



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Paramytha (2016:4), “Komputer berasal dari Bahasa latin yaitu *computer* yang berarti menghitung (*to computer to rection*). Kata komputer itu sendiri pada awalnya dipergunakan untuk menggambarkan pekerjaan orang yang melakukan perhitungan aritmatika dengan atau tanpa alat bantu.”

Kadir (2017:2), “Komputer adalah peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia.”

Dari beberapa definisi diatas penulis menyimpulkan bahwa komputer adalah alat elektronik yang dipergunakan untuk menggambarkan pekerjaan yang dilakukan oleh manusia.

2.1.2. Pengertian Basis Data

Sukamto (2018:43), “Basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.”

Kristanto (2018: 79) menyatakan “basis data adalah kumpulan data, yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi.”

Dari kedua pendapat diatas dapat disimpulkan bahwa basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis agar dapat diakses dengan mudah dan cepat.

2.1.3. Pengertian Perangkat Lunak

Kadir (2017:2), “Perangkat Lunak adalah instruksi-instruksi yang ditujukan kepada computer agar dapat melaksanakan tugas sesuai kehendak pemakai.”



Sukamto (2018:2), “Perangkat Lunak (Software) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan.”

Dari beberapa definisi diatas penulis menyimpulkan bahwa perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak agar dapat melaksanakan tugas sesuai kehendak pemakai.

2.1.4 Metode Pengembangan Sistem

Sukamto (2018:28), 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 terurut mulai dari analisis, desain, pengkodean, pengujian dan pemeliharaan.

Bentuk ini adalah bagian-bagian dalam pengembangan metode *waterfall* :

a. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean.

c. Pembuatan Kode Barang

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk



meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. Dengan demikian, pengujian metode ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program.

e. Pendukung (*support*) dan pemeliharaan (*maintenance*)

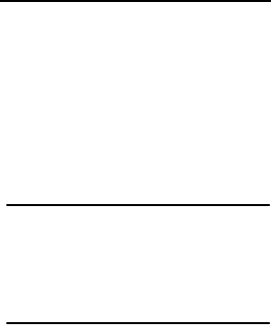
Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah mengirimkan ke user. Perubahan bisa terjadi adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

2.1 Teori Khusus

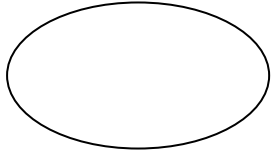

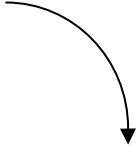
2.2.1 Pengertian DFD (*Data Flow Diagram*)

Sukanto (2018:70), “Data Flow Diagram (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (input) dan keluaran (output).”

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

| NO | Simbol | Keterangan |
|----|---|---|
| 1. |  | File atau basis data atau penyimpanan (storage); Pada pemodelan perangkat lunak yang akan Di implementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, table ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (Entity Relationship Diagram, Conceptual Data Model, Physical Data Model) |



| | | |
|---|--|---|
| 2 |  | Proses atau fungsi atau prosedur; pada pemodelan Perangkat lunak yang akan di implementasikan dengan Pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. |
| 3 |  | Entitas luar (external entity) atau masukan (input) atau keluaran (output) atau orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan. |
| 4 |  | Aliran data yang dikirim antar proses, dari penyimpanan. ke proses, atau dari proses ke masukan atau keluaran. |

(Sumber : Sukamto, 2018:71-72)

Sukamto (2018:72-73), menjelaskan, Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 (nol) atau sering disebut juga Context Diagram

DFD Level 0 (nol) menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 (nol) 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 (nol) yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-breakdown menjadi DFD Level



2(dua) Modul mana saja yang harus di-breakdown lebih lebih detail tergantung pada tingkat kedetailan 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 ada DFD Level 1(satu) yang di-breakdown.

4. Membuat DFD Level 3 dan seterusnya



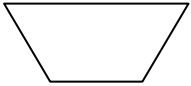
DFD Level 3,4,5 dan seterusnya merupakan breakdown dari modul pada DFD di atasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.2.2. Pengertian *Block chart*


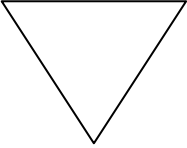

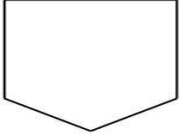
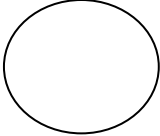

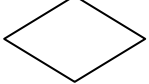
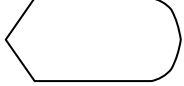
Kristanto (2008:68-70), menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.” Pembuatan blockhart harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Adapun simbol-simbol yang sering digunakan dalam block chart dapat dilihat pada tabel berikut ini:


Tabel 2.2 Simbol-simbol *Block chart*

| Simbol | Arti |
|---|--|
|  | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/berkas atau cetakan. |
|  | Multi Dokumen |
|  | Proses Manual |



| | |
|---|---|
|  | Proses yang dilakukan oleh komputer |
|  | Menandakan dokumen yang diarsipkan (arsip manual) |
|  | Data penyimpanan (<i>data storage</i>) |
|  | Terminasi yang mewakili simbol tertentu untuk Digunakan pada aliran lain pada halaman |
|  | Terminasi Yang mewakili simbol tertentu untuk Digunakan pada aliran lain pada halaman sama. |
|  | Terminasi yang menandai awal dan akhir Dari suatu aliran |
|  | Pengambil Keputusan |
|  | Layar Peraga (monitor) |



| | |
|---|------------------------------|
|  | Pemasukan data secara manual |
|---|------------------------------|


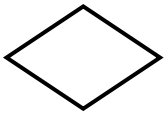
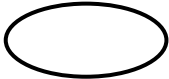
(Sumber : Kristanto, 2008:75-77)

2.2.3. Pengertian ERD (*Entity Relational Diagram*)

Yakub (2012: 60), menyatakan “*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 abstrak.”

Yakub (2012: 60), menyatakan berikut adalah simbol-simbol yang digunakan dalam entity relationship diagram (*ERD*), yaitu:

Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

| No. | Simbol | Keterangan |
|-----|---|---|
| 1. |  | Entitas, yaitu kumpulan dari objek yang dapat diidentifikasi secara unik |
| 2. |  | Relasi, yaitu hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain; satu ke satu, satu ke banyak, dan banyak ke banyak. |
| 3. |  | Atribut, yaitu karakteristik dari entity atau relasi yang merupakan penjelasan detail tentang entitas |

(Sumber: Yakub , 2012:60)



2.2.4. Pengertian *Flowchart*

Ladjamudin (20013:265), menyatakan “*flowchart* adalah bagian yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma.”

2.2.4.1. Simbol-simbol Pada *Flowchart*

Ladjamudin (2013: 266—268), menyatakan bahwa simbol yang digunakan dalam *flowchart* dibagi menjadi 3 (tiga) kelompok, yaitu:

1. *Flow Direction Symbols* (Simbol penghubung/ alur)

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan yang lain atau disebut juga *connecting line*, yaitu sebagai berikut:

Tabel 2.4 Tabel *Flow Direction Symbols* (Simbol Penghubung/ alur)

| | | |
|----|--|---|
| 1. | | <p>Simbol arus/ <i>flow</i></p> <p>Untuk menyatakan jalannya arus suatu proses</p> |
| 2. | | <p>Simbol <i>Communication link</i></p> <p>Untuk menyatakan bahwa adanya transisi suatu data/ informasi dari satu lokasi ke lokasi lainnya</p> |
| 3. | | <p>Simbol <i>Connector</i></p> <p>Untuk menyatakan sambungan dari satu proses ke proses yang lainnya dalam halaman/ lembaran yang sama</p> |
| 4. | | <p>Simbol <i>Offline Connector</i></p> <p>Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang berbeda</p> |



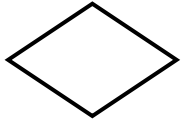
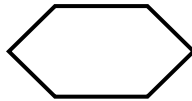


(Sumber: Ladjamudin 2013: 266—268)



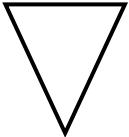

2. Processing Symbols (Simbol Proses)

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses/prosedur, berikut adalah simbol-simbolnya:

2.5 Tabel Processing Symbols (Simbol Proses)

| | | |
|----|---|--|
| 1. |  | Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses yang lainnya ke dalam halaman/lembar yang berbeda |
| 2. |  | Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual) |
| 3. |  | Simbol Decision/ logika Untuk menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/ tidak |
| 4. |  | Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga |
| 5. |  | Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program |
| 6. |  | Simbol Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard |



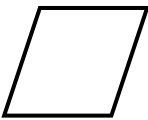
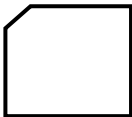
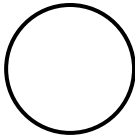
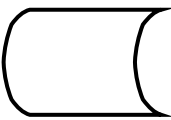

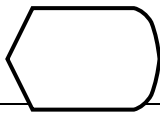
| | | |
|----|---|---|
| 7. |  | Simbol Off-line Storage Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu |
| 8. |  | Simbol Manual Input Untuk memasukkan data secara manual dengan menggunakan online keyboard. |

(Sumber: Ladjamudin 2013: 266—268)

3. Input-output Symbols (Simbol input-output)

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media input atau output, yaitu sebagai berikut:

2.6 Tabel *Input-output Symbols* (Simbol input-output)

| | | |
|----|---|---|
| 1. |  | Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya. |
| 2. |  | Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu |
| 3. |  | Simbol Magnetic-tape Unit Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic |
| 4. |  | Simbol Disk Storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk |
| 5. |  | Simbol Document Untuk mencetak laporan ke printer |
| 6. |  | Simbol Display Untuk menyatakan peralatan output yang |



| | | |
|--|--|--|
| | | digunakan berupa layar (video, komputer) |
|--|--|--|

(Sumber: Ladjamudin 2013: 266—268)

2.2.5. Pengertian Kamus Data

Sukamto (2018:73) menjelaskan, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*ouput*) dapat dipahami secara umum (memiliki standar cara penulisan).”

Kamus data memiliki beberapa simbol sebagai berikut :

Tabel 2.7 Simbol-simbol Kamus Data

| Simbol | Keterangan |
|--------|--------------------------------|
| = | Disusun atau terdiri dari |
| + | Dan |
| [] | Baik...atau... |
| { }n | N kali diulang/bernilai banyak |
| () | Data opsional |
| *...* | Batas komentar |

(Sumber: Shalahuddin 2013:74)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Indrajani (2018:3), “Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk menyelesaikan tugas-tugas khusus dari pemakai komputer.”

Paramytha (2016:56), “Aplikasi (application software) merupakan perangkat lunak yang dikembangkan untuk menyelesaikan sesuatu aplikasi tertentu.”



Kesimpulannya, Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi untuk menyelesaikan tugas-tugas khusus dari pemakai komputer.”

2.3.2 Pengertian Data

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

Setyaningrum (2013:1), “Data adalah catatan atas sekumpulan fakta yang belum mempunyai arti bagi penerimanya dan masih memerlukan suatu pengolahan.”

Kesimpulannya, Data adalah Sekumpulan angka-angka maupun karakter-karakter yang belum memiliki arti dan masih memerlukan suatu pengolahan.

2.3.3 Pengertian Proses

Menurut Kamus Besar Bahasa Indonesia (<https://kbbi.kemdikbud.go.id/entri/proses>, diunduh pada 22 Mei 2019, pukul 18.55 WIB) menyatakan bahwa Proses adalah runtunan perubahan peristiwa dalam perkembangan sesuatu rangkaian tindakan, pembuatan atau pengolahan yang menghasilkan produk.

2.3.4 Pengertian Distribusi

Menurut Kamus Besar Bahasa Indonesia (<https://kbbi.kemdikbud.go.id/entri/distribusi>, diunduh pada 22 Mei 2019, pukul 19.03 WIB) menyatakan bahwa “Distribusi adalah penyaluran, pembagian dan pengiriman kepada beberapa orang atau ke beberapa tempat.”



2.3.5 Pengertian Barang

Menurut Kamus Besar Bahasa Indonesia (<https://kbbi.kemdikbud.go.id/entri/barang>, diunduh pada 22 Mei 2019, pukul 19.08 WIB), menyatakan bahwa “Barang adalah benda umum segala sesuatu yang berwujud dan berjasad”.

2.3.6 Pengertian Aplikasi Pendataan dan Proses Pendistribusian Barang Berbasis *Web* pada Dinas Perumahan Rakyat, Kawasan Permukiman dan Pertanahan Kabupaten Banyuasin

Aplikasi Pendataan dan Proses Pendistribusian Barang Berbasis *Web* menggunakan *PHP* adalah suatu aplikasi untuk Dinas Perumahan Rakyat, Kawasan Permukiman dan Pertanahan Kabupaten Banyuasin agar bisa mengontrol proses barang masuk dan pendistribusiannya untuk lebih efektif dan menghemat kertas serta apabila ada nya pengecekan barang maka pegawai akan lebih mudah mencarinya di database.

2.4 Teori Program

2.4.1. Pengertian XAMPP

Palit (2015:2), “*XAMPP* adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.”

Nugroho (2013:1), “*Xampp* adalah program web lengkap yang dapat Anda pakai untuk belajar pemrograman web, khususnya *PHP* dan *MySQL*, paket ini dapat didownload secara gratis dan legal.”

Kesimpulannya, *XAMPP* adalah suatu program server yang terdiri dari *Apache (server)*, *MySQL (database)*, dan *PHP* digunakan untuk mengakses fungsi yang ada dalam halaman website tersebut agar bisa diakses oleh *user*.

2.4.2. Pengertian *MySQL*

Shalahuddin (2018:46), “*SQL (Structured Query Language) SQL* adalah bahasa yang digunakan untuk mengelola data pada *RDBMS*. *SQL* awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”



Nugroho (2013:26), “*MySQL* itu adalah *software* atau program *Database Server*, dia itu *software* atau program-nya. Sedangkan adalah Bahasa pemrogramannya, dia itu bahasa permintaan (*query*) dalam *database server*, termasuk dalam *MySQL* itu sendiri.”

Kesimpulannya, *MySQL* adalah bahasa yang digunakan untuk mengelola data pada *RDBMS* bersifat *open source* sehingga banyak digunakan di dunia.

2.4.3. Pengertian PHP

Nugroho (2013:153), “*PHP* itu adalah bahasa program yang digunakan untuk membuat aplikasi berbasis *web* (*website*, *blog*, atau aplikasi *web*).”

Raharjo (2016:38), “*PHP* adalah salah satu bahasa pemrograman pemrograman skrip yang dirancang untuk membangun aplikasi *web*.”

Kesimpulannya, *PHP* adalah bahasa pemrograman berbasis *server-side* yang bisa kita gunakan untuk membuat aplikasi web yang disisipkan pada *HTML*, yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi dekstop.



Gambar 2.1 Tampilan Logo PHP

2.4.3.1. Sintaks Dasar PHP

Kode (*Script*) *PHP* yang sering disebut dengan istilah *embedded script* yaitu script *PHP* yang disisipkan di antara script *HTML*. Jadi dapat dikatakan script *PHP* hanya ditulis atau disisipkan ketika dibutuhkan saja, seperti menampilkan data dari database meng-upload *file*, *delete data*, *edit data* dan lain sebagainya.

Contoh script :

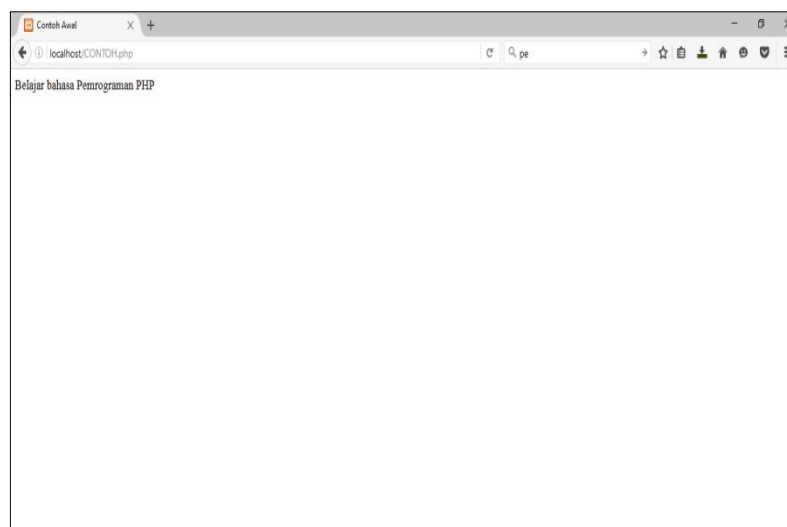
```
<HTML>
  <HEAD>
    <TITLE>Contoh Awal</TITLE>
```



```

</HEAD>
<BODY>
    <?php
        echo "Belajar bahasa Pemrograman PHP";
    ?>
</BODY>
</HTML>

```



Gambar 2.2 Contoh Script PHP

2.4.3.2. Tipe Data PHP

Tipe data *PHP* digunakan untuk menentukan jenis data yang akan disimpan dalam suatu variabel. Risnandar,dkk (2013:60-61), menjelaskan PHP mempunyai empat tipe data dasar sebagai berikut :

1. **Integer** merupakan tipe data yang mencakup semua bilangan bulat. Range bilangan integer adalah antara -2.147.4833.647 sampai dengan 2.147.483.647
2. **Floating point** merupakan tipe data yang mencakup semua bilangan desimal (bilangan yang memiliki angka dibelakang koma). Range bilangan floating point antara 1e308 sampai dengan 1e308.
3. **Character** merupakan tipe yang digunakan untuk menyimpan data-data yang berupa karakter (satu huruf). Penulisannya biasanya diapit dengan tanda kutip satu ('...').



4. ***String*** merupakan tipe data tersendiri dan tidak dapat dikelompokkan menjadi tipe data dasar. Penulisannya biasa diapit dengan tanda kutip dua (“...”.)”