BAB II

LANDASAN TEORI

2.1. Konsep Dasar Program

A. Definisi Program

Menurut sutarman dalam bukunya pengantar teknologi informasi menyatakan bahwa " Program adalah barisan perintah / intruksi yan disusun sehingga dapat dipahami oleh komputer dan kemudian dijalankan dalam bentuk barisan perhitungan numerik, yang mana barisan perintah tersebut berhingga, berakhir, dan menghasilkan output"(Maarif et al., 2017). Sekumpulan perintah yang diwujudkan dalam bentuk bahasa, kode skema, ataupun bentuk lain, yang bilamana digabungkan dengan media yang dapat dibaca dengan komputer akan mampu membuat komputer bekerja untuk melakukan fungsi-fungsi khusus termasuk persiapan dalam merancang intruksi-instruksi tersebut.

Adapun langkah-langkah yang ditempuh sebelum mengimplementasikan suatu permasalahan dalam program (Maarif et al., 2017), yaitu :

a. Mendefinisikan masalah
b. Mencari solusi masalah
c. Memilih teknik pemecahan masalah algaritma
d. Menulis program
e. Melakukan testing debbuging
f. Melakukan dokumentasi
g. Melakukan pemeliharan

B. Sistem

Menurut Lukman menyatakan bahwa “Sistem adalah suatu susunan yang teratur dari kegiatan-kegiatan yang saling berhubungan dan susunan prosedur-prosedur yang saling berhubungan, sinergi dari semua unsur-unsur dan elemen-elemen yang ada didalamnya, yang menunjang pelaksanaan dan mempermudah kegiatan-kegiatan utama tercapai dari suatu organisasi ataupun kesatuan kerja”(Tabrani & Mutaqin, 2019).

Sedangkan menurut Tohari “Sistem merupakan kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terhubung, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Selain itu sistem juga dapat didefinisikan sebagai sekumpulan objek-objek yang saling berelasi dan saling berinteraksi, serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan yang telah ditetapkan”(Rivaldi, 2018).

Berdasarkan kutipan diatas dapat disimpulkan bahwa sistem adalah perangkap unsur yang terdiri dari beberapa komponen yang saling berhubungan atau berkaitan satu sama lain untuk membentuk satu kesatuan yang berfungsi secara bersama-sama untuk mencapai tujuan dalam suatu lingkungan komplek.

1. Karakteristik Sistem

Sebuah sistem dapat dikatakan sistem apabila memiliki karakteristik atau sifat-sifat tertentu. Menurut Sutabri Karakteristik sistem informasi terdiri dari(Kurniati, 2014):

a. Komponen sistem (*Components*)

Suatu sistem terdiri dari komponen-komponen yang saling berhubungan, artinya saling bekerja sama membentuk satu kesatuan.

b. Batasan sistem (*Boundary*)

Ruang lingkup sistem adalah daerah yang membatasi antara sistem dengan lingkungan luarnya atau sistem dengan sistem yang lainnya.

c. Lingkungan luar sistem (*Environment*)

Bentuk apapun yang ada batasan sistem atau luar ruang lingkup yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem.

d. Penghubung sistem (*interface*)

Media yang mempertemukan sistem dengan subsistem lain disebut penghubung sistem atau *interface*.

e. Masukan sistem (*input*)

Energi yang dimasukan kedalam sistem disebut masukan sistem, yang bisa berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran sistem (*output*)

Hasil energi yang diselesaikan dan diklarifikasi menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi.

g. Pengolahan sistem (*process*)

Suatu sistem dapat memiliki suatu proses yang akan merubah masukan menjadi keluaran.

h. Sasaran sistem (*objective*)

Suatu sistem mempunyai tujuan dan sasaran yang pasti dan bersifat *deterministic*.

2. Klasifikasi Sistem

Menurut Sutabri "sistem dapat diklarifikasikan dalam beberapa sudut pandang diantaranya sebagai berikut" (Arisma Susanto, 2015):

1) Sistem abstrak (*abstarct system*) dan sistem fisik (*physical system).*

Sistem abstrak merupakan sistem yang berupa spekulasi yang tidak terlihat secara fisik, misalnya sistem teknologi yaitu sistem berupa pemikiran-

pemikiran kaitan antara manusia dengan tuhan. Sistem fisik merupakan sistem yang ada secara fisik.

2) Sistem alamiyah (*natural system)* dan sistem buatan manusia (*human made system).*

Sistem alami adalah sistem yang keberadaannya terjadi secara alami atau natural tanpa campuran tangan manusia. Sedangkan sistem buatan manusia merupakan hasil kerja manusia. Contoh sistem alamiah ialah sistem tata surya yang terdiri atas sekumpulan planet, gugus bintang dan lainnya. Contoh sistem abstrak bisa berupa sistem komponen yang ada sebagai hasil karya teknologi yang dikembangkan manusia.

3) Sistem tertentu (*deterministic system)* dan sistem tak tentu (*probabilistic system).*

Sistem tertentu merupakan sistem yang tingkah lakunya dapat ditentukan atau diperkirakan sebelumnya. Sedangkan sistem tak tentu adalah sistem yang tingkah lakunya tidak dapat ditentukan sebelumnya. Sistem aplikasi komputer adalah contoh sistem yang tingkah lakunya dapat ditentukan sebelumnya. Program aplikasi yang dibuatkan oleh manusia dengan menggunakan prosedur yang jelas, terstruktur dan baku.

4) Sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

Sistem tertutup adalah sistem yang tingkah lakunya tidak dipengaruhi oleh lingkungan luarnya. Sedangkan sebaliknya, sistem terbuka memiliki perilaku yang dipengaruhi oleh lingkungannya. Sistem aplikasi komputer tidak dipengaruhi oleh kondisi yang terjadi diluar sistem.

C. Bahasa pemprograman

Menurut Suprapto dalam (Maarif et al., 2017) “Bahasa pemrograman adalah notasi yang digunakan untuk menulis program (komputer). Bahasa ini dibagi menjadi tiga tingkatan yaitu bahasa mesin, bahasa tingkat rendah dan bahasatingkat tinggi”. Lain hal menurut Ritani dalam (Fadlilah & Perdana, 2019) “Bahasa Pemrograman merupakan notasi yang dipergunakan untuk mendeskripsikan proses komputasi dalam format yang dapat dibaca oleh komputer dan juga manusia”.

Secara garis besar terdapat dua kategori bahasa pemrograman yakni: bahasa pemrograman aras rendah (*low level*) dan bahasa pemrograman aras tinggi (*high level*). Bahasa pemrograman aras rendah cenderung mendekati level komputer, dapat dikatakan bahwa bahasanya ditulis mendekati atau sama dengan bahasa mesin komputer, hal ini sangat sulit ditulis karena bahasanya jauh dari bahasa manusia yang digunakan sehari-hari.

Bahasa pemrograman yang lebih mudah dipelajari adalah bahasa pemrograman aras tinggi. Dikatakan aras tinggi karena bahasanya mendekati level bahasa manusia sehingga lebih mudah dipahami. Tabel berikut adalah gambaran antara bahasa aras tinggi dan aras rendah :

Tabel II.1.

Bahasa Pemprograman Aras Tinggi

Nama	Penjelasan
BASIC	*Beginners All-purpose Symbolic Instruction Code*, adalah bahasa pemrograman yang sering digunakan untuk merancang program sederhana pada pemrograman pemula.
FORTRAN	*Formula Translator*, Bahasa pemrograman yang dibuat

	untuk menyelesaikan algoritma matematika yang kompleks.
COBOL	*Common Business-Oriented Language*. Bahasa pemrograman yang dibuat pada aplikasi bisnis.
PASCAL	Pemrograman terstruktur, bersifat umum, dan biasanya bahasa pemrogram ini banyak diajarkan.
C	Pemrogram terstruktur, bersifat umum. Bahasa ini dikembangkan oleh laboratorium Bell. Bahasa C ini bisa digunakan sebagai bahasa aras tinggi dan aras rendah.
C++	Dasar pengembangan C. C++ dapat digunakan sebagai bahasa berorientasi objek, yang tidak ditemukan pada bahasa C. Bahasa ini juga dikembangkan oleh laboratorium Bell.
C#	C# atau "*C sharp*". Bahasa ini ditemukan oleh microsoft untuk mengembangkan aplikasi pada aplikasi microsoft .NET.
JAVA	Bahasa ini merupakan bahasa berorientasi objek yang dikembangkan oleh *Sum Microsistem*. Dengan *Java* memungkinkan untuk pengembangan program yang berjalan pada jaringan internet atau pada *web browser*.
VISUAL BASIC	Bahasa pemrograman microsoft yang mana bahasa ini bertujuan untuk pengembangan perangkat lunak yang dapat mempermudah pemrogram dalam membuat

	aplikasi berbasis windows.

Sumber : (Maarif et al., 2017)

D. Aplikasi

"Aplikasi adalah perangkat lunak proses data yang berpacu pada sebuah komputasi. Aplikasi berasal dari bahasa Inggris *application* yang artinya penerapan, lamaran serta penggunaan. Sedangkan secara istilah adalah suatu program yang siap digunakan untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi dan penggunaan aplikasi lain yang bisa digunakan oleh suatu sasaran yang akan dituju" (Hendra, 2016).

Berikut definisi Aplikasi menurut para ahli :

1. Menurut Maarif, Nur, & Rahayu dalam (Fadlilah & Perdana, 2019) "aplikasi merupakan sebuah program komputer yang dibuat untuk melaksanakan dan mengerjakan tugas khusus dari *user* (pengguna)".
2. Menurut Hendrayudi dalam (Maarif et al., 2017) menerangkan bahwa "aplikasi merupakan sekumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu (khusus)".
3. Menurut Dhanta dalam (Hendra, 2016) menjelaskan "aplikasi (*application*) adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Word*, dan *Microsoft Excel* ".

E. Android

"Android merupakan *system* operasi untuk telepon selular berbasis linux. Android menawarkan *platform* terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri yang digunakan oleh berbagai macam piranti bergerak. Mulanya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, ditentukan *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia"(Afrilla et al., 2017).

1. Android Studio

> "Android studio merupakan IDE (Integrated Development Environment) resmi bagi pengembangan aplikasi Android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada tanggal 16 mei 2013 pada event Google I/O Conference di tahun 2013. Mulai saat itu, Android Studio mengantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android"(Juansyah, 2015).

Sumber : (Juansyah, 2015)

Gambar II.1. Android Studio

Android studio itu sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools) (Juansyah, 2015). Android studio memiliki fitur:

1. Projek berbasis pada Gradle Build.
2. *Refactory* dan sistematisasi bug yang cepat.
3. Tools baru yang diberi nama "Lint" diklaim mampu memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
4. Mendukung *Proguard And App-signing* untuk keamanan.
5. Memiliki GUI aplikasi android lebih mudah.
6. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan.

2. Versi Android

Banyaknya handphone pintar (*smartphone*) dan Tablet menggunakan sistem operasi yang berbeda. Semakin tinggi versi yang dibawanya, fiturnya semangkin

canggih dan banyak. Berikut uraian versi android, seperti pada Tabel II.2 memaparkan versi pengembang sistem operasi android.

Tabel II.2.

Versi Pengembang Sistem Android

Versi	Nama	Tanggal Rilis
1.5	Cupcake	30 April 2009
1.6	Donut	15 September 2009
2.0-2.1	Eclair	26 Oktober 2009
2.2	Froyo	20 Mei 2010
2.3-2.3.2	Gingerbread	06 Desember 2010
3.1	Honeycomb	10 Mei 2011
4.0.3-4.0.4	Ice Cream Sandwich	16 Desember 2011
4.1.x	Jelly Bean	09 Juli 2012
4.2.x	Jelly Bean	13 November 2012
4.3.x	Jelly Bean	24 Juli 2013
4.4.x	KitKat	31 Oktober 2013
5.0	Lollipop	15 Oktober 2014
6.0	Marshmallow	05 Oktober 2015
7.0	Nougat	22 Agustus 2016
8.0	Oreo	21 Agustus 2017

Sumber : Irsyad (Irawan, 2018).

3. Fitur Android

Android adalah open source dan tersedia secara bebas untuk produsen maupun kustomisasi, tidak ada konfigurasi perangkat keras atau perangkat lunak yang tetap.

Adapun berikut merupakan fitur-fitur dasar yang mendukung OS Android (DIMARZIO, 2017) adalah :

a. *Framework* Aplikasi yang didukung peggantian komponen dan *reusable*.

b. Mesin *Virtual Delvik* dioptimalkan bagi perangkat mobile.

c. *Integrated browser* berdasarkan *engine open source WebKit.*

d. Grafis yang dioptimalkan serta didukung oleh *libraries* grafis 2D,grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional akselerasi *hardware*).

e. SQLite untuk menyimpan data.

f. *Media support* yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF), GSM *Telephony* (tergantung *hardware*).

g. Bluetooth, EDGE, 3G dan WIFI (tergantung *hardware*).

h. Kamera, GPS, kompas dan *accelerometer* (tergantung *hardware*).

4. Arsitektur Android

Secara garis besar android terbagi menjadi lima bagian didalam empat lapisan utama (DIMARZIO, 2017). Yaitu :

a. *Applications*

Pada lapisan teratas terdapat aplikasi yang paling nampak oleh pengguna ketika menjalankan program (seperti telepon, kontak, browser, dan sebagainya).

b. *Application Framework*

Kerangka Aplikasi (*Application Framework*) memaparkan berbagai kemampuan OS Android untuk pengembang aplikasi sehingga mereka dapat memanfaatkannya dalam aplikasi mereka.

c. *Libraries*

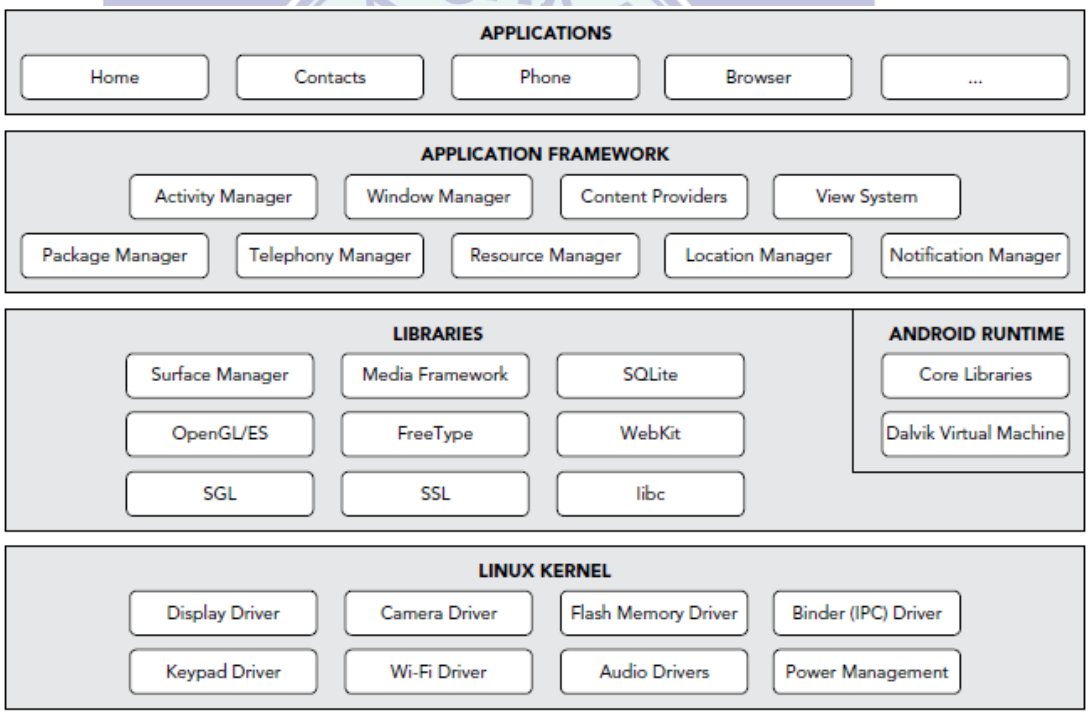
Berisi kode yang menyediakan fitur utama OS Android. Kemampuannya dapat diakses oleh pengembang aplikasi melalui *Application Framework*.

d. Android *Runtime*

Terletak pada lapisan yang sama dengan *Libraries* dan menyediakan satu set komponen inti yang memungkinkan pengembang untuk membuat aplikasi Android menggunakan bahasa pemprograman Java.

e. Linux Kernel

Linux Kernel adalah kernel yang menjadi dasar Android. Lapisan ini berisi semua pengandar perangkat *low-level* untuk berbagi komponen perangkat keras dari perangkat Android. Android dibangun di atas kernel Linux 2.6 sehingga dapat dihandalkan kecepatannya.



Sumber : (DIMARZIO, 2017)

Gambar II.2. *Android Architecture Diagram*

F. Java

"Java adalah bahasa pemrograman berorientasi objek dan bebas *platform*, dikembangkan oleh SUN Micro Sistem dengan sejumlah keunggulan yang memungkinkan Java digunakan sebagai bahasa pengembangan *enterprise*. JDK adalah perangkat lunak yang digunakan untuk melakukan proses kompilasi dari kode *java* ke *bytecode* yang dapat dimengerti dan dapat dijalankan oleh JRE

(Java Runtime Envirotment). JDK wajib terinstall pada komputer yang akan melakukan proses pembuatan aplikasi berbasis java, namun tidak wajib terinstall di komputer yang akan menjalankan aplikasi yang dibangun dengan java" (F, Hardhienata, & Herdianto, 2011).

1. *Java Development Kit* (JDK)

   Andi Juansyah mengungkapkan Bahwa :

   > "Java Development Kit (JDK) merupakan sekumpulan perangkat lunak yang bisa kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan JRE merupakan sebuah implementasi dari Java Virtual Machine yang benar-benar digunakan untuk menjalankan program java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber compiler java, bundling, debuggers, development libraries dan lain sebagainya"(Tasril, Wibowo, & Pancabudi, 2019).

2. *Android SDK (Software Development Kit)*

   > "Android SDK merupakan tools API *(Application Programming Interface)* yang dibutuhkan untuk memulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemograman java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi system operasi, *middleware* dan aplikasi kunci yang dirilis oleh *Goggle*. Saat ini disediakan *Android SDK (Software Development Kit)* sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemograman java. Sebagai platform aplikasi-netral, android memberikan kesempatan untuk membuat aplikasi yang dibutuhkan"(Sehang et al., 2018).

3. ADT (*Android Developer Tools*)

   Safaat dalam (Tasril et al., 2019) memaparkan bahwa "*Android Development Tools* (ADT) adalah plug-in yang didesain untuk IDE Eclipse yang memberi kita kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Eclipse".

G. Basis Data (*Database*)

Basis Data (*Database*) merupakan kumpulan data yang saling berkaitan serta disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi berbagai macam kebutuhan.

Fathansyah dalam (Rivaldi, 2018) menyatakan bahwa “Basis data adalah himpunan kelompok data (arsip) yang saling berkaitan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat”.

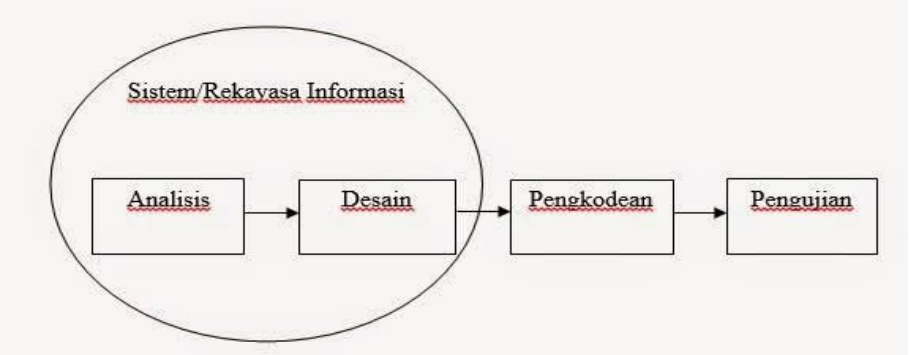
Basis Data adalah sebuah sistem yang dibangun dari beberapa komponen (Hanif, 2016), antara lain yaitu :

1. Perangkat keras (hardware) dalam sistem komputer terdapat sistem pengolahan basis data digital perangkat utama sebagai pengolah data dalam komputer.
2. Perangkat Lunak Aplikasi (software) lain yang mendukung dan bersifat opsional. Perangkat lunak digunakan untuk mendukung proses pengelolaan basis data. Misal: bahasa pemrograman C, basic pascal.
3. Sistem Operasi (*Operating System*). Sistem operasi adalah perangkat lunak yang digunakan untuk mengelola aplikasi basis data dan penggunaan sumber daya komputer.
4. Basis data lain yang mempunyai keterkaitan dan hubungan dengan basis data itu sendiri. Berisi atau memiliki objek-objek basis data seperti *file*, *table*, indeks. Mempunyai definisi struktur baik untuk basis data maupun objek-objek secara detail.
5. Sistem pengelola basis data *Database Management System* (DBMS), adalah program aplikasi untuk pengelolaan, seperti *Microsoft Acces*, *Oracle*, dan lainnya.
6. Pengguna (*User*), yaitu pemakai yang terlibat dalam pengelolaan basis dan penggunaan basis data.

H. Model Pengembangan Perangkat Lunak

Rosa dan Shalahuddin dalam (Rindina, Ferdiansyah, & Ishaq, 2019) menyimpulkan bahwa: "Model SDLC air terjun (waterfall) sering juga disebut model sekuensial linier (sequential linear) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurutan dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (support)".

Berikut adalah gambar model air terjun:



Sumber : Rosa dan Shalahuddin (Rindina et al., 2019)

Gambar II.3 Ilustrasi *waterfall*

Berikut tahapan-tahapan dalam metode Waterfall:

a. Analisis

Analisis sistem bertujuan untuk mengevaluasi dan mengidentifikasikan berbagai permasalahan, hambatan juga kebutuhan yang diharapkan dapat diusulkan perbaikan-perbaikan.

b. Desain

Desain perangkat lunak merupakan proses multi langkah yang fokus pada desain perancangan program perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan sampai ke representasi desain agar bisa diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengkodean

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini ialah program komputer sesuai dengan desain yang sudah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan bertujuan meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai yang diharapkan.

2.2.Teori Pendukung

A. *Entity Relationship Diagram*

Iswandi dalam (Wisky, 2017) menyatakan bahwa "*Entity Relationship* Diagram (ERD) memiliki dua komponen utama yaitu Entitas *(Entity)* dan Relasi *(Relation).* Kedua komponen ini, masing-masing dilengkapi dengan sejumlah atribut yang mempresentasikan seluruh fakta yang ada di dunia nyata".

Sedangkan komponen ERD Menurut Fatta dalam (Fauzi et al., 2019) mengemukakan bahwa, "Didalam ERD terdapat komponen-komponen khusus untuk menggambarkan elemen-elemen ERD". Antara lain yaitu :

a. Entitas

Entitas merupakan objek yang harus menampilkan beberapa kali *event* untuk menjadi sebuah entitas.

b. Atribut

Atribut merupakan informasi dari sebuah entitas yang berupa kata benda dan hanya digunakan oleh organisasi yang dimasukan dalam model.

c. *Identifier*

Identifier merupakan satu atau lebih atribut yang bisa menjadi *identifier* entitas secara unik mengidentifikasi setiap anggota dari entitas.

d. Kardinalitas

Hubungan antara entitas dalam suatu *instance* adalah kardinalitas artinya entitas dapat berelasi dengan *instance* lain di *instance* yang berbeda

e. Modalitas

Dapat dikatakan modalitas dari *instance* apabila sebuah entitas anak bisa ada tanpa sebuah relasi dengan *instance* dari entitas induk atau tidak.

B. *Logical Record Structure (LRS)*

Menurut Pradani dkk dalam (Fauzi et al., 2019) memberi batasan bahwa, "LRS adalah representasi dari struktur record-record pada tabel-tabel yang terbentuk dari hasil antar himpunan entitas".

C. *Unified Modelling Laguage* (UML)

Menurut Simaremare Apol dan Radityo dalam (Fauzi et al., 2019) mengemukakan bahwa," Seorang pengembang memungkinkan dapat membuat sebuah blueprint yang menggambarkan visi dari sebuah sistem dalam bentuk yang standar, mudah dipahami, dan memberikan mekanisme yang mudah dikomunikasikan kepada pihak lain menggunakan pemodelan bahasa visual UML (*Unified Modeling Language*)".

Unified Modeling Language (UML) berfungsi untuk menggambarkan struktur dan desain sistem yang akan di bangun. Diagram yang digunakan dalam perancangan sistem ini antara lain : *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram.*

1. *Use Case Diagram*

Sukamto & Shalahuddin dalam (Nur Ali Farabi , Andi Rosano, 2018) mengemukakan bahwa “Diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang dibuat”.

2. *Activity Diagram*

Sukamto & Shalahuddin dalam (Nur Ali Farabi , Andi Rosano, 2018) mengemukakan “Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

3. *Sequence Diagram*

Sukamto & Shalahuddin dalam (Nur Ali Farabi , Andi Rosano, 2018) mejelaskan bahwa “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”.

BINA SARANA INFORMATIKA UNIVERSITAS