



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Asropudin (2013:19) mengemukakan, “komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing computer”.

Kamus Besar Bahasa Indonesia Versi *Online* (2012-2016) mengemukakan, “alat elektronik otomatis yang dapat menghitung atau mengolah data secara cermat menurut yang diinstruksikan, dan memberikan hasil pengolahan, serta dapat menjalankan sistem multimedia (film, musik, televisi, faksimile, dan sebagainya), biasanya terdiri”.

Jadi, komputer adalah sebuah alat elektronik yang dapat memproses data sesuai dengan proses dan perintah-perintah yang telah diurutkan dan diprogramkan.

2.1.2. Pengertian Teknologi

Kamus Besar Bahasa Indonesia Versi *Online* (2012-2016) mengemukakan, “metode ilmiah untuk mencapai tujuan praktis, ilmu pengetahuan terapan, keseluruhan sarana untuk menyediakan barang-barang yang diperlukan bagi kelangsungan dan kenyamanan hidup manusia”.

Ngafifi (2014:42) mengemukakan, “teknologi merupakan salah satu aspek yang turut mempengaruhi setiap aktivitas, tindakan, serta perilaku manusia. Teknologi mampu mengubah pola hubungan dan pola interaksi antar manusia. Kehadiran teknologi merupakan sesuatu yang tidak dapat dipisahkan dari kehidupan manusia”.

Drucker dalam Djumli (2014:2) mengemukakan, “teknologi harus dianggap sebagai suatu sistem yaitu kumpulan dan satuan-satuan dan kegiatan-kegiatan yang saling berkaitan dan saling berkomunikasi”.



Jadi, teknologi adalah sebuah metode ilmiah untuk mencapai praktis sebagai suatu sistem yaitu kumpulan dan satuan-satuan yang turut mempengaruhi setiap aktifitas, tindakan, serta perilaku manusia yang saling berkaitan dan saling berkomunikasi.

2.1.3. Pengertian Perangkat Lunak (*Software*)

Yakub (2012:96) mengemukakan, “perangkat lunak terdiri dari rangkaian instruksi elektronik yang menyeluruh untuk mengerjakan sesuatu”.

Ladjamudin (2013:20) mengemukakan, “*software* merupakan kumpulan dari perintah/fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu”.

Sukanto dan Shalahuddin (2013:2) mengemukakan, “perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”.

Jadi, dapat disimpulkan bahwa perangkat lunak (*software*) adalah perangkat lunak dari rangkaian instruksi elektronik yang ditulis dengan aturan tertentu untuk memerintah komputer melaksanakan tugas seperti dokumentasi kebutuhan, model desain, dan cara penggunaan.

2.1.4. Pengertian Data

McLeod dalam Yakub (2012:5) mengemukakan, “data adalah kenyataan yang menggambarkan adanya suatu kejadian (*event*), data terdiri dari fakta (*fact*) dan angka yang secara relatif tidak berarti bagi pemakai”.

Asropudin (2013:22) mengemukakan, “data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi”.

Sutabri (2012:29) mengemukakan, “data merupakan kenyataan yang menggambarkan suatu kejadian serta merupakan suatu bentuk yang masih belum dapat bercerita banyak sehingga perlu diolah lebih lanjut melalui suatu model untuk menghasilkan informasi”.



Jadi, data adalah kenyataan terhadap fakta tentang suatu kejadian yang tidak berarti bagi pemakai untuk menghasilkan informasi.

2.1.5. Pengertian Pengolahan Data

Sutabri (2012:6) mengemukakan, “pengolahan data merupakan bahan mentah untuk diolah yang hasilnya kemudian menjadi informasi. Dengan kata lain, data yang telah diperoleh harus diukur dan dinilai baik dan buruk, berguna atau tidak dalam hubungannya dengan tujuan yang akan dicapai”.

Ladjamudin (2005:9) mengemukakan, “pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Jadi, pengolahan data adalah waktu yang digunakan dalam mengubah data menjadi informasi, sehingga informasi tersebut dapat dimanfaatkan oleh penggunaannya.

2.1.6. Pengertian Informasi

McLeod dalam Yakub (2012:8) mengemukakan, “informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya”.

Sutabri (2012:22) mengemukakan, “informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan”.

Kamus Besar Bahasa Indonesia Versi *Online* (2012-2016) mengemukakan, “penerangan, pemberitahuan, kabar atau berita tentang sesuatu, dan keseluruhan makna yang menunjang amanat yang terlihat dalam bagian-bagian amanat itu”.

Jadi, informasi adalah data yang diolah atau diklasifikasi menjadi lebih berguna dalam proses pengambilan keputusan yang menunjang amanat yang terlihat dalam bagian-bagian amanat itu.

2.1.7. Kualitas Informasi

Yakub (2012:9) mengemukakan “kualitas dari suatu informasi dapat dilihat dari dimensi-dimensi yang dimiliki oleh informasi”. Menurut Jogiyanto (dikutip



Yakub, 2012:9) kualitas dari informasi (*quality of information*) tergantung dari tiga hal yaitu: *accurate*, *timeliness*, dan *relevance*.

- a. Relevan (*relevance*), berarti informasi tersebut mempunyai manfaat untuk pemakainya dan relevansi informasi untuk tiap-tiap orang akan berbeda-beda.
- b. Tepat waktu (*timeliness*), berarti informasi tersebut datang pada penerima tidak boleh terlambat.
- c. Akurat (*accuracy*), berarti informasi harus bebas dari kesalahan-kesalahan, tidak menyesatkan dan harus jelas mencerminkan maksudnya.

2.1.8. Pengertian Sistem

Yakub (2012:2) mengemukakan, “sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu”.

Sutabri (2012:3) mengemukakan, “sistem adalah suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu”.

Jadi, dapat disimpulkan bahwa sistem adalah suatu kumpulan dari unsure, komponen yang terorganisasi, interaksi dan saling bergantung satu sama lain untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

2.1.9. Karakteristik Sistem

Sutabri (2012:13) menjelaskan tentang karakteristik atau sifat-sifat tertentu dari sistem yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut:

- a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.

- b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini



memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem adalah bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut.

d. Penghubung Sistem (*Interface*)

Penghubung sistem atau *interface* adalah media yang menghubungkan sistem dengan subsistem yang lain.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, di mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan *input* bagi subsistem lain.

g. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.



2.1.10. Klasifikasi Sistem

Sutabri (2012:15) menjelaskan bahwa sistem dapat diklasifikasikan dari beberapa sudut pandangan, yaitu:

a. Sistem abstrak dan sistem fisik.

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sistem fisik adalah sistem yang ada secara fisik, misalnya sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

b. Sistem alamiah dan sistem buatan manusia.

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sistem buatan manusia adalah sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*, misalnya sistem informasi komputer.

c. Sistem deterministik dan sistem probabilistik.

Sistem deterministik adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, misalnya sistem komputer yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sistem probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

d. Sistem terbuka dan sistem tertutup.

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya.



2.1.11. Metode Pengembangan Sistem

Sukamto dan Shalahuddin (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode 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 terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis.

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain.

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengodean.

Pada tahap pengodean, desain harus ditranlasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian.

Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau Pemeliharaan (*maintenance*).

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses



pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

2.1.12. Metode Pengujian Sistem

Yakub (2012:150) mengemukakan, “Metode pengujian sistem adalah suatu cara untuk menguji perangkat lunak dan data kemungkinan terjadi kesalahan”. Pengujian sistem dapat menggunakan metode *black box testing*. *Black box testing* berfokus pada pengujian persyaratan fungsional perangkat lunak yang dibuat untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program dan mampu mengungkapkan kesalahan yang lebih luas.

2.2. Teori Khusus

2.2.1. Data Flow Diagram (DFD)

Ladjamudin (2013:64) mengemukakan, “diagram aliran data/data flow diagram (*dfd*) merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil”.

Sukanto dan Shalahuddin (2013:70) mengemukakan, *dfd* dapat digunakan untuk mempresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. Ada 2 tahapan dalam *DFD*, yaitu:

a. Diagram Konteks.

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari *DFD* yang menggambarkan seluruh input ke sistem atau *output* dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Sistem dibatasi oleh *boundary* (dapat digambarkan dengan garis putus). Diagram konteks hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks.

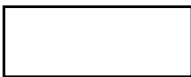
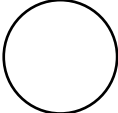
b. *DFD* Nol/Zero (Overview Diagram).

Diagram nol adalah diagram yang menggambarkan proses dari *dataflow* diagram. Diagram nol memberikan pandangan secara menyeluruh mengenai



sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal *entit*. Pada level ini sudah dimungkinkan adanya/digambarkannya data *store* yang digunakan. Untuk proses yang tidak dirinci lagi pada level selanjutnya, symbol ‘*’ atau ‘P (*functional primitive*)’ dapat ditambahkan pada akhir nomor proses. Keseimbangan *input* dan *output* (*balancing*) antara diagram 0 dengan diagram konteks harus terpelihara. Adapun simbol-simbol *Data Flow Diagram* adalah sebagai berikut:

Tabel 2.1. Simbol-Simbol *Data Flow Diagram* (DFD)

No.	Nama	Simbol	Keterangan
1.	Entiti Luar		Merupakan sumber atau tujuan dari aliran data dari atau ke sistem
2.	Aliran Data		Menggambarkan aliran data dari satu proses ke proses lainnya.
3.	Proses		Mentrasnformasikan data secara umum
4.	Berkas atau Tempat Penyimpanan		Menyimpan data atau file

(Sumber: Sukamto dan Shalahuddin, 2013:71)

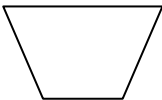
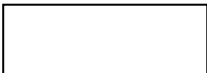
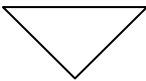
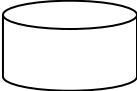
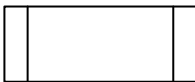
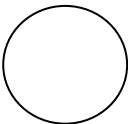
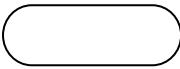
2.2.2. Block Chart

Kristanto (2011:68) mengemukakan, *block chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *block chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

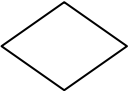


Adapun simbol-simbol *Block Chart* adalah sebagai berikut:

Tabel 2.2. Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
2.		Multi dokumen
3.		Proses manual
4.		Proses yang dilakukan oleh komputer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>data storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran

Lanjutan **Tabel 2.2.** Simbol-Simbol *Block Chart*

No.	Nama	Simbol
11.		Pengambilan keputusan (<i>decision</i>)
12.		Layar peraga (<i>monitor</i>)
13.		Pemasukan data secara manual

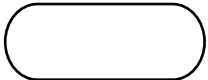
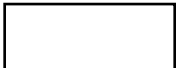
(Sumber: Kristanto, 2011:68)

2.2.3. Flowchart

Kadir (2012:16) mengemukakan, “*flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya”.

Ladjamudin (2013:211) mengemukakan, *flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma. Adapun simbol-simbol *Flowchart* adalah sebagai berikut:

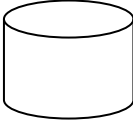
Tabel 2.3. Simbol-Simbol *Flowchart*

No.	Simbol	Arti
1.		Simbol Start atau End yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja

Lanjutan **Tabel 2.3.** Simbol-Simbol *Flowchart*

No.	Simbol	Arti
3.		Simbol Input/Output yang mendefinisikan masukan dan keluaran proses
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar symbol
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau ke sebuah pita <i>magnetic</i>

Lanjutan **Tabel 2.3.** Simbol-Simbol *Flowchart*

No.	Simbol	Arti
13.		Simbol database atau basis data

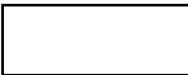
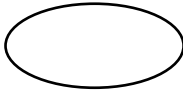
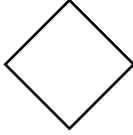
(Sumber: Kadir, 2012:17)

2.2.4. Entity Relationship Diagram (ERD)

Yakub (2012:60) mengemukakan, “*entity relationship diagram (ERD)* untuk mendokumentasikan data perusahaan dengan mengidentifikasi jenis entitas (*entity*) dan hubungannya *ERD* merupakan suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara abstrak”.

Sukanto dan Shalahuddin (2013:50) mengemukakan, *entity relationship diagram* digunakan untuk permodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan *Object Oriented Database Management System (OODBMS)* maka perancangan basis data tidak perlu menggunakan *ERD*. Adapun simbol-simbol *Entity Relationship Diagram* adalah sebagai berikut:

Tabel 2.4. Simbol-Simbol *Entity Relationship Diagram (ERD)*

No.	Nama	Simbol	Keterangan
1.	Entitas		Berupa orang, kejadian, atau benda di mana data akan dikumpulkan
2.	<i>Atribute</i>		Merupakan properti dari entitas. Nama atribut harus merupakan kata benda
3.	<i>Relationship</i>		Menunjukkan hubungan antar 2 entitas. Dideskripsikan dengan kata kerja

Lanjutan **Tabel 2.4.** Simbol-Simbol *Entity Relationship Diagram (ERD)*

No.	Nama	Simbol	Keterangan
4.	Link	—————	Sebagai penghubung antara entitas dan <i>relationship</i> serta entitas dan <i>atribute</i>

(Sumber: Yakub, 2012:60)

2.2.5. Kamus Data (*Data Dictionary*)

Sukamto dan Shalahuddin (2013:73) mengemukakan, “kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada *DFD*. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) data keluaran (*output*) dapat dipahami secara umum”.

Yakub (2012:168) mengemukakan, kamus data (*data dictionary*) merupakan daftar elemen data yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem sehingga user dan analisis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan data *storage*. Adapun simbol-simbol yang sering digunakan dalam kamus data adalah sebagai berikut:

Tabel 2.5. Simbol-Simbol Kamus Data

No.	Simbol	Arti
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik ...atau...
4.	{ } ⁿ	N kali diulang / bernilai banyak
5.	()	Data opsional
6.	*...*	Batas komentar

(Sumber : Sukamto dan Shalahuddin, 2013:74)



2.2.6. Daftar Kejadian (*Event List*)

Kristanto (2011:64) menjelaskan tentang daftar kejadian (*event list*) sebagai berikut:

Daftar kejadian digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Suatu kejadian mewakili satu aliran data atau proses dalam diagram konteks serta deskripsi penyimpanan yang digunakan untuk memodelkan data harus diperhatikan dalam kaitannya dengan daftar kejadian. Adapun cara-cara mendeskripsikan daftar kejadian adalah sebagai berikut:

- a. Pelaku adalah entiti luar, jadi bukan sistem.
- b. Menguji setiap entiti luar dan mencoba mengevaluasi setiap entiti luar yang terjadi pada sistem.
- c. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
- d. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan entiti luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
- e. Setiap aliran keluaran sebaiknya merupakan respondari kejadian.
- f. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
- g. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

2.3. Pengertian-Pengertian Judul

2.3.1. Pengertian Aplikasi

Asropudin (2013:6) mengemukakan, “aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu”.

Sutabri (2012:147) mengemukakan, “aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya.



Jadi, aplikasi adalah *software* yang dibuat oleh perusahaan komputer difungsikan secara khusus sesuai kemampuan yang dimilikinya untuk mengerjakan tugas-tugas tertentu.

2.3.2. Pengertian Penilaian

Kamus Besar Bahasa Indonesia Versi *Online* (2012-2016) mengemukakan, ~**pekerjaan** penentuan nilai dari suatu pekerjaan untuk menentukan skala gaji, syarat-syarat kenaikan pangkat, dan perangsang terhadap pekerjaan.

Lindayani (2013) mengemukakan, penilaian adalah rangkaian kegiatan untuk memperoleh, menganalisis dan menafsirkan data tentang proses dan hasil belajar peserta didik yang dilakukan secara sistematis dan berkesinambungan sehingga dapat menjadi informasi yang bermakna dalam pengambilan keputusan.

Jadi, penilaian adalah penentuan rangkaian kegiatan untuk menentukan syarat-syarat kenaikan pangkat sehingga dapat menjadi informasi yang bermakna dalam pengambilan keputusan.

2.3.3. Pengertian Jabatan Fungsional

PERKA BKN (2011:5) menyatakan, Jabatan Fungsional Analis Kepegawaian adalah Pegawai Negeri Sipil (PNS) yang diberi tugas, tanggung jawab, wewenang dan hak secara penuh oleh pejabat yang berwenang untuk melakukan kegiatan manajemen PNS dan pengembangan sistem manajemen PNS.

Peraturan Pemerintah Republik Indonesia Pasal 1(1) mengemukakan, “jabatan fungsional adalah kedudukan yang menunjukkan tugas, tanggung jawab, wewenang dan hak seseorang Pegawai Negeri Sipil dalam suatu satuan organisasi yang dalam pelaksanaan tugasnya didasarkan pada keahlian dan/atau ketrampilan tertentu serta bersifat mandiri”.

Jadi, jabatan fungsional adalah Pegawai Negeri Sipil (PNS) yang diberi tugas, wewenang dalam suatu organisasi yang dalam pelaksanaan tugasnya didasarkan pada keahlian tertentu serta bersifat mandiri.



2.3.4. Pengertian Daftar Usulan Penetapan Angka Kredit (DUPAK)

PERKA BKN (2011:5) menyatakan, Angka Kredit adalah satuan nilai dari tiap butir kegiatan dan/atau akumulasi butir-butir kegiatan yang harus dicapai oleh seorang Analis Kepegawaian dalam rangka pembinaan karier kepangkatan dan jabatannya.

Daftar Usulan Penetapan Angka Kredit (DUPAK) adalah daftar usulan yang memuat data perorangan Analis Kepegawaian yang berisi rincian butir kegiatan dengan mencantumkan nilai/angka kredit yang diperoleh dalam kurun waktu tertentu sebagai bahan penilaian dalam penetapan angka kredit.

Jadi, Daftar Usulan Penetapan Angka Kredit adalah daftar usulan yang memuat data perorangan Analis Kepegawaian yang berisi rincian butir-butir kegiatan yang harus dicapai dalam rangka karier kepangkatan dan jabatannya.

2.3.5. Aplikasi Penilaian Jabatan Fungsional dengan Daftar Usulan Penetapan Angka Kredit (DUPAK) pada Kantor Inspektorat Kabupaten Ogan Komering Ilir.

Aplikasi Penilaian Jabatan Fungsional dengan Daftar Usulan Penetapan Angka Kredit (DUPAK) pada Kantor Inspektorat Kabupaten Ogan Komering Ilir adalah suatu aplikasi yang digunakan memudahkan pegawai yang sebagai penanggungjawab dengan pengumpulan data-data pada bagian jabatan fungsional di Kantor Inspektorat Kabupaten Ogan Komering Ilir dalam mempermudah pegawai untuk mengajukan usulan penetapan jabatan dengan angka kredit dan mendapatkan data yang akurat.

2.4. Teori Program

2.4.1. Java

2.4.1.1. Pengertian Java

Rickyanto (2005:2) mengemukakan, “*java* adalah suatu teknologi di dunia *software* komputer. Selain merupakan suatu bahasa pemrograman, *java* juga merupakan suatu *platform*”.



Pemulis juga secara ringkas telah menjawab pertanyaan tersebut dalam kalimat tersebut, dan berikut ini penulis menjelaskan lebih lanjut:

Java merupakan teknologi di mana teknologi tersebut mencakup *java* sebagai bahasa pemrograman yang memiliki sintaks dan aturan pemrograman tersendiri, juga mencakup *java* sebagai *platform* di mana teknologi ini memiliki *virtual machine* dan *library* yang diperlukan untuk menulis dan menjalankan program yang ditulis dengan bahasa pemrograman.

2.4.1.2. Karakteristik Java

Rickyanto (2005:5) mengemukakan, secara singkat karakteristik *java* adalah:

a. Sederhana

Java tidak memiliki sintak yang aneh tetapi banyak menggunakan sintaks *C++* yang sudah banyak dikenal sehingga *java* tidak menuliskan bagi para *programmer*. Bahkan *java* memberikan banyak keunggulan dan kemudahan dibandingkan *C++*.

b. Berorientasi objek

Java merupakan pemrograman berorientasi objek yang murni. Dalam pemrograman *java* semua adalah objek terkecuali tipe data primitif.

c. Dapat didistribusikan dengan mudah

Sifat terdistribusi dari *java* sangat tampak sebagai *applet* dan *library* yang mampu berkerja dalam jaringan dan bekerja dengan objek terdistribusi dengan sangat baik. Saat ini *java* juga memiliki kemampuan untuk *server side processing*, yaitu teknologi *Java Servlet* dan *Java Server Pages*.

d. Aman

Aman karena program *java* memiliki *library security* setara *polity* yang membatasi akses *applet* di komputer *client*.

e. Diinterpretasi oleh interpreter



2.4.1.3. Graphical User Interface (GUI)

Kuniawan (2014:39) mengemukakan, beberapa contoh ilustrasi *library GUI* juga diperkenalkan guna membangun aplikasi *Java* berbasis *GUI*.

a. Swing Atau AWT ?

Kelas untuk *GUI* sudah disediakan dalam sebuah *library* yang dikenal dengan *Abstrak Windows Toolkit (AWT)*. *Library AWT* Cukup baik untuk tampilan sederhana, tetapi tidak cocok dalam pengembangan *GUI*. Hal ini terkait dengan *AWT*, merupakan *library* dengan *platform-oriented*, yang sangat spesifik terhadap *platform* sistem operasi yang digunakan. *Library AWT* terletak pada paket komponen **java.awt**.

Java swing merupakan bagian dari *JFC (Java Foundation Classes)* yang menyediakan API untuk menangani hal yang berhubungan dengan *GUI* bagi program *java*.

Pada umumnya kelas-kelas yang berada di dalam komponen *Swing* diawali dengan huruf J, misalkan: *JButton*, *JLabel*, *JTextField*, *JRadioButton*.

b. Frame

Pada *java* sebuah *frame* dapat diwakili oleh sebuah kelas, yaitu *JFrame*. Beberapa method yang penting dan sering digunakan.

Tabel 2.6. Beberapa method pada kelas *JFrame*.

Nama Method	Keterangan
setSize()	Menentukan ukuran frame
setLocation()	Menentukan lokasi frame pada bagian pojok kiri atas
setVisible()	Menampilkan frame
setDefaultCloseOperation()	Menentukan operasi ketika frame ditutup
setLocationRelativeTo()	Menentukan lokasi frame relative terhadap parameter komponen yang diinputkan
Pact()	Menentukan secara otomatis ukuran frame sesuai komponen yang dimasukkan

(Sumber : Rickyanto, 2005:39)



2.4.2. Basis Data (*Database*)

Sujatmiko (2012:76) mengemukakan, “*database* (basis data) adalah representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan”.

Menurut Sukamto dan Shalahuddin (2013:43) “basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat”.

Yakub (2012:51-52) mengemukakan, “basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi)”.

Jadi, basis data (*database*) adalah media untuk pengumpulan fakta atau data yang saling berhubungan (punya relasi) agar dapat diakses dengan mudah dan cepat.

2.4.3. MySQL

2.4.3.1. Pengertian MySQL

Sibero (2013:97) mengemukakan, “MySQL atau dibaca “My Sekuel” dengan (sic!) adalah suatu *RDBMS* (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data”.

Hidayatullah dan Kawistara (2014:180) mengemukakan, “MySQL adalah salah satu aplikasi *DBMS* yang sudah sangat banyak digunakan oleh pemrogram aplikasi *web*”.

Nugroho (2014:31) mengemukakan, “MySQL adalah *software* atau program aplikasi *database*, yaitu *software* yang dapat kita pakai untuk menyimpan data berupa informasi teks dan juga angka”.

Jadi, MySQL adalah aplikasi *database* yang digunakan untuk proses pengolahan dan penyimpanan data berupa teks dan juga angka.



2.4.3.2. Fungsi-Fungsi MySQL

Tabel 2.7. Fungsi-Fungsi MySQL

No	Nama Fungsi	Kegunaan
1.	mysql_affected_rows()	Mengambil jumlah <i>record</i> hasil eksekusi
2.	mysql_close()	Menutup koneksi <i>database</i>
3.	mysql_connect()	Membuka koneksi <i>database</i>
4.	mysql_create_db()	Membuat <i>database</i> baru
5.	mysql_db_name()	Mengambil nama <i>database</i> hasil dari mysql_list_dbs()
6.	mysql_drop_db()	Menghapus suatu <i>database</i>
7.	mysql_errno()	Mengambil nomor <i>index</i> pesan kesalahan
8.	mysql_error()	Mengambil pesan kesalahan
9.	mysql_fetch_array()	Mengambil hasil <i>record</i> dan mendefinisikan dalam bentuk <i>array</i>
10.	mysql_fetch_assoc()	Mengambil hasil <i>record</i> seperti mysql_fetch_array dengan tipe MYSQL_ASSOC
11.	mysql_free_result()	Mengosongkan memori hasil proses
12.	mysql_insert_id()	Mengambil nomor ID terakhir setelah eksekusi. Fungsi ini berlaku untuk tabel yang memiliki kolom ID dengan tipe AUTO_INCREMENT
13.	mysql_list_dbs()	Mengambil daftar <i>database</i> aktif pada <i>server</i>
14.	mysql_num_rows()	Mengambil jumlah baris data hasil <i>query</i>
15.	mysql_pconnect()	Membuka koneksi persisten <i>database</i>
16.	mysql_ping()	Menguji respon <i>server</i> atau membuka kembali koneksi <i>database</i> yang terputus
17.	mysql_query()	Mengirimkan perintah <i>query</i> pada <i>server</i>
18.	mysql_real_escape_string()	Mengubah karakter tanda baca menjadi bentuk yang aman pada <i>database</i>

Lanjutan **Tabel 2.7.** Fungsi-Fungsi *MySQL*

No	Nama Fungsi	Kegunaan
19.	<code>mysql_result()</code>	Mengambil data hasil
20.	<code>mysql_select_db()</code>	Memilih dan mengaktifkan <i>database</i>
21.	<code>mysql_unbuffered_query()</code>	Mengirimkan perintah pada <i>server</i> dan hasilnya tidak disimpan dalam <i>buffer memory</i>

(Sumber: Sibero, 2013:71-79)

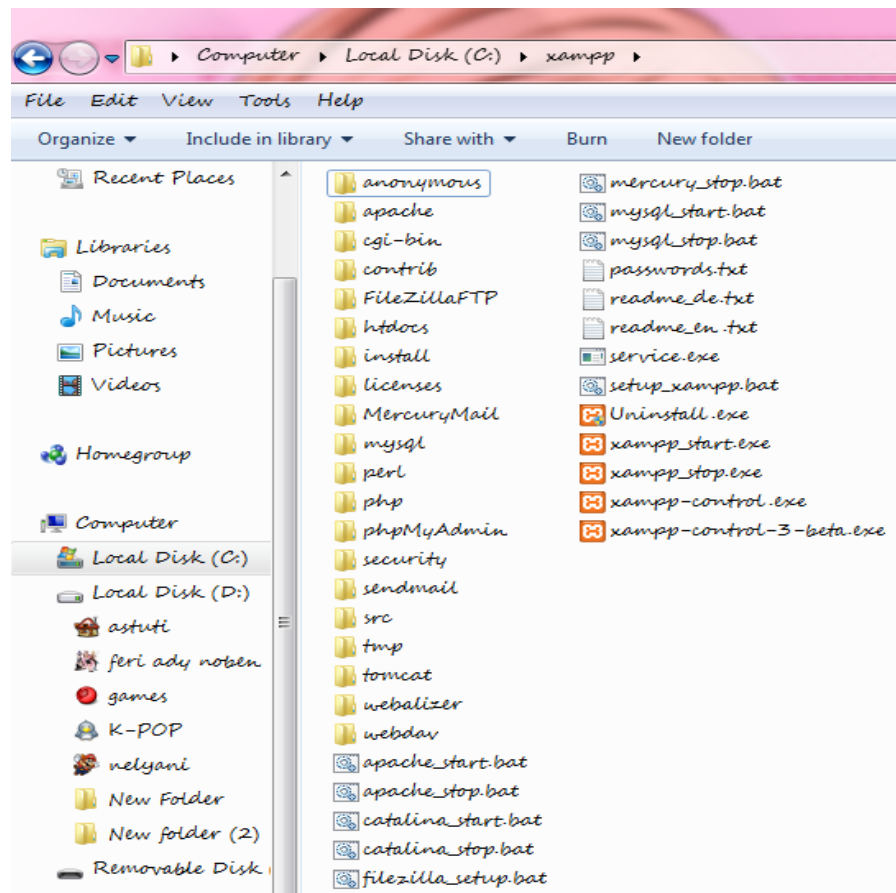
2.4.4. XAMPP

2.4.4.1. Pengertian XAMPP

Menurut Sidik (2014:72) “XAMPP (*X(Windows/Linux) Apache MySQL PHP dan Perl*) merupakan paket *server web PHP* dan *database MySQL* yang paling populer dikalangan pengembang *web* dengan menggunakan *PHP* dan *MySQL* sebagai *database-nya*”.

2.4.4.2. Memahami Folder XAMPP

Nugroho (2013:6) menyatakan bahwa program XAMPP yang telah diinstalasi memiliki folder standar yang terletak pada **C:\xampp**. Susunan folder hasil instalasi dapat dilihat pada gambar di bawah ini.



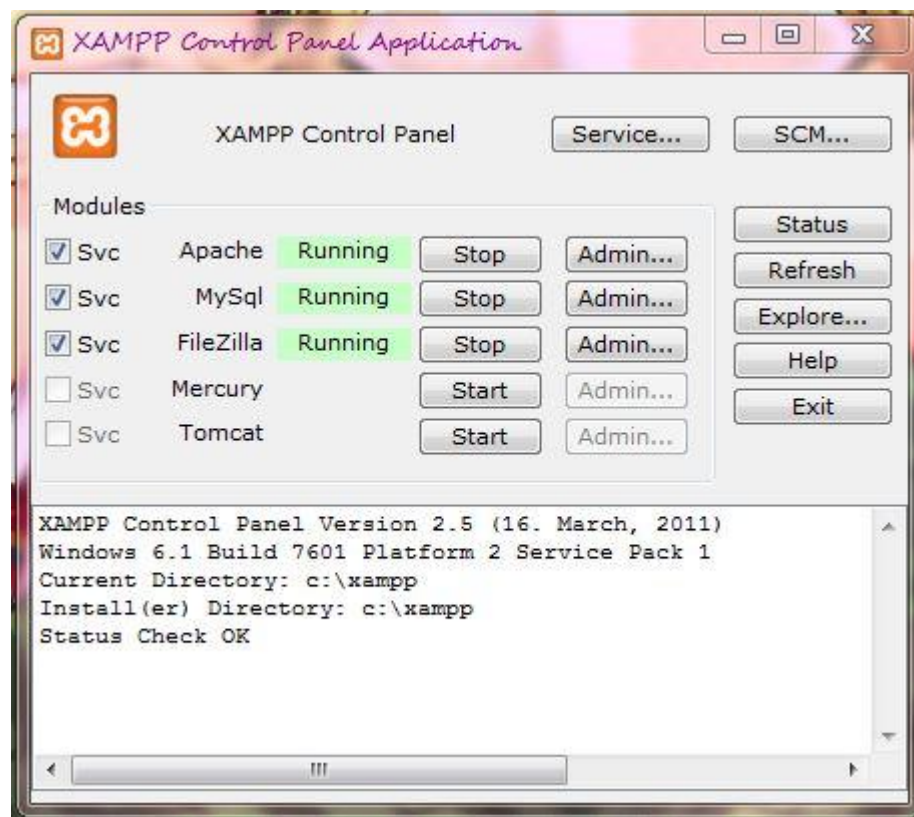
Gambar 2.1. Susunan Folder XAMPP di drive C:\

Di bawah *folder* utama **xampp**, terdapat beberapa *folder* penting. Berikut ini adalah penjelasan mengenai *folder-folder* penting tersebut.

- a. *Apache*, merupakan *folder* utama dari *Apache Webserver*.
- b. *Htdocs*, merupakan *folder* utama untuk menyimpan data-data latihan *web*, baik *PHP* maupun *HTML* biasa. Semua *folder* dan *file* program di *htdocs* bisa diakses dengan mengetikkan alamat `http://localhost/` di *browser*.
- c. *Mysql*, merupakan *folder* utama untuk *database MySQL server*.
- d. *Php*, merupakan *folder* utama untuk program *PHP*.



Untuk mengakses *XAMPP* yang telah di-instalasi, dapat dilakukan dengan mengklik tombol **Start** -> **All Programs** -> **Apache Friends** -> **XAMPP** -> **XAMPP Control Panel**. Adapun tampilan dari *XAMPP Control Panel* adalah sebagai berikut:



Gambar 2.2. Tampilan *XAMPP Control Panel*

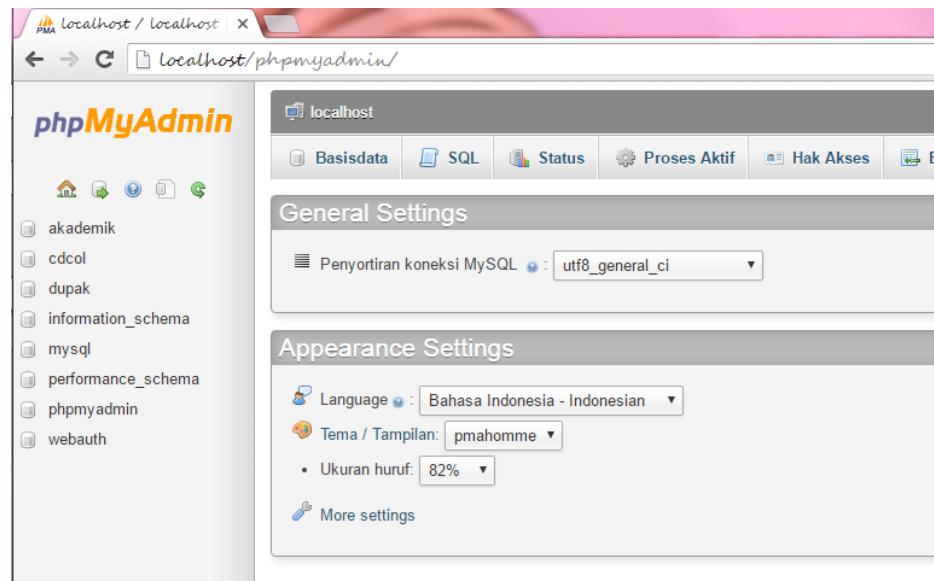
2.4.5. *PhpMyAdmin*

Nugroho (2013:15) mengemukakan bahwa “*phpMyAdmin* adalah aplikasi manajemen database server MySQL berbasis web”. Aplikasi *phpMyAdmin* memungkinkan programmer untuk mengelola database sebagai root (pemilik server) atau user biasa, membuat database baru, dan melakukan operasi perintah-perintah database secara lengkap. Berikut ini cara mengakses *phpMyAdmin* di browser.

1. Buka browser.
2. Pada kotak alamat, ketikkan `http://localhost/phpmyadmin/` (tulis dalam huruf kecil semua).



3. Lalu tekan **Enter**.



Gambar 2.3. Tampilan *phpMyAdmin*