

BAB II

TINJAUAN PUSTAKA

2.1. Landasan Teori

Landasan teori merupakan untuk untuk mendukung dalam perancangan sebuah penelitian baik secara teori maupun hal-hal yang berkaitan dengan penelitian ini. Di dalam penelitian ini, akan membahas berbagai teori yang berkaitan dengan skripsi ini, meliputi pengertian Teknologi mobile, Android, Android Studio, Android SDK (Software Development Kit), Google Maps, Java, MySQL dan UML.

2.1.1. Transportasi

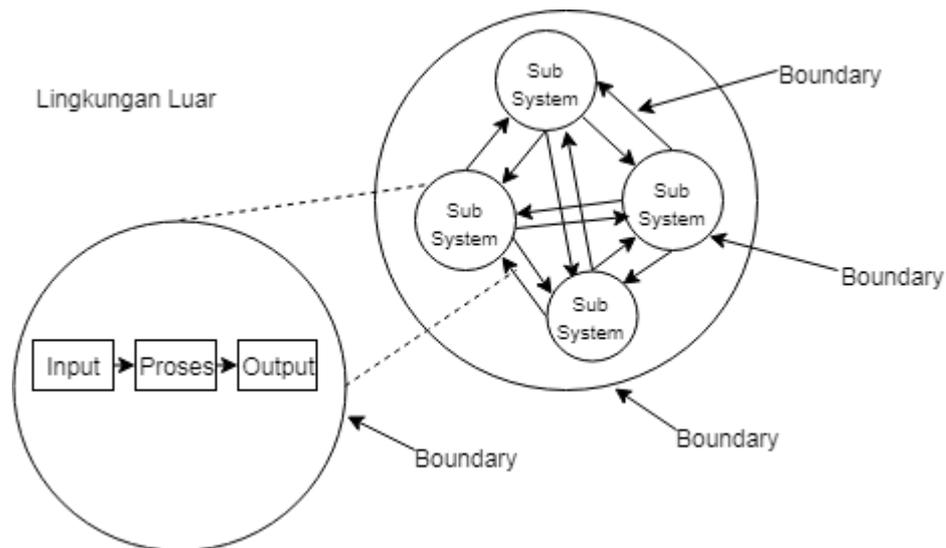
Transportasi merupakan kebutuhan penting di saat ingin bepergian kemanapun baik jarak dekat maupun jarak jauh. Penduduk menjadikan transportasi adalah bagian penting dalam mendukung aktifitas sehari-hari. Teknologi informasi yang berkembang saat ini dapat membantu memberikan sebuah sistem informasi yang terintegrasi dalam penggunaannya. Pemanfaatan teknologi informasi saat ini mampu membantu para penggunanya dalam melakukan kegiatan sehari-hari terutama bagi mahasiswa yang ingin pergi ke kampus tanpa harus memikirkan biaya yang mahal dan kendaraan umum yang sulit ditemukan. Terlebih lagi pengguna tidak perlu khawatir lagi dengan siapa dia akan pergi ke kampus.

2.1.2. Sistem

Sistem adalah sekumpulan unsur / elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Menurut Ludwig Von Bertalanffy, Sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur-unsur tersebut dengan lingkungan. Sementara menurut Anatol Rapoport, Sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.

2.1.3. Karakteristik Sistem

Suatu sistem memiliki karakteristik yang akan merepresentasikan ciri sebuah sistem itu sendiri. Berikut gambaran karakteristik dari sebuah sistem dibawah ini :



Gambar 2.1 Karakteristik Sistem

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan.

2. Batasan Sistem

Batasan merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungannya. Batasan sistem ini memungkinkan suatu sistem dipandang suatu kesatuan.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

5. Masukan Sistem

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem

Keluaran sistem adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya.

8. Sasaran Sistem

Suatu sistem mempunyai tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.4. Klasifikasi Sistem

Sistem dapat di klasifikasi berbagai sudut pandang, diantaranya adalah sebagai berikut :

1. Sistem Abstrak dan Fisik (*Physical Sistem*)

Sistem Abstrak yaitu sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara Fisik, misalnya sistem agama. Sementara sistem fisik adalah sistem yang keberadaannya dapat dilihat secara fisik, misalnya perusahaan, komputer.

2. Sistem Alamiah dan Buatan (*Human Made System*)

Sistem alamiah (natural system) yaitu sistem yang terbentuk melalui proses alami, misalnya sistem tatasurya, pencernaan. Sementara sistem buatan manusia adalah sistem yang dirancang dan dibangun oleh manusia yang melibatkan interaksi dengan mesin misalnya produksi di pabrik.

3. Sistem Tertentu dan Tidak Tertentu (*Probabilistic System*)

Sistem tertentu (deterministic system) yaitu sistem yang cara beroperasinya sudah dapat diprediksi, interaksi-interaksi didalamnya dapat dideteksi dengan pasti dan outputnya dapat diramalkan, misalnya pengolahan data. Sementara sistem tak tentu adalah sistem yang outputnya tidak dapat diprediksi dengan pasti karena mengandung unsur probabilitas.

4. Sistem Tertutup dan Terbuka (*Open System*)

Sistem tertutup yaitu sistem yang tidak berhubungan dengan dunia luar dan tidak terpengaruh dengan lingkungan luarnya (bekerja secara otomatis). Sebenarnya sistem tertutup tidak ada yang ada adalah relatif tertutup. Sementara sistem terbuka adalah sistem yang mempunyai hubungan dengan dunia luar dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan output untuk subsistem yang lain.

2.2. Informasi

Informasi adalah data yang diolah menjadi suatu bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan pada saat sekarang atau yang akan datang. Informasi juga merupakan fakta-fakta atau data yang telah diproses sedemikian rupa atau mengalami proses transformasi data sehingga berubah bentuk menjadi informasi.

2.3. Kualitas Sistem

Kualitas sistem informasi merupakan karakteristik dari informasi yang melekat mengenai sistem itu sendiri [DeLone dan McLean, 1992]. Pada penelitian

yang dilakukan oleh Istianingsih dan Utami pada tahun 2009 hasil estimasinya menunjukkan bahwa kualitas sistem terbukti secara signifikan berpengaruh positif terhadap kepuasan pengguna, dan pada penelitian yang dilakukan Adrianty pada tahun 2012 juga menyatakan bahwa kualitas sistem terbukti secara signifikan berpengaruh positif terhadap kepuasan pengguna. Begitu pula pada hasil penelitian yang dilakukan oleh Iranto dan Januarti pada tahun 2012 bahwa kualitas sistem berpengaruh positif terhadap kepuasan pengguna.

2.4. Kualitas Informasi

Kualitas informasi menunjukkan kualitas produk yang dihasilkan oleh aplikasi sistem informasinya dan informasinya akan mempunyai pengaruh pada pemakaiannya dan pada sistemnya. Kualitas informasi yang bermanfaat adalah relevan, akurat, tepat waktu, conciseness, jelas, dapat dikuantifikasi, dan konsisten. Semakin baik kualitas informasi, akan semakin tepat pula keputusan yang diambil. Kualitas dari suatu informasi (quality of information) tergantung dari 3 hal, sebagai berikut :

1. Akurat

Akurat berarti informasi harus bebas dari kesalahan-kesalahan dan tidak biasa atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (noise) yang dapat merubah atau merusak informasi tersebut.

2. Tepat pada waktunya

Tepat berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi. Dewasa ini mahalnya nilai informasi disebabkan harus cepatnya informasi tersebut didapat, sehingga diperlukan teknologi - teknologi mutakhir untuk mendapatkan, mengolah dan mengirimkannya.

3. **Relevan**

Relevan berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang relevan, tetapi relevan untuk akuntan. [1]

2.5. Teknologi Mobile

Teknologi mobile merupakan teknologi yang bergerak dengan kata lain teknologi yang telah tidak memakai penghubung kabel (nirkabel) dalam pemakaian teknologinya. Teknologi komunikasi mobile bisa disebut jaringan tanpa kabel yang pemakainya dengan penghubung sinyal. Teknologi komunikasi mobile adalah perkembangan dari teknologi komunikasi nirkabel yang semakin pesat dalam perkembangannya.

2.6. Teknologi RFID

Teknologi Radio Frequency Identification (RFID) adalah sebuah pengembangan teknologi pengambilan data secara otomatis atau pengenalan atau identifikasi obyek (Kenzeller,1999). Selama ini sistem otomatis yang dikenal adalah sistem bar code. Sistem bar code mempunyai keterbatasan dalam penyimpanan data serta tidak dapat dilakukan program ulang atas data yang tersimpan di dalamnya. Namun pada teknologi RFID, proses mengambil atau mengidentifikasi obyek atau data dilakukan secara contacless (tanpa kontak langsung).

Identifikasi obyek atau data pada teknologi RFID dilakukan dengan mencocokkan data yang tersimpan dalam memori tag/transponder dengan data yang dikirimkan oleh reader. RFID dibentuk oleh komponen utama tag (transponder), reader dan antenna. Tag dapat menggunakan daya (tag aktif) atau tidak (tag pasif) serta diletakkan pada obyek yang akan diidentifikasi. Pada tag pasif sinyal

dikirimkan oleh reader melalui gelombang elektromagnetik, kemudian tag akan merespon dan mengirimkan data/informasi di dalamnya (AIM Frequency Forums). Reader juga memiliki kemampuan untuk melakukan perubahan data pada tag selain membaca dan mengambil data informasi yang tersimpan dalam tag. Sedangkan antenna pada sistem RFID berpengaruh terhadap jarak jangkauan pembacaan atau identifikasi obyek.

Frekuensi yang digunakan pada tag adalah 125 KHz, 13,56 KHz dan microwave. Dalam konteks ini digunakan frekuensi 13,56 MHz dengan carrier ± 7 KHz dan sideband 9 dB μ A/m dengan jangkauan 30 meter (Dziersk, 2004). Kelebihan pada frekuensi 13,56 MHz antara lain tidak menggunakan baterai, biaya murah dan umur pakai lebih lama. Sedangkan pada tag dapat berupa card, riqid industrial, smart label dan harga tag berkisar antara \$0,05 - \$0,10 atau Rp453 - Rp906 (Kinsella, 2004). [2]

2.6.1. Android

Android adalah sistem operasi berbasis linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh android.inc, dengan dukungan finansial dari google yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007 bersamaan dengan didirikannya open handset alliance, konsorsium dari perusahaan-perusahaan hardware, software, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Adapun jenis-jenis versi android antara lain :

1. Android 1.5 Cupcake

Cupcake dirilis 30 april 2009. Cupcake menjadi versi android pertama yang menggunakan nama makanan. Konon katanya versi ini seharusnya versi 1.2, namun google memutuskan untuk membuatnya menjadi versi 1.5 cupcake adalah kue kecil yang dipanggang dalam cetakan berbentuk cup.

2. Android 1.6 Donut

Android versi 1.6 dirilis pada 15 September 2009. Pada versi ini diperbaiki beberapa kesalahan reboot, perubahan fitur foto dan video dan integrasi pencarian yang lebih baik. Donat merupakan panganan berbentuk cincin. Bulat bolong tengah. Adonan donat dimasak dengan cara digoreng dan biasanya disajikan dengan topping di atasnya.

3. Android 2.0 / 2.1 Eclair

Android 2.0 / 2.1 eclair dirilis 26 oktober 2009. Eclair adalah makanan penutup yakni kue yang biasanya berbentuk persegi panjang yang dibuat dengan krim ditengah dan lapisan coklat di atasnya.

4. Android 2.2 Froyo

Android 2.2 froyo dirilis 20 mei 2010. Merupakan versi terbaru dari sistem operasi Android yang telah dirilis Google. Android 2.2 Froyo generasi ke tujuh dari platform smartphone berbasis open source. Nama kode Froyo (dari kata frozen yogurt), update ke tujuh dari platform Android. Kemampuan Android Froyo diantaranya, yakni Flashlight dan fitur yang bisa mengubah smartphone menjadi hotspot atau sebagai router nirkabel.

5. Android 2.3 Gingerbread

Android versi 2.3 Gingerbread dirilis resmi tanggal 6 Desember 2010. Gingerbread memperkenalkan dukungan untuk komunikasi jarak dekat (NFC) (digunakan dalam solusi pembayaran mobile) dan inisiasi sesi protokol (SIP) (digunakan dalam telepon internet VoIP).

6. Android 3.0 Honeycomb

Android versi 3.0 honeycomb dirilis pada tanggal 22 Februari 2011. Merupakan versi yang dihentikan dukungannya dari platform Android yang dirancang untuk perangkat dengan ukuran layar yang lebih besar, terutama tablet. Selain penambahan fitur baru, Honeycomb memperkenalkan apa yang disebut tema antarmuka pengguna "hologram" yang baru dan model interaksi yang dibangun pada fitur utama Android, seperti multitasking, pemberitahuan dan widget.

7. Android 4.0 Ice Cream Sandwich

Android 4.0-4.0.2 API Level 14 dan 4.0.3-4.0.4 API Level 15 pertama dirilis 19 Oktober 2011. Dinamai Ice Cream Sandwich. Ice Cream Sandwich adalah lapisan es krim, biasanya rasa vanilla yang terjepit di antara dua kue coklat, dan biasanya berbentuk persegi panjang.

8. Android 4.1 Jelly bean

Android Jelly Bean diluncurkan pertama kali pada Juli 2012, dengan berbasis Linux Kernel 3.0.31. Terdiri dari Android 4.1 API Level 16, Android 4.2 API Level 17, Android 4.3 API Level 18. Penamaan mengadaptasi nama sejenis permen dalam beraneka macam rasa buah. Ukurannya sebesar kacang merah. Permen ini keras di luar tapi lunak di dalam serta lengket bila di gigit.

9. Android 4.4 KitKat

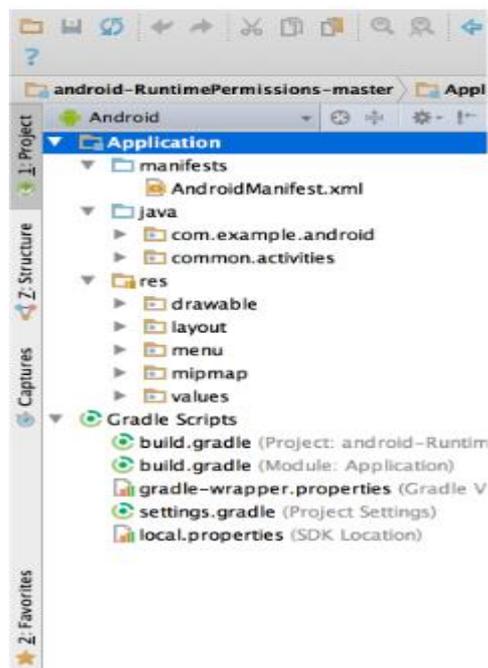
Android 4.4 Kitkat API level 19. Google mengumumkan Android KitKat (dinamai dengan izin dari Nestle dan Hershey) pada 3 september 2013. Dengan tanggal rilis 31 Oktober 2013. KitKat merupakan merk sebuah coklat yang dikeluarkan oleh Nestle. Rilis berikutnya setelah nama KitKat diperkirakan banyak pengamat akan diberi nomor 5.0 dan dinamai 'Pie'.

2.6.2. Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya

- a. Sistem versi berbasis Gradle yang fleksibel.
- b. Emulator yang cepat dan kaya fitur.
- c. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android.
- d. Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru.

- e. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh.
 - f. Alat pengujian dan kerangka kerja yang ekstensif.
 - g. Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain.
 - h. Dukungan C++ dan NDK.
 - i. Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.
1. Struktur Proyek



Gambar 2.2 File Proyek Tampilan Android

Dalam setiap proyek di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup antara lain :

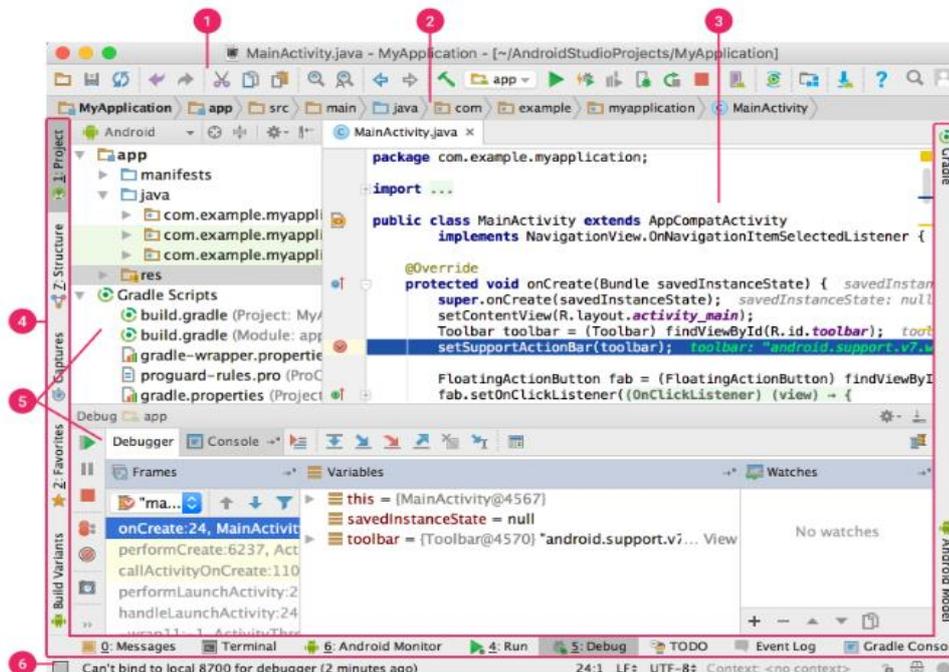
- a. Modul aplikasi android
- b. Modul pustaka
- c. Modul Google app Engine

Secara default, Android Studio akan menampilkan file proyek anda dalam tampilan proyek Android, seperti yang ditampilkan dalam gambar 2.1. tampilan disusun berdasarkan modul untuk memberikan akses secara cepat ke file sumber utama proyek.

Semua file versi terlihat dibagian atas dibawah Gradle Scripts dan masing-masing modul aplikasi berisi folder berikut :

- a. Manifest : Berisi file AndroidManifest.xml.
 - b. Java : Berisi file kode sumber Java, termasuk kode pengujian JUnit.
 - c. Res : Berisi semua sumber daya bukan kode, seperti tata letak XML, String UI, dan gambar bitmap.
2. Antarmuka Pengguna

Jendela utama Android Studio terdiri dari beberapa bidang logika yang didefinisikan dalam gambar 2.3.



Gambar 2.3 Jendela Utama Android Studio

1. **Bilah alat** : Memungkinkan untuk melakukan berbagai jenis tindakan, termasuk menjalankan aplikasi dan meluncurkan alat Android.
2. **Bilah navigasi** : Membantu anda bernavigasi diantara proyek dan membuka file untuk di edit, bilah memberikan tampilan struktur yang terlihat lebih ringkas dalam jendela project.
3. **Jendela editor** : adalah tempat membuat dan memodifikasi kode, bergantung pada jenis file saat ini, editor dapat berubah. Misalnya ketika melihat file tata letak, editor menampilkan layout editor.
4. **Bilah jendela alat** : muncul diluar jendela IDE dan berisi tombol yang memungkinkan meluaskan atau menciutkan jendela alat individual.
5. **Jendela alat** : memberi akses ke tugas tertentu seperti pengelolaan proyek, penelusuran, control versi, dan banyak lagi.
6. **Bilah status** : Menampilkan status proyek dan IDE itu sendiri, serta setiap peringatan atau pesan.

2.6.3. Android SDK (Software Development Kit)

Android SDK(Software Development Kit) adalah tools API(Application Programming Interface) yang diperlukan untuk memulai pengembangan suatu aplikasi pada platform android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware dan aplikasi kunci yang direlease oleh Google. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi netral, android memberi anda kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan aplikasi bawaan Handphone/smartphone. Adapun beberapa fitur android sebagai berikut :

1. Framework aplikasi yang mendukung penggantian komponen dan reusable.
2. Dalvik Virtual Machine dioptimalkan untuk perangkat mobile.
3. Integrated browser berdasarkan engine open source WebKit.

4. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi OpenGL ES 1.0.
5. SQLite untuk penyimpanan data/database.
6. Media support yang mendukung audio, video dan gambar.
7. Bluetooth, EDGE, 3G, WiFi (Tergantung hardware).
8. Kamera, GPS, kompas dan accelerometer (tergantung hardware).
9. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori dan plugin untuk IDE Eclipse. [3]

2.7. GPS (Global Positioning System)

GPS atau Global Positioning System, merupakan sebuah alat atau sistem yang dapat digunakan untuk menginformasikan penggunaannya dimana lokasinya berada (secara global) di permukaan bumi yang berbasis satelit. Data dikirim dari satelit berupa sinyal radio dengan data digital. Dimanapun pengguna tersebut berada, maka GPS bisa membantu menunjukkan arah. Layanan GPS ini tersedia gratis. Awalnya GPS hanya digunakan hanya untuk kepentingan militer, tapi pada tahun 1980-an dapat digunakan untuk kepentingan sipil. GPS dapat digunakan dimanapun juga dalam 24 jam. Posisi unit GPS akan ditentukan berdasarkan titik-titik koordinat latitude dan longitude. [4]

2.8. Google Maps

Google Maps merupakan layanan dari Google yang mempermudah penggunaannya untuk melakukan kemampuan pemetaan untuk aplikasi yang dibuat. Dan juga Google Maps adalah layanan pemetaan web yang dikembangkan oleh Google. Layanan ini memberikan citra satelit, peta jalan, panorama 360°, kondisi lalu lintas, dan perencanaan rute untuk bepergian dengan berjalan kaki, mobil, sepeda (versi beta), atau angkutan umum.

Google Maps untuk seluler dirilis pada bulan September 2008. Pada Agustus 2013, Google Maps bertekad untuk menjadi aplikasi yang paling populer di dunia untuk ponsel cerdas, dengan lebih dari 54% dari pemilik ponsel cerdas di seluruh dunia menggunakannya setidaknya sekali.

2.8.1. Pencarian Koordinat dengan Google Maps

Koordinat adalah suatu titik yang didapatkan dari hasil perpotongan dari garis latitude (lintang) dengan garis bujur (longitude) sehingga akan menunjukkan lokasi pada suatu daerah. Umumnya koordinat dibedakan menjadi koordinat Geographic dan Universal Transver Mercator (UTM). Koordinat untuk sebuah konten channel yang berupa Location Based Services (LBS) merupakan hal yang sangat penting. Koordinat memiliki peranan penting untuk menampilkan posisi objek. Satelit merupakan benda yang mengorbit benda lain dengan periode revolusi dan rotasi tertentu. Ada berbagai macam jenis satelit, di antaranya yaitu satelit alam dan satelit buatan. Dalam melakukan pencarian koordinat ini Penulis menggunakan satelit navigasi.

Satelit navigasi adalah satelit yang menggunakan sinyal radio yang disalurkan ke penerima di permukaan tanah untuk menentukan lokasi sebuah titik di permukaan bumi. Salah satu satelit navigasi yang sangat populer adalah Global Positioning System (GPS) milik Amerika Serikat selain itu ada juga Glonass milik Rusia. Bila pandangan antara satelit dan penerima di tanah tidak ada gangguan, maka dengan sebuah alat penerima sinyal satelit (penerima GPS) bisa diperoleh data posisi di suatu tempat dengan ketelitian beberapa meter dalam waktu nyata. Sebuah perangkat mobile memerlukan koneksi internet untuk mendapatkan sinyal langsung ke satelit dalam pencarian data GPS pada permukaan bumi. Dibutuhkan adanya sambungan ke internet untuk dapat menggunakan aplikasi Google Maps.

2.8.2. Google Maps API

Merupakan suatu fitur aplikasi yang memberikan fasilitas bagi user yang ingin mengintegrasikan Google Maps ke dalam website masing-masing dengan menampilkan data point milik sendiri. Dengan memakai Google Maps API, Google Maps bisa di-embed pada website eksternal. Supaya aplikasi Google Maps bisa keluar di website tertentu, dibutuhkan adanya API key. API (Application Programming Interface) key adalah kode unik yang digenerasikan untuk website tertentu, supaya server Google Maps dapat mengenali. API bukan hanya method dan set class atau signature dan fungsi yang sederhana. Melainkan tujuan utamanya

untuk mengatasi clueless dalam menciptakan software yang ukurannya besar, bermula dari hal yang sederhana hingga hal yang kompleks dan merupakan perilaku komponen yang tidak mudah dipahami. Perubahan ini dapat dipermudah dengan bantuan API.

2.9. Java

Java merupakan bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan-rekannya di suatu perusahaan perangkat lunak yang bernama Sun Microsystems, pada tahun 1991. Bahasa pemrograman ini mula-mula diinisialisasi dengan nama “Oak”, namun pada tahun 1995 diganti namanya menjadi “Java”.

2.9.1. Pengertian Java

Java berdiri di atas sebuah mesin penterjemah (interpreter) yang diberi nama Java Virtual Machine (JVM). JVM inilah yang akan membaca kode bit (bytecode) dalam file .class dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Oleh karena itu bahasa Java disebut sebagai bahasa pemrograman yang portable karena dapat dijalankan pada berbagai sistem operasi, asalkan pada sistem operasi tersebut terdapat JVM. Alasan utama pembentukan bahasa Java adalah untuk membuat aplikasi-aplikasi yang dapat diletakkan di berbagai macam perangkat elektronik, sehingga Java harus bersifat tidak bergantung pada platform (platform independent). Itulah yang menyebabkan dalam dunia pemrograman Java dikenal adanya istilah „write once, run everywhere“, yang berarti kode program hanya ditulis sekali, namun dapat 9 dijalankan di bawah kumpulan pustaka (platform) manapun, tanpa harus melakukan perubahan kode program.

2.9.2. Arsitektur Java

Secara arsitektur, Java tidak berubah sedikitpun sejak awal mula bahasa tersebut dirilis. Compiler Java (yang disebut dengan javac atau Java Compiler) akan mentransformasikan kode-kode dalam bahasa Java ke dalam suatu kode bit. Dimana bytecode adalah sekumpulan perintah hasil kompilasi yang kemudian dapat dieksekusi melalui sebuah mesin komputer abstrak, yang disebut dengan JVM (Java

Virtual Machine). JVM juga sering dinamakan sebagai interpreter, karena sifatnya yang selalu menerjemahkan kode-kode yang tersimpan dalam kode bit dengan cara baris demi baris. Untuk menjalankan program Java, maka file dengan ekstensi .java harus dikompilasi menjadi file kode bit. Dimana untuk menjalankan kode bit tersebut dibutuhkan JRE (Java Runtime Environment) yang memungkinkan pemakai untuk menjalankan program Java, hanya menjalankan, tidak untuk membuat kode baru lagi. JRE berisi JVM dan pustaka Java yang digunakan.

Dalam pembuatan aplikasi android diperlukan dasar dalam pemrograman Java. Jadi, untuk mengembangkan suatu aplikasi android developer dituntut untuk memiliki dasar dalam pemrograman java, sehingga tidak merasa kebingungan saat mengerjakan suatu aplikasi berbasis mobile. [4]

2.10. PHP

PHP pertama kali dibuat pada musim gugur tahun 1994 oleh Rasmus Lerdoff (rasmus@php.net), awalnya digunakan pada web sitenya untuk mencatat siapa saja yang berkunjung dan melihat biodatanya. Versi pertama yang direlease tersedia pada awal tahun1995, dikenal sebagai tools Personal Home Page, yang terdiri atas engine parser yang sangat sederhana yang hanya mengerti beberapa maksro khusus dan sejumlah utilitas yabg sering digunakan pada halaman – halaman web, seperti buku tamu, counter pengunjung, dan lainnya. Parser di program ulang pada pertengahan 1995 dan diberi nama PHP/FI versi 2.0. FI berasal dari paket Rasmus lainnya yang ditulis untuk menginterpretasi data dari form, yang kemudian dikombinasikan dengan tool Personal Home Page dan ditambahkan dukungan untuk database MSQl

2.10.1. Sejarah Singkat PHP

PHP pertama kali dibuat pada musim gugur tahun 1994 oleh Rasmus Lerdoff (rasmus@php.net), awalnya digunakan pada web sitenya untuk mencatat siapa saja yang berkunjung dan melihat biodatanya. Versi pertama yang direlease tersedia pada awal tahun1995, dikenal sebagai tools Personal Home Page, yang terdiri atas engine parser yang sangat sederhana yang hanya mengerti beberapa maksro khusus dan sejumlah utilitas yabg sering digunakan pada halaman – halaman

web, seperti buku tamu, counter pengunjung, dan lainnya. Parser di program ulang pada pertengahan 1995 dan diberi nama PHP/FI versi 2.0. FI berasal dari paket Rasmus lainnya yang ditulis untuk menginterpretasi data dari form, yang kemudian dikombinasikan dengan tool Personal Home Page dan ditambahkan dukungan untuk database MSQL

2.10.2. Bootstrap

Bootstrap User Interface adalah sebuah library desain untuk membangun aplikasi web yang responsif. Bootstrap adalah framework yang populer pada aplikasi web. Bootstrap diciptakan oleh dua orang programmer yang bekerja pada sebuah perusahaan sosial media Twitter. Sejak diluncurkan pada tahun 2011, bootstrap telah menjadi sebuah proyek yang hanya berbasis CSS menjadi sebuah framework yang lengkap karena berisi css, javascript icon dan lain-lain.

2.10.3. CSS

Cascading Style Sheet (CSS) merupakan aturan untuk mengatur beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman. Sama halnya styles dalam aplikasi pengolahan kata seperti Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file). Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML dan XHTML. CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran border, warna border, warna hyperlink, warna mouse over, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa style sheet yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda. [5]

2.11. 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

Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman 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.

JSON terbuat dari dua skruktur, yaitu :

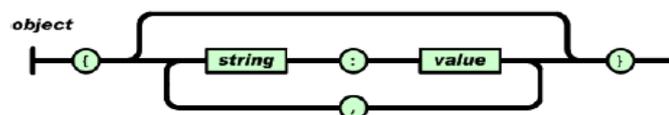
- a. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.
- b. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan 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 pemrograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini.

Berikut ini adalah bentuk JSON sebagai berikut :

1. Objek

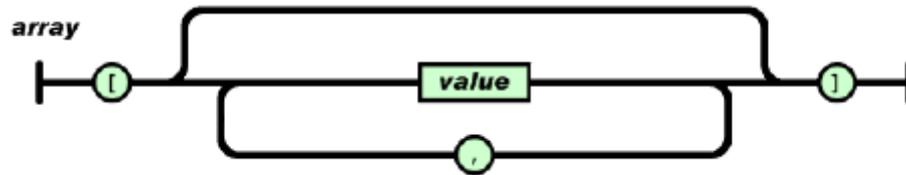
Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).



Gambar 2.4 Alur bentuk objek

2. Array

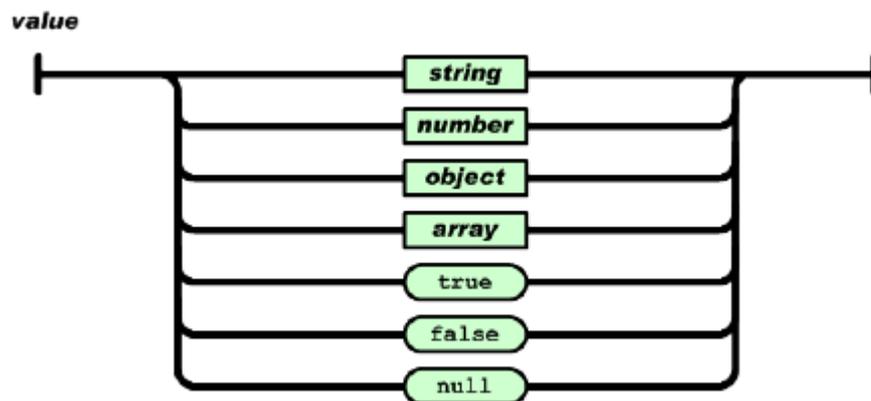
Array adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).



Gambar 2.5 Alur bentuk array

3. Value

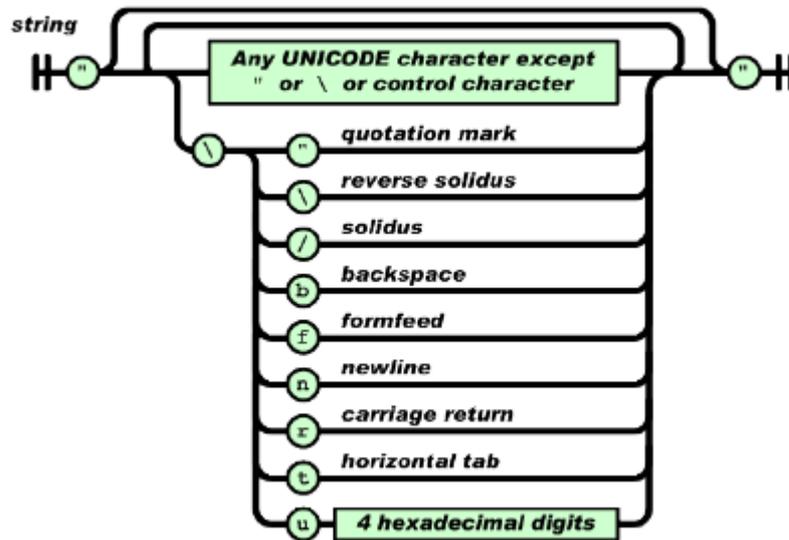
Value dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.



Gambar 2.6 Alur bentuk value

4. String

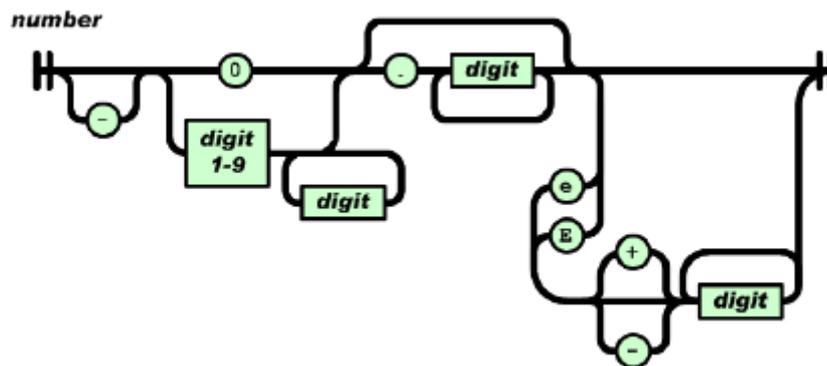
String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.



Gambar 2.7 Alur bentuk string

5. Number

Number (angka) sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan. [6]



Gambar 2.8 Alur bentuk number

2.12. MySQL

MySQL merupakan software RDBMS (Relational Database Management System) atau server database yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user multi-user), dan dapat melakukan suatu proses secara sinkron atau bersamaan

(multi-threaded). Saat ini, MySQL banyak digunakan di berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai keindustry, baik industri kecil, menengah, maupun besar.

Lisensi MySQL terbagi menjadi dua. Anda dapat menggunakan MySQL sebagai produk open source di bawah GNU (General Public License) secara gratis atau dapat membeli lisensi dari versi komersialnya. MySQL versi komersial tentu memiliki nilai lebih atau kemampuan-kemampuan yang tidak disematkan pada versi gratis. Pada kenyataannya, untuk keperluan industri menengah kebawah, versi gratis masih dapat digunakan dengan baik.

2.13. Structured Query Language (SQL)

SQL adalah pendekatan dari Structural Query Language, yang merupakan Bahasa atau kumpulan perintah standar yang digunakan untuk berkomunikasi dengan database. Perintah dalam SQL diklasifikasikan menjadi lima bagian besar,

Yaitu :

1. *Data Definition Language (DDL)*

Data Definition Language (DDL) adalah sekumpulan perintah SQL yang berkaitan dengan pembuatan, perubahan dan penghapusan database prosedur/fungsi, trigger, dan sebagainya.

Perintah SQL yang termasuk kategori DDL adalah :

- a. CREATE DATABASE, Berfungsi untuk membuat database dan objek-objek di dalam database berikut sintaks yang digunakan.
- b. CREATE TABLE, Berfungsi untuk membuat tabel di dalam database, berikut sintaks yang digunakan.
- c. CREATE INDEX , digunakan untuk membuat index berikut sintaks yang digunakan.
- d. CREATE VIEW, digunakan untuk membuat view. Berikut sintaks yang digunakan.

- e. DROP DATABASE, digunakan untuk menghapus basis data, berikut sintaks yang digunakan.
- f. DROP TABLE, digunakan untuk menghapus tabel. Berikut sintaks yang digunakan.
- g. DROP INDEX, digunakan untuk menghapus index. Berikut sintaks yang digunakan.
- h. DROP VIEW, digunakan untuk menghapus view. Berikut sintaks yang digunakan.
- i. ALTER TABLE, digunakan untuk mengubah struktur suatu tabel. Berikut beberapa sintaks yang digunakan.

2. *Data Manipulation Language (DML)*

Data Manipulation Language (DML) adalah kumpulan perintah SQL yang berkaitan dengan data atau isi dari suatu tabel. Dengan perintah-perintah di dalam DML, kita dapat memanipulasi (menambah, mengubah, dan menghapus) data yang terdapat pada suatu tabel secara mudah.

3. *Data Control Language (DCL)*

Data Control Language (DCL) adalah salah satu komponen SQL yang berfungsi untuk mengontrol hak akses user. Perintah yang termasuk ke dalam DCL adalah GRANT dan REVOKE. Berikut penjelasannya.

- a. GRANT, Perintah GRANT digunakan untuk memberikan hak akses (privilege) kepada user tertentu. Melalui perintah hak akses semacam ini, seseorang user memiliki keterbatasan dalam menggunakan database sehingga data akan aman dari pihak-pihak yang tidak berkepentingan. Untuk melihat daftar hak akses yang dimiliki oleh seorang user, gunakan perintah SHOW GRANTS. Untuk mengeksekusi perintah GRANT, anda perlu memiliki hak akses GRANT OPTION yang sebelumnya diberikan oleh administrator kepada anda.
- b. REVOKE, Perintah REVOKE merupakan kebalikan dari perintah GRANT, yang berfungsi untuk mencabut salah satu atau beberapa hak akses dari user

tertentu di dalam database. Sama seperti perintah GRANT, untuk menjalankan perintah REVOKE anda perlu memiliki hak akses GRANT OPTION.

4. *Transaction control Language (TCL)*

Selain pengolahan hak akses user, dalam SQL kita juga dapat mengontrol transaksi data yang telah dilakukan. Perintah-perintah yang berkaitan dengan pengontrolan transaksi data digolongkan ke dalam Transactional Control Language (TCL). Perintah yang termasuk ke dalam TCL adalah COMMIT dan ROLLBACK. Berikut penjelasannya.

- a. COMMIT, perintah COMMIT berfungsi untuk menyimpan perubahan-perubahan yang dilakukan terhadap database (melalui perintah INSERT, UPDATE, atau DELETE) secara permanent.
- b. ROLLBACK, perintah ROLLBACK merupakan kebalikan dari perintah COMMIT, yang berfungsi untuk membatalkan transaksi atau perubahan-perubahan yang telah dilakukan ke dalam database (melalui perintah INSERT, UPDATE, atau DELETE). Dengan melakukan pembatalan transaksi, data di dalam database akan kembali ke keadaan awal (keadaan sebelum dilakukan perubahan). Sehingga dengan kata lain, perubahan yang dilakukan tidak akan pernah disimpan ke dalam database.

2.14. Web Server

Web server adalah sebuah perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol komunikasi HTTP atau HTTPS atas berkas-berkas yang terdapat pada suatu situs web dalam layanan ke pengguna dengan menggunakan aplikasi tertentu.

2.14.1. Web Service

Web service adalah antarmuka yang mendeskripsikan sekumpulan operasi yang dapat diakses dalam sebuah jaringan melalui pesan XML yang telah distandartkan. W3C mendefinisikan web service sebagai sebuah sistem perangkat lunak yang dirancang untuk mendukung inter operasi mesin ke mesin di sebuah jaringan.

2.15. OOP (Object Oriented Programming)

Object Oriented Program (OOP) merupakan paradigma baru dalam rekayasa software yang didasarkan pada obyek dan kelas. (Ronald J.N., 1996). Diakui para ahli bahwa objectoriented merupakan metodologi terbaik yang ada saat ini dalam rekayasa software. Object-oriented memandang software bagian per bagian dan menggambarkan satu bagian tersebut dalam satu obyek.

Teknologi obyek menganalogikan sistem aplikasi seperti kehidupan nyata yang didominasi oleh obyek. Dengan demikian keunggulan teknologi obyek adalah bahwa model yang dibuat akan sangat mendekati dunia nyata yang masalahnya akan dipecahkan oleh sistem yang dibangun. Model obyek, atribut dan perlakuannya bisa langsung diambil dari obyek yang ada di dunia nyata

Ada 4 (empat) prinsip dasar dari pemrograman berorientasi obyek yang menjadi dasar kemunculan UML, yaitu abstraksi, enkapsulasi, modularitas dan hirarki. Berikut dijelaskan satu persatu secara singkat.

1. Abstraksi memfokuskan perhatian pada karakteristik obyek yang paling penting dan paling dominan yang bisa digunakan untuk membedakan obyek tersebut dari obyek lainnya.
2. Enkapsulasi menyembunyikan banyak hal yang terdapat dalam obyek yang tidak perlu diketahui oleh obyek lain. Dalam praktek pemrograman, enkapsulasi diwujudkan dengan membuat suatu kelas interface yang akan dipanggil oleh obyek lain, sementara didalam obyek yang dipanggil terdapat kelas lain yang mengimplementasikan apa yang terdapat dalam kelas interface.
3. Modularitas membagi sistem yang rumit menjadi bagian-bagian yang lebih kecil yang bisa mempermudah developer memahami dan mengelola obyek tersebut.
4. Hirarki berhubungan dengan abstraksi dan modularitas, yaitu pembagian berdasarkan urutan dan pengelompokkan tertentu. Misalnya untuk menentukan obyek mana yang berada pada kelompok yang sama, obyek mana yang merupakan komponen dari obyek yang memiliki hirarki lebih

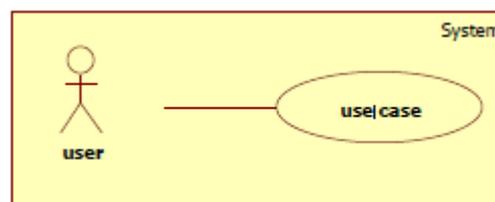
tinggi. Semakin rendah hirarki obyek berarti semakin jauh abstraksi dilakukan terhadap suatu obyek. [8]

2.16. UML (Unified Modelling Language)

UML (Unified Modeling Language) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (ObjectOriented). (Sri Dharwiyanti, Pengantar Unified Modeling Language, 2003). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. [9]

2.16.1. Use Case Diagram

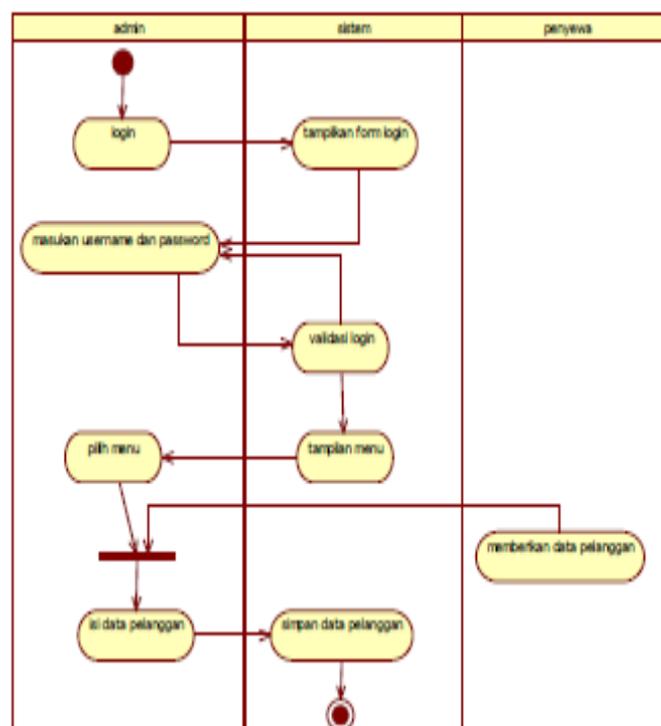
Use case diagram adalah proses pemodelan fungsi-fungsi sistem dalam konteks peristiwa-peristiwa bisnis, siapa yang mengawali dan bagaimana sistem itu memproses hal tersebut. Use case diagram memperlihatkan hubungan diantara aktor dan use case. Aktor merepresentasikan seorang user atau sub sistem lain yang akan berinteraksi dengan sistem. Sedangkan use case merupakan urutan kejadian yang menggambarkan interaksi antara user dengan sistem fungsionalitas sistem didefinisikan ke dalam use case dari sudut eksternal sistem yang berguna untuk uji kelayakan fungsionalitas. Gambar berikut dibawah ini menampilkan diagram UML :



Gambar 2.9 Use case diagram

2.16.2. Activity Diagram

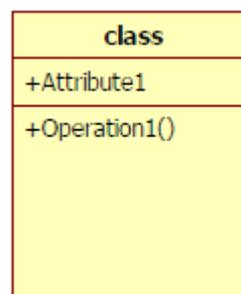
Merupakan penggambaran rangkaian aliran dari aktifitas, dipakai untuk menjelaskan aktifitas yang dibentuk dalam suatu operasi, sehingga dapat juga digunakan untuk aktifitas lainnya. Diagram activity ini menjelaskan alur kerja sebuah sistem. Activity diagram mirip dengan state diagram karena sejumlah aktivitas menggambarkan keadaan suatu proses dengan memperlihatkan urutan aktivitas yang dijalankan baik berupa pilihan maupun paralel. Diagram ini juga berguna untuk menganalisa sebuah use case dengan menggambarkan aksi-aksi yang diperlukan dan kapan aksi-aksi tersebut dijalankan, menjelaskan urutan algoritma yang kompleks dan memodelkan sejumlah aplikasi dengan proses paralel. Gambar berikut dibawah ini menampilkan activity diagram :



Gambar 2.10 Activity Diagram

2.16.3. Class Diagram

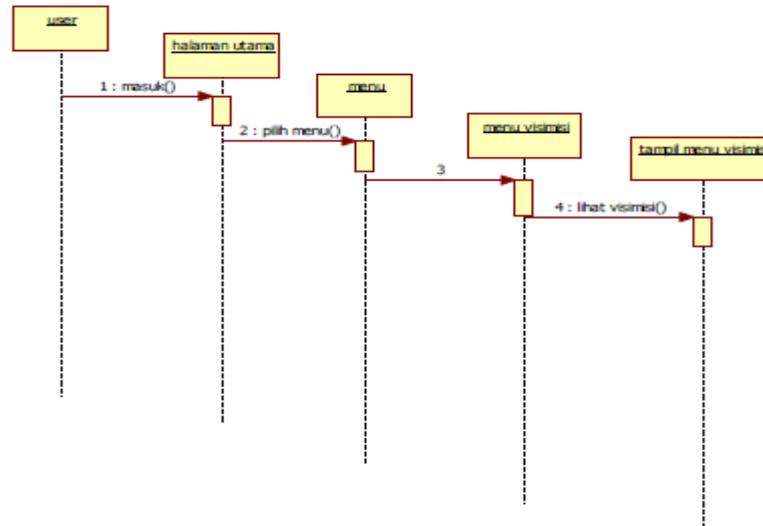
Merupakan penggambaran struktur statis class di dalam sistem. Class diagram merepresentasikan sesuatu yang ditangani oleh sistem. Diagram ini digunakan untuk menggambarkan tipe-tipe objek dan hubungannya dalam sebuah sistem. Class diagram memodelkan struktur class dan isinya dengan menggunakan elemen-elemen model seperti class, package dan objek. Class terdiri dari tiga bagian yaitu nama class, atribut dan operasi. Class yang didefinisikan secara global dapat diakses oleh objek diluar class tersebut. Gambar berikut dibawah ini menampilkan Class Diagram :



Gambar 2.11 Class Diagram

2.16.4. Sequence Diagram

Merupakan penggambaran kolaborasi dinamis antara sejumlah objek. Diagram ini menjelaskan bagaimana objek berinteraksi dengan lainnya dengan cara mengirim dan menerima pesan. Sequence diagram memiliki dua sumbu:sumbu vertikal dan sumbu horizontal. Sumbu vertikal putus-putus merepresentasikan "lifetime" objek dan sumbu horizontal menunjukkan sekumpulan objek. Sekumpulan objek diagram ini juga menyatakan interaksi khusus diantara objek yang terjadi pada beberapa tempat selama fungsi tertentu dijalankan. Komunikasi diantara objek direpresentasikan dengan garis horizontal disertai dengan nama operasinya. Gambar berikut dibawah ini menampilkan Sequence Diagram :



Gambar 2.12 Sequence Diagram