BAB II

LANDASAN TEORI

2.1 Tinjauan Perusahaan

Tinjauan perusahaan adalah untuk mengetahui keadaan diperusahaan diantaranya adalah sejarah berdirinya perusahaan, struktur organisasi perusahaan serta visi dan misi perusahan.

2.1.1 Sejarah Perusahaan

Toko blackshoes merupakan sebuah perusahaan yang bergerak di bidang penjualan sepatu. Toko blackshoes berdiri pada tahun 2009 atas dasar coba-coba saja, dan tak disangka usaha tersebut dapat menarik banyak konsumen serta berjalan dengan baik. Itulah salah satu alasan yang menyebabkan si pemilik mempertahankan usaha ini sampai sekarang dan menambah satu cabang baru yang beralamat di Jalan Dipatiukur No.66D / 49C Kota Bandung.

2.1.2 Visi dan Misi Perusahaan

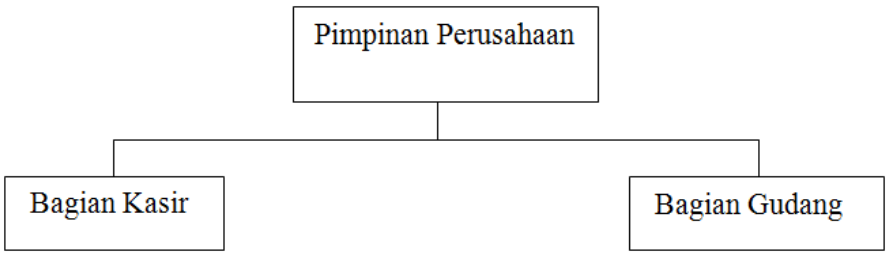
Visi dari toko blackshoes adalah untuk mengembangkan usaha atau bisnis dalam bidang sepatu dan yang dapat berimbas kepada pertumbuhan ekonomi dan dapat meningkatkan kesejahteraan orang banyak.

Adapun misi dari toko blackshoes adalah sebagai berikut :

1. Mengelola perusahaan yang sesuai dengan bisnis yang sehat dan didukung oleh teknologi tepat guna serta sumber daya manusia yang professional.

2. Mengembangkan usaha agar memiliki daya saing yang kuat.

2.1.3 Struktur Organisasi



Gambar II.1 Struktur Organisasi Perusahaan

Pimpinan Perusahaan : mengontrol kinerja semua karyawan

Bagian Kasir : mengelola keuangan perusahaan

Bagian Gudang : mengelola stok barang

2.2 Internet

2.2.1 Pengertian Internet

Menurut Robert J. Verzello yang diterjemahkan oleh Jogiyanto HM (1998:23)

"Internet adalah suatu jaringan komputer global yang terbentuk dari jaringan-jaringan komputer local dan regional, dengan adannya jaringan ini memungkinkan komunikasi data antar komputer-komputer yang terhubung ke jaringan tersebut".

Internet itu sendiri berasal dari kata *Interconnection Networking*, yang berarti hubungan dari banyak jaringan komputer dengan berbagai tipe dan jenis, dengan menggunakan tipe komunikasi seperti telepon, salelit, dan lainnya.

Dalam mengatur integrasi dan komunikasi jaringan komputer ini menggunakan protokol yaitu TCP/IP. TCP (*Transmission Control Protocol*) bertugas untuk memastikan bahwa semua hubungan bekerja dengan benar, sedangkan IP (*Internet Protocol*) yang mentransmisikan data dari satu komputer ke komputer lain. TPC/IP secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat di gunakan, mengatur dan mengirimkan paket-paket pengiriman data.

Untuk dapat ikut serta menggunakan fasilitas internet, Anda harus berlangganan ke salah satu ISP (*Internet Server Provider*) yang ada dan melayani daerah Anda. ISP ini biasanya disebut penyelenggara jasa internet. Anda bisa menggunakan fasilitas dari Telkom seperti Telkomnet Instan, speedy dan juga layanan ISP lain seperti first media, netzip dan sebagainya.

Internet memberikan banyak sekali manfaat, ada yang bisa memberikan manfaat baik dan buruk. Baik bila digunakan untuk pembelajaran informasi dan buruk bila digunakan untuk hal yang berbau pornografi, informasi kekerasan, dan lain-lainnya yang negatif.

Internet ini memungkinkan pengguna komputer di seluruh dunia untuk saling berkomunikasi dan berbagi informasi dengan cara saling mengirimkan *email*, menghubungkan komputer satu ke komputer yang lain, mengirim dan menerima file dalam bentuk teks, audio, video, membahas topik tertentu pada *newsgroup, website social networking* dan lain-lain.

2.3 M-Commerce

M-commerce didefinisikan sebagai *"semua kegiatan yang berhubungan dengan (potensial) transaksi komersial yang dilakukan melalui jaringan komunikasi dengan antarmuka perangkat nirkabel"* (Shuster, 2001, P.2). Tranasaksi *m-commerce*, melibatkan perpindahan kepemilikan atau hak untuk menggunakan barang dan jasa, yang dimulai dan / atau diselesaikan dengan menggunakan akses *mobile* ke-dimediasi jaringan komputer dengan bantuan perangkat elektronik. *M-commerce* merupakan bagian dari *E-commerce*, yang spesifik pada proses transaksi yang dilakukan secara elektronik melalui jaringan internet dan seluler. Selain dilakukan secara *remote* melalui jaringan seluler, transaksi *m-commerce* juga dapat dilakukan secara lokal *server* dengan menggunakan koneksi *Bluetooth, InfraRed* atau RFID.

2.3.1 Sejarah M-Commerce

M-commerce lahir pada 1997, ketika dua ponsel diaktifkan pada mesin penjual Coca Cola yang dipasang di Helsinki daerah di Finlandia . Mesin-mesin menerima pembayaran melalui SMS. Layanan perbankan pertama ponsel berbasis diluncurkan pada tahun 1997 oleh Merita Bank Finlandia, juga menggunakan SMS.

Pada 1998, penjualan pertama dari konten digital yang diunggah ke ponsel ini dimungkinkan ketika mengunduh nada dering komersial yang diluncurkan di Finlandia oleh Radiolinja.

Dua *platform* besar komersial nasional untuk *m-commerce* diluncurkan pada tahun 1999: Smart Money di Filipina , dan NTT DoCoMo’s i-Mode layanan

internet di Jepang. i-Mode menawarkan rencana pembagian pendapatan *revolusioner* dimana NTT DoCoMo menyimpan 9 persen dari biaya yang dibayarkan pengguna untuk konten, dan kembali 91 persen kepada pemilik konten.

M-Commerce yang berhubungan dengan layanan *mobile* menyebar dengan cepat pada awal 2000 . Norwegia meluncurkan ponsel pembayaran parkir. Austria menawarkan tiket kereta melalui perangkat *mobile*. Jepang menawarkan pembelian tiket penerbangan secara *mobile*.

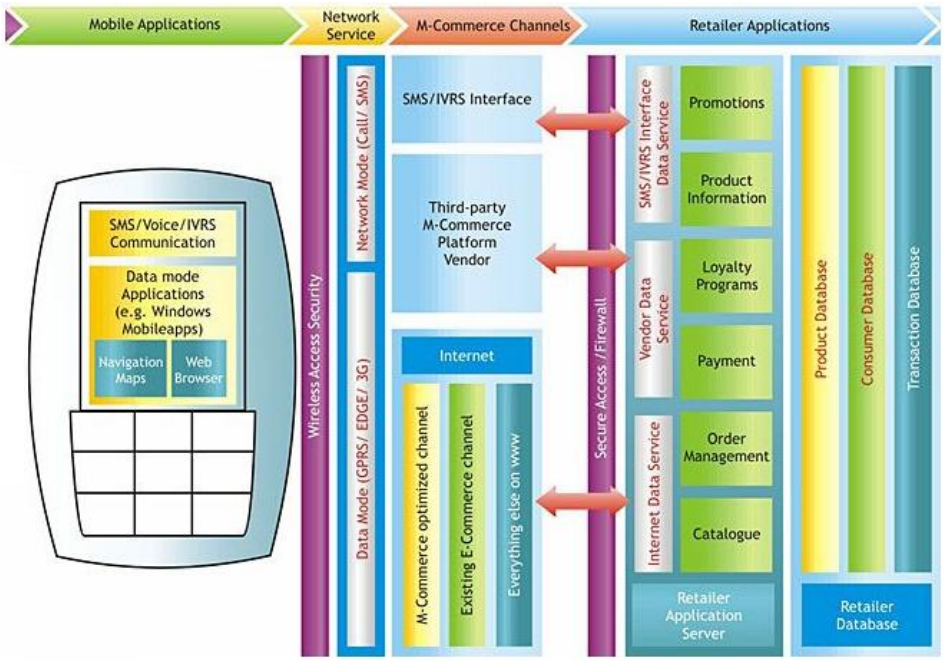
PDA dan *smartphone* telah menjadi sangat populer sehingga banyak yang melakukan bisnis dengan menggunakan *m-commerce* sebagai cara yang lebih efisien untuk berkomunikasi dengan pelanggan mereka. Dalam rangka untuk mengeksploitasi perdagangan pasar ponsel yang potensial, produsen ponsel seperti Nokia, Ericsson, Motorola, dan Qualcomm bekerja dengan operator seperti AT&T Wireless dan Sprint untuk mengembangkan *WAP enabled smartphone*. *Smartphone* menawarkan faks, *e-mail*, dan kemampuan telepon. Sejak peluncuran iPhone, *m-commerce* telah bergerak menjauh dari sistem SMS. SMS memiliki kerentanan keamanan yang signifikan dan masalah kemacetan, meskipun banyak tersedia dan mudah diakses. Selain itu, peningkatan dalam kemampuan perangkat *mobile* yang modern membuatnya bijaksana untuk menempatkan beban lebih dari sumber daya pada perangkat *mobile*.

Baru-baru ini, pemilik usaha batu bata dan mortir telah membuat upaya untuk mengambil keuntungan dari *m-commerce* dengan memanfaatkan sejumlah kemampuan *mobile* seperti layanan berbasis lokasi **(***location based servers*),

pemindahan *barcode*, dan lebih ke sistem pemberitahuan (*notification*) untuk menginformasikan pelanggan dalam melakukan proses transaksi berbelanja di toko. Dengan adanya *update notifikasi* dari toko memungkinkan pelanggan untuk mengakses manfaat umum dari belanja *online* (seperti *review* produk, informasi, dan kupon). Katalog dari penjual dapat menginformasikan produk kepada pelanggan secara elektronik, melalui perangkat *mobile* pelanggan. Dalam beberapa kasus, pedagang bahkan mungkin memberikan katalog elektronik, dari pada katalog dalam bentuk kertas untuk pelanggan. Beberapa pedagang menyediakan situs *mobile* yang disesuaikan untuk layar yang lebih kecil dan antarmuka pengguna terbatas perangkat *mobile*.

2.3.2 Arsitektur M-Commerce

Secara umum, arsitektur *m-commerce* dapat dilihat pada gambar II.2



Gambar II.2 Arsitektur Mobile Commerce[10]

1. ***Mobile Application***, Aplikasi *mobile* seperti *user interface* untuk web toko anda, dimana pengguna dapat mempersonalisasikan aplikasi dari produk yang telah diterbitkan di *Web Store*.
2. ***Network Server***, dasar dari lingkungan komputasi jaringan. Umumnya layanan jaringan yang diinstal pada satu atau lebih *server* untuk menyediakan sumber daya bersama untuk komputer *client*.
3. ***M-Commerce Channels***, Teknologi jaringan *mobile* yang menghubungkan pengguna dengan *retailer applications*.

4. ***Retailer Apllications***, Aplikasi *mobile* menampilkan produk yang akan dijual dan menciptakan cara yang menarik dan dinamis untuk berinteraksi dengan pelanggan.

2.3.3 Produk dan Layanan Yang Tersedia

2.3.3.1 Mobile Ticketing

Mobile ticketing adalah proses dimana pelanggan dapat memesan, membayar, mendapatkan dan memvalidasi tiket dari setiap lokasi dan kapan saja dengan menggunakan ponsel atau handset *mobile* lainnya. *Mobile ticketing* mengurangi biaya produksi dan distribusi yang berhubungan dengan kertas, kemacetan yang disebabkan pengantri tiket dapat mengahabiskan banyak waktu untuk pelanggan. Meningkatkan kenyamanan pelanggan dengan memberikan cara-cara baru dan sederhana untuk membeli tiket. *mobile ticketing* adalah contoh utama konvergensi telekomunikasi horisontal.

Tiket dapat dikirim ke ponsel dengan menggunakan berbagai teknologi. Tiket dapat dipesan dan dibatalkan pada perangkat *mobile* dengan mengunduh aplikasi bantuan sederhana, atau dengan mengakses portal WAP (*Wireless Application Protocol*) dari berbagai agen perjalanan atau penyedia layanan langsung. Tiket *mobile* dapat dibeli dalam berbagai cara termasuk *online*, melalui pesan teks atau melalui telepon dari panggilan suara, Halaman WAP, atau aplikasi *mobile* yang aman.

2.3.3.2 Location Based Server

Location based server adalah adalah layanan informasi atau hiburan, dapat diakses dengan perangkat *mobile* melalui jaringan selular dan kemampuan untuk memanfaatkan posisi geografis perangkat *mobile*. LBS dapat digunakan dalam berbagai konteks, seperti kesehatan, cari objek dalam ruangan, hiburan, pekerjaan, kehidupan pribadi, dll.

LBS termasuk layanan untuk mengidentifikasi lokasi seseorang atau benda, seperti menemukan mesin ATM terdekat perbankan atau keberadaan seorang teman atau karyawan. LBS meliputi pelacakan paket dan pelacakan kendaraan layanan. LBS dapat mencakup *m-commerce* ketika mengambil bentuk kupon atau iklan diarahkan pada pelanggan berdasarkan lokasi mereka saat ini.

2.3.3.3 Content Purchase and Delivery

Saat ini, pembelian dan pengiriman konten *mobile* digunakan sebagai sarana dari penjualan nada dering, *wallpaper*, dan *game* untuk ponsel. Konvergensi dari ponsel, pemutar audio portabel, dan pemutar video dalam satu perangkat akan meningkatkan transakasi pembelian dan pengiriman trek musik dan video. Kecepatan *download* yang disediakan dengan jaringan 4G memungkinkan konsumen untuk membeli film pada perangkat mobile dalam waktu yang singkat.

2.3.3.4 Mobile marketing

Penggunaan media ponsel sebagai sarana komunikasi pemasaran atau distribusi pesan dari setiap promosi atau iklan untuk pelanggan melalui jaringan

nirkabel. *Mobile marketing* dapat merujuk kepada salah satu dari dua kategori bunga, dimaksudkan untuk menggambarkan pemasaran dengan menggunakan perangkat *mobile*, seperti ponsel yang menggunakan SMS *Marketing* .

2.3.3.4.1 Mobile Marketing via SMS Marketing

Pemasaran pada ponsel dikenal sebagai SMS *Marketing*, SMS *Marketing* menjadi semakin populer sejak munculnya SMS (*Short Message Server*) pada awal 2000-an di Eropa dan beberapa bagian Asia ketika bisnis mulai mengumpulkan nomor ponsel dan ingin mengirimkan konten. Rata-rata, pesan SMS yang dibaca dalam waktu empat menit, membuat mereka sangat konversi. Selama beberapa tahun terakhir SMS *Marketing* telah menjadi saluran periklanan yang sah di beberapa bagian dunia. Karena Hal ini tidak seperti *e-mail* yang melalui internet publik, jaringan mereka sendiri telah menetapkan pedoman dan praktik terbaik untuk industri media *mobile* (termasuk iklan *mobile*). IAB (*Interactive Advertising Bureau*) dan *Mobile Marketing Association* , juga, telah menetapkan pedoman dan penginjilan penggunaan saluran mobile bagi pemasar.

Pemasaran melalui SMS telah berkembang pesat di Eropa dan Asia sebagai saluran baru untuk mencapai konsumen. SMS pada awalnya menerima liputan negatif di banyak bagian eropa. *Spam* mengirimkan beberapa konten yang tidak diinginkan ke konsumen, namun, sebagai pedoman yang diberlakukan oleh operator selular, SMS menjadi yang paling populer di cabang industri *Mobile Marketing* dengan beberapa juta SMS iklan yang dikirim setiap bulannya.

2.3.3.4.2 Mobile Marketing via MMS

Multimedia Messaging Server, atau MMS, adalah cara standar untuk mengirim pesan yang meliputi konten multimedia dari ponsel. Ini memperluas inti SMS (*Short Message Server*) kemampuan yang memungkinkan pertukaran pesan teks hanya sampai 160 karakter. Penggunaan yang paling populer adalah untuk mengirim foto-foto dari kamera, meskipun populer sebagai metode penyampaian berita dan konten hiburan termasuk video, gambar, halaman teks dan nada dering. Standar ini dikembangkan oleh *Open Mobile Alliance* (OMA).

MMS dapat berisi *mobile marketing* gambar, audio, teks, dan video. mobile konten ini disampaikan melalui MMS (*Multimedia Message Server).* Hampir semua ponsel baru diproduksi dengan layar warna yang mampu mengirim dan menerima standar pesan MMS. Kedua merek dapat salaing mengirim dan menerima banyak konten melalui MMS A2P (*application-to-person*) dengan jaringan *mobile* yang ditujukan kepada pengguna. Pada beberapa jaringan, merek juga dapat mensponsori pesan yang dikirim P2P (*person-to-person*).

2.3.3.4.3 Mobile Marketing via Bluetooth

Bluetooth adalah protokol untuk komunikasi nirkabel. Perangkat seperti ponsel, laptop, PC, printer, kamera digital, dan konsol video game dapat terhubung satu sama lain, dan dapat saling bertukaran informasi. Hal ini dilakukan dengan menggunakan gelombang radio. Awalnya, *bluetooth* dikembangkan untuk mengurangi jumlah kabel yang diperlukan untuk menghubungkan perangkat tersebut ke PC. *Bluetooth* hanya digunakan untuk jarak yang relatif singkat.

Ada standar yang berbeda. Kecepatan data bervariasi. Saat ini, *bluetooth* berada pada 1-3 MBit per detik. Aplikasi *bluetooth* khas untuk menghubungkan *headset* ke telepon selular, atau untuk menghubungkan *mouse* komputer, *keyboard* atau *printer*. Munculnya *bluetooth* dimulai sekitar 2003 dan beberapa perusahaan telah mulai membangun bisnis yang sukses. Kebanyakan dari bisnis ini menawarkan (*hotspot*) sistem yang terdiri dari beberapa jenis konten sistem manajemen distribusi dengan fungsi *bluetooth*. Teknologi ini memiliki keuntungan kecepatan transfer yang lebih tinggi dan juga merupakan teknologi berbasis radio, sehingga tidak dapat ditagih (gratis). Perangkat awal kemungkinan dibangun untuk pemasaran ponsel melalui *bluetooth*.

2.3.3.4.4 Mobile Marketing via Proximity Systems

Proximity marketing adalah distribusi lokal nirkabel dari konten iklan yang terkait dengan tempat tertentu. Komunikasi ditargetkan untuk kelompok tertentu dalam lokasi tertentu, misalnya untuk konten di tempat-tempat wisata hanya dapat didistribusikan ke perangkat yang terdaftar di luar area lokal.

Mobile marketing via Proximity Systems, disebut juga sebagai *Proximity Marketing*, bergantung pada GSM 03,41 yang mendefinisikan *Short Message Server - Cell Broadcast*. SMS-CB memungkinkan pesan (iklan, informasi publik, dll) yang akan disiarkan ke semua pengguna ponsel di area geografis tertentu.

2.3.3.4.5 Mobile Marketing via Location Based servers

Layanan berbasis lokasi (LBS) yang ditawarkan oleh beberapa jaringan telepon seluler sebagai cara untuk mengirim iklan dan informasi lainnya ke ponsel

pelanggan berdasarkan lokasi mereka saat ini. Penyedia layanan telepon seluler mendapatkan lokasi dari sebuah *chip* GPS dibangun ke dalam telepon, atau menggunakan penetapan tempat dengan pesawat radio dan *Trilateration* yang didasarkan pada kekuatan sinyal dari menara ponsel terdekat (untuk ponsel tanpa fitur GPS).

Sementara itu, LBS dapat diaktifkan tanpa teknik pelacakan GPS. Teknologi *mobile* WiMAX digunakan untuk memberikan dimensi baru untuk pemasaran *mobile*. Jenis baru dari pemasaran *mobile* dibayangkan antara BS (*Base Station*) dan berbagai CPE (*Consumer Premise Equipment*) dipasang pada *dashtop* kendaraan. Setiap kali kendaraan datang dalam jarak efektif BS, CPE *dashtop* dengan LCD touchscreen dapat diaktifkan dengan menyentuh jari atau tag suara.

2.3.3.4.6 Mobile marketing via QR codes

QR (*Quick Response*) kode telah tumbuh dan berkembang dalam popularitas di Asia dan Eropa. Pertama kali dikembangkan standar untuk pelacakan suku cadang kendaraan bermotor di Jepang. Bertindak sebagai *link hiper visual* untuk halaman. Kode QR merupakan sebuah teknologi *open source* dan generator gratis yang dapat ditemukan dengan mudah.

Perusahaan khusus menawarkan solusi pemasaran terintegrasi dengan menggabungkan generasi kode dengan fitur pelacakan dalam proses penjualan dan memberikan informasi produk lebih lanjut.

2.3.3.4.7 Mobile Banking

Mobile banking (juga dikenal sebagai M-Banking, mBanking, SMS Banking) adalah istilah yang digunakan untuk melakukan cek saldo, transaksi rekening, pembayaran, aplikasi kredit dan transaksi perbankan lainnya melalui perangkat *mobile* seperti ponsel atau *Personal Digital Assistant* (PDA).

Mobile banking hingga saat ini (2010) yang paling sering dilakukan melalui SMS atau *Mobile Web*. Apple sukses diawal dengan iPhone dan pertumbuhan yang cepat dari ponsel berbasis Google Android (sistem operasi) telah menyebabkan meningkatnya penggunaan program khusus *client* yang disebut aplikasi, yang kemudian diunggah ke perangkat mobile. *Mobile Banking* mengacu pada penyediaan dan layanan perbankan dan keuangan dengan bantuan telekomunikasi *mobile devices*. Lingkup layanan yang ditawarkan mungkin termasuk fasilitas untuk melakukan transaksi di bank dan pasar saham, untuk mengelola *account* dan untuk mengakses informasi disesuaikan.

2.3.4 Kelebihan M-Commerce

Secara sederhana, perbedaan antara proses perdagangan secara manual dengan menggunakan *m-commerce* dapat jelas terlihat, dimana pada proses dengan *m-commerce* dapat digunakan dimana pun pengguna berada dan dalam situasi apa pun. Dengan kata lain layanan *m-commerce* tersedia kapan pun dan dimana pun dibutuhkan. *m-commerce* lahir setelah *e-commerce* yang pada umumnya dilakukan melalui media internet. Kelahiran *m-commerce* tersebut dipicu oleh tingginya tingkat penetrasi *handphone* di seluruh dunia. Dibandingkan sistem *e-commerce* lainnya, kelebihan *m-commerce* adalah sebagai berikut:

1. ***Ubiquity***, aplikasi *m-commerce* dapat digunakan dimana pun pengguna berada dan dalam situasi apa pun. Dengan kata lain layanan *m-commerce* tersedia kapan pun dan dimana pun dibutuhkan.
2. ***Reachability***, melalui perangkat *handset* pengguna dapat bertransaksi dan berkomunikasi dengan pihak lain yang tidak terjangkau karena perbedaan ruang dan waktu.
3. ***Localization***, *m-commerce* memberikan layanan aplikasi informasi lokasi yang dibutuhkan oleh pengguna berdasarkan lokasi fisik penggunanya (*location based servers*).
4. ***Personalization***, aplikasi m-commerce hanya dapat digunakan oleh pengguna yang memiliki *handset* karena adanya perbedaan penomoran atau nomor unik yang dikeluarkan oleh penyelenggara telekomunikasi.
5. ***Dissemination***, *handset* memungkinkan pengguna menerima dan menyebarluaskan informasi secara cepat dan kini (saat itu juga).
6. ***Security***, pada umumnya *handset* dilengkapi dengan *smart card reader* dan *smart card*-nya itu sendiri. Sehingga dapat digunakan sebagai *secret authentication key.*
7. ***Convenience***, ukuran dan kualitas dari *handset* membuat pengguna nyaman dalam bertransaksi.

2.3.5 Kekurangan M-Commerce

Walaupun adanya *m-commerce* memberi banyak keuntungan, masih terdapat berbagai kekurangan dari *m-commerce* antara lain :

1. Keterbatasan perangkat.
2. Tingkat keberagaman perangkat, jaringan dan sistem operasi yang sangat tinggi, membutuhkan standardisasi *platform* antar vendor.
3. Tingginya tingkat kehilangan / pencurian *handphone*.
4. Bertambahnya tingkat kerawanan terhadap *security* ketika data ditransfer.
5. *Mobile device* tidak bisa menawarkan grafik atau daya proses layaknya PC.
6. Layar yang kecil dari *mobile device* membatasi kompleksitas aplikasi.
7. Masing-masing jaringan memiliki pendekatan yang berbeda terhadap pemahaman *m-commerce*.

2.3.6 Manfaat M-Commerce

Manfaat yang diperoleh bagi organisasi pemilik *M-Commerce* :

1. Memperluas *market place* hingga kepasar nasional dan internasional.
2. *M-Commerce* menurunkan biaya pembuatan, pemrosesan, pendistribusian, penyimpanan, dan pencarian informasi yang menggunakan kertas.
3. *M-Commerce* mengurangi waktu antara penerimaan produk dan jasa.
4. *M-Commerce* memperkecil biaya telekomunikasi.
5. Akses informasi menjadi lebih cepat.
6. Mengurangi biaya transportasi dan menigkatkan fleksibilitas.

Manfaat bagi konsumen antara lain :

1. *M-Commerce* memungkinkan pelanggan untuk berbelanja atau melakukan transaksi selama 24 jam sehari sepanjang tahun dari hampir setiap lokasi.
2. *M-Commerce* memberikan lebih banyak pilihan kepada pelanggan, mereka bisa memilih berbagai produk dari banyak vendor.
3. *M-Commerce* menyediakan produk dan jasa yang tidak mahal kepada pelanggan dengan cara mengunjungi banyak tempat dan melakukan perbandingan secara cepat.
4. Pelanggan bisa menerima informasi yang relevan secara detil dalam hitungan detik, bukan lagi hari atau minggu.

Manfaat bagi masyarakat antara lain :

1. *M-Commerce* memungkinkan orang untuk bekerja didalam rumah dan tidak harus keluar rumah untuk berbelanja. Ini berakibat menurunkan arus kepadatan lalu lintas dijalan serta mengurangi polusi udara.
2. *M-Commerce* memungkinkan orang di negara-negara dunia ketiga dan wilayah pedesaan untuk menikmati aneka produk dan jasa yang akan susah mereka dapatkan tanpa *M-Commerce*.

2.3.7 Klasifikasi M-Commerce

Penggolongan m-commerce yang lazim dilakukan orang ialah berdasarkan sifat transaksinya, antara lain :

1. *Business to Business* (B2B)

   Jenis transaksi dimana pembeli biasanya membeli dalam jumlah besar karena akan dijual kembali.

2. *Business to Consumer* (B2C)

   Jenis transaksi dimana pembelinya perorangan dan tidak punya tujuan untuk menjualnya kembali biasanya semacam toko online yang menjual berbagai macam barang.

3. *Consumer to Consumer* (C2C)

   Jenis transaksi dimana pembelinya perorangan yang tidak mempunyai tujuan untuk dijual kembali dan penjualnya juga perorangan yang tidak menyediakan bermacam-macam barang melainkan hanya beberapa barang saja.

4. *Consumer to Business* (C2B)

   Termasuk ke dalam kategori ini adalah perseorangan yang menjual produk atau layanan kepada organisasi, dan perseorangan yang mencari penjual, berinteraksi dengan mereka dan menyepakati suatu transaksi.

2.3.8 Metode Pembayaran

Konsumen dapat menggunakan berbagai bentuk pembayaran dalam *mobile commerce*, termasuk:

1. *Premium-rate telephone numbers*, yang berlaku biaya untuk konsumen jarak jauh tagihan
2. Biaya ditambahkan ke tagihan telepon seluler konsumen, termasuk potongan untuk rencana panggilan pra-bayar
3. Kartu kredit, Sebuah sistem pembayaran sederhana *web mobile* ini juga dapat mencakup kartu kredit yang memungkinkan aliran pembayaran konsumen untuk memasukkan rincian kartu mereka untuk melakukan pembelian.
4. *Layanan Micropayment*
5. *Stored value cards*, sering digunakan dengan aplikasi perangkat *mobile* atau toko-toko music
6. *Direct Mobile Billing*, Konsumen menggunakan opsi penagihan ponsel selama checkout pada situs *e-commerce* untuk melakukan pembayaran.
7. *Mobile web payments* (WAP), Konsumen menggunakan halaman web yang ditampilkan atau aplikasi tambahan diunduh dan diinstal pada ponsel untuk melakukan pembayaran.

Online Wallets, Perusahaan *online* seperti *Paypal*, Pembayaran *Amazon* dan Google *Checkout* juga memiliki opsi seluler. Sistem ini dapat diintegrasikan

dengan secara langsung atau dapat dikombinasikan dengan operator dan pembayaran kartu kredit melalui *platform* web pembayaran *mobile*.

2.3.9 Sistem M-Commerce

Terdapat sejumlah besar *payment* sistem untuk *e-commerce* dan *m-commerce*. Beragamnya sistem tersebut disebabkan oleh faktor-faktor sebagai berikut:

1. Waktu Pembayaran.

   Perbedaan yang terletak pada waktu pembayaran yang dilakukan oleh pengguna relatif terhadap waktu transaksi, yaitu apakah dilakukan pada saat transaksi, sebelum transaksi dilakukan atau setelah transaksi dilakukan.

2. Jumlah Pembayaran

   Pada dasarnya dapat dibagi menjadi dua, yaitu transaksi dengan jumlah pembayaran besar, dan kecil. Terdapat perbedaan yang signifikan antara transaksi besar dan kecil. Pada transaksi dengan nilai yang besar, perlu dilakukan authentikasi melalui institusi finansial yang terpercaya. Sedangkan untuk transaksi kecil, authentikasi cukup hanya dilakukan pada level jaringan operator, antara lain melalui SIM *Card*.

3. Isu Anonim

   Anonim berarti indentitas pengguna / pelanggan tidak dapat diketahui oleh merchant. Terdapat sistem yang sepenuhnya anonim, anonim parsial atau bahkan tidak anonim. Masing-masing sifat tersebut dibutuhkan untuk transaksi yang berbeda-beda.

4. Validasi yang dilakukan secara *online* atau *offline*

Pada transaksi *M-Commerce*, tahap-tahap transaksi pada dasarnya adalah mirip dengan yang terjadi pada transaksi konvensional. Hanya saja pada kasus *remote payment*, pengiriman detail informasi transaksi dilakukan melalui jaringan seluler. Sehingga keamanan jaringan seluler juga menjadi perhatian khusus. Selain itu, proses pengiriman informasi transaksi juga melibatkan *protocol browser*, yang berupa WAP(*Wireless Application Protocol*), atau *protocol system messaging* seperti SMS (*ShortMessage Server*) dan USSD (*Unstructured Supplementary Server Data*). Keamanan dari setiap *server* tersebut juga harus diperhatikan untuk dapat mendukung sistem *M-Commerce* yang aman. Selain transaksi yang bersifat *remote, M-Commerce* juga dapat dilakukan untuk transaksi lokal, yang pada umumnya menggunakan koneksi *bluetooth,infrared* atau RFID.

Secara umum, tahapan proses pada *m-commerce* dapat dibedakan menjadi 4 tahap sebagai berikut:

1. *Set-up* dan Konfigurasi

   Proses ini termasuk instalasi aplikasi khusus pada *handset* yang akan digunakan pada *M-Commerce*. Selain itu, untuk beberapa sistem *M-Commerce* proses ini juga melibatkan proses pembelian atau penambahan nilai uang pada aplikasi tersebut.

2. Inisiasi Pembayaran

   Pada tahap ini informasi pembayaran dikirimkan melalui jaringan seluler atau protokol *wireless* lainya kepada *merchant*.

3. Authentikasi

Tahap ini merupakan tahap yang paling penting pada transaksi. Pada tahap ini diperiksa apakah pengguna memang berhak melakukan transaksi, serta memenuhi persyaratan finansial tertentu. Pada sebagian sistem pembayaran, proses ini melibatkan authentikasi berdasarkan SIM *Card*.

4. Penyelesaian Pembayaran

   Proses ini dilakukan ketika pengguna telah berhasil di-authentikasi, demikian juga transaksi itu sendiri telah berhasil di-authentikasi. Analoginya pada proses transaksi konvensional adalah dengan dicetaknya bukti pembayaran.

2.3.10 Teknologi Jaringan Mobile

Dari tahun ke tahun, teknologi jaringan *mobile* telah mengalami evolusi mulai dari sistem analog hingga ke sistem digital. Perkembangan ini dikategorikan menjadi beberapa standar teknologi di setiap generasi, yaitu:

1. Teknologi 1G

   Generasi pertama jaringan *mobile* menggunakan teknologi nirkabel analog, dimana setiap kanal *voice* mempunyai pita frekuensi sempit yang menggunakan teknologi akses jamak divisi frekuensi (*Frequency Division Multiple Access*, FDMA). Teknologi *mobile* pertama ini berkembang sebelum tahun 1997 dan mencapai sukses yang luar biasa. Pada akhir tahun 1997, jumlah penggunajaringan *mobile* di seluruh dunia telah mencapai 210 juta dan di akhir tahun 2001 diperkirakan mencapai 700 juta, sehingga spektrum pita frekuensi radio yang tersedia tidak lagi mencukupi untuk mengakomodasi jumlah pengguna jaringan *mobile* yang terus bertambah.

2. Teknologi 2G

   Evolusi teknologi *mobile* di tahun 1997-1999 adalah perubahan dari teknologi nirkabel analog menjadi teknologi nirkabel digital, dimana setiap kanal frekuensi yang lebih lebar dibagi menjadi celah waktu periodik (*Time Division Multiple Access*, TDMA). Ini berarti bahwa beberapa panggilan menggunakan kanal frekuensi secara bersama pada setiap satuan waktu tertentu, sehingga memungkinkan layanan untuk jumlah pengguna yang lebih besar, di atas frekuensi yang tersedia Standar teknologi yang terkenal pada generasi ini adalah standar GSM yang tidak hanya menggunakan teknologi nirkabel digital, tapi juga meletakkan dasar-dasar jelajah internasional (*international roaming*), sehingga pengguna GSM dapat membawa terminal telepon *mobile* miliknya pada saat melakukan perjalanan dari satu negara ke negara lain, ia masih dapat membuat dan menerima panggilan.

3. Teknologi 2.5G atau 2G+

   Teknologi GSM terus berkembang dan telah memberikan suatu kelompok layanan suara dan data yang kaya akan fitur, sehingga muncullah standar baru yaitu standar GPRS yang memungkinkan layanan jaringan data yang selalu siap pakai, kapasitas lebih tinggi, akses *content* layanan berbasis Internet atau berbasis komunikasi PSD (*Packet Switch Data*). Dan teknologi *mobile* ini akan terus berkembang dengan sangat cepat, saat ini sudah mencapai teknologi 3G bahkan sudah dirancang teknologi 4G, sehingga di masa mendatang teknologi *mobile* merupakan teknologi yang banyak diminati dan sangat menjanjikan.

4. Teknologi 3G

   Third generation technology merupakan sebuah standar yang ditetapkan oleh *International Telecommunication Union* (ITU) yang diadopsi dari IMT-2000 untuk diaplikasikan pada jaringan telepon selular. Istilah ini umumnya digunakan mengacu kepada perkembangan teknologi telepon nirkabel versi ke-tiga. Pada dasarnya perkembangan teknologi komunikasi ini disebabkan oleh keinginan untuk selalu memperbaiki kinerja, kemampuan dan efisiensi dari teknologi generasi sebelumnya.

5. Teknologi 4G

   4G adalah singkatan dari *fourth generation technology*. Istilah ini umumnya digunakan mengacu kepada pengembangan teknologi telepon seluler. 4G merupakan pengembangan dari teknologi 3G. Nama resmi dari teknologi 4G ini menurut IEEE (*Institute of Electrical and Electronics Engineers*) adalah "*3G and beyond*".

2.3.10.1 Fasilitas dan Layanan Jaringan Mobile

Standar teknologi *mobile* yang digunakan memiliki fungsi dan layanan masing-masing. Ada dua standar umum yang digunakan di dunia saat ini, yaitu:

1. GSM (*Global System for Mobile Communication*)

   GSM merupakan teknologi generasi kedua (2G) dalam dunia komunikasi *mobile* dan standar yang paling banyak digunakan di seluruh dunia Layanan yang disediakan oleh GSM meliputi layanan telekomunikasi (*teleservers*), layanan data (*data server*) dan juga terdapat layanan tambahan seperti layanan penerusan panggilan (*call forwading*), pengalihan panggilan (*divert call*), identifikasi pemanggil (*call identification*), panggilan menunggu (*call waiting*) dan percakapan beberapa pengguna sekaligus (*multiparty conversation*). Kelemahan yang paling besar dari GSM adalah kecepatannya masih rendah hanya 9,6 Kbps, sehingga untuk ke depan diprediksikan akandigantikan oleh teknologi GPRS, karena GPRS memiliki *bandwidth* dan kecepatan yang tinggi.

2. GPRS (*General Packet Radio System*)

   GPRS masih merupakan kelompok dari generasi teknologi kedua (2G) yang dikembangkan dan diberi nama teknologi 2.5G atau 2G+. Kecepatan transmisi datanya mencapai 115 Kbps. GPRS memiliki kemampuan menampilkan foto dan video dengan kecepatan tinggi. Kemampuan tambahan yang dimiliki GPRS. Pertama, dapat memelihara keutuhan komunikasi data dan suara pada saat sedang bergerak. Kedua, pengguna dapat segera terhubung ke nomor

yang dituju kapan pun mereka menginginkannya, sehingga tidak tergantung pada lokasi dimanapun berada tanpa mengalami *delay* yang lama. Ketiga, dengan tingkat kecepatan tinggi yang dimiliki GPRS sangat dimungkinkan untuk melakukan proses *download file*. Fasilitas yang dimiliki oleh GPRS adalah dapat mengakses intranet, menggunakan *e-mail* dan fax serta *unified messaging* (pesan terpadu, satu pesan untuk semua jenis komunikasi). Selain itu, terdapat juga layanan seperti *e-commerce*, *m-commerce*, *e-banking*, transfer antar rekening dan pembayaran tagihan.

3. HSPA (*High Speed Packet Access*)

   HSPA adalah koleksi protokol telepon genggam dalam ranah 3,5G yang memperluas dan memperbaiki kinerja protokol *Universal Mobile Telecommunications System* (UMTS). Jaringan HSPA sebagian besar tersebar pada spektrum 1900 MHz dan 2100 MHz namun beberapa berjalan pada 850 MHz. Spektrum yang lebih besar digunakan karena operator dapat menjangkau area yang lebih luas serta kemampuannya untuk *refarming* dan realokasi spektrum UHF.

2.3.11 Security M-Commerce

Pihak-pihak yang terkait dalam transaksi *mobile payment* adalah pengguna (pembeli), operator jaringan, institusi *financial* dan *merchant* (penyedia produk / jasa yang akan dibeli pengguna). Seluruh pihak tersebut memiliki kebutuhan akan jaminan *security* sebagai berikut:

1. Pengguna, menuntut dijaminnya *security* pada *account*-nya, dan juga *privacy*, sehingga pihak lainnya sedapat mungkin tidak mengetahui identitas pribadi pengguna.
2. Operator jaringan seluler, merupakan fasilitator dari *m-commerce*, memiliki perhatian khusus dalam menyampaikan informasi transaksi secara aman melalui jaringannya.
3. Institusi *financial*, memiliki perhatian khusus pada integritas dari sistem pembayaran, sehingga dapat mengurangi resiko terjadinya *fraud* atau *bad-debt*.
4. *Merchant*, menuntut agar proses pembayaran menjadi mudah disisi pengguna, sehingga akan membangkitkan transaksi. Selain itu *merchant* juga menuntut agar pembayaran dari institusi *financial* dapat berlangsung dengan sempurna.

Security memegang peranan penting dalam *m-commerce*. Karena hanya dengan jaminan *security M-Commerce* dapat dijamin *cost effective* sehingga layak untuk diselenggarakan. Elemen-elemen security yang perlu diperhatikan dalam *m-commerce* adalah sebagai berikut:

1. *Authentikasi*, yang memungkinkan pihak fasilitator pembayaran (antara lain institusi keuangan) untuk memastikan bahwa pihak yang menggunakan sistem pembayaran adalah pihak yang berhak.
2. *Confidentiality*, yang memastikan bahwa pihak lain yang tidak berhak tidak dapat mengakses data pembayaran.
3. *Data Integrity*, yang memastikan bahwa data pembayaran tidak berubah setelah pengguna menyetujui seluruh detail transaksi.
4. *Non-repudiation*, yang mengikat seluruh pihak yang terlibat sehingga tidak dapat menyangkal seluruh proses yang telah dilakukannya.

Transaksi *m-commerce* pada umumnya berlangsung melalui media *transport* yang disediakan oleh jaringan seluler. Sehingga perlu diperhatikan juga keamanan dari jaringan yang digunakan.

Selain jaringan operator, perlu diperhatikan juga keamanan dari *handset* dan aplikasi *m-commerce* yang digunakan. Namun demikian, terdapat beberapa keterbatasan yang diakibatkan oleh:

1. Koneksi yang bersifat dinamik, melalui beberapa jaringan akses yang terkadang tidak aman atau tidak dapat dipercaya.
2. Keterbatasan yang ditimbulkan oleh protokol komunikasi, antara lain *bandwidth* dan *latency*.
3. Keterbatasan perangkat, antara lain daya dan kinerjanya.
4. Belum matangnya teknologi *client*

2.3.11.1 Security Pada Level Jaringan

Security dari teknologi jaringan *mobile :*

1. GSM

   Merupakan sistem dengan pengguna terbanyak saat ini. Pada awalnya hanya mendukung koneksi *circuit* data 9,6 Kbps. Layanan saat ini SMS, WAP, GPRS, High speed CSD. *Security* pada GSM menggunakan IMSI (kode internasional pelanggan) dan Ki (kunci yang digunakan untuk authentikasi) yang disimpan di SIM *Card*. *Enkripsi air interface* menggunakan *symmetric key* yang diturunkan dari Ki. Kelemahan utama dari GSM adalah bahwa jaringan inter operator tidak terenkripsi. Kelemahan lainnya adalah authentikasi hanya dilakukan satu arah, yaitu dari sisi jaringan kepada perangkat *end-user*, sehingga dimungkinkan terjadinya penyadapan oleh perangkat yang ‘menyamar’ sebagai BTS dari sisi perangkat *end-user*.

2. UMTS

   Pada jaringan UMTS telah dilakukan perbaikan terhadap aspek *security* dibandingkan yang dimiliki oleh jaringan GSM. Perbaikan tersebut mencakup proses authentikasi dan enkripsi data pelanggan. Proses authentikasi yang semula dilakukan secara satu arah, pada jaringan UMTS dilakukan secara mutual, yaitu dari *network* kepada handset dan dari handset kepada *network*. Sedangkan dari sisi pengamanan pesan atau percakapan yang dikirimkan oleh pengguna, proses enkripsi selalu dilakukan secara *end-to-end*, kecuali jika baik *handset* dan *network* sepakat untuk berkomunikasi tanpa enkripsi. *Integrity protection* digunakan untuk menjaga *signaling message*. UMTS

menggunakan algoritma *chipper* yang baru dan *encryption keys* yang lebih panjang.

3. *Wireless* LAN Secara default tidak menerapkan security pada air *interface*-nya. Kemudian IEEE menerapkan WEP (*Wired Equivalent Privacy*), yang memberikan:
    a. Autentikasi kepada *Access Point*.
    b. *Integrity* dan *Confidentiality* dari MAC Frame.
4. Bluetooth

    Ad hoc piconet, pada personal *environment*, sangat potensial untuk *m-commerce*. Pada *link layer security* dilakukan dengan *challenge response protocol* untuk authentikasi dan stream *chipper algorithm* untuk user dan *signaling data*. Jika perangkat-perangkatnya tidak saling menukarkan *key*, maka yang ditukarkan adalah PIN yang harus sama, dapat dengan cara input manual pada *handset* maupun dengan cara mengimportnya dari aplikasi yang sama.

2.3.11.2 Security Pada Level Transport

Kebanyakan transaksi berlangsung dengan tidak hanya melibatkan jaringan akses saja, tapi juga sering melibatkan jaringan yang dikelola oleh pihak ketiga. Sehingga diperlukan juga jaminan keamanan yang bersifat *end-to-end*.

1. SSL/TLS

   Merupakan salah satu *protocol* yang paling banyak digunakan di internet saat ini. Salah satu penerapannya adalah HTTPS. SUN telah berhasil mengembangkan versi *client side* dari SSL untuk device dengan keterbatasan *processing* dan *memory*, dinamakan *kilobytes* SSL (KSSL). Walaupun tidak mendukung *client side authentication*, dan hanya mendukung beberapa *chipper* yang umum digunakan, namun kelebihan SSL adalah dapat dijalankan di atas platform J2ME, dengan hanya menghabiskan sedikit *memory*.

2. WTLS

   WAP forum telah menstandarkan *protocol security* pada *layer transport* yang disebut WTLS, dan mengimplentasikannya pada WAP 1. WTLS memfasilitasi sekuriti antara perangkat *mobile* dengan WAP *Gateway*, yang selanjutnya melakukan konversi kepada SSL/TLS. Sehingga memang belum memfasilitasi *security* yang sifatnya *end-to-end*, dan WAP *Gateway* harus dapat dipercaya. Namun demikian, saat ini WAP forum telah mengusulkan suatu *stack protocol* WAP 2, yang identik dengan TCP/IP untuk media *wireless*. Sehingga dengan demikian *security* yang sifatnya *end-to-end* dapat diselenggarakan.

2.3.11.3 Security Pada Level Server

1. SMS

   Walaupun hanya dapat memfasilitasi maksimal 160 karakter. Pengirim dan penerima SMS diidentifikasi dengan menggunakan IMSI, sehingga keamanan layanan ini termasuk tinggi, karena didukung oleh keamanan GSM itu sendiri (keamanan SIM *Card*). Sehingga SMS dapat digunakan sebagai authentikasi, setidaknya pada jaringan GSM itu sendiri. Selanjutnya pesan SMS dikirimkan melalui *signaling plane* GSM, dimana tidak tersedia *security* yang sifatnya *end-to-end*, sehingga memang seluruh pihak yang terlibat pada *m-commerce* harus percaya sepenuhnya kepada operator GSM.

2. SIM *Tool Kit*

   Memungkinkan operator atau *provider* lain untuk membuat suatu aplikasi khusus yang tertanam di SIM *Card*. Aplikasi tersebut bertugas untuk mengirimkan, menerima, serta mengartikan sebuah SMS atau USSD. STK memungkinkan aplikasi pengirim (aplikasi yang terdapat pada SIM *Card*) untuk mengirimkan pesan yang sudah terproteksi kepada aplikasi penerima (pada *server payment operator*). Mekanisme *security* yang mungkin untuk diterapkan adalah:

   a. Authentikasi
   b. Integritas Informasi Transaksi
   c. Integritas urutan transaksi dan deteksi pengulangan transaksi
   d. Bukti Penerimaan dan eksekusi transaksi
   e. Kerahasiaan Informasi

2.4 Paypal

2.4.1 Definisi Paypal

Paypal adalah salah satu alat pembayaran (*payment procesors*) menggunakan internet yang terbanyak digunakan di dunia dan teraman. Pengguna internet dapat membeli barang di *ebay*, lisensi *software original*, keanggotaan situs, urusan bisnis, mengirim dan menerima donasi/sumbangan, mengirim uang ke pengguna *paypal* lain di seluruh dunia dan banyak fungsi lainnya dengan mudah dan otomatis menggunakan internet atau *mobile*, *paypal* mengatasi kekurangan dalam pengiriman uang tradisional seperti cek atau *money order* yang prosesnya dapat memakan waktu *paypal* seperti rekening bank, pertama anda membuat *account*, lalu dana akan ditarik langsung dari kartu kredit anda ke *account* tersebut atau dengan dana dari transferan *account paypal* orang lain ke Saldo / *Balance Paypal* anda, dan anda sudah dapat menggunakan *account paypal* untuk bertransaksi.

Februari 2010 *paypal* sudah menerima 190 negara dan 23 mata uang, pengguna *paypal* di Indonesia masih harus menggunakan hitungan US dollar karena rupiah belum ada di *paypal*, dana yang ditarik langsung dari kartu kredit akan dikonversi ke US dollar otomatis.

ebay adalah tempat lelang/jual-beli/pasar *online* terbesar di dunia, dengan pengguna lebih dari 200 juta orang di seluruh dunia (data juni 2006), anda dapat mencari dan membeli apapun, mulai dari barang elektronik, kendaraan, hingga barang-barang langka dari seluruh pengguna *ebay* di dunia, cara pembayaran

utama di *ebay* adalah dengan *paypal* karena lebih aman dan otomatis, meskipun ada penjual yang menerima pembayaran dengan cara lain.

2.4.2 Sejarah Paypal

Paypal 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*.

2.4.3 Keuntungan Paypal

Pembeli / *Buyer* yang merasa dirugikan atau tidak sesuai deskripsi yang diberitahukan bisa meminta uang kembali kepada Penjual / *Seller* 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

Dari sini bisa diambil kesimpulan 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.

2.5 Object Oriented Programming

Object Oriented Programming atau Pemrograman Berorientasi Objek adalah konsep pemrograman yang difokuskan pada penciptaan kelas yang merupakan abstraksi/*blueprint/prototype* dari suatu objek. Kelas ini harus mengandung sifat (data) dan tingkah laku (*method*) Umum yang dimiliki oleh objek-objek yang kelak akan dibuat (diinstansiasi). Data dan *method* merupakan anggota (member) dari suatu kelas.

2.5.1 Konsep Pemrograman Berorientasi Objek

Hampir selama sepuluh tahun belakangan ini telah diperkenalkan paradigma pemrograman baru yang dikenal dengan konsep Pemrograman Berorientasi Objek (PBO) atau *Object Oriented Programming* (OOP). Konsep ini membagi program menjadi objek-objek yang saling berinteraksi satu sama lain.

Objek adalah kesatuan entitas yang memiliki sifat dan tingkah laku. Objek adalah benda, baik benda yang berwujud nyata maupun benda yang tidak nyata (konsep).

Konsep-konsep pemrogram berorientasi objek dalam Java secara umum sama dengan yang digunakan oleh bahasa-bahasa lain. Konsep yang akan dibahas terdapat dalam bahasa selain Java. Namun, terkadang terdapat perbedaan-perbedaan kecil antara penerapan konsep-konsep tersebut dalam masing-masing bahasa. Perbedaan-perbedaan ini juga akan dijelaskan seiring penjelasan masing-masing konsep.

1. **Objek**

   Objek adalah kesatuan entitas (benda), baik yang berwujud nyata ataupun hanya suatu sistem atau konsep yang memiliki sifat karakteristik dan fungsi.

2. **Kelas**

   Kelas adalah pemodelan dari objek yang berisi informasi (aturan) tentang sifat karakteristik (data) dan tingkah laku (*method*) yang dimiliki oleh objek tersebut.

3. ***Instance***

   Dalam istilah pemrograman, objek merupakan *instans* (perwujudan) dari suatu kelas. Instansiasi adalah perwujudan kelas menjadi objek.

4. **Data**

   Data adalah sifat karakteristik yang dimiliki oleh objek. Dalam beberapa referensi, data dianalogikan juga sebagai *variable*, *property*, atau *atribut*.

5. ***Method***

   Method adalah prosedur atau fungsi yang dimiliki oleh suatu objek. Dalam beberapa referensi, *method* dianalogikan juga dengan istilah prosedur, fungsi, dan *behavior*.

6. **Hierarki Kelas**

   Suatu kelas dapat dibuat sebagai kelas yang benar-benar baru atau dapat dibentuk dari kelas yang sudah ada. Baru di sini dalam pengertian, semua data dan *method* yang dimiliki oleh kelas tersebut dideskripsikan dan dan dibuat sendiri, sedangkan bila membentuk dari kelas yang sudah ada bermakna anda menggunakan data dan *method* yang telah dideskripsikan dalam suatu kelas

acuan (*foundamen*). Proses pembentukan kelas baru dari kelas yang sudah ada disebut sebagai pewarisan (*inheritanace).* Kelas yang menjadi bisnis bagi pembentukan kelas lainya disebut sebagai Kelas Super, sedangkan kelas yang terbentuk dinamakan Kelas Sub.

Proses pewarisan ini akan menghasilkan suatu hirarki kelas. Kelas yang terletak pada puncak hierarki disebut sebagai Kelas Abstrak. Sifat dari kelas abstrak adalah hanya berisi deskripsi data dan *method* yang sangat umum, sehingga tidak mungkin diimplementasikan secara terperinci. Oleh sebab itu, kelas abstrak ini tidak dapat langsung diinstansiasi menjadi objek.

2.5.2 Tiga Pilar Object Oriented Programming

1. ***Inheritance***

Inheritance adalah proses pewarisan data dan *method* dari suatu kelas kepada kelas yang lain. Pewarisan ini bersifat menyeluruh, sehingga semua data dan *method* yang dimiliki oleh kelas asalnya akan diturunkan kepada kelas baru.

2. ***Encapsulation***

Encapsulation adalah bagaimana anda “membungkus” data dan method yang menyusunn kelas sehingga kelas dapat dipandang sebagai modul. Termasuk di sini adalah menentukan tingkat aksesibilitas dari masing-masing data dan *method* serta menyembunyikan detail implementasinya. Dalam pemrograman, menyembunyikan detail ini dikenal dengan istilah *Information Hiding*.

3. ***Polymorphism***

Polymorphism bermakna sesuatu yang dimiliki banyak bentuk. Dalam pemrograman, istilah polimorfi dapat diartikan sebagai modul yang memiliki nama sama, namun memiliki *behavior* (tingkah laku yang berbeda) sehingga *listing code* implementasinya berbeda.

2.5.3 Keuntungan Object Oriented Programming

Ada enam keuntungan yang diperoleh bila menggunakan OOP, diantaranya sebagai berikut:

1. Alami (*Natural*)
2. Dapat Diandalkan (*Reliable*)
3. Dapat Digunakan Kembali(*Reuseable*)
4. Mudah Untuk Di-Maintain (*Maintainable*)
5. Datap Diperluas (*Ekstendable*)
6. Efisiensi Waktu

Berikut ini adalah contoh beberapa bahasa pemrograman yang menggunakan konsep OOP, diantaranya:

7. C++
8. Visual C++
9. Delphi
10. Visual Basic
11. Java
12. PHP 5

2.5.3.2 Sejarah Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Sejarah java berawal pada tahun 1991 ketika perusahaan Sun Micro System memulai *Green Project,* yakni projek penelitian untuk membuat bahasa yang akan digunakan *pada chip-chip embedded* untuk *device intelligent consumer electronic*. Bahasa tersebut haruslah bersifat *multiplatform,* tidak tergantung kepada vendor yang memanufaktur *chip* tersebut.

Dalam penelitiannya, *Projek Green* berhasil membuat *prototype* semacam PDA (*Personal Data Assistance*) yang dapat berkomunikasi antara satu dengan yang lain dan diberi nama Star 7. Ide berawal untuk membuat sistem operasi bagi Star 7 berbasis C dan C++, setelah berjalan beberapa lama, James Gosling, salah satu anggota team merasa kurang puas dengan beberapa karakteristik dari kedua bahasa tersebut berusaha mengembangkan bahasa lain. Bahasa tersebut kemudian dinamakan Oak, Belakangan Oak beralih nama menjadi Java.

Karena pada awalnya ditujukan untuk pemrograman *device* kecil, Java memiliki karakteristik berukuran kecil, efisien, dan portable untuk berbagai *hardware*. *Project Green* sempat terancam terhenti karena dalam perkembangannya, *device* ini belum memiliki pasar seperti yang diramalkan semula. Selanjutnya Java diarahkan untuk pemrograman internet. Secara kebetulan, fitur-fitur Java yang telah disebutkan sebelumnya sangat sesuai bagi pengembangan internet sehingga dalam beberapa tahun belakangan ini Java telah menjadi primadona untuk pemrograman yang berbasis internet.

2.5.3.3 Keunggulan Java

Java memiliki beberapa keunggulan bila dibandingkan dengan bahasa pemrograman lainnya. Ada beberapa aspek yang akan dibahsa, diantaranya :

1. **Bersifat Sederhana dan Relatif Mudah**

   Java dimodelkan sebagian dari bahasa C++, namun dengan memperbaiki beberapa karakteristik C++, seperti mengurangi kompleksitas beberapa fitur, penambahan fungsionalitas, serta penghilangan beberapa aspek pemicu ketidakstabilan sistem pada C++.

2. **Berorientasi Pada Objek (*Object Oriented*)**

   Java adalah pemrograman yang berorientasi objek (OOP), bukan seperti Pascal, Basic, C yang berbasis prosedural. Dalam memecahkan masalah, java membagi program menjadi objek-objek, kemudian memodelkan sifat dan tingkah laku masing-masing. Selanjutnya, Java menentukan dan mengatur interaksi antara objek yang satu dengan lainnya.

3. **Bersifat Terdistribusi**

   Pada dekade perkembangan PC (*Personal Computer*), komputer hanya bersifat sebagai *workstation* tunggal, tidak terhubung satu sama lain. Saat ini, sistem komputerisaasi cenderung terdistribusi, mulai dari *workstation client*, *email server*, *database server*, *web server*, *proxy server*, dan sebagaianya.

4. **Bersifat *Multiplatform***

   Dewasa ini kita mengenal banyak platform *Operating System*, mulai dari Windows, Apple, berbagai varian UNIX dan Linux, dan sebagainya, Pada umumnya, program yang dibuat dan dikompile disuatu platform hanya bisa

dijalankan diplatform tersebut. Java bersifat *multiplatform*, yakni dapat di-terjemahkan oleh Java Interpreter pada berbagai sistem operasi.

5. **Bersifat *MultiThread***

   Thread adalah proses yang dapat dikerjakan oleh program dalam suatu waktu. Java bersifat *Multithreaded*, artinya dapat mengerjakan beberapa proses dalam waktu yang hampir bersamaan.

6. ***Class Library* Yang Lengkap**

   Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.

2.6 Android

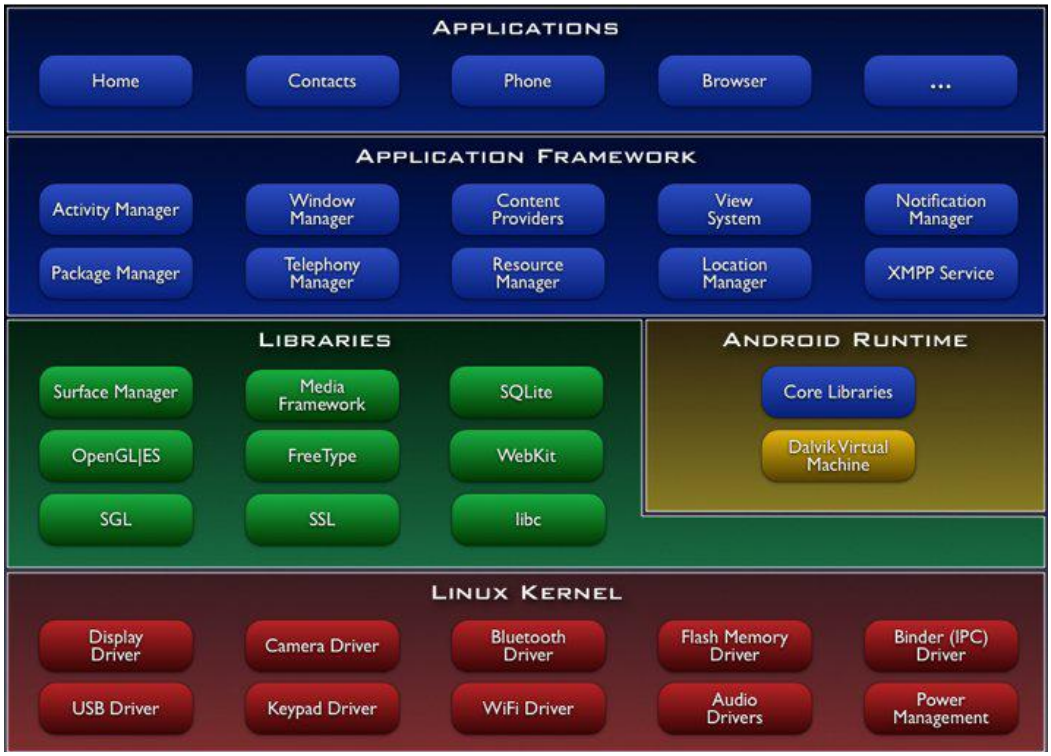
Android merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi, *middleware* dan aplikasi inti yang di *release* oleh GOOGLE. Aplikasi pada platform ini dikembangkan dalam bahasa pemrograman java. Android sendiri memiliki banyak fitur diantaranya adalah:

1. **Application Framework.** *Programmer* dapat menggunakan beberapa fungsi yang telah disediakan.
2. **Dalvik virtual machine**. Tiap aplikasi dalam Android memiliki *instance virtual machine* yang dapat bekerja secara efisien dalam lingkungan memori yang terbatas.
3. **Integrated browser.** *Web browser* berbasis WebKit *engine* terdapat pada browser *default* Android atau pun dapat diintegrasikan dengan aplikasi lain.
4. **Optimized graphics**. *Library* grafis 2D yang kaya dan 3D berbasis OpenGL ES 1.0 yang mendukung akselerasi *hardware*.
5. **SQLite**. Basis data relasional yang ringan namun sangat *powerful.*
6. **Media Support**. Mendukung berbagai format audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
7. **GSM Telephony**. Mendukung fungsi komunikasi GSM.
8. **Bluetooth, EDGE, 3G** dan **WiFi**. Mendukung komunikasi pada jaringan (tergantung *hardware*).
9. **Kamera**, **GPS**, **kompas** dan ***accelerometer***. Mendukung berbagai fitur yang disediakan oleh *hardware*.

10. **Tools Developer** yang lengkap. Termasuk *device emulator, tools* untuk *debugging, profiling* memori dan performa, *plugin* untuk Eclipse IDE.

2.6.1 Arsitektur Android

Android terdiri dari beberapa *stack software* yang terdiri dari: *Applications, Application Framework, Libraries, Android Runtime* dan *Kernel Linux.* Arsitektur lengkap platform ini dapat dilihat pada Gambar II.3.



Gambar II.3 Arsitektur Platform Android

1. ***Aplication***

Application merupakan program yang langsung berhubungan dengan *user*. Baik program yang merupakan bawaan dari Android sendiri maupun program yang dibuat oleh *developer* menggunakan bahasa pemrograman java.

Contoh program bawaan dari platform Android sendiri adalah *email client*, program SMS, *calendar*, *maps*, *web browser*, *contact* dan sebagainya.

2. ***Aplication Framework***

Lapisan ini berisi sekumpulan API yang dapat digunakan oleh *programmer* maupun *core application* dari Android. Lapisan ini dirancang untuk memudahkan penggunaan komponen dari Android sendiri. Aplikasi manapun dalam Android dapat berbagi fungsi sehingga aplikasi lain dapat memanfaatkannya.

3. ***Libraries***

Android mendukung beberapa *library* C/C++ yang digunakan pada berbagai komponen Android. Kemampuan ini dapat diakses oleh *developer* melalui Android *application framework*. Beberapa *library* diantaranya adalah:

1. *System* C *library*. Implementasi *library* C standar (libc).
2. *Media Libraries*. Mendukung berbagai format multimedia (termasuk MPEG4,H.264, MP3, AAC, AMR, JPG, PNG).
3. *Surface Manager*. Mengatur akses ke subsistem *display*.
4. LibWebCore. *Engine web browser* modern.
5. SGL. *Engine* grafis 2D.
6. 3D *Library*. Implementasi OpenGL ES 1.0 yang mendukung akselerasi hardware.
7. FreeType. *Rendering* untuk *bitmap* dan *vector font*.
8. SQLite. Basis data relasional yang kecil namun sangat ampuh.

4. ***Android Runtime***

Tiap aplikasi pada Android memiliki proses-nya masing-masing. Tiap aplikasi tersebut memiliki *instans* dari Dalvik virtual machine (VM). Dalvik virtual machine dirancang agar suatu *device* dapat menjalankan beberapa VM secara efisien. Dalvik VM mengeksekusi file dengan format Dalvik Executable format (.dex) yang dirancang untuk meminimalkan *memory footprint* . Dalvik VM berbasis *register*, dan dapat menjalankan kelas-kelas yang dikompilasi dengan bahasa pemrograman java dan ditransformasikan menjadi format .dex. Dalvik VM sendiri bergantung pada Kernel Linux untuk fungsi dasarnya, seperti *threading* dan manajemen memori secara *low-level*.

5. **Linux Kernel**

Android menggunakan Kernel Linux versi 2.6 sebagai sistem utama. Fungsi kernel yang digunakan antara lain untuk keamanan, manajemen memori, manajemen proses, manajemen jaringan dan *driver model*. Kernel juga berfungsi sebagai *layer* abstrak antara *hardware* dan lapisan lainnya pada *software stack*.

2.6.2 Siklus Hidup Aplikasi Android

Pada banyak kasus, tiap aplikasi pada Android masing-masing memiliki Linux proses. Proses ini diciptakan untuk aplikasi tersebut pada saat kode program tersebut akan dieksekusi, akan terus berjalan sampai tidak dibutuhkan lagi dan sistem memerlukan memori untuk aplikasi lain. Konsep dasar program Android adalah suatu proses pada aplikasi tidak langsung di atur oleh aplikasi tersebut. Proses tersebut ditentukan oleh sistem melalui kombinasi : informasi

aplikasi bagi sistem, seberapa pentingnya aplikasi tersebut bagi *user* dan ketersediaan memori. Contoh kasus *life-cycle bug* adalah *IntentReceiver* yang menciptakan *thread* pada saat menerima *intent* pada *method* onReceiveIntent() dan selesai dari fungsi tersebut. Setelah fungsi tersebut selesai, sistem menganggap bahwa *IntentReceiver* tersebut menjadi tidak aktif sehingga ia memiliki proses yang tidak dibutuhkan (kecuali terdapat komponen aplikasi lain di dalamnya). Oleh karena itu, sistem dapat menghapus proses tersebut kapanpun apabila dibutuhkan. Hal tersebut akan menghapus semua *thread* yang berjalan pada proses tersebut. Solusi masalah ini adalah memakai *Server* dari *IntentReceiver*, sehingga sistem mengetahui bahwa ada suatu yang aktif dalam proses tersebut.

Sebagai *programmer*, kita harus mengetahui bagaimana cara yang tepat untuk merancang aplikasi. Kesalahan pada perancangan akan membuat suatu aplikasi akan dihentikan oleh sistem pada saat yang tidak diinginkan. Untuk menentukan proses mana yang akan dihapus pada situasi *low memory*, Android memiliki "*importance hierarchy*" berdasarkan komponen dan *state* yang sedang berjalan. Urutan berdasarkan kepentingan tersebut adalah:

1. ***Foreground process***

   Adalah suatu proses yang dibutuhkan untuk berinteraksi dengan *user* saat ini. Berbagai komponen suatu aplikasi dapat mengakibatkan proses di dalamnya menjadi *foreground proses*. Suatu proses dikatakan *foreground* apabila memenuhi kondisi sebagai berikut:

1. Menjalankan aktifitas pada *top screen* yang sedang berinteraksi dengan *user* (setelah *method* onResume() dipanggil).
2. Mempunyai *IntentReceiver* yang sedang berjalan (method IntentReceiver.onReceiveIntent() sedang dieksekusi).
3. Memilik server yang fungsi *callback*-nya (Server.onCreate(), Server.onStart() atau Server.onDestroy()) sedang dieksekusi.

Hanya akan terdapat beberapa proses seperti itu dalam sistem, proses ini hanya akan dihapus apabila sistem tidak memiliki pilihan lain. Contohnya apabila memori tinggal sedikit sehingga proses *foreground* tidak bisa lagi berjalan. Hal ini diperlukan untuk menjaga *user interface* tetap interaktif dengan *user*.

2. ***Visible process***

Adalah proses yang memegang aktifitas yang dapat dilihat *user* pada layar tetapi tidak *foreground* (pada saat *method* onPause() dipanggil). Hal ini dapat terjadi, sebagai contoh *activity foreground* muncul dengan dialog sehingga *activity* sebelumnya terlihat di belakangnya. Proses ini sangat penting dan tidak akan di hapus kecuali untuk menjaga aplikasi *foreground* tetap berjalan.

3. ***Server process***

Adalah proses yang memegang Server yang dimulai dengan method StartServer(). Meskipun proses ini secara langsung tidak dapat dilihat oleh *user*, biasanya proses ini menjalankan perintah dari *user* (seperti memutar lagu, men-*download* data dari jaringan dan lain-lain). Sistem akan tetap

membiarkan proses ini berjalan selama proses *foreground* dan *visible* masih bisa berjalan.

4. ***Background process***

   Adalah proses yang memegang aktifitas yang saat ini tidak dapat dilihat oleh *user* (method onStop() sedang dipanggil). Proses ini tidak secara langsung mempengaruhi *user*. Apabila *activity life cycle* telah dipenuhi dengan benar, sistem dapat menghapus proses ini sewaktu-waktu untuk memberikan memori tambahan. Biasanya banyak ditemukan proses semacam ini, sehingga proses tersebut disimpan dalam LRU (*Least Recently Used*) untuk menjaga agar proses yang baru saja dilihat oleh *user* akan dihapus paling akhir pada kondisi *low memory*.

5. ***Empty Process***

   Adalah proses yang tidak memegang suatu komponen yang sedang aktif. Alasan mengapa membiarkan proses semacam ini adalah untuk melakukan *cache* sehingga proses *startup* menjadi lebih cepat apabila aplikasi tersebut akan dijalankan kembali. Sistem seringkali menghapus proses semacam ini untuk mencapai keseimbangan dari *system resource* antara cached processes dan kernel *caches*.

2.7 UML

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. *Unified Modeling Language* (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat (*tool*) desain berorientasi objek dari IBM.

UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

2.7.1 Sejarah UML

Sampai era tahun 1990 puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi booch, metodologi coad, metodologi OOSE, metodologi OMT, metodologi shlaer-mellor, metodologi wirfs-brock, dsb. Masa itu terkenal dengan masa perang metodologi dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan kelompok/perusahaan lain yang menggunakan metodologi yang berlainan.

Dimulai pada bulan Oktober 1994 Booch, Rumbaugh dan Jacobson, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan mempelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 direlease draft pertama dari UML (versi 0.8) [11]. Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh Object Management Group.

2.7.2 Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek, yaitu:

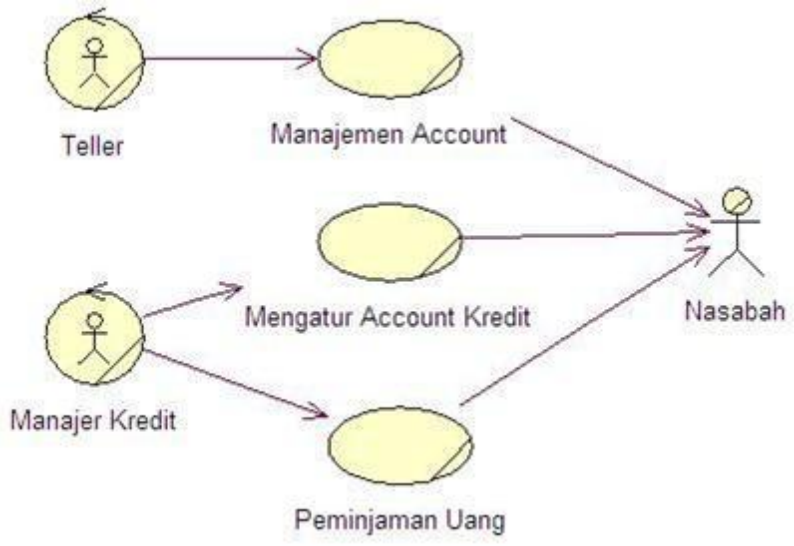
1. Use Case Diagram untuk memodelkan proses bisnis.
2. Conceptual Diagram untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
3. Sequence Diagram untuk memodelkan pengiriman pesan (*message*) antar *objects*.
4. Collaboration Diagram untuk memodelkan interaksi antar *objects*.

5. State Diagram untuk memodelkan perilaku *objects* di dalam sistem.
6. Activity Diagram untuk memodelkan perilaku Use Cases dan *objects* di dalam *system*.
7. Class Diagram untuk memodelkan struktur kelas.
8. Object Diagram untuk memodelkan struktur *object*.
9. Component Diagram untuk memodelkan komponen *object*.
10. Deployment Diagram untuk memodelkan distribusi aplikasi.

1. **Use Case Diagram**

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case*

yang satu merupakan spesialisasi dari yang lain. Penggambaran use case diagram dapat di lihat dalam gambar II.4.

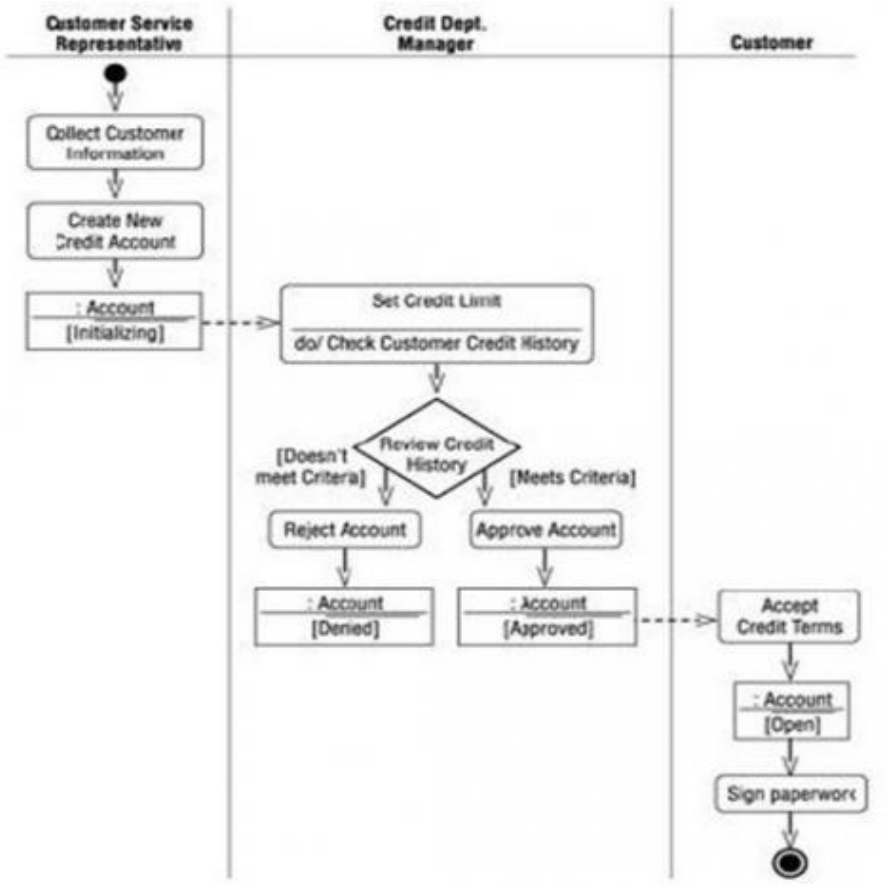


Gambar II.4 Contoh Use Case Diagram

2. **Activity Diagram**

Activity diagrams menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity diagram merupakan state diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu activity diagram tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktifitas dari level atas secara umum.

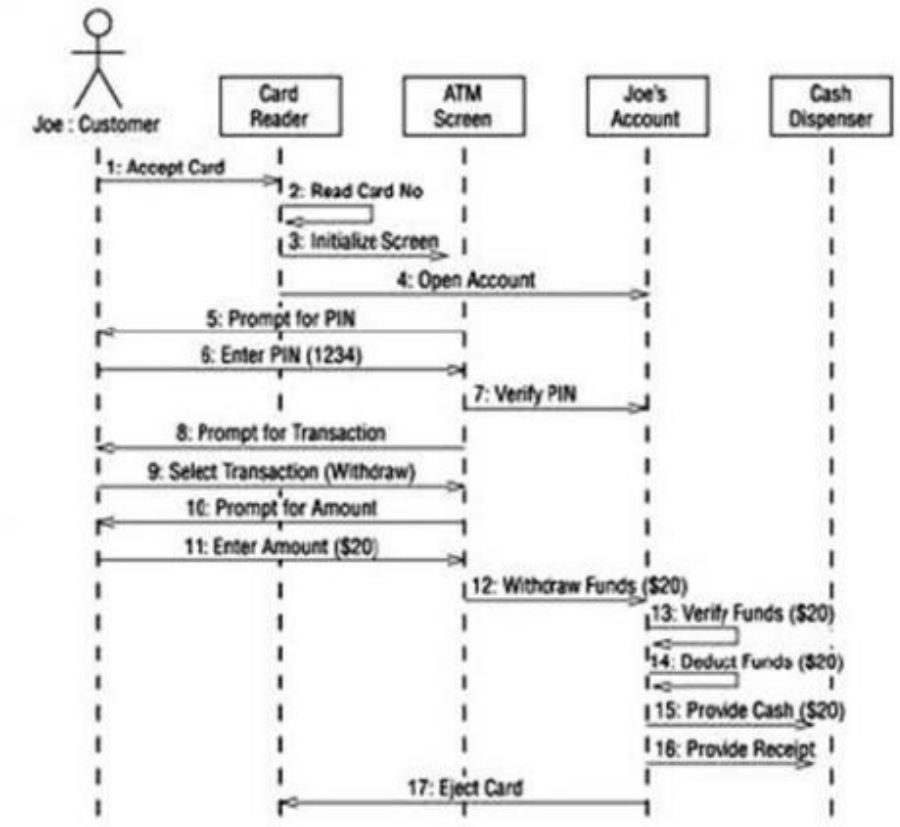
Sebuah aktifitas dapat direalisasikan oleh satu *use case* atau lebih. Aktifitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktifitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. Activity diagram dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktifitas tertentu. Contoh Activity diagram dapat di lihat pada gambar II.5.



Gambar II.5 Contoh Activity Diagram

3. **Sequence Diagram**

Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case: interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi. Contoh sequence diagram dapat di lihat pada gambar II.6.

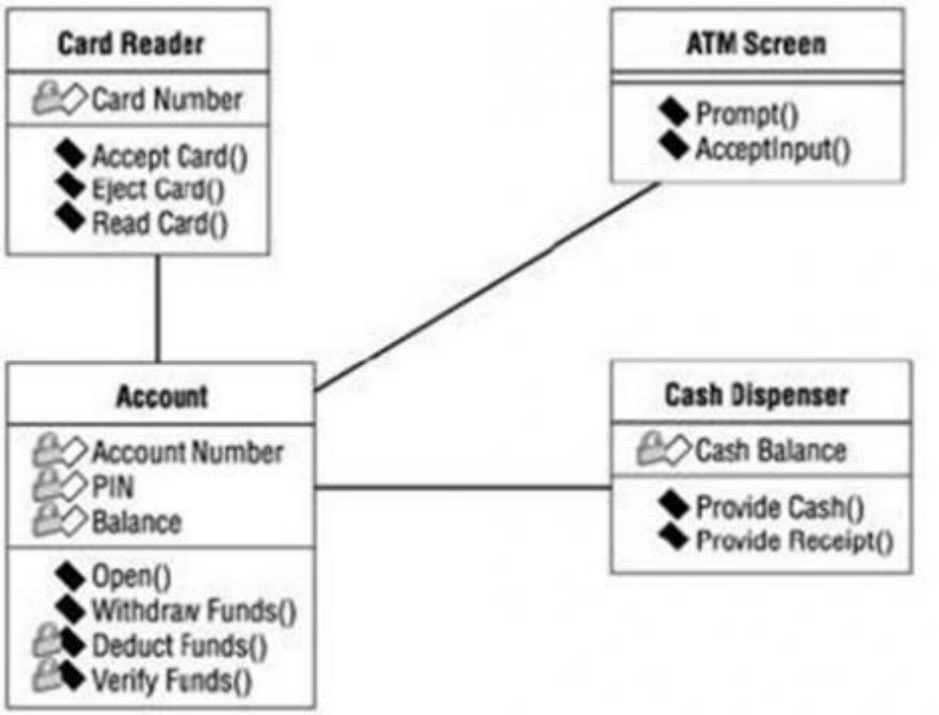


Gambar II.6 Contoh Sequence Diagram

4. **Class Diagram**

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk

mencapai suatu tujuan. Penggambaran class diagram dapat di lihat dalam gambar II.7.



Gambar II.7 Contoh Class Diagram