BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Hotel Golden Flower Bandung

Hotel Golden Flower adalah salah satu jenis akomodasi yang mempergunakan sebagai atau keseluruhan bagian untuk jasa pelayanan penginapan, penyedia makanan dan minuman serta jasa lainnya bagi masyarakat umum yang dikelola secara komersil terutama untuk wisatwan. Hotel Golden Flower Bandung beralamat di Jl Asia-Afrika No 15-17 Bandung 40111, Telp/Fax (022) 4239999 / (022) 4209889, e-mail: bandung@golden-flower.co.id.

2.1.1 Sejarah Hotel Golden Flower Bandung

Hotel Golden Flower Bandung merupakan salah satu hotel dari KAGUM Hotels. KAGUM Hotels adalah salah satu cabang dari KAGUM Group yang didirikan oleh Bapak Henry Husada. Hotel Golden Flower Bandung merupakan hotel berbintang 4(****).

Hotel Golden Flower Bandung didirikan pada tanggal 1 Juli 2009 untuk menunjang kepariwisataan di Kota Bandung yang dikelola oleh KAGUM Hotels, khususnya dalam penyediaan akomodasi bagi wisatawan. Hotel Golden Flower Bandung yang berlokasi di Jalan Asia-Afrika No 15-17 Bandung 40111. Selain KAGUM Hotels, KAGUM Group juga memiliki beberapa perusahaan yang bernaung dibawahnya. Antara lain sebagai berikut : KAGUM Fashion yang memiliki 20 *outlet* yang tersebar di seluruh kota Bandung. KAGUM Kuliner. KAGUM Trans yang menyediakan pelayanan jasa sewa kendaraan. KAGUM *Finance*, KAGUM *Land*, dan KAGUM *Property*. KAGUM *Hotels* sendiri telah memiliki 35 *Hotels* di seluruh Indonesia.

Hotel Golden Flower Bandung merupakan hotel bintang 4(****) yang memiliki konsep design modern. Hotel ini memiliki exterior dan interior yang sangat baik. Fasilitas yang dimiliki hotel ini sangat memadai dan canggih. Selain itu, Hotel Golden Flower Bandung juga berlokasi di Jl. Asia-Afrika yang terkenal sebagai jalan historis yang memiliki sejarah panjang. Terletak di tengah Kota Bandung. Hotel Golden Flower sangat strategis untuk kebutuhan bisnis maupun

liburan dikarenakan berjarak tempuh hanya 10 menit dari Bandara Husein Sastranegara. Hotel ini juga dekat pusat belanja terutama produk-produk *fashion*, Pasar Baru yang merupakan pasar garmen terbesar dan terlengkap di Bandung dan dapat di tempuh dalam waktu hanya 5 menit..

Hotel Golden Flower Bandung memiliki 193 kamar dengan berbagai tipe yang dapat dipilih sesuai kebutuhan para pelanggan hotel. Kamar-kamar ini memiliki fasilitas, antara lain mini bar, sarana pembuat teh dan kopi, *safety box*, perlengkapan mandi, akses internet, televisi, telepon, dan lain sebagainnya. Selain itu Hotel Golden Flower Bandung memiliki fasilitas bisnis yang ditawarkan untuk pelanggan hotel atau tamu hotel, antara lain 22 ruang *meeting room* dengan ruangan terbesar yang dapat menampung sampai dengan 600pax, fasilitas rapat dapat memadai ditunjang perlengkapan *audiovisual*, *business center* yang memperlengkapi kebutuhan rapat, serta layanan gratis akses internet di hampir semua area hotel.

Karena lokasinya berada tepat di Jalan Asia-Afrika, Hotel Golden Flower Bandung juga menawarkan fasilitas olahraga lainnya seperti kolam renang dan *fitness center*. Dari segi hiburan, para pelanggan hotel dapat menikmati fasiitas spa dengan pelayanan pijat refleksi dan piat seluruh badan. Fasilitas spa pun tersedia dengan dekorasi desain yang unik, di dalam ruangan maupun di luar ruangan.

Hotel Golden Flower Bandung juga menyediakan restoran untuk melengkapi kebutuhan para pelanggan hotel akan makanan dan minuman. Lotus adalah restoran yang akan memanjakan lidah para pelanggan hotel dengan menghidangkan berbagai jenis makanan Asia yang disajikan oleh tangan ahli seorang *executive chef*. Sementara untuk tempat bersantai sambil menikmati pemandangan Kota Bandung, Hotel Golden Flower menghadirkan *cake shop* yang akan menyajikan makanan dan minuman ringan.

2.1.2 Profil Singkat Golden Flower Bandung

Informasi Umum:

1. Tipe : Business M.I.C.E Hotel
2. Style : *Modern Event Hotel*

3. Manajemen :PT. Mandiri Anugerah Jaya Utama (KAGUM Hotels) subsidiary KAGUM Group
4. Alamat : Jl. Asia-Afrika No.15-17 Bandung 40111
5. Lokasi : Pusat Kota Bandung
6. Telepon : (002) 423 9999
7. Fax : (002) 420 9889
8. Website : www.golden-flower.co.id/bandung
9. Jumlah Lantai : 5 Lantai
10. Jumlah Kamar : 193 Kamar
11. Fasilitas Parkir Mobil : *Private Basment Parking*
12. Jarak Hotel-Bandara : 10 menit

2.1.3 Logo Hotel Golden Flower Bandung

Logo atau Lambang adalah suatu identitas dari sebuah perusahaan, hampir semua perusahaan memiliki logo untuk mempertegas keberadaannya, begitu juga Hotel Golden Flower Bandung yang memiliki Logo sebagai berikut:

Sumber Gambar : www.golden-flower.co.id/bandung

Gambar 2.1 Logo Hotel Golden Flower

2.1.4. Fasilitas-fasilitas yang Terdapat di Hotel Golden Flower Bandung

Adapun beberapa fasilitas yang di sediakan oleh Hotel Golden Flower Bandung diantaranya:

2.1.4.1. Fasilitas Kamar

Hotel berlantai 5 ini memiiki kamar dengan berbagai *type* seperti berikut:

Tabel 2.1 Tabel Fasilitas Kamar

TYPE	SIZE	TWIN	KING	TOTAL ROOM
SUPERIOR	27 m²	101	50	151
DELUXE	32 m²	–	18	18
GRAND DELUXE	39 m²	–	13	13
EXECUTIVE	48 m²	–	6	6
SUITE	60 m²	–	5	5

Berikut adalah fasilitas-fasilitas kamar sesuai pada tabel 2.1

1. Tempat tidur nyaman dengan menggunakan impor mewah
2. Air dingin dan panas dengan perlatan dan produk mandi yang lengkap
3. AC individu
4. Akses internet berkecepatan tinggi
5. LCD TV *Flat* dengan 52 *channels*
6. Safe deposit box
7. Tempat membuat kopi dan teh
8. Koran harian
9. 24 jam layanan kamar
10. Ruang merokok dan non-merokok

2.1.4.2. Fasilitas *Meeting Room*

Hotel Golden Flower Bandung yang berorientasi untuk keperluan meeting atau bisnis lainnya sehingga hotel ini memiliki 22 *venue* untuk keperluan pertemuan, *event*, seperti *wedding*, ulang tahun dan lain sebagainya. Adapun rinciannya sebagai berikut :

Tabel 2.2 Fasilitas *Meeting Room*

VENUE	SIZE	CAPACITY					
		STANDING	THEATER	CLASS ROOM	ROUND TABLE	DOUBLE U-SHAPE	U-SHAPE
TULIP	21,7 x 9,7	150	150	90	100	60	42
TULIP 1	9,7 x 8	–	40	30	30	–	18
TULIP 2	9,7 x 6,8	–	40	30	20	–	18
TULIP 3	9,7 x 6,7	–	40	30	20	–	18
HELICONIA	7,3 x 6	–	20	10	10	–	10
ORCHID	14,2 x 7,4	–	50	30	50	40	27

VENUE	SIZE	CAPACITY					
		STANDING	THEATER	CLASS ROOM	ROUND TABLE	DOUBLE U-SHAPE	U-SHAPE
PEONY	14,3 x 6,3	–	50	30	30	–	34
ASTER	25,5 x 5,4	100	80	50	60	–	45
ASTER 1	11 x 5,4	–	30	15	20	–	15
ASTER 2	7,3 x 5,4	–	30	15	20	–	12
ASTER 3	7 x 5,4	–	30	28	20	–	15
GERBERA	14,3 x 7,4	–	70	54	60	50	30
GERBERA 1	8,6 x 7,4	–	50	24	40	27	18
GERBERA 2	7,4 x 5,6	–	16	10	20	10	9
ACACIA	12 x 7,7	–	50	30	50	40	27
ANTHURIUM	12 x 6,3	–	50	30	30	–	34
LILY	7 x 5,4	–	30	12	–	–	9
MENTHA	7,4 x 4,5	–	30	20	20	–	15
NIGELLA	7,4 x 4,5	–	20	12	20	–	15
ZINEA	7,4 x 4,5	–	20	12	20	–	15
PROTEA	7,4 x 4,5	–	20	12	20	–	15

Berikut adalah fasilitas-fasilitas yang didapatkan ;

1. RESIDENTIAL
   - *HALF DAY : 1x coffee break, 1 x lunch/dinner dan room*
   - *FULL DAY : 2x coffee break, 1 x lunch/dinner dan room*
   - *FULL BOARD : 2x coffee break, 1x lunch, 1x dinner dan room*
2. NON RESIDENTIAL
   - *HALF DAY : 1x coffee break, 1 x lunch/dinner*
   - *FULL DAY : 2x coffee break, 1 x lunch/dinner*
   - *FULL BOARD : 2x coffee break, 1x lunch, 1x dinner*

2.1.4.3. Fasilitas Penunjang Lainnya

Selain Fasilitas *meting room*, *venue* dan kamar yang ditawarkan oleh Hotel Golden Flower Bandung, di hotel ini memiliki fasilitas penunjang lainnya yang dapat membuat pengunjung menjadi lebih nyaman jika melakukan aktivitas meeting atau akomodasi di Hotel Golden Flower Bandung, diantaranya yaitu :

1. Lotus Garden *Restaurant*

   Restaurant dengan menu yang sangat beragam. Memiliki masakkan Barat, Asia, dan Indonesia.

   Kapasitas tempat duduk :220 pax

Jam Buka : 06 AM - 10 PM
06 AM – 10.30 PM (*Weekend*)

2. *Cake Shop*

   Menjual *cake* yang dibuat oleh chief-chief *pastry* terbaik di Bandung.

   Jam Buka : 09 AM – 10 PM

3. *Wine Cellar – The Winery*

   Tersedia untuk pelanggan hotel yang memesan *wine* dengan pilihan banyak sekali *wine* seperti *White Wines, Red Wines, Sparking Wines, Fortified Wines.*

4. IRISH *Lounge*

   Tempat bersantai yang sangat nyaman dengan suasana yang menyejukan.

   Kapasitas tempat duduk : 22 pax (*sofa*) 21 pax (*tables*)

   Jam Buka : 10 AM – 11 PM

2.1.4.4. *Ballrooms*

Hotel Golden Flower Ballroom

Ukuran : 27,5 x 20,2 m

Kapasitas : 800 pax
500 pax (*Theater Style*)
250 pax (*Round Table Style*)

2.1.4.5. *Wellness*

1. *Hotel Golden Flower Wellness*
2. *Golden Flower Swimming Pool*
3. *Golden Flower Sauna*
4. *Fitness Center*

Fasilitas Fitness yang lengkap dan berdekatan dengan kolam renang dan *Spa*.

Jam Buka : 08 AM – 21 PM

Fasilitas : *Personal Training, Free Had & Bath Towel,* Loker Pribadi

Peralatan : *Crosstrainer, Static Bike, Multi Bench,*

Double Set,Treadmill

5. Purnama Agung *Spa*

Massage tradisional dan aroma terapi dengan minyak spesial yang di produksi oleh Purnama Agung *Spa*. Terletak di lantai teratas gedung.

Jam Buka : 08 AM – 23 PM

Fasilitas : *Full body treatment, Refreshing treament, Acupressur Aromatheraphy Massage*.

2.1.4.6. *Entertainment*

Hiburan dengan tujuan untuk menjadi yang terbaik di sediakan oleh Hotel Golden Flower Bandung.

a. Batik-Ku & Mr. Bean (*Fashion Shops*)

Fashion Shosp yang menjual batik spesial dan berbagai macam oleh-oleh.

b. *Legend Club* & *Karaoke*

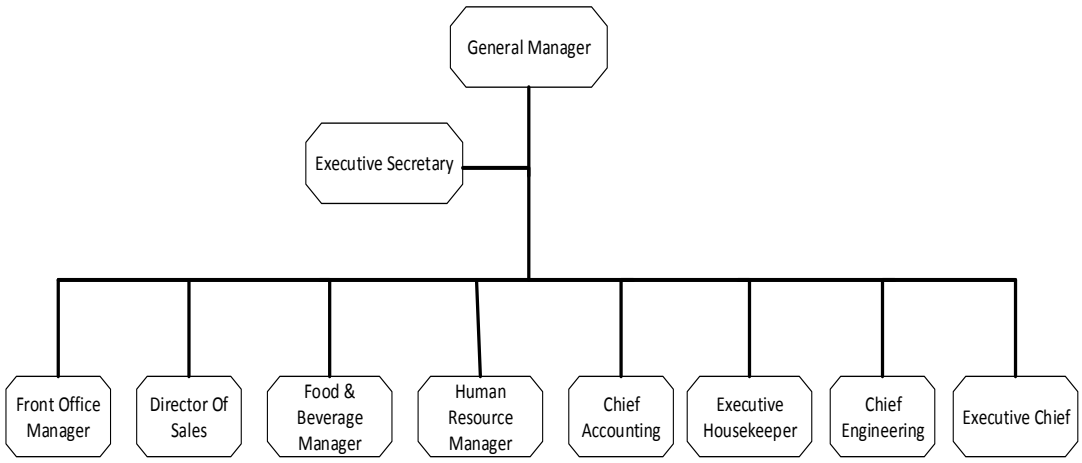
Ruangan : 8 Ruangan

Kapasitas : 90 pax

Jam Buka : 6 PM – 3 AM

2.1.5. Struktur Organisasi di Hotel Golden Flower Bandung

Struktur organisasi merupakan susunan seluruh organisasi yang terkait di Hotel Golden Flower Bandung, dumulai dari tertinggi sampai terendah. Untuk lebih jelasnya dapat dilihat pada gambar 2.2



Sumber Gambar : Dokumentasi Golden Flower Bandung

Gambar 2.2 Struktur Organisasi Hotel Golden Flower Bandung

Keterangan :

1. FOM : *Front Office Manager*
2. DOS : *Director Of Sales*
3. FBM : *Food an Bevarage Manager*
4. HRM : *Human Resource Manager*
5. CA : *Chief Accounting*
6. Exc. HK : *Executive Housekeeper*
7. CE : *Chief Enginering*
8. Exc. Chief : *Executive Chief*

Berikut ini tugas dan tanggung jawab Department yang ada di Hotel Golden Flower Bandung :

1. *Sales & Marketing Departement*

   Bertugas untuk mencari pelanggan hotel sebanyak-banyaknya dengan cara melakukan kunjungan ke berbagai perusahaan, presentasi ataupun mempromosikan hotel melalui brosur-brosur, surat kabar, media elektronik dan media cetak.

   Public Relation juga termasuk dalam Sales & Marketing Departement, yang bertugas menyebarkan informasi mengenai keberadaan hotel dengan segala aktifitasnya yang perlu diketahui oleh masyarakat luas.

2. *Food And Beverage Departement*

   Bertugas melayani pelanggan hotel dalam menyajikan makanan dan minuman di Restaurant, Bar, Room Service dan lainnya.

3. *Front Office Departement*

   Bertugas menerima pesanan kamar untuk pelanggan hotel serta menerima pelanggan hotel dan segala administrasinya termasuk informasi yang diperlukan pelanggan hotel serta mengurus barang-barang milik pelanggan hotel pada saat datang dan pulang.

4. *Human Resource Manager Departement*

   Bertugas mengurus segala hal yang menyangkut ketenagakerjaan seperti penerimaan karyawan, pemecatan / pengeluaran karyawan, administrasi karyawan, training karyawan, training sekolah, kesehatan karyawan, absensi, penggujian dan lainnya. HRD membawahi Driver dan Security yang bertugas mengurus kendaraan operasional hotel dan menjaga serta memelihara keamanan di area hotel, baik yang menyangkut barang maupun orang termasuk karyawan dan pelanggan hotel.

   Selain itu juga Human Resource Departement memiliki tugas melakukan rencana, program, kebijakan, prosedur, pemeliharaan, koordinasi, actuating dan pengawasan atas recuitment dan seleksi pribadi karyawan, kesejahteraan, upah dan gaji juga pelatihan dan pengembangan dan hal-hal umum untuk mendukung Operasi Hotel. Dan mengkoordinasikan fungsi-fungsi SDM dan kegiatan dengan regional dan staf perusahan yang sesuai.

5. *Chef Kitchen Departement*

   Bertugas menyediakan keperluan makanan dan minuman pelanggan hotel yaitu makanan dipersiapkan dan diolah di dapur. Termasuk juga membersihkan alat-alat dapur dan peralatan makan minum lainnya. Selain itu juga department ini bertugas untuk membuat market list atau barang dan bahan apa saja yang akan di beli untuk event yang akan ada di hotel serta menjalin hubungan baik dengan supplier barang sehingga akan tetap terjalin harmonisasi dalam pengiriman barang.

6. *Accountinng Departement*

   Bertugas mengatur, mengawasi dan membukukan keluar masuk uang, barang dan segala kekayaan milik hotel termasuk pengelolaan data dan laporan yang diperlukan.

   Bertanggung jawab atas pengembangan, pengarahan dan koordinasi pencatatan yang baik atas transaksi keuangan perusahaan, memelihara pencatatan yang memadai dari semua kekayaan, harta benda dan semua transaksi lainnya dari perusahaan dengan pihak luar. Menyiapkan laporan rencana keuangan dan menjaga sistem kontrol keuangan cepat dan menguntungkan sesuai dengan semua peraturan, tata cara dan ketentuan.

7. *Housekeeping Departement*

   Bertugas mengurus kebersihan kamar serta area hotel secara keseluruhan termasuk mengurus persediaan dan pembagian pakaian seragam karyawan serta kebutuhan-kebutuhan pelanggan hotel di kamar.

8. *Enginering Departement*

   Bertugas memelihara dan menjaga agar air, listrik, telepon, televisi, lift, Air Conditioning termasuk peralatan hotel lainnya tetap berfungsi dengan baik dan lancar.

2.2 Landasan Teori

Pada landasan teori akan diterangkan teori-teori yang berhubungan dengan judul penulisan ini diantaranya pembahasan mengenai Penerapan *Geofencing* sebagai media pencarian objek wisata dan pelayanan pelanggan berbasis android di Hotel Golden Flower Bandung meliputi Aplikasi *Mobile*, Pengertian Aplikasi, Pengertian *Mobile*, Pengertian Aplikasi *Mobile*, Karakteristik Perangkat *Mobile*, Android, Teori Kualitas, Objek Wisata, Basis Data, DBMS, SQL, Pemograman Berorientasi Objek, Konsep Dasar Berorientasi Objek, *Geofencing*, Jaringan Internet, GPS, LBS, *Google Maps*, Android Studio, HTML5, PHP, JavaScript, JSON, Database MySQL, *Unified Modeling Laguange* (UML), Web Server dan Web Service. Berikut ini akan dibahas mengenai teori-teori yang akan menjadi sumber kajian dari aplikasi yang akan dibangun.

2.2.1 Aplikasi *Mobile*

Aplikasi *mobile* adalah sebuah aplikasi yang dapat dilakukan dengan mobilitas dengan menggunakan PDA, telepon seluler atau *Handphone*. Dengan menggunakan aplikasi *mobile* pengguna dengan mudah melakukan aktifitas seperti hiburan, pekerjaan, berjualan dan lain sebagainya menggunakan telepon seluler.

2.2.2 Pengertian Aplikasi

Menurut Buyens **[**1**]** Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Jika ingin mnegembangkan program aplikasi sendiri, makaa untuk menulis program aplikasi tersebut dibutuhkan suatu bahasa pemograman, yaitu language software, compiler ataupun interpreter. Jadi *language software* merupakan bahasanya dan program yang ditulis merupakan program aplikasinya. *Language software* berfungsi agar dapat menulis program dengan bahasa yang lebih mudah dan akan menterjemahkannya ke dalam bahasa mesin supaya bisa dimengerti oleh komputer. Bila hendak mengembangkan suatu program aplikasi untuk memecahkan permasalahan yang besar dan rumit, maka supaya program aplikasi tersebut dapat berhasil dengan baik, maka dibutuhkan prosedur dan perencanaan yang baik dalam mengembangkannya.

2.2.3 Pengertian *Mobile*

Mobile diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat yang lain, misalnya telepon *mobile* berarti terminal telepon yang dapat berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi.

2.2.4 Pengertian Aplikasi Mobile

Menurut Buyens **[**1**]** Aplikasi mobile berasal dari kata *application* dan *mobile*. *Application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju

sedangkan mobile dapat di artikan sebagai perpindahan dari suatu tempat ke tempat yang lain.

Aplikasi *mobile* dapat di artikan sebuah program aplikasi yang dapat dijalankan atau digunakan walaupun pengguna berpindah-pindah dari satu tempat ke tempat yang lain serta mempunyai ukuran yang kecil. Aplikasi *mobile* ini dapat di akses melalui perangkat nirkabel, pager, PDA, telepon seluler, *smartphone*, dan perangkat sejenisnya.

2.2.5 Karakteristik Perangkat *Mobile*

Perangkat *mobile* memiliki banyak jenis dalam hal ukuran, desain dan tampilan, tetapi mereka memiliki karakteristik yang sangat berbeda dari sistem desktop. Berikut karakteristik perangkat *mobile*, diantaranya:

1. Ukuran Kecil

   Perangkat *mobile* memiliki ukuran yang kecil. Konsumen mengiginkan perangkat yang kecil untuk kenyamanan dan mobilitas mereka.

2. *Memory* yang terbatas

   Perangkat *mobile* juga memiliki *memory* yang kecil, yaitu *primary* (RAM) dan *secoundary* (disk). Pembatasan ini adalah salah satu faktor yang mempengaruhi penulisan program untuk berbagai jenis dari perangkat ini. Dengan pembatasan jumlah dari *memory*, pertimbangan-pertimbangan khusus harus diambil untuk memelihara pemakaian dari sumber daya yang mahal ini.

3. Daya proses yang terbatas

   Sistem *mobile* tidaklah setangguh rekan mereka yaitu desktop. Ukuran, teknologi dan biaya adalah beberapa faktor yang mempengaruhi status dari sumber daya ini. Seperti harddisk dan RAM, anda dapat menemukan mereka dalam ukuran yang pas dengan sebuah kemasan kecil.

4. Mengkonsumsi daya yang rendah

   Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin desktop. Perangkat ini harus menghemat daya karena mereka berjalan pada keadaan dimana daya yang disediakan dibatasi oleh baterai-baterai.

5. Kuat dan dapat diandalkan

   Karena perangkat *mobile* selain dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan dan sesekali tetesan-tetesan air.

6. Konektivitas yang terbatas

   Perangkat *mobile* memiliki *bandwith* rendah, bebrapa dari mereka bahkan tidak tersambung, kebanyakan dari mereka menggunakan koneksi *wireless*.

7. Masa hidup yang pendek

   Perangkat-perangkat konsumen ini menyala dalam hitungan-hitungan detik kebanyakan dari mereka selalu menyala. Kita ambil kasus dari *handphone*, mereka booting dalam hitungan detik dan kebanyakan orang tidak mematikan *handphone* mereka bahkan ketika malam hari. PDA akan menyala jika anda menekan tombol power mereka.

Aplikasi *mobile* terdapat beberapa tipe, berikut beberapa tipe aplikasi *mobile* :

A. Berdasarkan Teknologi yang digunakan :

1. *Server-Request Based*

   Mobile Equipment (ME) merequest data dari server, WAP, menggunakan WML, sebuah *markup language* khusus berbasis XML.

   Keuntungan :

   a. Platform Independent
   b. Mudah diterapkan
   c. Jenis aplikasi beragam

Kerugian :

a. Masih banyak ME yang tidak support WAP
b. Pembayaran untuk data
c. Interface terbatas.

Implementasi :

a. Menggunakan Web Server dengan setting tambahan di konfigurasi
b. File WML dibuat selayaknya membuat file HTML

2. SMS *based*

Short Message Service (SMS) pengiriman pesan singkat antar ME dengan sebuah server terbatas ±160 karakter.

Keuntungan :

a. Platform Independent
b. ME sebagian besar sudah dapat menerapkan SMS

Kerugian :

a. Pembayaran untuk setiap aksi
b. *Interface* hanya text, user harus ingat perintah
c. Panjang text sangat terbatas.

Implementasi :

a. Menggunakan *interface* antara ME dengan server
   Aplikasi menunggu data dari *interface* dan mengirim data melalui *interface* tersebut.

3. SIM-*Card Based*

Aplikasi diletakkan diatas SIM-Card, aplikasi dapat diakses melalui menu baru yang akan muncul di ponsel. Jacavard, khusus untuk javacard aplikasi bersifat *downloadable*, pengirim data lewat SMS.

Keuntungan :

a. Platform Independent
b. Data disimpan secara lokal dan dapat digunakan bila berganti ME

Kerugian :

a. Size terbatas
b. Aplikasi harus sangat kecil
c. Sulit diterapkan

Implementasi :

a. Download Javacard
b. Gunakan Simulator
c. *Mobile* Equipment Based

Aplikasi diletakkan di *Mobile Equipment*, contoh Pemograman diatas Syimbian OS, Pengiriman data SMS.

Keuntungan :

a. *Interface* sangat mudah
b. Jenis aplikasi sangat beragam, bahkan dapat digunakan untuk aplikasi berat.

Kerugian :

a. Platform Dependent
b. Bila mengubah ME kemungkinan aplikasi dan data tidak dapat digunakan.

B. Berdasarkan Fungsi :

1. *Information*
   Pengiriman informasi ke ME user
2. *Communications*
   Komunikasi anatr user dengan satu atau lebih user lain
3. *Entertaiment*
   Aplikasi menghibur seperti games
4. *Productivity Tools*
   Aplikasi yang dapat meningkatkan produktivitas walaupun sedang berpindah
5. *Commerce and Banking*
   Jual beli melalui ME

6. *Location based*

   Aplikasi yang mengubah data sesuai dengan lokasi user. Lokasi dapat ditentukan oleh GPS dan sel yang ditempati oleh user.

2.2.6 Android

Android merupakan sistem operasi berbasis *Linux* yang dirancang untuk perangkat telepon seluler seperti telepon pintar dan tablet. Sistem operasi lainnya seperti *Windows Mobile*, i-Phone 4, Symbian S40 dan masih banyak lagi yang menawarkan kekayaan isi dan keoptimalan berjalan di atas perangkat hardware yang ada. Akan tetapi, sistem operasi yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh kerena itu, adanya keterbatasan distribusi aplikasi pihak ketiga untuk platform mereka.

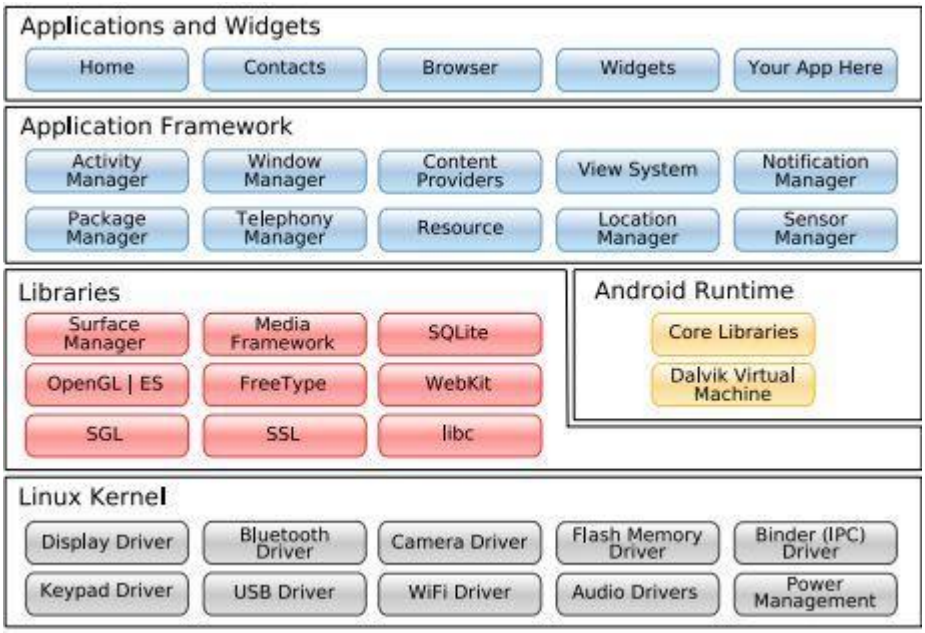
Android menawarkan sebuah lingkungan yang berbeda untuk pengembang. Setiap aplikasi memiliki tingkatan yang sama. Android tidak membedakan antara aplikasi inti dengan aplikasi pihak ketiga. *Application Programming Interface* (API) yang di sediakan menawarkan akses ke hardware, maupun data-data ponsel sekalipun, atau data sistem sendiri. Bahkan pengguna dapat menghapus aplikasi inti dan menggantikannya dengan aplikasi pihak ketiga. [**2**]

2.2.6.1 Arsitektur Android

Secara garis besar Arsitektur Android terdiri atas *Application and Widgetds*, *Application Frameworks*, *Libraries*, *Android Run Time* dan *Linux Kernel* dijelaskan sebagai berikut [**3**]:

a. Application dan Widgets

Application and Widgets merupakan layer dimana kita berhubungan dengan aplikasi kita. Di layer terdapat aplikasi inti browser, kontak, email dan lain-lain. Hampir semua aplikasi ditulis menggunakan bahasa pemograman Java.



Sumber Gambar : http://developer.android.com **[**3**]**

Gambar 2. 3 Arsitektur Android

b. Applicatio Framework

Android Adela "*Open Development Platform*" yang ditawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi. Pengembang memiliki akses penuh menuju API *framework* seperti yang dilakukan oleh aplikasi kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*). Komponen-komponen yang termasuk di dalan Application Framework adalah sebagai berikut:

1. *Views*
2. *Content Provider*
3. *Resource Manager*
4. *Notification Manager*
5. *Activity Manager*

c. Libraries

Libraries ini adalah layer dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses libraries untuk menjalankan aplikasinya.

Berjalan diatas Kernel, layer ini meliputi berbagai library C/C++ inti seperti Libc SSL, serta :

1) *Libraries* media untuk pemutaran media audio dan video
2) *Libraries* untuk manajemen tampilan
3) *Libraries Graphics* mencakup SGL I *OpenGL* untuk grafis 2D dan 3D.
4) *Libraries SQLite* untuk dukungan database
5) *Libraries* SSL dan *WebKit* terintegrasi dengan web browser dan *security*
6) *Libraries LiveWebcore* mencakup modern web browser dengan *engine embedded webview*
7) *Libraries* 3D yang mencakup implementasi OpenGL ES1.0 API's

d. Android Run Time

Layer yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan Implementasi Linux. Dalvik Virtual Machine (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam *Android Run Time* dibagi menjadi dua bagian yaitu :

1. *Core Libraries*: Aplikasi Android dibangun menjadi dalam bahasa Java, sementara Dalvik sebagai virtual mesinnya bukan Virtual Machine Java, sehingga diperlukan sebuah libraries yang berfungsi untuk menterjemahkan bahasa Java/C yang ditangani oleh Core *Libraries*.
2. *Dalvik Virtual Machine* : Virtual mesin berbasis register yang dioptimalkan unutk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat Linux Karnel untuk melakukan threading dan menajemen tingkat rendah.

e. Linux Kernel

Linux Kernel adalah layer dimana inti dari sistem operasi Android itu berada. Berisi file-file sistem yang mengatur sistem processing, memory,

resource, drivers, dan sistem-sistem Operasi Android lainnya. *Linux Kernel* yang digunakan Android adalah *Linux Kernel release* 2.6

2.2.7 Teori Kualitas

Menurut Roger S. Pressman [**4**] "Kualitas sistem merupakan aplikasi atau produk merupakan persyaratan yang menjelaskan masalah, desain model solusi, kode yang membuat program dapat dieksekusi dan pengujian yang menguji perangkat lunak untuk menemukan kesalahan".

Software menjadi salah satu komponen penting dalam kehidupan manusia. Software digunakan untuk mempermudah pekerjaan-pekerjaan yang memerlukan keakuratan, ketepatan serta efisiensi waktu yang tinggi.

2.2.7.1 Faktor-faktor Yang Mempengaruhi Kualitas

McCall dan Cavano dalam buku Roger S. Pressman, [**4**] menjelaskan bahwa faktor-faktor yang mempengaruhi Software terbagi menjadi tiga, yaitu :

1. Operasi Produk (menggunakannya)
2. Revisi Produk (mengubahnya)
3. Transisi produk (memodifikasi untuk berkerja dalam lingkungan yang berbeda)

Adapun hubungan antara faktor-faktor kualitas "(kerangka kerja)" dan aspek-aspek lain dari proses rekayasa perangkat lunak adalah :

Pertama, kerangka kerja memberikan suatu mekanisme untuk manajer proyek untuk mengenali kualitas-kualitas apa yang penting. Kualitas tersebut merupakan atribut perangkat lunak, sebagai tambahan untuk kokreksi dan kinerja fungsionalnya, yang mempunyai implikasi daur hidup.

Kedua, kerangka kerja memberikan alat untuk menilai secara kuantitatif seberapa baik kemajuan pengembangan.

Ketiga, kerangka kerja memberikan interaksi yang lebih dalam pada software.

2.2.7.2 Kualitas Software (Perangkat Lunak)

Kualitas perangkat lunak adalah gangguan yang kompleks dari berbagai faktor yang akan bervariasi pada aplikasi dan pelanggan yang berbeda yang

membutuhkannya. Faktor yang yang mempengaruhi kualits perangkat lunak dapat dikategorikan ke dalam dua kelompok besar, yaitu :

1. Faktor yang dapat secara langsung diukur.
2. Faktor yang tidak dapat secara langsung diukur.

Menurut McCall dan kawan-kawan pada tahun 1977 dalam Roger S. Pressman telah mengusulkan suatu penggolongan faktor-faktor atau dimensi-dimensi yang mempengaruhi kualitas suatu software. Pada dasarnya McCall menitikberatkan faktor-faktor tersebut menjadi 3 (tiga) aspek penting, yaitu yang berhubungan dengan :

1. Sifat-sifat operasional dari software (*Product Operation*)
2. Kemampuan software dalam menjalani perubahan (*Product Revision*)
3. Daya adaptasi atau penyesuaian software terhadap lingkungan baru (*Product Transition*)

Indikator yang menentukan kulitas suatu sistem menurut Roger S. Pressman membuat sejumlah faktor-faktor kualitas yang disingkat FURPS (*Functionality, Usability, Performance, Realibity dan Supportability*),dimensi-dimensinya dapat diuraikan sebagai berikut :

1. *Functionality*, dinilai melalui evaluasi bentuk himpunan dan kemampuan program, generalitas fungsi-fungsi yang disampaikan dan keamanan keseluruhan sistem.
2. *Usability*, dinilai dengan mempertimbangkan faktor manusia, keseluruhan estetika, konsistensi, dan dokumentasi.
3. *Reliability*, dievaluasi melalui pengukuran frekuensi dan besarnya kegagalan, akurasi hasil *output*, *mean time between failure* (MTBF),kemampuan untuk pulih dari kegagalan dan predik-tabilitas program.
4. *Performance*, diukur melalui kecepatan pemrosesan, waktu respon, konsumsi kode sumber, throughput, dan efisiensi.
5. *Suportability*, menggabungkan kemampuan yang memperluas program (*ekstensibilitas*), kemampuan beradaptasi, dan

kemampuan pelayanan (atribut-atribut ini mempresentasikan suatu teknologi yang lebih, umum *maintainability*), serta testabilitas, kompatibilitas (kemampuan untuk mengumpulkan dan mengontrol elemen dari konfigurasi perangkat lunak, kecocokan di mana suatu system dapat dipasang dan kecocokan di mana masalah dapat dialokasikan.

Perangkat lunak dikatakan baik apabila secara utuh dan sempurna memenuhi kriteria spesifik dari organisasi perusahaan yang membutuhkan. Hal ini sering di istilahkan sebagai pemenuhan terhadap "*User Requirement*" (kebutuhan pengguna software yang telah terlebih dahulu didefinisikan secara jelas dan detail).

Menurut Roger S. Pressman [**4**] " Software adalah perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan".

Perangkat lunak merupakan kumpulan beberapa perintah yang dieksekusi oleh mesin komputer dalam menjalankan pekerjaannya. Perangkat lunak merupakan catatan bagi mesin komputer untuk menyimpan perintah, maupun dokumen serta arsip lainnya.

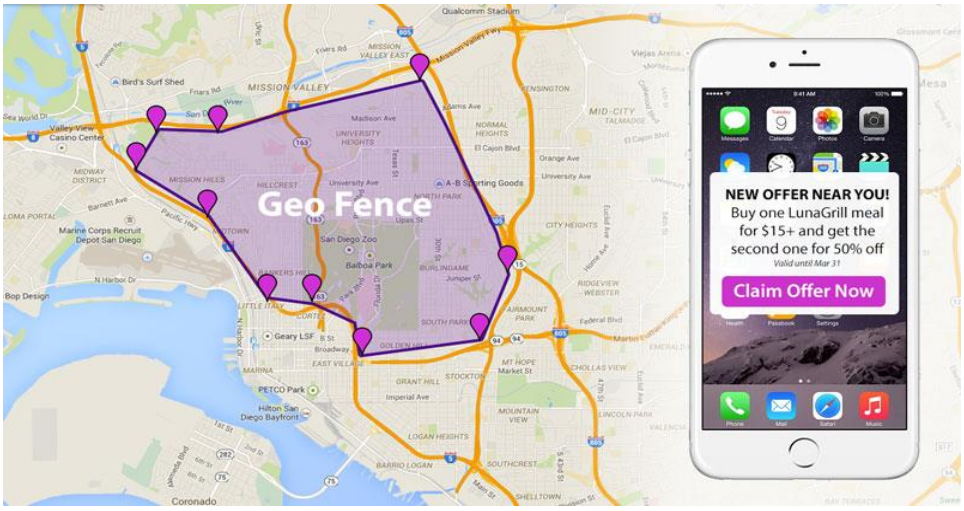
Perangkat lunak juga dapat disimpan sedemikian rupa oleh komputer itu sendiri, data yang disimpan ini dapat berupa program atau instruksi yang akan dijalankan oleh perintah, maupun catatan-catatan yang diperlukan oleh komputer untuk menjalankan perintah yang dijalankannya.

2.2.8 *Geofencing*

Geofencing adalah parameter virtual untuk wilayah geografis di dunia nyata. Parameter Virtual yang ditetapkan ini bias menjadi status ataupun dinamis. Salah satu contoh khas dari *geofence* dinamis adalah radius sekitar atau titik lokasi. *Geofene* statis dapat di atur dari batas yang telah ditetapkan, seperti zona kehadiran di sekolah atau batas lingkungan. Kostum digital *geofencing* juga dapat digunakan, jika perangkat yang menyediakan layanan berbasis lokasi (LBS) pengguna masuk atau keluar *geofencing*, perangkat akan menerima pesan notifikasi. Notifikasi ini mungkin berisi informasi tentang lokasi dari pernagkat.

Notifikasi *Geo-fence* mungkin dikirimkan ke ponsel sebagai contoh meningkatkan setiap bentuk lain dari tindakan

Kita bisa menggambarkan *geo-fencing* sebagai LBS proaktif. Dalam banyak kasus sistem proaktif jauh lebih nyaman dari yang reaktif, dimana pengguna harus secara eksplisit meminta data untuk lokasi. Ada berbagai jenis lokasi pariwisata untuk memperbaiki posisi GPS yang dapat diuji. Misalnya, apakah pengguna di dekat dengan tempat tujuan (POI) atau ke pengguna lain.



Sumber Gambar: http:// code.tutsplus.com [**5**]

Gambar 2. 4 *Geofencing*

Berikut cara kerja *Geofencing*, yaitu:

1. *Transition Dwell*, menunjukkan bahwa pengguna memasuki daerah yang sudah di beri tanda..
2. *Transition Enter*, menunjukkan ketika pengguna memasuki wilayah dipantau.
3. *Transition Exit*, menunjukkan kapan pengguna keluar daerah
4. *Geofencing Request*, memberikan pemberitahuan untuk
5. *Geofencing API*, titik masuk semua interaksi dengan google API.

Geofencing sekarang ini menjadi salah satu wajah baru di Layanan Berbasis Lokasi. Sekarang pengembangan asli untuk LBS sendiri lebih atau kurang selesai. Kita bias melihat beberapa standar umum dan aplikasi (berfikir tentang berbagai implementasi layanan tempat-tempat seperti *Foursquare*, *Twitter place*, *Google place*, *Facebook place*, dan lain-lain) semua dari mereka memudahkan anda dalam berbagi info lokasi atau mendapatkan beberapa hal yang berada di

dekatnya. Dan geo-fence membuka pintu untuk beberapa personalisasi. Anda bisa mendapatkan beberapa kustom disesuaikan dengan data ditempat yang benar, terutama ketika kedua aliran data dan tempat (area) untuk menerima itu dihasilkan secara dinamis. Pada kedua, geo-fence akan terintegrasi dalam waktu terdekat dengan sensor dan aplikasi M2M. [**5**]

2.2.9 Objek Wisata

Objek wisata adalah segala sesuatu yang dapat menarik wisatwan untuk menjadi sasaran kunjungannya. Sehingga sesuatu dapat ddikatakan sebagai objek wisata mempunyai prasarat, mempunya daya tarik tersendiri bagi para wisatawan. [**6**]

2.2.10 Pelayanan

Pelayanan adalah suatu kegiatan atau urutan kegiatan yang terjadi dalam interaksi langsung antara seseorang dengan orang lain atau mesin secara fisik, dan menyediakan kepuassan pelanggan. Dalam kamus besar Bahasa Indonesia dijelaskan pelayanan sebagai usaha melayani kebutuhan orang lain. Sedangkan melayani adalah membantu menyiapkan (mengurus) apa yang diperlukan seseorang. [**7**]

2.2.11 Pelanggan

Pelanggan adalah orang yang secara terus-menerus datang berkali-kali ke suatu tempat yang sama untuk membeli produk atau mendapatkan jasa agar dapat memenuhi keinginannya.

2.2.12 Basis Data

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. [**8**]

Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apapun bentuknya, entah berupa file teks ataupun *Database Management System* (DBMS). Kebutuhan basis data dalam sistem informasi meliputi :

1. Masukkan, menyimpan, dan mengambil data.
2. Membuat laporan berdasarkan data yang telah disimpan.

2.2.13 DBMS

DBMS (*Database Management System*) atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data. Sitem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data [**8**]. Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut :

1. Menyediakan fasilitas untuk mengelola akses data
2. Mampu menangani integritas data
3. Mampu menangani akses data yang dilakukan secara bersamaan
4. Mampu menangani backup data

2.2.14 SQL

SQL (*Structured Query Language*) adalah bahasa yag digunakan untuk mengelola data pada RDBMS. SQL pertama dikembangkan berdasarkan teori aljabar relasional dan kalkulus.

Meskipun SQL diadopsi dan diacu sebagai bahasa standar oleh hampir sebagian besar RDBMS yang beredar saat ini, tetapi tidak semua standar yang tercantum dalam SQL diimplementasikan oleh seluruh DBMS tesebut. Sehingga sering kali ada perbedaan perilaku (hasil yang ditampilkan) oleh DBMS yang berbeda padahal *query* yang dimasukkan sama. **[**8**]**

2.2.15 Pemograman Berorientasi Objek

Metodologi berorientasi objek yaitu suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangakaian aktivitas analisis berorientasi objek, percancangan

berorientasi objek, pemograman berorientasi objek, dan pengujian berorientasi objek. **[**8**]**

Pada saat ini, metode berorientasi objek banyak dipilih karena metodologi lama banyak menimbulkan masalah seperti adanya kesulitan pada saat mentransformasi hasil dari satu tahap pengembangan ke tahap berikutnya, misalkan pada metode pendekatan terstruktur, jenis aplikasi yang dikembangkan saat ini berbeda dengan masa lalu. Aplikasi yang dikembangkan pada saat ini beragam dengan platform yang berbeda-beda, sehingga menimbulkan tuntutan kebutuhan metodologi pengembangan yang dapat mengakomodasi ke semua jenis aplikasi tersebut. Keuntungan menggunakan metodologi berorientasi objek adalah sebagai berikut :

1. Meningkatkan Produktivitas

   Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut (*reusable*).
2. Kecepatan Pengembangan

   Karena sistem yang dibangun dengan baik dan benar pada saat analisis dan perancangan akan menyebabkan berkurangnya kesalahan pada saat pengodean.
3. Kemudahan Pemeliharaan

   Karena dengan model objek, pola-pola yang cenderung tetap dan stabil dapat dipisahkan dan pola-pola yang mungkin sering berubah-ubah.
4. Adanya Konsistensi

   Karena sifat pewarisan dan penggunaan notasi yang sama pada saat analisis, perancangan maupun pengkodean.
5. Meningkatkan Kualitas Perangkat Lunak

   Karena pendekatan pengembangan lebih dekat dengan dunia nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

2.2.16 Konsep Dasar Beorientasi Objek

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan pbjek-objek dunia nyata.

Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dan abstraksi objek, kelas, hubungan antar kelas sampai abstraksi sistem. Saat mengabstraksikan dan memodelkan objek, data dan proses-proses yang dipunyai oleh objek akan dienkapsulasi (dibungkus) menjadi suatu kesatuan.

Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancangan, pemograman, dan penguian perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu. **[**8**]**

Sistem berorientasi objek merupakan sebuah sistem yang dibangun dengan berdasarkan metode berorinetasi objek dengan komponennya dibungkus (dienkapulasi) menjadi kelompok data dan fungsi. Setiap komponen dalam sistem tersebut dapat mewarisi atribut dan sifat dan komponen lainnya, dan dapat berinteraksi satu sama lain.

Berikut ini adalah beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek :

1. Kelas (*Class*)

   Kelas adalah sekumpulan objek-objek dengan karaketristik yang sama. Kelas merupakan definisi status dan himpunan objek yang sama yang mungkin lahir atau diciptakan dan kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (metode/operasi), hubungan (*relationship*) dan arti. Suatu kelas dapat diturunkan dan kelas yang lain, dimana atribut dan kelas semula dapat diwariskan ke kelas yang baru. Secara teknik kelas adalah sebuah struktur pada kode program yang menggunakan metodologi berorientasi objek.

2. Objek (*Object*)

   Objek adalah abstraksi dari sesuatu yang mewakili dunia nyata benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan. Secara teknis, sebuah kelas saat program dieksekusi maka akan dibuat sebuah objek. Objek dilihat dari segi teknis adalah elemen pada saat *runtime* yang akan diciptakan, dimanipulasi dan dihancurkan saat eksekusi sehingga sebuah objek hanya ada saat sebuah program dieksekusi. Jika masih dalam bentuk kode, disebut seagai kelas jadi pada saat *runtime* (saat sebuah program dieksekusi), yang kita punya adalah objek, di dalam teks program yang kita lihat hanyalah kelas.

3. Metode (*Method*)

   Operasi atau metode pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuah kelas boleh memiliki lebih dari satu metode atau operasi. Metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri. Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek. Metode atau operasi dapat berasal dari *event*, aktifitas atau aksi keadaan, fungsi, atau kelakuakn dunia nyata. Contoh metode atau operasi misalnya *Read*, *Write*, *Move*, *Copy*, dan sebaagainya.

4. Atribut (*Attribute*)

   Atribut dari sebuah kelas adalah variable global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis, nama dan sebagainya.

5. Abstraksi (*Abstraction*)

   Prinsip untuk mempresentasikan dunia nyata yang komplek menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

6. Enkapsulasi (*Encapsulation*)

   Pembungkus atribut data dan layanan operasi-operasi yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya.

7. Pewarisan (*Inheritance*)

   Mekanismen yang memungkinkan satu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dan dirinya.

8. Antarmuka (*Interface*)

   Antarmuka sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode pada sebuah *interface* dapat diimplementasikan oleh kelas lain.

9. *Reusability*

   Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainnya yang melibatkan objek tersebut.

10. Generalisasi dan Spesialisasi

    Menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus. Misalnya kelas yang lebih umum (generalisasi) adalah kendaraan darat dan kelas khususnya (spesialisasi) adalah mobil, motor dan kereta.

11. Komunikasi Antar Objek

    Komunikasi antar objek dilakukan lewat pesan (*message*) yang dikirim dari satu objek ke objek lainnya.

12. Polimorfisme (*Polymorphism*)

    Kemampuan suatu objek digunakan di banyak tujuan yang berbeda dengan nama yang sehingga menghemat baris program.

13. *Package*

Package adalah sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam package yang berbeda.

2.2.17 Jaringan Internet

Internet adalah salah satu jaringan komputer yang dapat menghubungkan beberapa komputer diseluruh dunia tanpa batas dimana orang yang memiliki komputer dapat terhubung dalam jaringan ini hanya dengan melakukan koneksi ke layanan internet seperti modem, speedy, LAN, Wifi dan lain sebagainya.

Internet merupakan satu jaringan komputer global yang terbentuk dari jaringan-jaringan komputer lokal dan regional yang memungkinkan komunikasi data antar komputer yang terhubung ke jaringan tersebut. Pada tahun 1969 empat buah *interface message processor* (IMP) dikirimkan ke 4 perguruan tinggi yaitu UCLA, *Standard Research Institute* (SRI), *University of California Santa Barbara* (USCB), dan *University of Utah*. Jaringan ke empat tempat ini disebut sebagai ARPANET, yang disponsori oleh *Defense of Advanced Reserch Project Agency* (DARPA) milik Pentagon, Departemen Pertahanan Amerika Serikat. **[**9**]**

2.2.18 GPS (*Global Positioning System*)

GPS adalah singkatan dari *Global Positioning System*, yang merupakan sistem navigasi dengan menggunakan teknlogi satelit yang dapat menerima sinyal dan satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal glombang mikro ke bumi. Sinyal ini diterima oleh alat penerima (*receiver*) di permukaan, dimana GPS receiver ini akan mengumpulkan informasi dari satelit GPS, seperti :

a. Waktu

   GPS *receiver* menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.

b. Lokasi

   GPS memberikan informasi lokasi dalam tiga dimensi:

   1. Latitude
   2. Longitude

3. Elevasi

c. Kecepatan

Ketika berpindah tempat, GPS dapat menunjukkan informasi kecepatan berpindah tersebut.

d. Arah Perjalanan

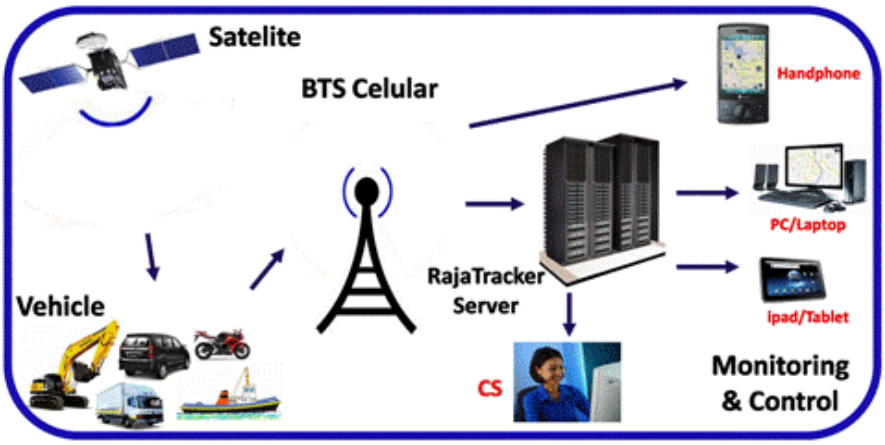
GPS dapat menunjukkan arah tujuan.

e. Simpan Lokasi

Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS *receiver*.

f. Komulasi Data

GPS receiver dapat menyimpan informasi track, seperti total perjalanan yang sudah pernah dilakukan, kecepatan rata-rata, kecepatan paling tinggi, kecepatan paling rendah, waktu/jam sampai tujuan, dan sebagainnya. [**10**]



Sumber Gambar: http://borneogps.com [**10**]

Gambar 2. 5 Cara Kerja GPS

2.2.18.1 Akurasi GPS

Posisi yang ditunjukkan oleh suatu GPS mempunyai faktor kesalahan atau juga disebut tingkat akurasi . sebagai contoh suatu alat GPS menunjukan titil kooordinat dengan tingkat akurasi 5 meter, itu berarti posisi pengguna bias berada dalam range radius 5 meter dari titik yang ditunjukkan tersebut. Mengapa tingkat akurasi yang terlihat biasa berubah-ubah? Kadang terlihat 10 meter, 15 meter, atau

5 meter. Ada beberapa hal yang bisa mempegaruhi tingkat akurasi tersebut, antara lain :

a. Kesalahan Ephemeris. Terjadi jika satelit tidak dapat mentransmisikan posisinya di orbit dengan tepat.
b. Keadaan Ionosphere. Ionosphere berada pada jarak sekitar 43-50 mil di atsa permukaan bumi. Satelit yang melewati ionosphare akan menjadi lambat dikarenakan adanya plasma (gas dengan tingkat kepadatan rendah). Walaupun GPS *receiver* berusaha untuk mengkoreksi/memperbaiki faktor keterlambatan yang terjadi tetap saja aktivitas tertentu dari plasma bisa menyebabkan kesalahan perhitungan.
c. Keadaan Troposphere. Troposphere adalah bagian terendah dari atmosfer sampai dengan ketinggian sekitar 11 mil dari permukaan tanah. Variasi pada temperatur, tekanan, dan kelembaban bisa menyebabkan perbedaan kecepatan penerimaan gelombang radio.
d. Kesalahan Waktu. Karena penempatan jam atom pada setiap GPS *receiver* tidak berjalan sebagaimana mestinya. Kesalahan waktu dari GPS *receiver* yang tidak presisi dapat menimbulkan ketidakakurasian.
e. Kesalahan Multipath. Terjadi karena sinyal satelit membentur permukaan keras (seperti bangunan atau tebing) sebelum mencapai GPS *receiver*. Hal tersebut bisa menyebabkan tejadinya delay sehingga perhitunggan jarak menjadi tidak akurat.
f. Buruknya Sinyal Satelit. Keadaan langit yang terhalang akan menyebabkan GPS sulit menerima data satelit. Sebuah sinyal satelit yang pada hari tertentu diterima dengan sangat bagus belum tentu pada hari lain bisa diterima dengan kualitas yang sama walaupun user berdiri pada tempat yang sama. Hal tersebut dikarenakan posisi dari satelit yang terus bergerak atau bisa juga disebabkan faktor penghalang lain seperti pohon, gedung bertingkat, dan sebagainya. **[**10**]**

Ada beberapa macam-macam sensor GPS, yaitu:

1. *Sensor Accelerometer*: Sensor ini akan mendeteksi kecepatan teleponmu ketika digerakkan. Pembacaan *Accelerometer* akan melonjak pada setiap gerakan. Pertahankan *accelerometer* untuk tetap datar pada permukaan dan pembacaannya akan stabil. Hal ini juga digunakan untuk menentukan dimana ponsel berada dalam 3 dimensi. Fungsi dari *Acceleromater* yang paling umum adalah merubah tampilan layar secara otomatis dari landscape menjadi portrait atau sebaliknya.
2. *Sensor Gyroscope* : Giroskop merupakan selangkah lebih maju dari *Accelerometer*. Meski accelerometer bisa menunjukkan di mana ponsel berada dalam tiga dimensi, tetapi tidak bisa mengatakan bagaimana cara berputar di dalam posisi tiga dimensi. Jadi, giroskop membantu untuk mendeteksi rotasi sebuah perangkat, atau peregerakan ke atas, bawah, kanan dan kiri, maju dan mundur. Sensor ini membantu dalam bermain game FPS dan Racing.
3. *Sensor Magnetometer* : Memiliki kemampuan untuk mendeteksi medan magnet. Dan, sensor ini digunakan oleh aplikasi Compass untuk mendeteksi arah Kutub Utara dari planet.
4. *Sensor Proximity* : *Proximity* terdiri dari dua elemen, yakni LED dan detektor cahaya. Sensor ini diletakkan dekat dengan lubang suara dari ponselmu. Sensor ini sebagian besar sangat membantu dalam situasi ketika kamu sedang menelepon dan menempelkan ponsel di telingamu dan layar akan mati secara otomatis untuk mencegah terbukanya aplikasi tertentu akibat adanya sentuhan atau gesekan, dan layar akan menyala kembali ketika dijauhkan dari telinga.
5. Sensor cahaya : Sensor ini mendeteksi seberapa terang cahaya di sekitarnya, di mana ponsel diletakkan.
6. *Sensor Barometer* (GPS) : Sensor ini kebanyakan ditemukan dalam ponsel kelas *high end*. Ia akan mendeteksi seberapa tinggi jarak ponsel dari atas permukaan laut. Sensor ini akan memberikan akurasi GPS yang lebih baik

2.2.18.2 Android dan GPS

Saat ini, teknologi berkembang dengan pesat. Dulu ponsel sekedar digunakan untuk menelepon dan SMS saja. Sekarang ponsel sudah mendunia manjadi kotak kecil ajaib yang serba bisa. Salah satunya adalah ponsel dengan sistem operasi Android. Dengan standarisasi fitur dan hardware yang dimiliki, menjadi ponsel Android ponsel canggih yang disukai banyak orang. Tidak lagi canggih karena adanya fitur MMS, radio, atau internet berkecepatan tinggi tapi juga karena ditanamkannya fitur teknologi satelit di dalamnya. Perangkat GPS *receiver* yang dulu besar dan ekslusif, sekarang sudah bisa dimiliki dengan "hanya" membeli sebuah ponsel.

Dengan ponsel berteknologi satelit (GPS), banyak hal yang bisa dilakukan. Ingin melihat di mana posisi user sekarang dalam sebuah peta? Mengambil foto/video yang sudah dilengkapi dengan koordinat? Ingin tahu jalur olahraga bersepeda dengan yang sudah pernah user lalui? Atau ingin tahu dimana posisi ponsel yang hilang dicuri orang? Semua itu bisa dilakukan dengan ponsel Android yang memiliki fitur GPS. Tidak hanya itu, user juga dapat pergi ke tempat wisata tertentu dengan dipandu gambar dan suara dari sebuah ponsel. Bahkan lebih jauh lagi, GPS dapat digunakan untuk membantu memberikan peringatan dini terhadap terjadinya bencana alam.

Sekarang ini banyak sekali pengembang-pengembang aplikasi untuk sistem operasi Android termasuk aplikasi-aplikasi GPS. Yang menyenangkan aplikasi-aplikasi tersebut jenisnya beragam dan jumlahnya pun banyak. **[**10**]**

2.2.18.3 Latitude dan Longitude

Latitude dan Longitude adalah sebuah koordinat yang ada di bumi dan digunakan untuk menentukan kedudukan kita di atas bumi. **[**11**]**

2.2.18.3.1 Latitude

Latitude adalah garis yang melintang dari kutub utara dan kutub selatan. Titik 0 adalah sudut ekuator, tanda + menunjukkan arah ke atas menuju kutub utara, sedangkan tanda – di koordinat Latitude menuju ke kutub slatan. Titik yang dipakai dari 0 ke-90 derajat arah kutub utara, dan 0 ke-90 derajat ke kutub selatan. **[**11**]**

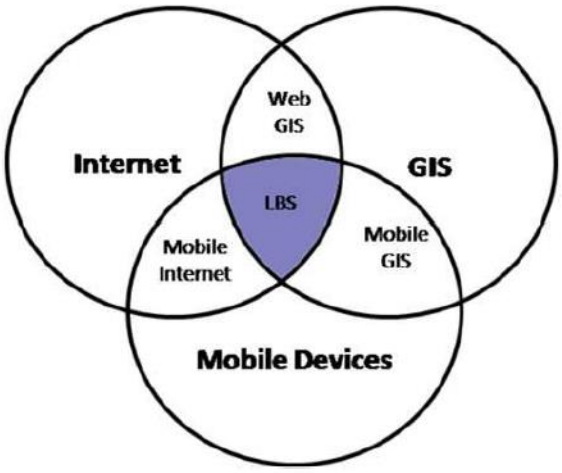
2.2.18.3.2 Longtitude

Longitude adalah garis lintang. Angka dari sudut bundar bumi horizontal. Tiitk diawali dari 0 ke-180 derajat, dan - ke arah sebaliknya. Titik 0 dimulai dari garis negara Inggris. Mengarah ke Indonesia akan menjadi angka positif. Kebalikannya koordinat Logitude minus adalah arah kebalikan. [**11**]

2.2.19 LBS (*Location Based-Service*)

Salah satu ciri mobile *phone* atau tablet pc adala portabilitas, sehingga tidak mengherankan babhwa beberapa fitur Android sangat menarik seperti layanan yang memungkinkan anda menemukan, mencari serta menvisualisasikan posisi kita ke dalam peta lokasi fisik seperti *Google Maps*. Kita dapat membuat peta bebasis *Google Maps* dan menjadikannya sebagai elemen dalam layout *User Interface* (UI) yang kita rancang. Kita dapat melakukan akses penuh ke peta *Google Maps* dan memungkinkan kita untuk mengontrol pengaturan tampilan, mengubah tampilan zoom lokasi, dan memindahkan lokasi tampilan. **[**10**]**

Location Based Service adalah *service* yang berfungsi untuk mencari dengan teknologi *Global Positioning Service* (GPS) dan *Google's cell-based location*. *Map* dan layanan berbasis lokasi menggunakan lintang dan bujur untuk menentukan lokasi geografis, namun sebagai user kita membutuhkan alamat atau posisi *realitime* kita bukan nilai lintang dan bujur. Android menyediakan *geocoder* yang mendukung forward dan *reverse geocoding*. Menggunakan *geocoder*, anda dapat mengkonversikan nilai lintang bujur menjadi alamat dunia nyata atau sebaliknya.



Sumber Gambar: GPS pada Android [**10**]

Gambar 2. 6 Teknologi Location Based Service

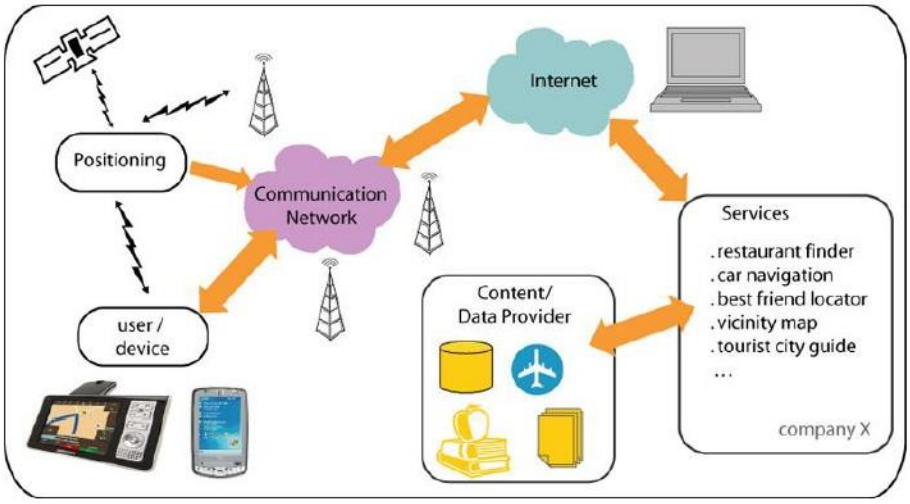
Location based service atau layanan berbasis lokasi adalah istilah umu yang digunakan untuk mengambarkan teknologi yang digunakan untuk menemukan lokasi perangkat yang kita gunakan. Dua unsur utama LBS adalah :

1. *Location Manager* (*API Maps*)

   Menyediakan tools atau resource untuk LBS, *Application Programming Interface* (API) *Maps* menyediakan fasilitas untuk menampilkan, memanipulasi *maps*/peta berserta fitur-fitur lainnya seperti tampilan satelit, *street* (jalan), maupun gabungannya. Paket ini berada pada com.google.android.maps.

2. *Location Providers* (*API Location*)

   Menyediakan teknologi pencarian lokasi yang digunakan oleh *device* atau perangkat. API *Location* berhubungan dengan data GPS dan data lokasi *realtime*. API *Location* berada pada paket Android yaitu dalam paket Androdi *Location*. Dengan *Location Manager*, kita dapat menentukan lokasi kita saat ini. Track gerakan atau perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan. **[**10**]**



Sumber Gambar: GPS pada Android [**10**]

Gambar 2. 7 Cara Kerja LBS

2.2.20 *Google Maps*

Google Maps adalah peta online atau membuka peta secara online, dapat dilakukan secara mudah melalui layanan gratis dari *Google*. Bahkan layanan ini menyediakan API (*Appliction Programming Interface*) yang memunginkan developer lain untuk memanfaatkan aplikasi ini di aplikasi buatannya. Tampilan *Google Maps* pun dapat dipilih, berdasarkan foto asli atau peta gambar rute saja.

Google Maps adalah layanan gratis yang diberikan *Google* dan sangat popular. *Google Maps* adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, *Google Maps* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu *browser*. Kita dapat menambahkan fitur *Google Maps* dalam web yang tlah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan *Google Maps API. Google Maps API* adalah suatu *library* yang berbentuk JavaScript.

Cara membuat *Google Maps* untuk di tampilkan pada suatu web atau blog sangat mudah hanya dengan membutuhkan pengetahuan mengenai HTML serta JavaScript, serta koneksi Internet yang sangat stabil. Dengan menggunakan *Google Maps API*, kita dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga kita dapat fokus hanya pada data-data

yang akan ditampilkan. Dengan kata lain, kita hanya membuat suatu data sedangkan peta yang akan ditampilkan adalah milik *Google* sehingga kita tidak dipusingkan dengan membuat peta suatu lokasi, bahkan dunia.



Sumber Gambar: http://googlemaps.coom [**10**]

Gambar 2. 8 Google Maps

Pada *Google Maps* API terdapat 4 Jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah :

a. *ROADMAP*, untuk menampilkan peta biasa 2 dimensi.
b. *SATELLITE*, untuk menampilkan foto satelit.
c. *TERRAIN*, unutk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai.
d. *HYBRID*, akan menunjukkan foto satelit yang diatasnya tergambar apa yang tampil pada ROADMAP (jalan dan nama kota). **[**10**]**

2.2.21 Android Studio

Android Studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran Android Studio ini di umumkan oleh *Google* pada 16 Mei 2013. Sejak saat itu, Android Studio mengantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android.



Sumber Gambar: http://developer.android.com [**12**]

Gambar 2. 9 Android Studio

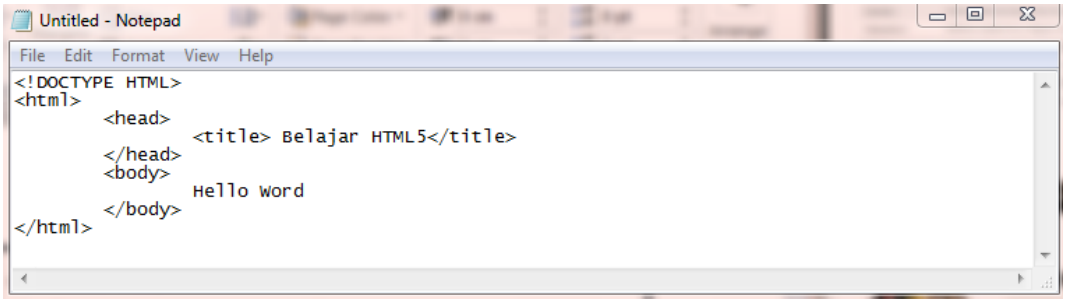
Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (*Android Development Tools*). Android Studi memiliki fitur:

a. Projek berbasis pada *Gradle Build*
b. *Refactory* dan pembenahan bug yang cepat
c. *Tools* baru yang bernama "*Lint*" diklaim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat
d. Mendukung *Proguard And App-signing* untuk keamanan
e. Memiliki GUI aplikasi android lebih mudah
f. Didukung oleh *Google Cloud Platform* untuk setiap aplikasi yang berkembang. **[**12**]**

2.2.22 HTML5

Hyper Text Markup Language (HTML) adalah bahasa dari *Word Wide Web* (WWW) yang dipergunakan untuk menyusun dan membentuk dokumen agar dapat ditampilkan pada program browser. **[**13**] [**9**]** Dengan menggunakan bahasa pemograman HTML yang memiliki tags <> atau markup, dapat membangun struktur dan layout pada halaman web sehingga dapat ditampilkan di browser. HTML5 adalah sebuah bahasa markah untuk menstrukturkan dan menampilkan isi dari Waring Wera Wauna, sebuah teknologi inti dari Internet.

HTML5 adalah pengembangan kelima dari HTML yang pertama pada tahun 1989 oleh Tim Berners-Lee dan keempat HTML4 pada tahun 1997 oleh *Word Wide Web Consortium* (W3C) dan hingga bulan Juni 2011 masih dalam pengembangan. Tujuan utama pengembangan HTML5 adalah untuk memperbaiki teknologi HTML agar mendukung teknologi multimedia terbaru, mudah di baca oleh manusia dan juga mudah dimengerti oleh mesin. HTML5 merupakan salah satu karya Konsorttium Waring Wera Wanua (*Word Wide Web Consortium*, W3C) untuk mendefinisikan sebuah bahasa markah tunggal yang dapat ditulis dengan cara HTML ataupun XHTML. HTML5 merupakan jawaban atas pengembangan HTML 4.01 dan XHTML 1.1 yang selama ini berjalan terpisah dan diimplementasikan secara berbeda-beda oleh banyak perangkat lunak pembuat web. **[**14**] [**9**]** Berikut contoh sintak pada HTML5, yaitu:



Sumber Gambar: HTML5 **[**13**]**

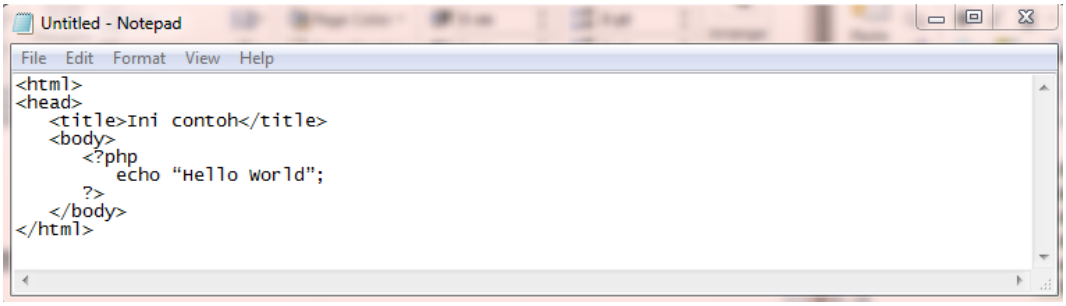
Gambar 2. 10 Contoh Sintak HTML5

2.2.23 PHP

PHP (*Hypertext Preprocessor*) adalah sebuah bahasa pemograman yang berbentuk Scripting, sistem kerja dari program ini adalah sebagai interpreter bukan sebagai compiler. **[**15**]** PHP merupakan bahasa yang hanya dapat berjalan pada server yang hasilnya dapat di tampilkan pada klien. Dalam mengeksekusi kode PHP pada sisi server (disebut server side) berbeda dengan mesin maya Java yang mengeksekusi program pada sisi klien (*client side*). Prosess eksekusi kode PHP yang disisipkan pada halaman HTML.

PHP merupakan bahasa standar yang digunakan dalam dunia website, PHP adalah bahasa pemograman yang berbentuk script yang diletakan di dalam server

web. Sekitar tahun 1994, Rasmus Lerdorf telah meletakkan bersama Per-script untuk membuat siapa yang telah melihat resumenya terkesan. Kemudian sedikit demi sedikit user mulai menyukai script ini. Berikut contoh sintak yang ada pada PHP, yaitu:



Sumber Gambar : Aplikasi Pemograman Web Dinamis dengan PHP dan MySQL [**15**]

Gambar 2. 11 Contoh Sintak PHP

2.2.24 JavaScript

JavaScript pada awalnya diciptakan untuk menangani validasi dari sebuah formulir dalam website yang sebelumnya ditugaskan pada bahasa pemograman server seperti *Perl*. Namun dengan berkembangnya zaman JavaScript menjadi pemain penting dalam perkembangan teknologi website.

JavaScript adalah bahasa scrip berdasarkan pada objek yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen HTML. **[**9**]** Java dibuat untuk berinterkasi dengan halaman *website* dan dibuat dengan 3 bagian :

1. *ECMAScript*, yang didefinisikan dalam ECMA-262 dan menjadi bagian utama dalam menjalankan fungsionalitas.
2. *Document Object Model* (DOM), yang menyediakan *method* dan *interface* untuk bekerja dengan isi dari halaman website.
3. *Browser Object Model* (BOM), yang menyediakan *method* dan *interface* untuk berinteraksi dengan browser.

2.2.25 JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data. **[**16**]**

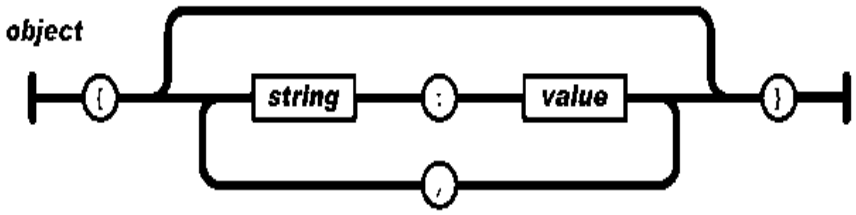
JSON terbuat dari dua struktur, yaitu:

a. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
b. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemprograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemprograman yang juga berdasarkan pada struktur data ini.

JSON menggunakan bentuk sebagai berikut:

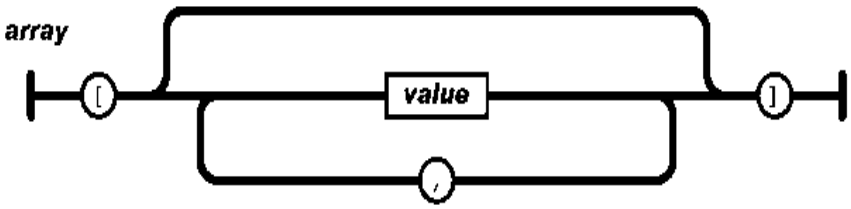
Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).



Sumber Gambar: Kolaborasi Dahsyat Android dengan PHP & MySQL [**16**]

Gambar 2. 12 Objek JSON

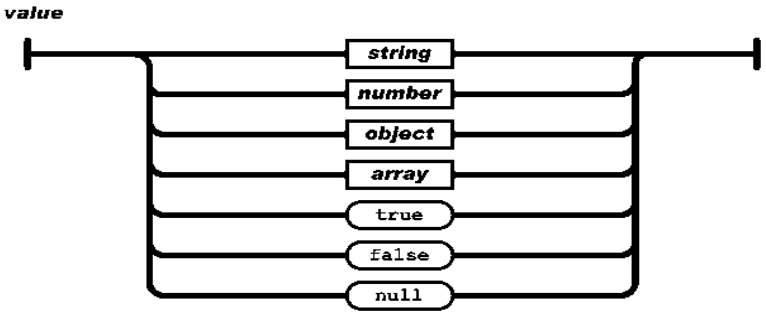
Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).



Sumber Gambar: Kolaborasi Dahsyat Android dengan PHP & MySQL [**16**]

Gambar 2. 13 Array JSON

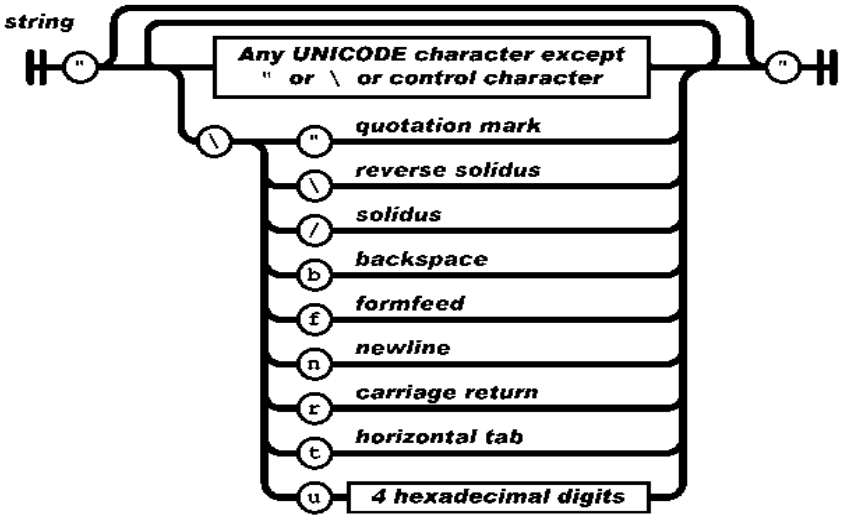
Nilai (*value*) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau *true* atau *false* atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.



Sumber Gambar: Kolaborasi Dahsyat Android dengan PHP & MySQL [**16**]

Gambar 2. 14 Value JSON

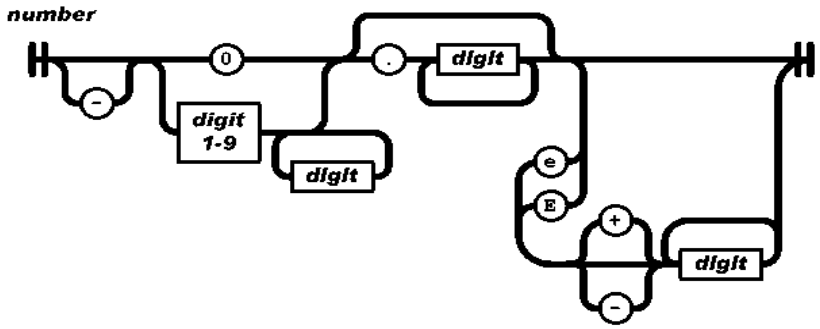
String adalah kumpulan dari nol atau lebih karakter *Unicode*, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan *backslash escapes* "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.



Sumber Gambar: Kolaborasi Dahsyat Android dengan PHP & MySQL [**16**]

Gambar 2. 15 String JSON

Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.



Sumber Gambar: Kolaborasi Dahsyat Android dengan PHP & MySQL [**16**]

Gambar 2. 16 Number JSON

Spasi kosong (*whitespace*) dapat disisipkan di antara pasangan tanda-tanda tersebut, kecuali beberapa detil encoding yang secara lengkap dipaparkan oleh bahasa pemprograman yang bersangkutan. **[**16**]**

2.2.26 MySQL

Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. MySQL adalah sebuah perangkat lunak sistem manajemen basis data dikenal dengan DBMS (*Database Management System*), database ini multithread, multiuser. MySQL adalah implementasi dari sistem manajemen basis data relasional (RDBMS). MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah database berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris. MySQL AB merupakan perusahaan komersial Swedia yang mensponsori dan memiliki MySQL.

Pendiri MySQL AB adalah dua orang Swedia yang bernama David Axmark, Allan Larsson dan satu orang Firlandia bernama Michael "Monty". Setiap pengguna MySQL dapat menggunakannya secara bebas yang didistribusikan gratis dibawah lisensi GPL (*General Public License*) namun tidak boleh menjadikan produk turunan yang bersifat komersial.

MySQL merupakan database server yang sangat terkenal di dunia, semua itu tak lain karena bahasa yang digunakan untuk mengakses database yaitu SQL. SQL (*Structured Query Language*) pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset Sn Jose, IBM yang bernama sistem R. Kemudian SQL juga dikembangkan oleh Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses pengaksesan database lebih *user-friendly* dibanding dengan yang lain, misalnya dBase atau Clipper karena mereka masih menggunakan perintah-perintah pemrograman murni. SQL dapat digunakan secara berdiri sendiri maupun dilekatkan pada bahasa pemograman seperti C dan Delphi.

Elemen dari SQL yang paling dasar antara lain pernyataan, nama, tipe data, ekspresi, konstanta dan fungsi bawaan. Pernyataan perintah dari SQL yang digunakan untuk meminta sebuah tindakan kepada DBMS.

Sistem database menyediakan bahasa untuk mendefinisikan database *Data Definition Language* (DDL) serta bahasa untuk memanipulasi database *Data*

Manipulation Language (DML) untuk melakukan operasi-operasi tertentu pada database dan juga *Data Control Language* (DCL).

1. *Data Definition Language* (DDL)
   Kumpulan perintah SQL yang digunakan untuk membangun database, bias juga memiliki arti lain perintah untuk membuat struktur dasar database dan table. Perintah-perintah yang termasuk DDL adalah *statement connect, create,show, drop, rename* dan *alter*.
2. *Data Manipulation Language* (DML)
   Kumpulan statement yang digunakan sebagai perintah untuk mengelola data dalam sebuah table. Perintah DML digunakan setelah menggunakan DDL antara lain seperti *statement insert, update, delete,* dan *select*.
3. *Data Control Language* (DCL)
   Perintah yang digunakan untuk keperluan keamanan (*security*) database dengan membuat hak akses tertentu bagi setiap user. Statement pada DCL antara lain *grant, revoke, set* dan *lock table.*

SQL adalah sebuah konsep pengoperasian database terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomastis. Sintak SQL pada MySql:

1. *CREAT TABLE*
   Digunakan untuk membuat table baru
   Sintak: **CREAT TABLE** namatabel (namakolom typekolom(*size*), namakolom typekolom(*size*), …);
2. *SELECT*
   Berfungsi untuk menampilkan data dari table, sintak dasarnya seperti berikkut:
   Sintak: **SELECT** namakolom,namakolom,… FROM namatabel **WHERE** kriteria;
3. *UPDATE*
   Digunakan untuk memperbaiki informasi/data pada table

Sintak: **UPDATE** namatabel **SET** namakolom=value **WHERE** kriteria;

4. *DELETE*

   Untuk menghapus record/data dari table

   Sintak: **DELETE FROM** namatabel **WHERE** kriteria;

5. *INSERT INTO*

   Untuk menambah informasi/data baru kedalam table

   Sintak: **INSERT INTO** namatabel(kolom) **VALUES** (‘data1’,data2,…);

6. *CREATE DATABASE*

   Digunakan untuk membuat database baru

   Sintak: **CREATE DATABASE** nama_database; (nama database bebas berdasarkan kasus).

7. *DROP DATABASE*

   Digunakan untuk menghapus database

   Sintak: **DROP DATABASE** nama_database; (nama database bebas berdasarkan kasus).

8. *DROP TABLE*

   Digunakan untuk menghapus tabel

   Sintak: **DROP TABLE** nama_table;

9. *RENAME*

   Untuk mengganti nama pada tabel

   Sintak: **RENAME**namatabel yang diganti **TO** namatabelbaru;

10. *DESC TABLE*

    Deskripsi tabel atau kolom

    Sintak: **DESCRIBE** | desc nama_table [nama_kolom]

11. *ALTER TABLE and ADD*

    Untuk menambahkan kolom/field

    Sintak: **ALTER TABLE** namatabel **ADD** namakolom typekolom(size);

12. *ALTER TABLE and CHANGE*

    Untuk mengganti nama kolom/field

    Sintak: **ALTER TABLE** namatabel **CHANGE** namakolomyangdiganti namakolombaru typekolombaru(size);

13. *ALTER TABLE and MODIFY*

    Untuk mengganti type atau size pada kolom atau field

    Sintak: **ALTER TABLE** namatabel **MODIFY** namakolom typekolom(size);

14. *GRANT*

    Memberikan akses kepada user terhadap table, bisa juga digunakan untuk membuat user baru

    Sintak: **GRANT** (all privileges | (select, insert, update, delete, references(column list), usage)) on table to user, … [with grant option]

15. *LOCK TABEL*

    Untuk menutup/mengunci akses user pada table

    Sintak: **LOCK TABLE** nama_table [as alias] read | write [table_name read | write]

16. *UNLOCK TABLE*

    Untuk membuka akses pada table yang sebelumnya dikunci

    Sintak: **UNLOCK** nama_table

17. *LOAD DATA INIFILE*

    Perintah untuk membaca data dari file teks. Sintak: load data inifile syntax. **[**16**] [**15**]**

2.2.27 UML (*Unified Modeling Language*)

Menurut Nugroho(2010:6) **[**17**]**, "UML (*Unified Modeling Language*) adalah perangkat lunak yang berparadigma "berorientasi objek". Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami".

Menurut Sibero **[**18**]**, "UML adalah bahasa yang digunakan untuk mem-visualisasikan, mendefinisikan, membangun dan membuat dokumen dari

arsitektur perangkat lunak. UML dapat digunakan pada semua proses melalui metodologi pengembangan perangkat lunak dan melakukan implementasinya pada teknologi yang berbeda".

Dengan menggunakan UML dapat membantu tim dari sebuah proyek untuk berkomunikasi, memeriksa potensi rancangannya dan menyetujui arsitektur rancangan dari proyek software tersebut. *Unified Modeling Language* mempunyai beberapa tujuan, yaitu:

a. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untukmengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
b. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
c. Menyatukan praktek-praktek terbaik yang terdapat dalam permodelan.

Konsep OOAD (*Object Oriented Analysis Design*) mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu:

1. Analisis berorientasi objek (OOP)

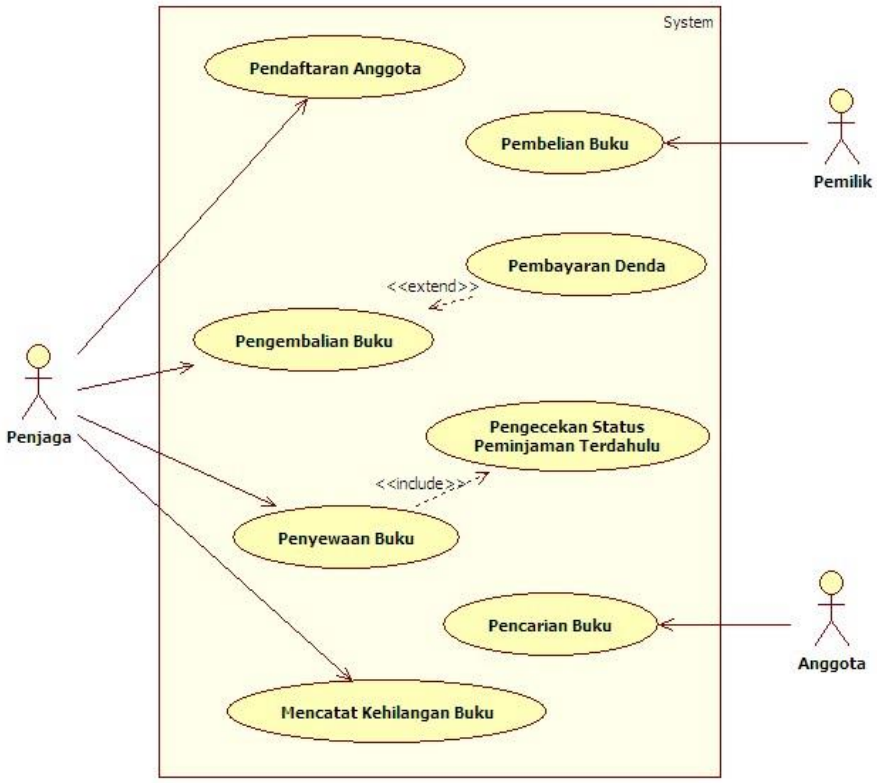
   Metode analisis yang memeriksa *requirement* (syarat/keperluan) yang harus dipenuhi sebuah sistem dari sudur pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. *Unified Modeling Language* (UML) adalah bahasa untuk mengspesifikasi, memvisualisasi, membangun dan mendokumentasi *artefacts* (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak), seperti pada pemodelan bisnis dan sistem non-perangkat lunak lainnya. UML merupakan bahasa standar untuk penulisan blueprint software yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak.

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan

untuk mendesain secara cepat. Beriktu ini beberapa bagian UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan sejumlah eksternal aktor dan hubungannya ke *use case* yang berikan oleh sistem.

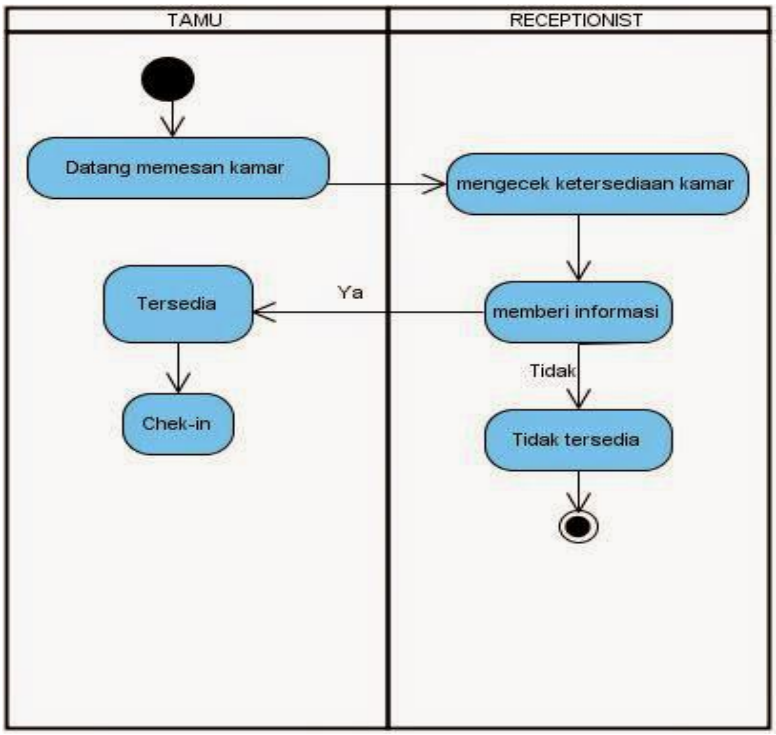


Sumber Gambar: UML for Java Programmers [**19**]

Gambar 2. 17 Use Case Diagram

2. *Activity Diagram*

Activity diagram menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainnya seperti *use case*.

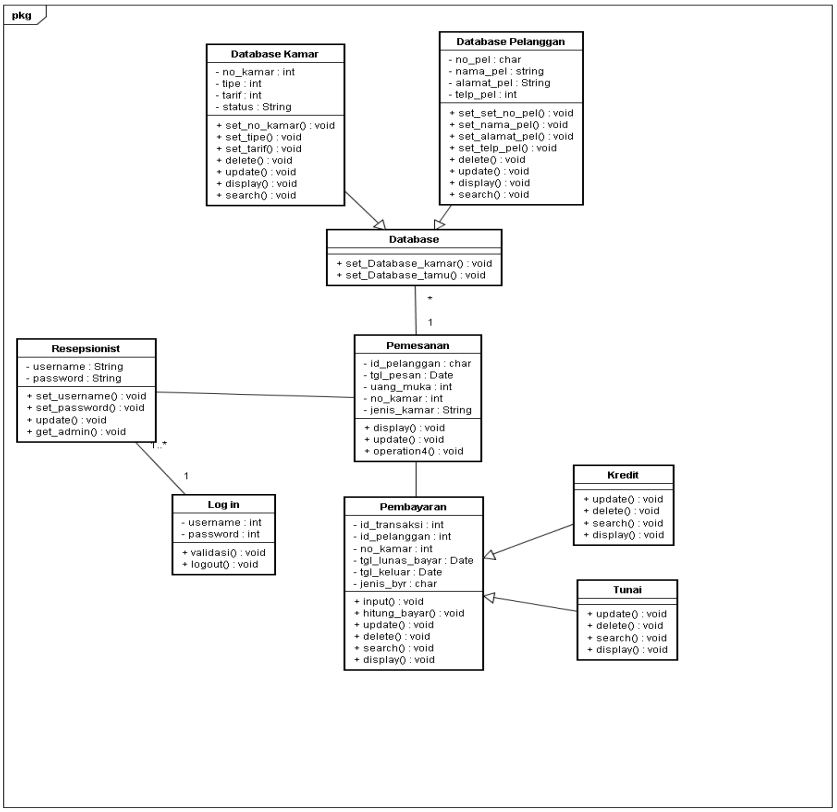


Sumber Gambar: UML for Java Programmers [**19**]

Gambar 2. 18 Activity Diagram

3. *Class Diagram*

Class diagram menggambarkan struktur dan deskripsi *class*, *package* (paket) dan objek berserta hubungan satu sama lain seperti *containment* (penahanan), pewarisan, *asosiasi* dan lain-lain.

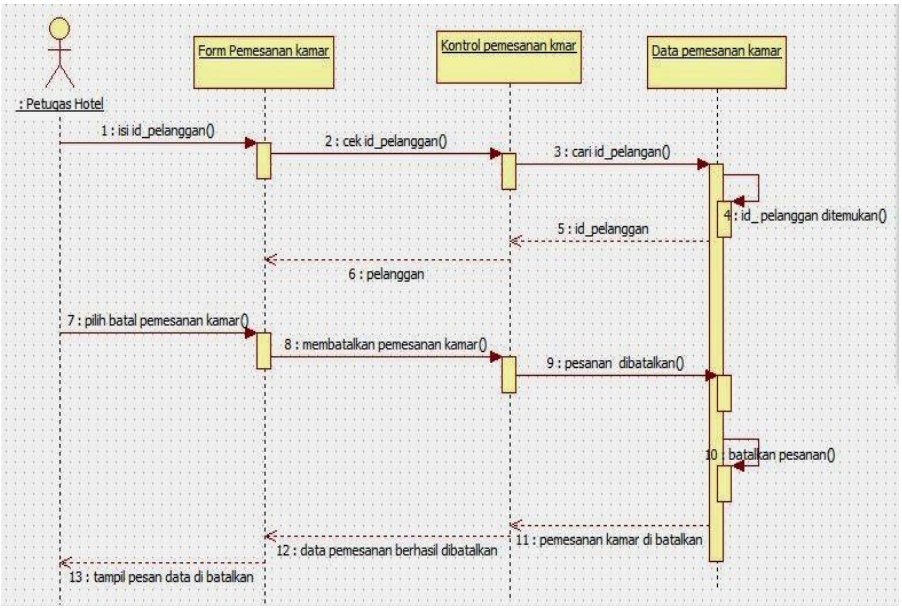


Sumber Gambar: UML for Java Programmers [**19**]

Gambar 2. 19 Class Diagram

4. *Sequance Diagram*

Sequence diagram menggambarkan kolabirasi dinamis antara sejumlah objek dan untuk memajukan rangkaian pesan yang dikirim antara objek juga interaksi. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. **[**20**]**



Sumber Gambar: UML for Java Programmers [**19**]

Gambar 2. 20 Sequence Diagram

2.2.28 Web Server

Web Server adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan web browser dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk HTML. Web server yang terkenal diantaranya adalah *Apache* dan *Microsoft Internet Information Service* (IIS). **[**21**]**

2.2.29 Web Service

Web Service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. Web service digunakan sebagai suatu fasilitas yang disediakan oleh suatu web site untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan web service. Web service menyimpan data informasi dalam format XML sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa compiler.

Web service bertujuan untuk meningkatkan kolaborasi antar pemogram dan perusahan, yang memungkinkan sebuah fungsi di dalam Web Service dapat

dipinjam oleh aplikasi lain tanpa perlu mengetahui detil pemograman yang terdapat di dalamnya.

Beberapa alasan menggapa digunakannya Web Service adalah sebagai berikut:

1. Web Sevice dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. Web Service memiliki kemudahan dalam proses deploymentnya karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. Web Service cukup di-upload ke web server dan siap diakses oleh pihak-pihak yang telah diberikan orientasi.
3. Web service berjalan di port 80 yang merupakan protocol standar HTTP, dengan demikian web service tidak memerlukan konfigurasi khusus di sisi firewall. **[**22**]**

2.2.30 Teori Kuesioner

Kuesioner merupakan teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk dijawabnya. Kuesioner merupakan teknik pengumpulan data yang efisien bila peneliti tahu dengan pasti variable yang akan diukur dan tahu apa yang bisa diharapkan dari responden. Selain itu, kuesioner juga cocok digunakan bila jumlah responden cukup besar dan tersebar diwilayah yang luas. Kuesioner dapat berupa pertanyaan atau pernyataan tertutup atau terbuka, dapat diberikan kepada responden secara langsung atau dikirim melalui POS atau internet.

Bila penelitian dilakukan pada lingkup yang tidak terlalu luas, sehingga kuesioner dapat diantarkan langsung dalam waktu yang tidak terlalu lama, maka tidak perlu dikirm melalui POS. Dengan adanya kontak langsung antara peneliti dengan responden akan menciptakan suatu kondisi yang cukup baik, sehingga responden dengan sukarela akan memberikan data obyektif dan cepat.

Uma Sekaran (1992) dalam buku Prof. Dr. Sugiyono [13] mengemukakan beberapa prinsip dalam penulisan angket sebagai teknik pengumpulan data yaitu: prinsip penulisan, pengukuran, dan penampilan fisik.

1. Prinsip Penulisan Angket

Prinsip ini menyangkut beberapa faktor yaitu : isi dan tujuan pertanyaan, bahasa yang digunakan mudah, pertanyaan terbuka terutup-negatif positif, pertanyaan tidak mendua, tidak menanyakan hal-hal yang sudah lupa, pertanyaan tidak mengarahkan, panjang pertanyaan dan urutan pertanyaan.

2. Prinsip Pengukuran

Angket yang diberikan kepada responden adalah merupakan instrumen penelitian, yang digunakan untuk mengukur variabel yang akan diteliti. Oleh karena itu instrument angket tersebut harus dapat digunakan untuk mendapatkan data yang valid dan reliabel tentang variabel yang diukur.

3. Penampilan Fisik Angket

Penampilan fisik angket sebagai alat pengumpulan data akan mempengaruhi respon atau keseriusan responden dalam mengisi angket. Angket yang dibuat di kertas buram, akan mendapat respon kurang menarik bagi responden, bila dibandingkan dengan kertas yang bagus dan berwarna. Tetapi angket yeng dicetak di kertas yang bagus berwarna akan menjadi mahal dalam pembiayaan.

2.2.30.1 Skala Likert

Skala likert [13] digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dalam penelitian fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti yang selanjutnya disebut sebagai variabel penelitian. Dengan skala *Likert*, maka variabel yang akan diukur dijabarkan menjadi indikator variabel.

Kemudian indicator tersebut dijadikan sebagai titik tolak untuk menyusun item item instrumen yang dapat berupa pernyataan atau pertanyaan.Jawaban setiap item instrumen yang menggunakan skala *Likert* mempunyai gradasi dari sangat positif sampai sangat negatif, yang berupa kata-kata antara lain :

1. sangat setuju	1. setuju
2. setuju	2. sering

3. ragu-ragu	3. kadang-kadang
4. tidak setuju	4. hamper tidak pernah
5. sangat tidak setuju	5. tidak pernah
1. sangat positif	1. baik sekali
2. positif	2. cukup baik
3. netral	3. kurang baik
4. negaif	4. sangat tidak baik
5. sangat negative	

Untuk keperluan analis kuantitatif, maka jawaban itu dapat diberi skor, misalnya:

1. Setuju/selalu/sangat positif diberi skor	5
2. Setuju/sering/positif diberi skor	4
3. Ragu-ragu/kadang-kadang/netral diberi skor	3
4. Tidak setuju/hampir tidak pernah/negative diberi skor	2
5. Sangat tidak setuju/tidak pernah/sangat positif diberi skor	1

Instrumen dalam penelitian yang menggunakan metoda skala *Likert* dapat dibuat pertanyaan atau pernyataan-nya dalam bentuk checklist ataupun pilihan ganda

a. Contoh bentuk checklist

Berilah jawaban pertanyaan berikut sesuai engan pendapat anda, dengan cara memberi tanda (✔) pada kolom yang tersedia

No.	Pertanyaan	Jawaban				
		SS	ST	RG	TS	STS
1. 2.	Apakah anda setuju, sistem informasi yang dibuat ini mampu memudahkan dan mengatasi masalah yang ada sebelumnya ……		✔			

Keterangan :

SS = Sangat setuju

ST = Setuju

RG = Ragu-ragu

TS = Tidak setuju

STS = Sangat tidak setuju

Bila :

SS diberi skor = 5

ST diberi skor = 4

RG diberi skor = 3

TS diberi skor = 2

STS diberi skor = 1

Kemudian dengan teknik pengumpulan data angket, maka instrumen tersebut diberikan kepada 100 orang pegawai yang diambil secara acak. Dari 100 orang pegawai setelah dilakukan analisi misalnya

25 orang pegawai menjawab SS
40 orang pegawai menjawab ST
5 orang pegawai menjawab RG
20 orang pegawai menjawab TS
10 orang pegawai menjawab STS

Berdasarkan data tersebut 65 orang atau 65% pegawai menjawab setuju dan sangat setuju. Jadi kesimpulannya mayoritas pegawai setuju dengan adanya sistem informasi.

Data tersebut juga dapat dianalisis berdasarkan skoring setiap jawaban dari responden. Berdasarkan skor yang telah ditetapkan maka

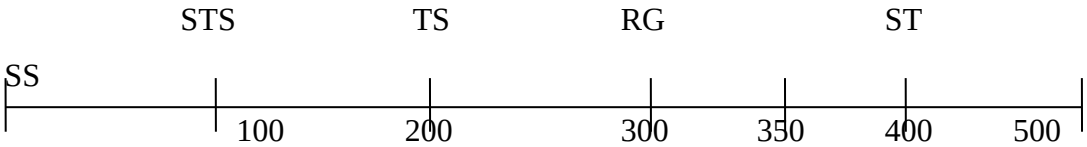
Jumlah skor untuk 25 pegawai menjawab SS = 25 x 5 = 125
Jumlah skor untuk 40 pegawai menjawab ST = 40 x 4 = 160
Jumlah skor untuk 5 pegawai menjawab RG = 5 x 3 = 15
Jumlah skor untuk 20 pegawai menjawab TS = 20 x 2 = 50

Jumlah skor untuk 10 pegawai menjawab STS	= 10 x 1 = 10
Jumlah	= 350

Jumlah skor ideal (kriterium) untuk seluruh item = 5 x 100 = 500 (SS) skor tertinggi jumlah skor rendah 1 x 100 = 100 (STS).

Jadi berdasarkan data itu maka tingkat persetujuan terhadap sistem informasi yang dibuat = (350 : 500) x 100% = 70%

Secara kontinum dapat digambarkan seperti berikut:



Jadi berdasarkan data yang diperoleh dari 100 responden maka data 350 terletak pada daerah setuju.

Bila berdasarkan pada kelompok responden, maka dapat diketahui bahwa :

1. 25 prosen menyatakan sangat setuju (25/100 = 25%)
2. 40 prosen menyatakan setuju (40/100 = 40%)
3. 5 prosen menyatakan ragu-ragu (5/100 = 5%)
4. 20 prosen menyatakan tidak setuju (20/100 = 20%)
5. 10 prosen menyatakan sangat tidak setuju (10/100 = 10%)

b. Contoh bentuk pilihan ganda

Berilah satu jawaban terhadap pertanyaan berikut sesuai dengan pendapat anda, dengan member tanda lingkaran pada nomor jawaban yang tersedia.

1. Apakah anda setuju, sistem informasi yang dibuat mampu memudahkan dan mangatasi masalah yang ada sebelumnya?
   a. Sangat tidak setuju
   b. Tidak setuju
   c. Ragu-ragu
   d. Setuju
   e. Sangat setuju

Akan tetapi dengan bentuk pilihan ganda itu, maka jawaban dapat diletakkan pada tempat yang berbeda-beda. Semisal untuk jawaban diatas "sangat setuju" diletakkan di paling akhir untuk selanjutnya jawaban "sangat tidak setuju diletakkan di paling awal. Dalam penyusunan instrumen untuk variabel tertentu, sebaiknya butir butir pertanyaan dibuat dalam bentuk kalimat positif, netral, ataupun negative sehingga responden dapat menjawab dengan serius dan konsisten.

Dengan cara demikian maka kecendrungan responden untuk menjawab kolom tertentu dari bentuk checklist dapat dikurangi. Dengan model ini juga selalu membaca pertanyaan setiap item instrumen dan juga jawabannya. Pada bentuk *checklist*, sering jawaban tidak dibaca, karena letak jawaban sudah menentu. Tetapi dengan bentuk *checklist* juga akan didapat keuntungan dalam hal ini singkat dalam pembuatannya, hemat kertas, mudah mentabulasikan data, dan secara visual lebih menarik. Data yang diperoleh dari skala tersebut adalah berupa interval.