fungsi *Hash*

Hash adalah suatu teknik "klasik" dalam Ilmu Komputer yang banyak digunakan dalam praktek secara mendalam. *Hash* merupakan suatu metode yang

secara langsung mengakses *record-record* dalam suatu tabel dengan melakukan transformasi aritmatik pada *key* yang menjadi alamat dalam tabel tersebut. *Key* merupakan suatu input dari pemakai di mana pada umumnya berupa nilai atau string karakter.

Pelacakan dengan menggunakan *Hash* terdiri dari dua langkah utama, yaitu:

a. Menghitung Fungsi *Hash*. Fungsi *Hash* adalah suatu fungsi yang mengubah *key* menjadi alamat dalam tabel. Fungsi *Hash* memetakan sebuah *key* ke suatu alamat dalam tabel. Idealnya, *key-key* yang berbeda seharusnya dipetakan ke alamat-alamat yang berbeda juga. Pada kenyataannya, tidak ada fungsi *Hash* yang sempurna. Kemungkinan besar yang terjadi adalah dua atau lebih *key* yang berbeda dipetakan ke alamat yang sama dalam tabel. Peristiwa ini disebut dengan *collision* (tabrakan). Karena itulah diperlukan langkah berikutnya, yaitu *collision resolution* (pemecahan tabrakan).
b. *Collision Resolution. Collision resolution* merupakan proses untuk menangani kejadian dua atau lebih *key* di-*hash* ke alamat yang sama. Cara yang dilakukan jika terjadi *collision* adalah mencari lokasi yang kosong dalam tabel *Hash* secara terurut. Cara lainnya adalah dengan menggunakan fungsi *Hash* yang lain untuk mencari lokasi kosong tersebut.

2. Sejarah MD5

MD5 adalah salah satu dari serangkaian algortima(merupakan salah satu fungsi *Hash*) *message digest* yang didesain oleh Profesor Ronald Rivest dari MIT (Rivest, 1994). Saat kerja analitik menunjukkan bahwa pendahulu MD5, yaitu MD4 mulai tidak aman, MD5 kemudian didesain pada tahun 1991 sebagai pengganti dari MD4 (kelemahan MD4 ditemukan oleh Hans Dobbertin).

Pada tahun 1993, den Boer dan Bosselaers memberikan awal, bahkan terbatas, hasil dari penemuan *pseudo-collision* dari fungsi kompresi MD5. Dua vektor inisialisasi berbeda *I* dan *J* dengan beda 4-bit diantara keduanya.

MD5compress(I,X) = MD5compress(J,X)

Pada tahun 1996 Dobbertin mengumumkan sebuah kerusakan pada fungsi kompresi MD5. Dikarenakan hal ini bukanlah serangan terhadap fungsi *hash* MD5 sepenuhnya, hal ini menyebabkan para pengguna kriptografi menganjurkan pengganti seperti WHIRLPOOL, SHA-1 atau RIPEMD-160.

Ukuran dari *hash* — 128-bit — cukup kecil untuk terjadinya serangan *brute force.*. MD5CRK adalah proyek distribusi mulai Maret 2004 dengan tujuan untuk menunjukkan kelemahan dari MD5 dengan menemukan kerusakan kompresi menggunakan *brute force attack*. Bagaimanapun juga, MD5CRK berhenti pada tanggal 17 Agustus 2004, saat [[kerusakan *hash*]] pada MD5 diumumkan oleh Xiaoyun Wang, Dengguo Feng, Xuejia Lai dan Hongbo Yu. Serangan analitik mereka dikabarkan hanya memerlukan satu jam dengan menggunakan IBM P690 cluster.

Pada tanggal 1 Maret 2005, Arjen Lenstra, Xiaoyun Wang, and Benne de Weger mendemontrasikan kunstruksi dari dua buah sertifikat X.509 dengan *public key* yang berbeda dan *hash* MD5 yang sama, hasil dari demontrasi menunjukkan adanya kerusakan. Konstruksi tersebut melibatkan *private key* untuk kedua *public key* tersebut. Dan beberapa hari setelahnya, Vlastimil Klima menjabarkan dan mengembangkan algortima, mampu membuat kerusakan Md5 dalam beberapa jam dengan menggunakan sebuah komputer notebook. Hal ini menyebabkan MD5 tidak bebas dari kerusakan.

Dikarenakan MD5 hanya menggunakan satu langkah pada data, jika dua buah awalan dengan *hash* yang sama dapat dibangun, sebuah akhiran yang umum dapat ditambahkan pada keduanya untuk membuat kerusakan lebih masuk akal. Dan dikarenakan teknik penemuan kerusakan mengijinkan pendahuluan kondisi *hash* menjadi arbitari tertentu, sebuah kerusakan dapat ditemukan dengan awalan apapun. Proses tersebut memerlukan pembangkitan dua buah *file* perusak sebagai *file template*, dengan menggunakan blok 128-*byte* dari tatanan data pada 64-*byte*

batasan, *file-file* tersebut dapat mengubah dengan bebas dengan menggunakan algoritma penemuan kerusakan.

Hash MD5 lama, yang dibuat sebelum serangan-serangan tersebut diungkap, masih dinilai aman untuk saat ini. Khususnya pada *digital signature* lama masih dianggap layak pakai. Seorang *user* boleh saja tidak ingin membangkitkan atau mempercayai *signature* baru menggunakan MD5 jika masih ada kemungkinan kecil pada teks (kerusakan dilakukan dengan melibatkan pelompatan beberapa bit pada bagian 128-*byte* pada masukan *hash*) akan memberikan perubahan yang berarti. Penjaminan ini berdasar pada posisi saat ini dari kriptoanalisis. Situasi bisa saja berubah secara tiba-tiba, tetapi menemukan kerusakan dengan beberapa data yang belum-ada adalah permasalahan yang lebih susah lagi, dan akan selalu butuh waktu untuk terjadinya sebuah transisi.

2.2.16 Sistem Rekomendasi

Sistem rekomendasi merupakan model aplikasi dari hasil observasi terhadap keadaan dan keinginan *customer*. Sistem rekomendasi memanfaatkan opini seseorang terhadap suatu produk dalam *domain* atau kategori tertentu, untuk membantu seseorang dalam memilih produk. Untuk mempermudah *customer* mengambil keputusan yang tepat dalam menentukan produk yang akan dibelinya [8]. Metode yang akan digunakan adalah *Collaborative Filtering.*

Collaborative filtering merupakan proses penyaringan atau pengevaluasian *item* menggunakan opini orang lain. *Collaborative filtering* melakukan penyaringan data berdasarkan kemiripan karakteristik konsumen sehingga mampu memberikan informasi yang baru kepada konsumen karena sistem memberikan informasi berdasarkan pola satu kelompok konsumen yang hampir sama. Perbedaan minat pada beberapa anggota kelompok menjadikan sumber informasi baru yang mungkin bermanfaat bagi anggota kelompok lainnya.

Secara umum proses pemberian rekomendasi terdiri atas tiga langkah, yaitu:

1. Penemuan similar *user*.

2. Pembuatan ketetenggaan (*neighborhood*).
3. Perhitungan prediksi berdasarkan tetangga yang dipilih.

Collaborative filtering menghasilkan prediksi atau rekomendasi bagi pengguna atau *customer* yang dituju terhadap satu *item* atau lebih. *Rating* dalam *Collaborative filtering* dapat berbentuk, seperti:

1. Model *rating* skalar yang terdiri atas *rating* numerik seperti 1 sampai 5.
2. Model *rating* biner dengan memilih antara setuju atau tidak setuju, atau dapat pula baik atau buruk.
3. *Rating unary* dapat mengindikasikan bahwa pengguna telah mengobservasi atau membeli *item* atau merating *item* dengan positif.

Tidak tersedianya *rating* mengindikasikan tidak terdapat informasi yang menghubungkan pengguna dengan *item*. *Rating* dapat dikumpulkan secara eksplisit, implisit, ataupun gabungan antara eksplisit dan implisit. *Rating* eksplisit yaitu *rating* yang didapatkan pada saat *customer* atau pengguna diminta menyediakan opini terhadap *item* tertentu. *Rating* implisit yaitu *rating* yang didapatkan melalui aksi yang dilakukan *customer*.

Algoritma *Collaborative filtering* dibagi ke dalam dua kelas yang berbeda menurut teori dan kepraktisannya, yaitu algoritma non-probabilistik dan algoritma probabilistik. Suatu algoritma dianggap probabilistik bila algoritma probabilistik. Algoritma tersebut mewakili distribusi probabilitas saat menghitung prediksi *rating* atau daftar rangking rekomendasi. Algoritma non-probabalistik yang terkenal yaitu *nearest neighbours algorithm*. Algoritma ini dibagi menjadi dua kelas yaitu *user-based* dan *item-based*.

1. *User-Based Collaborative filtering*

User-based nearest neighbour algorithm menggunakan teknik statistika untuk menemukan sekumpulan pengguna, dikenal sebagai tetangga (*neighbour*), yang memiliki sejarah setuju dengan pengguna yang menjadi sasaran. Setelah sekumpulan tetangga terbentuk, sistem menggunakan algoritma yang berbeda untuk

menggabungkan kesukaan *neighbours* untuk menghasilkan prediksi atau rekomendasi N-teratas untuk *active user*.

2. *Item-Based Collaborative filtering*

Item-based Collaborative filtering merupakan metode rekomendasi yang didasari atas adanya kesamaan antara pemberian *rating* terhadap suatu produk dengan produk yang dibeli. Dari tingkat kesamaan produk, kemudian dibagi dengan parameter kebutuhan *customer* untuk memperoleh nilai kegunaan produk. Produk yang memiliki nilai kegunaan tertinggilah yang kemudian dijadikan rekomendasi [10].

Pemberian nilai rekomendasi terdiri dari beberapa langkah, yaitu:

1. Pengecekan *customer*. Jika diketahui *customer* C login ke sistem, maka sistem akan mengecek siapa saja *customer* dengan riwayat pembelian atau peratingan yang mirip dengan *customer* C.
2. Jika sudah dikertahui maka sistem akan menghitung *distance* (Dis) dari setiap produk yang sama antara *customer* C dengan *customer* lainnya menggunakan persamaan:

   $$Dis(produk) = (RPperson - Rpotherperson)^2 \quad \ldots. (2.1)$$

   Keterangan:

   Dis(produk) : Nilai *distance* produk yang sama.

   RPperson : Rating produk yang diberikan oleh *customer* C.

   RPotherperson : Rating produk yang diberikan oleh *customer* lain.
3. Setelah itu sistem akan menghitung similiarity (Sim) atau tingkat kesamaan antara *customer* C dengan *customer* lainnya dengan persamaan:

   $$Sim(customer) = 1/(1+\sum Dis) \quad \ldots..(2.2)$$

   Sim(*customer*) : Nilai similiarity *customer* C dengan *customer* yang lain.

   Dis(produk) : Nilai *distance* produk yang sama.

Hitung nilai tingkat kesamaan ini pada tiap *customer* yang memiliki kemiripan *history* pembelian dengan *customer* C.

4. Setelah didapat maka pembangkitan nilai rekomendasi produk yang belum pernah dibeli *customer* C akan dihitung dengan persamaan:

   Rek(produk) : Angka rekomendasi untuk produk yang belum dibeli *customer* C.

   Sim(*customer*) : Nilai similiarity *customer* C dengan *customer* yang lain.

   RP : Rating produk yang belum dibeli oleh *customer* C.

5. Setelah nilai rekomendasi dibangkitkan maka sistem akan merekomendasikan item dengan nilai rekomendasi tertinggi.

2.2.17 Basis Data

Basis data (*database*) adalah kumpulan dari berbagai data yang saling berhubungan satu dengan yang lainnya [7]. Basis data tersimpan di perangkat keras, serta dimanipulasi dengan menggunakan perangkat lunak. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau informasi yang akan disimpan. *Database* merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna atau *user*.

Penyusunan basis data meliputi proses memasukkan data kedalam media penyimpanan data dan diatur dengan menggunakan perangkat Sistem Manajemen Basis Data (*Database Management System DBMS*). Manipulasi basis data meliputi pembuatan pernyataan (*query*) untuk mendapatkan informasi tertentu, melakukan pembaharuan atau penggantian (*update*) data.

2.2.17.1 Elemen basis data

1. Entitas

Sekumpulan objek yang terdefinisikan yang mempunya karakteristik sama dan bisa dibedakan dengan yang lainnya. Objek dapat berupa barang, orang, tempat atau suatu kejadian.

2. Atribut

   Deskripsi data yang bisa mengidentifikasi entitas yang membedakan entitas tersebut dengan entitas lain. Seluruh harus cukup untuk menyatakan identitas obyek, atau dengan kata lain, kumpulan atribut dari setiap entitas dapat mengidentifikasi keunikan individu.

3. Data Value (Nilai Data)

   Data actual atau informasi yang disimpan pada tiap data, elemen, atau atribut.

4. File/Tabel

   Kumpulan dari beberapa record.

5. Record/Tuple

   Kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap. Satu record mewakili satu data atau informasi.

2.2.17.2 Kelebihan dan Kekurangan Basis Data

Kelebihan dari basis data:

1. Mengurangi duplikasi data.
2. Meningkatkan integritas data.
3. Memelihara independensi data.
4. Meningkatkan keamanan data.
5. Manipulasi data lebih canggih
6. Mudah untuk diakses

Kekurangan dari basis data:

1. Sistem lebih rumit, jadi memerlukan tenaga ahli dalam desain, program dan implementasi.
2. Bila ada akses yang tidak benar, kerusakan dapat terjadi.

3. Karena sebuah data di tempat terpusat, kerusakan software dan hardware dapat terjadi.
4. Proses pemeliharaan dapat memakan waktu karena ukurannya yang besar.

2.3 Perancangan Sistem

2.3.1 Flowmap

Flowmap adalah campuran peta dan flow chart, yang menunjukan pergerakan benda dari satu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. Flowmap menolong analisis atau programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

2.3.2 *Pengenalan UML (Undified Modeling Language)*

Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Seperti yang kita ketahui bahwa menyatukan banyak kepala untuk menceritakan sebuah ide dengan tujuan untuk memahami hal yang sama tidaklah mudah, oleh karena itu diperlukan sebuah bahasa pemodelan perangkat lunak yang dapat dimengerti oleh banyak orang.

Banyak orang yang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah *Data Flow Diagram* (DFD) untuk memodelkan lunak yang menggunakan pemrogramanan prosedural atau strukural, kemudian juga ada *State Transition Diagram* (STD) yang digunakan untuk memodelkan sistem *real time* (waktu nyata).

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

Seperti yang kita ketahui bahwa banyak hal di dunia sistem informasi yang tidak dapat dibakukan, semua tergantung kebutuhan, lingkungan dan konteksnya. Begitu juga dengan perkembangan penggunaan UML bergantung pada level abstraksi penggunaannya. Jadi, belum tentu pandangan yang berbeda dalam penggunaan UML adalah suatu yang salah, tapi perlu ditelaah dimanakah UML digunakan dan hal apa yang kita gunakan sehari-hari, belum tentu penyampaian bahasa dengan puisi adalah hal yang salah. Sistem informasi bukanlah ilmu pasti, maka jika ada banyak perbedaan dan interpretasi di dalam bidang sistem informasi merupakan hal yang sangat wajar.

2.3.2.1 Sejarah UML

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967 [7]. Bahasa pemograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada *developer* pengembang bahasa pemrograman berorientasi objek selanjutnya.

Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman *Smalltalk* pada awal 198o-an yang kemudian diikuti dengan perkembangan bahasa pemograman beorientasi objek yang lainnya

seperti C objek, C++, *Eiffel*, dan CLOS. Secara aktual, penggunaan bahasa pemrograman berorientasi objek pada saat itu masih terbatas, namun telah banyak menarik perhatian saat itu. Sekitar lima tahun setelah Smalltalk berkembang, maka berkembang pula metode pengembangan berorientasi objek. Metode yang pertama diperkenalkan oleh Sally Shlaer dan Stephen Mellor (Shlaer-Mellor, 1988) dan Peter Coad dan Edward Yourdon (Coad-Yourdon, 1991), dikuti oleh Grady Booch (Booch, 1991), James R. Rumbaugh, Michael R, William Lorensen, Frederick Eddy, william Premerlani (Rumbaugh-Premerlani-Eddy-Lorensen, 1991), dan masih banyak lagi.

Buku terkenal yang juga berkembang selanjutnya adalah karangan Ivar (Jacobson, 1992) yang menerangkan perbedaan pendekatan yang focus pada *use case* dan proses pengembangan. Sekitar lima tahun kemudian muncul buku yang membahas mengenai metodologi berorientasi objek yang diikuti dengan buku-buku yang lainnya. Di dalamnya juga mengenai konsep, definisi, notasi, terminologi, dan proses mengenai metodologi berorientasi objek.

Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Usaha penyatuan ini banyak mengambil dari metodolog-metodologi yang berkembang saat itu. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep seperti konsep *Object Modeling Technique* (OMT) dari Rumbaugh dan Booch (1991), konsep *The classes, Responsibilities, collaborators* (CRC) dari Rebecca Wirfs-Brock (1990), konsep pemikiran Ivar Jacobson, dan beberapa konsep Iainnya dimana James R. Rumbaigh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama *Rational Software Coorporation* menghasilkan bahasa yang disebut dengan *Unified Modeling Languague* (UML).

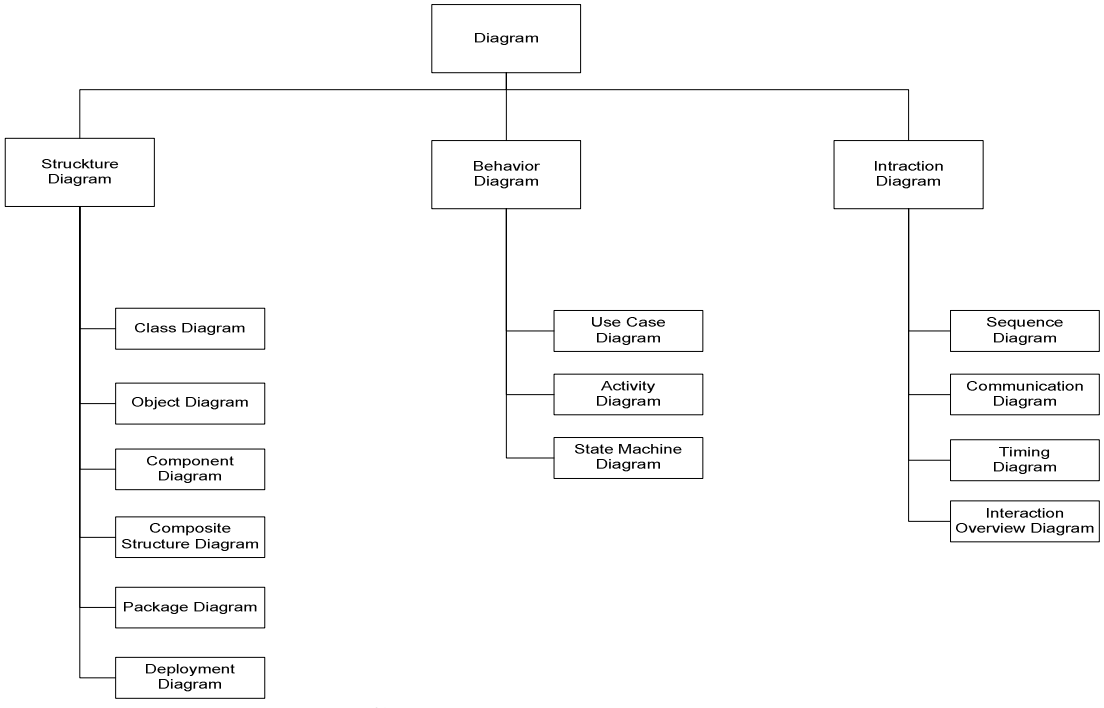
Pada 1996, *Object Management Group* (OMG) mengajukan proposal agar adanya standardisasi pemodelan berorientasi objek dan pada bulan september 1997 UML diakomodasi oleh OMG sehingga sampai saat ini UML telah memberikan

kontribusinya yang cukup besar di dalam metodologi berorientasi objek dan hal-hal yang terkait di dalamnya.

Secara fisik, UML adalah sekumpulan spesifikasi yang dikeluarkan OMG. UML terbaru adalah UML 2.3 yang terdiri dari 4 macam spesifikasi, yaitu *Diagram Interchange Specification*, UML *Infrastructure*, *Superstructure*, dan *Object Coratraint Language* (OCL). Seluruh spesifikasi tersebut dapat diakses di website http://www.omg.org.

2.3.2.2 Diagram UML

Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar di bawah.



Gambar 2.3 Diagram UML

Berikut ini penjelasan dari pembagian kategori tersebut.

1. *Struckture diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

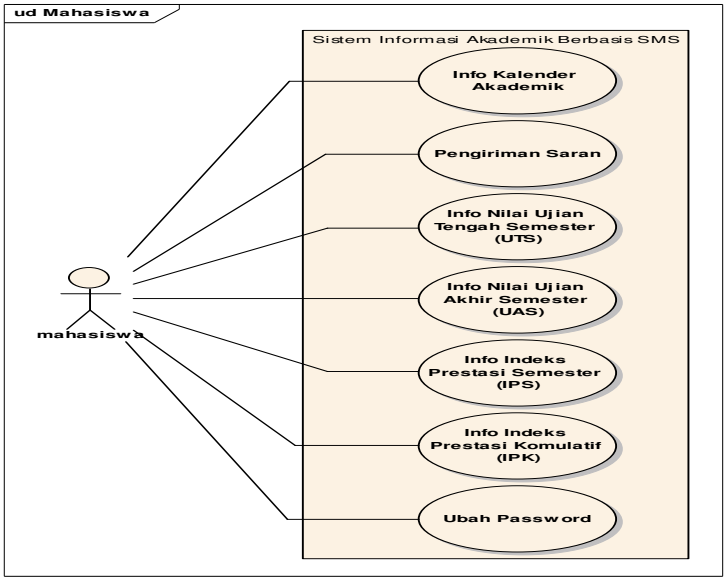
2. *Behavior diagrams* yaitu koumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antarsubsistem pada suatu sistem.

2.3.3 *Use case* Diagram

Use case atau diagram *use case* merupakan pemodelan utuk kelakuan (*behavior*) sistem informasi yang akan dibuat [7]. *Use case* diagram terdiri atas diagram untuk *use case* dan aktor. Aktor merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.

Use case merepresentasikan operasi-operasi yang dilakukan oleh aktor. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnya. Aktor yang melakukan operasi dihubungkan dengan garis lurus ke *use case*.

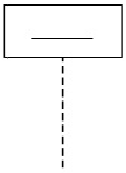
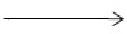
Contoh *use case* diagram :



Gambar 2.4 Contoh *Use case* Diagram

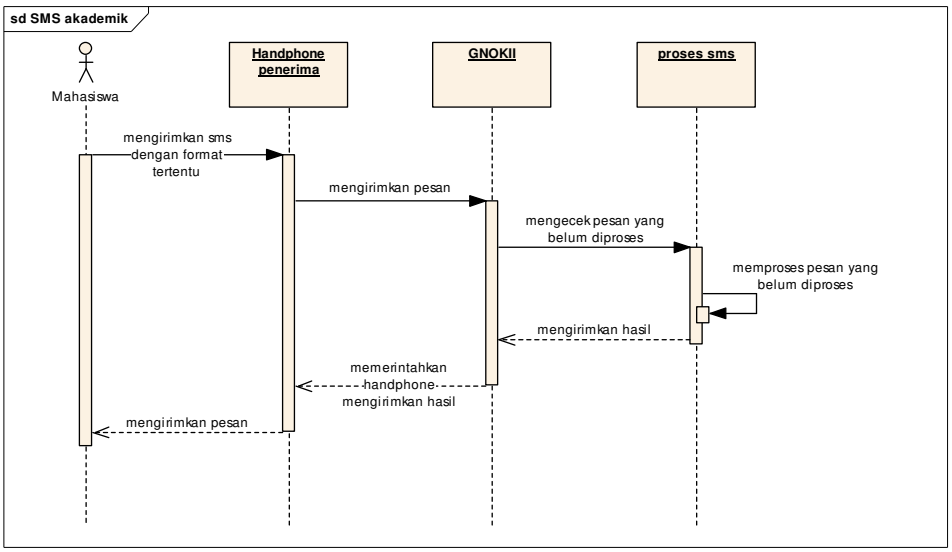
2.3.4 Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. Sequence diagram terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) [7]. Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan output tertentu.

Simbol	Keterangan
	Obiect Timeline
	Aktivasi
	Message

Gambar 2.5 Simbol dan Keterangan pada Sequence Diagram

Contoh Sequence Diagram :



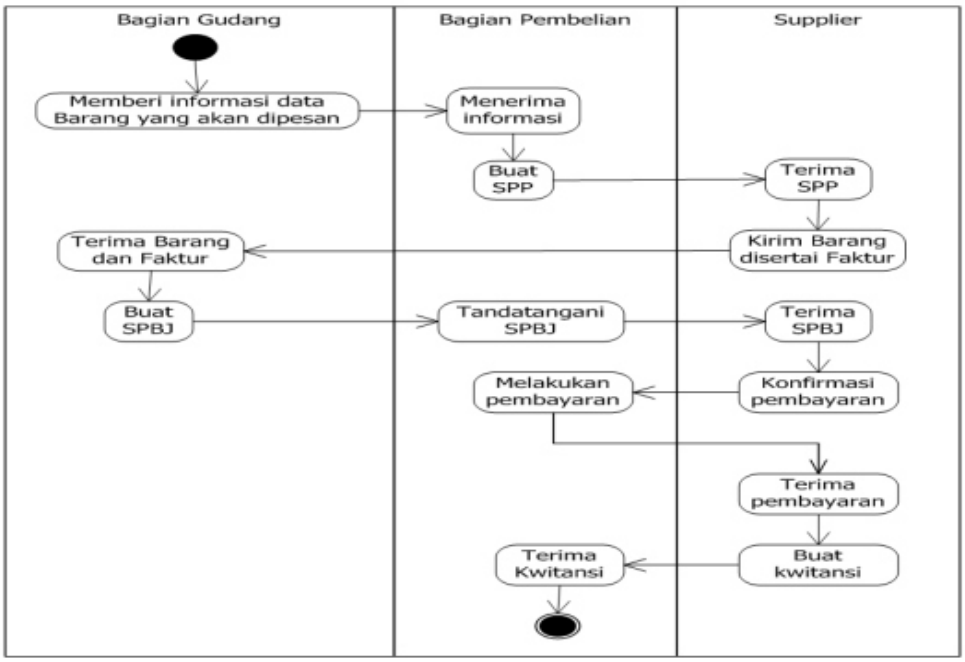
Gambar 2.6 Contoh Sequence Diagram

2.3.5 Activity Diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, kemungkinan yang mungkin terjadi, dan bagaimana mereka berakhir [7]. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Contoh Activity diagram:



Gambar 2.7 Contoh Activity Diagram

2.3.6 Class Diagram

Diagram Kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem [7]. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

   Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas yang menangani tampilan sistem

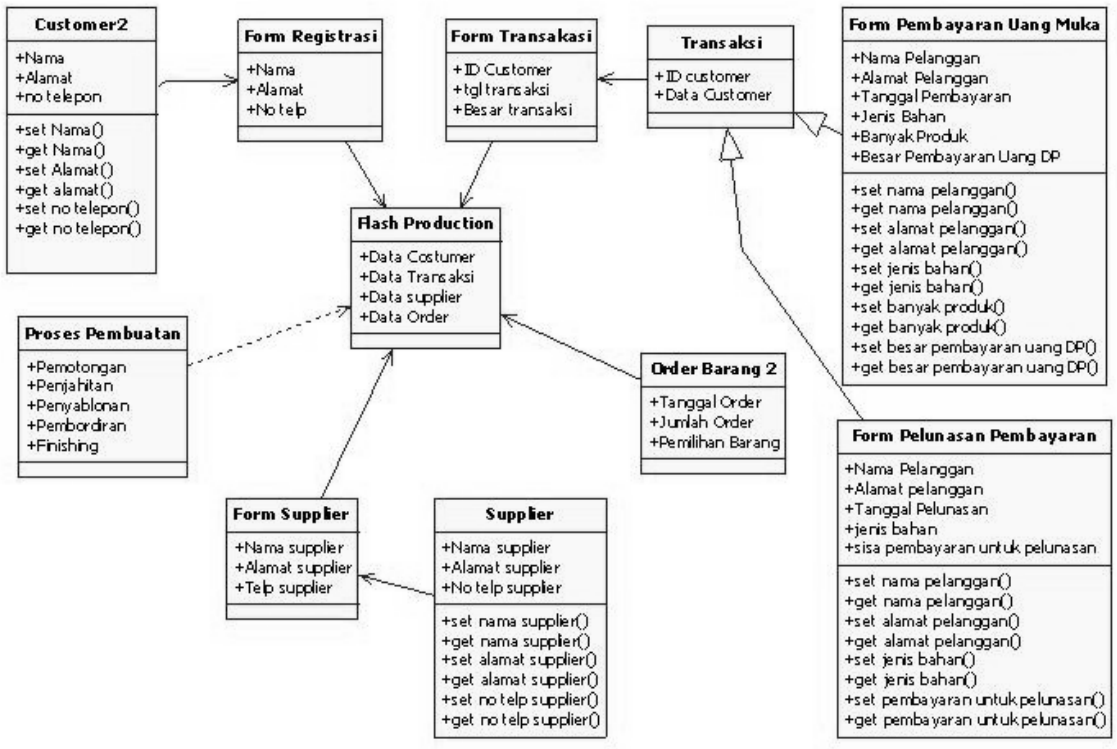
   Kelas yang mendefinisikan dan mengatur tampilan ke pemakai
3. Kelas yang diambil dari pendefinisian *use case*

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.

4. Kelas yang diambil dari pendefinisian data

   Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun disimpan ke basis data.

Jenis-jenis kelas di atas juga dapat digabungkan satu sama lain sesuai dengan pertimbangan yang dianggap baik asalkan fungsi-fungsi yang sebaiknya ada pada struktur kelas tetap ada. Susunan kelas juga dapat ditambahkan kelas utilitas seperti koneksi ke basis data, membaca file teks, dan lain sebagainya sesuai kebutuhan.
Berikut adalah contoh class diagram yang akan digambarkan pada gambar 2.8 berikut:



Gambar 2.8 Contoh Class Diagram