



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

Menurut Irwansyah (2014:2), “Komputer merupakan perangkat elektronik yang beroperasi dibawah kendali instruksi yang memiliki memori, dapat menerima dan mengolah data sesuai dengan aturan yang ditetapkan, menghasilkan hasil atau informasi dan menyimpan hasil tersebut untuk digunakan di masa depan atau di kemudian hari”. Senada dengan pendapat diatas, menurut Mulyono (2010:1), “Komputer adalah seperangkat alat elektronik yang terdiri atas peralatan *input*, dan peralatan *output* yang memberikan informasi, serta bekerja secara otomatis”. Menurut Puspitosari (2013:1), “Komputer adalah sebuah alat yang digunakan untuk mengolah data menurut perintah yang telah dirumuskan”.

Maka dapat disimpulkan bahwa komputer adalah alat atau mesin yang mengolah data menjadi informasi yang berguna dan dapat dipahami oleh manusia (*user*).

##### 2.1.2 Pengertian Perangkat Lunak

Menurut Sukamto dan Shalahuddin (2016:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*). Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat disebut perangkat lunak (*software*)”. Kadir (2014:71), “Perangkat Lunak (*software*) atau program yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data”.

Dari keterangan diatas dapat disimpulkan bahwa perangkat lunak (*software*) merupakan program komputer yang terhubung dengan dokumentasi perangkat lunak yang bertugas untuk mencapai tujuan dari si pengguna.



### 2.1.3 Pengertian Android

Menurut Safaat (2015:1), “Android adalah sebuah sistem operasi untuk perangkat lunak berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel / smartphone”. Senada dengan pendapat diatas, menurut Juhara (2016:1), “Android adalah sistem operasi berbasis Linux yang dimodifikasi untuk perangkat bergerak (mobile devices) yang terdiri dari sistem operasi, *middleware* dan aplikasi-aplikasi utama”.

Menurut dua pendapat yang senada diatas, maka penulis simpulkan bahwa Android merupakan sebuah sistem operasi untuk perangkat lunak berbasis linux yang dimodifikasi untuk perangkat bergerak yang mencakup sistem operasi, *middleware* dan aplikasi.

### 2.1.4 Metode Pengembangan Sistem

Menurut Sukamto dan Shalahuddin (2016:28 – 30) Model SDLC air terjun (*waterfall*) sering disebut model liner (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar model air terjun:

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan intensif untuk mespesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini menstranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke

---



representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pembuatan kode program

Desain harus ditranslasikan ke dalam program sistem. 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 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 perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk perangkat lunak baru.

## 2.2 Teori Design

### 2.2.1 Pengertian DFD (*Data Flow Diagram*)

Menurut Sutabri (2012:116–117), “Pendekatan analisis terstruktur diperkenalkan oleh DeMarco (1978) dan Gane Sarson (1979) melalui buku metodologi struktur analisis dan desain sistem informasi. Mereka menyarankan untuk menggunakan *data flow diagram* (DFD) dalam menggambarkan atau membuat model sistem. Meskipun namanya *data flow diagram* yang seakan-akan mencerminkan penekanan pada data namun sebenarnya DFD lebih menekankan pada segi proses. Adapaun pengertian secara umum dari *data flow diagram* ini adalah suatu *network* yang menggambarkan suatu sistem automat/komputerisasi, manualisasi atau gabungan dari keduanya yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan

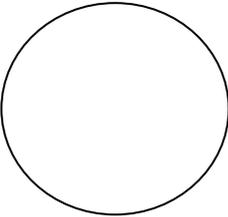


aturan mainnya. Keuntungan penggunaan DFD adalah memungkinkan untuk menggambarkan sistem dari level yang paling tinggi kemudian menguraikannya menjadi level yang lebih rendah (dekomposisi). Sedangkan kekurangan penggunaan DFD adalah tidak menunjukkan proses pengulangan (*looping*), proses keputusan dan proses perhitungan”.

Menurut Sukanto dan Salahuddin (2016:70-72), “DFD (*Data Flow Diagram*) digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi – fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi – bagi bagiannya dengan fungsi-fungsi dan prosedur – prosedur”.

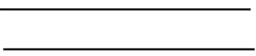
Notasi – notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

**Tabel 2.1** Notasi pada DFD (Edward Yourdon dan Tom DeMarco)

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.</p> <p>Catatan : Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>



**Lanjutan Tabel 2.1** Notasi pada DFD (Edward Yourdon dan Tom DeMarco)

Notasi	Keterangan
	<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi table basis-basis data yang dibutuhkan, table-table ini juga harus sesuai dengan perancangan table-table pada basis data (<i>Entity Relationship Diagram (ERD)</i>, <i>Conceptual Data Model (CDM)</i>, <i>Physical Data Model (PDM)</i>).</p> <p>Catatan : Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>
	<p>Entitas Luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai / berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>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>).</p>

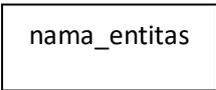
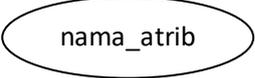
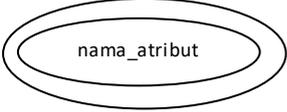
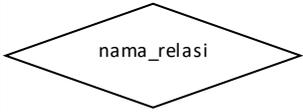
### 2.2.2 Pengertian ERD (*Entity Relationship Diagram*)

Menurut Setyaningrum (2013:17), “*Entity Relationship Diagram (ERD)* merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpanan”. Menurut Sukamto dan Salahuddin (2016:50 – 51), “ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka



perancangan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Elis), notasi Crow's Foot dan beberapa notasi lain. Namun yang banyak digunakan adalah simbol – simbol yang digunakan pada ERD dengan notasi Chen :

**Tabel 2.2** Simbol-Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Keterangan
1.	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.
2.	Atribut  	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer  	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
4.	Atribut multivalai / <i>multivalue</i>  	<i>Field</i> atau kolom data butuh disimpan dalam satu entitas yang dapat dimiliki nilai lebih dari satu.
5.	Relasi  	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.



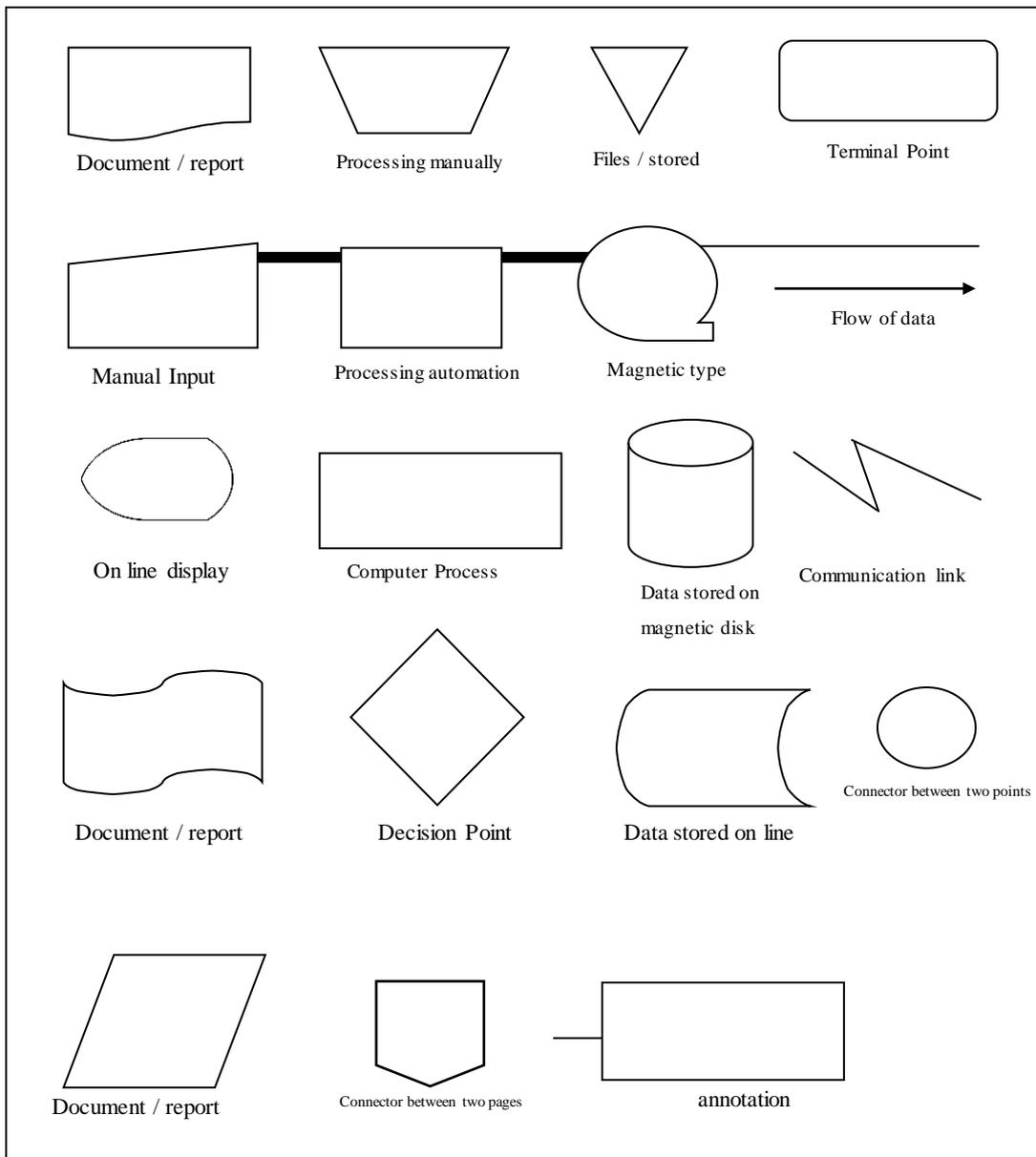
**Lanjutan Tabel 2.2** Simbol-Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Keterangan
6.	Asosiasi / <i>association</i>  	<p>Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.</p> <p>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</p>



### 2.2.3 Pengertian *Flowchart*

Menurut Indrajani (2015:36-37), “*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah, khususnya yang perlu dipelajari dan dievaluasi lebih lanjut”.

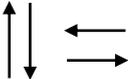
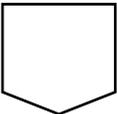
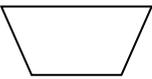


**Gambar 2.1** Notasi System Flowchart



Menurut Sitorus (2015:14), “*Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu.” Ladjamudin (2013:263 – 268) mengemukakan bahwa, “*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu masalah. Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan sistem yang lain. Simbol ini disebut juga *connecting line*. simbol – simbol tersebut adalah sebagai berikut”.

**Tabel 2.3** Simbol-Simbol dalam *Flowchart*

No	Simbol	Keterangan
1.		Simbol arus/flow Untuk menyatakan jalannya arus suatu proses
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.
4.		Simbol Offline Connector Untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
5.		Simbol Offline Connector Untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
6.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).

Lanjutan Tabel 2.3 Simbol-Simbol dalam *Flowchart*

No	Simbol	Keterangan
7.		Simbol Decision/logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya / tidak.
8.		Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
9.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program
10.		Simbol Keying Operating Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
11.		Simbol off-line storage Untuk menunjukkan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu
12.		Simbol Manual input Untuk memasukkan data secara manual dengan menggunakan online keyboard.
13.		Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
14.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu.
15.		Simbol Magnetic-tape unit Untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya
16.		Simbol Disk storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
17.		Simbol Document Untuk mencetak laporan ke printer.



### Lanjutan Tabel 2.3 Simbol-Simbol dalam *Flowchart*

No	Simbol	Keterangan
18.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

#### 2.2.4 Pengertian *Blockchart*

Menurut (Arifin:2018), “*Blockchart* berfungsi memodelkan masukan, keluaran, refrensi, master, proses ataupun transaksi dalam simbol-simbol tertentu. Pada dasarnya tidak berorientasi pada fungsi, waktu ataupun aliran data tetapi lebih ke arah proses (saling melengkapi dengan PS). Simbol-simbol yang digunakan dalam Block Chart relatif umum digunakan dalam banyak sistem”.

#### 2.2.5 Metode Pengujian Perangkat Lunak

##### 2.2.5.1 Pengertian Metode Pengujian

Sukamto dan Shalahuddin (2016:272) menyatakan, “Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain kasus uji yang spesifik dan metode pengujian”.

##### 2.2.5.2 Metode Pengujian

Secara umum pola pengujian perangkat lunak adalah sebagai berikut:

1. Pengujian dimulai dari level komponen hingga integrasi antar komponen menjadi sebuah sistem.
2. Teknik pengujian berbeda-beda sesuai dengan berbagai isi atau unit uji dalam waktu yang berbeda-beda pula bergantung pada pengujian pada bagian mana yang dibutuhkan.



3. Pengujian dilakukan oleh pengembang perangkat lunak, dan jika untuk proyek besar, pengujian bisa dilakukan oleh tim uji yang tidak terkait dengan tim pengembang perangkat lunak (*independent test group* (ITG)).
4. Pengujian dan penirkutuan (*debugging*) merupakan aktivitas yang berbeda tetapi penirkutuan (*debugging*) harus diakomodasikan pada berbagai strategi pengujian.

### 2.2.5.3 Black-Box Testing (Pengujian Kotak Hitam)

Sukamto dan Shalahuddin (2016:275), “*Black-box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan”.

Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji coba yang dilakukan adalah:

- a. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
- b. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalkan nama pemakai benar tapi kata sandi salah, atau sebaliknya atau keduanya salah.

## 2.3 Teori Judul

### 2.3.1 Pengertian Aplikasi

Budiharto (2013:5) menyatakan bahwa “*Applications* merupakan program yang dapat berjalan di komputer tersendiri (*stand alone computer*), dari mulai program yang simpel sampai dengan program besar dan rumit”. Sedangkan Mulyono (2010:109) mengemukakan bahwa, “Perangkat lunak yang khusus ditambahkan dalam sistem operasi yang ada untuk melakukan pekerjaan khusus sesuai dengan kebutuhan pengguna komputernya disebut *software* aplikasi”. Menurut Puspitosari (2013:11), “Program Aplikasi atau biasa disebut dengan

---



Aplikasi merupakan sebuah program yang berjalan pada sistem operasi dan dibuat untuk membantu user mengerjakan sesuatu untuk meningkatkan produktivitasnya”.

Dari keterangan diatas dapat disimpulkan bahwa aplikasi merupakan software pengolahan data yang yang bertugas sebagai media interaksi antara *user* dan *software* yang digunakan.

### **2.3.2 Pengertian Travel**

Menurut (Ransel:2018), “Travel adalah sebuah perjalanan seseorang dari suatu tempat ke tempat lain dengan didalamnya mengandung unsur strategi, tantangan, pengetahuan serta tujuan dan dapat ditempuh dalam kurun waktu yang cukup lama mulai dari seminggu atau bahkan bertahun-tahun yang dilakukan oleh berbagai kalangan dari pemuda bahkan orang yang sudah berumur tua”.

### **2.3.3 Pengertian Pemesanan (*Booking*)**

Menurut Kamus Besar Bahasa Indonesia (2008:64), “Pemesanan (*Booking*) merupakan proses, cara, perbuatan memesan atau memesankan”.

### **2.3.4 Pengertian Aplikasi Travel *Booking* dan Rekapitulasi berbasis Web *Mobile* pada Travel Dery Wisata**

Aplikasi Travel *Booking* berbasis Web *Mobile* pada CV. Dery merupakan aplikasi yang dibangun untuk mempermudah pekerjaan pihak konsumen dan pihak Travel Dery Wisata dalam melakukan pemesanan dan rekapitulasi harian setiap cabang Travel Dery Wisata.

## **2.4 Teori Program**

### **2.4.1 Pengertian Basis Data (*Database*)**

Menurut Setyaningrum (2013:2), “Basis Data merupakan kumpulan data yang terdiri dari *atribut* dan *relationship* dari informasi suatu instansi perusahaan yang dirancang untuk memenuhi kebutuhan informasi dari sebuah organisasi”. Setiawan (2015:25), “*Database* (Basis Data) adalah kumpulan informasi yang

---



disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut”.

Dari beberapa definisi di atas penulis menyimpulkan bahwa *Database* merupakan sekumpulan data yang terdiri dari berbagai macam dan dikelompokkan berdasarkan kriteria tertentu.

## 2.4.2 Sekilas Tentang PHP

### 2.4.2.1 Pengenalan PHP

Menurut Setiawan (2015:33), “*Hypertext Preprocessor (PHP)* merupakan bahasa pemrograman script yang paling banyak dipakai saat ini. PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted) yang berwujud sekumpulan script yang digunakan untuk mengolah data form dari web. PHP banyak dipakai untuk membuat situs web yang dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. PHP biasanya berjalan pada sistem operasi linux”.

### 2.4.2.2 Script Dasar PHP

Contoh Script dasar :

```
<HTML>
<HEAD>
<TITLE>Contoh Pertama</Title>
</HEAD>
<?php>
Print(“Halo mas Bunafit Nugroho , Apa kabar “);
?>
<BODY>
</BODY></HTML>
```



### 2.4.3 Pengertian HTML

Menurut Setiawan (2015:33), “*Hypertext Markup Language (HTML)* adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser internet”. Menurut Pramono (2015:1), “HTML (*Hypertext Markup Language*) adalah bahasa pengkodean untuk menghasilkan dokumen-dokumen hypertext untuk digunakan di World Wide Web. Akan tetapi HTML bukanlah sebuah bahasa pemrograman, HTML hanya berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun”. Menurut Asropudin (2013:15), “*Hypertext Markup Language* adalah bahasa komputer yang digunakan untuk membuat sebuah halaman webpage.”

Maka Penulis simpulkan bahwa HTML (*Hypertext Markup Language*) merupakan bahasa yang digunakan dalam penyusunan dan pengembangan situs web dengan menggunakan beragam tata penulisan dan berisikan berbagai macam informasi.

### 2.4.4 Pengertian MySQL

Menurut Faizal dan Irmawati (2015:4), “*MySQL* adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris : *database management system*) atau DBMS yang *multithread, multi-user* dengan sekitar 6 (enam) juta instalasi di seluruh dunia”. Setiawan (2015:30), “*MySQL* adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia.

Maka penulis simpulkan bahwa *MySQL* merupakan database server yang akrab dan sering digunakan dan dapat menyimpan data dalam bentuk tabel-tabel.

### 2.4.5 Pengertian Webserver

Untuk bisa mengakses aplikasi web yang akan dibuat oleh penulis, maka diperlukan yang namanya *webserver*. Penulis menggunakan *XAMPP* sebagai *webserver* dalam membangun Aplikasi Travel *Booking* berbasis Web *Mobile* pada CV Dery.



#### **2.4.6 Pengertian XAMPP**

Menurut Riyanto (2014:11), “XAMPP merupakan paket PHP dan MySQL berbasis *open source* yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP. XAMPP mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket”. Sedangkan menurut Nugroho (2013:1) “XAMPP adalah paket program web lengkap yang dapat anda pakai untuk belajar pemrograman web, khususnya *PHP* dan *MySQL*, paket ini dapat didownload secara gratis dan legal”. Menurut Adelheid (2013:1), “XAMPP adalah perangkat lunak yang dikembangkan oleh apache. Kegunaan dari aplikasi ini adalah untuk membangun server yang berdiri sendiri (*localhost*)”.

Maka Penulis simpulkan bahwa XAMPP merupakan sebuah paket lengkap yang biasanya dipakai untuk belajar dan mempermudah penggunaan PHP.