



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.3.1 Pengertian Pengolahan Data

Kristanto (2018:8), menyatakan bahwa “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan. Sedangkan menurut Pane, dkk (2020:1), menyatakan “Pengolahan data adalah suatu proses menerima dan mengeluarkan data menjadi bentuk lain yaitu berupa informasi”. Selain itu, Herlina (2019:9), menyatakan bahwa “Pengolahan data adalah bagian dari rangkaian kegiatan penelitian setelah pengumpulan data”.

##### 2.3.2 Pengertian Perangkat Lunak

Menurut Kadir (2021:2), “Perangkat lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai. Di samping itu, menurut Rosa dan Shalahuddin (2018:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”. Sedangkan menurut IEEE dikutip Hendradjaya (2019:4), “Perangkat lunak adalah komputer program, prosedur dan dokumen serta data yang terkait dengan pengoperasian sistem komputer”.

##### 2.3.3 Pengertian Komputer

Kadir (2021:2), “Komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia”. Selain itu menurut Sihotang (2019), menyatakan bahwa “Komputer berasal dari bahasa latin “*computare*” yang mengandung arti menghitung”. Sedangkan menurut Matondang, dkk (2019:1), “Komputer adalah sebuah perangkat yang digunakan



untuk mengolah suatu data dan kemudian dapat disimpan ke dalam perangkat penyimpanan yang terdapat di dalam komputer”.

### 2.3.4 Metode Pengembangan Sistem

Penelitian Laporan Akhir ini menggunakan metode pengembangan sistem *Waterfall*. Rosa dan Shalahuddin (2018:28), menyatakan bahwa

Metode SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linear (*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*).

Di samping itu menurut Rizky dikutip Rusmawan (2019:89), menyatakan bahwa

*Waterfall* model sebagai salah satu teori dasar dan seakan wajib dipelajari dalam konteks siklus hidup perangkat lunak. Beliau juga berpendapat bahwa *Waterfall* model merupakan sebuah siklus hidup yang terdiri dari mulai fase hidup perangkat lunak sebelum terjadi hingga pascaproduksi. *Waterfall* model memiliki definisi sendiri bahwa sebuah hidup perangkat lunak memiliki sebuah proses yang linear dan sekuensial.

Sedangkan menurut Nasution dikutip Rusmawan (2019:89), “*Waterfall* merupakan salah satu metode pengembangan sistem informasi yang bersifat sistematis dan sekuensial, artinya setiap tahapan dalam metode ini dilakukan secara berurutan dan berkelanjutan”.

### 2.3.5 Studi Kelayakan

Menurut Kristanto (2018:42), “Studi kelayakan merupakan tahap yang paling penting, karena di dalamnya menyangkut berbagai aspek sistem baru yang diusulkan”. Dalam definisi lain Ashari (2021), menyatakan bahwa “Studi kelayakan (*Feasibility study*) adalah suatu studi atau pengkajian apakah suatu usulan proyek/gagasan usaha apabila dilaksanakan dapat berjalan dan berkembang sesuai dengan tujuannya atau tidak”.



Di samping itu menurut Boisot, Max H. dikutip Sulasih, dkk (2021:4), “Studi kelayakan merupakan sebuah evaluasi dan analisis potensi proyek yang diusulkan yang didasarkan pada ekstensif dan variabel dan untuk meminimalkan risiko untuk memberikan kenyamanan penuh pada keputusan pembuat untuk melanjutkan proyek”.

### **2.3.6 Kebutuhan Fungsional**

Kurniasari (2018), menyatakan bahwa “Kebutuhan fungsional diartikan sebagai layanan yang harus terdapat pada suatu sistem dan bagaimana sistem bereaksi serta berperilaku terhadap inputan dan situasi tertentu”. Selain itu, menurut Karlsson dikutip Kurniasari (2018), kebutuhan fungsional sering digunakan untuk menggambarkan prosedur untuk memulai dan mengakhiri operasi sistem perangkat lunak. Di samping itu, menurut Setiyani dan Tjandra (2021), “Kebutuhan fungsional merupakan kebutuhan akan fasilitas yang dibutuhkan serta aktivitas apa saja yang dilakukan oleh sistem secara umum”.

### **2.3.7 Kebutuhan Non-Fungsional**

Karlsson dikutip Kurniasari (2018), menyatakan bahwa “Istilah kebutuhan non-fungsional sering digunakan untuk menentukan karakteristik yang harus dimiliki oleh sistem yang tidak dapat dideskripsikan sebagai kebutuhan fungsional”. Sedangkan menurut Kurniasari (2019), “Kebutuhan non-fungsional merupakan batasan dari layanan atau fungsi yang ditawarkan oleh sistem secara keseluruhan”. Di samping itu menurut Hakim (2019), “Kebutuhan non-fungsional atau aspek kualitas merupakan sebuah konsep penting dalam rekayasa kebutuhan”.

## **2.2 Teori Khusus**

### **2.2.1 Pengertian Kamus Data**

Kristanto (2018:72), menyatakan bahwa “Kamus data adalah kumpulan-kumpulan elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau file di dalam sistem”. Di



samping itu menurut Rusmawan (2019:36), “Kamus data merupakan katalog fakta data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi”. Sedangkan Rianto (2021:133), menyatakan bahwa “Kamus data adalah kumpulan informasi terpusat terkait data”.

**Tabel 2.1** Simbol-simbol pada Kamus Data

| No | Simbol | Keterangan                     |
|----|--------|--------------------------------|
| 1. | =      | Terdiri atas                   |
| 2. | +      | Dan                            |
| 3. | ()     | Opsional                       |
| 4. | [ ]    | Memilih salah satu alternatif  |
| 5. | **     | Komentar                       |
| 6. | @      | Identifikasi atribut kunci     |
| 7. |        | Pemisah alternative simbol [ ] |

Sumber : Kristanto (2018:71).

### 2.2.2 Pengertian *Unified Modeling Language* (UML)



Sumber : wikipedia.org

**Gambar 2.1** Logo *Unified Modeling Language* (UML)

Menurut Rosa dan Shalahudin (2018:137), “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”. Dalam definisi lain Sri Dharwiyanti dikutip Rusmawan (2019:71), menyatakan bahwa “UML (*Unified Modeling Language*) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun perangkat lunak”. Di samping itu menurut Herlinah dan Musliadi

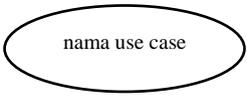
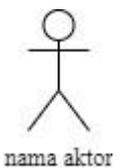


(2019:50), “UML merupakan salah satu alat bantu pengembangan sistem berorientasi objek”.

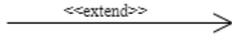
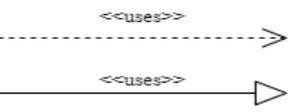
### 2.2.3 Pengertian *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2018:155), “*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang dibuat”. Selain itu menurut Shelly dan Rosenblatt dikutip Rusmawan (2019:72), “*Use Case* adalah representasi visual yang mewakili interaksi antara pengguna dan sistem informasi dalam UML”. Sedangkan menurut Caniago (2020) “*Use Case Diagram* adalah sesuatu atau proses merepresentasikan hal-hal yang dapat dilakukan oleh aktor dalam menyelesaikan sebuah pekerjaan”.

**Tabel 2.2** Simbol-simbol pada *Use Case Diagram*

| No | Simbol  | Deskripsi  |
|----|---|--|
| 1. |  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i> .   |
| 2. |  | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan dengan menggunakan kata benda di awal frase nama aktor. |
| 3. |  | Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.   |

Lanjutan Tabel 2.2 Simbol-simbol pada *Use Case Diagram*

| No | Simbol   | Deskripsi   |
|----|--|---|
| 4. | Ekstensi/ <i>extend</i><br><br>                     | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan. |
| 5. | Generalisasi/ <i>generalization</i><br><br>         | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.  |
| 6. | Menggunakan/ <i>include</i> / <i>uses</i><br><br> | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.  |

Sumber : Rosa dan Shalahuddin (2018:156-158).

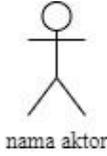
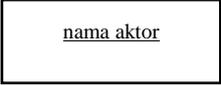
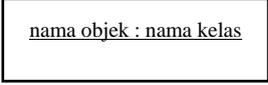
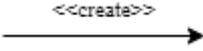
#### 2.2.4 Pengertian *Sequence Diagram*

Rosa dan Shalahuddin (2018:165), menyatakan bahwa “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”. Dalam definisi lain Adi Nugroho dikutip Rusmawan (2019:84), menyatakan bahwa “*Sequence Diagram* adalah diagram interaksi yang disusun berdasarkan urutan waktu”. Selain itu menurut Haqi (2019:31), “*Sequence diagram* yaitu salah satu jenis diagram pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence diagram* juga dapat menggambarkan urutan atau tahapan

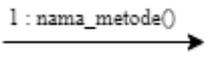
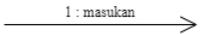
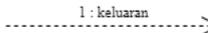
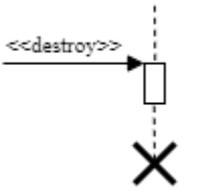


yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada use case diagram”.

**Tabel 2.3** Simbol-simbol pada *Sequence Diagram*

| No | Simbol   | Deskripsi  |
|----|--|--|
| 1. | <p>Aktor</p>  <p>atau</p>  <p>tanpa waktu aktif</p> | <p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p> |
| 2. | <p>Garis hidup/ <i>lifeline</i></p>   | <p>Menyatakan kehidupan suatu objek.</p>   |
| 3. | <p>Objek</p>    | <p>Menyatakan objek yang berinteraksi pesan.</p>   |
| 5. | <p>Pesan tipe create</p>    | <p>Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.</p>  |

Lanjutan Tabel 2.3 Simbol-simbol pada *Sequence Diagram*

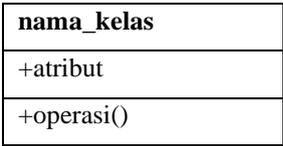
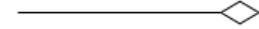
| No | Simbol  | Deskripsi  |
|----|---|--|
| 6. |    | Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri.  |
| 7. |    | Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.                   |
| 8. |    | Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.                   |
| 9. |  | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy. |

Sumber : Rosa dan Shalahuddin (2018:165-167).

### 2.2.5 Pengertian *Class Diagram*

Menurut Rosa dan Shalahuddin (2018:141), “Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem”. Sedangkan menurut Whitten dikutip Suendri (2019), “Kelas sebagai suatu set objek yang memiliki atribut dan perilaku yang sama kadang disebut kelas objek”. Selain itu, menurut Destriana (2021:19) “Class diagram adalah bentuk visual dari sebuah struktur atau sistem program pada jenis-jenis yang di bentuk dimana Class diagram merupakan alur jalannya *database* pada sebuah sistem”.

Tabel 2.4 Simbol-simbol pada *Class Diagram*

| No | Simbol  | Deskripsi   |
|----|---|---|
| 1. | <p>Kelas</p>   | Kelas pada struktur sistem.   |
| 2. | <p>Antarmuka / <i>interface</i></p>                  | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.   |
| 3. | <p>Asosiasi/ <i>association</i></p>                | Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .   |
| 4. | <p>Asosiasi bearah/ <i>direct association</i></p>  | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| 5. | <p>Generalisasi</p>                                | Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).   |
| 6. | <p>Kebergantungan/ <i>dependency</i></p>           | Relasi antarkelas dengan makna kebergantungan antarkelas.   |
| 7. | <p>Agregasi/ <i>aggregation</i></p>                | Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> ).  |

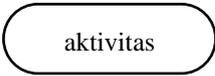
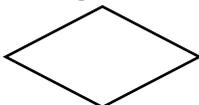
Sumber : Rosa dan Shalahuddin (2018:146-147).



### 2.2.6 Pengertian *Activity Diagram*

Rosa dan Shalahuddin (2018:161), menyatakan bahwa “Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. Di samping itu, menurut Glossary of key Terms dikutip Rusmawan (2019:79), “*Activity Diagram* adalah diagram aktivitas adalah bentuk visual dari alir kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, pengulangan, dan *concurrency*”. Sedangkan menurut Haqi (2019:31), “*Activity diagram* atau diagram aktivitas yaitu salah satu jenis diagram pada UML yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem”.

**Tabel 2.5** Simbol-simbol pada *Activity Diagram*

| No | Simbol  | Deskripsi  |
|----|---|--|
| 1. | Status awal<br>                  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.        |
| 2. | Aktivitas<br>                    | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.             |
| 3. | Percabangan/ <i>decision</i><br> | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.                    |
| 4. | Penggabungan/ <i>join</i><br>    | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.           |
| 5. | Status akhir<br>                 | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 6. | Swimlane  | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.       |



**Lanjutan Tabel 2.5** Simbol-simbol pada *Activity Diagram*

| No | Simbol   | Deskripsi |
|----|--|-----------|
|    | <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p style="text-align: center;">nama swimlane</p> </div> <p style="text-align: center;">Atau</p> <div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">nama swimlane</p> </div> |           |

Sumber : Rosa dan Shalahuddin (2018:162-163).

## 2.3 Teori Judul

### 2.3.1 Pengertian Aplikasi

Menurut Indrajani (2018:3), “Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk penyelesaian tugas-tugas khusus dari pemakai komputer”. Sedangkan menurut Kadir (2021:4), “Aplikasi adalah program biasa digunakan untuk menyatakan hasil karya yang berupa instruksi-instruksi untuk mengendalikan komputer”. Selain itu menurut Huda dan Priyatna (2019), “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”.

### 2.3.2 Pengertian Jual Beli

Menurut Pudjirahargo dan Muhith (2019:24), “Kata “jual beli” adalah frase (gabungan dua kata) yang terdiri dari dua kata dimana masing-masing kata itu



memiliki makna yang saling berlawanan”. Sedangkan menurut Wajdi dan Lubis (2021:239), “Pada intinya jual beli merupakan suatu perjanjian tukar-menukar barang/ benda yang mempunyai manfaat untuk penggunanya, kedua belah pihak sudah menyepakati perjanjian yang telah dibuat”. Di samping itu Syuhada dan Sungarso (2021:133), menyatakan bahwa “Menurut istilah, jual beli adalah kegiatan tukar-menukar suatu barang dengan barang yang lain dengan cara tertentu yaitu akad”.

### 2.3.3 Pengertian Website

Mulyani (2020:69), menyatakan bahwa “*Website* merupakan sekumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan menggunakan sebuah browser menggunakan URL *website*”.

Sedangkan menurut Abdulloh (2018:1), menyatakan bahwa “*Website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang diseluruh dunia”.

Di samping itu menurut Indrajani (2018:53), “*Website* merupakan halaman-halaman yang digunakan untuk menampilkan informasi, gambar gerak, suara, dan atau gabungan dari semuanya itu, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling berhubungan melalui *link-link*”.

### 2.3.3 Pengertian Aplikasi Jual Beli

Aplikasi Jual Beli adalah program berbentuk perangkat lunak yang berjalan pada sistem untuk melakukan transaksi jual beli. Dengan adanya aplikasi ini, perusahaan dapat melakukan proses transaksi jual beli dengan menggunakan aplikasi yang mempunyai *security* yang aman.



## 2.4 Teori Program

### 2.4.1 Pengertian *Hypertext Markup Language* (HTML)



Sumber: wikipedia.org (2022)

**Gambar 2.2** Logo Html

Menurut Adi (2020:2), “HTML (*Hypertext Markup Language*) sendiri merupakan sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, atau digunakan untuk membuat layout dan tampilan website”. Di samping itu, menurut Jubilee Enterprise (2018:21), “Sebutan lazim untuk HTML adalah bahasa markup (markup language) seperti yang ada di dalam singkatan HTML itu sendiri. Itu artinya, HTML adalah bahasa struktur untuk menandai bagian-bagian dari sebuah halaman”.

Sedangkan menurut Abdulloh (2018:7), “HTML merupakan singkatan dari *Hypertxt Markup Language* yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari website”.

### 2.4.2 Pengertian *Cascading Style Sheets* (CSS)



Sumber: wikimedia.org (2022)

**Gambar 2.3** Logo *Cascading style sheets* (CSS)

Menurut Abdulloh (2018:45), “CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai



property yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan”.

Di samping itu menurut Adi (2020:39), “CSS merupakan salah satu kode pemrograman yang digunakan untuk mengatur tampilan dan menghias halaman website supaya menarik”. Di sisi lain Mulyani (2020:77), menyatakan bahwa “*Cascading Style Sheets* adalah kumpulan perintah yang digunakan untuk menjelaskan tampilan sebuah halaman situs web dalam *mark-up language*”.

### 2.4.3 Pengertian *JavaScript*



Sumber: freepnglogos.com (2022)

**Gambar 2.4** Logo *JavaScript*

Menurut Mulyani (2020:106), “Bahasa Pemrograman yang sangat populer di kalangan web developer ini juga menjadi salah satu bahasa yang bisa kita pakai untuk membuat aplikasi mobile”. Di samping itu Abdulloh (2018:193), menyatakan bahwa “*JavaScript* merupakan bahasa pemrograman web yang pemrosesannya dilakukan di sisi client”. Sedangkan menurut Siahaan dan Sianipar (2018:1), “*Javascript* merupakan bahasa skript populer yang dipakai untuk menciptakan halaman Web yang dapat berinteraksi dengan pengguna dan dapat merespon *event* yang terjadi pada halaman”.

### 2.4.4 Pengertian *Hypertext Preprocessor (PHP)*





Sumber: php.net (2022)

### Gambar 2.5 Logo Hypertext Preprocessor (PHP)

Menurut Adi (2020:2), “PHP merupakan program *Server Side Scripting* yaitu program yang dapat dikompilasi atau diterjemahkan ke dalam server, sehingga dapat menghasilkan aplikasi web dinamis”. Dalam definisi lain Jubilee Enterprise (2018:1), “PHP merupakan bahasa pemrograman yang digunakan untuk membuat website dinamis dan interaktif”. Sedangkan menurut Abdulloh (2019:127), “PHP merupakan kependekan dari *PHP Hypertext Preprocessor* yaitu bahasa pemrograman web yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server”.

#### 2.4.5 Pengertian XAMPP



Sumber: wikimedia.org (2022)

### Gambar 2.6 Logo XAMPP

Menurut Jubilee Enterprise (2018:3), “XAMPP merupakan server yang paling banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula”. Sedangkan Menurut Jubilee Enterprise (2019:3), “XAMPP merupakan server yang paling banyak digunakan untuk para programmer PHP, khususnya level pemula”.

Di samping itu menurut Sidik (2020:3), menyatakan bahwa

XAMPP kependekan dari X Apache MySQL PHP Perl, X adalah sistem operasi (Windows, Linux, Unix), merupakan paket software yang terdiri dari server web (Apache), database (MySQL – Maria DB), dan pengembangan aplikasi (PHP dan Perl); disebut juga sebagai software Stack.



#### 2.4.6 Pengertian PHP MyAdmin



Sumber: wikimedia.org (2022)

**Gambar 2.7** Logo Php MyAdmin

Menurut Nugroho (2019:171), “PhpMyAdmin merupakan sebuah program bebas yang berbasis web yang dibuat menggunakan aplikasi PHP, tujuan dibuatnya program ini adalah untuk mengakses database MySQL, intinya adalah digunakan untuk menjadi Administrator dari Server MySQL”.

Sedangkan menurut Nugroho (2019:171), “Dengan adanya program ini akan mempermudah dan mempersingkat kinerja kita, dengan kelebihan-kelebihan yang ada mengakibatkan para pengguna awam tidak harus mampu untuk mengetahui sintaks-sintaks SQL dalam pembuatan database dan table”. Di samping itu, menurut Yudhanto (2018:12), menyatakan bahwa “PhpMyAdmin adalah aplikasi berbasis web yang digunakan untuk melakukan pengelolaan database MySQL dan atau tool yang populer untuk mengelola database MySQL”.

#### 2.4.7 Pengertian MySQL



Sumber: png.img.com (2022)

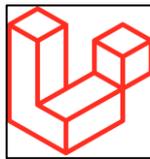
**Gambar 2.8** Logo MySQL

Jubilee Enterprise (2018:2), menyatakan bahwa “MySQL merupakan server yang melayani database”. Di samping itu, menurut Adi (2020:2), “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang multialur, dan multipengguna”.



Dalam definisi lain, Nugroho (2019:133), menyatakan bahwa “MySQL merupakan database yang paling digemari dikalangan programmer WEB, dengan alasan bahwa program ini merupakan database yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data”.

#### 2.4.8 Pengertian Laravel



Sumber: wikipedia.org (2022)

**Gambar 2.9** Logo Laravel

Menurut Abdulloh (2019:3), “Laravel merupakan salah satu *framework* PHP paling populer yang akan memudahkan programmer dalam membuat aplikasi web berbasis PHP”. Sedangkan menurut Mulyani (2020:82), “Laravel adalah sebuah *frameworks website* berbasis PHP yang sangat ekspresif”. Selain itu menurut Aminudin dikutip Khusni, dkk (2021), “Laravel adalah sebuah Framework PHP dirilis dibawah lisensi MIT dengan kode sumber yang sudah disediakan oleh Github”.

#### 2.4.9 Pengertian Visual Studio Code



Sumber: commons.wikimedia.org (2022)

**Gambar 2.10** Logo Visual Studio Code

Menurut Mulyani (2020:107), “Visual studio code adalah text editor keluaran dari perusahaan microsoft”. Di dalam definisi lain Hartati (2020), menyatakan bahwa “Visual studio code adalah *software* yang sangat ringan, namun



kuat *editor* kode sumbernya yang berjalan dari *desktop*". Sedangkan menurut Fauzan dan Roza (2020:117), "Visual studio code merupakan sebuah teks editor yang digunakan dalam pembuatan sistem informasi ini".