

BAB II LANDASAN TEORI

2.1 Android

Android merupakan sebuah sistem operasi yang bersifat mobile dan berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti gerak. Antarmuka pengguna Android umumnya berupa manipulasi langsung, menggunakan gerakan sentuh yang serupa dengan tindakan nyata, misalnya menggeser, mengetuk, dan mencabit untuk memanipulasi objek di layar, serta papan ketik virtual untuk menulis teks.

2.2 Aplikasi Mobile

Aplikasi mobile saat ini sangat dibutuhkan karena alat-alat telekomunikasi yang tersebar di seluruh dunia membutuhkan aplikasi-aplikasi yang dapat mempermudah pekerjaan penggunanya dimanapun dan kapanpun terutama dalam hal informasi. Aplikasi ini dapat diakses melalui perangkat nirkabel seperti pager, seperti telepon seluler dan PDA.

2.2.1 Pengertian 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.

2.2.2 Mobile Registration

Sebagian kecil memori komputer yang dipakai untuk tempat penampungan data dengan ketentuan bahwa data yang terdapat dalam register dapat diproses dalam berbagai operasi dengan melihat berapa besar kemampuan menampung register tersebut dan mobile registration merupakan suatu pendaftaran yang dapat dilakukan secara luas dan mudah.

2.3 Sistem

Tata Sutabri (2012) mendefinisikan Sistem adalah sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.

Menurut Sutabri (2012) sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu :

1. Komponen sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar atau sering disebut “supra sistem”.

2. Batasan sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan luar sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar

tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau interface. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan sistem (*Input*)

Energi yang dimasukkan ke dalam sistem tersebut masukkan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contoh, di dalam suatu unit sistem komputer, "Program" adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan "Data" adalah signal input untuk diolah menjadi informasi.

6. Keluaran sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

7. Pengolahan sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem

tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran dan tujuan yang telah direncanakan.

2.4 Pengertian PHP

Yunita Trimarsiah (2009, p.8) mendefinisikan bahwa PHP adalah singkatan dari Personal Home Page yang merupakan bahasa standar yang di gunakan dalam dunia website. PHP adalah bahasa programan yang berbentuk script yang di letakkan di dalam web server. PHP dapat diartikan sebagai Hypertext Preeprocessor, ini merupakan bahasa yang hanya dapat berjalan pada server yang hasilnya dapat ditampilkan pada klien.

2.5 Konsep Dasar Informasi

2.5.1 Data

Edhy Sutanta (2004) data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam kelompok lambang tertentu yang tidak acak yang menunjukkan jumlah, tindakan, atau hal.

2.5.2 Informasi

Tata Sutabri (2012) Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari tak berguna menjadi berguna bagi penerimanya.

Nilai informasi berhubungan dengan keputusan maka informasi menjadi tidak diperlukan keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks sebuah keputusan.

2.6 World Wide Web (WWW)

Kustiyahningsih & Anamisa (2011) mengatakan *World Wide Web* (WWW), lebih dikenal dengan web yang merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet dengan fasilitas *hypertext* untuk menampilkan data berupa *text*, gambar, suara, animasi dan data multimedia lainnya. Sehingga web pada awalnya adalah ruang informasi dalam internet, dengan menggunakan teknologi *hypertext*, pemakai dituntut menemukan informasi dengan mengikuti *link* yang disediakan dalam dokumen web yang ditampilkan dalam *web browser*. Situs atau web dapat dikategorikan menjadi dua yaitu :

1. Web Statis

Web statis adalah web yang menampilkan informasi-informasi yang sifatnya statis(tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan web tersebut.

2. Web Dinamis

Web dinamis adalah web yang menampilkan informasi serta dapat berinteraksi dengan pengguna. Web yang dinamis memungkinkan pengguna untuk berinteraksi menggunakan form sehingga dapat mengolah informasi yang ditampilkan. Web dinamis bersifat interaktif, tidak kaku dan terlihat lebih indah.

2.7 HTML5

Indra Yatini (2014), mendefinikan HTML5 merupakan teknologi inti dari Internet adalah bahasa *markup* untuk penataan dan penyajian konten world wide web. Tujuan utama HTML 5 adalah meningkatkan bahasa dengan dukungan multimedia yang tetap mudah dibaca dan dimengerti.

Fitur baru HTML 5 antara lain;

1. Unsur `<canvas>` untuk menggambar 2D.
2. Unsur `<video>` dan `<audio>` untuk media pemutaran.
3. Dukungan untuk penyimpanan lokal.

4. Konten baru dengan elemen spesifik seperti <article>, <footer>, <header>, <nav>, <section>.

Untuk menambah keluwesan pemformatan, pada HTML5 telah dispesifikasikan pengkodean application programming interfaces (APIs). Geolocation harus dilaksanakan pada browser. Pedoman umum adalah bahwa masing-masing browser harus berusaha untuk menyeimbangkan akurasi dengan daya konsumsi pertimbangan.

2. 8 Perangkat Lunak yang Digunakan

Untuk membangun *website* diperlukan beberapa perangkat lunak yang digunakan dalam membangun *website* tersebut. Beberapa perangkat lunak yang digunakan adalah sebagai berikut :

2.8.1 Notepad ++

Notepad++ merupakan sebuah penyunting teks dan penyunting kode sumber yang tersedia dalam beberapa sistem operasi seperti Windows, Linux, dan Mac OS X. Notepad ++ menggunakan komponen Scintilla untuk dapat menampilkan dan menyunting teks dan berkas kode sumber berbagai bahasa pemrograman. Notepad ++ didistribusikan sebagai perangkat lunak bebas. Proyek ini dilayani oleh *Sourcefoge.net* dengan telah diunduh lebih dari 27 juta kali dan dua kali memenangkan penghargaan *SorceForge Community Choice Award for Best Developer Tools*.

2.8.2 Framework Bootstrap

Menurut Husein (2013, p.4) *Bootstrap* merupakan *framework* ataupun *tools* untuk membuat aplikasi *website* ataupun situs *web responsive* secara cepat, mudah dan gratis, karena *website* yang dibangun oleh peneliti merupakan *website* yang dapat diakses dalam perangkat *mobile* ataupun *personal computer*. *Bootstrap* terdiri dari CSS dan HTML untuk menghasilkan *Grid*, *Layout*, *Typography*, *Table*, *Form*, *Navigation*, dan lain lain. Di dalam *bootstrap* juga sudah terdapat *jquery plugins* untuk menghasilkan komponen

UI yang cantik seperti *Transitions, Modal, Dropdown, Scrollspy, Tooltip, Tab, Popever, Alert, Button, Carousel*, dan lain lain.

2.8.3 XAMPP

Library IBM (2004), XAMPP merupakan aplikasi server yang menggabungkan beberapa aplikasi *server* yang biasa digunakan di *web server*. Berikut beberapa komponen – komponen yang terdapat pada XAMPP, yaitu Apache (*web server*), MySQL (*database server*), Filezila FTP *server*, Mercury Mail (*mail server*), phpMyAdmin (*web-based interface MySQL*).

2.9 Database

2.9.1 Definisi Database

Menurut Yasin (2012) mengatakan Basis data (*database*) adalah kumpulan informasi yang akan disimpan didalam komputer secara sistematis, sehingga dapat digunakan oleh suatu program komputer untuk memperoleh informasi dari basis data tersebut. Basis data adalah sekumpulan data yang terhubung satu sama lain secara logika dan suatu deskripsi data yang dirancang untuk memenuhi kebutuhan informasi suatu organisasi. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau informasi yang akan disimpan. *Database* merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna atau use.

2.9.2 Tujuan Database

1. Kecepatan dan Kemudahan (*Speed*)

Pemanfaatan basis data memungkinkan kita untuk dapat menyimpan data atau melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah.

2. Efisiensi Ruang Penyimpanan(*Space*)

Karena keterkaitan erat antara kelompok dalam basis data, maka reduksi data pasti selalu ada. Dengan basis data, efisiensi/optimalisasi penggunaan ruang penyimpanan dapat dilakukan karena kita dapat melakukan penekanan jumlah reduksi data, baik menerapkan sejumlah pengkodean atau membuat relasi-relasi antar kelompok data yang saling berhubungan.

3. Keakuratan (*Accuracy*)

Pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan/batasan tipe data, domain data, keunikan data dan sebagainya yang secara ketat dapat diterapkan dalam sebuah basis data, sangat berguna untuk menekan ketidakakuratan/penyimpanan data.

4. Ketersediaan (*Availability*)

Pertumbuhan data sejalan waktu akan semakin membutuhkan ruang penyimpanan yang besar. Padahal tidak semua data selalu kita gunakan/butuhkan. Karena itu kita dapat melakukan pemilihan data, sehingga data yang sudah jarang kita gunakan dapat kita pindahkan ke dalam media penyimpanan offline.

5. Kelengkapan (*Completeness*)

Untuk mengkomodasi kebutuhan kelengkapan data yang semakin berkembang, maka kita tidak hanya dapat menambah record-record data, tetapi juga dapat melakukan perubahan struktur dalam basis data, baik dalam penambahan objek baru (tabel) atau dengan penambahan *field-field* baru pada suatu tabel.

6. Keamanan (*Security*)

Ada sejumlah sistem pengolahan basis data yang tidak menerapkan aspek keamanan dalam sebuah basis data.

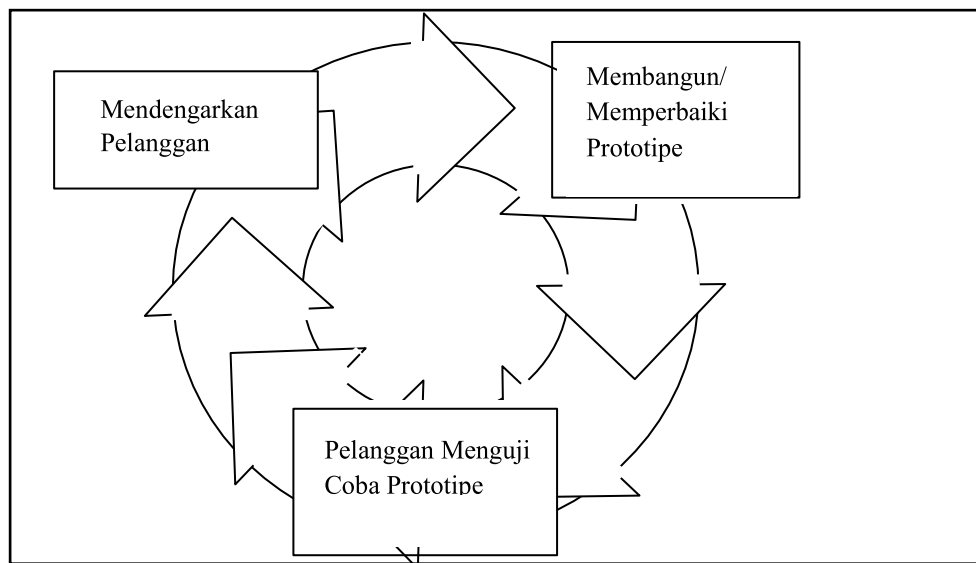
7. Kebersamaan pemakai (*Shareability*)

Pemakai basis data seringkali tidak terbatas pada satu pemakai saja atau di satu lokasi saja oleh satu sistem aplikasi.

2.10 Metode Pengembangan Perangkat Lunak

2.10.1 Metode Prototype

Pressman (2012) menyatakan bahwa *Prototyping Paradigma* dimulai dengan pengumpulan kebutuhan, pengembang bertemu dengan pengguna dan mengidentifikasi objektif keseluruhan dari perangkat lunak, selanjutnya mengidentifikasi segala kebutuhan yang diketahui secara garis besar di mana definisi-definisi lebih jauh merupakan keharusan, kemudian dilakukan perancangan kilat, lalu diakhiri dengan evaluasi *prototyping* yang dapat dilihat pada gambar 2.5 berikut :



Gambar 2.1 Model Prototype.

1. Pengumpulan kebutuhan

Developer dan klien bertemu untuk menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Detail kebutuhan mungkin tidak dibicarakan disini, pada awal pengumpulan kebutuhan.

Selanjutnya peneliti akan melakukan analisis terhadap data apa saja yang dibutuhkan, seperti analisis terhadap sistem yang berjalan, analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, dan analisis kebutuhan materi pembelajaran.

2. Perancangan

Perancangan dilakukan dengan cepat dan rancangan mewakili semua aspek software yang diketahui, dan rancangan ini menjadi dasar pembuatan *prototype*. Dalam tahap ini peneliti akan membangun sebuah versi *prototype* yang dirancang kembali dimana masalah-masalah tersebut diselesaikan

3. Evaluasi *prototype*

Pada tahap ini, calon pengguna mengevaluasi *prototype* yang dibuat dan digunakan untuk memperjelas kebutuhan *software*. *Software* yang sudah jadi dijalankan dan akan dilakukan perbaikan apabila kurang memuaskan. Perbaikan termasuk dalam memperbaiki kesalahan atau kerusakan yang tidak ditemukan pada langkah sebelumnya.

2.10.2 Unified Modeling Language (UML)

Yasin (2012) mendefinisikan *Unified Modelling Language* (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak, UML menawarkan sebuah standar untuk merancang model sebuah sistem. Tujuan Penggunaan UML yaitu untuk memodelkan suatu sistem yang menggunakan konsep berorientasi objek dan menciptakan bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

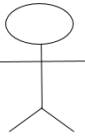
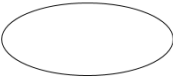

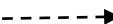
Menurut Verdi Yasin (2012,p.268) tipe-tipe diagram UML adalah sebagai berikut :

1. Use Case Diagram

Use case diagram adalah gambar dari beberapa atau seluruh aktor dan *use case* dengan tujuan yang mengenali interaksi mereka dalam suatu sistem. *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dan sistem.

Dalam *use case diagram* terdapat istilah seperti aktor, *use case* dan *case relationship*. Penjelasan simbol pada tabel 2.1.

Tabel 2.1 Simbol Use Case.


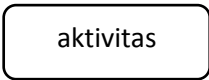
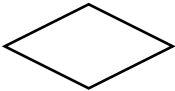


Simbol	Keterangan
	Aktor : Seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang dikembangkan.
	<i>Use case</i> : perangkat tertinggi dari fungsionalitas yang dimiliki sistem.
	<i>Association</i> : adalah relasi antara aktor dan <i>use case</i> .
	<i>Generalisasi</i> : untuk memperlihatkan struktur pewaris yang terjadi.

2. Activity Diagram

Activity diagram menggambarkan rangkaian aliran dari aktifitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi. *Activity Diagram* berupa flow chart yang digunakan untuk memperlihatkan aliran kerja dari sistem.

Notasi yang digunakan dalam *activity diagram* adalah sebagai berikut :

Tabel 2.2 Simbol *Activity Diagram*.

Simbol	Deskripsi
Status awal 	Status awal aktifitas system, sebuah diagram aktivitas memiliki sebuah status awal
aktivitas 	Aktivitas yang di lakukan system, aktivitas biasanya diawali dengan kata kerja
Percabangan/decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
status akhir 	Status akhir yang di lakukan system, sebuah diagram aktivitas memiliki sebuah status akhir

3) *Sequence Diagram*

Sequence diagram menggambarkan kolaborasi dinamis antara sejumlah dan untuk menunjukkan rangkaian pesan yang dikirim antar objek juga interaksi antar objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. *Sequence* diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu.

Dalam *sequence* diagram terdapat 2 simbol yaitu :

- a. Actor, untuk menggambarkan pengguna sistem.
- b. *Lifeline*, untuk menggambarkan kelas dan objek.

4) *Class Diagram*

Class diagram menggambarkan dtruktur data dan desripsi *class*, *package*, dan objek beserta hubungan satu sama lain. *Class* diagram berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain. *Class* memiliki 3 area pokok yaitu nama, atribut dan metode.