BAB 2
TINJAUAN PUSTAKA

2.1 Profil Disporbudpar Kota Cirebon

Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon atau bisa disingkat Disporbudpar kota Cirebon adalah dinas yang bergerak dalam bidang olahraga, budaya dan pariwisata di kota Cirebon yang beralamat di Jalan Brigjen Dharsono no. 5 By Pass Cirebon dengan nomor telepon/fax (0231) 486856 / (0231) 486856.

2.1.1 Sejarah Disporbudpar Kota Cirebon

Sejarah Singkat Berdirinya Dinas Kebudayaan dan Pariwisata. Pada Tahun 1962-1967 bidang Pariwisata ditangani oleh Bidang Kebudayaan dari Depertemen Pendidikan dan Pariwisata, tetapi khusus untuk Kotamadya DT II Cirebon di bentuk Pariwisata daerah yang berdiri sendiri dan berlokasi di. Jl. Siliwangi dan berada dibawah naungan Departemen Pendidikan dan Kebudayaan. Pada tahun 1974, berdasarkan Intruksi Gubernur Kaparda diubah menjadi Dinas Kebudayaan dan Pariwisata dan Kepala Dinasnya dijabat oleh Alumni dari Akademi Pariwisata yaitu Arie Muslihudin, BA. Dan masih berjalan secara non struktural, kemudian berdasarkan peraturan Daerah No. 5 tahun 1983 dan Perubahan Kembali mengenai 12 Urusan Daerah Tingkat I yang sepenuhnya diserahkan kepada Kepala Daerah Tingkat II. Pembukuan dan Kewenangan ini berjalan sampai sekarang.

Pada tahun 1983 Kepala Dinas Kota Cirebon dijabat oleh Drs. Enjang Sudarsono. Tahun 1983 dijabat oleh Drs. Sukarno. Pada 1988 Kepala Dinas dijabat oleh H. Arie Muslihudin yang ketika dikantor Diparda masih di jalan Siliwangi No. 100.

Pada tahun 1989 Kantor Diparda bertempat di Jl. Tentara Pelajar No. 99 Cirebon. Pada Tahun 1991 Kantor Diparda DT II Cirebon pindah ke Jl. Brigjend Dharsono No. 5 By Pass Cirebon pada waktu itu dikepalai oleh Drs. H. Sri Kuncoro, MBA. Pada tahun 2001 Kantor Diparda di jabat oleh Muhammad Irsad Sidik, SH. Kemudian diganti oleh Dra. Tati Suryawati, M.SI.

Pada tahun 2002 Kantor Diparda dijabat oleh Drs. Janto Subijanto, MM, MBA. Berdasarkan Peraturan Daerah Kota Cirebon No. 6 tahun 2001 (Lembaran Daerah Tahun 2001 No. 8, Lembaran Daerah tahun 2001 No. 5) Bahwa nama Diparda diubah menjadi Dinas Kebudayaan dan Pariwisata yang bertempat di Jl. Brigjend Dharsono No. 5 By Pass Cirebon.

Pada tahun 2005 Kepala Dinas Kebudayaan dan Pariwisata Kota Cirebon dijabat oleh Drs. Wawan Sopyan, MSI. Dan pada tahun itu pula beliau pindah ke Dinas Kebudayaan dan Pariwisata Provinsi Jawa Barat, selanjutnya posisinya dijabat oleh PLT. Drs. Dedi Ucu Safrudin. Pada Desember 2005 telah dilantik Kepala Dinas Kota Cirebon yang Definitif yaitu Moch. Hanafiah, SH, MH. Dan pada tahun 2008 Kepala Dinas Kebudayaan dan Pariwisata dijabat oleh Dr. Wahyo, M.Pd..

Berdasarkan Peraturan Pemerintah No. 41 Tahun 2007 tentang Organisasi Perangkat Daerah dan Peraturan Daerah No. 14 Tahun 2008, beban kerja dinas ditambah dengan bergabungnya Bidang Kepemudaan dan Keolahragaan yang semula bernaung di bawah Dinas Pendidikan Kota Cirebon. Pada awal tahun 2009 Dinas Kebudayaan dan Pariwisata Kota Cirebon akhirnya berganti menjadi Dinas Pemuda, Olahraga, Kebudayaan dan Pariwisata Kota Cirebon.

2.1.2 Visi dan Misi Disporbudpar Kota Cirebon

Visi dan misi dari Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon adalah sebagai berikut:

2.1.2.1 Visi

Menyejahterakan Masyarakat Kota Cirebon Melalui Pemberdayaan Pemuda, Olahraga, Budaya dan Pariwisata.

2.1.2.2 Misi

1. Meningkatkan Kualitas SDM Bidang pemuda, olahraga, kebudayaan dan pariwisata

2. Memberdayakan pemuda melalui pembinaan moral, wawasan kebangsaan, kepemudaan dan profesionalisme untuk membentuk jiwa yang mandiri dan produktif
3. Meningkatkan pembinaan dan fasilitas olahraga untuk menghasilkan olahraga yang berprestasi.
4. Melestarikan dan mengembangkan kebudayaan sebagai obyek daya tarik wisata serta memperkokoh jati diri bangsa.
5. Mengembangkan potensi pariwisata dan pemasaran produk pariwisata sebagai upaya meningkatkan kunjungan wisata yang menunjang Pendapatan Asli Daerah (PAD) dan daya beli masyarakat.

2.1.3 Logo Disporbudpar Kota Cirebon

Logo Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon tempat penelitian dilaksanakan dapat dilihat pada Gambar 2.1.



Gambar 2.1 Logo Disporbudpar Kota Cirebon

Logo Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon yang terdapat pada Gambar 2.1 menggambarkan karakteristik dari kota CIrebon tempat penelitian dilaksanakan. Makna dari logo Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon yaitu :

A. Bentuk Lambang Daerah

Lambang daerah terdiri dari 3 (tiga) bagian yaitu :

1. Bagian atas berupa sebuah pita yang bertuliskan Kota Cirebon

2. Bagian dalam berupa sebuah perisai yang didalamnya terdapat gambar sebagai berikut :
    a. Bagian atas berupa sebuah daun jati dan sembilan bintang
    b. Bagian tengah berupa garis bergerigi sembilan buah
    c. Bagian bawah berupa lukisan laut berombak dan gambar udang rebon.
3. Bagian bawah berupa sebuah pita yang bertuliskan Gemah Ripah Loh Jinawi.

B. Tata Warna Lambang Daerah

Tata warna lambing daerah adalah sebagai berikut :

1. Warna dasar perisai
    a. Perisai bagian atas berwarna kuning emas
    b. Perisai bagian bawah berwarna putih
2. Isi Perisai
    a. Daun jati berwarna hijau tua
    b. Lukisan laut berombak berwarna biru
    c. Gambar udang rebon berwarna kuning emas
    d. Garis bergerigi sembilan buah berwarna hitam
    e. Sembilan bintang berwarna putih
3. Warna dasar lambang adalah berwarna hitam yang menghiasi perisai dan pita.

C. Arti dari Lambang Daerah

Lambang daerah yang dilukiskan dalam tata warna sebagai mana yang tertuang dalam Peraturan Daerah No.2 Tahun 1989 sebagai berikut :

1. Daun jati yang berwarna hijau tua, mengandung arti bahwa pada zaman dahulu di Cirebon ada seorang pemimpin para wali yang berbudi luhur dan bertahta serta disemayamkan di Gunungjati dengan nama Syarif Hidayatullah atau Sunan Gunungjati yang menyebarkan Agama Islam di tanah Jawa.

2. Sembilan buah bintang berwarna putih, mengandung arti Walisanga. Kota Cirebon terkenal sebagai tempat berkumpulnya para wali untuk bermusyawarah dalam hubungannya dengan ilmu Agama Islam yaitu :
    a. 4 (empat) buah bintang diatas dasar kuning emas menggambarkan ilmu syariat, hakekar, terekat dan ma'rifat.
    b. 5 (lima) buah bintang di dalam gambar daun jati menggambarkan rukun Islam, yaitu syahadat, sholat, zakat, puasa dan haji.
3. Lukisan laut berombak berwarna biru, mengandung arti bahwa masyarakat Kota Cirebon mempunyai kegiatan bekerja di daerah pantai (nelayan), dengan penuh keikhlasan (jalur putih) dalam menunaikan kewajiban masing-masing untuk kepentingan bangsa dan Negara
4. Gambar udang rebon berwarna kuning emas, mengandung arti bahwa hasil laut telah memberikan kemakmuran kepada masyarakat Cirebon. Adapun udang rebon merupakan bahan baku untuk pembuatan terasi yang terkenal dari Kota Cirebon
5. Garis bergerigi sembilan buah berwarna hitam yang melukiskan benteng yang mendatar berpuncak sembilan buah, menggambarkan arti bahwa Kota Cirebon bercita-cita melaksanakan pembangunan di segala bidang/sektor di seluruh kotanya untuk kemakmuran rakyat.
6. Perisai yang bersudut lima, mengandung arti bahwa perjuangan dalam mempertahankan dan menegakkan Negara Kesatuan Republik Indonesia yang diprolamasikan tanggal 17 Agustus 1945.
7. Warna dasar kuning emas perisai bagian atas melambangkan Kota Cirebon sebagai kota pantai yang bercita-cita melaksanakan pembangunan untuk mewujudkan masyarakat yang tertib, tentram, adil dan makmur.
8. Warna putih pada perisai bagian bawah melambangkan Kota Cirebon letaknya di pinggir laut atau Kota Pantai yang siap sedia (jalur biru) memberikan hasil laut yang berguna dan berharga bagi kehidupan rakyatnya

9. Pita melingkari perisai dengan warna kuning melambangkan persatuan, kebesaran dan kejayaan.
10. Dasar lambang yang berwarna hitam melambangkan keabadian.

2.1.4 Dasar Hukum Pelaksanaan Tugas

Peraturan Daerah Nomor : 14 tahun 2008, sebagaimana diubah dengan Peraturan Daerah Nomor : 13 Tahun 2011 tentang Dinas – dinas Daerah pada Pemerintah Kota Cirebon dan Peraturan Walikota Cirebon Nomor : 6 Tahun 2012 tentang Organisasi dan Tata Kerja Dinas Pemuda, Olahraga, Kebudayaan dan Pariwisata Kota Cirebon.

2.1.5 Tugas Pokok dan Fungsi SKPD

TUGAS POKOK : Melaksanakan urusann Pemerintah Daerah berdasarkan asas otonomi dan tugas pembatuan bidang pemuda dan olahraga serta bidang kebudayaan dan bidang pariwisata.

FUNGSI :

1. Perumusan kebijakan teknis bidang pemuda dan olahraga serta bidang kebudayaan dan bidang pariwisata
2. Penyelenggaraan urusan pemerintahan dan pelayanan umum bidang pemuda dan olahraga serta bidang kebudayaan dan bidang pariwisata
3. Pembinaan dan pelaksanaan tugas bidang pemuda dan olahraga serta bidang kebudayaan dan pariwisata
4. Pelaksanaan tugas lain yang diberikan oleh Walikota sesuai dengan tugas dan fungsinya

2.1.5.1 Struktur Organisasi Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon

Struktur organisasi merupakan pengelompokkan pekerjaan berdasarkan jabatan yang dimiliki oleh masing-masing orang di sebuah pemerintahan. Setiap pemerintahan memiliki struktur organisasi yang berbeda-beda disesuaikan dengan

bidangnya, salah satunya adalah Dinas Pemuda Olahraga Kebudayaan dan Pariwisata Pemerintah Kota Cirebon. Struktur organisasi yang terdapat di Disporbudpar kota Cirebon adalah sebagai berikut :

- **Kepala Dinas**, membawahkan:

1. Sekretaris, membawahkan :
    a. Sub Bagian Umum
    b. Sub Bagian Program dan Pelaporan
    c. Sub Bagian Keuangan
2. Kepala Bidang Kepemudaan, membawahkan :
    a. Seksi Bina Kelembagaan dan Kemitraan Pemuda
    b. Seksi Bina Kepemimpinan dan Kewirausahaan Pemuda
3. Kepala Bidang Keolahragaan, membawahkan :
    a. Seksi Bina Olahraga Prestasi
    b. Seksi Bina Olahraga Rekreasi dan Masyarakat
4. Kepala Bidang Kebudayaan, membawahkan :
    a. Seksi Bina Seni, Film, Bahasa dan Sastra
    b. Seksi Bina Nilai Tradisional, Sejarah dan Kepurbakalaan
5. Kepala Bidang Kepariwisataan, membawahkan :
    a. Seksi Pemasaran Pariwisata
    b. Seksi Bina Usaha Pariwisata
6. Unit Pelaksana Teknis Daerah (UPTD)
    1. Kepala UPTD Sarana Prasarana Pemuda dan Olahraga, membawahkan :
        a. Kasubag Tata Usaha
    2. Kepala UPTD Pembibitan dan Pengembangan Olahraga, membawahkan :
        a. Kasubag Tata Usaha
    3. Kepala UPTD Pelayanan Informasi Budaya dan Pariwisata, membawahkan :
        a. Kasubag Tata Usaha

2.2 Landasan Teori

Landasan Teori merupakan definisi, konsep yang telah disusun secara sistematis dan dasar yang kuat dalam sebuah penelitian. Landasan teori yang digunakan dalam penyusunan Pembangunan Aplikasi *Geofencing* di kota Cirebon berbasis *mobile* meliputi Aplikasi Mobile, Pengertian Aplikasi, Pengertian Mobile, Pengertian Aplikasi Mobile, Karakteristik Perangkat Mobile, Android, Teori Kualitas, Basis Data, DBMS, SQL, Pemrograman Berorientasi Objek, Konsep Dasar Berorientasi Objek, Geofencing, Jaringan Internet, GPS, LBS, Google Maps, Phonegap, HTML5, PHP, JavaScript, JSON, Database MySQL, *Unified Modeling Language* (UML), Web Server dan Web Service.

2.2.1 Aplikasi Mobile

Aplikasi mobile saat ini sangat dibutuhkan karena alat-alat telekomunikasi yang tersebar di seluruh dunia membutuhkan aplikasi-aplikasi yang dapat memepermudah pekerjaan penggunanya dimanapun dan kapanpun terutama dalam hal informasi.

2.2.2 Pengertian Aplikasi

Menurut Buyens [3] aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Jika ingin mengembangkan program aplikasi sendiri, maka untuk menulis program aplikasi tersebut, dibutuhkan suatu bahasa pemrograman, yaitu *language software*, yang dapat berbentuk *assembler*, *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.

Sekarang, banyak sekali program-program aplikasi yang tersedia dalam bentuk paket-paket program. Ini adalah program-program aplikasi yang sudah

ditulis oleh orang lain atau perusahaan-perusahaan perangkat lunak. Beberapa perusahaan perangkat lunak telah memproduksi paket-paket perangkat lunak yang mempunyai reputasi internasional. Program-program paket tersebut dapat diandalkan, dapat memenhi kebutuhan pemakai, dirancang dengan baik, relatif bebas dari kesalahan-kesalahan, user friendly (mudah digunakan), mempunyai dokumentasi manual yang memadai, mampu dikembangkan untuk kebutuhan mendatang, dan didukung perkembangannya. Akan tetapi, bila permasalahannya bersifat khusus dan unik, sehingga tidak ada paket-paket program yang sesuai untuk digunakan, maka dengan terpaksa harus mengembangkan program aplikasi itu sendiri.

2.2.3 Pengertian Mobile

Mobile diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat yang lain, misalnya telepon mobile berarti bahwa 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 [3] 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.

Maka 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 layout, tetapi mereka memiliki karakteristik yang sangat berbeda dari sistem desktop.

Berikut karakteristik perangkat mobile, diantaranya :

a. Ukuran yang kecil

Perangkat mobile memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.

b. Memory yang terbatas

Perangkat mobile juga memiliki memory yang kecil, yaitu primary (RAM) dan secondary (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.

c. 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.

d. 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.

e. Kuat dan dapat diandalkan

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

f. Konektivitas yang terbatas

Perangkat mobile memiliki bandwith rendah, beberapa dari mereka bahkan tidak tersambung. Kebanyakan dari mereka menggunakan koneksi wireless.

g. Masa hidup yang pendek

Perangkat-perangkat konsumen ini menyala dalam hitungan detik kebanyakan dari mereka selalu menyala. Coba ambil kasus sebuah 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

   SMS : Short Message Service. Pengiriman pesan singkat antar ME / dengan sebuah server Terbatas ± 160 characters.

   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.

b. Aplikasi menunggu data dari interface dan mengirim data melalui interface tersebut.

3. SIM-card based

Aplikasi diletakkan di atas SIM-card, aplikasi dapat diakses melalui menu baru yang akan muncul di ponsel. Javacard, khusus untuk javacard aplikasi bersifat downloadable, pengiriman 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 Pemrograman di atas Symbian 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 antara user dengan satu atau lebih user lain.
3. Entertainment
   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 mobile yang tumbuh di tengah sistem operasi lainnya yang berkembang dewasa ini. Sistem operasi lainnya seperti Windows Mobile, i-Phone OS, Symbian dan masih banyak lagi juga 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 karena itu, adanya keterbatasan distribusi aplikasi pihak ketiga untuk platform mereka.

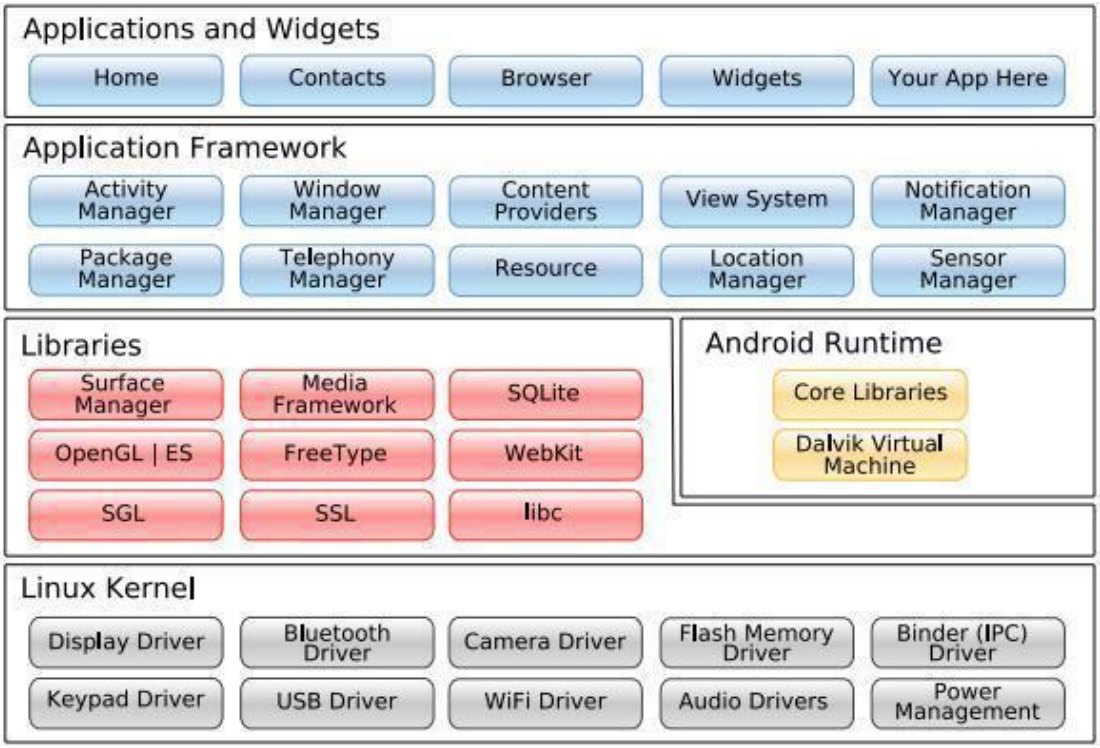
Android menawarkan sebuah lingkungan yang berbeda untuk pengembang. Setiap aplikasi memiliki tingkatan yang sama. Android tidak membedakan antara aplikasi inti dengan aplikasi pihak ketiga. Application Programming Interface (API) yang disediakan menawarkan akses ke hardware, maupun data-data ponsel sekalipun, atau data sistem sendiri. Bahkan pengguna dapat menghapus aplikasi inti dan menggantikannya dengan aplikasi pihak ketiga. [4]

2.2.1.1 Arsitektur Android

Arsitektur Android dapat digambarkan seperti pada gambar 1 dan secara garis besar Arsitektur Android dapat dijelaskan sebagai berikut:

a. Application dan Widgets

   Application dan Widgets ini adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita download aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di layer terdapat aplikasi inti termasuk klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain. Hampir semua aplikasi ditulis menggunakan bahasa pemrograman Java.



Gambar 2.2 Arsitektur Android

b. Aplication Framework

Android adalah "Open Development Platform"yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi resource, menjalankan service background, mengatur alarm, dan menambah status notifications, dan sebagainya. 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). Sehingga bisa kita simpulkan Application Frameworks ini adalah layer dimana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti content providers yang berupa sms dan

panggilan telepon. Komponen-komponen yang termasuk di dalam Application Frameworks adalah sebagai berikut :

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

c. Libraries

Libraries ini adalah layer dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses libraries untuk menjalankan aplikasinya. Berjalan di atas Kernel, layer ini meliputi berbagai library C/C++ inti seperti Libc SSL, serta :

a. Libraries media untuk pemutaran media audio dan video
b. Libraries untuk manajemen tampilan
c. Libraries Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D
d. Libraries SQLite untuk dukungan database
e. Libraries SSL dan WebKit terintegrasi dengan web browser dan security
f. Libraries LiveWebcore mencakup modern web browser dengan engine embedded webview
g. Libraries 3D yang mencakup implementasi OpenGL ES1.0 API's. [4]

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:

a. Core Libraries: Aplikasi Android dibangun dalam bahasa Java, sementara Dalvik sebagai virtual mesinnya bukan Virtual Machine

Java, sehingga diperlukan sebuah libraries yangberfungsi untuk menterjemahkan bahasa Java/C yang ditangani oleh Core Libraries.

b. Dalvik Virtual Machine: Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat Linux Kernel untuk melakukan threading dan manajemen tingkat rendah. [4]

e. Linux Kernel

Linux Kernel adalah layer dimana inti dari system operasi Android itu berada. Berisi file-file system yang mengatur sistem processing, memory, resource, drivers, dan sistem-sistem operasi Android lainnya. Linux Kernel yang digunakan Android adalah Linux Kernel release 2.6. [4]

2.2.7 Teori Kualitas

Membicarakan tentang pengertian atau definisi kualitas dapat berbeda makna bagi setiap orang, karena kualitas memiliki banyak kriteria dan sangat tergantung pada konteksnya. Banyak pakar dibidang kualitas yang mencoba untuk mendefinisikan kualitas berdasarkan sudut pandangnya masing-masing.

Menurut Roger S. Pressman [5] “Kualitas sistem, 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 pada saat ini menjadi salah satu komponen penting dalam kehidupan manusia. software digunakan untuk mempermudah pekerjaan - pekerjaan yang memerlukan keakuratan, ketepatan dan efisiensi waktu yang tinggi.

2.2.7.1 Faktor – Faktor Yang Mempengaruhi Kualitas

McCall dan Cavano dalam buku Roger S.Pressman (2002:109), menjelaskan bahwa faktor-faktor yang mempengaruhi Software terbagi menjadi tiga, yaitu :

1. Operasi produk (menggunakannya)
2. Revisi produk (mengubahnya)
3. Transisi produk (memodifikasinya untuk bekerja 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 koreksi 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 mempengaruhi kualitas 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).

Dalam penelitian ini penulis hanya membahas satu faktor/dimensi saja yaitu faktor-faktor/dimensi-dimensi yang berkaitan dengan sifat-sifat operasional software, yang 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 untuk memperluas program (ekstensibilitas), kemampuan beradaptasi, dan kemampuan pelayanan (atribut-atribut ini mempresentasikan sutau teknologi yang lebih umummaintainability),
   serta testabilitas, kompatibilitas (kemampuan untuk
   mengumpulkan dan mengontrol elemen dari konfigurasi perangkat lunak, kecocokan di mana suatu sistem dapat dipasang, dan kecocokan di mana masalah dapat dilokalisasikan.

Perangkat lunak dikatakan baik apabila dapat 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 [5] “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 ini

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 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. [6]

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

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

2.2.9 DBMS

DBMS (Database Management System) atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data [6]. 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

2.2.10 SQL

SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya 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 tersebut. Sehingga kadang-kadang ada perbedaan perilaku (hasil yang ditampilkan) oleh DBMS yang berbeda padahal query yang dimasukkan sama. [6]

2.2.11 Pemrograman Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan peralngkan lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientas objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek. [6]

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, misalnya pada metode pendekata 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 dunai nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

2.2.12 Konsep Dasar Beroirientasi 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 objek-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 (dihubungkus) menjadi suatu kesatuan.

Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancanggan, pemrograman, dan pengujian perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu. [6]

Sistem berorientasi objek merupakan sebuah sistem yang dibangun dengan berdasarkan metode berorientasi objek adalah sebuah sistem yang komponennya dibungkus (dienkapsulasi) menjadi kelompok daata 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 karakteristik yang sama. Kelas merupakan definisi statis 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 tekniks kelas adalah sebuah struktur dalam pembuatan perangkat lunak. Kelas merupakan bentuk struktur pada kode program yang menggunakan metodologi berorientasi objek.

2. Objek (*object*)

   Objek adalah abstraksi dari sesuatu yang mewakili dunia nyata benda, manusia, satua organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstra. 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 makan akan dibuat sebuah objek. Objek dilihat darisegi teknis adalah elemen pada saat runtime yang akan diciptakan, dimanipulasi, dan dihancurkan saat eksekusi sehinga

sebuah objek hanya ada saat sebuah program dieksekusi. Jika masih dalam bentuk kode, disebut sebagai 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 kelakuan dunnia nyata. Contoh metode atau operasi misalnya *Read*, *Write*, *Move*, *Copy*, dan sebagainya.

4. Atribut (*attribute*)

   Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas. Atribut dapat beruoa 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 merepresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

6. Enkapsulasi (*encapsulation*)

   Pembungkusa atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerja-nya.

7. Pewarisan (*inheritance*)

   Mekanisme 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 Antarobjek

    Komunikasi antarobjek 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.

2.2.13 Geofencing

Geofence adalah parameter virtual untuk wilayah geografis di dunia nyata. Parameter virtual yang ditetapkan ini bisa menjadi statis ataupun dinamis. Salah satu contoh khas dari geofence dinamis adalah radius sekitar toko atau titik lokasi. geofence 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 perangkat. 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, di mana pengguna harus secara eksplisit meminta data untuk lokasi. Ada berbagai jenis lokasi Peristiwa untuk memperbaiki posisi GPS yang dapat diuji. Misalnya, apakah pengguna di dekat dengan tempat tujuan (POI) atau ke pengguna lain.

Geofence sekarang ini menjadi salah satu wajah baru di Layanan Berbasis Lokasi. Sekarang pengembangan asli untuk LBS sendiri lebih atau kurang selesai. Kita bisa melihat beberapa standar umum dan aplikasi (berpikir tentang berbagai implementasi layanan Tempat-tempat seperti Foursquare, Twitter places, Google place, Facebook place, dll) Semua dari mereka memudahkan Anda dalam berbagi lokasi info 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. [7]

2.2.14 Jaringan Internet

Internet adalah suatu jaringan computer global yang terbentuk dari jaringan-jaringan computer local dan regional yang memungkinkan komunikasi data antar computer yang terhubung ke jaringan tersebut. Internet awalnya merupakan suatu rencana dari Departemen Pertahanan Amerika Serikat (US Department of Defense) pada sekitar tahun 1960. Dimulai dari suatu proyek yang dinamakan ARPANET atau Advanced Research Project Agency Network. Beberapa universitas di Amerika Serikat diantaranya UCLA, Stanford, UC Santa Barbara dan University of Utah, diminta bantuan dalam mengerjakan proyek ini dan awalnya telah berhasil menghubungkan empat computer di lokasi universitas berbeda tersebut.

2.2.15 GPS (Global Positioning System)

GPS adalah singkatan dari Global Positioning System, yang merupakan sistem navigasi dengan menggunakan teknologi satelit yang dapat menerima sinyal dari satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang 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 sebagainya. [8]

2.2.1 Akurasi GPS

Posisi yang ditunjukkan oleh suatu GPS mempunyai faktor kesalahan atau juga disebut tingkat akurasi. Sebagai contoh suatu alat GPS menunjukkan titik koordinat dengan tingkat akurasi 5 meter, itu berarti posisi pengguna bisa berada dalam range radius 5 meter dari titik yang ditunjukkan tersebut. Mengapa tingkat akurasi yang terlihat bisa berubah-ubah? Kadang terlihat 10 meter, 15 meter, atau 5 meter. Ada beberapa hal yang mempengaruhi 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 atas permukaan bumi. Satelit yang melewati ionosphere 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 terjadinya delay sehingga perhitungan 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. [8]

2.2.2 Android dan GPS

Dewasa ini, teknologi berkembang dengan pesat. Dulu ponsel hanya sekedar digunakan untuk menelpon dan SMS saja. Sekarang ponsel sudah menjelma menjadi kotak kecil ajaib yang serba bisa. Salah satunya adalah ponsel dengan sistem operasi Android. Dengan standarisasi fitur dan hardware yang dimiliki, menjadikan 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. Ya, perangkat GPS receiver yang dulu besar dan eksklusif, 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 data koordinat? Ingin tahu jalur olahraga bersepeda 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. [8]

2.2.3 Latitude dan Longitude

Latitude dan longitude adalah sebuah koordinat yang ada di bumi dan digunakan untuk menentukan kedudukan kita di atas bumi. [9]

2.2.3.1 Latitude

Latitude adalah garis yang melintang dari kutub utara dan kutub selatan. Titik 0 adalah sudut ekuator, tanda + menunjukan arah ke atas menuju kutub utara, sedangkan tanda minus di koordinat Latitude menuju ke kutub selatan. Titik yang dipakai dari 0 ke 90 derajat arah kutub utara, dan 0 ke -90 derajat ke kutub selatan. [9]

2.2.3.2 Longitude

Longitude adalah garis lintang. Angka dari sudut bundar bumi horisontal. Titik 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 Longitude minus adalah arah kebalikan. [9]

2.2.3 LBS (*Location Based-Service*)

Salah satu ciri mobile phone atau tablet pc adalah portabilitas, sehingga tidak mengherankan bahwa beberapa fitur Android sangat menarik seperti layanan yang memungkinkan anda menemukan, mencari serta menvisualisasikan posisi kita ke dalam peta lokasi fisik seperti GoogleMaps. Kita dapat membuat peta berbasis GoogleMaps dan menjadikannya sebagai element dalam layout User Interface (UI) yang kita rancang. Kita dapat melakukan akses penuh ke peta GoogleMaps, dan memungkinkan kita untuk mengontrol pengaturan tampilan, mengubah tampilan zoom lokasi, dan memindahkan lokasi tampilan. [8]

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 realtime kita bukan nilai lintang dan bujur. Android menyediakan geocoder yang mendukung forward dan reverse geocoding. Menggunakan geocoder, anda dapat mengkonversi nilai lintang bujur menjadi alamat dunia nyata atau sebaliknya.

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

a. Location Manager (API Maps)

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

b. Location Providers (API Location)

   Menyediakan teknologi pencarian lokasi yang digunakan oleh device/perangkat. API Location berhubungan dengan data GPS dan data

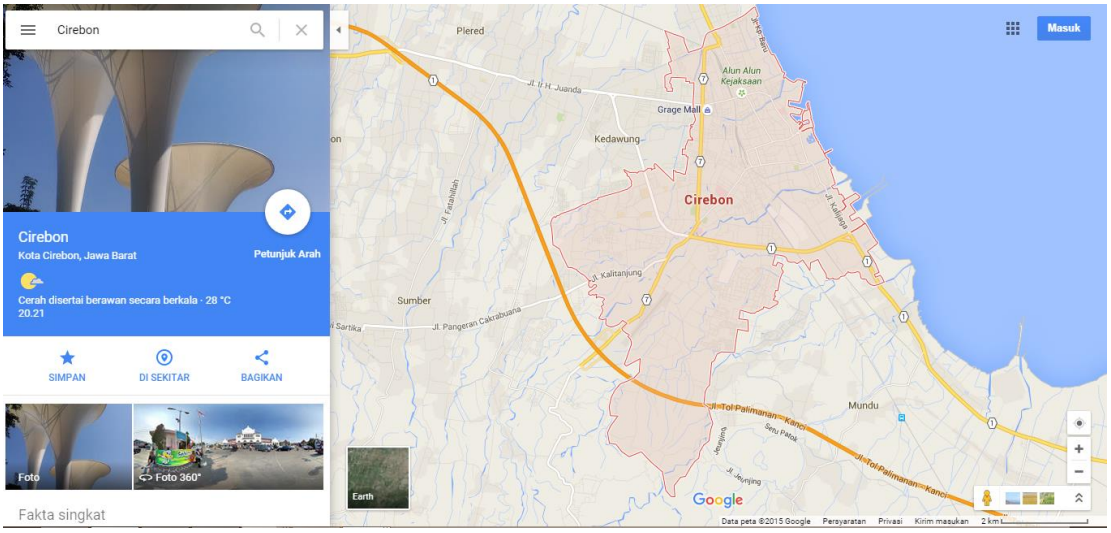
lokasi realtime. API Location berada pada paket Android yaitu dalam paket android.location. Dengan Location Manager, kita dapat menentukan lokasi kita saat ini, Track gerakan/perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan. [8]

2.2.4 Google Maps

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

GoogleMaps adalah layanan gratis yang diberikan oleh Google dan sangat popular. GoogleMaps adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah seperti pada gambar 2. Dengan kata lain, GoogleMaps merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur GoogleMaps dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan GoogleMaps API. GoogleMaps API adalah suatu library yang berbentuk JavaScript.

Cara membuat GoogleMaps untuk ditampilkan pada suatu web atau blog sangat mudah hanya dengan membutuhkan pengetahuan mengenai HTML serta JavaScript, serta koneksi Internet yang sangat stabil. Dengan menggunakan GoogleMaps 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 mambuat peta suatu lokasi, bahkan dunia.



Gambar 2.3 Google Maps

Pada GoogleMaps 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, untuk 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 pula apa yang tampil pada ROADMAP (jalan dan nama kota). [8]

2.2.5 PhoneGap

PhoneGap adalah open source framework untuk membuat cross-platfrom native applications menggunakan teknologi web mulai dari HTML, CSS, dan JavaScript. Tipe dari aplikasi ini disebut sebagai hybrid application. PhoneGap diciptakan untuk mempermudah *mobile development*. PhoneGap bekerja dengan cara merubah *web application package* menjadi *native application*. Aplikasi yang telah dibuat akan ditampilkan dalam bentuk *web view* yang memungkinkan pengguna untuk melakukan interaksi dengan aplikasi tersebut. Tampilan dari aplikasi yang dibuat oleh PhoneGap dibentuk oleh CSS yang di implementasikan kedalam HTML dan untuk komunikasi data bisa menggunakan XML atau JSON.

Aplikasi yang telah selesai dibuat dengan PhoneGap akan dirubah dengan *Software Development Kit (SDK)* platform yang didukung oleh PhoneGap menjadi *native application*.

Saat ini PhoneGap mendukung *platform* terkenal antara lain Apple iOS, Google Android, HP/Palm webOS, Microsoft Windows Phone, Nokia Symbian, RIM BlackBerry, Samsung Bada. Dikarenakan fungsi setiap platform berbeda-beda, maka PhoneGap menyediakan *Application Programming Interface (API)* untuk mempermudah pengembang diantaranya adalah *Accelerometer, Camera, Capture, Compass, Connection, Contacts, Device, Events, File, Geolocation, Media, Notification, dan Storage.* API tersebut dibuat agar pengembang dapat mengakses fungsi *native application* melalui *javascript* dengan *syntax* yang sama di semua *platform*. Tidak semua *platform* dapat menggunakan API PhoneGap karena keterbatasan *software* maupun *hardware* tiap *platform*. [10]

2.2.6 HTML5

HyperText Markup Language (HTML) adalah bahasa pemograman lain untuk membangun sebuah halaman *web*. 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 Wanua*, sebuah teknologi inti dari Internet.

HTML5 adalah revisi kelima dari HTML (yang pertama kali diciptakan pada tahun 1990 dan versi keempatnya, HTML4, pada tahun 1997) dan hingga bulan Juni 2011 masih dalam pengembangan. Tujuan utama pengembangan HTML5 adalah untuk memperbaiki teknologi HTML agar mendukung teknologi multimedia terbaru, mudah dibaca oleh manusia dan juga mudah dimengerti oleh mesin. HTML5 merupakan salah satu karya *Konsortium Waring Wera Wanua (World 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* [11]. Contoh skrip struktur umum HTML5 sebagai berikut :

Skrip HTML

```
<!DOCTYPE html>
<html>
    <head>
        <title>Web Warung Tegal</title>
        <style type="text/css">
        </style>
        <!--[if lt IE 9]>
        <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
        <![endif]-->
    </head>
    <body>
        <div class="wrapper">
        </div><!-- .wrapper -->
    </body>
</html>
```

Gambar 2.4 Contoh Skrip HTML5

2.2.7 PHP

PHP adalah suatu bahasa *server-side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan diantara HTML, dan karena bahasa *server-side*, maka bahasa PHP akan dieksekusi di server, yang dikirimkan ke browser adalah "hasil jadi" bentuk HTML, dan kode PHP tidak akan terlihat. PHP dahulu merupakan proyek pribadi dari Rasmus Lerdorf tahun 1994 (dengan dikeluarkannya PHP versi 1 yang digunakan untuk membuat home page pribadi yang dipublikasikan 8 juni 1995). Versi pertama ini berupa kumpulan script PERL. Untuk versi kedua Ramus menulis ulang script-script PERL tersebut menggunakan bahasa C, kemudian menambahkan fasilitas untuk form HTML dan koneksi MYSQL. Adapun PHP didapat dari singkatan *PHP Hypertext Preprocessor*.

Setelah mengalami perkembangan, Zeev Suraski dan Andi Gutmans (termasuk Rasmus) yang tergabung dalam suatu kelompok open source, maka mereka memulai versi 3 PHP nya, PHP telah menampakkan keunggulan sebagai salah satu bahasa *server scripting* yang handal. Melalui perkembangan yang pesat ini banyak fasilitas yang ditambahkan oleh kelompok ini PHP disebut sebgai "PHP: Hypertext Preprocessor". Pada bulan Mei 2000 dirilis PHP versi 4 yang dikeluarkan oleh Zend Engine yaitu sebuah produk yang diciptakan oleh Suraski dan Gutmans. Kehandalan PHP diperlengkap dengan dirilisnya PHP versi 5 pada 13 Juli 2004 yang mendukung OOP serta berbagai manajemen database yang luas dan

penanganan error yang mudah. Sintak yang digunakan berasal dari bahsa C, Java maupun PERL. Sampai tulisan ini dibuat versi PHP yang terbaru adalah versi 5.2.0 yang dirilis pada November 2006.

PHP merupakan bahasa script yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. Oleh karena itu, spesifikasi server lebih berpengaruh pada eksekusi dari script php daripada spesifikasi client. Namun tetap diperhatikan bahwa web yang dihasilkan tentunya harus dapat dibuka oleh browser pada client. Dalam hal ini versi dari html yang digunakan harus didukung oleh browser client. PHP termasuk dalam *Open Source Product*. Jadi dapat menambah *source code* dan dapat mendistribusikannya secara bebas. PHP juga diedarkan secara gratis. Kita bisa mendapatkannya secara gratis. PHP juga dapat berjalan diberbagai web server semisal IIS, Apache, PWS.

```
<?php
        header('Access-Control-Allow-Origin: *');

        require_once("../class/about.php");

        $obj = new About();

        echo json_encode($obj);
?>
```

Gambar 2.5 Contoh skrip php

2.2.8 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 *scripting* yang dibuat untuk berinteraksi 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*.

```
<script type="text/javascript">
        function onclick_callback ()
        {
        alert ("Hello, World!");
        }
</script>
<span onclick="onclick_callback();">Click Here</span>
```

Gambar 2.6 Contoh skrip javascript

2.2.9 JSON

JavaScript Object Notation (JSON) pertama kali dikenalkan oleh Douglas Crockford adalah bagian dari *JavaScript*, dibuat dengan menggunakan pola dari JavaScript untuk merepresentasikan data terstruktur.

Crockford membuat JSON sebagai alternatif dari *(eXtended Markup Language*) XML untuk mengakses data terstruktur dari *JavaScript* dengan menggunakan *method eval()* dan tidak membutuhkan pembuatan dari DOM. Bagian terpenting dari JSON adalah format data bukan bahasa pemrogramannya. JSON bukan bagian dari *JavaScript* walaupun mereka saling berbagi *syntax*, JSON juga tidak semata-mata digunakan oleh *JavaScript*. JSON merupakan data format, jadi JSON memiliki *parsers* dan *serializers* yang ada dalam banyak bahasa pemograman.

2.2.10 Database 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 system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. MySQL adalah implementasi dari system manajemen basisdata relasional (RDBMS). MySQL dibuah oleh TcX dan telah dipercaya mengelola system 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 yang memiliki MySQL. Pendiri

MySQL AB adalah dua orang Swedia yang bernama David Axmark, Allan Larsson dan satu orang Finlandia 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.

Pada saat ini MySQL merupakan database server yang sangat terkenal di dunia, semua itu tak lain karena bahasa dasar yang digunakan untuk mengakses database yaitu SQL. SQL (Structured Query Language) pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama system R. Kemudian SQL juga dikembangan oleh Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses pengaksesan database lebih user-friendly dibandingan dengan yang lain, misalnya dBase atau Clipper karena mereka masih menggunakan perintah-perintah pemrograman murni. SQL dapat digunakan secara berdiri sendiri maupun di lekatkan 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).

a. Data Definition Language (DDL)

   Kumpulan perintah SQL yang digunakan untuk membangun database, bisa juga memiliki arti lain perintah untuk membuat struktur dasar database dan

tabel. Perintah-perintah yang termasuk DDL adalah statment connect, create, show, describe, drop, rename, dan alter.

b. Data Manipulation Language (DML)

Kumpulan statment yang digunakan sebagai perintah untuk mengelola data dalam sebuah tabel. Perintah DML digunakan setelah menggunakan DDL anatar lain seperti statment insert, update, delete, dan select.

c. Data Control Language (DCL)

Perintah yang digunakan untuk keperluan keamanan (security) database dengan membuat hak akses tertentu bagi setiap user. Statment pada DCL antara lain grant, revoke, set, dan lock table.

Dalam MYSQL ada beberapa operasi dasar yang lebih dikenal dengan operasi CRUD yaitu:

1. *Create*

   Create adalah operasi penambahan data baru kedalam table. Terdapat 2 Query untuk menambah data, yang pertama adalah penambahan yang tidak menspesifikasikan nama kolom yang akan ditambahkan dan hanya memberikan isi dari tabelnya. Querynya adalah sebagai berikut:

   INSERT INTO table_name VALUES (value1,value2,value3,...);

   Untuk yang kedua adalah menginputkan data dengan mencantumkan nama kolom yang akan diisikan dengan isi di dalamnya. Querynya adalah sebagai berikut:

   INSERT INTO table_name (column1,column2,column3,...)
   VALUES (value1,value2,value3,...);

2. *Read*

   Read adalah operasi untuk menampilkan semua atau sebagian data yang berada di dalam database. Querynya adalah sebagai berikut:

   SELECT column_name,column_name
   FROM table_name;

3. *Update*

   Update adalah operasi untuk mengubah data yang ada di dalam database. Querynya adalah sebagai berikut:

   UPDATE table_name SET column1=value1,column2=value2,...

   WHERE some_column=some_value;

4. *Delete*

   Delete adalah operasi untuk menghapus data yang ada di dalam table. Querynya adalah sebagai berikut:

   DELETE FROM table_name

   WHERE some_column=some_value;

2.2.11 *Unified Modeling Language* (UML)

Analisis dan desain berorientasi objek adalah cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. Pengertian berorientasi objek berarti bahwa cara mengorganisasikan perangkat lunak sebagai kumpulan dari objek tertentu yang memiliki struktur data perilakunya.

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

1. Analisis berorientasi objek (OOA)

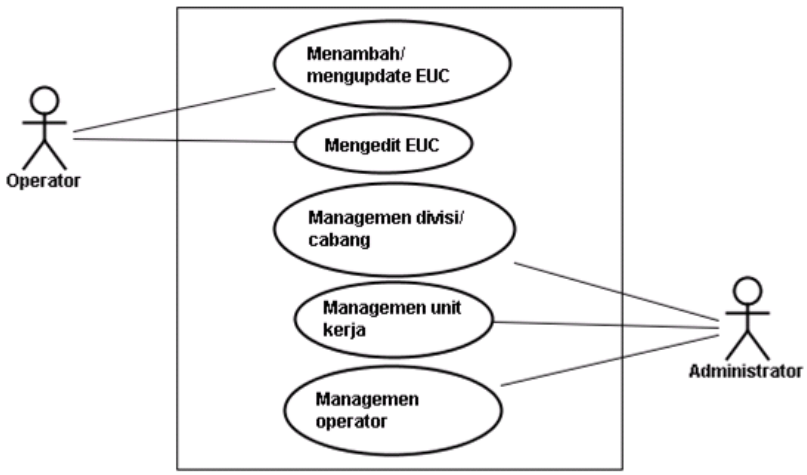
Metode analisis yang memerika requirement (syarat/keperluan) yang harus dipenuhi sebuah sistem dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. Unified Modeling Language (UML) adalah bahasa untuk mengspesifikasi, memvisualisasi, membangun dan mendokumentasi 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. Berikut ini merupakan beberapa bagian dari UML adalah sebagai berikut :

a. Use Case Diagram

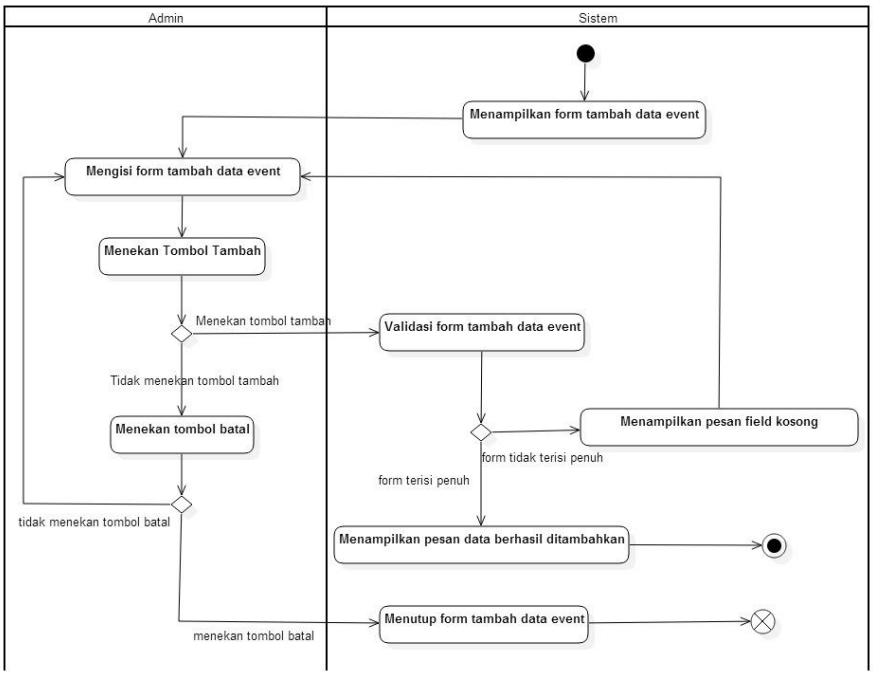
Use case diagram menggambarkan sejumlah eksternal aktor dan hubungannya ke use case yang diberikan oleh sistem.



Gambar 2.7 Contoh *Use Case Diagram*

b. Activity Diagram

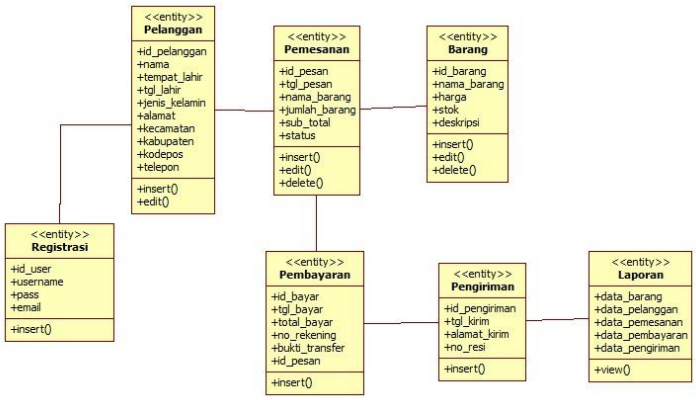
Activity diagram menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainya seperti *use case*.



Gambar 2.8 Contoh *Activity Diagram*

c. Class Diagram

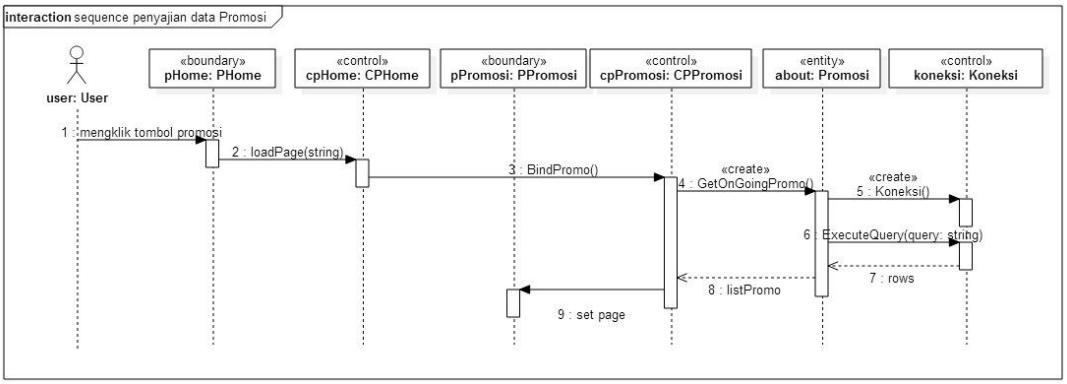
Class diagram menggambarkan struktur dan deskripsi class, package (paket) dan objek beserta hubungan satu sama lain seperti containment (penahanan), pewarisan, asosiasi dan lain-lain.



Gambar 2.9 Contoh *Class Diagram*

d. Sequence Diagram

Sequence Diagram menggambarkan kolaborasi dinamis antara sejumlah objek dan untuk menunjukan rangkaian pesan yang dikirim antara objek juga interaksi. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.



Gambar 2.10 Contoh *Sequence Diagram*

2.2.12 Web Server

Tahun 1989, Tim Berners-Lee mengajukan pada perusahaannya, CERN (European Organization for Nuclear Research) sebuah proyek yang bertujuan untuk mempermudah pertukaran informasi antar para peneliti dengan menggunakan sistem hiperteks. Sebagai hasil atas implementasi proyek ini, tahun 1990 Berners-Lee menulis dua program komputer yaitu :

a. sebuah peramban yang dinamainya sebagai WorldWideWeb.
b. server web pertama di dunia, yang kemudian dikenal sebagai CERN httpd, yang berjalan pada sistem operasi NeXTSTEP.

Dari tahun 1991 hingga 1994, kesederhanaan serta efektifitas atas teknologi yang digunakan untuk berkunjung serta bertukar data melalui World Wide Web membuat kedua aplikasi tersebut diadopsi pada sejumlah sistem operasi agar dapat digunakan oleh lebih banyak individu, ataupun kelompok. Awalnya adalah organisasi penelitian, kemudian berkembang dan digunakan di lingkungan pendidikan tinggi, dan akhirnya digunakan dalam industri bisnis. Tahun 1994, Tim Berners-Lee memutuskan untuk membakukan organisasi World Wide Web

Consortium (W3C) untuk mengatur pengembangan-pengembangan lanjut atas teknologi-teknologi terkait lainnya (HTTP, HTML, dan lain-lain) melalui proses standardisasi.

2.2.13 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 pemrogram dan perusahaan, yang memungkinkan sebuah fungsi di dalam Web Service dapat dipinjam oleh aplikasi lain tanpa perlu mengetahui detil pemrograman yang terdapat di dalamnya.

Beberapa alasan mengapa digunakannya web service adalah sebagai berikut:

1. Web service 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 deployment-nya, 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 otorisasi.
3. Web service berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian web service tidak memerlukan konfigurasi khusus di sisi firewall.