

BAB II

LANDASAN TEORI

2.1 Data

Data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam sekelompok lambing tertentu yang tidak acak, yang menunjukkan jumlah, tindakan, atau hal. Data dapat merupakan catatan-catatan dalam kertas, buku, atau tersimpan sebagai file dalam basis data[1]. Data dapat dianalogikan dengan sejumlah blok yang biasa digunakan anak-anak untuk membentuk berbagai struktur sesuai dengan imajinasi mereka. Data yang terdapat dalam suatu organisasi dapat diolah sehingga menghasilkan informasi yang bisa digunakan oleh para eksekutif di organisasi untuk membantu pengambilan keputusan[2].

2.2 Sistem

Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama. Secara garis besar, sebuah *system* informasi terdiri atas tiga komponen utama. Ketiga komponen tersebut mencakup *softower*, *hardware*, dan *brainwar*. Ketiga komponen ini saling berkaitan satu sama lain[3]. Pada dasarnya, sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Sebagai gambaran, jika dalam sebuah sistem terdapat elemen yang tidak memberikan manfaat dalam mencapai tujuan yang sama, maka elemen tersebut dapat dipastikan bukanlah bagian dari sistem[2].

2.3 Informasi

Inoformasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang[1]. Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti, dan manfaat. Proses pengelolaan ini memerlukan teknologi [3].

2.4 Sistem Informasi

Berdasarkan definisi mengenai sistem dan informasi yang telah dijelaskan di atas, maka dapat dinyatakan bahwa sistem informasi merupakan gabungan dari empat bagian utama. Keempat bagian utama tersebut mencakup perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur, dan Sumber Daya Manusia (*SDM*) yang terlatih. Keempat bagian utama ini saling berkaitan untuk menciptakan sebuah sistem yang dapat mengolah data menjadi informasi yang bermanfaat [3].

2.5 Rancang Bangun

Rancang bangun merupakan kegiatan untuk menterjemahkan hasil analisa kedalam bentuk bahasa program dan implementasi komponen-komponen sistem serta menciptakan sistem baik secara keseluruhan ataupun sebagian[4].

2.6 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat piranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.[5].

Secara garis besar, arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut:

- a. *Applications* dan *Widgets* ini adalah layer dimana berhubungan dengan aplikasi saja, di mana biasanya *download* aplikasi dijalankan kemudian dilakukan instalasi dan jalankan aplikasi tersebut.
- b. *Applications frameworks* ini adalah layer di mana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena

pada layer inilah aplikasi dapat dirancang dan dibuat, seperti *contentproviders* yang berupa sms dan panggilan telepon.

- c. *Libraries* ini adalah layer di mana 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 dan SSL.
- d. *Android Run Time Layer* yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan *Implementasi Linux*.
- e. *Linux Kernel* adalah layer di mana inti dari *operating system* dari 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.

2.6.1 Android Studio

Android Studio yang merupakan 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 saat membuat aplikasi Android, misalnya sistem versi berbasis Gradle yang fleksibel, emulator yang cepat dan kaya fitur, lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android, instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru, template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh, alat pengujian dan kerangka kerja yang ekstensif, alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain, dukungan C++ dan NDK, dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.[6].

2.7 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di *Sun Microsystems* saat ini merupakan bagian dari *Oracle* dan dirilis tahun 1995. Bahasa ini banyak

mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin.

Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, “Tulis sekali, jalankan di mana pun”. Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.[7]

2.8 M-Learning

M-Learning adalah pembelajaran yang terjadi ketika pelajar tidak pada lokasi yang tetap atau telah ditentukan, dan pembelajaran yang terjadi ketika pelajar mengambil keuntungan dari kesempatan belajar yang ditawarkan oleh teknologi mobile [8]. *M-Learning* adalah pembelajaran yang unik karena pelajar dapat mengakses materi pembelajaran, arahan dan aplikasi yang berkaitan dengan pembelajaran, kapanpun dan dimanapun melalui perangkat telekomunikasi seperti handphone, smartphone, atau tablet [9].

2.9 Basis Data (Database)

Elemen basis data pada sistem informasi berfungsi sebagai media untuk penyimpanan data dan informasi yang dimiliki oleh sistem informasi bersangkutan. Setiap aplikasi dan sistem yang memiliki data di dalamnya (*dengan disertai proses manipulasi data berupa insert, delete, edit/update*), pasti memiliki sebuah basis data [3]. Maka istilah basis data dapat dipahami sebagai suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangka data, data di simpan dengan cara-cara tertentu sehingga mudah digunakan atau di tampilkan kembali [1].

2.9.1 MY SQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya. SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasianbasisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis[10].

2.10 ERD

ERD adalah diagram yang menggambarkan keterkaitan antara *table* bederta dengan *field-field* di dalamnya dalam suatu database sistem. Sebuah database memuat minimal sebuah *table* dengan beberapa *field* (kolom) [3]. Bagi perancang/analisa sistem, ERD berguna untuk memodelkan sistem yang nantinya basis datanya akan dikembangkan. Model ini juga membantu perancang/analisa sistem pada saat melakukan analisa dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antar data di dalamnya [1].

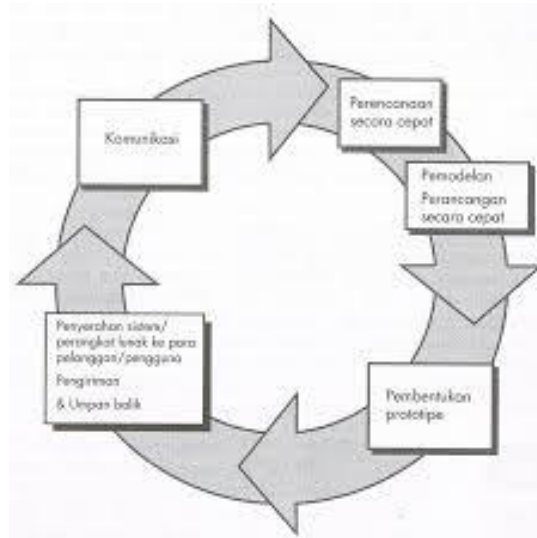
2.11 Metode Prototipe

Prototype merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program yang cepat dan bertahap sehingga dapat di evaluasi oleh pemakai [2].

Secara garis besar, sasaran prototype adalah sebagai berikut.

- a. Mengurangi waktu sebelum pemakai melihat sesuatu yang konkret dari usaha pengembangan sistem.
- b. Menyediakan umpan balik yang cepat dari pemakai kepada pengembang.
- c. Membantu menggambarkan kebutuhan pemakai dengan kesalahan yang lebih sedikit.

- d. Meningkatkan pemahaman pengembangan dan pemakai terhadap sasaran yang seharusnya dicapai oleh sistem.
- e. Menjadikan keterlibatan pemakai sangat bearti dalam analisa dan desain sistem.




Gambar 1. Protipe

2.12 Unified Modeling Language (UML)

UML adalah standarisasi internasional untuk notaris dalam bentuk grafik, yang menjelaskan tentang analisis dan desain perangkat lunak yang di kembangkan dengan pemrograman berorientasi objek. Sebuah UML memiliki setidaknya Sembilan buah diagram di dalamnya, namun umumnya digunakan tiga buah diagram saja [3].

- a. *Use Case Diagram* menggambarkan aliran kegiatan dan proses bisnis yang dilakukan oleh pengguna (*actor*). Pada diagram *use case* ini, setiap kegiatan digambarkan kedalam sebuah *use case* berbentuk lonjong dan terdapat minimal seorang actor.

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .







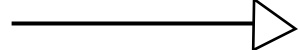
2.		<i>Include</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
5.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.

Table 1. Use Case Diagram

- b. *Class* diagram menggambarkan struktur kelas yang terdapat pada perangkat lunak yang akan dibangun.

NO	Simbol	Nama	Keterangan
1		Class	Menambahkan kelas baru pada Diagram
2		<i>Association</i>	Menggambar relasi asosiasi
3		<i>Generalization</i>	Menggambar relasi

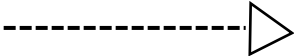






			generalisasi
4		<i>Realize</i>	Menggambarkan relasi realisasi
5		<i>Association class</i>	Menghubungkan kelas asosiasi (association class) pada suatu relasi asosiasi
6		<i>Return message</i>	Menggambarkan pengembalian dari pemanggilan prosedur
7		<i>Aggregation</i>	Menggambarkan relasi agregasi

Table 2. *Class Diagram*

c. *Sequence* diagram menggambarkan *sequence* (aliran) pengiriman pesan yang terjadi di aplikasi, sebagai bentuk interaksi dengan pengguna (*user*).

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Orang yang berinteraksi dengan sistem.
2.		<i>Life Line</i>	Mengindikasikan keberadaan sebuah objek dalam basis waktu.
3.		<i>Message</i>	Mengindikasikan komunikasi antar objek.


4.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
5.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

Table 3. *Sequence Diagram*