

BAB II

TINJAUAN PUSTAKA

1.1 Profil Tempat Penelitian

1.1.1 Sejarah Program Studi Teknik Informatika UNIKOM

Pada tahun 1994, Dr. Ir. Eddy Suryanto Soegoto mendirikan Lembaga Pendidikan Komputer Indonesia Jerman (LPKIG) untuk program pendidikan 1 tahun, menempati kampus Jl. Dipati Ukur No. 102. Di tahun ketiga, nama Lembaga Pendidikan Komputer Indonesia Jerman (LPKIG) diubah menjadi Indonesian Germany Institute (IGI). Pada tanggal 24 Desember 1998 dibentuk Yayasan Science dan Teknologi yang dilanjutkan dengan Pengajuan ke Kopertis IV Jabar untuk pendirian STIE IGI dan STMIK IGI.

Pada bulan Agustus 1999 keluar SK Mendiknas No. 143/D/O/1999 atas STIMK IGI dengan 5 Program Studi yaitu Teknik Informatika S1, Manajemen Informatika D3 & D1, Teknik Komputer D3 & D1, Teknik Informatika D3 & D1, Komputerisasi Akuntansi D3 & D1.

Mengantisipasi pesatnya perkembangan Teknologi Informasi, IPTEK, Era Globalisasi dan Milenium ke-3 serta untuk memberikan yang terbaik bagi masa depan mahasiswa/i nya, Yayasan Science dan Teknologi kemudian mengajukan usulan ke DIKTI melalui Kopertis Wilayah IV Jabar untuk melakukan merger atas kedua Sekolah Tinggi diatas untuk menjadi Universitas Komputer Indonesia .

Pada hari selasa, tanggal 8 Agustus 2000 keluar SK Mendiknas No. 126/D/O/2000 atas nama Universitas Komputer Indonesia, disingkat UNIKOM. Akreditasi Mendiknas atas UNIKOM disertai menetapkan kembali status Terdaftar kepada Program Studi Teknik Informatika jenjang pendidikan program S1 (Sarjana).

1.1.2 Visi, Misi dan Tujuan Program Studi Teknik Informatika UNIKOM

Berikut merupakan Visi, Misi dan Tujuan dari Program Studi Teknik Informatika UNIKOM.

1.1.2.1 VISI

Menjadi Program Studi Teknik Informatika yang unggul dan terdepan yang dapat menjawab tantangan perkembangan Teknologi Informasi dan Komputer yang berkembang sangat pesat, mampu menyediakan tenaga ahli kompeten, memiliki integritas kepribadian tinggi, dan tanggap terhadap perkembangan ilmu pengetahuan dan teknologi secara global serta berwawasan entrepreneur.

1.1.2.2 MISI

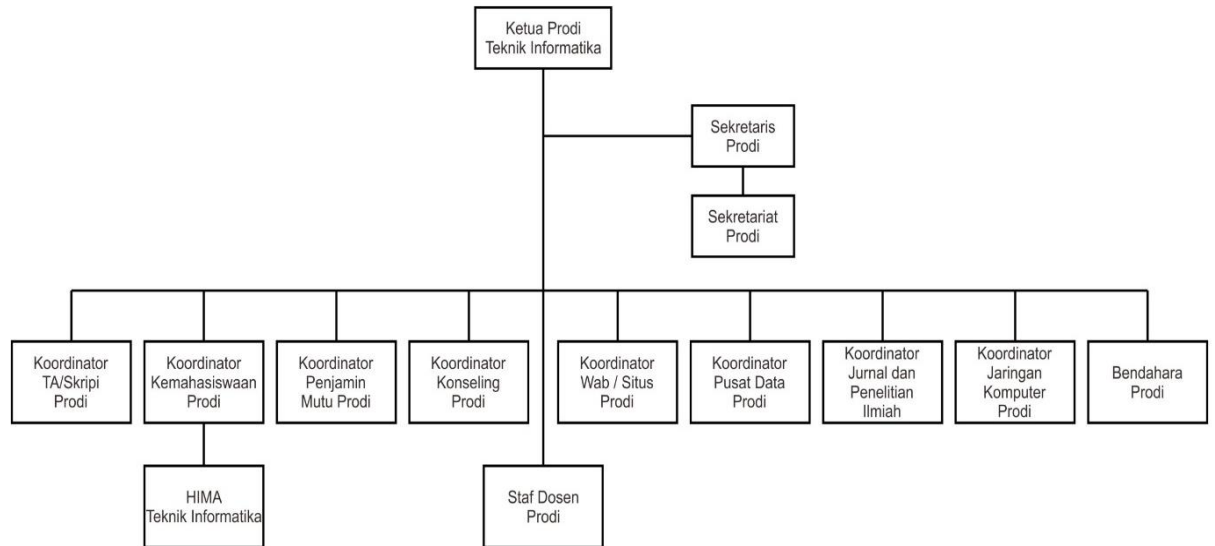
Menyelenggarakan Pendidikan Tinggi yang dapat menghasilkan lulusan berkualitas bidang Teknologi Informasi dan Komunikasi yang berwawasan Teknologi Informasi dan Komunikasi dan entrepreneurship, serta siap bersaing secara global. Melakukan kegiatan Penelitian dan pengembangan bidang Teknologi Informasi dan Komunikasi terkini berdasarkan suatu roadmap yang jelas dan berujung pada publikasi ilmiah dan/atau produk nyata. Melakukan berbagai kegiatan pengabdian kepada masyarakat /industri sebagai upaya untuk berkontribusi menyelesaikan problem nyata di masyarakat/industri.

1.1.2.3 Tujuan

Menghasilkan Sarjana yang memiliki pengetahuan dan kompetensi dalam bidang Teknologi Informasi dan Komputer, menghasilkan karya yang sesuai dengan kebutuhan, serta beriman dan bertakwa kepada Tuhan Yang Maha Kuasa.

1.1.3 Struktur Organisasi

Berikut merupakan struktur organisasi dari Program Studi Teknik Informatika UNIKOM :



Gambar 2. 1 Struktur Organisasi Program Studi Teknik Informatika UNIKOM

1.2 Landasan Teori

Landasan teori merupakan kumpulan teori-teori yang menjadi dasar pembangunan aplikasi ini yang di kutip dari berbagai referensi. Landasan teori dimanfaatkan sebagai fokus penelitian dengan fakta dilapangan. Selain itu landasan teori juga bermanfaat untuk memberikan gambaran umum tentang latar penelitian dan sebagai bahan pembahasan hasil penelitian

1.2.1 Presensi

Presensi adalah suatu pendataan kehadiran, bagian dari pelaporan aktivitas suatu institusi, atau komponen institusi itu sendiri yang berisi data kehadiran yang disusun dan diatur sedemikian rupa sehingga mudah untuk dicari dan dipergunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan (Purwanto, 2009)

1.2.2 Sistem Presensi

Sistem Presensi yang merupakan sebuah sistem yang di gunakan untuk mencatat daftar kehadiran setiap anggota instansi tersebut. Sistem presensi mencatat identitas anggota instansi dan waktu keluar- masuk anggotanya sistem presensi juga mempunyai kemampuan untuk memberikan laporan yang akurat. Oleh karena itu, kebanyakan sekolah memanfaatkan daftar kehadiran mahasiswa untuk menentukan kehadiran mahasiswa tersebut. Selain itu, daftar kehadiran juga dapat memberikan informasi seberapa produktif sekolah itu dengan mahasiswa yang di milikinya. (Rinjap, A, Sompie, & Lantang, 2014)

1.2.3 QR Code

Quick Response Code atau yang sering disingkat dengan *QR Code* merupakan sebuah barcode dua dimensi yang diperkenalkan oleh Perusahaan Jepang Denso Wave pada tahun 1994. Jenis barcode ini awalnya digunakan untuk melacak persediaan di bagian manufaktur kendaraan dan sekarang sudah digunakan dalam berbagai industri perdagangan dan jasa. Pada dasarnya bahwa *QR Code* dikembangkan sebagai suatu kode yang memungkinkan isinya untuk dapat diterjemahkan dengan kecepatan tinggi (Rouillard, 2008). *QR Code* terdiri dari

sebuah untai kotak persegi yang disusun dalam suatu pola persegi yang lebih besar, yang disebut sebagai modul. Gambar berikut ini, menunjukkan gambaran dari sebuah *QR Code*. [3]



Gambar 2. 2 Contoh *QR Code*

1.2.3.1 Karakteristik *QR Code*

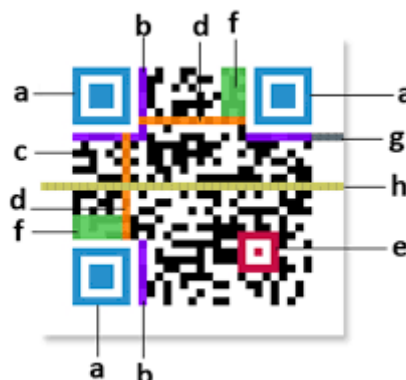
Karakteristik dari *QR Code* yaitu dapat menampung jumlah data yang besar. Secara teori sebanyak 7089 karakter numerik maksimum data dapat tersimpan di dalamnya, kerapatan tinggi (100 kali lebih tinggi dari kode simbol linier) dan pembacaan kode dengan cepat. *QR Code* juga memiliki kelebihan lain baik dalam hal untuk kerja dan fungsi (Ariadi, 2020). Berikut ini merupakan kelebihan unjuk kerja dan fungsi yang dimiliki oleh *QR Code*

1. Pembacaan data dari segala arah (360 derajat)
Pembacaan kode matriks dengan menggunakan sensor kamera CCD (*Charge Coupled Device*) dimana data akan memindai baris perbaris dari citra yang ditangkap dan kemudian disimpan dalam memori dengan menggunakan perangkat lunak tertentu *finding pattern* akan dikenali dan posisi simbol dideteksi.
2. Ketahanan terhadap penyimpanan simbol
Simbol matriks 2 dimensi akan rentan terhadap penyimpanan bentuk ketika ditempatkan pada permukaan yang tidak rata, sehingga sensor pembaca menjadi miring karena sudut antara sensor CCD dan simbol matriks 2 dimensi ini telah berubah. Untuk memperbaiki penyimpanan ini, *QR Code*

memiliki perata pola (*Alignment pattern*) yang menyusun dengan jarak yang teratur dalam suatu daerah.

3. Fungsi pemulihan data
QR Code mempunyai empat tingkatan koreksi *error* (7%, 15%, 25%, dan 30%) di dalam mengendalikan kerusakan yang akibatnya kotor ataupun rusak. *QR Code* memanfaatkan algoritma *Reed-Solomon* yang tahan terhadap kerusakan tinggi.
4. Kemampuan encode karakter kanji dan kana jepang
QR Code berkembang pesat di negara jepang. Hal ini menyebabkan perkembangan *QR Code* untuk dapat menerima input data berupa karakter yang non-alfabetis. Ketika pembuatan *QR Code* dengan inputan berupa huruf jepang, maka data tersebut akan di ubah ke dalam bentuk biner 16 bit (2 byte) untuk karakter tunggal, sedangkan untuk gabungan karakter akan di encode dalam biner 13 bit.
5. Fungsi linking pada simbol
QR Code juga memiliki kemampuan dapat dipecah menjadi beberapa bagian dengan maksimum pembagiannya 16 bagian. Dengan fungsi linking ini, maka *QR Code* dicetak pada daerah yang tidak terlalu luas untuk sebuah *QR Code* tunggal.

1.2.3.2 Anatomi QR Code



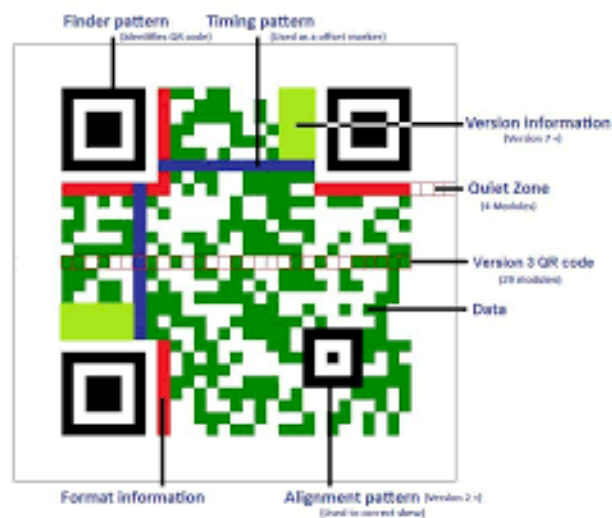
Gambar 2. 3 Anatomi *QR Code*

Beberapa penjelasan anatomi *QR Code* menurut (Ariadi, 2020) adalah sebagai berikut: [3]

- a. *Finder Pattern* berfungsi untuk identifikasi letak *QR Code*.
- b. *Format Information* berfungsi untuk informasi *error correction level* dan mask pattern.
- c. Data berfungsi untuk menyimpan data yang di kodekan.
- d. *Timing Pattern* merupakan pola yang berfungsi untuk identifikasi koordinat pusat *QR Code*, berbentuk modul hitam.
- e. *Alignment Pattern* merupakan pola yang berfungsi memperbaiki penyimpangan *QR Code* terutama distorsi non linier.
- f. *Version Information* adalah versi dari sebuah *QR Code*.
- g. *Quiet Zone* merupakan bagian kosong.
- h. *QR Code Version* adalah versi dari *QR Code* yang digunakan.

1.2.3.3 Struktur QR Code

QR Code memiliki bagian-bagian struktur yang akan penulis jelaskan pada gambar di bawah ini (Ariadi, 2011).



Gambar 2. 4 Struktur QR Code

Berikut ini merupakan penjelasan dari istilah- istilah yang berkenaan dengan gambar *QR Code* di atas:

1. *Finding Pattern* merupakan pola untuk mendeteksi posisi dari *QR Code*.
2. *Timing pattern* merupakan pola yang digunakan untuk identifikasi koordinat pusat dari *QR Code*, dibuat dalam bentuk modul hitam putih bergantian.

3. *Version Information* merupakan Versi dari sebuah *QR Code*, versi terkecil adalah 1 (21 x 21) modul dan versi terbesar adalah 40 (177 x 177) modul.
4. *Quiet Zone* merupakan daerah kosong di bagian terluar *QR Code* yang mempermudah mengenali pengenalan QR oleh sensor CCD.
5. *QR Code version* merupakan versi *QR Code*. Pada contoh gambar, versi yang digunakan adalah versi 3 (29 x 29 modul).
6. Data merupakan daerah tempat data tersimpan atau data dikodekan.
7. *Alignment Pattern* merupakan pola yang digunakan untuk memperbaiki penyimpangan QR Code terutama distorsi non linier.
8. *Format information* merupakan informasi tentang error correction level dan mask pattern.

1.2.4 Smartphone

Smartphone adalah telepon pintar yang memiliki kemampuan seperti komputer. *Smartphone* diklasifikasikan sebagai *high end mobile phone* yang dilengkapi dengan kemampuan *mobile computing*. Dengan kemampuan *mobile computing* tersebut, *smartphone* memiliki kemampuan yang tak bisa dibandingkan dengan ponsel biasa. *Smartphone* yang pertama kali muncul merupakan kombinasi dari fungsi suatu *personal digital assistant* (PDA) dengan telepon genggam ataupun telepon dengan kamera. Seiring dengan perkembangannya, kini *smartphone* juga mempunyai fungsi sebagai *media player portable*, *low end digital compact camera*, *pocket video camera* dan GPS. *Smartphone* modern juga dilengkapi dengan layar touch screen resolusi tinggi, browser yang mampu menampilkan full web seperti pada PC, serta akses data *Wi-Fi* dan internet broadband.[4]

1.2.4.1 Karakteristik Smartphone

Beberapa karakteristik yang umum ada pada smartphone yaitu:

1. Mobile OS
Mobile OS yang sering digunakan pada smartphone adalah: Symbian OS, iPhone OS, Windows Mobile OS, RIM Blackberry, Linux, Palm OS, Android.
2. Open Source.
3. Web Feature.
4. Enhanced Hardware

Fitur hardware eksternal seperti layar sentuh lebar dan sensitif, built-in keyboard, resolusi kamera tinggi, sisi kamera depan untuk video conferences.

Mobile PC Pada umumnya *smartphone* memiliki prosesor yang cukup tinggi, selain itu memiliki penyimpanan memori yang besar dan memiliki RAM tambahan yang cukup besar seperti sebuah PC desktop atau laptop.

1.2.5 Android

Menurut Nasruddin Safaat H (Pemrograman aplikasi *mobile smartphone* dan tablet PC berbasis *android* 2012:1) *android* adalah sebuah sistem operasi pada *handphone* yang bersifat terbuka dan berbasis pada sistem operasi *Linux*. *Android* bisa digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka. *Android* menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, *GoogleInc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. [5]

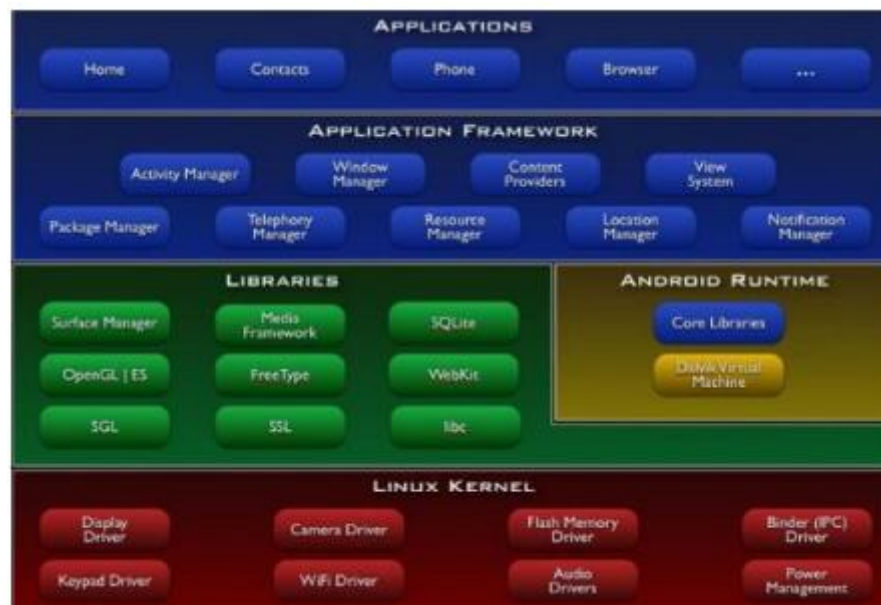
1.2.5.1 Versi Android

Android memiliki sejumlah pembaharuan semenjak rilis aslinya. Pembaharuan ini dilakukan untuk memperbaiki bug dan menambah fitur-fitur yang baru. Berikut merupakan versi-versi yang dimiliki Android sampai saat ini:

Versi Android	Nickname
Android Versi 1.5	Cupcake
Android Versi 1.6	Donut
Android Versi 2.0	Eclair
Android Versi 2.2	Frozen Yogurt – <u>Froyo</u>
Android Versi 2.3	Gingerbread
Android Versi 3.0	Honeycomb
Android Versi 4.0	Ice Cream Sandwich
Android Versi 4.1	Jelly Bean
Android Versi 4.4	KitKat
Android Versi 5.0	Lollipop
Android Versi 6.0	Marshmallow
Android Versi 7.0	Nougat
Android Versi 8.0	Oreo
Android Versi 9.0	Pie
Android 10	
<u>Androd 11</u>	

1.2.5.2 Arsitektur Android

Android merupakan kernel Linux yang menyediakan dan mengatur alur proses aplikasi. Gambar 2.5 merupakan struktur dari sistem operasi Android.



Gambar 2. 5 struktur dari sistem operasi Android

(Sumber : <http://developer.android.com/guide/basics/what-is-android.html>)

Arsitektur Android terdiri atas:

1. Application

Application merupakan bagian yang memuat aplikasi – aplikasi yang dapat digunakan oleh pengguna perangkat Android. Pada bagian Application ini Android memasukkan satu set aplikasi inti yang meliputi email client, program SMS, kalender, peta, *browser* dan kontak. Selain aplikasi inti seperti yang terdapat pada arsitektur *Android*, aplikasi - aplikasi tambahan yang diinstal sendiri oleh pengguna juga akan menempati bagian Application dan memiliki hak akses yang sama terhadap *Application Framework*. (Developer Android, 2012).

2. Application Framework

Application Framework merupakan bagian yang dapat digunakan oleh pengembang aplikasi dalam membuat aplikasi android. Dengan menyediakan pengembangan platform yang terbuka, *Android* menawarkan kemampuan pengembangan untuk membangun aplikasi yang sangat kaya dan inovatif. Pengembang diberikan kebebasan untuk mengambil keuntungan dari perangkat keras, mengakses informasi, *run background services*, *set alarm*, serta menambahkan pemberitahuan ke status bar. Dalam mengembangkan aplikasi, pengembang memiliki akses ke *framework API (Application Programming Interface)* yang sama dengan yang dapat diakses oleh aplikasi –aplikasi inti dari android. (Developer Android, 2012)

3. Libraries

Libraries merupakan kumpulan kode yang dapat digunakan oleh komponen atau program lain. Penulisan kode pada Libraries ditulis menggunakan bahasa pemrograman C/C++.

Beberapa Libraries yang terdapat pada Android yaitu:

1. System C Library, implementasi BSD yang diturunkan dari sistem *library standar C (libc)*.
2. *Media Libraries*, berdasarkan *Open CORE Packet Video*; library dapat mendukung pemutaran dan perekaman audio dan format video, serta 39 file gambar, termasuk MPEG4, H.264, MP3, AAC, AMR, JPG, dan PNG.
3. *Surface Manager*, library yang mengelola penggambaran dalam bentuk komposisi 2D, grafis 3D.
4. LibWebCore

5. *SGL (Scalable Graphics Library)*, library yang menangani pengelolaan grafis 2D.
 6. *3D libraries*, library yang menangani pengelolaan grafis 3D.
 7. *FreeType*, library yang menangani pengelolaan rendering font.
 8. *SQLite*, digunakan untuk pengelolaan database. (Developer Android, 2012).
4. **Android Runtime**
 Android Runtime memiliki dua bagian utama, yaitu:
- a. **Core Libraries**
 - b. **Dalvik Virtual Machine (DVM)**
 Aplikasi Android ditulis dengan menggunakan bahasa pemrograman Java. Dalvik Virtual Machine (DVM) adalah sebuah virtual machine yang digunakan untuk menterjemahkan instruksi –instruksi program Java ke dalam instruksi yang dimengerti oleh sistem operasi. Namun dalam platform Android virtual machine yang digunakan bukan *Java Virtual Machine (JVM)* tetapi *Dalvik Virtual Machine (DVM)*. *Dalvik Virtual Machine* adalah sebuah virtual machine yang di optimasi untuk perangkat yang memiliki memori kecil, sumber daya terbatas, dan kemampuan proses yang kecil. *Dalvik Virtual Machine* mengeksekusi file dalam bentuk Dalvik executable (.Dex). (Andry, 2011).
5. **Linux Kernel**
 Dalam bagian ini android menggunakan modifikasi dari Linux Kernel versi 2.6. Bagian ini bertanggung jawab untuk mengelola dan berkomunikasi dengan perangkat keras dimana android berjalan. Pemilihan Linux Kernel sebagai inti dari android adalah karena dukungan dan kestabilannya terhadap berbagai macam 40 komponen perangkat keras. Pada bagian ini disediakan driver (program pengendali) perangkat keras, pengelolaan memori, pengelolaan proses, pengelolaan jaringan, dan keamanan. (Developer Android, 2012).

1.2.6 Internet

Internet adalah singkatan dari Interconnected Networking yang apabila diartikan dalam Bahasa Indonesia berarti kumpulan komputer yang terhubung di

dalam beberapa rangkaian jaringan. Internet merupakan salah satu hasil dari kecanggihan dan kemajuan ilmu pengetahuan dan teknologi buatan manusia. Internet merupakan sebutan untuk sekumpulan jaringan komputer yang dapat menghubungkan berbagai situs akademik, pemerintahan, komersial, organisasi, hingga perorangan. Lebih lanjut dijelaskan bahwa internet mampu untuk menyediakan akses untuk layanan telekomunikasi dan berbagai sumber daya informasi untuk jutaan pemakaiannya yang tersebar di seluruh dunia. Internet memiliki berbagai macam layanan-layanan internet meliputi komunikasi secara langsung seperti email dan juga chatting, diskusi seperti Usenet News, email dan juga milis serta sumber daya informasi yang terdistribusi (World Wide Web, Gopher), remote login, dan lalu lintas file (Telnet, FTP), dan lain-lainnya.[6]

1.2.7 PHP

PHP (akronim dari PHP: Hypertext Preprocessor) adalah bahasa pemrograman yang berfungsi untuk membuat website dinamis maupun aplikasi web. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan database, file dan folder, sehingga membuat PHP bisa menampilkan konten yang dinamis dari sebuah website. Blog, Toko Online, CMS, Forum, dan Website Social Networking adalah contoh aplikasi web yang bisa dibuat oleh PHP. PHP adalah bahasa scripting, bukan bahasa tag-based seperti HTML. PHP termasuk bahasa yang cross-platform, ini artinya PHP bisa berjalan pada sistem operasi yang berbeda-beda (Windows, Linux, ataupun Mac). Program PHP ditulis dalam file plain text (teks biasa) dan mempunyai akhiran “.php”. Untuk dapat berjalan, PHP membutuhkan web server, yang bertugas untuk memproses file-file php dan mengirimkan hasil pemrosesan untuk ditampilkan di browser client. Oleh karena itu, PHP termasuk server-side scripting (script yang diproses di sisi server). [7]

1.2.8 MySQL

MySQL adalah suatu perangkat lunak database relasi (Relational Database Management System atau RDBMS), seperti halnya ORACLE, Postgresql, MS

SQL, dan sebagainya. MySQL AB menyebut produknya sebagai *database open source* terpopuler di dunia. Berdasarkan riset dinyatakan bahwa di platform Web, dan baik untuk kategori *open source* maupun umum. [8]

1.2.9 Web Service

Web Service merupakan suatu komponen software yang merupakan self-containing, aplikasi modular self-describing yang dapat dipublikasikan, dialokasikan, dan dilaksanakan pada web. (Wahli dkk, 2006) Web Service adalah teknologi yang mengubah kemampuan internet dengan menambahkan kemampuan transactional web, yaitu kemampuan web untuk saling komunikasi dengan pola program to program (P2P). Fokus web selama ini didominasi oleh komunikasi program to user dengan interaksi business to customer (B2C), sedangkan transactional web akan didominasi oleh P2P dengan interaksi business to business.[9]

1.2.10 API

Application Programming Interface (API) adalah sebuah teknologi yang memfasilitasi pertukaran informasi atau data antara dua atau lebih aplikasi perangkat lunak. API adalah antarmuka virtual antara dua fungsi perangkat lunak yang saling bekerja sama, seperti antara sebuah word processor dan sebuah spreadsheet. Sebuah API mendefinisikan bagaimana cara programmer memanfaatkan suatu fitur tertentu dari sebuah komputer. API tersedia untuk sistem windowing, sistem file, sistem database, serta sistem jaringan. [10]

1.2.11 Konsep Perancangan Berorientasi Objek

Pendekatan perancangan berorientasi objek menganalogikan sistem aplikasi seperti kehidupan yang memiliki elemen-elemen objek didalamnya. Dalam membangun sistem yang berorientasi objek, langkah awal yang dilakukan adalah melakukan proses analisis dan perancangan yang berorientasi pada objek-objek yang terlibat didalam sistem. Tujuan dilakukannya proses tersebut adalah untuk mempermudah pengembang dalam mendesain program dalam bentuk objek-objek

dan hubungan antar objek tersebut yang kemudian dimodelkan dalam sistem nyata. Perusahaan software Rational Software, telah membentuk konsorsium dengan berbagai organisasi untuk meresmikan pemakaian Unified Modelling Language (UML) sebagai Bahasa standar dalam Object Oriented Analysis Design (OOAD). [11]

1.2.12 Unified Modelling Language

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML merupakan himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat (tool) untuk mendukung pengembangan sistem tersebut. UML merupakan sistem arsitektur yang bekerja dalam OOAD dengan satu bahasa yang konsisten untuk menentukan, visualisasi, konstruksi dan mendokumentasikan artifact yang terdapat dalam sistem. Artifact adalah sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa software. Artifact dapat berupa model, deskripsi atau software

1.2.12.1 Diagram UML

UML dalam mendokumentasikan rancangan menyediakan berbagai macam diagram untuk memodelkan aplikasi perangkat lunak berorientasi objek . Namun, dalam penelitian ini hanya menggunakan empat macam diagram saja untuk memodelkannya. Empat macam diagram yang digunakan diantaranya yaitu:

1. Use Case Diagram

Use case diagram adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. Use case bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut scenario. Setiap scenario mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan

demikian secara singkat bisa dikatakan use case adalah serangkaian scenario yang digabungkan bersama-sama oleh tujuan umum pengguna.

Use case merupakan alat bantu guna menstimulasi pengguna potensial untuk mengatakan tentang suatu sistem dari sudut pandangnya. Tidak selalu mudah bagi pengguna untuk menyatakan bagaimana mereka bermaksud menggunakan sebuah sistem. Karena sistem pengembangan tradisional sering ceroboh dalam melakukan analisis, akibatnya pengguna seringkali sulit menemukan jawaban tatkala dimintai masukan tentang sesuatu. Diagram use case menunjukkan 3 aspek dari sistem yaitu : actor, use case dan sistem atau sub sistem boundary. Actor mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan use case.

2. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Diagram ini menunjukkan sejumlah contoh objek dan message (pesan) yang diletakan diantara objek-objek ini di dalam use case. Komponen utama sequence diagram terdiri atas objek yang dituliskan dengan kotak segiempat bernama. Message diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progress vertical. Objek diletakan di dekat bagian atas diagram dengan urutan dari kiri ke kanan. Mereka diatur dalam urutan guna menyederhanakan diagram, istilah objek dikenal juga dengan participant, setiap participant terhubung dengan garis titik-titik yang disebut lifeline. Sepanjang lifeline ada kotak yang disebut activation, activation mewakili sebuah eksekusi operasi dari participant. Panjang kotak ini berbanding lurus dengan durasi activation.

3. Activity Diagram

Activity diagram adalah bagian penting dari UML yang menggambarkan aspek dinamis dari sistem. Logika prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam activity diagram. Activity diagram mempunyai peran seperti halnya flowchart, 32 akan tetapi perbedaanya dengan flowchart adalah activity diagram bisa mendukung perilaku paralel sedangkan flowchart tidak bisa.

Tujuan dari activity diagram adalah untuk menangkap tingkah laku dinamis dari sistem dengan cara menunjukkan aliran pesan dari satu aktifitas ke aktifitas lainnya. Secara umum activity diagram digunakan untuk menggambarkan diagram alir yang terdiri dari banyak aktifitas dalam sistem dengan beberapa fungsi tambahan seperti percabangan, aliran paralel, swim lane, dan sebagainya. Sebelum menggambarkan sebuah activity diagram, perlu adanya pemahaman yang jelas tentang elemen yang akan digunakan dalam activity diagram. Elemen utama dalam activity diagram adalah aktifitas itu sendiri. Aktifitas adalah fungsi yang dilakukan oleh sistem setelah aktifitas teridentifikasi, selanjutnya yang perlu diketahui adalah bagaimana semua elemen tersebut berasosiasi dengan constraint dan kondisi lalu perlu penjabaran tata letak dari keseluruhan aliran agar bisa ditransormasikan ke activity diagram.

4. Class Diagram

Class diagram adalah diagram statis. Diagram ini mewakili pandangan statis dari suatu aplikasi. Class diagram tidak hanya digunakan untuk memvisualisasikan, menggambarkan dan mendokumentasikan berbagai aspek sistem tetapi juga membangun kode eksekusi dari aplikasi perangkat lunak. Class diagram menggambarkan atribut, operation dan juga constraint yang terjadi pada sistem. Class diagram banyak digunakan dalam pemodelan sistem OO karena mereka adalah satusatunya diagram UML yang dapat dipetakan langsung dengan bahasa berorientasi objek. Class diagram menunjukkan koleksi class, antarmuka, asosiasi, kolaborasi, dan constraint, dikenal juga sebagai diagram structural.

1.2.12.2 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh System Analyst dalam tahap analisis persyaratan proyek pengembangan sistem. ERD merupakan alat peraga yang memberikan dasar untuk desain database relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan

detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk database.

Dalam Entity Relationship Diagram (ERD) terdapat beberapa komponen, yaitu sebagai berikut:

1. Entitas (Entity)

Entitas adalah suatu objek yang memiliki hubungan dengan objek lain. Dalam ERD digambarkan dengan bentuk persegi panjang.

2. Hubungan (Relationship)

Merupakan keterangan entitas dapat berhubungan dengan entitas lain, hubungan ini disebut dengan entity relationship yang digambarkan dengan garis. Terdapat 4 tipe relasi dalam database yaitu :

- a. One-to-One

Artinya satu data memiliki satu data pasangan di tabel lain. Jadi, jika dua tabel berelasi one-to-one artinya setiap record di entitas pertama hanya akan berhubungan dengan satu record di entitas kedua begitu pula sebaliknya. Contohnya relasi antara tabel pegawai dan alamat pegawai. Satu dokumen pegawai hanya berhubungan dengan satu record alamat pegawai begitu pula sebaliknya.

- b. One-to-Many

Artinya satu data memiliki beberapa data pasangan di tabel lain. Jadi, misalkan terdapat relasi antara tabel satu dan tabel dua yang berarti satu dokumen pada tabel satu boleh berelasi (mempunyai) dengan banyak dokumen pada tabel dua. Namun, satu dokumen pada tabel dua hanya boleh berelasi dengan satu dokumen saja pada tabel satu.

- c. Many-to-One

Artinya beberapa data memiliki satu data pasangan di tabel lain. Jadi, misalkan terdapat relasi antara tabel satu dengan tabel dua, dapat diartikan bahwa satu dokumen pada tabel satu hanya boleh berelasi (mempunyai) dengan satu dokumen pada tabel dua. Tetapi, satu dokumen pada tabel dua boleh berelasi dengan banyak dokumen pada tabel satu

- d. Many-to-Many

Artinya beberapa data memiliki beberapa data pasangan di tabel lain. Jadi, jika tabel satu berelasi dengan tabel dua dengan relasi many-to-many berarti bahwa ada banyak dokumen di entitas satu dan entitas dua yang saling berhubungan satu sama lain.

3. Atribut Atribut adalah elemen dari entitas yang berfungsi sebagai deskripsi karakter entitas dan digambarkan dengan bentuk elips

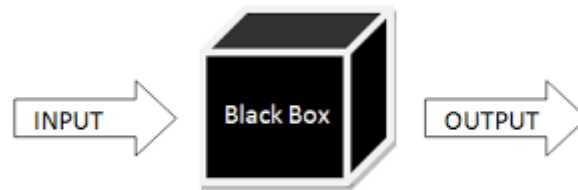
1.2.13 Black Box Testing

Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian black box atau bisa disebut pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan menggunakan perangkat lunak, serta dilihat apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian black box testing harus dibuat dengan kasus benar dan kasus salah. Black box testing juga disebut pengujian tingkah laku, memusat pada kebutuhan fungsional perangkat lunak. Teknik pengujian black box memungkinkan memperoleh serangkaian kondisi masukan yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. [12]

Beberapa jenis kesalahan yang dapat diidentifikasi adalah fungsi tidak benar atau hilang, kesalahan antar muka, kesalahan pada struktur data (pengaksesan basis data), kesalahan performansi, kesalahan inisialisasi dan akhir program. Equivalence Partitioning merupakan metode black box testing yang membagi domain masukan dari program kedalam kelas-kelas sehingga test case dapat diperoleh. Equivalence Partitioning berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. Kasus uji yang didesain untuk equivalence partitioning berdasarkan pada evaluasi dari kelas ekuivalensi untuk kondisi masukan yang menggambarkan kumpulan keadaan yang valid atau tidak. Kondisi masukan dapat berupa spesifikasi nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi Boolean.

Pengujian black box berusaha menemukan kesalahan dalam kategori:

1. Fungsi – fungsi yang tidak benar atau hilang.
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.



Gambar 2. 6 Black Box Texting