



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

Menurut Ansori dan Yulmaini (2019:139), "Komputer adalah suatu peralatan elektronik yang dapat menerima input, mengolah input, memberikan informasi, menggunakan suatu program yang tersimpan di memori computer, dapat menyimpan program dan hasil pengolahan serta bekerja secara otomatis".

Sedangkan menurut Blissmer (dikutip Harmayani, et al., 2021:3), "Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima input, memproses input sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan output dalam bentuk informasi".

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa komputer adalah sebuah alat elektronik yang dapat digunakan untuk mengelola data dengan berbagai proses sehingga menghasilkan informasi yang bermanfaat bagi penggunaannya.

##### 2.1.2 Pengertian Perangkat lunak (*Software*)

Menurut Indra (dikutip Sudarso, 2022:5), "*Software* adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer. Data elektronik yang disimpan dapat berupa program atau instruksi yang akan menjalankan perintah. *Software* dapat diartikan juga sebagai jenis program yang digunakan untuk pengoperasian komputer dan peralatannya".

Menurut Ali (dikutip Sudarso, 2022:5), "Perangkat lunak aplikasi merupakan program yang ditujukan untuk menyelesaikan suatu permasalahan dalam aplikasi yang tertentu yang sudah dibuat dengan menggunakan perangkat lunak bahasa (*language software*)".



Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa perangkat lunak adalah program yang berisikan sebuah instruksi untuk menjalankan perintah.

### **2.1.3 Pengertian Perangkat Keras (*Hardware*)**

Menurut Ansori dan Yulmaini (2019:56), “Perangkat keras komputer (*hardware*) adalah semua bagian fisik komputer, dan dibedakan dengan data yang berada di dalamnya atau yang beroperasi di dalamnya, seperti: *Motherboard, Power Supply, Processor, RAM, Hardisk, CD Drive, Battery CMOS, VGA Card, Sound Card, dll*”.

Menurut Blissmer (dikutip Ansori dan Yulmaini, 2019:56), “Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima input, proses input sesuai instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan output dalam bentuk informasi”.

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa perangkat keras adalah semua bagian fisik computer yang mampu melakukan beberapa tugas.

### **2.1.4 Pengertian Data**

Menurut Gunadi dan Widiyanto (dikutip Wahono dan Ali, 2021:227), “Data adalah bahan mentah yang perlu dilakukan pengolahan sehingga menghasilkan informasi atau keterangan, baik kualitatif maupun kuantitatif yang menunjukkan fakta sehingga dapat memberi manfaat bagi peneliti atau memberi gambaran kepada peneliti tentang kondisi atau sebuah keadaan”.

Menurut Dewi (2020:226), “Data adalah deskripsi dari suatu benda, peristiwa, aktivitas dan transaksi yang direkam, dikelompokkan dan disimpan dalam jumlah yang besar dan belum diolah menjadi sebuah informasi”.

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa data adalah bahan mentah dari deskripsi sesuatu yang perlu dikelola untuk menghasilkan sebuah informasi.



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

Menurut Lubis (dikutip Sudarso, 2022:3), "Basis data merupakan Suatu sistem penyusunan dan pengelolaan *record-record* dengan menggunakan komputer, dengan tujuan untuk menyimpan atau merekam serta memelihara data secara lengkap pada sebuah organisasi/perusahaan, sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk kepentingan proses pengambilan keputusan".

Menurut Sudarso (2022:3), "Basis data adalah tempat berkumpulnya data yang saling berhubungan dalam suatu wadah (perusahaan/organisasi) bertujuan agar dapat mempermudah dan mempercepat untuk pemanggilan atau pemanfaatan kembali data tersebut".

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa basis data adalah kumpulan data yang saling berhubungan yang menyediakan sebuah informasi.

### 2.1.6 Pengertian Internet (*Interconnected Network*)

Menurut Maharani, et al (2021:2), "Internet adalah suatu jaringan komunikasi yang memiliki fungsi untuk menghubungkan antara satu media elektronik dengan media elektronik yang lain dengan cepat dan tepat".

Menurut Putri (2020:93), "Internet merupakan jaringan komputer yang terdiri dari berbagai perangkat komputer yang terhubung melalui protokol untuk memeberikan informasi dan internet menyediakan sumber-sumber yang dapat di akses oleh manusia di seluruh dunia".

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa internet adalah jaringan yang menghubungkan komputer satu dengan yang lain untuk saling bertukar informasi.



### 2.1.7 Metode Pengembangan Sistem

Penelitian Laporan Akhir ini menggunakan metode pengembangan sistem *Waterfall*. Menurut Pressman (dikutip Sanubari, et al, 2021:41), menyatakan bahwa “Metode air terjun yang sering disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dinamakan 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 (*modeling*), konstruksi (*construction*), serta penyerahan sistem kepada para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan”.

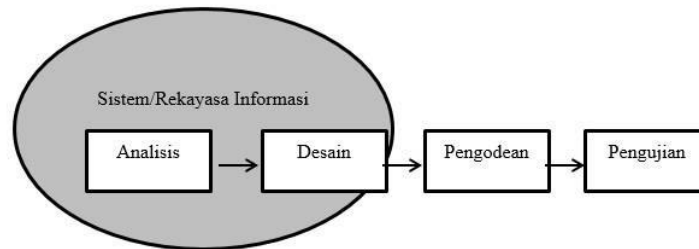
Menurut Hidayanti, et al (2020:269), menyatakan bahwa “Metode *Waterfall* adalah suatu proses pengembangan perangkat lunak berurutan, dimana kemajuan teknologi dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian”.

Menurut Rosa dan Shalahuddin (2018:28-30), menyatakan bahwa “Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*)”.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa Metode *Waterfall* adalah suatu proses atau metode pengembangan sistem secara berurutan yang mengalir ke bawah (seperti air terjun) melalui tahapan-tahapan *Requirement Analysis, Design System, Implementation, Testing dan Maintenance*.



Berikut adalah gambar metode waterfall:



**Gambar 2.1 Tahapan-tahapan Metode Waterfall**

Sumber : Rosa dan Shalahuddin (2018:29)

a. *Requirement Analysis* (Analisis Kebutuhan Perangkat Lunak)

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. *System Design* (Desain Sistem)

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengodean. Tahap ini menstranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. *Implementation/Coding* (Implementasi/Pengkodean)

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program computer sesuai dengan desain yang telah dibuat pada tahap desain.

d. *Testing* (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.



e. *Support & Maintenance* (Pendukung dan Pemeliharaan)

*Support & Maintenance* merupakan tahap akhir dalam model Waterfall. Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul. Dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

## 2.2 Teori Judul

### 2.2.1 Pengertian Aplikasi

Menurut Dewi, et al (2021:28), "Aplikasi adalah suatu program di dalam komputer atau handpone yang digunakan untuk menjalankan suatu program yang telah dibuat".

Menurut Huda dan Priyatna (2019:82), "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".

Dari kedua pendapat di atas, maka dapat disimpulkan bahwa aplikasi adalah *software* yang dimiliki oleh sebuah computer atau handphone yang berguna untuk mengerjakan tugas tertentu sesuai dengan kemampuan yang dimilikinya.

### 2.2.2 Pengertian Pengelolaan

Menurut Malayu S.P. Hasibuan (2021:13), "pengelolaan adalah ilmu dan seni mengatur proses pemanfaatan sumber daya manusia secara efektif, yang didukung oleh sumber-sumber lain dalam organisasi untuk mencapai tujuan tertentu".

Menurut Irwan, Muhammad (dikutip Anwar, 2020), "Pengelolaan adalah proses yang memberikan pengawasan pada semua hal yang terlibat dalam



pelaksanaan kebijaksanaan dan pencapaian tujuan dan proses melakukan kegiatan tertentu dengan menggerakkan tenaga orang lain”.

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa Pengelolaan merupakan proses yang terdiri atas tindakan-tindakan perencanaan, pengorganisasian, dan pengendalian yang dilakukan untuk menentukan serta mencapai sasaran yang telah ditentukan melalui pemanfaatan sumber daya manusia dan lainnya.

### **2.2.3 Pengertian Stok**

Menurut Yanuarsyah, et al., (2021:61-68) stok (persediaan) adalah bahan atau barang yang disimpan dan akan digunakan oleh perusahaan untuk memenuhi tujuan lain sebagai contoh digunakan dalam proses produksi, sebagai suku cadang dari peralatan atau mesin maupun dijual kembali.

Menurut Ristono (dikutip Ardiana dan Loekito, 2018) persediaan dapat diartikan sebagai barang yang disimpan untuk digunakan dan dijual pada masa yang akan datang, persediaan terdiri dari bahan baku dan barang setengah jadi serta produk yang sudah jadi

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa stok adalah berupa bahan mentah dalam proses dan barang jadi yang disimpan untukantisipasi permintaan atau memenuhi tujuan tertentu.

### **2.2.4 Pengertian Barang**

Menurut Kotler (dalam Sunyoto 2019:69), "Barang adalah sesuatu yang dapat ditawarkan ke pasar untuk diperlihatkan, dimiliki, dipakai atau dikonsumsi sehingga dapat memuaskan keinginan atau kebutuhan".

Menurut Sunyoto (2019:82), "Yang perlu diperhatikan dalam suatu barang adalah kualitas, ukuran, bentuk, daya Tarik, labeling, branding, packaging dan sebagainya untuk menyesuaikan selera yang sedang tumbuh".

Berdasarkan pengertian diatas dapat disimpulkan bahwa barang adalah segala sesuatu yang berwujud atau tidak berwujud yang dapat digunakan oleh pengguna.



## 2.3 Teori Khusus

### 2.3.1 *Unified Modelling Language (UML)*

Menurut Rosa dan Shalahuddin (2018:133), "*UML (Unified Modelling Language)* adalah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek".

Menurut Prasetya et al., (2022:15), "UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya".

Dari beberapa definisi diatas penulis menyimpulkan bahwa *UML (Unified Modelling Language)* adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasikan sebuah sistem pengembangan perangkat lunak berorientasi objek.

#### 2.3.1.1 *Klasifikasi Diagram UML (Unified Modelling Language)*

Menurut Rosa dan Shalahuddin (2018:140), *UML (Unified Modelling Language)* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

a. *Structure Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

b. *Behavior Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

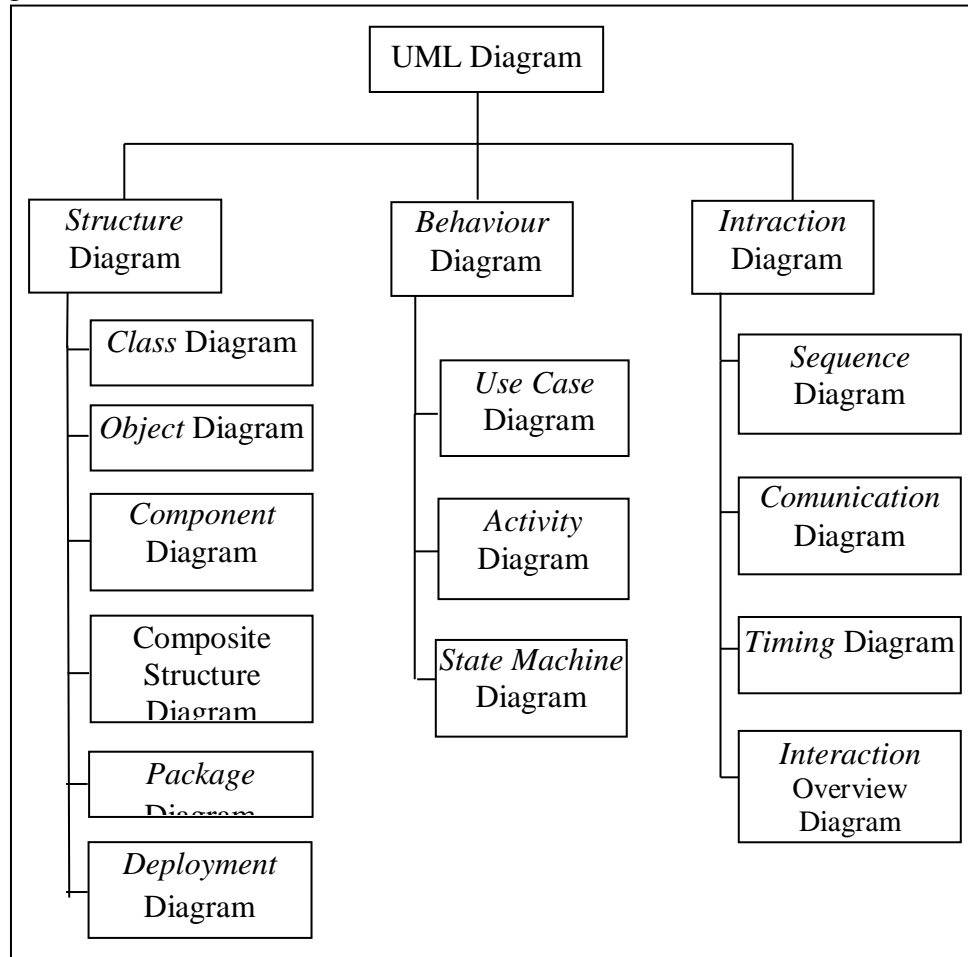
c. *Interaction Diagram*

Yaitu kumpulan diagram yang digunakan untuk menggambarkan interkasi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.





Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini:



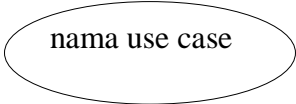
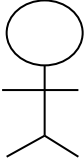

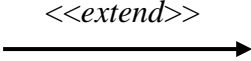
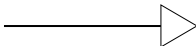
**Gambar 2.2** Klasifikasi Diagram UML (*Undefined Modelling Language*)

(Sumber: Rosa dan Shalahuddin (2016:140))

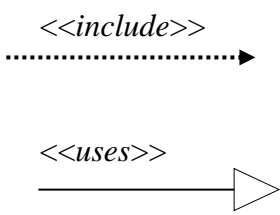
### 2.3.2 Use Case Diagram

Menurut Rosa dan Shalahuddin (2018:155), *Use case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.	<p><i>Use Case</i></p>  <p>nama use case</p>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
2.	<p>Aktor / <i>actor</i></p>  <p>nama aktor nama_interface</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
3.	<p>Asosiasi / <i>association</i></p> 	Komunikasi antar aktor dan use case yang berpartisipasi pada use case atau use case yang memiliki interaksi dengan aktor
4.	<p>Ekstensi / <i>extend</i></p> 	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; ditambahkan, missal arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i> -nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.
5.	<p>Generalisasi/<i>generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya




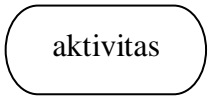
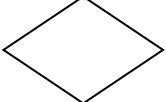

6.	Menggunakan / <i>include</i> / <i>uses</i>  	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .
----	---	--

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


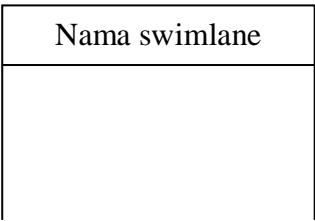
### 2.3.3 Activity Diagram

Menurut Rosa dan Shalahuddin (2016:161), “*Activity Diagram* adalah diagram yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak”.

**Tabel 2.2** Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	Status Awal  	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas  	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>decision</i>  	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>join</i>  	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.



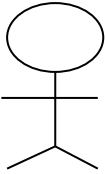
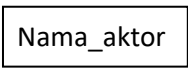
5.	Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber: Rosa dan Shalahuddin (2016:162-163))





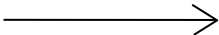
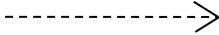
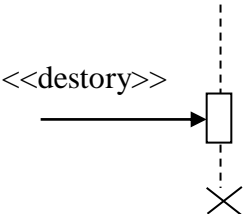
### 2.3.4 Sequence Diagram

Menurut Rosa dan Shalahuddin (2016:165), “*Sequence diagram* adalah diagram yang menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu daur hidup objek dan message yang dikirimkan dan diterima antar objek”.

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

No	Simbol	Deskripsi
1.	Aktor  Atau  tanpa waktu aktif	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 <i>actor</i>



2.	<p>Garis hidup/ <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek
3.	<p>Objek</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>nama objek : nama kelas</p> </div>	Menyatakan objek yang berinteraksi pesan
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
5.	<p>pesan tipe <i>create</i></p> <p style="text-align: center;">&lt;&lt;create&gt;&gt;</p> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	<p>Pesan tipe <i>call</i></p> <p>1 : nama_metode()</p> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
7.	<p>Pesan tipe <i>send</i></p> <p>1 : masukan</p> 	Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	<p>Pesan tipe <i>return</i></p> <p>1 : keluaran</p> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .






(Sumber: Rosa dan Shalahuddin (2016:165-167))

### 2.3.5 Class Diagram

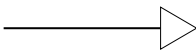
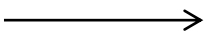
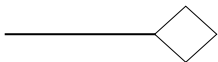
Menurut Rosa dan Shalahuddin (2018:141), “Class Diagram atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi”.

Menurut Tohari (dikutip Tabrani dan Aghniya, 2019:43), mendefinisikan bahwa "Kelas (*class*) adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan perancangan berorientasi objek".

**Tabel 2.4** Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <div style="border-bottom: 1px solid black; padding: 2px 5px;">nama_kelas</div> <div style="border-bottom: 1px solid black; padding: 2px 5px;">+ atribut</div> <div style="padding: 2px 5px;">+ operasi ()</div> </div>	Kelas pada struktur sistem
2.	<p style="text-align: center;"><i>Antarmuka / interface</i></p> <div style="text-align: center;">  <p>nama_interface</p> </div>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3.	<p style="text-align: center;"><i>Asosiasi / association</i></p> <div style="text-align: center;">  </div>	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	<p style="text-align: center;"><i>Asosiasi berarah / directed association</i></p> <div style="text-align: center;">  </div>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .



5.	<p style="text-align: center;">Generalisasi</p> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	<p style="text-align: center;">Kebergantungan/ <i>dependency</i></p> 	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.	<p style="text-align: center;">Agregasi / <i>aggregation</i></p> 	Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> )

(Sumber: Rosa dan Shalahuddin (2016:144-147))

### 2.3.6 Kamus Data

Menurut Rosa dan Shalahuddin (2018:141), mengungkapkan bahwa “Kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD yang berupa kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Menurut Puspita dan Aggadini (dikutip Prastianto dan Rostianti, 2020:26), “Kamus data (*data dictionary*) adalah suatu penjelasan tertulis mengenai data yang ada didalam database. Kamus data pertama berbasis dokumen, kamus data itu tersimpan berbentuk hard copy dengan mencatat semua penjelasan data dalam bentuk tercetak”.

Kamus data memiliki beberapa symbol untuk menjelaskan informasi tambahan sebagai berikut :

**Tabel 2.5** Simbol-simbol pada Kamus Data

No	Simbol	Keterangan
1	=	Disusun atau terdiri dari
2	+	Dan
3	[]	Baik...atau...
4	{ }"	N kali diulang/bernilai banyak
5	()	Data opsional
6	*...*	Batas komentar

(sumber : Rosa dan Shalahuddin (2018:74))

## 2.4 Teori Program

### 2.4.1 Pengertian *HTML (Hyper Text Markup Language)*

Menurut Yuda dan Faris (2023:43), "*HTML (Hypertext Markup Language)* yaitu sebuah intruksi pemrograman yang dituliskan dalam bentuk baris-baris kode, yang dikenal dengan istilah Tag. Script *HTML* dikenal hanya mampu menghasilkan sebuah website dengan output yang sederhana, baik dari segi tampilan maupun kapabilitas yang dimilikinya. Maintenance terhadap halaman web (seperti menambah dan meremajakan isi website), hanya dapat dilakukan dengan mengubah script *HTML* yang ada dalam program".

Menurut Zainy, et al (2022:336), "*HTML* adalah salah satu Bahasa pemrograman yang digunakan dalam pembuatan sebuah website".

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa *HTML* merupakan sebuah dokumen yang berfungsi untuk membentuk halaman website.

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

Menurut Betha Sidik (dikutip Noviyanti et al., 2021:70), "*PHP* merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen *HTML* secara on the fly yang dieksekusi di *server web*, dokumen *HTML* yang dihasilkan dari suatu aplikasi bukan dokumen *HTML* yang dibuat dengan menggunakan editor teks atau editor *HTML*".





Menurut Simare dan Yana (2022:72), "*PHP* adalah bahasa pemrograman *script server side* yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan web. Bahasa pemrograman ini memang dirancang untuk para pengembang web agar dapat menciptakan suatu halaman *web* yang bersifat dinamis".

Berdasarkan pendapat di atas dapat disimpulkan bahwa *PHP* adalah bahasa pemrograman yang berbentuk *script* atau perintah-perintah yang dapat dijalankan pada teknologi web.

#### **2.4.3 Pengertian CSS (*Cascading Style Sheet*)**

Menurut Saputra (dikutip Handayani et al., 2019:34), "Kepanjangan dari CSS adalah *Cascading Style Sheet* yang merupakan suatu bahasa pemrograman suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur, dan seragam".

Menurut Kurniawan (dikutip Arizha et al., 2019:52), menjelaskan bahwa "CSS atau *Cascading Style Sheets* adalah sebuah dokumen yang berisi aturan yang digunakan untuk memisahkan isi dengan layout dalam halaman-halaman web yang dibuat".

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa CSS merupakan dokumen web yang berfungsi untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan beragam.

#### **2.4.4 Pengertian MySQL (*MY Structure Query Language*)**

Menurut Tebai dan Wijanarko (2022:119), "*MySQL* merupakan pemrograman / sistem manajemen database (kumpulan data yang terstruktur) yang menggunakan basis Bahasa *SQL (Structured Query Language)*".

Menurut Adi Nugroho (dikutip Renaldy dan Rustam, (2022:29), "*MySQL (My Structured Query Language)* adalah suatu sistem basis data relation atau *Relational Database Management System (RDBMS)* yang mampu bekerja secara cepat dan mudah digunakan oleh banyak user".



Berdasarkan pendapat di atas, dapat disimpulkan bahwa MySQL adalah salah satu jenis Database Management System (DBMS) yang digunakan untuk membuat aplikasi berbasis web dinamis yang bersifat open source.

#### **2.4.5 Pengertian XAMPP**

Menurut Imam Mulhim (dikutip Tumini dan Fitria, 2021:14), "XAMPP adalah pake instalasi program yang terdiri atas program *apache HTTP Server*, *MySQL*, *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *perl*.

Menurut Jubilee Enterprise (dikutip Tumini dan Fitria, 2021:14), "XAMPP adalah server yang paling banyak digunakan untuk keperluan belajar *PHP* secara mandiri, terutama bagi programmer pemula".

Dari definisi di atas penulis menyimpulkan bahwa *XAMPP* adalah paket kumpulan software lengkap berfungsi untuk memudahkan instalasi *PHP* bagi para pengembang web yang di dalamnya terkandung *Web Server Apache*, *database MySQL*, dan *PHP interpreter*.

#### **2.4.6 Pengertian Visual Studio Code**

Menurut Agustini dan Kurniawan (2019:155), "Visual Studio Code adalah kode editor sumber yang dikembangkan oleh Microsoft untuk Windows, dan macOS. Termasuk dukungan untuk *debugging* kontrol git yang tertanam dan *Github*, penyorotan sintaksis, penyelesaian kode cerdas, snippet, dan *refactoring* kode. Ini sangat dapat disesuaikan, pengguna untuk mengubah tema, pintasan keyboard, preferensi, dan menginstal ekstensi fungsionalitas tambahan".