



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

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

Shelly dan Vermaat (2012:6), “Sebuah komputer merupakan sebuah perangkat elektronik, yang beroperasi di bawah perintah pengendali yang disimpan dalam memorinya, dimana dapat menerima data, memproses data berdasarkan aturan tertentu, mencetak hasilnya, dan menyimpan data untuk penggunaannya di masa depan”.

Setyadi (2010:1), “Komputer adalah alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan/*system* yang ada”.

2.1.2 Pengertian Internet

Shelly dan Vermaat (2012:11), “Internet adalah sekumpulan jaringan dari seluruh dunia yang menghubungkan jutaan perusahaan, badan pemerintahan, institutsi pendidikan, dan perorangan”.

Muslim dan Dayana (2016:39), “Internet (kependekan dari interconnection networking) secara harfiah ialah sistem global dari seluruh jaringan komputer yang saling terhubung menggunakan standar internet protocol suite (TPC/IP) untuk melayani miliaran pengguna di seluruh dunia”.

Fauziah (2014:1), “Internet merupakan jaringan global yang menghubungkan suatu jaringan yang satu dengan jaringan yang lainnya di seluruh dunia”.

2.1.3 Pengertian Perangkat Lunak

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



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

Utami dan Asnawati (2015:2), “Perangkat lunak (software) adalah program yang berisi kumpulan instruksi untuk melakukan proses pengolahan data”.

2.1.4 Metode Pengembangan Sistem

Menurut Sukanto dan Shalahuddin (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.

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. Dalam penelitian ini pengujian sistem akan menggunakan pengujian Black-Box. Pengujian Black-box berfokus pada persyaratan fungsional perangkat lunak. 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 dikirimkan ke *user*. Perubahan bisa terjadi adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Juansyah (2015:2), “Aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju”.

Afandi dan Saputra (2013:50), “Aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan”.

Jogiyanto berpendapat (dalam Rahman dan Santoso, 2015:79), “Aplikasi adalah sekelompok atribut yang terdiri dari beberapa form, report yang disusun sedemikian rupa sehingga dapat mengakses data. Aplikasi merupakan program yang berisikan perintah-perintah untuk melakukan pengolahan data”.

2.2.2 Pengertian Konsultasi

Siswohardjono berpendapat (dalam Khuzaijah, 2018:241), “Konsultasi adalah wawancara antara dua orang dewasa dengan tujuan bahan yang diperolehnya dapat membuat suatu pola pengertian baru atau keputusan yang lebih mantap terhadap sesuatu”.

Rosyadi berpendapat (dalam Komarudin, 2014:97), “Konsultasi adalah aktivitas konsultasi atau perundingan seperti klien dengan penasehat hukumnya”.



2.2.3 Pengertian Jasa

Dharmesta berpendapat (dalam Yuliana, 2012:41), “Jasa adalah suatu kegiatan yang dapat berwujud maupun tidak berwujud yang dilakukan untuk melayani konsumen”.

Tjiptono berpendapat (dalam Triaman, dkk., 2017:298), “Jasa adalah sesuatu yang dapat diidentifikasi secara terpisah, tidak terwujud, ditawarkan untuk memenuhi kebutuhan”.

Kotler mengungkapkan (dalam Suratno, dkk., 2016:43), “Jasa adalah setiap tindakan atau kegiatan yang dapat ditawarkan oleh suatu pihak kepada pihak lain, yang pada dasarnya tidak berwujud atau tidak mengakibatkan kepemilikan”.

2.2.4 Pengertian Arsitek

Marhiyanto (2014:15), “Arsitek adalah ahli dalam membuat rancang bangun dan memimpin konstruksi”.

Adhie (2015:11), “Arsitek adalah ahli bangunan”.

2.2.5 Pengertian Bangunan

Darwin berpendapat (dalam Pertiwi, dkk., 2014:2), “Bangunan adalah konstruksi teknis yang ditanamkan atau melekat pada tanah maupun perairan”.

Qanun berpendapat (dalam Wardhani dan Fadhli, 2017:11), “Bangunan adalah bumi dan atau bangunan yang dimiliki, dikuasai dan atau dimanfaatkan oleh orang pribadi atau badan, perkebunan, perhutanan dan pertambangan”.

2.2.6 Pengertian Aplikasi Konsultasi dan Jasa Arsitek Bangunan pada CV RAN *Architecture* Berbasis Web

Dari uraian diatas dapat disimpulkan bahwa pengertian Aplikasi Konsultasi dan Jasa Arsitek Bangunan pada CV RAN *Architecture* merupakan suatu aplikasi yang berfungsi untuk mempermudah arsitek dalam berkomunikasi dengan konsumen serta sebagai wadah untuk menginput data-data dalam pemesanan jasa arsitek, sehingga dapat mempersingkat waktu ketika proses pemesanan jasa arsitek.



2.3 Teori Khusus

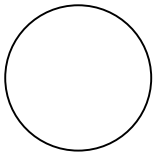
2.3.1 Data Flow Diagram (DFD)

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

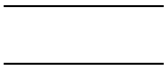
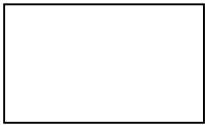

Saputra (2018:5), “*Data Flow Diagram* atau yang disingkat DFD merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas”.

Novienty dan Prapanca (2016:88), “*Data Flow Diagram* (DFD) merupakan suatu grafik yang menjelaskan sebuah sistem dengan bentuk-bentuk atau simbol untuk menggambarkan aliran data dari proses-proses yang saling berhubungan”.

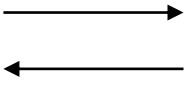
Tabel 2.1 Simbol-simbol pada Data Flow Diagram

No.	Simbol	Keterangan
1		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 catatan : nama yang diberikan pada sebuah proses biasanya berupa kata kerja

Lanjutan **Tabel 2.1** Simbol-simbol pada Data Flow Diagram

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 :</p> <p>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 terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>catatan :</p> <p>nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) berupa kata benda.</p>
4		<p>Duplikat 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>

Lanjutan **Tabel 2.1** Simbol-simbol pada Data Flow Diagram

5		<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 :</p> <p>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, Shalahuddin (2018:71))

Sukamto dan Shalahuddin (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 diatasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.3.2 Entity Relationship Diagram (ERD)

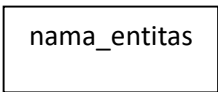
Sukamto dan Shalahuddin (2018:50) menjelaskan, “*Entity Relational Diagram (ERD)* digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain”.

Afandi dan Saputra (2013:52), “ERD adalah model konseptual yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis”.


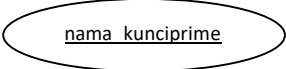
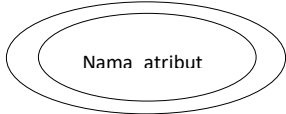
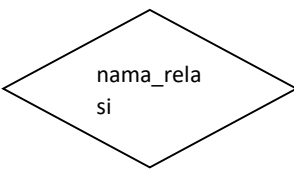
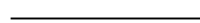
Aqil (dalam Rahman dan Santoso, 2015:81), “*Entity Relationship Diagram* adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis”.

Sukamto, Shalahuddin (2018:50) menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu :

Tabel 2.2 Simbol-simbol pada *Entity Relational Diagram*

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 adar dapat diakses oleh aplikasi komputer;

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

		penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama table.
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 multivali / <i>multivalve</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan



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

		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
--	--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------

(Sumber : Sukamto, Shalahuddin (2018 :75))

2.3.3 Flowchart

2.3.3.1 Pengertian *Flowchart*

Rahman dan Santoso (2015:81), “*Flowchart* adalah representasi secara grafik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek”.


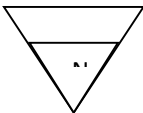
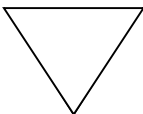
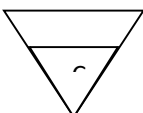
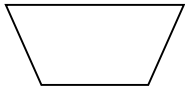



Saputra (2018:19), “Flowchart adalah suatu diagram yang menggambarkan alur kerja dari suatu sistem”.

Supardi (2013:51), “Bagan alir (*flowchart*) merupakan bagan (*chart*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika”.

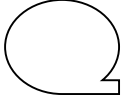
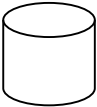

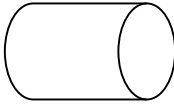
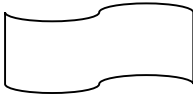
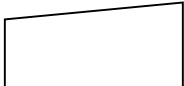


2.3.3.2 Simbol-simbol pada *Flowchart*

Adapun simbol-simbol yang sering digunakan dalam *flowchart* dapat dilihat dalam table berikut ini:

Tabel 2.4 Simbol-Simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
1.	Simbol dokumen		Menunjukkan dokumen input untuk proses manual, mekanik atau komputer.
2.	Simbol simpanan offline		File non-komputer yang diarsip urut angka (<i>numerical</i>)/
			File non-komputer yang diarsip urut huruf (<i>alphabetical</i>).
			File non-komputer yang diarsip urut tanggal (<i>cronological</i>).
3.	Simbol kegiatan manual		Menunjukkan pekerjaan manual.
4.	Simbol kartu plong		Menunjukkan <i>input/output</i> yang menggunakan kartu plong (<i>punched card</i>).
5.	Simbol proses		Menunjukkan kegiatan proses dari operasi program komputer.
6.	Simbol operasi luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer.



Lanjutan **Tabel 2.4** Simbol-Simbol *Flowchart*

7.	Simbol pita magnetic		Menunjukkan <i>input/output</i> menggunakan pita magnetik.
8.	Simbol <i>hard disk</i>		Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i> .
9.	Simbol <i>diskette</i>		Menunjukkan <i>input/output</i> menggunakan <i>diskette</i> .
10.	Simbol drum magnetic		Menunjukkan <i>input/output</i> menggunakan drum magnetik.
11.	Simbol pita kertas berlubang		Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang.
12.	Simbol <i>keyboard</i>		Menunjukkan <i>input</i> menggunakan <i>on-line keyboard</i> .
13.	Simbol display		Menunjukkan output yang ditampilkan di monitor.
14.	Simbol pita kontrol		Menunjukkan penggunaan pita kontrol (control tape) dalam batch control total untuk pencocokan di proses batch processing.

Lanjutan **Tabel 2.4** Simbol-Simbol *Flowchart*

15.	Simbol hubungan komunikasi		Menunjukkan proses transmisi data melalui channel komunikasi.
16.	Simbol penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
17.	Simbol input/output		Simbol input/output (input/output symbol) digunakan untuk mewakili data input/output.
18.	Simbol garis alir		Simbol garis alir (flow lines symbol) digunakan untuk menunjukkan arus dari proses.
19.	Simbol keputusan		Simbol keputusan (decision symbol) digunakan untuk suatu penyeleksian kondisi didalam program.
20.	Simbol proses terdefinisi		Simbol proses terdefinisi (<i>predifined prosesmsymbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.

Lanjutan **Tabel 2.4** Simbol-Simbol *Flowchart*

21.	Simbol persiapan		Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran.
22.	Simbol titik terminal		Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk awal dan akhir dari suatu proses.

(Sumber: Supardi, (2013:53–59))

2.3.4 Kamus Data

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

Sukamto dan Shalahuddin (2018:74), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

Table 2.5 Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	Dan
3	[[]]	baik ...atau...
4	{ } ⁿ	n kali diulang/ bernilai banyak
5	()	data operasional



Lanjutan **Table 2.5** Simbol-simbol dalam Kamus Data

6	*...*	batas komentar
---	-------	----------------

(Sumber : Sukanto, Shalahuddin (2018:74))

2.4 Teori Program

2.4.1 Pengertian Basis Data (*Database*)

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

Enterprise (2014:1), “*Database* adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap *database* mempunyai perintah tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya”.

Indera, dkk (2015:39), “Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan dengan yang lainnya, tersimpan di perangkat keras kompter dan digunakan perangkat lunak untuk memanipulasinya”.

2.4.2 Pengertian HTML (*Hypertext Markup Language*)

Shelly dan Vermaat (2012: 676), “*Hypertext Markup Language* (HTML) adalah bahasa pemformat khusus yang digunakan oleh para pemrogram untuk memformat dokumen-dokumen agar dapat ditampilkan di *Web*.”

Fauziah (2014:2), “HTML merupakan standard bahasa pemrograman yang populer dan digunakan untuk menampilkan dokumen yang kita buat di halaman web.”

Oktavian berpendapat (dalam Anwar dan Irawan, 2017:114), “HTML (*Hyper Text Markup Language*) adalah suatu bahasa yang dikenali web browser untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa”.

2.4.3 Pengertian CSS

Wismakarma dan Nina Rahayu beerpendapat (dalam Novienty dan Prapanca, 2016:86), “CSS atau *Cascading Style Sheet* adalah salah satu Bahasa pemrograman



desain web (*style sheet language*) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan menggunakan Bahasa penanda (*make up language*)”.

Muslim dan Dayana (2016:40), “CSS adalah singkatan dari *cascading style sheets*, berisi rangkaian intruksi yang menentukan bagaimana suatu text akan tertampil di halaman web”.

2.4.4 Pengertian PHP

Agus Saputra dkk berpendapat (dalam Novienty dan Prapanca, 2016:86), “PHP memiliki kepanjangan *PHP Hypertext Preprosesor*, merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu website dinamis”.

Enterprise (2018:1), “PHP merupakan bahasa pemrograman yang digunakan untuk membuat website dinamis dan interaktif.”



Gambar 2.1 Tampilan Logo PHP

2.4.5 Pengertian MySQL

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

Enterprise (2017:3), “*MySQL* adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak dipakai untuk berbagai kebutuhan”.

Indera, dkk (2015:40), “*MySQL* adalah sebuah *server database SQL multiuser dan multithreaded*”.



Gambar 2.2 Logo MySQL

2.4.6 Pengertian XAMPP

Enterprise (2018:3), “XAMPP merupakan server yang paling banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula.”

Novienty dan Prapanca (2016:86), “XAMPP merupakan paket PHP dan MySQL berbasis *open source* yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP”.

Rahman dan Santoso (2015:80), “Xampp adalah suatu bundle web server yang populer digunakan untuk coba-coba di windows karena kemudahan instalasinya. Bundle program open source tersebut berisi antara lain server web Apache, interpreter PHP, dan basis data MySQL”.



Gambar 2.3 Logo XAMPP

2.4.7 Pengertian JavaScript



Pahlevi, dkk (2018:28), “*JavaScript* adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*”.

Winarno, dkk (2014:2), “*JavaScript* adalah bahasa pemrograman *client-side* untuk memberikan efek dinamis, interaktif dan bersifat dependen, yang dieksekusi di browser”.

2.4.8 Pengertian *Sublime Text*

Pahlevi, dkk (2018:29), “*Sublime Text* merupakan salah satu text editor yang sangat powerful yang dapat meningkatkan produktivitas dan mengembangkan kualitas kode yang tinggi”.

Syifani dan Dores (2018:25), “*Sublime Text* adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim, Aplikasi ini sangatlah fleksibel dan powerfull. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan sublime-packages”.

Sibarani berpendapat (dalam Novienty dan Prapanca, 2016:87), “*Sublime Text* adalah editor teks yang digunakan untuk menerjemahkan hasil analisa dan rancangan kedalam bentuk code”.

Sibarani berpendapat (dalam Novienty dan Prapanca 2016:87), beberapa fitur yang diunggulkan dari aplikasi *Sublime Text* :

1. *Goto Anything*

Fitur yang sangat membantu dalam membuka file ataupun menjelajahi isi dari file hanya dengan beberapa *key storkes*.

2. *Multiple Selections*

Fitur ini memungkinkan user untuk mengubah secara interaktif banyak baris sekaligus, mengubah nama variabel dengan mudah, dan memanipulasi file lebih cepat dari sebelumnya.

3. *Command Pallete*

Dengan hanya beberapa *key storkes*, user dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalui menu.



4. *Distraction Free Mode*

Bila user memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu user dengan memberikan tampilan layar penuh. *Split Editing* dapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan editing perpecahan. Mengedit sisi file dengan sisi, atau mengedit dua lokasi disatu file. Anda dapat mengedit dengan banyak baris dan kolom yang user inginkan.

5. *Instant Project Switch*

Menangkap semua file yang dimasukkan kedalam project pada aplikasi ini. Terintegrasi dengan fitur *Goto Anything* untuk menjelajahi semua file yang ada ataupun untuk beralih ke file dalam project lainnya dengan cepat.

6. *Plugin API*

Dilengkapi dengan *plugin API* berbasis *Phyton* sehingga membuat aplikasi ini sangat tangguh.

7. *Customize Anything*

Aplikasi ini memberikan user fleksibilitas dalam hal pengaturan fungsional dalam aplikasi ini.

8. *Cross Platform*

Aplikasi ini dapat berjalan hampir disemua operating system modern seperti Windows, OS X, dan Linux based operating system.