

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Induktif

Kajian penelitian terdahulu disebut juga dengan kajian induktif. Kajian ini dimaksudkan untuk mencari kajian dari peneliti terdahulu, sehingga dapat diketahui arah dan *research on* atau arah penelitian dan kajian - kajian yang telah dilakukan oleh peneliti-peneliti terdahulu (Adisajjana, 2016). Berikut adalah beberapa kajian induktif atau penelitian terdahulu dari penelitian ini:

Pertama, penelitian berjudul “*Pembuatan Aplikasi Cek Tagihan Listrik Berbasis Android*”. Hasil Pengujian, didapatkan aplikasi sudah dapat memenuhi fungsi utamanya. Aplikasi sudah dapat menampilkan hasil tagihan bulan yang diinginkan, menyimpan data, dan membuat grafik. Penelitian berikut ini memperkuat referensi ini dengan memberikan info mengenai tahapan pembuatan aplikasi dan pengujiannya (Agan & Santoso, 2013).

Kedua, penelitian berjudul “*Perancangan dan Pembuatan Perangkat Lunak Aplikasi Android untuk Pengolahan Data Transaksi pada Perusahaan Telekomunikasi "X" dengan menggunakan Pentaho*”. Aplikasi berbasis Android untuk pengolahan data transaksi Call dan SMS pada perusahaan telekomunikasi “X” dengan menggunakan datamart ini telah berhasil dibuat. Proses ETL dilakukan menggunakan tool Kettle Pentaho, selain itu aplikasi dapat menampilkan data dari *datamart* dalam bentuk grafik menggunakan pustaka AChartEngine. Jurnal berikut ini dapat memperkuat penelitian ini dengan memberikan referensi mengenai banyak penggunaan tool pembuatan aplikasi dan referensi pengujian aplikasi dengan menggunakan kuesioner (Hastomo & Yuhana, 2013).

Ketiga, penelitian berjudul “*Aplikasi Sistem Informasi Hotel Berbasis Android*”. Sistem Informasi Hotel Berbasis Android mampu menampilkan informasi yang ada pada modul *front office* dan dapat menggantikan proses-proses manual dengan fitur Android. Sistem informasi diimplementasikan ke dalam perangkat smartphone Android dengan bantuan IDE Eclipse dan JSON, kemudian aplikasi yang dibangun berbasiskan web service dimana aplikasi memerlukan koneksi jaringan internet untuk memperoleh data. Jurnal berikut ini memperkuat penelitian ini dengan memberikan gambaran *web service* untuk keperluan data (Kusuma et.al., 2015).

Keempat, penelitian berjudul “*Pengembangan Aplikasi Respons Sms Dan Panggilan Telepon Menggunakan Android Text To Speech Dan Proximity Sensor Bagi Pengemudi Mobil*”. Hasil dari penelitian ini yaitu penggunaan smartphone bagi pengemudi mobil dengan menggunakan *proximity sensor* dan *Android Text to Speech* (TTS) dapat diimplementasikan. Sistem dirancang untuk memudahkan pengguna memilih pesan dan mengirimkan SMS dengan modul keyword. Selanjutnya dilakukan pengujian tingkat usability aplikasi secara keseluruhan menggunakan *System Usability Scale* (SUS) *Questionnaire* dengan kriteria *usability* yang dinilai berupa *efficiency* dan *satisfaction*. Hasil penelitian menunjukkan nilai keseluruhan SUS sebesar 76 yang mengindikasikan bahwa aplikasi respons SMS dan telepon ini secara keseluruhan memiliki tingkat usability yang baik. Jurnal berikut ini memperkuat penelitian ini dengan memberikan referensi mengenai pengujian dengan SUS Questionnaire (Mariani et.al., 2017).

Kelima, penelitian berjudul “*Aplikasi Sistem Informasi Layanan Kesehatan Berbasis Android*”. Hasil dari penelitian ini yaitu telah berhasilnya dibuat aplikasi sistem informasi layanan kesehatan berbasis android yang mudah digunakan dengan mendapat respon sangat setuju sebanyak 100%. Dengan begitu aplikasi ini telah diterima dengan baik pada masyarakat. Aplikasi sistem informasi layanan kesehatan berbasis android dengan menggunakan software Java IDE Eclipse Juno dan hardware smartphone ASUS. Jurnal berikut ini memperkuat penelitian ini dengan memberikan referensi mengenai tahapan pembuatan aplikasi beserta alatnya (Fathoni et.al., 2016).

Keenam, penelitian berjudul “*Aplikasi Pengaduan Masyarakat untuk melaporkan Kejadian Pungutan Liar di Kabupaten Kudus berbasis Android*”. Aplikasi ini adalah suatu sistem aplikasi pengaduan masyarakat berbasis Android yang berfungsi untuk melaporkan terjadinya pungutan liar di kabupaten kudus. Keunggulan aplikasi ini sudah menggunakan teknologi GIS (*Geographic Information Sistem*). Jurnal berikut ini memperkuat penelitian ini dengan memberikan referensi mengenai *use case* dan *activity diagram* (Pratama et.al., 2017).

Ketujuh, penelitian berjudul “*Perancangan Aplikasi E-Canteen Berbasis Android dengan Menggunakan Metode Object Oriented Analysis & Design (OOAD)*”. Hasil penelitian ini didapatkan bahwa dalam perancangan aplikasi e-Canteen, ditemukan beberapa kendala. Pada konsep aplikasi saat pertama kali dibuat, terdapat layanan dimana pelanggan dapat mem-book meja kantin sehingga ketika pelanggan tiba di kantin mereka dapat langsung menempati yang mereka book terlebih dahulu dengan aplikasi. Namun,

dikarenakan keterbatasan kemampuan dalam coding, layanan tersebut diputuskan untuk dihilangkan. Jurnal berikut ini dapat memperkuat penelitian ini dengan memberikan referensi mengenai diagram sistem aplikasi dan mengenai pembatasan fitur karena keterbatasan kemampuan (Neyfa & Tamara, 2016).

Kedelapan, penelitian berjudul “*Perancangan dan Implementasi Sistem Informasi Posyandu terintegrasi berbasis Android*”. Aplikasi ini dibuat bersifat mobile dan berbasis pada sistem operasi android. Nantinya aplikasi ini akan bersifat *client server*, dimana server berfungsi untuk mengelola database. Aplikasi ini dikerjakan dengan menggunakan bahasa pemrograman java dan MySQL sebagai *database*. Perancangan database tersebut menggunakan ERD dan pengujian dengan blackbox. Jurnal berikut ini dapat memperkuat penelitian ini dengan memberikan referensi mengenai ERD, MySQL sebagai media database, beserta cara pengujiannya (Frima et.al., 2016).

Kesembilan, penelitian berjudul “*Aplikasi Penjualan Online Berbasis Android (Studi Kasus Toko Hoax Merch)*” Hasil Penelitian ini didapatkan bahwa aplikasi ini dapat mempermudah dalam penyusunan dan penghitungan pajak reklame, membuat proses penghitungan pajak reklame dengan cepat dan akurat sehingga dapat meminimalisir waktu kerja, membuat laporan penghitungan pajak reklame dan tersimpan dengan baik dan rapih serta tidak mudah untuk rusak bahkan hilang. Jurnal berikut ini dapat memperkuat penelitian ini dengan memberikan referensi mengenai analisa sistem dan perancangan desainnya (Marjito & Tesaria, 2016).

Kesepuluh, penelitian berjudul “*Pengembangan Pocket Mobile Learning Berbasis Android*”. Penelitian ini bertujuan untuk mengembangkan media pembelajaran pocket mobile learning berbasis android. Metode penelitian ini menggunakan model pengembangan ADDIE (*Analysis, Design, Development, Implementation, dan Evaluation*). Penelitian ini telah menghasilkan sebuah luaran berupa media pembelajaran pocket mobile learning berbasis android pada mata kuliah fisika. Media ini layak digunakan berdasarkan validasi dari ahli materi, ahli media dan respon mahasiswa terhadap media pembelajaran yang dibuat. Jurnal berikut ini dapat memperkuat penelitian ini dengan memberikan referensi mengenai model tahapan aplikasi yaitu ADDIE (Dasmo et.al., 2017).

2.2 Kajian Deduktif

2.2.1 Studi Islam

Studi Islam secara etimologis merupakan terjemahan dari Bahasa Arab Dirasah Islamiyah. Sedangkan Studi Islam di barat dikenal dengan istilah *Islamic Studies*. Maka Studi Islam secara harfiah adalah kajian mengenai hal-hal yang berkaitan dengan Islam. Makna ini sangat umum sehingga perlu ada spesifikasi pengertian terminologis tentang studi Islam dalam kajian yang sistematis dan terpadu. Dengan perkataan lain, Studi Islam adalah usaha sadar dan sistematis untuk mengetahui dan memahami serta membahas secara mendalam tentang seluk-beluk atau hal-hal yang berhubungan dengan agama Islam, baik berhubungan dengan ajaran, sejarah maupun praktik - praktik pelaksanaannya secara nyata dalam kehidupan sehari-hari (Muhaimin et.al., 2007).

2.2.2 Dakwah

Ditinjau dari segi bahasa “Da“wah” berarti: panggilan, seruan atau ajakan. Bentuk perkataan tersebut dalam bahasa Arab disebut mashdar. Sedangkan bentuk kata kerja (fi“il) nya adalah berarti memanggil, menyeru, atau mengajak (Da“a, Yad“u, Da“watan). Orang yang berdakwah biasa disebut dengan Da“i dan orang yang menerima dakwah atau orang yang didakwahi disebut Mad“u (Munawir, 1997).

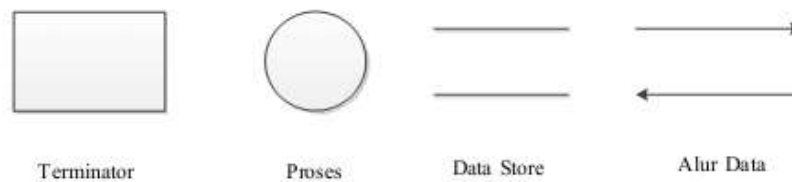
2.2.3 Desain Antarmuka / *User Interface* (UI)

User interface atau antarmuka pengguna adalah sebuah jembatan yang menghubungkan pengguna dengan sebuah produk. Desain antarmuka sebuah produk harus mempertimbangkan nilai efektivitasnya agar pengguna tidak kesulitan dengan antarmuka yang didesain. Desainer juga harus mempertimbangkan karakteristik teknis dan keterbatasan dari perangkat keras dan perangkat lunak yang ada pada produk (Barbara, 2007). Menurut Wilbert O. Galitz, ada beberapa prinsip dalam membangun *User Interface*, diantaranya (Galitz, 2007):

1. Sebuah antarmuka harus benar-benar hanya perpanjangan dari seseorang,
2. UI harus memiliki kegunaan,
3. UI juga harus mudah dipelajari,
4. UI harus melayani baik sebagai konektor maupun pemisah,
5. Sistem harus mudah dan menyenangkan untuk digunakan.

2.2.4 DFD (*Data Flow Diagram*)

DFD atau diagram aliran data adalah sebuah teknis grafis yang menggambarkan aliran infoasi dan transformasi yang diaplikasikan pada saat bergerak dari input menjadi output. DFD dapat digunakan untuk menyajikan sebuah sistem atau perangkat lunak pada setiap tingkat abstraksi (Pressman, 2002). Dapat dilihat pada Gambar 2.1:



Gambar 2.1 Simbol Komponen DFD

2.2.5 ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah diagram yang menyediakan cara mendeskripsikan rancangan basis data pada tingkatan logis. ERD didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan hubungan antar objek (Simarmatar, 2006).

ERD ada dua jenis, yaitu Chen dan Crows foot, dan mempunyai tiga komponen utama yaitu entitas, atribut dan relasi yang dapat dilihat dalam tabel 2.2. Berikut penjelasan komponen-komponennya, yaitu:

1. Entitas (*Entity*)

Entitas merupakan representasi dari kumpulan objek atau benda dalam dunia nyata yang harus memenuhi persyaratan, yaitu:

- Setiap entitas harus bersifat unik
- Setiap entitas mempunyai peran tertentu dalam sistem
- Tiap entitas dapat mendeskripsikan oleh satu atribut atau lebih.

2. Atribut (*Attributes*)

Atribut berperan dalam mengekspresikan karakter setiap entitas.

3. Relasi (*Relationship*)

Relasi menggambarkan hubungan antar entitas. Dalam menggambarkan hubungan entitas ada dua jenis *relationship* yang digunakan, yaitu:

a. *Optionality*

Dalam optionality terbagi menjadi dua jenis, yaitu:

- 1) *Mandatory Relationship*
- 2) *Optional Relationship*

b. *Cardinality*

Untuk ERD Chen, dalam *Cardinality* terbagi menjadi tiga jenis, yaitu:

1) *One to One Relationship*

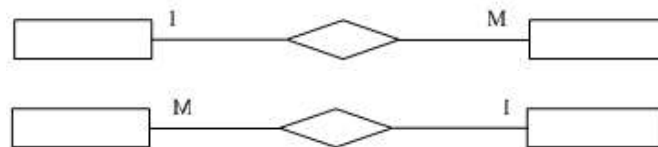
Setiap entitas dalam himpunan entitas pertama berhubungan paling banyak satu entitas pada entitas kedua. Gambar 2.2 merupakan gambaran hubungan satu ke satu.



Gambar 2.2 Simbol Relasi *One to One*

2) *One to Many / Many to one Relationship*

Setiap entitas himpunan entitas pertama berhubungan dengan banyak entitas pada himpunan entitas kedua tetapi tidak berlaku sebaliknya. Gambar 2.3 adalah hubungan satu ke banyak atau sebaliknya.



Gambar 2.3 Simbol Relasi *One To Many*


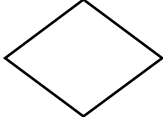
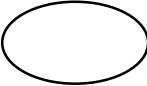

3) *Many to Many Relationship*

Setiap entitas dalam himpunan entitas pertama berhubungan dengan banyak entitas pada himpunan kedua, berlaku sebaliknya. Gambar 2.4 adalah hubungan entitas banyak ke banyak.




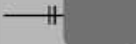
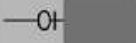



Gambar 2.4 Simbol Relasi *Many To Many*

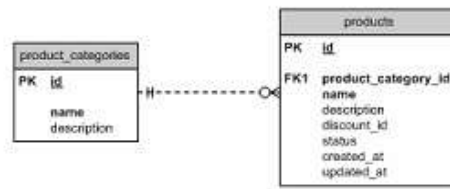
Tabel 2.1 Simbol ERD Chen

Simbol	Nama
	Entitas , adalah objek dalam dunia nyata yang dapat dibedakan dari objek lain.
	Relasi , hubungan antara beberapa entitas.
	Atribut , berfungsi mendeskripsikan karakter entitas.
	Garis , menghubungkan atribut dengan entitas dan kumpulan entitas dengan relasi.

Untuk ERD Crowsfoot, dalam *Cardinality* terbagi menjadi lima jenis, yaitu dijelaskan Tabel 2.2:

Tabel 2.2 Simbol ERD Crowsfoot

CARDINALITY INTERPRETATION	MINIMUM INSTANCES	MAXIMUM INSTANCES	GRAPHIC NOTATION
Exactly one (one and only one)	1	1	 - or - 
Zero or one	0	1	
One or more	1	many (>1)	
Zero, one, or more	0	many (>1)	
More than one	>1	>1	



Gambar 2.5 Contoh Relasi

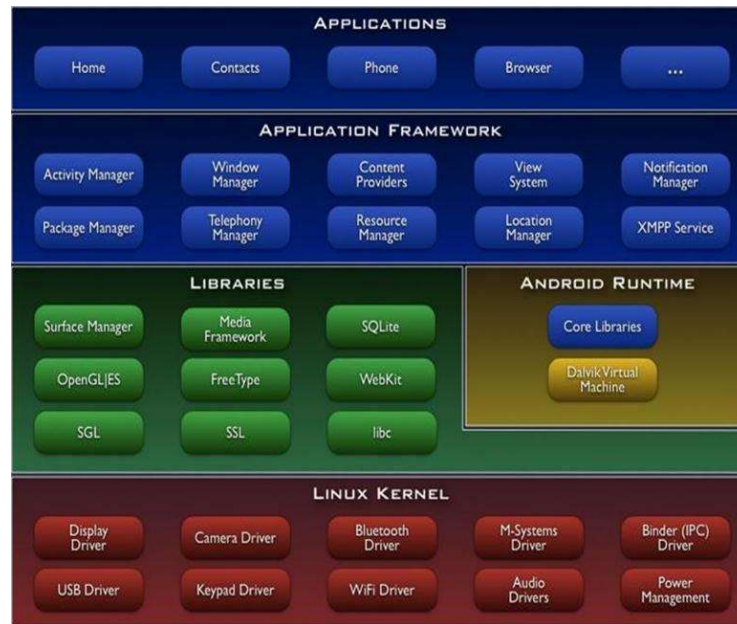
Gambar diatas merupakan contoh relasi antar entitas *product category* dengan *product*. Kardinalitas pada relasi tersebut menunjukkan bahwa setiap produk hanya memiliki satu kategori produk dan kategori produk dapat memiliki nol hingga banyak produk. Atribut *name* pada entitas *products* merupakan atribut yang harus diisi, sedangkan atribut *product_category_id* merupakan atribut *foreign key* yang menunjuk ke entitas *product_categories*.

2.2.6 Sistem Operasi Android

Android adalah sistem operasi berbasis kernel linux dan dirancang terutama untuk perangkat seluler layar sentuh seperti ponsel pintar dan komputer tablet (Varma et.al., 2015). Pada awalnya sistem operasi ini dikembangkan oleh sebuah perusahaan bernama Android Inc. Dari sinilah awal mula nama Android muncul. *Android Inc*, adalah sebuah perusahaan *start-up* kecil yang berlokasi di Palo Alto, California, Amerika Serikat yang didirikan oleh Andy Rubin bersama Rich Miner, Nick Sears, dan Chris White. Pada bulan Juli 2005, perusahaan tersebut diakuisisi oleh Google dan para pendirinya bergabung ke Google. Andy Rubin sendiri kemudian diangkat menjadi Wakil Presiden divisi *Mobile* dari Google.

Tujuan sistem operasi ini adalah untuk menyediakan *platform* yang terbuka, yang memudahkan orang mengakses internet menggunakan telepon seluler. Android juga dirancang untuk memudahkan pengembang aplikasi membuat aplikasi dengan batasan yang minim sehingga kreativitas pengembang menjadi lebih berkembang (Andry, 2011).

Arsitektur Android terdiri dari bagian-bagian seperti Gambar 2.7 berikut:



Gambar 2.6 Arsitektur Android (Developers, 2011)

Berikut ini merupakan penjelasan dari Arsitektur Android tersebut (Developers, 2011):

1. *Linux Kernel*, bagian yang berisi file-file sistem untuk mengatur *processing*, *memory*, *resource*, *driver*, dan sistem operasi android lainnya.
2. *Libraries*, merupakan kumpulan kode yang dapat digunakan oleh komponen atau program lain. Pada bagian ini kodenya ditulis menggunakan bahasa pemrograman C/C++. Fungsi masing-masing komponen di *libraries* ini adalah sebagai berikut:
 - a. *Surface Manager*: mengelola penggambaran dan komposisi komponen dilayar (jendela, tombol, teks *input*, objek tiga dimensi dan sebagainya).
 - b. *Open GLES*: *engine* grafis tiga dimensi.
 - c. *SGL*: *engine* grafis dua dimensi.
 - d. *Media Framework*: *codecs* untuk merekam dan memutar berbagai format video dan audio.
 - e. *Free Type*: untuk *rendering font*.
 - f. *SSL*: untuk keamanan komunikasi jaringan.
 - g. *SQLite*: pengelolaan database untuk penyimpanan data.
 - h. *Webkit*: *web browser engine* Android.
 - i. *Libc* : *library C* standar

3. *Android Runtime*, Pada bagian inilah aplikasi android berjalan. aplikasi android ditulis menggunakan bahasa pemrograman Java dan untuk menjalankannya diperlukan sebuah *virtual machine* yang menerjemahkan instruksi-instruksi program Java tersebut ke instruksi yang dimengerti oleh sistem operasi. Namun berbeda dengan *platform* lainnya, android tidak menggunakan *Java Virtual Machine* (JVM), melainkan menggunakan *Dalvik Virtual Machine* (DVM). *Dalvik Virtual Machine* adalah sebuah *virtual machine* yang dioptimasi untuk perangkat yang memiliki memori kecil, sumber tenaga terbatas, dan kemampuan prosesor yang kecil.
4. *Application Framework*, bagian ini adalah *framework* atau perangkat yang dapat digunakan oleh pengembang aplikasi dalam membuat aplikasi Android. Bagian ini menyertakan program-program yang mengelola fungsi-fungsi dasar seperti informasi lokasi, fungsi telepon, pengelolaan perpindahan antarprogram atau proses, dan sebagainya. Pengembang aplikasi dapat memiliki akses *framework* yang sama dengan yang dapat diakses oleh aplikasi-aplikasi utama dari android (phone dialer, buku telepon, *SMS*, *e-mail*, kalender, dan sebagainya). Di sini terlihat keinginan dari android untuk menjadi platform yang terbuka, sehingga memungkinkan pengembang aplikasi untuk lebih kreatif menciptakan aplikasi yang menarik dan inovatif.
5. *Applications*, bagian ini terdapat aplikasi-aplikasi yang dapat digunakan oleh pengguna perangkat android. Selain aplikasi inti seperti yang ada di diagram arsitektur, aplikasi-aplikasi tambahan yang dipasang sendiri oleh pengguna akan menempati bagian ini dan memiliki hak akses yang sama terhadap *Application Framework*. Apabila diperlukan, pengguna ataupun *vendor* dapat menggantikan aplikasi inti yang ada dengan aplikasi buatan sendiri yang memiliki fungsi yang sama. Hal ini membuat kustomisasi sistem operasi android jauh lebih bebas dibandingkan sistem operasi perangkat *mobile* lainnya.

2.2.7 Android Studio

Android SDK adalah perangkat lunak atau tool API (*application Programming Interface*) yang diperlukan untuk mulai mengembangkan hasil aplikasi pada platform Android yang menggunakan bahasa pemrograman Java (Supardi, 2012). Java API

adalah sebuah layer pemrograman Java yang berisi class-class yang sudah didefinisikan dan antarmuka pemrograman yang akan membantu pada pengembangan aplikasi dalam perancangan aplikasi (Prabawati, 2010).

2.2.8 Databases Server

Server basis data adalah sebuah program komputer yang menyediakan layanan pengelolaan basis data dan melayani komputer atau program aplikasi basis data yang menggunakan *client-server*. Sistem manajemen basis data (SMBD) pada umumnya menyediakan fungsi-fungsi *server* basis data, dan beberapa SMBD (seperti halnya MySQL atau Microsoft SQL *Server*) sangat bergantung kepada model *client-server* untuk mengakses datanya (Kadir, Dasar Pemograman WEB Dinamis Menggunakan PHP, 2008).

2.2.9 XAMPP

Menurut Wicaksono, XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolah data MySQL di komputer lokal. XAMPP berperan sebagai *server web* pada komputer lokal. XAMPP juga dapat disebut sebuah *Cpanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat dimodifikasi *website* tanpa harus online atau terakses dengan internet (Wicaksono, 2008).

2.2.10 MySQL

MySQL merupakan salah satu *database* atau tempat penyimpanan data yang berkembang di lingkungan *open source* dan didistribusikan secara free (gratis) dibawah lisensi GPL. MySQL merupakan RDBMS (*Relational Database Management Sistem*) *server*. RDBMS adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model *relational*. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya (Prasetyo, 2004).

2.2.11 PHP

PHP (*Personal Home Page*) merupakan salah satu bahasa pemograman yang berjalan dalam sebuah *web server* dan berfungsi sebagai pengolah data dari *database*

(MySQL) pada sebuah *server* (Sitorus, 2012). Sistem *database* yang dapat didukung oleh PHP adalah Oracle, MySQL, Ms. Access, Sybase, dan PostgreSQL (Palilu, 2012). PHPMyAdmin adalah salah satu alat berbasis PHP untuk pengelolaan *database* yang berbasis *web*. Sebuah aplikasi *open source* yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan PHPMyAdmin, kita dapat membuat *database*, membuat tabel, memasukkan, menghapus dan mengupdate data dengan GUI (*Graphical User Interface*) dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual (Suprianto, 2008).

2.2.12 Java

Java adalah bahasa pemrograman serbaguna yang dapat digunakan untuk membuat suatu program (Kadir, 2005). Sedangkan menurut Isak Rickyanto, Java merupakan teknologi dimana teknologi tersebut mencakup java sebagai bahasa pemrograman yang memiliki sintaks dan aturan pemrograman tersendiri, juga mencakup java sebagai *platform* dimana teknologi ini memiliki *virtual machine* dan *library* yang diperlukan untuk menulis dan menjalankan program yang ditulis dengan bahasa pemrograman java (Rickyanto, 2002).

2.2.13 XML

XML adalah sebuah standar yang digunakan untuk menstrukturkan informasi dalam sebuah dokumen menjadi sejumlah bagian dan untuk mengidentifikasi bagian tersebut. Dokumen bukan hanya berupa dokumen teks, tetapi termasuk juga data gambar, persamaan matematika, rumus bangun kimia, dan berbagai jenis informasi yang dapat distrukturkan (Anderson, 2000) (Marchal, 2000).

2.2.14 Model ADDIE

Model ADDIE adalah salah satu model desain sistem pembelajaran yang memperlihatkan tahapan-tahapan dasar sistem yang sederhana dan mudah dipelajari. ADDIE adalah akronim dari *Analyze, Design, Development, Implementation, dan Evaluate* (Branch, 2009). Model ADDIE terdiri dari lima tahap, yaitu:

1. Analisis (*Analyze*)

Analisis merupakan langkah pertama dari model desain sistem pembelajaran ADDIE. Langkah analisis melalui dua tahap yaitu:

- a. Analisis Pengguna dan Fungsionalitas
Analisis Pengguna Fungsionalitas dilakukan untuk mengetahui dan mengklarifikasi pengguna atau *user* serta perannya pada aplikasi tersebut.
 - b. Analisis Kebutuhan
Analisis kebutuhan merupakan langkah yang diperlukan untuk menentukan kebutuhan yang diperlukan sebagai input, proses, dan output pada aplikasi.
2. Desain (*Design*)
Tahap desain merupakan tahapan perencanaan aplikasi. Peneliti memodelkan atau mendesain bagaimana suatu aplikasi berjalan dan cara kerjanya, ditunjukkan dengan menguraikan relasi antara hubungan data didalam aplikasi, serta antara pengguna dengan aplikasi
 3. Pengembangan (*Development*)
Langkah pengembangan merupakan langkah untuk merancang desain yang akan dibuat. Rancangan tersebut digunakan sebagai gambaran bentuk sistem dari perancangan berdasarkan analisis yang di ambil, dan hasil rancangan tersebut akan ditampilkan dibagian implementasi.
 4. Implementasi (*Implementation*)
Tahap implementasi merupakan hasil nyata dari pengembangan yang telah dilakukan. Artinya, pada tahap ini semua yang telah dikembangkan diaplikasikan atau diatur sedemikian rupa sesuai dengan peran atau fungsinya agar bisa diimplementasikan dan dievaluasi.
 5. Evaluasi (*Evaluate*)
Tahap evaluasi merupakan tahapan pengujian aplikasi, yang mana pengujian tersebut membutuhkan suatu metode tertentu, seperti *blackbox* dan *whitebox testing*, ataupun yang dipakai pada penelitian kali ini yaitu *SUS Questionnaire* yang berfungsi sebagai penguji apakah aplikasi tersebut layak digunakan atau tidak.

2.2.15 Usability Testing

Usability merupakan bagian dari bidang ilmu multi disiplin *Human Computer Interaction* (HCI). *Human Computer Interaction* merupakan bidang ilmu yang berkembang sejak tahun 1970 yang mempelajari bagaimana mendesain tampilan layar komputer dalam suatu aplikasi sistem informasi agar nyaman dipergunakan

oleh pengguna (Nugroho, 2009). *Usability Testing* adalah salah satu cara untuk melakukan evaluasi sebuah produk (Rubin & Chisnell, 2008). Salah satu alat usability testing adalah *SUS Questionnaire*. Penentuan jumlah partisipan/responden dalam pengujian *usability* ini berdasarkan penelitian Faulkner yang mengusulkan aturan untuk sampel yang optimal dalam melakukan pengujian *usability* yaitu 10 responden, karena ukuran sampel dari 10 peserta kemungkinan besar akan mengungkapkan minimal 80% dari masalah *usability*. Aplikasi berada pada kategori “*Acceptable*” berdasarkan *Acceptability Range* yang diberikan pada skor SUS dengan rentang nilai 70-100 (Faulkner, 2003).