



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perangkat Lunak

Menurut Rianto, Indra (2021:5), mendefinisikan bahwa “*Software* (Perangkat Lunak) merupakan suatu data yang diprogram sedemikian rupa dan disimpan dalam bentuk digital yang tidak terlihat secara fisik tetapi tersimpan dalam media penyimpanan komputer.”

Menurut Sukamto dan Shalahuddin (2013:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumen perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Berdasarkan beberapa definisi perangkat lunak diatas, dapat disimpulkan bahwa perangkat lunak atau *software* adalah instruksi-instruksi atau data yang diformat secara digital, yang bisa dibaca dan ditulis oleh komputer sesuai kehendak pemakai.

2.1.2 Pengertian Komputer

Menurut Rianto, Indra (2021:3), “Komputer merupakan suatu perangkat elektronika yang memiliki kemampuan untuk menerima dan mengolah data menjadi informasi, menjalankan program yang tersimpan dalam memori, serta dapat bekerja secara otomatis berdasarkan perangkat aturan tertentu.”

Menurut Prawiro (dalam Harmayani, dkk, 2021:1), “Komputer adalah perangkat elektronik yang dapat digunakan untuk mengolah data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga menghasilkan informasi bermanfaat bagi penggunanya.”



Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa komputer adalah alat elektronik yang dapat mengolah data digital, dengan mengikuti serangkaian program yang digunakan untuk membantu pelaksanaan pekerjaan.

2.1.3 Pengertian *Internet*

Menurut Lubis, dkk (2020:16), “*Internet (Interconnecting Network)* adalah kumpulan komputer yang saling berhubungan dalam suatu jaringan. Disebut internet karena dapat menghubungkan komputer dengan jaringan komputer yang lebih luas di seluruh dunia.”

Menurut Arif, Muh dan Eby Waskito Makalalag (2020:108), “*Internet* adalah jaringan antar komputer di seluruh dunia yang dihubungkan melalui media telepon atau satelit yang dapat digunakan untuk menjalin komunikasi tanpa batas.”

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa *Internet* adalah jaringan global yang menghubungkan seluruh jaringan di dunia agar saling berinteraksi dan melakukan pertukaran informasi.

2.1.4 Pengertian *Data*

Menurut Irfannuddin (2019:117), “Data merupakan informasi yang bersifat mentah atau tidak teratur dalam bentuk satuan atau kumpulan angka, huruf, atau simbol mengenai suatu kondisi, ide, atau objek.”

Menurut Tofik, dkk (2022:1), menyatakan bahwa “Data merupakan fakta-fakta yang belum diolah atau deskripsi dasar dari aktivitas ataupun transaksi yang direkam, diklasifikasikan disimpan, namun tidak diorganisasikan untuk suatu tujuan yang spesifik.”

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa data adalah kumpulan atas angka, huruf, dan fakta yang akan diklasifikasi, dimodifikasi, dan diolah oleh program agar dapat menjadi informasi yang dapat berguna bagi penggunanya.



2.1.5 Pengertian Basis Data (*Database*)

Menurut Fitri, Rahmi (2020:1), “Basis data adalah kumpulan data yang terorganisir, yang umumnya disimpan dan diakses secara elektronik dari suatu sistem komputer.”

Menurut Anggaro, dkk (2021:1), menyatakan bahwa “Basis data (*database*) merupakan sekumpulan data yang saling berhubungan dan disimpan dalam komputer secara sistematis dan mempunyai arti secara implisit serta dapat diolah dan diperiksa.”

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa basis data (*database*) adalah kumpulan data yang saling berhubungan, terorganisir secara sistematis, menggunakan program komputer agar dapat dimanipulasi dan diperiksa dengan cepat dan mudah.

2.1.6 Metode Pengembangan Sistem

Penelitian Laporan Akhir ini menggunakan metode pengembangan sistem Waterfall. Menurut Pressman (dikutip Sanubari, dkk, 2020:41), menyatakan bahwa “Metode air terjun atau yang sering disebut metode waterfall sering dinamakan siklus hidup klasik (*classic life cycle*), 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 (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan.”

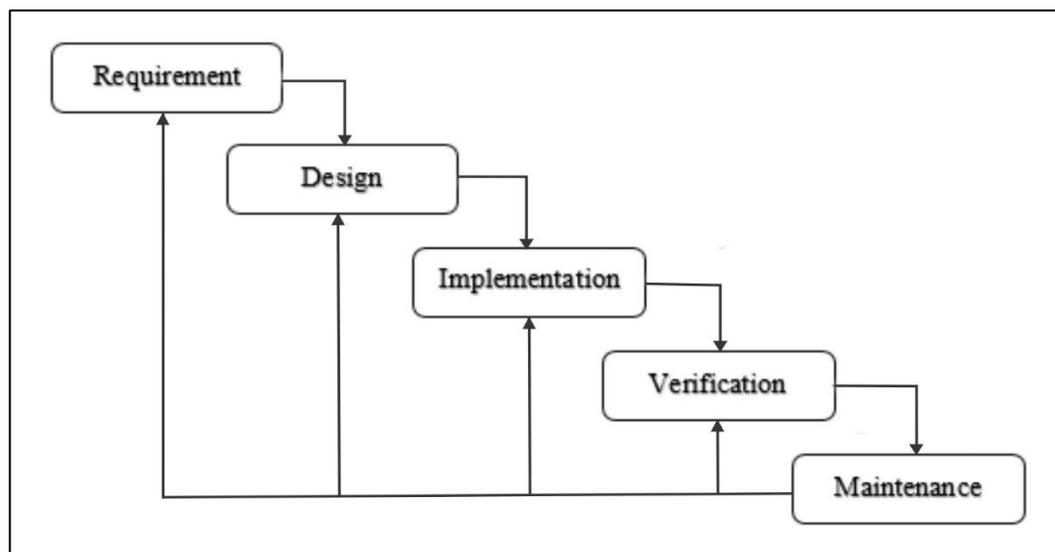
Menurut Widiyanto (2019), Metode Waterfall ini biasa disebut dengan “siklus hidup klasik” atau yang sekarang disebut model air terjun. Metode ini mengambil pendekatan yang terstruktur dan tersistematis mulai dari tingkat kebutuhan sistem hingga ke tahap pemeliharaan. Disebut air terjun karena seperti air terjun yang jatuh satu per satu, model ini menggunakan penyelesaian tahap sebelumnya baru kemudian dapat dilanjutkan ke tahap berikutnya. Menurut



Widianto, Mohammad Haldi (2019), Metode Waterfall ini biasa disebut dengan “siklus hidup klasik” atau yang sekarang disebut model air terjun. Metode ini mengambil pendekatan yang terstruktur dan tersistematis mulai dari tingkat kebutuhan sistem hingga ke tahap pemeliharaan. Disebut air terjun karena seperti air terjun yang jatuh satu per satu, model ini menggunakan penyelesaian tahap sebelumnya baru kemudian dapat dilanjutkan ke tahap berikutnya.

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*, *Verification & Testing*, dan *Maintenance*.

Tahapan metode waterfall dapat dilihat pada gambar di bawah ini:



Sumber : Sanubari, dkk (2020:41)

Gambar 2.1 Tahapan-Tahapan Metode Waterfall

Berdasarkan gambar di atas, pengembangan sistem dengan menggunakan metode waterfall memiliki beberapa tahapan yang berurutan yaitu: *Requirement* (analisis kebutuhan), *Design System* (desain sistem), *Coding* (pengkodean) & *Testing* (pengujian), Penerapan Program, Pemeliharaan. Tahapan-tahapan dari metode waterfall adalah sebagai berikut:



a. *Requirement Analisis*

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

b. *System Design*

Spesifikasi kebutuhan tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem dipersiapkan. Desain Sistem membantu dalam menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

c. *Implementation*

Tahap ini dimana 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*.

d. *Integration & Testing*

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.

e. *Operation & Maintenance*

Operation & Maintenance merupakan tahap akhir dalam model waterfall. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaiki implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

Sumber : Sanubari, dkk (2020:41-42).



2.1.7 Studi Kelayakan

Menurut Sutrisno dikutip oleh Suprpto dalam Jurnal nya (Pemahaman Umum Studi Kelayakan Proyek *Infrastuktur*) mengemukakan bahwa Pengertian 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. Objek atau subjeck materi studi kelayakan adalah usulan proyek/gagasan usaha. Usulan proyek/gagasan usaha tersebut dikaji, diteliti, dan diselidiki dari berbagai aspek tertentu apakah memenuhi persyaratan untuk dapat berkembang atau tidak. Dalam studi kelayakan yang distudi (diteliti) misalnya aspek pemasaran, aspek tehnik, aspek proses termasuk input, out put dan pemasaran, aspek komersial, aspek yuridis, aspek social budaya, aspek paedagogis dan aspek ekonomi. Sedangkan Menurut Yacob Ibrahim dikutip oleh Suprpto dalam Jurnal nya (Pemahaman Umum Studi Kelayakan Proyek *Infrastuktur*) mengemukakan bahwa Studi Kelayakan (*feasibility study*) adalah kegiatan untuk menilai sejauh mana manfaat yang dapat diperoleh dalam melaksanakan suatu kegiatan usaha /proyek dan merupakan bahan pertimbangan dalam mengambil suatu keputusan, apakah menerima atau menolak dari suatu gagasan usaha /proyek yang direncanakan. Pengertian layak dalam penilaian ini adalah kemungkinan dari gagasan usaha/proyek yang akan dilaksanakan memberikan manfaat (*benefit*), baik dalam arti *financial benefit* maupun dalam arti *social benefit*.

Dari kedua pendapat tentang pengertian Studi Kelayakan diatas dapatlah disimpulkan bahwa Studi kelayakan adalah kegiatan menganalisa, mengkaji dan menelilti berbagai aspek tertentu suatu gagasan usaha/proyek yang akan dilaksanakan atau telah dilaksanakan, sehingga memberi gambaran layak (*feasible-go*) atau tidak layak (*no feasible-no go*) suatu gagasan usaha/proyek apabila ditinjau dari manfaat yang dihasilkan (*benefit*) dari proyek/gagasan usaha tersebut baik dari sudut *financial benefit* maupun *social benefit*.



2.1.8 Kebutuhan Fungsional

Menurut Hadiprakoso, Raden Budiarto (2020:88), Kebutuhan Fungsional mendefinisikan fungsi atau *fitur* dari sebuah perangkat lunak, fungsi tersebut dapat berupa input dan output, proses perhitungan, rincian teknis, manipulasi dan pengolahan data atau fungsi spesifik lainnya yang seharusnya dipenuhi oleh sistem.

2.1.9 Kebutuhan Non-Fungsional

Menurut Hadiprakoso, Raden Budiarto (2020:88), Kebutuhan Non-Fungsional merupakan batasan pada layanan atau fungsi yang ditawarkan oleh sistem (Kebutuhan Non-Fungsional merupakan syarat pendukung). Batasan ini menyangkut masalah kinerja, keandalan atau standar tertentu.

2.2 Teori Khusus

2.2.1 Kamus Data

Menurut Syam, dkk (2021:201), mengungkapkan bahwa “Kamus Data adalah kumpulan daftar elemen data yang mengalir pada sistem sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum artinya memiliki standard cara penulisan.”

Menurut Budiarto, Raden (2021:72), mendefinisikan bahwa “Kamus data (*data dictionary*) berisi semua atribut nama dan karakteristik untuk setiap tabel dengan kata lain kamus data berisi meta data tabel-tabel dalam basis data.”

Berdasarkan beberapa pengertian di atas, dapat disimpulkan bahwa Kamus Data (*data dictionary*) merupakan katalog yang berisi kumpulan fakta data agar proses *input* dan *output* pada sistem perangkat lunak dapat dengan mudah dipahami.

Kamus data biasanya berisi :

- a. Nama-nama dari data
- b. Digunakan pada proses-proses yang terkait data



- c. Deskripsi data
- d. Informasi tambahan seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

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

Tabel 2.2 Notasi Aturan Penulisan Kamus Data

No.	Simbol	Keterangan
1.	=	sama dengan atau terdiri dari atau terbentuk dari
2.	+	Dan
3.	[]	pilih salah satu
4.	{ }	Literasi atau pengulangan
5.	()	(pilihan) option
6.	*	Komentar
7.		Pemisah

Sumber : Subakti, dkk (2022:61).

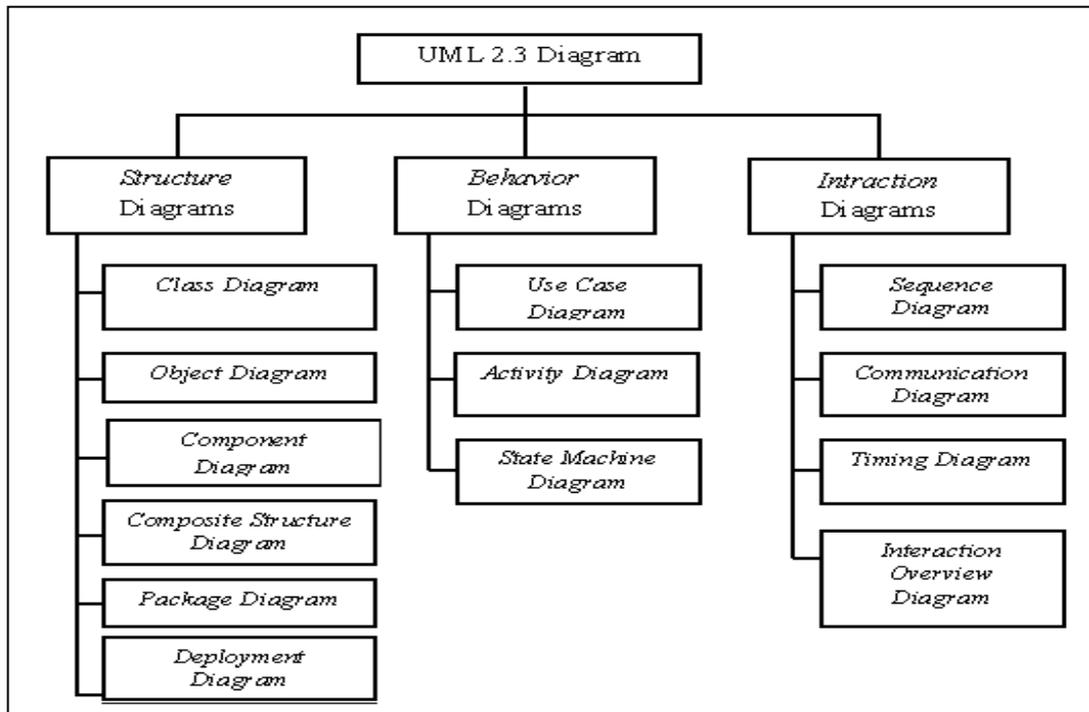
2.2.2 *Unified Modeling Language (UML)*

2.2.2.1 *Pengertian Unified Modeling Language (UML)*

Menurut Pressman dalam Sa'ad (2020:46), "UML adalah sebuah patokan bahasa untuk menulis kerangka kerja terperinci dari sebuah perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membangun dan mendokumentasikan sebuah sistem perangkat lunak."

2.2.2.2 *Kategori Unified Modeling Language (UML)*

Unified Modeling Language (UML) terdiri dari 13 macam diagram dan dikelompokkan menjadi 3 kategori sebagai berikut:



Sumber : Sukanto dan Shalahuddin (2016:140).

Gambar 2.1 Kategori dan Macam-Macam Diagram UML

Berikut penjelasan singkat dari pembagian kategori pada diagram UML menurut Sukanto dan Shalahuddin (2016:141):

- 1) *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- 2) *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- 3) *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

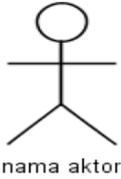


2.2.3 Jenis-Jenis Diagram *Unified Modeling Language* (UML)

2.2.3.1 Pengertian *Use Case Diagram*

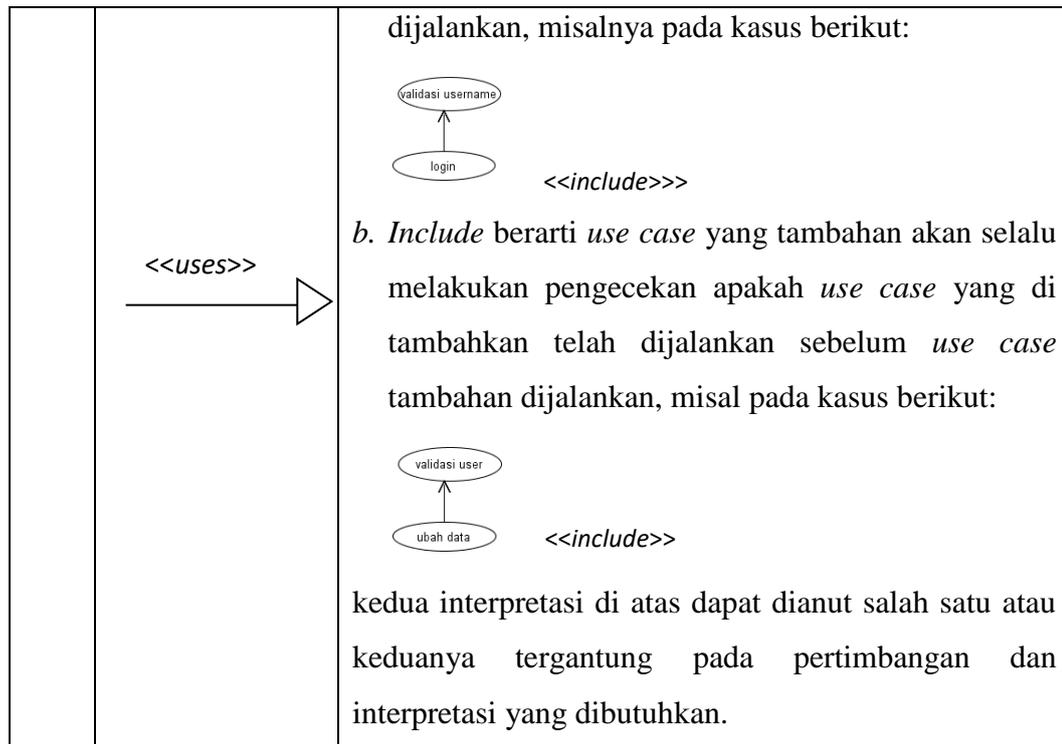
Menurut Pressman dalam Sa'ad (2020:46), "Use Case Diagram membantu menentukan fungsi dan fitur dari perangkat lunak. Dalam diagram ini, gambar yang menyerupai boneka kayu mewakili aktor yang berhubungan dengan kategori dari pengguna. Di dalam diagram *Use Case*, para aktor terhubung oleh garis ke *Use Case* yang mereka kerjakan."

Tabel 2.3 Simbol-Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.	<p><i>Use case</i></p> 	<p>fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal-awal frase nama <i>use case</i>.</p>
2.	<p>aktor / <i>actor</i></p> 	<p>orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi 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 actor.</p>
3.	<p>asosiasi/ <i>association</i></p> 	<p>komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.</p>

Lanjutan Tabel 2.3 Simbol-Simbol *Use Case Diagram*

No	Simbol	Deskripsi
4.	<p>ekstensi / <i>extend</i></p> <p>.....→ <<extend>></p>	<p>relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang di tambahkan 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 sama dengan <i>use case</i> yang ditambahkan, misalnya</p> <pre> graph TD A([validasi usernama]) -- "<<extend>>" --> B([validasi user]) C([validasi sidik jari]) -- "<<extend>>" --> B </pre> <p>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</p>
5.	<p>Generalisasi / <i>generalization</i></p> <p>————→▷</p>	<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,</p> <pre> graph TD A([ubah data]) --> B([mengolah data]) C([hapus data]) --> B </pre> <p>misalnya: arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).</p>
6.	<p>menggunakan / <i>include</i> / <i>uses</i></p> <p>.....→ <<include>></p>	<p>relasi 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</p> <p>ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <p>a. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan</p>



Sumber : Sukamto dan Shalahuddin (2016:156-158).

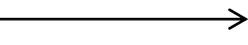
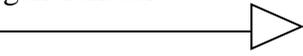
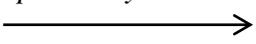
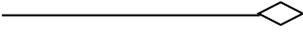
2.2.3.2 Pengertian Class Diagram

Menurut Pressman dalam Sa'ad (2020:49), "Unsur-unsur utama dari diagram kelas adalah kotak, yang merupakan ikon yang digunakan untuk mewakili kelas dan *interface*. Setiap kotak dibagi menjadi bagian-bagian *horizontal*. Bagian atas berisi nama kelas, bagian tengah berisi daftar atribut kelas dan bagian tengah bawah merupakan *operation* dari kelas tersebut."

Tabel 2.4 Simbol-Simbol Class Diagram

No	Simbol	Deskripsi
1.	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>nama_kelas</p> <hr/> <p>+atribut</p> <hr/> <p>+operasi()</p> </div>	Kelas pada struktur sistem.

Lanjutan Tabel 2.4 Simbol-Simbol *Class Diagram*

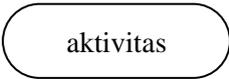
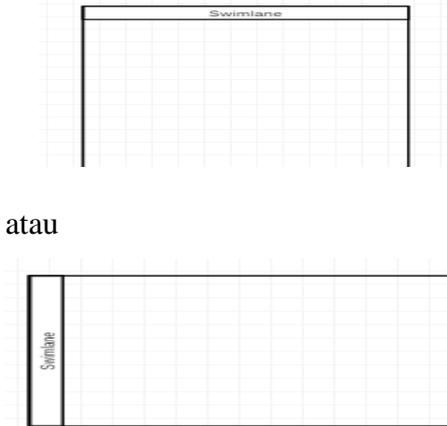
No	Simbol	Deskripsi
2.	antarmuka / <i>interface</i> 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i> .
4.	asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus).
6.	kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas.
7.	agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

Sumber : Sukamto dan Shalahuddin (2016:146-147).

2.2.3.3 Pengertian *Activity Diagram*

Menurut Pressman dalam Sa'ad (2020:51), “Sebuah diagram *activity* menggambarkan perilaku dinamis dari sistem atau bagian dari sistem melalui aliran kontrol antara tindakan yang sistem lakukan. Hal ini mirip dengan sebuah *flowchart* kecuali bahwa sebuah diagram *activity* dapat menunjukkan arus bersamaan.”

Tabel 2.5 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 di mana 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.	Swimlane  atau	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

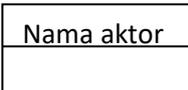
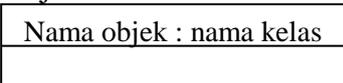
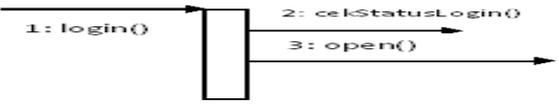
Sumber : Sukamto dan Shalahuddin (2016:162-163).



2.2.3.4 Pengertian Sequence Diagram

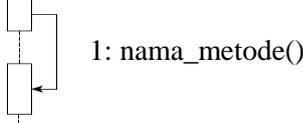
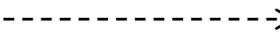
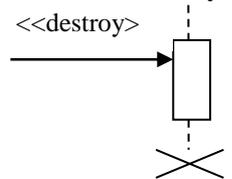
Menurut Pressman dalam Sa'ad (2020:52), "suatu diagram yang menggambarkan interaksi antar objek akan mengindikasikan komunikasi di antara objek-objek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang diperuntukkan oleh objek-objek yang melakukan suatu tugas atau aksi tertentu. Objek-objek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya di paling kiri dari diagram."

Tabel 2.6 Simbol-Simbol Sequence Diagram

No	Simbol	Deskripsi
1.	<p><i>Actor</i></p>  <p>nama aktor</p> <p>Atau</p>  <p>Nama aktor</p> <p>tanpa waktu aktif</p>	<p>orang, proses, atau sistem lain yang berinteraksi dengan 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>.</p>
2.	<p>Garis hidup / <i>lifeline</i></p> 	<p>menyatakan kehidupan suatu objek.</p>
3.	<p>Objek</p>  <p>Nama objek : nama kelas</p>	<p>menyatakan objek yang berinteraksi pesan.</p>
4.	<p>Waktu aktif</p> 	<p>menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p>  <p>1: login() 2: cekStatusLogin() 3: open()</p> <p>maka cek Status Login () dan open() dilakukan di dalam metode login() aktor tidak memiliki waktu aktif.</p>



• Lanjutan Tabel 2.6 Simbol-Simbol *Sequence* Diagram

No	Simbol	Deskripsi
5.	Pesan tipe <i>create</i> 	menyatakan objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6.	Pesan tipe <i>call</i> 	menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, 
7.	Pesan tipe <i>send</i> 	menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	Pesan tipe <i>return</i> 	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.	Pesan tipe <i>destroy</i> 	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 : Sukamto dan Shalahuddin (2016:165-167).



2.3 Teori Judul

2.3.1 Pengertian Perancangan

Menurut Mohamadi,Loekman (2020:5), "Perancangan adalah suatu sistem yang berlaku untuk segala macam jenis perancangan dimana titik beratnya adalah melihat suatu persoalan tidak secara terpisah atau tersendiri, melainkan sebagai suatu kesatuan di mana satu masalah dengan lainnya saling kait mengkait."

Menurut Habibi,Roni dkk(2020:111), menyatakan bahwa "Perancangan adalah merancang serta memodelkan sebuah aplikasi maupun sebuah sistem dengan baik dalam artian sesuai dengan tujuan yang diinginkan yang isinya adalah langkah-langkah operasi dalam proses pengolahan data dan juga prosedur-prosedur untuk mendukung operasi sistem tersebut."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa perancangan adalah suatu sistem rekayasa rancang bangun untuk memberikan gambaran secara umum dan fungsionalitas dari aplikasi yang akan dibuat kepada pemakai.

2.3.2 Pengertian Aplikasi

Menurut Zahro,Aftalin dan Cahyo Hasanudin (2022:402), menyatakan bahwa "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."

Menurut Yusuf, dkk (2019:244), mendefinisikan bahwa "Aplikasi merupakan penerapan, menyimpan sesuatu hal, data permasalahan, pekerjaan kedalam suatu sarana atau media yang digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan tersebut sehingga berubah menjadi bentuk yang baru tanpa menghilangkan nilai-nilai dasar hal, data, permasalahan atau permasalahan atau pekerjaan."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa aplikasi merupakan perangkat lunak (*software*) yang dibuat untuk menolong beberapa aktivitas manusia dalam melaksanakan tugas tertentu.



2.3.3 Pengertian Penjadwalan

Menurut Cahyadi, dkk (2021:32) menyatakan bahwa "Penjadwalan merupakan fungsi pengambilan keputusan yang meliputi kegiatan pengurutan pekerjaan (*sequencing*), waktu mulai dan selesainya pekerjaan (*timing*), serta urutan proses suatu proyek pekerjaan (*routing*)."

Menurut Hutahean, Wendy Sepmady (2018:39) "Penjadwalan adalah pengaturan waktu dari suatu kegiatan operasi, yang mencakup kegiatan mengalokasikan fasilitas, peralatan, maupun tenaga kerja, dan menentukan urutan pelaksanaan bagi suatu kegiatan operasi."

Berdasarkan beberapa definisi diatas, maka dapat disimpulkan bahwa penjadwalan adalah suatu proses pengambilan keputusan yang mencakup urutan pelaksanaan bagi suatu kegiatan operasi.

2.3.4 Pengertian Patroli

Menurut Peraturan Kabaharkam Polri No. 4 Tahun 2011 tentang Patroli Pasal 1 angka (3) pengertian patroli adalah Salah satu kegiatan kepolisian yang dilakukan 2 (dua) orang atau lebih anggota Polri, sebagai usaha mencegah bertemunya niat dan kesempatan, dengan jalan mendatangi, menjelajahi, mengamati/ mengawasi/ memperhatikan situasi dan kondisi yang diperkirakan akan menimbulkan segala bentuk pelanggaran dan/ atau tindak pidana, yang menuntut/memerlukan kehadiran Polri untuk melakukan tindakan-tindakan kepolisian guna memelihara ketertiban dan menjamin keamanan masyarakat.

Dengan demikian, dapat disimpulkan bahwa patroli adalah suatu bentuk kegiatan bergerak dari suatu tempat ke tempat tertentu yang dilakukan oleh anggota Sabhara guna mencegah terjadinya suatu tindak kriminal, memberikan rasa aman, perlindungan dan pengayoman kepada masyarakat.



2.3.5 Pengertian Pengawasan

Menurut Poernomo, Freddy (2020:124), "Pengawasan adalah tindakan untuk memeriksa apakah sesuatu telah dilaksanakan sesuai dengan rencana, perintah, dan prinsip-prinsip yang telah ditetapkan".

Menurut Chandra, M. Jeffri Arlides Chandra dan JT. Pareke (2018:73), menyatakan bahwa "Pengawasan adalah suatu proses untuk menetapkan pekerjaan apa yang sudah dilaksanakan, menilainya dan mengoreksi bila perlu dengan maksud supaya pelaksanaan pekerjaan sesuai dengan rencana semula."

Berdasarkan penjelasan diatas dapat disimpulkan bahwa Pengawasan adalah suatu proses untuk menetapkan pekerjaan yang telah dilaksanakan sesuai dengan rencana.

2.3.6 Pengertian Website

Menurut Elgamar (2020:3), "*Website* merupakan sebuah media yang memiliki banyak halaman yang saling terhubung (*hyperlink*), dimana *website* memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara dan animasi atau penggabungan dari semuanya."

Menurut Adelheid, Andrea (2013:1), "*Website* adalah kumpulan dari halaman-halaman situs yang terangkum dalam sebuah domain atau sub domain yang tempatnya berada dalam world wide di dalam internet."

Berdasarkan penjelasan diatas dapat disimpulkan bahwa *Website* adalah halaman yang berisi informasi berupa teks, gambar, suara, dan video yang dapat disimpan pada komputer lokal maupun *server web*, yang terhubung melalui link *hyperlink* dan dapat dijadikan atau ditampilkan pada browser.

2.3.7 Pengertian Aplikasi Penjadwalan Patroli dan Pengawasan Berbasis Website Pada Polres Ogan Ilir

Aplikasi Penjadwalan Patroli dan Pengawasan Berbasis Website adalah aplikasi untuk memudahkan intansi pemerintahan dalam melakukan proses pengelolaan pencatatan penjadwalan patroli yang lebih terstruktur, efektif, dan



efisien serta memudahkan pegawai dalam proses pembagian tugas anggota. Dengan adanya aplikasi ini, intansi dan pegawai dapat melakukan proses pengelolaan pencatatan penjadwalan patroli menggunakan koneksi internet yang dimana aplikasi ini mempunyai *Security* yang aman. Intansi dan pegawai pun dapat mengetahui mengenai penjadwalan kegiatan patroli dan pembagian tugas anggota dengan mudah.

2.4 Teori Program

2.4.1 Pengertian HTML

Menurut Faisal, M Reza dan Friska Abadi (2020:12), menyatakan bahwa "*HyperText Markup Language (HTML)* adalah bahasa yang digunakan untuk membuat antarmuka halaman *web*."

Menurut Devi, Merysa Arsta (2020:1), "*Hypertext Markup Language (HTML)* adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi dalam sebuah penjelajah web internet dan memformat hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang teritegrasi."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa HTML (*Hypertext Markup Language*) adalah skrip yang berupa tag-tag untuk membuat dan mengatur struktur serta menandai bagian-bagian dari sebuah halaman website untuk menampilkan konten di web.

2.4.2 Pengertian JavaScript

Menurut Azis, dkk (2019:10), "Javascript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak hanya indah saja."



Menurut Marisa, Fitri (2016:42), "Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa JavaScript adalah bahasa pemrograman web yang pemrosesannya dilakukan di sisi client untuk memberikan efek dinamis, interaktif dan bersifat independen, yang dieksekusi di browser.

2.4.3 Pengertian CSS

Menurut Rerung, Rintho Rante (2018:133), "CSS adalah singkatan dari *Cascading Style Sheets*, CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi dimana HTML ditujukan untuk membuat struktur, atau konten dari halaman web."

Menurut Surya, Candra dan Miftahul Jannah (2020:94), "Cascading Style Sheets disingkat CSS merupakan suatu kumpulan intruksi yang digunakan untuk mengatur elemen-elemen HTML."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa CSS adalah bahasa pengkodean atau dokumen yang berfungsi mengatur elemen HTML dengan berbagai *property* untuk menata gaya tampilan halaman web agar lebih cantik dan indah saat di tampilkan di web browser.

2.4.4 Pengertian PHP

Menurut Oetomo, Hening Widi dan Pontjo Bambang Mahargiono (2020:1), menyatakan bahwa "PHP adalah singkatan dari *Hypertext Preprocessor*, yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan, pembuatan, dan pengembangan sebuah situs web dan biasanya digunakan bersamaan dengan HTML."

Menurut Sandi, dkk (2020:57), menyatakan bahwa "*Hypertext Preprocessor* atau disingkat PHP adalah bahasa pemrograman berbasis web yang ditulis oleh dan untuk pengembang web."



Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa PHP adalah bahasa pemrograman web berbasis *server-side programming* (bahasa pemrograman yang diproses disisi server).

2.4.5 Pengertian Laravel

Menurut Nurhidayah, dkk (2020:13), mendefinisikan bahwa "Laravel adalah *framework* backend yang menggunakan bahasa pemrograman PHP yang memungkinkan anda memberikan aplikasi luar biasa."

Menurut Harianto, dkk (2019:14), mendefinisikan bahwa "Laravel merupakan sebuah *framework PHP* yang akan membantu para web developer untuk menciptakan aplikasi istimewa secara simple, kode yang bersih dan menyenangkan."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa Laravel adalah sebuah *framework* yang bersifat *open source* untuk membangun *website* / aplikasi dengan menggunakan bahasa pemrograman PHP.

2.4.6 Pengertian XAMPP

Menurut Setyawan, M.Yusril Helmi dan Dinda Ayu Pratiwi (2019:70), menyatakan bahwa "XAMPP adalah perangkat lunak komputer yang sistem penamaannya diambil dari akronim kata apache, MySQL atau MariaDB, PHP, dan Perl."

Menurut Roza, dkk (2020:82), menyatakan bahwa "XAMPP adalah perangkat lunak bebas, yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program."

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa XAMPP merupakan aplikasi pendukung bersifat *open source* yang menggabungkan aplikasi ke dalam satu paket yaitu *Apache*, *MySQL*, *6* dan *PHP my admin* untuk membaca file yang memiliki format PHP.



2.4.7 Pengertian MySQL

Menurut Fitri, Rahimi (2020:2), "MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data."

Menurut Indrawan, Gede dan I Nyoman Yoga Setyawan (2018:3), "MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya sangat cepat, multi user serta menggunakan perintah dasar SQL (*Structured Query Language*)."

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa MySQL adalah *database management system* sebagai bahasa interaktif penghubung perangkat aplikasi dengan *database server* yang digunakan dalam melakukan pengelolaan data.

2.4.8 Pengertian Visual Studio Code

Menurut Habibi, Roni dan Alwan Suryansah (2020:62), "*Visual Studio Code (VS Code)* ini adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows".

Menurut Saputra, M.Harry K dan Fadila (2020:48), menyatakan bahwa "Aplikasi *Visual Studio Code* merupakan sebuah editor kode sumber yang dikembangkan oleh Microsoft untuk Windows, Linux, dan macOS".

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Visual Studio Code* adalah sebuah aplikasi editor yang digunakan pada kode dan teks yang dikembangkan oleh Microsoft, Linux, dan macOS.