



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Krisbiantoro (2018:1), “Komputer merupakan alat yang dipakai untuk mengolah dan memproses data menurut perintah yang telah dirumuskan”.

Selain itu, Januarti (dalam Jurnal Ikhsan dan Kurniawan, 2015:13) menyatakan bahwa “Komputer adalah sebuah mesin hitung elektronik yang secara cepat menerima informasi masukan digital dan mengolah informasi tersebut menurut seperangkat instruksi yang tersimpan dalam komputer tersebut dan menghasilkan keluaran informasi yang dihasilkan setelah diolah”.

Berdasarkan beberapa definisi di atas dapat disimpulkan bahwa komputer adalah alat yang digunakan untuk melakukan pengolahan data, dimana proses pengolahan data ini dapat dilakukan komputer secara cepat sesuai instruksi yang telah diprogram pada perangkat komputer tersebut.

2.1.2 Pengertian Data

Menurut Anggraeni dan Irviani (2017:12), “Data adalah deskripsi tentang benda, kejadian, aktivitas, dan transaksi, yang mempunyai makna atau tidak berpengaruh secara langsung kepada pemakai”.

Sedangkan menurut Pane, Sari, dan Wicaksono (2020:7), “Data merupakan sekumpulan keterangan atau fakta mentah berupa simbol, angka, kata-kata, atau citra, yang didapatkan melalui proses pengamatan atau pencarian ke sumber-sumber tertentu”.

Berdasarkan beberapa definisi di atas dapat disimpulkan bahwa data merupakan kumpulan fakta mentah yang berfungsi untuk mendeskripsikan sesuatu baik menggunakan simbol, angka, kata, maupun citra.



2.1.3 Pengertian Informasi

Menurut Prehanto (2020:12), “Informasi merupakan hasil pengolahan data dengan cara tertentu sehingga lebih berarti dan berguna bagi penerimanya”.

Sedangkan menurut Anggraeni dan Irviani (2017:13), “Informasi adalah sekumpulan data/fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima”.

Berdasarkan beberapa definisi di atas dapat disimpulkan bahwa informasi adalah hasil dari pengolahan data yang dapat memberikan manfaat bagi penerimanya.

2.1.4 Pengertian Aplikasi

Menurut Pane, Sari, dan Wicaksono (2020:132), “Pengertian aplikasi adalah program yang dibuat dengan tujuan melaksanakan fungsi sesuai dengan kegunaan aplikasi, penggunaannya dan jenis aplikasi itu sendiri”.

Tirtobisono (dalam Jurnal Sallaby, Utami, dan Arliando, 2015:172) menyatakan bahwa, “Aplikasi adalah istilah yang digunakan untuk pengguna komputer bagi pemecahan masalah. Biasanya istilah aplikasi dipasangkan untuk digabungkan dengan suatu perangkat lunak”.

Berdasarkan beberapa definisi di atas dapat disimpulkan bahwa aplikasi adalah suatu program yang digunakan untuk melakukan pemecahan masalah tertentu sesuai dengan fungsinya.

2.1.5 Pengertian *Store Assistant*

Assistant atau dalam bahasa Indonesia disebut asisten, menurut Kamus Besar Bahasa Indonesia (KBBI) Online, “asisten adalah orang yang bertugas membantu orang lain dalam melaksanakan tugas profesional, misalnya dalam pekerjaan, profesi, dan kedinasan”.

Sehingga, *Store Assistant* dapat diartikan sebagai orang atau sistem yang bertugas untuk membantu kegiatan yang ada pada sebuah toko, seperti pencatatan penjualan, pembelian, dan jumlah barang.



2.1.6 Pengertian PWA (*Progressive Web App*)

Santoso (2019:3-5) menyatakan bahwa, *Progressive Web Application* (PWA) adalah sebuah *website* yang dibangun menggunakan teknologi web modern, namun dapat berlaku seperti sebuah *Mobile App*. PWA memiliki beberapa kelebihan, diantaranya :

1. Respon seperti *Mobile App*

Pada PWA, semuanya di-cache di dalam “*App Shells*” sehingga tampak seperti aplikasi *Mobile App*.

2. Layout Responsif

PWA menggunakan *responsive layout*, walaupun ini bukan kelebihan utamanya. *User* melakukan *browsing website* melalui *browser* yang *support Service Worker* akan mengambil keuntungan dari *layout* PWA ini.

3. Tersedia di Mode *Offline*

Setelah Anda membuka PWA di *web browser* Anda, maka *cache* situs web akan disimpan secara otomatis di *web browser* Anda. Itu berarti Anda nantinya dapat dengan mudah membuat beberapa halaman web dalam mode *offline*.

4. *Add To Home Screen* (A2HS)

Seperti *bookmark page* di *web browser*, Anda dapat menambahkan PWA ke layar *Home* Anda untuk akses cepat. Nah, yang spesial adalah PWA bisa bekerja *offline* juga, mirip seperti *Mobile App*.

5. *Push Notification*

Setelah Anda membuka PWA di *web browser* Anda, maka Anda akan diminta untuk mengizinkan pemberitahuan “*Allow Push Notification*” yang akan memungkinkan Anda untuk mendapatkan pesan dan peringatan cepat dari situs web.

2.2 Teori Khusus

2.2.1 Pengertian Pemrograman Terstruktur

Menurut Sukamto dan Shalahuddin (2013:67-68) pemrograman terstruktur adalah suatu paradigma pemrograman yang membagi program ke dalam beberapa



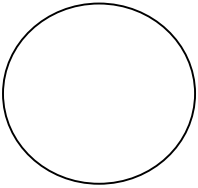
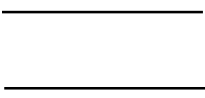
fungsi atau prosedur yang ditulis secara sekuensial atau terurut, sesuai dengan kebutuhan program tersebut.

2.2.2 Data Flow Diagram (DFD)



Kristanto (2018:61) menyatakan bahwa DFD adalah suatu model logika yang menggambarkan hal-hal yang terjadi selama data diproses, mulai dari asal dan tujuan data, dimana data disimpan, sumber proses yang menghasilkan data tersebut hingga interaksi dan proses yang terjadi pada data tersebut.

Ada 2 teknik dasar DFD yang umum dipakai yaitu Gane/Sarson dan Yourdon/De Marco. Berikut adalah notasi-notasi DFD menurut Yourdon/De Marco:

Tabel 2.1. Simbol–simbol *Data Flow Diagram* (DFD)

No.	Notasi	Keterangan
1.		<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>
2.		<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 tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM)).</p> <p>Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>



3.		<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 berkaitan dengannya aliran data 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>
4.		<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> <p>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>

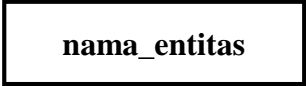
Sumber : Sukamto dan Shalahuddin (2013:71-72)

2.2.3 Entity Relationship Diagram (ERD)



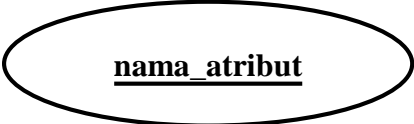
Mulyani (2016:100) menyatakan bahwa, “ERD merupakan *tools* yang digunakan untuk memodelkan struktur data dengan menggambarkan entitas dan hubungan antara entitas (*relationship*) secara abstrak (konseptual)”.

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

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

No.	Simbol	Deskripsi
1.	<p>Entitas (<i>Entity</i>)</p> 	<p>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</p>



2.	<p>Relasi (<i>Relationship</i>)</p> 	<p>Relasi (<i>relationship</i>) adalah hubungan yang terjadi antara satu entitas dengan entitas lainnya.</p>
3.	<p>Atribut</p> 	<p>Atribut adalah karakteristik dari entitas atau dari relasi yang menyediakan penjelasan detail tentang entitas atau relasi tersebut.</p>
4.	<p>Atribut kunci primer</p> 	<p>Atribut kunci primer sebagai <i>field</i> atau kolom data yang perlu disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> (<i>Primary Key</i>) yang diinginkan biasanya berupa id, kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut bersifat unik (berbeda tanpa ada yang sama).</p>
5.	<p>Penghubung (<i>Line/connector</i>)</p> <p>1 ————— M</p>	<p>Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki kemungkinan jumlah pemakaian. Kemungkinan jumlah pemakaian maksimum keterhubungan antara entitas satu dengan entitas yang lainnya disebut kardinalitas.</p>



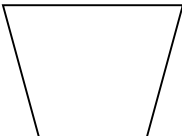

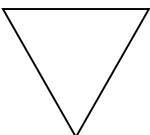
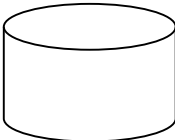
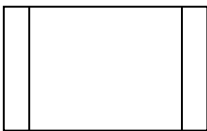
Sumber : Syahidi dan Subandi (2018:34-36)

2.2.4 Blockchart

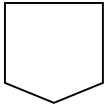
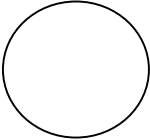

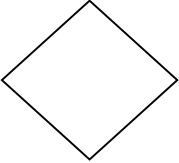


Kristanto (2018:75) menyatakan bahwa, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”.

Adapun simbol-simbol yang sering digunakan dalam *Blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.3 Simbol-simbol *Blockchart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bundel, 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.
11.		Pengambilan keputusan (<i>decision</i>).
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukan data secara manual.

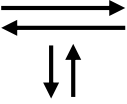

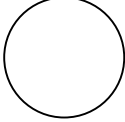
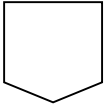
Sumber : Kristanto (2018:75-77)

2.2.5 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.”


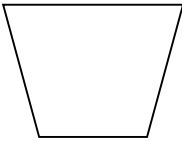
Simbol-simbol *flowchart* untuk *flow direction*, *processing*, dan *input/output* dicantumkan pada tabel berikut ini.

**Tabel 2.4** Simbol-simbol *Flowchart* untuk *Flow Direction Symbol*

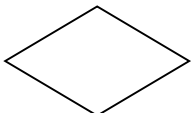

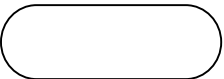
No.	Simbol	Maknanya
1.		Simbol arus/ <i>flow</i> Menyatakan jalannya suatu proses
2.		Simbol <i>communication link</i> Menyatakan transmisi data dari satu lokasi ke lokasi lain
3.		Simbol <i>connector</i> Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
4.		Simbol <i>offline connector</i> Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda

Sumber : Jalinus dan Ambiyar (2016:39)

Tabel 2.5 Simbol-simbol *Flowchart* untuk *Processing Symbol*

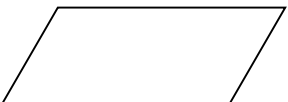
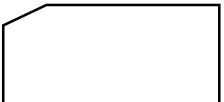
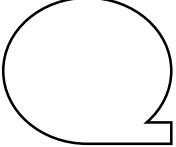
No.	Simbol	Maknanya
1.		Simbol <i>process</i> Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
2.		Simbol <i>manual</i> Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer





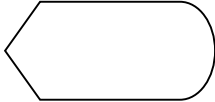
3.		<p>Simbol <i>decision</i></p> <p>Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya/tidak</p>
4.		<p>Simbol <i>predefined process</i></p> <p>Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal</p>
5.		<p>Simbol <i>terminal</i></p> <p>Menyatakan permulaan atau akhir suatu program</p>

Sumber : Jalinus dan Ambiyar (2016:40)

Tabel 2.6 Simbol-simbol *Flowchart* untuk *Input/Output Symbol*

No.	Simbol	Maknanya
1.		<p>Simbol <i>input/output</i></p> <p>Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung peralatan</p>
2.		<p>Simbol <i>punched card</i></p> <p>Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu</p>
3.		<p>Simbol <i>magnetic tape</i></p> <p>Menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis</p>



4.		Simbol <i>disk storage</i> Menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
5.		Simbol <i>document</i> Mencetak keluaran dalam bentuk dokumen (melalui printer)
6.		Simbol <i>display</i> Mencetak keluaran dalam layar monitor

Sumber : Jalinus dan Ambiyar (2016:40-41)

2.2.6 Kamus Data (*Data Dictionary*)

Maniah dan Hamidin (2017:59) menyatakan bahwa Kamus data (KD) berfungsi membantu pelaku sistem untuk memahami aplikasi secara detail, kamus data mengorganisasi semua elemen data yang digunakan dalam sistem dengan presisi yang sedemikian rupa sehingga pemakai dan penganalisa sistem memiliki dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

Berikut ini merupakan simbol-simbol yang digunakan dalam penulisan kamus data:

Tabel 2.7 Simbol-simbol Kamus Data

No	Simbol	Uraian
1.	=	Terdiri dari, mendefinisikan, diuraikan menjadi
2.	+	Dan
3.	()	Menunjukkan suatu elemen yang bersifat pilihan (opsional). Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi atau nol untuk <i>field-field</i> numeric pada struktur file.



4.	{ }	Menunjukkan elemen-elemen <i>repetitive</i> , juga disebut kelompok berulang atau tabel-tabel. Kemungkinan bisa ada satu atau beberapa elemen berulang di dalam kelompok tersebut. Kelompok berulang bisa mengandung keadaan-keadaan tertentu, seperti misalnya, jumlah pengulangan yang pasti atau batas tertinggi dan batas terendah untuk jumlah pengulangan.
5.	[]	Menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa kedua-duanya ada secara bersamaan. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain (dengan kata lain, memilih salah satu dari sejumlah alternatif, seleksi).
6.		Pemisah sejumlah alternatif pilihan antara <i>symbol</i> []
7.	@	Identifikasi atribut kunci
8.	**	Komentar

Sumber : Maniah dan Hamidin (2017:60)

2.3 Teori Program

2.3.1 Pengertian PHP

Enterprise (2018:1) menyatakan bahwa PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif. Dinamis artinya tampilan *website* dapat berubah menyesuaikan dengan kondisi tertentu. Sedangkan interaktif artinya PHP dapat memberi *feedback* bagi *user*.

2.3.2 Sintak Dasar PHP

2.3.2.1 Embedded Sript

```
<!DOCTYPE html>
<html>
  <head>
    <title>.:: Embedded Script :.:</title>
```



```

</head>
<body>
  <h3>
    <?php
      echo "Text ini menggunakan teknik EMBEDDED SCRIPT";
    ?>
  </h3>
</body>
</html>

```

2.3.2.2 Non Embedded Script

```

<!DOCTYPE html>
<?php
echo "<html>";
echo "  <head>";
echo "    <title>.:: Embedded Script :.:</title>";
echo "  </head>";
echo "  <body>";
echo "    <h3>";
echo "      Text ini menggunakan teknik EMBEDDED SCRIPT";
echo "    </h3>";
echo "  </body>";
echo "</html>";
?>

```

2.3.3 Pengertian MySQL

Menurut Sianipar (2015:1), “MySQL secara inheren merupakan sistem dengan *database* jaringan, sehingga setiap klien dapat berkomunikasi dengan server yang dijalankan secara lokal pada mesin pengguna atau dengan server yang dijalankan di tempat lain, bisa saja di suatu mesin di benua lain”.

Sedangkan Enterprise (2018:2-3) mengungkapkan bahwa, *MySQL* merupakan server yang melayani *database*. Beberapa hal yang menyebabkan *MySQL* menjadi sangat populer, yaitu :



- a. Berlisensi *open-source*, sehingga Anda dapat menggunakannya secara gratis.
- b. Merupakan program yang *powerful* dan menyediakan fitur yang lengkap.
- c. Menggunakan bentuk standar bahasa SQL.
- d. Dapat bekerja dengan banyak sistem operasi dan dengan bahasa-bahasa pemrograman, seperti PHP, PERL, C, C++, JAVA, dan lain-lain.
- e. Bekerja dengan cepat dan baik, bahkan dengan data set yang banyak.
- f. Sangat mudah digunakan dengan PHP untuk pengembangan aplikasi web.
- g. Mendukung banyak *database*, sampai 50 juta baris atau lebih dalam suatu tabel.
- h. Dapat dikustomisasi sesuai dengan keinginan.

2.3.4 Pengertian XAMPP

Menurut Dadan dan Developers (2015:28), “XAMPP adalah salah satu aplikasi *web server* apache yang terintegrasi dengan *MySQL* dan *PhpMyAdmin*”.

Sedangkan menurut Haqi (2019:8), “XAMPP adalah perangkat lunak (*free software*) bebas, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program”.

Berdasarkan beberapa definisi di atas dapat ditarik kesimpulan bahwa XAMPP adalah aplikasi *web server* yang dapat digunakan untuk membangun suatu program pada berbagai sistem operasi.

3.3.5. Pengertian *PhpMyAdmin*

Menurut Maxicom (dalam Sa’ad, 2020:38), “*PhpMyAdmin* adalah sebuah perangkat lunak yang berbentuk halaman atau situs pada *web server* yang berfungsi untuk mengakses *database MySQL* secara GUI hingga *user* tidak harus memasukkan perintah-perintah *SQL* untuk melakukan proses pengelolaan *database*”.

Sedangkan menurut Putri (dalam Hikmah, Supriadi, dan Alawiyah, 2015:2), “*PhpMyAdmin* merupakan aplikasi yang dapat digunakan untuk membuat *database*, pengguna (*user*), memodifikasi tabel, maupun mengirim *database* secara cepat dan mudah tanpa harus menggunakan perintah (*command*) *SQL*”.



Berdasarkan beberapa definisi di atas dapat ditarik kesimpulan bahwa *PhpMyAdmin* merupakan aplikasi dalam bentuk *website* yang dapat memudahkan pengguna dalam proses pengolahan *database*.

3.3.6. Pengertian Sublime Text

Menurut Supono dan Putratama (2018:14), “Sublime Text merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi”.

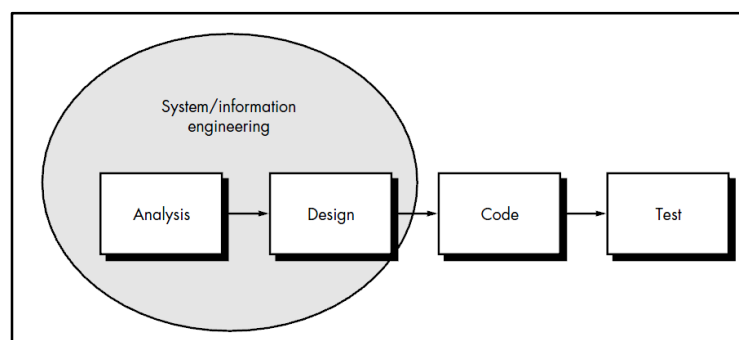
Sedangkan menurut Ardhana (dalam Sa’ad, 2020:39), “Sublime Text adalah salah satu program yang digunakan untuk melakukan *editor*, seperti HTML, PHP, CSS”.

Berdasarkan beberapa definisi di atas dapat ditarik kesimpulan bahwa Sublime Text merupakan aplikasi *text editor* yang dapat digunakan untuk keperluan *editing* berbagai bahasa pemrograman seperti PHP, C, Javascript, dan lain-lain.

3.3.7. Metode Pengembangan Sistem

3.3.7.1 Metode Waterfall

Rossa dan Shalahuddin (2013:28-30) 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*). Berikut adalah gambar model air terjun :



Gambar 3.1 Ilustrasi model *waterfall*



- a. Analisis kebutuhan perangkat lunak
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. Pembuatan kode program
Pada tahap pengkodean, 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
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*)
Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan-perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.