

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu ini menjadi satu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Pada penelitian sebelumnya yang dilakukan (Desita, 2021) dalam jurnal yang berjudul **“Sistem Informasi Penerimaan Beasiswa Bagi Pegawai Negeri Sipil (PNS) Tugas Belajar Berbasis Web di Kantor Badan Pengembangan Sumber Daya Manusia (BPSDM) Aceh”**. Permasalahannya ialah Fakta yang ditemukan di lapangan bahwa proses pendaftaran dan seleksi penerima beasiswa saat ini masih dilakukan secara manual yaitu calon penerima beasiswa mendatangi kantor BPSDM untuk mendaftarkan diri dengan cara mengisi formulir identitas serta membawa seluruh berkas persyaratan untuk diserahkan ke panitia, sehingga terjadi antrian pada saat panitia melakukan pengecekan kelengkapan berkas persyaratan tersebut. Permasalahan selanjutnya, pada saat admin (staff BPSDM) mencari data penerima beasiswa Tugas Belajar tahun-tahun sebelumnya, membutuhkan waktu lama karena terlalu banyaknya berkas-berkas pendaftar beasiswa dikarenakan penyimpanan dokumen belum terkomputerisasi masih dalam bentuk kertas di lemari arsip. Jika mekanisme pengelolaan data masih dilakukan secara manual, mengakibatkan banyaknya berkas-berkas yang sulit untuk di temukan serta kurang efektif dimana panitia seleksi membutuhkan ketelitian dan waktu yang cukup lama untuk merekap semua berkas dokumen yang terjadi pada hari itu.

Oleh karena itu, penelitian tersebut akan membuat suatu sistem yang dibangun dapat membantu proses pemberian beasiswa mulai dari pendaftaran online, penginputan identitas diri dari calon penerima beasiswa, riwayat pendidikan terakhir, serta jurusan, jenjang, dan Universitas yang akan ditempuh oleh calon

penerima beasiswa serta pengumuman hasil seleksi siapa saja yang berhak menerima Beasiswa Tugas Belajar Bagi Pegawai Negeri Sipil.

Pada penelitian sebelumnya yang dilakukan (Hardiansyah, 2020) dalam jurnal yg berjudul “**Perancangan Basis Data Sistem Informasi Perwira Tugas Belajar (Sipatubel) Pada Kementerian Pertahanan**”. Permasalahannya ialah pada kasus ini salah satu Kementerian Republik Indonesia yaitu Kementerian Pertahanan (Sekretariat Jendral Biro Kepegawaian) mempunyai masalah pendataan dan pendataan itu adalah pendataan pegawai atau perwira Kemhan yang mendapatkan izin belajar di luar lembaga Kemhan dan pendataan perwira yang sedang melaksanakan tugas belajar atau izin belajar. Sekretariat Jendral Biro Kepegawaian Kementerian Pertahanan atau Setjen Ropeg Kemhan sedang merancang aplikasi yang dinamakan SIPATUBEL yang merupakan singkatan dari Sistem Informasi Perwira Tugas Belajar. Pada SIPATUBEL ini, mereka sudah mendapat konsep aplikasinya saja tetapi untuk basis data mereka masih memiliki kekurangan untuk merancang basis data SIPATUBEL.

Pada penelitian sebelumnya yang dilakukan (Fahdian, 2021) dalam jurnal yang berjudul “**Aplikasi Izin, Mutasi dan Pensiun pada Dinas Pekerjaan Umum dan Penataan Ruang Kota Banjarbaru**”. Permasalahan yang dihadapi semua pemrosesan data khususnya pada bagian kepegawaian masih dilakukan secara manual dimana dalam proses pembuatan permintaan izin kerja, mutasi dan pensiun dalam pembuatan laporan setiap bulannya masih menggunakan Microsoft Word maupun Excel, sehingga sering kali menghadapi permasalahan dalam kegiatan oprasionalnya selain itu penyimpanan file juga tersebar secara terpisah di komputer yang berbeda-beda sehingga menyulitkan pengumpulan data secara lengkap, sehingga membutuhkan waktu yang cukup lama untuk memproses data yang terkumpul. Dengan adanya masalah tersebut maka diperlukan suatu Aplikasi website manajemen izin sehingga Memudahkan dalam pembuatan laporan dan pemrosesan data izin kerja, mutasi dan pensiun terbaru serta dapat mempermudah para karyawan melakukan proses izin kerja, mutasi dan pensiun serta mendapatkan informasi-informasi yang berhubungan dengan izin secara lengkap dan terperinci, bagian kepegawaian diharapkan dapat membuatkan sistem informasi secara online.

Pada penelitian sebelumnya yang dilakukan (Erlangga, 2021) dalam jurnal yang berjudul “**Aplikasi Mutasi Kerja dan Pengusulan Pensiun Pada Dinas Perkebunan dan Peternakan Provinsi Kalimantan Selatan**”. Permasalahannya pada proses pengolahan data mutasi kerja dan pensiun staf masih menggunakan software pengolah angka, perekapan data masih menggunakan catatan pada buku besar secara manual, sehingga staf mengalami kesulitan dalam proses pencarian data ketika pegawai yang sudah mutasi kerja atau pensiun ingin mencari data lamanya. Pegawai yang ingin mutasi kerja atau pensiun harus menghubungi bagian staf terlebih dahulu untuk mengetahui semua persyaratan, jika ada berkas pengusul belum lengkap maka staf akan menghubungi pegawai yang bersangkutan untuk melengkapinya kembali sehingga membuat waktu pengerjaan kurang efisien. Oleh karena itu dibutuhkan sistem yang dapat memudahkan pegawai mendapatkan informasi dimana dan kapan saja, membuat proses pendataan lebih mudah dan efisien dengan adanya pencarian data mutasi kerja dan pensiun.

Pada penelitian sebelumnya yang dilakukan (Anwari, 2021) dalam jurnal yang berjudul “**Aplikasi Sistem Informasi Pensiun Pendidik dan Tenaga Kependidikan Pegawai Negeri Sipil pada Dinas Pendidikan dan Kebudayaan Provinsi Kalimantan Selatan**”. Dalam hal pengolahan data pensiun Pendidik dan Tenaga Kependidikan Pegawai Negeri Sipil (PNS) masih menggunakan aplikasi Microsoft Excel, sehingga proses pengolahan data pensiun Pendidik dan Tenaga Kependidikan Pegawai Negeri Sipil (PNS) Dinas Pendidikan dan Kebudayaan Provinsi Kalimantan Selatan mengalami kesulitan dalam mendeteksi kesalahan dalam penginputan data pensiun Pendidik dan Tenaga Kependidikan Pegawai Negeri Sipil (PNS) Dinas Pendidikan dan Kebudayaan Provinsi Kalimantan Selatan dan sering terjadi keliruandalam pemberian laporan data pensiun Pendidik dan Tenaga Kependidikan Pegawai Negeri Sipil (PNS) Dinas Pendidikan dan Kebudayaan Provinsi Kalimantan Selatan. dalam hal pengolahan berkas dan pengarsipan masih tergolong kurang praktis, kurangnya pengelolaan data dan informasi dalam hal waktu penggunaannya, dalam sistem informasi pensiun pendidik dan tenaga kependidikan pegawai negeri sipil pada dinas pendidikan dankebudayaan provinsi kalimantan selatan belum memiliki adanya sistem yang

dapat mempermudah pengelolaan data pensiun. Maka dari itu dibutuhkan sebuah sistem informasi yang dapat mengelola data kepengsianan pendidik dan tenaga kependidikan yang mendukung jalannya proses pensiun Pegawai Negeri Sipil.

Dari penelitian-penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis. Pada judul laporan akhir penulis disini mengimplementasikan sistem yang akan dibuat didasari dengan penelitian-penelitian terdahulu untuk dapat diterapkan pada penelitian ini. Dari penelitian-penelitian tersebut belum adanya tampilan surat disposisi berkas data pensiun dalam pengerjaannya dan tidak adanya tampilan grafik pada halaman aplikasi untuk melihat jumlah data pensiun dalam perbulan atau pertahun. Maka dari itu, penulis membuat laporan akhir dengan spesifikasi aplikasi tersebut.

2.2 Pengertian Sistem

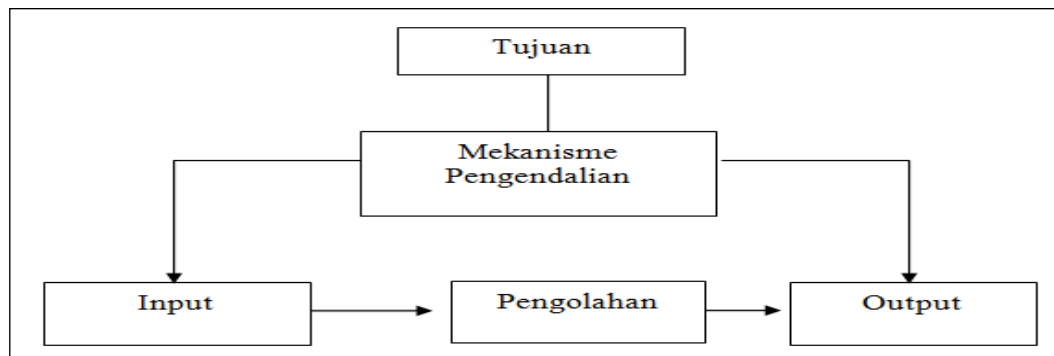
Sistem adalah suatu satuan (*entity*) yang terdiri dari dua atau lebih komponen yang erat berhubungan dan saling terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan tertentu.

Hal diatas dapat diperkuat oleh beberapa penulis lain yang mengungkapkan bahwa:

1. Menurut Mcleod (2004) sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan.
2. Menurut O'Brien (2005) sistem adalah sekelompok komponen yang saling berhubungan, bekerja bersama untuk mencapai tujuan bersama dengan menerima input serta menghasilkan output dalam proses transformasi yang teratur.
3. Menurut Sutanto dalam Djahir dan Pratita (2015) mengemukakan bahwa sistem adalah kumpulan atau grup dari subsistem atau bagian atau komponen apapun, baik fisik ataupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu.

2.3 Ele men-elemen Sistem

Dalam bukunya Sistem Informasi Manajemen, Mcleod (2004) mengatakan bahwa tidak semua sistem memiliki kombinasi elemen yang sama, tetapi suatu susunan dapat diperlihatkan pada gambar 2.1, berikut :



Gambar 2.1 Elemem-Element

Sistem Keterangan gambar :

Sumber daya input diubah menjadi sumber daya ouput. Sumber daya mengalir dari elemen input, melalui elemen transformasi, ke elemen output. Suatu mekanisme pengendali memantau proses transformasi untuk menyakinkan bahwa sistem tersebut memenuhi tujuannya. Sebagian sistem dapat mengendalikan operasi mereka sendiri yang disebut sistem lingkaran tertutup. Sebagian lagi sistem yang tidak dapat mengendalikan operasi mereka sendiri disebut sistem lingkaran terbuka.

2.4 Pengertian Informasi

Informasi adalah hasil dari proses pengolahan data yang memiliki arti dan berguna bagi yang menerimanya. Informasi sangat diperlukan oleh semua pihak baik organisasi ataupun perorangan.

Hal diatas dapat diperkuat oleh beberapa penulis lain yang mengungkapkan bahwa :

1. Menurut Mcleod (2004) informasi adalah suatu data yang telah diproses atau data yang memiliki arti.
2. Menurut Laudon (2004) informasi adalah data yang telah dibentuk ke dalam *form* yang berarti dan berguna untuk manusia.
3. Menurut O'Brien (2005) informasi adalah data yang dikonversi atau

diubah menjadi konteks yang berarti dan berguna bagi para pemakai akhir tertentu.

Informasi yang baik tentunya adalah sebuah informasi yang cepat, tepat dan akurat yang dapat mendukung pihak manajemen di dalam melakukan pengambilan keputusan. Baik atau buruknya keputusan tergantung dari kualitas informasi yang diterima oleh pemakai.

Menurut Parker (2003) syarat-syarat informasi yang baik sebagai berikut:

1. Ketersediaan : Tersedianya informasi merupakan syarat yang cukup mendasar sehingga dapat diperoleh untuk dimanfaatkan.
2. Mudah Dipahami : Informasi tidak boleh berbelit-belit tapi hendaknya mudah dipahami oleh setiap pembuat keputusan.
3. Relevan : Informasi yang diperlukan harus sesuai atau cocok dengan kebutuhan pengembangan organisasi.
4. Bermanfaat : Informasi harus mempunyai kegunaan bagi organisasi atau perorangan.
5. Tepat Waktu : Informasi harus tersedia tepat pada waktunya sesuai dengan kebutuhan.
6. Keandalan : Sumber informasi harus dapat dipercaya keberadaannya sehingga terjamin kebenarannya.
7. Akurat : Informasi harus bebas dari berbagai kesalahan atau kekeliruan komputasi dan transkripsi, sehingga jelas maknanya bagi kepentingan organisasi atau perorangan.
8. Konsisten : Informasi tidak boleh mengundang kontradiksi di dalam penyajiannya.
9. Jelas : Informasi harus bebas dari keraguan dan harus jelas keberadaannya.

- 10. Menyeluruh : Informasi harus lengkap dan utuh untuk kepentingan kegiatan tertentu.
- 11. Selektif : Informasi harus teruji keunggulannya.
- 12. Fleksibel : Informasi harus memiliki daya adaptasi terhadap kebutuhan yang berbeda.
- 13. Orisinal : Informasi harus asli dan tidak mengada-ada

2.5 Pengertian Sistem Informasi

Sistem informasi adalah sebuah sistem yang terdiri dari pengumpulan, pemasukan, pemrosesan data, penyimpanan, pengelolaan, pengendalian dan pelaporan sehingga tercapai sebuah informasi yang mendukung pengambilan keputusan di dalam organisasi sehingga organisasi dapat mencapai sasaran dan tujuannya.

Hal di atas dapat diperkuat oleh beberapa penulis lain yang mengungkapkan bahwa :

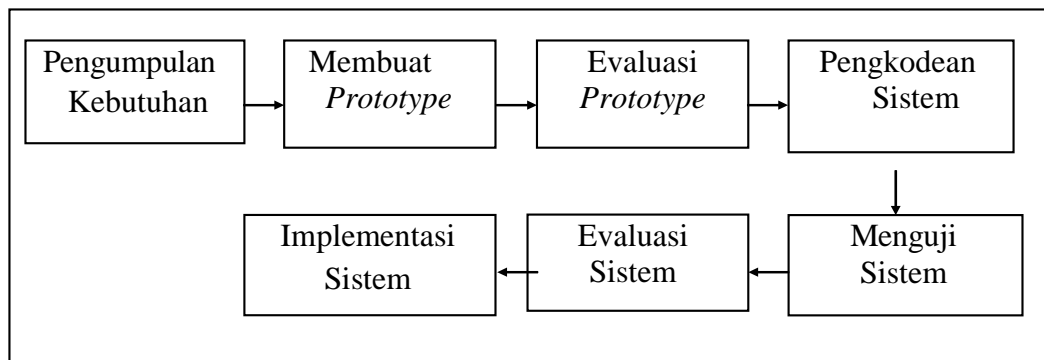
1. Menurut Laudon (2004) sistem informasi adalah sekumpulan komponen yang saling berhubungan dan fungsinya mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk membantu manajer dalam mengambil keputusan, menganalisis dan menggambarkan masalah yang kompleks dalam suatu organisasi.
2. Menurut O'Brien (2005) sistem informasi adalah kombinasi teratur apapun dari orang-orang, *hardware*, *software*, jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi.

2.6 Metode Pengembangan Sistem dengan Metode *Prototyping*

Metode yang digunakan dalam perancangan sistem ini yaitu *prototyping*. Hanif (2007) mengatakan *prototyping* sebagai proses iteratif dalam pengembangan sistem dimana kebutuhan diubah ke dalam sistem yang bekerja yang secara terus menerus diperbaiki melalui kerjasama antara pengguna dan analisa.

Menurut Sri (2016), *prototyping* merupakan teknik pengembangan sistem yang menggunakan *prototype* untuk menggambarkan sistem, sehingga pengguna atau pemilik sistem mempunyai gambaran pengembangan sistem yang akan dilakukannya. Dengan teknik *prototyping*, pengembang bisa membuat *prototype* terlebih dahulu sebelum mengembangkan sistem yang sebenarnya.

Berdasarkan pendapat para ahli diatas dapat ditarik kesimpulan bahwa *Prototype* didefinisikan sebagai satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam membentuk yang telah selesai. Semua rancangan diagram atau model yang dibuat tidak diharuskan telah sempurna dan final dalam *prototype*. Tujuan utama dari penyiapan rancangan adalah sebagai alat bantu dalam memberi gambaran sistem seperti materi dan menu yang perlu dimasukkan dalam *prototype* yang akan dikembangkan. Adapun tahapan-tahapan dalam *prototype* adalah sebagai berikut :



Gambar 2.2 Tahap Metode *Prototyping*

1. Pengumpulan Kebutuhan

Pengguna dan pengembang sama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan dan garis besar sistem yang akan dibuat.

2. Membuat *Prototype*

Membangun *Prototype* dengan perancangan sementara yang berfokus pada penyajian kepada pengguna. Contohnya dengan membuat *input* dan format *output*.

3. Evaluasi *Prototype*

Evaluasi ini dilakukan oleh pengguna apakah *prototype* yang sudah dibangun sesuai dengan keinginan pengguna. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak *prototype* direvisi dengan mengulangi langkah 1, 2, dan 3.

4. Pengkodean Sistem

Dalam tahap ini *prototype* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5. Menguji Sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus di uji dahulu sebelum digunakan.

6. Evaluasi Sistem

Pengguna mengevaluasi apakah sistem yang dibangun sudah sesuai dengan yang diharapkan. Jika telah selesai maka dilanjutkan dengan melakukan langkah ke 7. Jika belum maka mengulangi langkah 4 dan 5.

7. Implementasi Sistem

Impelementasi sistem atau menggunakan sistem yang berarti sistem telah diuji dan siap untuk digunakan.

2.7 Desain Sistem

Ada banyak pendapat mengenai pendefinisian *desain* sistem. Sebagai contoh, dalam buku Hartono (1989) :

1. Menurut Jhon Bruch & Gary Grudnitski, *desain* sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.

2. Menurut Robert J. Verzello/Jhon Reuter III, tahap setelah analisis dari siklus pengembangan sistem : pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi; menggambarkan bagaimana suatu sistem dibentuk.

3. Dan menurut George M. Scott pada buku, *desain* sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan; tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem.

Dengan demikian desain sistem dapat disimpulkan sebagai pendefinisian dari kebutuhan-kebutuhan fungsional untuk mempersiapkan rancang bangun implementasi yang berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen terpisah kedalam satu kesatuan yang utuh dan berfungsi.

2.7.1 Data Flow Diagram (DFD)

Kristanto (2008) mengatakan bahwa:

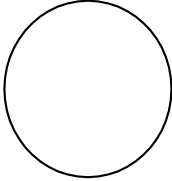

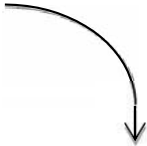
Data Flow Diagram merupakan suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Sukamto dan Shalahuddin (2014) mengatakan bahwa:

Data Flow Diagram atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (*input*) dan keluaran (*output*). *DFD* tidak sesuai untuk memodelkan sistem yang menggunakan pemrograman berorientasi objek.

Sukamto dan Shalahuddin (2014) mengatakan bahwa notasi-notasi pada *DFD* (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

| NOTASI | KETERANGAN |
|---|---|
|  | <p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p> |
|  | <p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan. Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p> |
|  | <p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.</p> |

Menurut Sukamto dan Shalahuddin (2014), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD *Level 0* atau sering disebut juga *Context Diagram*

DFD *Level 0* menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD *Level 0* digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD *Level 1*

DFD *Level 1* digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD *Level 1* merupakan hasil *breakdown* DFD *Level 0* yang sebelumnya sudah dibuat.

3. Membuat DFD *Level 2*

Modul-modul pada DFD *Level 1* dapat di *breakdown* menjadi DFD *Level 2*. Modul mana saja yang harus di *breakdown* lebih detail tergantung pada tingkat kedetilan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di *breakdown* lagi. Untuk sebuah sistem, jumlah DFD *Level 2* sama dengan jumlah modul pada DFD *Level 1* yang di *breakdown*.

4. Membuat DFD *Level 3* dan seterusnya.

DFD *Level 3, 4, 5* dan seterusnya merupakan *breakdown* dari modul pada DFD *Level* di atasnya. *Breakdown* pada *level 3, 4* dan *5* dan seterusnya aturannya sama persis dengan DFD *Level 1* atau *Level*.


2.7.2 Diagram Alir (*Flowchart*)


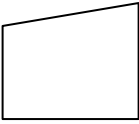
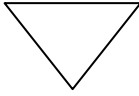
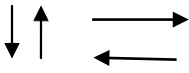
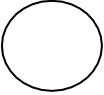
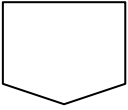
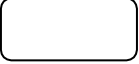
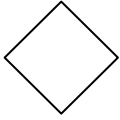
Kadir (2017) menjelaskan diagram alir dapat dikelompokkan ke dalam dua kategori, yaitu diagram alir sistem (*system flowchart*) dan diagram alir program (*program flowchart*).

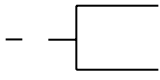
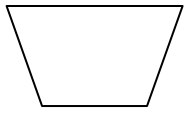
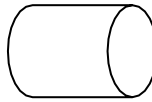


1. Diagram alir sistem biasa digunakan oleh analis sistem untuk menggambarkan aliran data atau struktur *file* di dalam suatu sistem.
2. Diagram alir program biasa dipakai oleh pemrogram untuk menjelaskan langkah-langkah di dalam suatu program.

Adapun simbol-simbol yang digunakan pada diagram alir adalah sebagai berikut:

Tabel 2.2 Simbol-simbol Diagram Alir (*Flowchart*)

| No | Nama | Simbol | Keterangan |
|----|---------|---|---|
| 1. | Dokumen |  | Sebuah dokumen atau laporan. dokumen dapat dibuat dengan tangan atau cetak oleh komputer. |

| | | | |
|----|------------------------------------|---|--|
| 2. | Pemrosesan Komputer |  | Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer, biasanya menghasilkan data atau informasi. |
| 3. | Keying (<i>typing verifying</i>) |  | Menunjukkan pemasukan data kedalam komputer melalui <i>onlineterminal</i> atau perangkat terminal <i>input/output</i> . |
| 4. | Arsip |  | Arsip dokumen disimpan dan diambil secara manual. Huruf didalamnya menunjukkan cara pengurutan arsip. |
| 5. | Arus Dokumen/ Pemrosesan |  | Arah arus dokumen atau pemrosesan; arus normal adalah kekanan atau kebawah. |
| 6. | Penghubung Dalam Sebuah Halaman |  | Menghubungkan bagian alir pada halaman yang sama. Simbol ini digunakan untuk menghindari terlalu banyak anak panah yang saling melintang. |
| 7. | Penghubung Pada Halaman Berbeda |  | Menghubungkan bagian alir pada halaman yang berbeda. Simbol ini digunakan untuk menghindari terlalu banyak anak panah yang saling melintang. |
| 8. | Terminal |  | Digunakan untuk memulai, mengakhiri atau titik henti dalam sebuah proses atau program. |
| 9. | Keputusan |  | Sebuah tahap pembuatan keputusan; digunakan dalam bagan alir program komputer untuk menunjukkan cabang bagi alternatif cara. |

| | | | |
|-----|-----------------------------|--|---|
| 10. | Anotasi |  | Tambahan penjelasan deskriptif atau keterangan, atau catatan sebagai klasifikasi. |
| 11. | Operasi Manual |  | Menunjukkan proses yang dikerjakan secara manual |
| 12. | Penyimpanan/ <i>Storage</i> |  | Menunjukkan akses langsung perangkat penyimpanan. |
| 13. | Masukan ke sistem | Dari Pemasok  | Karena kegiatan luar sistem tidak perlu digambarkan dalam bagan alir, maka diperlukan simbol untuk menggambarkan masuk ke sistem yang digambarkan dalam bagan alir. |
| 14. | Keluar sistem lain |  Ke sistem penjualan | Karena kegiatan luar sistem tidak perlu digambarkan dalam bagan alir, maka diperlukan simbol untuk menggambarkan keluar ke sistem yang lain. |

Sumber : Kadir (2017)

2.8 Basis Data (*Database*)

Basis Data terdiri dari kata basis dan data. Basis dapat diartikan sebagai markas atau gudang. Sedangkan data adalah catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasinya. Himpunan kelompok data yang saling terhubung dan diorganisasikan sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah. Kumpulan data dalam bentuk *file*/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronik, untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian

data sesuai tujuan.

Database mempunyai 8 operasi dasar diantaranya adalah *Create database*, *Drop database*, *create table*, *Drob table*, *Insert*, *Read*, *Update* dan *Delete*. Sebuah database menyimpan data di dalam tabel, dimana setiap tabel memiliki baris dan kolom. Data merupakan fakta atau angka. Tabel berisi tentang berbagai jenis hal, sebagai contoh data barang, data dosen. Setiap baris pada tabel yang berisi data yang disebut *record* dan setiap kolom yang menyimpan karakteristik umum untuk semua baris disebut *fields*.

Field adalah kumpulan dari karakter yang membentuk satu arti, maka jika terdapat *field* misalnya seperti NomorBarang atau NamaBarang, maka yang dipaparkan dalam *field* tersebut harus yang berkaitan dengan nomor barang dan nama barang. Atau definisi *field* yang lainnya yaitu tempat atau kolom yang terdapat dalam suatu tabel untuk mengisikan nama-nama (data) *field* akan di isikan.

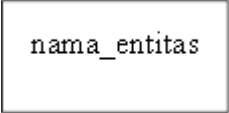
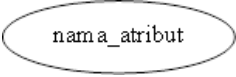

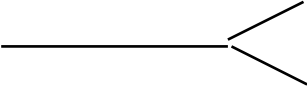
“Record adalah kumpulan *field* yang sangat lengkap, dan biasanya dihitung dalam satuan baris. Tabel adalah kumpulan dari beberapa *record* dan juga *field*. *File* adalah terdiri dari *record-record* yang menggambarkan dari satu kesatuan data yang sejenis. Misalnya seperti *file* nama barang berisikan data tentang semua nama barang yang ada” Rachmadi (2020).

2.9 Entity Relationship Diagram (ERD)

Sukamto dan Shalahuddin (2014) “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.”

Berikut adalah simbol-simbol yang digunakan pada ERD adalah sebagai berikut :

Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

| Simbol | Deskripsi |
|--|---|
| Entitas / <i>entity</i>  | Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel. |
| Atribut  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas |
| Relasi  | Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja |
| Asosiasi / <i>association</i>  | Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B. |

Sumber : Sukamto dan Shalahuddin (2018)

2.10 Pengertian SQL (*Structured Query Language*)

Sebuah bahasa untuk mengakses data didalam sebuah database relasional. Berikut ini akan diuraikan beberapa pengertian *Structured Query Language* (SQL) menurut para ahli.

Menurut Utami & Sukrisno (2008) “*SQL (Structured Query Language)* pada dasarnya adalah bahasa komputer standar yang ditetapkan untuk mengakses dan memanipulasi sistem *database*”.

Sedangkan menurut Priyadi (2014) “*Structured Query Language (SQL)* adalah bahasa pemrograman khusus untuk mengirim pencarian data (*query*) ke *database*”.

Berdasarkan penjelasan diatas dapat disimpulkan bahwa, *Structured Query Language* (SQL) adalah bahasa standar yang digunakan untuk mengakses atau mengirim pencarian data ke *database*.

2.11 Pengertian MySQL

Enterprise (2016), “MySQL merupakan server yang melayani *database*. Untuk membuat dan mengolah *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah)”. Sedangkan Adi dalam Radillah (2018), “MySQL adalah sebuah program *database* server yang mampu menerima dan mengirim datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL (*Structure Querred Language*)”.

Berdasarkan pendapat para ahli diatas dapat disimpulkan bahwa, MySQL adalah program database yang digunakan untuk membuat dan mengelola *database* dengan cepat, multi user serta menggunakan *query* (perintah).

2.12 Pengertian XAMPP

Madcoms (2018), “XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla dan lain-lain”.

Menurut Sidik (2019), menyatakan bahwa XAMPP ((Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket *server website* PHP dan *database* MySQL yang paling populer dikalangan pengembang *website* dengan

menggunakan PHP dan MySQL sebagai *database*.

Berdasarkan penjelasan di atas, dapat disimpulkan bahwa XAMPP merupakan sebuah paket kumpulan aplikasi server *website* PHP dan *database* MySQL.

2.13 Pengertian HTML

Kustiyahningsih & Anamisa (2017), Menjelaskan bahwa HTML kependekan dari *Hyper Text Markup Language* yang *file* teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *website page*. *File-file* HTML ini berisi instruksi-instruksi yang kemudian diterjemahkan oleh browser yang ada dikomputer *client (user)* sehingga isi formasinya dapat ditampilkan secara visual dikomputer pengguna (*user*).

Hidayatulah dan Kawistara (2018), menjelaskan tentang HTML sebagai berikut :

Hypertext Markup Language (HTML) adalah Bahasa standar yang digunakan untuk menampilkan halaman *website*. Yang bisa dilakukan dengan HTML yaitu:

1. Mengatur tampilan dan halaman *website* dan isinya.
2. Membuat tabel dalam halaman *website*.
3. Mempublikasikan dalam halam *website* secara *online*.
4. Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi via *website*.

Contoh: Setiap dokumen HTML diawali dan diakhiri dengan tag HTML.

Dari penjelasan di atas dapat disimpulkan bahwa *Hyper Text Markup Language* (HTML) adalah bahasa standar yang merupakan teks murni yang berisi instruksi yang diterjemahkan oleh browser dan ditampilkan secara *visual* di komputer pengguna.

2.14 Pengertian PHP (*Hypertext Preprocessor*)

Abdulloh (2016), "PHP singkatan dari *Hypertext Preprocessor* yang merupakan server-side programming, yaitu bahasa pemrograman yang diproses disisi server". Sedangkan menurut Putratama (2016), "PHP (PHP: Hypertext

Processor) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server-side yang dapat ditambahkan ke dalam HTML”.

Berdasarkan pendapat para ahli diatas dapat disimpulkan bahwa, PHP adalah bahasa pemograman berbasis server-side yang bisa kita gunakan untuk membuat aplikasi *website* yang ditanamkan kedalam HTML, yang dijalankan di dalam sebuah webserver, dan juga bisa menghasilkan tampilan *website* yang dinamis.

2.15 JavaScript

Suatu bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam *website* browser menjadi lebih interkatif. Berikut ini akan diuraikan beberapa pengertian *JavaScript* menurut para ahli.

Menurut Wahyono (2009) “*JavaScript* adalah yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen *HTML*”. Sedangkan Menurut Saputra (2012) “*Javascript* adalah bahasa pemograman yang digunakan untuk membuat *website* lebih dari dinamis dan interaktif”.

Berdasarkan penjelasan diatas dapat disimpulkan bahwa, *JavaScript* adalah kumpulan skrip yang fungsinya berjalan pada suatu dokumen HTML dan dapat digunakan dalam membuat *website* agar lebih dinamis.

2.16 BootStrap

Sebuah *framework* HTML dan CSS yang berfungsi untuk situs dan aplikasi *website*. Berikut ini akan diuraikan beberapa pengertian *BootStrap* menurut para ahli. Menurut Ridha (2007) “*Twitter Bootstrap* adalah sebuah alat bantu (*framework*) *HTML* dan *CSS* untuk membuat sebuah tampilan halaman *website* yang elegan dan *support* segala macam *device*.Sedangkan Menurut Rivaldi (2015) “*BootStrap* merupakan *Framework* ataupun *Tools* untuk membuat aplikasi *website* ataupun situs *website responsive* secara cepat, mudah dan gratis”.

Berdasarkan teori para ahli diatas dapat disimpilkan bahwa, *BootStrap* adalah sebuah alat bantu untuk membuat tampilan halaman *website* menjadi elegan, cepat,dan mudah.

2.17 *Codeigniter*

Sebuah *framework* berbasis PHP yang dibangun untuk membuat aplikasi web dengan sederhana dan elegan. Berikut ini akan diuraikan beberapa pengertian *CodeIgniter* menurut para ahli.

Menurut Ridha (2007) “*CodeIgniter* adalah sebuah *framework* PHP yang bersifat *Open Source* dan dapat digunakan untuk mempercepat pengembang dalam membuat aplikasi web. Sedangkan Menurut Andriyani (2016) “*CodeIgniter* (CI) merupakan aplikasi *open source* yang berupa *framework* dengan model *MVC* (*Model, View, Controller*) untuk membangun website dinamis menggunakan PHP”.

Berdasarkan teori para ahli diatas dapat disimpulkan bahwa, *CodeIgniter* adalah framework PHP dengan model *MVC* yang sifatnya open source untuk membangun website yang dinamis.

2.18 **Pengertian Pengujian Sistem**

Al Fatta (2007), Pengujian sistem merupakan proses mengeksekusi sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut cocok dengan spesifikasi sistem dan berjalan sesuai dengan lingkungan yang diinginkan. Pengujian sistem sering diasosiasikan dengan pencarian bug, ketidaksempurnaan program, kesalahan pada baris program yang menyebabkan kegagalan pada eksekusi sistem perangkat lunak.

2.19 *Visual Studio Code*

Visual Studio Code adalah editor *source code* yang dikembangkan oleh Microsoft untuk Windows, Linux dan MacOS. Ini termasuk dukungan untuk debugging, GIT Control yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan kode *refactoring*. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, *shortcut keyboard*, dan preferensi. Visual Studio Code gratis dan *open source*, meskipun unduhan resmi berada di bawah lisensi *proprietary*.

Kode Visual Studio didasarkan pada Elektron, kerangka kerja yang digunakan untuk menyebarkan aplikasi Node.js untuk desktop yang berjalan pada Blink Layout. Meskipun menggunakan kerangka Elektron, Visual Studio Code tidak menggunakan Atom dan menggunakan komponen editor yang sama (diberi kode nama “Monaco”) yang digunakan dalam Visual Studio Team Services yang sebelumnya disebut Visual Studio Online Lardinois (2015).

2.20 Black Box Testing

Menurut Pressman (2010),” *Black-Box* testing berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan engineers untuk memperoleh set kondisi input yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program”.

Black-Box testing berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan perilaku (behavior) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

2.21 Pengertian Hosting

Menurut Arifin dan Yolanda Krisnadita (2017), “Hosting merupakan tempat penyimpanan data website dimana didalamnya meliputi kapasitas penyimpanan, bandwidth yang merupakan sebuah kapasitas yang digunakan untuk mengukur jumlah pengunjung website serta database”.

Menurut Aliyun (2014) dalam Samsul Arifin Dkk, “Hosting juga memiliki arti layanan berbasis internet sebagai tempat penyimpanan data atau tempat menjalankan aplikasi ditempat terpusat yang disebut dengan sever dan dapat diakses melalui jaringan internet”.