

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

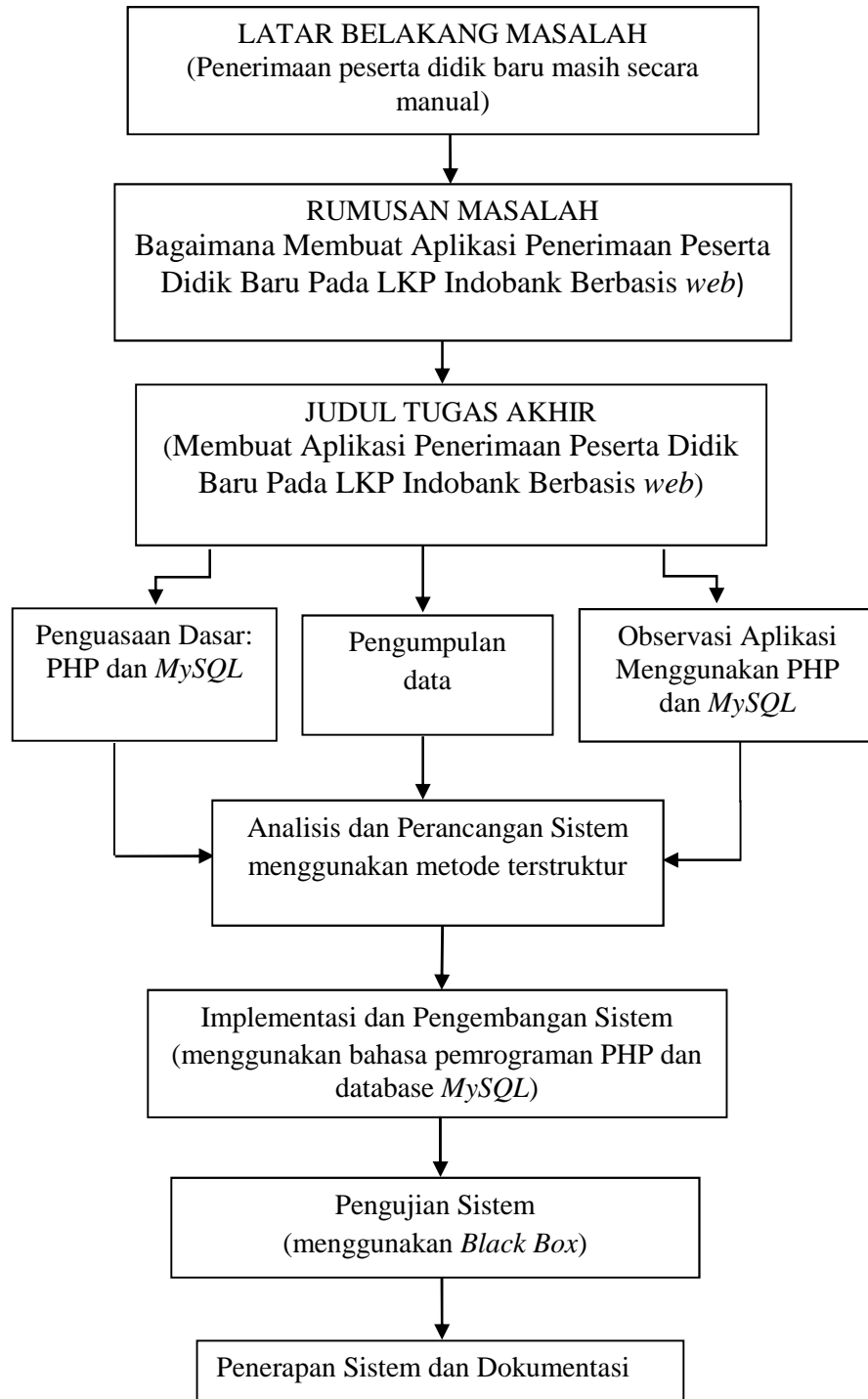
“Sistem Informasi Penerimaan Siswa Baru pada SMPN 2 Pangaroan (2016)” Laporan Skripsi oleh Akhmad Mubarak Program Studi Sistem Informasi STIKOM Surabaya. Sistem Informasi yang digunakan masih menggunakan program *offline*, sehingga peserta harus datang ke sekolah untuk mendaftar dan melihat langsung untuk hasil dari seleksi penerimaan siswa baru sehingga menyebabkan borosnya biaya dan waktu. Untuk program Sistem Informasi yang dilakukan secara *offline* informasi yang dipublikasikan tidak cepat tersampaikan untuk para calon siswa baru atau pun orang tua dari calon siswa yang mendaftar. Kami mengusulkan agar program dirubah menjadi *online*, agar informasi dapat tersampaikan dimanapun dan kapanpun.

Galih Budi (2016), Aplikasi Pendaftaran Mahasiswa Baru Berbasis *web* STMIK Akakom Yogyakarta. Dalam batasan masalah yang dibuat oleh penulis terdapat tes seleksi secara *online* ketika melakukan pendaftaran *online* yaitu seleksi berdasarkan tes ujian *online*. Hal ini dapat mempengaruhi minat peserta didik baru yang ingin mendaftar ke Sekolah tersebut. Mengacu pada tes ujian *online*, siswa baru yang telah mendaftar masih harus datang langsung ke sekolah untuk mengikuti tes ujian *online* sebagai salah satu syarat diterima menjadi siswa sekolah tersebut, hal ini menyebabkan tidak efisiannya waktu. Sedangkan untuk pihak sekolah sendiri harus menyediakan fasilitas seperangkat komputer untuk mereka yang mengikuti tes ujian *online*. Hal ini menyebabkan membengkaknya biaya sekolah. Dari beberapa alasan di atas, maka kami mengusulkan agar tes ujian masuk secara *online* ditinjau kembali dalam hal penghematan waktu dan biaya.

Dari beberapa tinjauan pustaka diatas diharapkan bisa membuat aplikasi yang dapat melengkapi atau menyempurnakan dari kekurangan-kekurangan tersebut.

## 2.2 Kerangka Pemikiran

Kerangka pemikiran dalam penelitian ini dapat digambarkan pada Gambar 2.1.



Gambar 2.1 Kerangka Pemikiran Penelitian

Keterangan Diagram Kerangka Pemikiran adalah sebagai berikut:

1) Latar Belakang Masalah

Identifikasi adanya pengolahan sistem secara manual yang menjadi sebuah permasalahan di LKP Indobank Cepu. Permasalahan yang di bahas merupakan permasalahan penerimaan peserta didik baru hanya meliputi proses pendaftaran, tes untuk mengetahui kemampuan calon peserta didik dan pelaporan peserta didik baru.

2) Rumusan Masalah

Bagaimana membuat aplikasi penerimaan peserta didik baru pada LKP Indobank berbasis *web*.

3) Judul Tugas Akhir

Judul yang sekiranya tepat untuk menangani permasalahan yang ada di LKP Indobank Cepu.

4) Pengumpulan Data

Semua data yang di butuhkan dikumpulkan, baik melalui *interview* dengan beberapa pihak terkait antara lain dengan bapak Istiadi S.Kom selaku direktur, ibu Siti Wahidah SE dan beberapa karyawan. Selain itu juga melakukan Observasi di LKP Indobank Cepu.

5) Penguasaan Dasar (PHP dan *MySQL*)

Aplikasi sederhana di buat sebagai bahan percobaan dengan tujuan agar dapat lebih menguasai bahasa pemrograman PHP dan database *MySQL*, sehingga hasilnya lebih maksimal.

6) Observasi Aplikasi

Aplikasi yang sudah ada dilakukan pengamatan, baik dari karya ilmiah, buku atau internet sehingga dapat dijadikan referensi untuk membangun aplikasi.

7) Analisis dan Perancangan Sistem

Analisis dan perancangan sistem pada tugas akhir ini menggunakan metode terstruktur. Analisis sistem terdiri dari sistem yang sedang berjalan yang meliputi teknik normalisasi, *entity relationship diagram* (ERD), data *flow diagram* (DFD), kamus data dan *flowchart*.

#### 8) Implementasi dan Pengembangan Sistem

Menerapkan dan membangun sistem yang telah di buat, mulai dari penulisan kode program menggunakan PHP, *WEB* dan databasenya *MYSQL*, instalasi pada pengembangan aplikasi dan pelatihan *user*.

#### 9) Pengujian Sistem

Pengujian sistem ini dilakukan bertujuan untuk mengetahui jika ternyata masih ada kesalahan ataupun kekurangan pada sistem yang telah dibuat. Metode pengujian sistem yang digunakan adalah *black box*.

#### 10) Penerapan Sistem dan Dokumentasi

Sistem telah siap digunakan setelah melewati tahap pengujian dan membuat dokumentasi dari keseluruhan kegiatan penyusun tugas akhir.

## 2.3 Teori Pendukung

### 2.3.1 Aplikasi

Aplikasi adalah suatu perangkat lunak yang dibuat khusus untuk memenuhi kebutuhan berbagai aktivitas dan pekerjaan, misalnya; pelayanan masyarakat, aktivitas niaga, periklanan, *game*, dan berbagai aktivitas lainnya (Hengky W. Pramana, 2015). Istilah aplikasi sendiri diambil dari bahasa Inggris "*application*" yang dapat diartikan sebagai penerapan atau penggunaan. Dalam pengembangannya, aplikasi dapat dikategorikan dalam tiga kelompok, diantaranya;

- a. Aplikasi *desktop*, yaitu aplikasi yang hanya dijalankan di perangkat PC komputer atau laptop.
- b. Aplikasi *web*, yaitu aplikasi yang dijalankan menggunakan komputer dan koneksi internet.
- c. Aplikasi *mobile*, yaitu aplikasi yang dijalankan di perangkat mobile di mana untuk kategori ini penggunaannya sudah banyak sekali.

Umumnya suatu aplikasi dapat berjalan di berbagai perangkat yang dioperasikan oleh *operating system* (OS) yang ada di perangkat tersebut. Adapun beberapa kriteria yang menandakan suatu aplikasi berkualitas dan bermanfaat bagi penggunanya;

- a. Aplikasi dapat memenuhi kebutuhan *user*.
- b. Aplikasi dapat berjalan di *multi-platform*.
- c. Aplikasi dapat merespon instruksi dengan cepat serta membutuhkan *resource (processor, memory, storage)* yang rendah.

### **2.3.2 Penerimaan Peserta Didik Baru (PPDB)**

Penerimaan peserta didik baru (PPDB) adalah kegiatan penerimaan dan seleksi calon peserta pendidikan dan pelatihan pada sekolah atau lembaga pendidikan, hal tersebut berkaitan dengan kemampuan dasar akademik dan minat bakat terhadap jenjang sekolah yang dituju sebagai bentuk awal pengendalian penjaminan dan penetapan mutu pendidikan. Dalam mendukung upaya jenjang pendidikan kearah tujuan yang diinginkan.

PPDB merupakan salah satu unsur dari komponen Siswa dalam sebuah lembaga pendidikan. PPDB dengan segala sistemnya, dilakukan untuk mengetahui dan mengukur Input sekolah guna membantu perkembangan sekolah serta diharapkan dapat memberikan kontribusi yang tinggi dalam merencanakan dan melaksanakan kegiatan pendidikan dan pelatihan dimasa yang akan datang. Proses seleksi yang dilakukan adalah suatu proses penilaian terhadap kemampuan awal calon peserta didik dari sisi kemampuan akademik, minat dan bakat peserta dengan menjadikan semua bukti hasil seleksi calon peserta sebagai acuan pengambilan keputusan dalam menentukan kelulusan dan keabsahan siswa yang diterima.

Proses seleksi yang dilaksanakan pada tingkat sekolah, merupakan bagian integral dari upaya peningkatan mutu calon peserta didik sekolah serta sebagai gambaran awal bagi pelayanan pendidikan dan selanjutnya dapat dipergunakan sebagai bahan pembinaan dan pengembangan siswa pada pelaksanaan kegiatan belajar mengajar.

Penerimaan Peserta Didik Baru (PPDB) merupakan proses penting untuk menjaring calon siswa baru sesuai dengan kriteria sekolah (Lampiran Peraturan Dinas Pendidikan Kabupaten Blora, Nomor : 522.1/162 Tahun 2011.)

Penerimaan Peserta Didik Baru (PPDB) perlu dilaksanakan secara objektif, transparan, akuntabel, dan tidak diskriminatif karena menyangkut kepentingan

esensial masyarakat yaitu pendidikan. Masyarakat akan memantau proses ini secara seksama Apakah representasi kepentingannya sudah berakomodasi atau belum. Keseluruhan proses itu diharapkan dapat menjamin proses PPDB yang akuntabel sehingga dapat mendorong terciptanya sistem pendidikan yang berkualitas.

### **2.3.3 Analisis Sistem**

Analisis sistem adalah proses kerja untuk menguji sistem informasi yang sudah ada dengan lingkungannya sehingga diperoleh petunjuk berbagai kemungkinan perbaikan yang dapat dilakukan dalam meningkatkan kemampuan sistem (Mardi, 2011). Analisis sistem ditujukan untuk menyediakan tim proyek dalam pemahaman yang lebih menyeluruh terhadap masalah-masalah dan kebutuhan-kebutuhan yang memicu proyek. Area lingkup proyek yang didefinisikan selama awal proyek dipelajari dan dianalisis untuk memperoleh pemahaman yang lebih rinci mengenai apa yang berkerja dan tidak berkerja dan apa yang dibutuhkan.

### **2.3.4 Analisis dan Perancangan Sistem**

Analisis dan perancangan sistem informasi berisis serangkaian langkah dan dokumentasi baku yang harus diikuti oleh pengembang sistem informasi untuk menjamin diperolehnya sistem informasi yang bisa menyelesaikan permasalahan-permasalahan dan mempermudah pengembangan sistem dimasa depan.

Analisis sistem didefinisikan sebagai bagaimana memahami dan menspesifikasi dengan detail apa yang harus dilakukan oleh sistem. Sementara sistem desain menjelaskan dengan detail bagaimana bagian-bagian dari sistem informasi diimplementasikan. Analisis dan desain sistem informasi bisa didefinisikan sebagai proses organisasional kompleks dimana sistem informasi berbasis komputer diimplementasikan (Hanif Al. Fatta, 2016).

### **2.3.5 Perancangan Sistem Secara Terstruktur**

Melakukan perancangan sistem secara terstruktur, dilakukan tinjauan pustaka untuk memahami dan menspesifikasi apa yang harus dilakukan sistem.

a. Teknik Normalisasi (*Normalization*)

Normalisasi perancangan sistem secara terstruktur diartikan sebagai suatu teknik yang menstrukturkan/memecah/mendekomposisi data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengelolaan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpanan-penyimpanan (*anomallies*) yang terjadi akibat adanya kerangkapan data dalam relasi dan inefisiensi pengolahan.

Proses normalisasi akan menghasilkan relasi yang optimal, yaitu:

- 1) Memiliki struktur *record* yang konsisten secara logik.
- 2) Memiliki struktur *record* yang mudah untuk dimengerti.
- 3) Memiliki struktur *record* yang sederhana dalam pemeliharaan.
- 4) Memiliki struktur *record* yang mudah untuk ditampilkan kembali untuk memenuhi kebutuhan pemakai.
- 5) Minimalisasi kerangkapan data guna meningkatkan kinerja sistem.

Bentuk-bentuk normal *first norm form/1NF*, *second norm form/2NF*, dan *third norm form/3NF* dikemukakan oleh E.F.Codd. sedangkan bentuk normal *Boyce-Codd norm form /BCNF* dikemukakan oleh R.F.Boyce dan E.F.Codd. bentuk normal BCNF, *forth norm form/4NF*, dan *fifth norm form/5NF* dapat terjadi pada relasi-relasi yang membangun PK berupa *composite key*. Bentuk *Domain Key Norm Form/DKNF* dan *Restriction Union Norm Form/RUNF* dapat terjadi pada relasi-relasi yang bersifat sangat spesifik, sehingga tidak semua relasi memungkinkan untuk mencapai level ini.

Umumnya rancangan relasi dalam basis data telah optimal jika memenuhi kriteria 3NF. Level normalisasi ditentukan berdasarkan kriteria bentuk normal, bukan banyaknya langkah menstrukturkan/dekomposisi/pemecahan sebuah relasi. Teori normalisasi dibangun menurut konsep level normalisasi. Level normalisasi atau sering disebut sebagai bentuk normal suatu relasi dijelaskan berdasarkan kriteria tertentu pada bentuk normal. Bentuk normal yang dikenal hingga saat ini meliputi bentuk 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, DKNF dan RUNF (Hanif Al Fatta, 2016).

a. *First Normal Form (1NF)*

Bentuk normal yang pertama atau 1NF mensyaratkan beberapa kondisi dalam sebuah database, berikut adalah fungsi dari bentuk normal pertama ini. Menghilangkan duplikasi kolom dari tabel yang sama. Buat tabel terpisah untuk masing-masing kelompok data terkait dan mengidentifikasi setiap baris dengan kolom yang unik (*primary key*).

Pada intinya bentuk normalisasi 1NF ini mengelompokkan beberapa tipe data atau kelompok data yang sejenis agar dapat dipisahkan sehingga anomali data dapat di atasi. Contoh adalah ketika kita ingin menghapus, mengupdate, atau menambahkan data peminjam, maka kita tidak bersinggungan dengan data buku atau data penerbit. Sehingga inkonsistensi data dapat mulai di jaga.

b. *Second normal form (2NF)*

Syarat untuk menerapkan normalisasi bentuk kedua ini adalah data telah dibentuk dalam 1NF, berikut adalah beberapa fungsi normalisasi 2NF.

1. Menghapus beberapa subset data yang ada pada tabel dan menempatkan mereka pada tabel terpisah.
2. Menciptakan hubungan antara tabel baru dan tabel lama dengan menciptakan foreign key.
3. Tidak ada atribut dalam tabel yang secara fungsional bergantung pada candidate key tabel tersebut.

Pada intinya bentuk kedua ini adalah tidak boleh ada field yang berhubungan dengan field lainnya secara fungsional. Contoh Judul Buku tergantung dengan id\_Buku sehingga dalam bentuk 2NF judul buku dapat di hilangkan karena telah memiliki tabel master tersendiri.

c. *Third Normal Form (3NF)*

Normalisasi database dalam bentuk 3NF bertujuan untuk menghilangkan seluruh atribut atau field yang tidak berhubungan dengan primary key. Dengan demikian tidak ada ketergantungan transitif pada setiap kandidat key. Syarat dari bentuk normal ketiga atau 3NF adalah :



- 1) Memenuhi semua persyaratan dari bentuk normal kedua.
  - 2) Menghapus kolom yang tidak tergantung pada primary key.
- d. *Boyce Codd normal form* (BCNF)

Merupakan sebuah teknik normalisasi database yang sering disebut 3.5NF, memiliki hubungan yang sangat erat dengan bentuk 3NF. Pada dasarnya adalah untuk menghandle anomali dan overlooping yang tidak dapat di handle dalam bentuk 3NF. Normalisasi database bentuk ini tergantung dari kasus yang disediakan, tidak semua tabel wajib di normalisasi dalam bentuk BCNF.

### **2.3.6 Entity-Relationship Diagram (ERD)**

ERD berguna untuk memodelkan sistem yang nantinya akan dikembangkan basis datanya. Model ini juga membantu perancang basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antar data didalamnya. Bagi pemakai, model ini sangat membantu dalam hal pemahaman model sistem dan rancangan basis data yang akan dikembangkan oleh perancang basis data. Sebuah diagram ERD tersusun atas tiga komponen, yaitu:

#### **Entitas (*Entity*)**

Entitas menunjukkan obyek-obyek dasar yang terkait didalam sistem. Obyek dasar dapat berupa orang, benda ,atau hal yang keterangannya perlu disimpan didalam basis data. Untuk menggambarkan entitas dilakukan dengan mengikuti aturan sebagai berikut :

- 1) Entitas dinyatakan dengan symbol persegi panjang.
- 2) Nama entitas dituliskan didalam simbol persegi panjang.
- 3) Nama entitas berupa: kata benda, tunggal.
- 4) Nama entitas sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

Penentuan entitas dalam suatu sistem perlu dilakukan dengan cermat dan hati-hati. Tidak semua orang, benda atau hal dapat disebut entitas. Hanya orang, benda, dan hal yang terkait dengan sistem dan keterangannya perlu disimpan dalam basis data saja yang dapat disebut entitas (Andri. Kristanto, 2012).

Derajat relasi atau kardinalitas rasio adalah jumlah maksimum hubungan antara satu entitas dengan entitas lainnya. Derajat relasi dibagi menjadi beberapa jenis yaitu :

3. *One to one* (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

4. *One to Many* (1:M)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

5. *Many to Many* (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

a. Atribut (*Attribute*)

Menurut Tjiptono (2009;103) Atribut sering pula disebut sebagai *property*, merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan sebagai basis data. Atribut berfungsi sebagai penjelas sebuah entitas. Untuk menggambarkan atribut dilakukan dengan mengikuti aturan sebagai berikut:

- 1) Atribut dinyatakan dengan *symbol ellips*.
- 2) Nama atribut dituliskan didalam *symbol ellips*.
- 3) Nama atribut berupa: kata benda, tunggal.
- 4) Nama atribut sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.
- 5) Atribut dihubungkan dengan entitas yang bersesuaian dengan menggunakan sebuah garis.

b. Kerelasian Antar Entitas (*Relationship*)

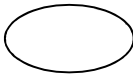
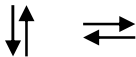
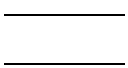
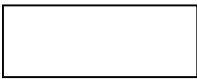
Mengenai pengertian Kerelasian Antar Entitas (*Relationship*) menurut Sutanta (2011:91). Kerelasian adalah kejadian atau transaksi yang terjadi diantara dua buah entitas yang keterangannya perlu disimpan dalam basis data. Kejadian atau transaksi yang tidak perlu disimpan dalam basis data bukan termasuk kerelasian. Aturan penggambaran kerelasian antar entitas adalah sebagai berikut :

- 1) Kereliasian dinyatakan dengan symbol belah ketupat.
- 2) Nama kereliasian dituliskan didalam symbol belah ketupat.
- 3) Kereliasian menghubungkan dua entitas.
- 4) Nama kereliasian berupa: kata kerja aktif (diawali dengan awalan me), tunggal.
- 5) Nama kereliasian sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

### 2.3.7 Diagram Alir Data (DAD)/ Data Flow Diagram(DFD)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Berikut adalah simbol-simbol data *Flow* diagram pada Tabel 2.1.

Tabel 2.1. Simbol-simbol data *Flow* diagram DFD

NO	NAMA SIMBOL	SIMBOL	KETERANGAN
1	Proses		Menunjukkan sebagai proses komputerisasi
2	Alir		Digunakan untuk menunjukkan arus dari proses
3	Penyimpanan		kumpulan data atau paket data hasil dari proses maupun penyediaan data untuk proses
4	Link		Digunakan untuk menghubungkan <i>entity</i> dengan atribut

DFD menggambarkan penyimpanan data dan proses yang mentransformasikan data. DFD menunjukkan hubungan antara data pada sistem dan proses pada sistem. Ada dua teknik dasar DFD yang umum dipakai yaitu Gane and Sarson dan Yourdon and De Marco (Andri Kristanto, 2012).

### 2.3.8 Kamus Data

Kamus data (*data dictionary*) yaitu tempat berkumpulnya semua jenis data yang terlibat dalam proses yang terjadi (Andri Kristanto, 2012). Menurut Kristanto (2016), kamus data adalah katalogi fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data selain digunakan untuk dokumentasi dan mengurangi redundansi, juga dapat digunakan untuk:

- 1) Memvalidasi diagram aliran data dalam hal kelengkapan dan keakuratan.
- 2) Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan.
- 3) Menentukan muatan data yang disimpan dalam file-file.
- 4) Mengembangkan logika untuk proses-proses diagram aliran data.

Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Tahap analisis sistem, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dalam pemakaian sistem tentang data yang mengalir disistem, yaitu tentang data yang masuk ke sistem tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Tahap perancangan sistem, kamus data dibuat berdasarkan arus data yang ada di DFD. Kamus data mendefinisikan elemen-elemen data dengan fungsi sebagai berikut (Aritonang, 2016):

- 1) Menjelaskan arti aliran data dan penyimpanan data dalam DFD.
- 2) Mendeskripsikan komposisi paket data yang bergerak melalui aliran.
- 3) Mendeskripsikan komposisi penyimpanan data.
- 4) Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran.
- 5) Mendeskripsikan hubungan detail antara penyimpanan (yang akan menjadi titik perhatian dalam *entity-relationship* diagram).

### 2.3.9 Flowchart

Menurut Indrajani (2017), *flowchart* adalah bagian yang menunjukkan aliran data didalam program atau prosedur sistem secara logika. Tujuan utama dari penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, dan jelas dengan menggunakan

symbol-simbol standar. Tahap penyelesaian masalah yang disajikan harus jelas, efektif, dan tepat.

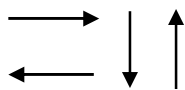


*Flowchart* digunakan sebagai alat bantu komunikasi, maka dalam membuat sistem *flowchart* ini harus mampu menunjukkan pekerjaan secara keseluruhan dari yang akan dibuat. Menurut Sterneckert (2010) yang dikutip oleh Indrajani (2017), menyarankan untuk membuat model *flowchart* yang berbeda sesuai dengan perspektif pemakaian sehingga dikenal ada empat jenis diagram alir secara umum:

- 1) Diagram Alir Dokumen, menunjukkan kontrol dari sistem aliran dokumen.
- 2) Diagram Alir Data, menunjukkan kontrol dari sebuah sistem aliran data.
- 3) Diagram Aliran Sistem, menunjukkan kontrol dari sistem aliran secara fisik.
- 4) Diagram Aliran Program, menunjukkan kontrol dari program sebuah sistem.

a. *Flowchart* Sistem

*Flowchart* Sistem merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Dengan kata lain, *flowchart* ini merupakan dekripsi secara grafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem. Berikut adalah simbol - simbol *Flowchart* sistem pada Tabel 2.2.

Tabel 2.2. Simbol-Simbol *Flowchart* Sistem

No	Simbol	Nama Simbol	Keterangan
1		<i>Arus / Flow</i>	Penghubung antara prosedur / proses
2		<i>Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang sama
3		<i>Off-line Connector</i>	Simbol keluar / masuk prosedur atau proses dalam lembar / halaman yang lain

*Flowchart* Sistem terdiri dari data yang mengalir melalui sistem dan proses yang mentransformasikan data itu. Data dan proses dalam *flowchart* sistem dapat digambarkan secara *online* (dihubungkan langsung dengan komputer) atau *offline* (tidak dihubungkan langsung dengan komputer, misalnya mesin tik, cash register atau kalkulator).


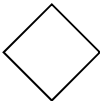
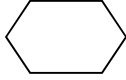
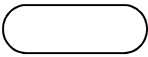

b. *Flowchart* Skematis

Sekilas *flowchart* skematis tampak menyerupai *flowchart* sistem, karena memang kedua *flowchart* ini sama-sama digunakan untuk menggambarkan prosedur/proses dalam sistem. *Flowchart* skematis (*schematic flowchart*) memiliki simbol yang lebih beragam. Simbol *flowchart* skematis lebih kompleks menggunakan gambar-gambar komputer dan peralatan lainnya. Hal ini bertujuan untuk memudahkan orang-orang yang tidak familier dengan simbol-simbol dalam *flowchart* untuk membaca informasi yang disampaikan.

c. *Flowchart* Program

Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Berikut adalah simbol-simbol *Flowchart* Program pada Tabel 2.3.

Tabel 2.3. Simbol-Simbol *Flowchart* Program

No	Simbol	Nama Simbol	Keterangan
1		<i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan Komputer
2		<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban / aksi
3		<i>Predefined Process</i>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan didalam storage
4		<i>Terminal</i>	Simbol untuk permulaan atau akhir dari suatu program
5		<i>Manual Input</i>	Simbol untuk pemasukan data secara <i>manual on-line keyboard</i>

Bagan alir program dibuat dari derivikasi bagan alir sistem. Bagan alir program dapat terdiri dari dua macam, yaitu bagan alir logika program (*program logic flowchart*) dan bagan alir program komputer terinci (*detailed computer program flowchart*). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir logika program ini dipersiapkan oleh analis sistem.

### 2.3.10 WEB

Menurut (Yuhefizar, 2014), “*Web* adalah suatu metode untuk menampilkan informasi di internet, baik berupa teks, gambar, suara maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan (*link*) satu dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui sebuah *browser*”. *Web* menyediakan informasi bagi pemakai komputer yang terhubung dengan internet. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Adapun cara kerja *Web* adalah sebagai Berikut :

1. Informasi *web* disimpan dalam dokumen dalam bentuk halaman-halaman *web* atau *web page*.
2. Halaman *web* tersebut disimpan dalam komputer *server web*.
3. Sementara dipihak pemakai ada komputer yang bertindak sebagai komputer *client* dimana ditempatkan program untuk membaca halaman *web* yang ada di *server web (browser)*.
4. *Browser* membaca halaman web yang ada di *server web*.

#### a. PHP

Menurut (Swastika, 2015) “PHP merupakan bahasa berbentuk *scrip* yang ditempatkan dalam server dan diproses di server, hasilnya akan dikirimkan ke *client*, tempat pemakai menggunakan browser”. *PHP (Hypertext Preprocessor)* adalah sebuah bahasa pemrograman yang didesain secara spesifik untuk membuat

sebuah situs *web* yang bersifat dinamis dan interaktif. PHP adalah bahasa pemrograman *web* yang sangat populer, bahkan dapat dikatakan sebagai bahasa pemrograman *web* yang paling banyak digunakan. PHP pertama kali dikembangkan oleh Rasmus Lerdorf pada tahun 1995, dimana pada saat itu kepanjangan dari PHP masih berupa *Personal Home Page*. PHP pada awalnya hanyalah ditujukan untuk membuat sebuah situs web pribadi yang sederhana, namun PHP ternyata menjadi sangat populer dan kemudian berkembang menjadi sebuah bahasa pemrograman web yang lengkap seperti sekarang ini.

Dalam penggunaannya, PHP akan menyisipkan perintah-perintah khusus di dalam sebuah dokumen HTML. Perintah-perintah PHP pada umumnya dikenal dengan istilah *PHP statemen*. Sebuah dokumen HTML yang telah disisipkan pernyataan-pernyataan PHP akan menjadi sebuah dokumen PHP atau yang juga dikenal dengan istilah *PHP script*.

#### b. *MySQL*

“*MySQL* adalah *multiuser database* yang menggunakan bahasa *structured query language* (Bimo Sunarfrihantono, 2012 : 65)”. *MySQL* dalam operasi *client-server* melibatkan *server daemon MySQL* di sisi *server* dan berbagai macam program serta *library* yang berjalan di sisi *client*. *MySQL* mampu menangani data yang cukup besar. *SQL* adalah bahasa standar yang digunakan untuk mengakses *database server*. Bahasa ini awalnya dikembangkan oleh IBM, namun telah diadopsi dan digunakan sebagai standar industri. Dengan menggunakan *SQL*, proses akses *database* menjadi lebih *user-friendly* dibandingkan dengan menggunakan *dBASE* atau *clipper* yang masih menggunakan perintah-perintah pemrograman.

*MySQL* merupakan program yang *multi-threaded*, sehingga dapat dipasang pada *server* yang memiliki *multi-CPU*.

- a. Didukung program-program umum seperti *C*, *C++*, *Java*, *Perl*, *PHP*, *Python*, *TCL APIs* dll.
- b. Bekerja pada berbagai *platform*. (tersedia berbagai versi untuk berbagai sistem operasi).



- c. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem database.
- d. *MySQL* merupakan *software* yang *free*. Sedangkan *software database* lainnya seperti *ORACLE* merupakan *software* yang harus di beli.
- e. *MySQL* dan *PHP* saling terintegrasi. Maksudnya adalah pembuatan database dengan menggunakan sintak *PHP* dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan script server-side seperti *PHP* dapat langsung dimasukkan ke database *MySQL* yang ada di server dan tentunya web tersebut berada di sebuah web server.

### 2.3.11 Pengujian Sistem

Menurut Quadri dan Farooq (2014: 1), pengujian sistem adalah proses verifikasi dan validasi apakah sebuah aplikasi *software* atau program memenuhi persyaratan bisnis dan persyaratan teknis yang mengarahkan desain dan pengembangan dan cara kerjanya seperti yang diharapkan dan juga mengidentifikasi kesalahan yang penting yang digolongkan berdasarkan tingkat severity pada aplikasi yang harus diperbaiki.

Menurut Nidhra dan Dondeti (2012:1), pengujian sistem adalah teknik yang sering digunakan untuk verifikasi dan validasi kualitas suatu software. Pengujian software adalah prosedur untuk eksekusi sebuah program atau sistem dengan tujuan untuk menemukan kesalahan.

Kesimpulan yang dapat diambil dari pendapat-pendapat tersebut adalah pengujian sistem merupakan proses verifikasi dan validasi apakah software memenuhi requirement dan mengidentifikasi kesalahan-kesalahan yang ditemukan saat eksekusi program.

Pengujian sistem pada tugas akhir ini menggunakan metode *black box*. Menurut Black (2012 :3), Tester menggunakan *Black-Box Tests*, sering digunakan untuk menemukan *bug* dalam *high level operations*, pada tingkatan fitur, profil operasional dan skenario customer. Tester dapat membuat pengujian fungsional *black box test* berdasarkan pada apa yang harus sistem lakukan. *black box test* melibatkan pemahaman rinci mengenai domain aplikasi, masalah bisnis yang dipecahkan oleh sistem dan misi yang dilakukan sistem. *black box test* paling baik

dilakukan oleh penguji yang memahami desain sistem, setidaknya pada tingkat yang tinggi sehingga mereka dapat secara efektif menemukan bug umum untuk jenis desain. Menurut Nidhra dan Dondeti (2012:1), *black box test* juga disebut *functional testing*, sebuah teknik pengujian fungsional yang merancang *test case* berdasarkan informasi dari spesifikasi.