



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sanders (dalam Wahyudin dan Munir, 2018:1) mengatakan, “Komputer adalah sistem elektronik untuk memanipulasi data yang cepat dan tepat serta dirancang dan diorganisasikan supaya secara otomatis menerima dan menyimpan data input, memprosesnya, dan menghasilkan output berdasarkan instruksi-instruksi yang telah tersimpan di dalam memori”.

Menurut Hamacher, Vranesic dan Zaky dalam Sindu dan Paramartha (2018:2) “Komputer didefinisikan sebagai sebuah mesin penghitung elektronik yang cepat dapat menerima informasi input digital, memprosesnya sesuai dengan suatu program yang tersimpan di memorinya (*stored program*) dan menghasilkan *output* informasi”.

2.1.2 Pengertian Perangkat Lunak

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

Menurut Pahul dalam Sindu dan Paramartha (2018:192), “*Software* atau perangkat lunak adalah kumpulan dari data-data elektronik yang diformat, disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca, dan ditulis oleh komputer”.

2.1.3 Pengertian Website

Menurut Yuhfizar dalam Sa’ad (2020:3), “*Website* merupakan keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah website biasanya dibangun atas banyak halaman web yang saling berhubungan”.



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

2.1.4 Pengertian Data

Menurut Kristanto (2018:7), “Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata”.

Dantes et.al. (2018:3) menuliskan, “Data adalah nilai/*value* yang dapat merepresentasikan deskripsi dari suatu objek atau kejadian (*event*). Objek yang dimaksud dapat berupa manusia (pegawai, siswa, pembeli, pelanggan), barang dan hewan”.

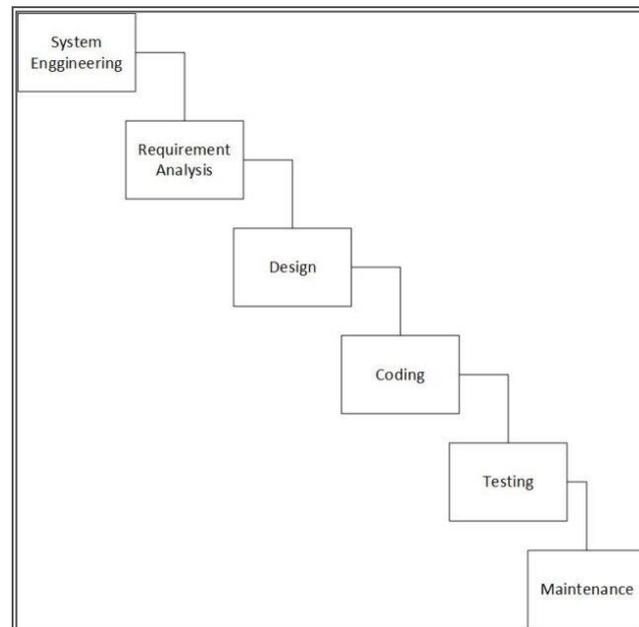
2.1.5 Pengertian Basis Data (Database)

Menurut Rosa A.S dan Shalahuddin (2018:43), “Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan” .

Menurut Dantes et.al. (2018:22), “Basis data adalah kumpulan data yang saling berhubungan. Basis data adalah kelompok data (arsip) yang saling berhubungan dan tanpa pengulangan (redundansi) yang disimpan dalam media elektronik dan diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan dengan cepat dan mudah”.

2.1.6 Metode Pengembangan Sistem Waterfall

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 terjatuh satu per satu, model ini menggunakan penyelesaian tahap sebelumnya baru kemudian dapat dilanjutkan ke tahap berikutnya.



Gambar 2.1 Model *Waterfall* menurut Widiyanto

Tahapan-tahapan dalam model ini menurut Widiyanto adalah sebagai berikut:

a. Rekayasa Sistem

Tahap ini dimulai dengan mencari kebutuhan seluruh sistem untuk diterapkan dalam perangkat lunak.

b. Analisis Kebutuhan

Pada tahap ini pengembang sistem membutuhkan komunikasi yang bertujuan memahami perangkat lunak yang diharapkan oleh pengguna dan keterbatasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung.

c. Desain

Spesifikasi persyaratan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan.

d. Pengodean

Pada fase ini, sistem terintegrasi dengan sintaks sehingga sistem informasi dapat digunakan sesuai kebutuhan, yang akan diintegrasikan pada tahap selanjutnya.



e. Pengujian

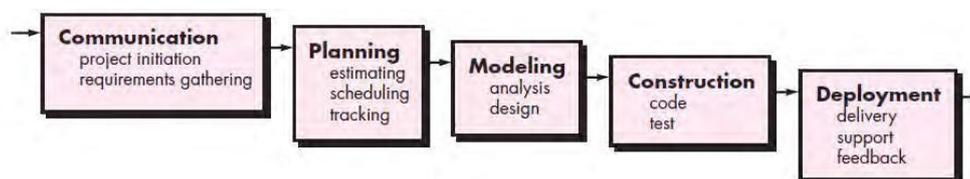
Semua unit yang telah dikembangkan dan pengkodean yang benar diuji langsung untuk penggunaannya, seperti menggunakan pengujian blackbox. Metode pengujian blackbox ini adalah pengujian yang hanya memperhatikan hasil masukan dan keluaran sistem.

f. Pemeliharaan

Fase terakhir dalam model air terjun. Perangkat lunak atau sistem informasi yang sudah jadi, dijalankan kemudian dipelihara, karena tidak selamanya sistem akan berjalan dengan baik. Maka, itulah fungsi pemeliharaan, guna menghindari sistem dari bug-bug yang timbul.

Menurut Pressman (2015:42), model waterfall adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Nama model ini sebenarnya adalah “Linear Sequential Model”. Model ini sering disebut juga dengan “classic life cycle” atau metode waterfall. Model ini termasuk ke dalam model generic pada rekayasa perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam Software Engineering (SE). Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

Fase-fase dalam Waterfall Model menurut referensi Pressman :



Gambar 2.2 Model Waterfall Menurut Pressman

a. Communication (Project Initiation & Requirements Gathering)

Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya



komunikasi dengan customer demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek, seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi software. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, dan internet.

b. Planning (Estimating, Scheduling, Tracking)

Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan tracking proses pengerjaan sistem.

c. Modeling (Analysis & Design)

Tahapan ini adalah tahap perancangan dan permodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur software, tampilan interface, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

d. Construction (Code & Test)

Tahapan Construction ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

e. Deployment (Delivery, Support, Feedback)

Tahapan Deployment merupakan tahapan implementasi software ke customer, pemeliharaan software secara berkala, perbaikan software, evaluasi software, dan pengembangan software berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya. (Pressman, 2015:17)



2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Idcloudhost (2020) menuliskan, “Aplikasi (lebih dikenal sebagai aplikasi) adalah perangkat lunak yang menggabungkan beberapa fitur tertentu dengan cara yang dapat diakses oleh pengguna”.

Menurut Rachmad Hakim S (2018), Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya.

Indrajani (2018:3), “Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk penyelesaian tugas-tugas khusus dari pemakai komputer”.

2.2.2 Pengertian Penanganan

Menurut Kamus Besar Bahasa Indonesia (KBBI) penanganan memiliki satu arti yaitu penanganan dan berasal dari kata dasar tangan. Penanganan memiliki arti yang menyatakan sebuah tindakan yang dilakukan dalam melakukan sesuatu. Penanganan juga dapat berarti proses, cara, perbuatan menangani sesuatu yang sedang dialami.

2.2.3 Pengertian Pasien

Pasien adalah orang yang memiliki kelemahan fisik atau mentalnya menyerahkan pengawasan dan perawatannya, menerima dan mengikuti pengobatan yang ditetapkan oleh tenaga kesehatan yang dikemukakan oleh Prabowo (dalam Wilhamda, 2011) . Sedangkan (Aditama, 2002) berpendapat bahwa pasien adalah mereka yang diobati dirumah sakit. Menurut (Soejadi, 1996) pasien adalah individu terpenting dirumah sakit.

2.2.4 Pengertian Framework

Menurut Hakim (2010:3) menjelaskan bahwa, *Framework* adalah koleksi atau kumpulan potongan-potongan program yang disusun atau diorganisasikan



sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal.

Sedangkan menurut Raharjo (2015:2), *Framework* adalah suatu kumpulan kode berupa pustaka (*library*) dan alat (*tool*) yang dipadukan sedemikian rupa menjadi satu kerangka kerja (*framework*) guna memudahkan dan mempercepat proses pengembangan aplikasi web.

2.2.5 Pengertian Klinik

Klinik merupakan salah satu bentuk perusahaan jasa yang memberikan jasa pelayanan kesehatan. Perusahaan jasa itu sendiri adalah perusahaan yang kegiatan utamanya memberikan pelayanan atau menjual jasa dengan tujuan mencari laba (Ahman dan Indriani, 2007).

Klinik adalah fasilitas pelayanan kesehatan yang menyelenggarakan dan menyediakan pelayanan medis dasar dan atau spesialisik, diselenggarakan oleh lebih dari satu jenis tenaga kesehatan dan dipimpin oleh seorang tenaga medis (Permenkes RI No.9, 2014) .

2.2.6 Pengertian Kantor

Menurut Kamus Besar Bahasa Indonesia (KBBI), Kantor adalah balai (gedung, rumah, ruang) tempat mengurus suatu pekerjaan atau disebut juga tempat bekerja.

menurut Nuraida (2008), kantor adalah tempat diselenggarakannya kegiatan tata usaha dimana terdapat ketergantungan sistem antara orang, teknologi dan prosedur untuk menangani data dan informasi mulai dari menerima, mengumpulkan, mengolah, menyimpan, sampai menyalurkannya.

2.2.7 Pengertian Kepolisian Resort

Sesuai dengan Kamus Besar Bahasa Indonesia (KBBI), arti kata polisi adalah suatu badan yang bertugas memelihara keamanan, ketentraman, dan ketertiban umum (menangkap orang yang melanggar hukum), merupakan suatu anggota



badan pemerintah (pegawai negara yang bertugas menjaga keamanan dan ketertiban).

Di Indonesia istilah “polisi” dikemukakan oleh salah satu pakar ilmu hukum yang bernama Dr.Sadjijono, menurut Sadjijono istilah “polisi” adalah sebagai organ atau lembaga pemerintah yang ada dalam negara, sedangkan istilah “Kepolisian” adalah sebagai organ dan sebagai fungsi. Sebagai organ, yakni suatu lembaga pemerintahan yang terorganisasi dan terstruktur dalam organisasi negara. Sedangkan sebagai fungsi, yakni tugas dan wewenang serta tanggungjawab lembaga atas kuasa undang-undang untuk menyelenggarakan fungsinya, antara lain memelihara keamanan dan ketertiban masyarakat, penegakan hukum, perlindungan, pengayom dan pelayan masyarakat.

Dari uraian-uraian tentang istilah “polisi” dan “kepolisian” di atas maka dapat dimaknai sebagai berikut: istilah polisi adalah sebagai organ atau lembaga pemerintah yang ada dalam negara. Sedangkan istilah Kepolisian sebagai organ dan fungsi. Sebagai organ, yakni suatu lembaga pemerintah yang terorganisasi dan terstruktur dalam ketatanegaraan yang oleh undang-undang diberi tugas dan wewenang dan tanggung jawab untuk menyelenggarakan kepolisian. Kepolisian Resort adalah struktur komando Kepolisian Republik Indonesia di daerah kabupaten/kota. Kepolisian Resort di wilayah perkotaan biasa disebut "Kepolisian Resort Kota". "Kepolisian Resort Kota Besar" biasanya digunakan untuk ibu kota provinsi. "Kepolisian Resort" dikepalai oleh seorang Kepala Kepolisian Resort.

2.3 Teori Khusus

2.3.1 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”.

Menurut Ariani R. Sukamto (2017), UML adalah bahasa visual dalam pemodelan dan komunikasi mengenai sebuah sistem yang disertai dengan penggunaan diagram teks-teks pendukung.



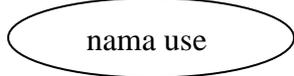
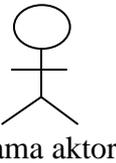
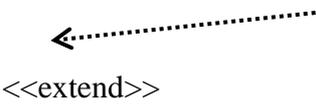
2.3.2 Jenis-Jenis Diagram UML

2.3.2.1 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 bonek 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.

Menurut Rosa dan Shalahuddin (2015: 155) Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat.

Tabel 2.1 Simbol-simbol *Use Case Diagram*

| No. | Simbol | Deskripsi |
|-----|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| 1 |  | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| 2 |  | Spesifikasi himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
| 3 |  | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 4 |  | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
| 5 |  | Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit. |



Lanjutan Tabel 2.1 Simbol-simbol *Use Case Diagram*

| | | |
|---|--------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| 6 | Asosiasi / <i>Association</i>  | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
|---|--------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------|

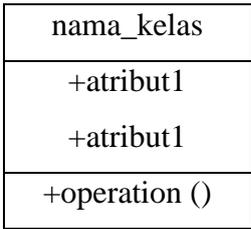
2.3.2.2 Pengertian Class Diagram

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.

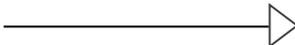
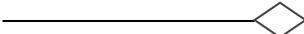
Menurut Rosa dan Shalahuddin (2015: 141) diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelaskelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut pola dan metode atau operasi :

- Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas
- Operasi atau metode adalah fungsi yang dimiliki oleh suatu kelas

Tabel 2.2 Simbol-simbol *Class Diagram*

| No | Simbol | Deskripsi |
|----|----------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| 1 | Kelas  | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 2 | antarmuka / <i>interface</i>  interface | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek. |
| 3 | Asosiasi  | Apa yang menghubungkan antara objek satu dengan objek lainnya. |

Lanjutan Tabel 2.2 Simbol-simbol *Class Diagram*

| | | |
|---|------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4 | Asosiasi berarah / directed association  | Relasi <i>antarclass</i> 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 | Ketergantungan / <i>dependency</i>  | Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>), akan mempengaruhi elemen yang bergantung padanya (elemen yang tidak mandiri). |
| 7 | Agregasi  | Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>). |

2.3.2.3 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.

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa dan Shalahuddin, 2015: 161).

Tabel 2.3 Simbol-simbol *Activity Diagram*

| No | Simbol | Nama | Deskripsi |
|----|-------------------------------------------------------------------------------------|-------------|----------------------------------------|
| 1. |  | Status awal | Bagaimana objek dibentuk atau diawali. |

Lanjutan Tabel 2.3 Simbol-simbol *Activity Diagram*

| | | | |
|---|--|------------------------|--------------------------------------------------------------------------------------|
| 2 | | Aktivitas | Memperlihatkan bagaimana kelas antarmuka saling berinteraksi satu sama lain. |
| 3 | | Percabangan / decision | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| 4 | | Penggabungan / join | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5 | | Status akhir | Bagaimana objek dibentuk dan dihancurkan. |
| 6 | | Swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

2.3.2.4 Pengertian Sequence Diagram

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

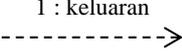
Menurut Rosa dan Shalahuddin (2015: 165) diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek.

Tabel 2.4 Simbol-simbol *Sequence Diagram*

| No | Simbol | Nama | Keterangan |
|----|--------------|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | | Garis hidup <i>/ lifeline</i> | Menyatakan kehidupan suatu <i>object</i> atau menyatakan keberadaan sebuah <i>object</i> , <i>lifeline</i> biasanya garis putus <i>vertical</i> yang ditarik dari sebuah objek. |
| 2. | Atau | Aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi dibuat sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi <i>actor</i> adalah gambar orang, tapi belum tentu merupakan orang. |
| 3. | | Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi pesan, digambarkan segi empat yang disimpan di <i>lifeline</i> yang menggambarkan suatu objek yang akan melakukan aksi. |
| 4. | | Pesan tipe <i>create</i> | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat |
| 5. | | Pesan tipe <i>call</i> | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode |
| 6 | | Pesan tipe <i>send</i> | Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim |

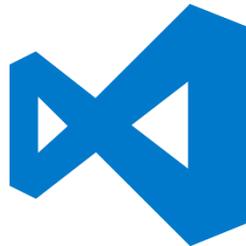


Lanjutan Tabel 2.4 Simbol-simbol *Sequence Diagram*

| | | | |
|---|-----------------------------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7 |  | Pesan tipe keluaran | 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 |
|---|-----------------------------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

2.4 Teori Program

2.4.1 Pengertian Visual Studio Code



Gambar 2.3 Logo *Visual Studio Code*

Menurut Salamah (2021:1) 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. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code (seperti C++, C#, Python, Go, Java, dst).

Menurut Faisal (2017:13) Visual Studio Code adalah integrated development environment (IDE) yang dikembangkan oleh Microsoft untuk mempermudah software developer mengembangkan aplikasi pada platform milik Microsoft.



2.4.2 Pengertian HTML



Gambar 2.4 Logo *Hyper Text Mark Up Language*

Kaban (2019:4), HTML merupakan singkatan dari Hyper Text Markup Language yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (World Wide Web Consortim) berupa tag-tag yang menyusun setiap elemen dari website.

Menurut Enterprise (2018:21), “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”.

2.4.3 Pengertian CSS



Gambar 2.5 Logo *Cascading Style Sheets*

Sulistiono (2018:2) menjelaskan bahwa CSS (*Cascading Style Sheet*) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan bahasa pemrograman. Sama halnya seperti style dalam aplikasi pengolahan kata, Microsoft Word yang dapat mengatur beberapa style, misalnya heading, subbab, bodytext, footer, images, dan style lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (file).



Lewenusa (2019:1) menuliskan, “CSS (*Cascading Style Sheet*) secara sederhana adalah sebuah metode yang digunakan untuk mempersingkat penulisan tag HTML, seperti *font*, *color*, *text* dan tabel menjadi lebih ringkas sehingga tidak terjadi pengulangan penulisan. CSS digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format berbeda”.

2.4.4 Pengertian Bootstrap



Gambar 2.6 Logo *Bootstrap*

Menurut Alatas dalam Sa’ad (2020:31), “Bootstrap adalah *framework* atau tools CSS yang memudahkan pengembangan untuk membangun website menarik dan responsif. Bootstrap memberikan solusi rapi dan seragam terhadap solusi umum, tugas *interface* yang setiap pengembang hadapi”.

Menurut Abdulloh (2018:261), “Bootstrap merupakan *framework* CSS paling populer dari sekian banyak *framework* CSS yang ada. Bootstrap memungkinkan desain sebuah web menjadi responsive sehingga dapat dilihat dari berbagai macam ukuran device dengan tampilan tetap menarik”.

2.4.5 Pengertian PHP



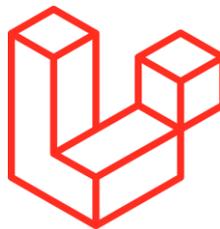
Gambar 2.7 Logo *Hypertext Preprocessor*



Menurut Arief dalam Sa'ad (2020:24), “PHP adalah salah satu bahasa *server-side-scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server-side-scripting* maka sintaks dan perintah-perintah PHP akan diesksekusi di *server* kemudian hasilnya akan dikirimkan ke browser dengan format HTML”.

Menurut Enterprise (2018:1), “PHP merupakan bahasa pemrograman yang digunakan untuk membuat website dinamis dan interaktif. Dinamis artinya, website tersebut bias berubah-ubah tampilan dan kontennya sesuai kondisi tertentu”.

2.4.6 Pengertian Laravel



Gambar 2.8 Logo *Laravel*

Menurut Supardi dan Sulaeman (2019:1), “Laravel adalah sebuah *framework* PHP yang dirilis di bawah lisensi MIT dan dibangun dengan konsep MVC (*Model View Controller*). Laravel merupakan pengembangan *website* berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi yang menyediakan sintaks yang espresif, jelas dan menghemat waktu”.

Menurut Aminudin (2015:1) Laravel adalah sebuah Framework PHP dirilis dibawah lisensi MIT dengan kode sumber yang sudah disediakan oleh Github, sama seperti framework-framework yang lain, Laravel dibangun dengan konsep MVC (Model-Controller-View), kemudian Laravel dilengkapi juga command line tool yang bernama “Artisan” yang bisa digunakan untuk packaging bundle dan instalasi bundle melalui command prompt.



Berikut ini beberapa fitur yang dimiliki oleh framework Laravel menurut Aminudin (2015:5) :

1. Bundles
Bundles yaitu sebuah fitur dengan system pengemasan modular dan berbagai bundle telah tersedia untuk digunakan dalam aplikasi Anda.
2. Eloquent ORM
Eloquent ORM merupakan penerapan PHP lanjutan dari pola “active record” menyediakan metode internal untuk mengatasi kendala hubungan antara objek database. Pembangun query Laravel Fluent didukung Eloquent.
3. Application Logic
Application Logic merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan Controllers maupun sebagai bagian dari deklarasi Route. Sintaks yang digunakan untuk mendefinisikannya mirip dengan yang digunakan oleh framework Sinatra.
4. Reverse Routing
Reverse Routing mendefinisikan hubungan antara link dan route, sehingga jika suatu saat ada perubahan pada route secara otomatis akan tersambung dengan link yang relevan. Ketika link yang dibuat dengan menggunakan nama-nama dari route yang ada, secara otomatis laravel akan membuat URI yang sesuai.
5. Restful Controllers
Restful Controllers memberikan sebuah option (pilihan) untuk memisahkan logika dalam melayani HTTP GET dan permintaan POST.
6. Class Auto Loading
Class Auto Loading menyediakan otomatis loading untuk class-class PHP, tanpa membutuhkan pemeriksaan manual terhadap jalur masuknya. Fitur ini mencegah loading yang tidak perlu.
7. View Composers
View Composers adalah kode unit logical yang dapat dijalankan ketika sebuah view di load.



8. IoC Container

IoC Container memungkinkan untuk objek baru yang dihasilkan dengan mengikuti prinsip control pembalik, dengan pilihan contoh dan referensi dari objek baru sebagai Singletons.

9. Migrations

Migrations menyediakan versi sistem control untuk skema database, sehingga memungkinkan untuk menghubungkan perubahan adalah basis kode aplikasi dan keperluan yang dibutuhkan dalam merubah tata letak database. Mempermudah dalam penempatan dan memperbarui aplikasi.

10. Unit Testing

Unit Testing mempunyai peran penting dalam framework Laravel, dimana unit testing ini mempunyai banyak tes untuk mendeteksi dan mencegah regresi. Unit testing dapat dijalankan melalui fitur “artisan command-line”.

11. Automatic Pagination

Automatic Pagination menyederhanakan tugas dari penerapan halaman, menggantikan penerapan yang manual dengan metode otomatis yang terintegrasi ke Laravel.

2.4.7 Pengertian Javascript



JavaScript

Gambar 2.9 Logo Javascript

Menurut Wicaksono dalam Sa’ad (2020:32), “*Javascript* adalah bahasa pemrograman berbasis *client side scripting*, yang berarti program dieksekusi di sisi (browser) klien, klien dapat melihat skrip dan mengetahui algoritma dari program yang ditulis”.



Menurut Kadir dan Triwahyuni (2013:325) “JavaScript adalah bahasa pemrograman yang biasa diletakkan bersama kode HTML untuk menentukan suatu tindakan”.

2.4.8 Pengertian MySQL



Gambar 2.10 Logo MySQL

Enterprise (2018:2) menegaskan, “MySQL merupakan server yang melayani database. Untuk membuat dan mengolah database, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL”.

Menurut Sibero dalam Sa’ad (2020:35), “MySQL adalah suatu *RDBMS* (*Relational Data Base Management System*), yaitu aplikasi sistem yang menjalankan fungsi pengolahan data”.

2.4.9 Pengertian XAMPP



Gambar 2.11 Logo Xampp

Menurut Riyanto dalam Isty & Afifah (2018) “XAMPP merupakan paket web server berbasis open source yang dapat dipasang pada beberapa sistem operasi yang ada (Windows, Linux, dan Mac OS)”.

Menurut Mawaddah dan Fauzi (2018:2), “XAMPP adalah software web server apache yang di dalamnya tertanam server MySQL yang didukung dengan bahasa pemrograman PHP untuk membuat website yang dinamis. XAMPP sendiri dapat berjalan pada berbagai macam platform seperti Windows, Linux, Mac OS X dan Solaris”.