

BAB II

TINJAUAN PUSTAKA

2.1 *Internet*

Internet adalah jaringan komputer yang menghubungkan antar jaringan secara global, *internet* dapat juga disebut jaringan alam suatu jaringan yang luas. Seperti halnya jaringan komputer lokal maupun jaringan komputer area, *internet* juga menggunakan protokol komunikasi yang sama yaitu TCP/IP (*Transmission Control Protocol / Internet Protocol*) (Nofyat dkk., 2018). Sejarah pertama kali *Internet* ditemukan pada tahun 1960-an dan digunakan untuk kebutuhan riset di bidang militer. Sedangkan di Indonesia, *Internet* diperkenalkan pertama kali pada tahun 1994 yang diprakarsai oleh beberapa orang ahli di bidang IT. Perkembangan *Internet* dari tahun ke tahun mengalami peningkatan yang pesat seiring dengan kebutuhan manusia yang terus meningkat, sehingga manfaat *internet* dapat dirasakan pada berbagai bidang kehidupan.

2.2 *Aplikasi*

Aplikasi dapat diartikan sebagai suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia Huda, B., & Priyatna, B. (2019). Aplikasi adalah Program siap pakai yang dapat digunakan menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan.

2.3 *Website*

Website (situs *web*) adalah merupakan alamat (URL) yang berfungsi sebagai tempat penyimpanan data dan informasi dengan berdasarkan topik tertentu. *Web* adalah sistem *hypertext*, terdiri dari jutaan halaman teks yang dihubungkan oleh *hyperlink-hyperlink* (Sanjaya & Hesinto, 2018).

Website adalah sejumlah halaman *web* yang memiliki topik saling terkait antara halaman yang satu dengan halaman lain, yang biasanya ditempatkan pada sebuah server *web* yang dapat diakses melalui jaringan internet ataupun jaringan wilayah lokal (LAN) (Susilowati, 2019) .

Adapun cara kerja *web* adalah sebagai berikut:

1. Informasi *web* disimpan dalam dokumen dalam bentuk halaman-halaman *web* atau *web page*.
2. Halaman *web* tersebut disimpan dalam *computer server web*.
3. Ada komputer yang bertindak sebagai *computer client* yang mana ditempatkan program untuk membaca halaman *web* yang ada di *server web*.
4. *Browser* membaca halaman *web* yang ada di *server web*.

2.4 E-Commerce

E-Commerce merupakan aktivitas pembelian dan penjualan melalui jaringan internet dimana pembeli dan penjual tidak bertemu secara langsung, melainkan berkomunikasi melalui media internet Menurut (Ahmadi, 2020). *E-Commerce* adalah suatu proses membeli dan menjual produk-produk secara elektronik oleh konsumen dan dari perusahaan ke perusahaan dengan komputer sebagai perantara transaksi bisnis. *E-commerce* adalah segala bentuk aktivitas transaksi jual beli barang ataupun jasa dengan menggunakan media elektronik, Sandi, E. K. (2022).

2.4.1 Komponen E-Commerce

Ada beberapa kelebihan yang dimiliki *e-commerce* dan tidak dimiliki oleh transaksi bisnis secara offline yaitu:

1. Produk : banyak jenis produk yang bisa dipasarkan dan dijual melalui internet seperti pakaian, mobil, sepeda dll.
2. Tempat menjual produk : tempat menjual adalah internet yang harus dimiliki dominan dan hosting.
3. Cara menerima pesanan : email, telpon, sms dan lain-lain.
4. Cara pembayaran : credit card, paypal, tunai
5. Metode pengiriman : menggunakan pos Indonesia, EMS, atau JNE
6. Customer service : email, contact us, telepon chat jika tersedia dalam software.

2.4.2 Jenis-jenis *E-Commerce*

E-commerce memiliki berbagai transaksi dalam menerapkan sistemnya. Jenis-jenis transaksi *e-commerce* diantaranya sebagai berikut:

1. *Collaborative Commurce (C-commerce)*

Collaborative commurce yaitu kerjasama secara elektronik antara rekan bisnis. Kerja sama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (*supply chain*).

2. *Business To Business (B2B)*

E-commerce tipe ini meliputi transaksi antar organisasi yang dilakukan di *electronic market*.

3. *Business To Consumers (B2C)Business to consumers* Yaitu penjual adalah suatu organisasi dan pembeli adalah individu

4. *Consumer To Business (C2B)*

Dalam *Consumer to business* memberitahukan kebutuhan atas suatu produk atau jasa tertentu, dan para pemasok bersaing untuk menyediakan produk atau jasa tersebut ke konsumen.

5. *Costumer To Customer (C2C)*

Costumer to customer yaitu konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dalam *Costumer to customer* menjual produk atau jasa ke costumer lain. Dapat juga disebut sebagai pelanggan kepelanggan yaitu orang yang menjual produk dan jasa kesatu sama lain.

2.5 *Hypertext Preprocessor (PHP)*

PHP singkatan dari *Perl Hypertext Preprocessor* adalah suatu bahasa scripting khususnya digunakan untuk *web development*". PHP memiliki sifat server side scripting sehingga untuk menjalankan PHP harus menggunakan *web server* (Hidayatullah, 2017).



Gambar 2.1 Logo PHP

2.6 HTML

Menurut Hidayatullah dan Kawistara dalam jurnal yang ditulis oleh (Fitria & Permatasari, 2018) “*HyperText Markup Language (HTML)* adalah bahasa *standard* yang digunakan untuk menampilkan *web*”. Untuk mengetikkan script HTML dapat menggunakan text editor seperti Notepad sebagai paling bentuk sederhana atau text editor khusus yang dapat mengenali setiap unsur script HTML dan menampilkannya dengan warna yang berbeda sehingga mudah dibaca, seperti Notepad++, Sublime Text, dan masih banyak lagi aplikasi lain yang sejenis (Abdulloh, 2018). HTML disimpan dengan ekstensi *.html, jika di dalamnya tidak mengandung skrip PHP. Jika di dalamnya mengandung skrip PHP, maka disimpan dengan ekstensi *.php.



Gambar 2.2 Logo HTML

2.7 CSS *Cascading Style Sheet*

CSS adalah kependekan dari *Cascading Style Sheet*, berfungsi untuk mempercantik penampilan HTML atau menentukan bagaimana elemen HTML ditampilkan. Berikut beberapa pengertian CSS menurut para ahli: *Cascading Style Sheet* yang artinya gaya menata halaman bertingkat, yaitu setiap satu elemen telah

diformat dan mempunyai anak dan telah diformat, maka anak dari elemen tersebut mengikuti format induknya secara otomatis. CSS adalah singkatan dari *Cascading Style Sheet*. Kegunaan CSS yaitu untuk mengatur tampilan dokumen HTML, sebagai contoh pengaturan jarak antar baris, teks, format border, warna bahkan hingga penampilan file gambar.(Rudjiono & Saputro, 2021).

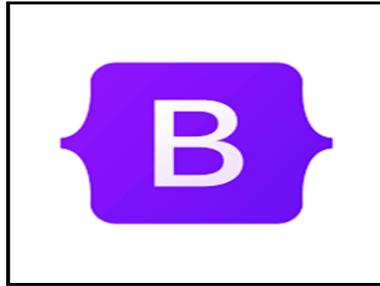
2.8 JavaScript

“*JavaScript* ialah suatu bahasa scripting yang digunakan sebagai fungsionalitas dalam membuat suatu web” Menurut Hidayatullah dan Kawistara dalam jurnal yang ditulis oleh (Fitria & Permatasari, 2018).

Javascript adalah bahasa pemrograman untuk sisi *client* atau *client side*. *Javascript* adalah bahasa pemrograman yang mendekati bahasa manusia atau bisa dikatakan bahasa tingkat tinggi, maka dari itu *javascript* mudah di pelajari. *Javascript* sendiri tujuannya di buat untuk memperkaya fitur pada *website* agar lebih dinamis, seperti untuk menampilkan dan menghilangkan objek-objek pada *website* kemudian dengan fungsi *javascript* dapat memanggil kembali objek yang di hilangkan tersebut (Mariko, 2019).

2.9 Bootstrap

Bootstrap adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *bootstrap* adalah *template* desain *web* dengan fitur *plus*. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai HTML dan CSS. Dengan menggunakan *bootstrap* seorang *developer* dapat dengan mudah dan cepat dalam membuat *front-end* suatu *website*. Anda hanya perlu memanggil *class-class* yang diperlukan, membuat tombol, grid, tabel, navigasi dan lainnya. *Bootstrap* sudah menyiapkan kumpulan komponen *class interface* dasar yang telah diolah sedemikian rupa untuk menyiptakan sebuah tampilan yang menarik dan ringan. Selain komponen *class interface*, *bootstrap* juga mempunyai grid yang berfungsi untuk mengatur *layout* pada halaman *website*. Selain itu pengembang juga bisa menambahkan class dan CSS tambahan sendiri, sehingga memungkinkan untuk membuat desain yang lebih variatif (Christian dkk 2018).



Gambar 2.3 Bootstrap

2.10 Database

Database adalah sekumpulan data yang sudah disusun rapi sedemikian rupa dengan ketentuan atau aturan tertentu yang saling berelasi antara satu sama lain sehingga memudahkan pengguna dalam mengelolanya juga memudahkan memperoleh informasi yang diinginkan. Selain itu ada pula yang mendefinisikan *database* sebagai beberapa kumpulan file, tabel atau arsip yang saling terhubung satu sama lain yang disimpan dalam media elektronik (Arizi, 2017).

Database juga merupakan kumpulan data yang saling terhubung satu dengan yang lainnya, tersimpan di *server* atau didalam komputer sendiri (*localhost*) serta menggunakan perangkat lunak tertentu untuk memanipulasinya. *Database* adalah salah satu komponen yang paling penting dalam sistem informasi. Beberapa *Database* yang ada saat ini adalah : MySQL, SQL Server, Ms.Access, Oracle, dan PostgreSQL (Sofwan & Christyono, 2017).

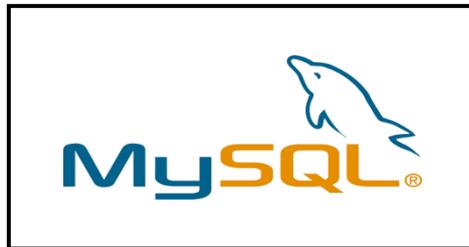
2.11 MySQL

“MySQL (*My Structure Query Language*) adalah sistem manajemen database SQL yang bersifat *Open Source* dan paling populer saat ini. Sistem *database* MySQL mendukung beberapa fitur seperti *multithreaded*, *multi-user* dan *SQL database management system (DBMS)*”. Menurut Hidayatullah dan Kawistara dalam jurnal yang ditulis oleh (Fitria & Permatasari, 2018). Beberapa keistimewaan dari MySQL, antara lain:

1. Portabilitas, artinya stabil pada berbagai sistem operasi seperti Windows, Linux, Mac OS.
2. Distribusikan secara open source atau bebas dibawah lisensi GNU *General*

Public License.

3. Multiuser, dapat digunakan oleh beberapa user dalam waktu secara bersamaan tanpa mengalami masalah.
4. Performance tuning, mempunyai kecepatan dalam pemrosesan *query* dibandingkan mesin database lainnya.
5. Konektivitas, melakukan koneksi dengan klien dengan klien dengan menggunakan protokol TCP/IP, *Unix socket*, serta *Named Pipes*.
6. Dapat menangani perubahan struktur tabel dengan perintah *ALTER TABLE* dibandingkan PostgreSQL maupun Oracle.
7. Mendukung penggunaan tipe kolom yang kompleks seperti *signed/unsigned integer*, *float*, *double*, *char*, *date*, *timestamp*, dan lain sebagainya.



Gambar 2.4 Logo MySQL

2.12 XAMPP

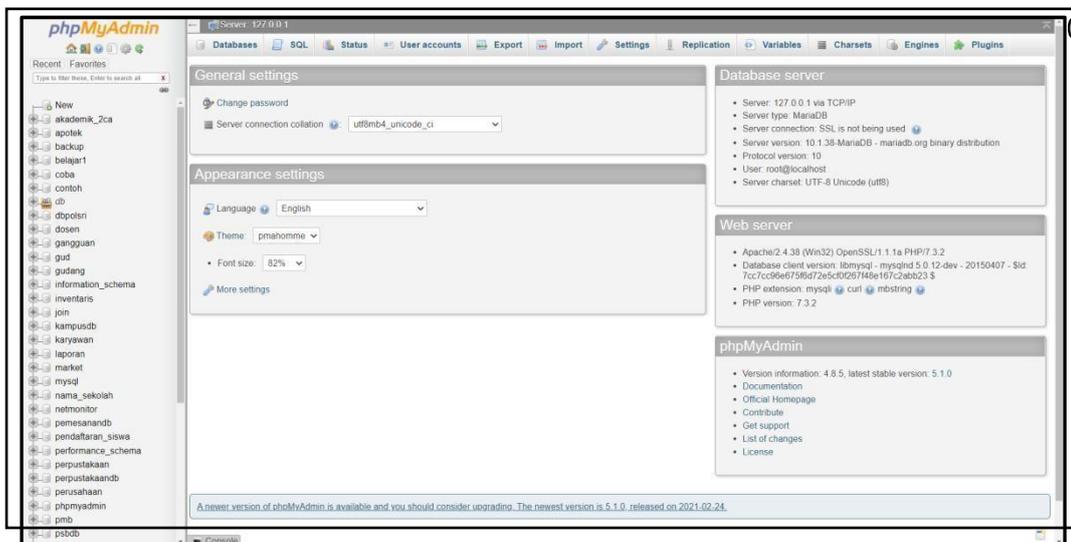
XAMPP adalah salah satu paket instalasi apache, PHP, dan MySQL secara instan yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut (Sihotang, 2018). XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari apache, mysql, phpMyAdmin, PHP, Perl, Freetype2, dan lainnya. XAMPP berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan *web* memerlukan PHP, apache, MySQL dan phpMyAdmin serta *software-software* yang terkait dengan pengembangan *web*. Dengan menggunakan XAMPP, kita tidak perlu menginstall aplikasi-aplikasi tersebut satu persatu (Sofwan, 2017). Gambar 3.7 berisi tampilan control panel XAMPP



Gambar 2.5 Tampilan control panel XAMPP

2.13 PHPMyAdmin

PHPMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL melalui Jejaring Jagat Jembar (World Wide Web). phpMyAdmin mendukung berbagai operasi MySQL, diantaranya mengolah basis data, table-tabel, bidang (*fields*), relasi (*relations*), *indeks*, pengguna (*users*), perijinan (*permission*), dan lain-lain (Haqi, 2019). Gambar 3.8 berisi tampilan *dashboard* PHPMyAdmin



Gambar 2.6 Tampilan Dashboard PHPMyAdmin

2.14 Codeigniter

CodeIgniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling powerful saat ini karena di dalamnya terdapat fitur lengkap aplikasi web dimana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu, *CodeIgniter* juga saat ini banyak digunakan khususnya bagi *developer* web untuk mengembangkan aplikasi berbasis webnya tersebut. Sebelum memahami *CodeIgniter*, anda juga harus mengetahui dasar apa itu MVC. MVC (*Model View Controller*) adalah suatu metode yang memisahkan data logic (*Model*) dari presentation logic (*View*) dan process logic (*Controller*) atau secara sederhana adalah memisahkan antara desain *interface*, data, dan proses (Hidayatullah, 2019). Dalam metode MVC terdapat tiga komponen yaitu :

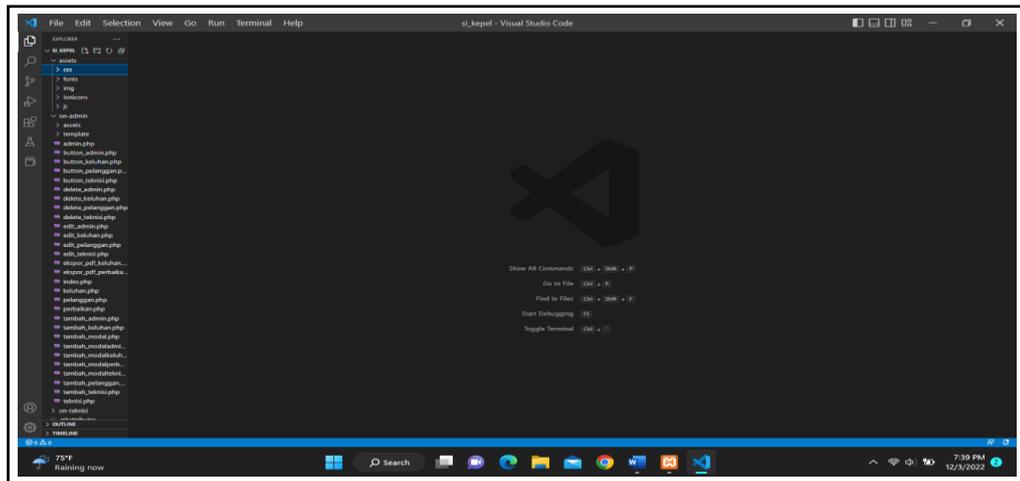
- *Model*, model mengelola basis data (RDBMS) seperti MySQL ataupun Oracle RDMS. Model berhubungan dengan database sehingga biasanya dalam model akan berisi class ataupun fungsi untuk membuat (*create*), melakukan pembaruan (*update*), menghapus data (*delete*), mencari data (*search*), dan mengambil data (*select*), pada database. Selain itu juga model akan berhubungan dengan perintahperintah query sebagai tindak lanjut dari fungsi-fungsi (*create, update, delete, select*).

- *View*, *view* adalah bagian *User Interface* atau bagian yang nantinya merupakan tampilan untuk end-user. View bisa berupa halaman html, css, rss, javascript, jquery, ajax, dan lain lain. Karena metode yang dipakai merupakan MVC sehingga dalam view tidak boleh terdapat pemrosesan data ataupun pengaksesan yang berhubungan dengan database. Sehingga view hanya menampilkan data-data hasil dari model dan *controller*.

- *Controller*, adalah penghubung antara view dan model, maksudnya ialah model tidak apat berhubungang langsung dengan view ataupun sebaliknya jadi, *controller* inilah yang digunakan sebagai jembatan dikeduanya. Sehingga tugas *controller* ialah sebagai pemrosesan data atau alur logic program, menyediakan variable yang akan ditampilkan di *view*, pemanggilan model sehingga model dapat mengakses database, *error handling*, validasi atau check terhadap suatu inputan. (Hidayatullah, 2017). Gambar 3.6 merupakan gambar konsep aliran M-V-C

2.15 Visual Studio Code

Visual Studio Code (VS Code) adalah kode editor yang bersifat *open source* untuk pengembangan dan debugging aplikasi cloud dan situs *Website* modern yang tersedia secara gratis di *Linux*, *OS X* dan *Windows*. *VS Code* mendukung lebih dari 30 bahasa pemrograman, markup dan database yang berbeda, beberapa di antaranya adalah PHP, HTML, CSS dan SQL (Gamma, 2016). Gambar 3.9 berisi tampilan dashboard *Visual Studio code*.



Gambar 2.7 Tampilan Dashboard *Visual Studio Code*

2.16 Blackbox Testing

Blackbox Testing merupakan metode pengujian perangkat lunak yang digunakan untuk menguji perangkat lunak tanpa mengetahui struktur internal kode atau program. Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya, Zuhri, K., & Ikhwan, A. (2020). 22 Pada *Blackbox Testing* ini dilakukan pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh customer. *Black-box testing* ini lebih menguji ke tampilan luar (Interface) dari suatu aplikasi agar mudah digunakan oleh pengguna. Pengujian ini tidak melihat dan menguji *source code program*. *Black-box testing* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya hanya terfokus pada informasi domain. Jadi metode ini bertujuan untuk memeriksa, setelah tahap akhir proyek, apakah perangkat lunak atau aplikasi berfungsi dengan baik, dan melayani penggunanya

secara efisien. Biasanya, penguji mencari fungsi yang hilang atau salah; antarmuka, kinerja, inisialisasi program dan kesalahan keluar; struktur data atau kesalahan akses basis data eksternal.

2.17 *Whitebox Testing*

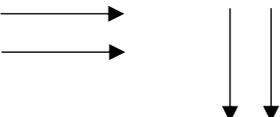
Metode pengujian yang biasa diterapkan pada evaluasi sistem adalah Pengujian *Blackbox* dan Pengujian *Whitebox*. (Andriyadi dkk, 2019). Kedua metode ini memiliki perbedaan. Misalnya, pengujian *Blackbox* hanya mengevaluasi fungsi sistem luar tanpa mengevaluasi kode pengujian bagian dalam. Pengertian lain, *White Box Testing* adalah salah satu cara untuk menguji suatu aplikasi atau software dengan melihat modul untuk memeriksa dan menganalisis kode program ada yang salah atau tidak. Jika modul ini dan telah diproduksi dalam output yang tidak memenuhi persyaratan, kode akan dikompilasi ulang dan diperiksa lagi sampai mencapai apa yang diharapkan, singkatnya *White Box Testing* ini menguji dengan cara melihat *Pure Code* dari suatu aplikasi/software yang diuji tanpa memperdulikan Tampilan atau UI dari aplikasi tersebut.

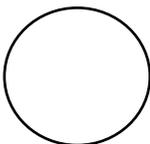
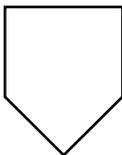
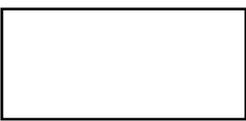
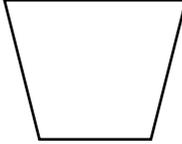
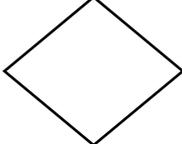
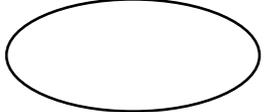
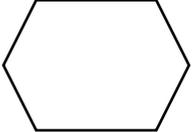
2.18 *Flowchart*

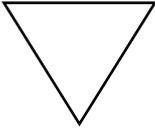
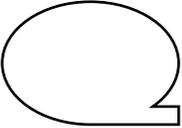
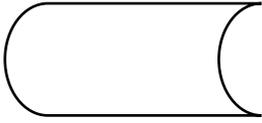
Flowchart atau diagram alir merupakan *chart* (bagan) yang menunjukkan hasil (*flow*) dalam program atau prosedur sistem secara logika. Digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut (Sitorus, 2018).

Tabel 2.1 yang berisi simbol-simbol yang digunakan dalam *flowchart* disertai dengan keterangan fungsinya :

Tabel 2. 1 Simbol-simbol *Flowchart*

NO	SIMBOL	KETERANGAN
1.		Simbol arus / <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses

2.		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3.		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses halaman yang berbeda
4.		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
5.		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer
6.		Simbol <i>decision</i> , berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan.
7.		Simbol <i>terminal</i> , berfungsi untuk menyatakan permulaan atau akhir suatu program
8.		Simbol <i>predefined process</i> , berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9.		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>

NO	SIMBOL	KETERANGAN
10.		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
11.		Simbol <i>manual input</i> , berfungsi untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i>
12.		Simbol <i>input/output</i> , berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
13.		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
14.		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
15.		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)
16.		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu

2.19 Metode Pengembangan Sistem

Metode air terjun atau yang sering disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), nama model ini sebenarnya

adalah “*Linear Sequential Model*” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*medelling*), konstruksi (*contruction*), serta penyerahan sistem ke para pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan.

2.19.1 Waterfall

Metode *waterfall* merupakan Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain pengodean, pengujian dan tahap pendukung atau support” (Rosa dan Shalahuddin, 2018). Model *waterfall* pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam *Software Engineering* (SE). Saat ini model *waterfall* merupakan model pengembangan perangkat lunak yang sering digunakan. Model pengembangan ini melakukan pendekatan secara sistematis dan berurutan. Disebut *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan.

2.19.2 Tahapan Waterfall

1) Requirements

Pada tahap ini, pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau *survey* langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2) Design

Pada tahap ini, pengembang membuat desain sistem yang membantu menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3) *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*.

4) *Verification*

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan

2.20 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek (Rosa Dan Shalahuddin, 2018). UML (*Unified Modeling Language*) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (OOAD&D/*object oriented analysis and design*) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an. UML merupakan gabungan dari metode Booch, Rumbaugh (OMT) dan Jacobson. Tetapi UML mencakup lebih luas daripada OOAD. Pada pertengahan saat pengembangan UML, dilakukan standarisasi proses dengan OMG (*Object Management Group*) dengan harapan UML bakal menjadi bahasa standar pemodelan pada masa yang akan datang (yang sekarang sudah banyak dipakai oleh berbagai kalangan) (Pratama, 2017).

2.20.1 *Activity Diagram*

Activity diagram adalah diagram yang menggambarkan sifat dinamis secara alamiah sebuah sistem dalam bentuk model aliran dan kontrol dari aktivitas ke aktivitas lainnya, Menurut Apol dan Radtyo dalam (Fauzi, 2019), *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau

proses bisnis atau menu yang ada pada perangkat lunak, *activity diagram* menerangkan tentang aktivitas-aktivitas yang dapat dilakukan oleh seorang *entity* atau pengguna yang akan diterapkan pada aplikasi. *Activity diagram* memodelkan alur kerja (*work flow*) sebuah proses bisnis dan urutan aktivitas dalam suatu proses. *Activity diagram* di gunakan sebagai penjelelasan aktivitas program tanpa melihat koding atau tampilan. Berikut ini adalah simbol- simbol yang digunakan dalam activity diagram serta keterangannya seperti yang dijelaskan pada tabel 2.3.

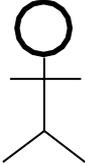
Tabel 2.3 Simbol *activity diagram*

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.20.2 Sequence diagram

Sequence diagram menggambarkan interaksi antara sejumlah objek dalam urutan waktu". (Tabrani, 2019). Sequence diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya sequence diagram digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antar objek, juga interaksi antar objek, dan menunjukkan sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem (MR Julianti, MI Dzulhaq, A Subroto, 2019).

Tabel 2.4 Simbol *Sequence Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem.
	<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan.
	<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari form.
	<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel.
	<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message.
	<i>A message</i>	Menggambarkan pengiriman pesan.