



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Aditama et al., (2022:38) “Komputer adalah kumpulan objek yang dapat ditemukan di berbagai tempat, seperti di kantor atau rumah, sebagai alat untuk melakukan berbagai tugas”.

Hanafri et al., (2019:88), “Pada awalnya, kata komputer berarti orang berprofesi dalam bidang aritmatika, yaitu yang melakukan perhitungan aritmatika dengan menggunakan alat ataupun tidak. Tapi, arti kata ini mengalami perubahan. Istilah komputer mempunyai arti yang luas dan berbeda bagi setiap orang. Istilah komputer (*computer*) diambil dari bahasa latin *computere* yang berarti menghitung (*to compute atau to reckon*).”

Berdasarkan definisi komputer dapat disimpulkan bahwa komputer merupakan alat elektronik yang bisa mengelolah data dengan mengikuti perintah/prosedur program yang sudah dirancang sehingga memberikan informasi dengan akurat.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Menurut Bratha, (2022:348), “*Software* merupakan suatu urutan perintah atau instruksi untuk menjalankan berbagai komponen perangkat keras secara operasional dan akurat dan optimal”.

Siswanto (2021:30) mengemukakan bahwa, “*Software* komputer adalah perangkat lunak komputer untuk mengontrol perangkat keras. Bisa juga diartikan sebagai data yang diprogram dan disimpan secara digital dalam bentuk aplikasi, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer. Dengan kata lain, bagian sistem komputer yang tidak berwujud”.

Berdasarkan pendapat diatas maka dapat disimpulkan bahwa perangkat lunak (*Software*) adalah suatu objek yang berupa sebuah program yang



didalamnya terdapat kumpulan perintah untuk mengolah ataupun memproses informasi di komputer.

2.1.3 Pengertian Basis Data (*Database*)

Fathansyah (dalam Pakpahan et al. 2020:111) “Mengemukakan bahwa “Basis data terdiri atas dua kata, yaitu basis dan data basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang / berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. Sehingga dapat didefinisikan basis data adalah kumpulan file / tabel / arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik”.

Agustin (2022:4) mengemukakan bahwa, “Basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.”.

Berdasarkan pendapat diatas maka dapat disimpulkan bahwa basis data (*Database*) adalah sekumpulan data yang disimpan dalam komputer secara sistematis yang mempunyai perintah tertentu untuk membuat, mengakses dan mengatur data yang ada di dalamnya.

2.1.4 Pengertian Internet

Wiwi dan Syahlanisyiam, (2022:14), “Internet adalah suatu sistem jaringan yang dapat menghubungkan satu perangkat ke perangkat lainnya. Internet adalah sistem global dari seluruh jaringan komputer yang saling terhubung menggunakan standar Internet Protocol Suite”.

Setiyani (dalam Putri, 2020:92), “Internet merupakan jaringan komputer yang terdiri dari berbagai perangkat komputer yang terhubung melalui protokol untuk memberikan informasi dan internet menyediakan sumber-sumber yang dapat di akses oleh manusia di seluruh dunia”.

Berdasarkan pendapat diatas maka dapat disimpulkan bahwa internet adalah jaringan komputer yang menghubungkan semua komputer melalui



protokol untuk memberikan informasi dan menyediakan sumber-sumber yang dapat diakses oleh semua manusia diseluruh dunia.

2.2 Teori Khusus

2.2.1 Pengertian *Unified Modelling Language (UML)*

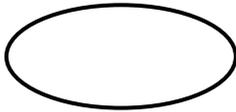
Menurut Mulyani (dalam Julianti, Dzulhaq, and Subroto 2019:93) “UML merupakan teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada system”.

Kurniawan et al., (2021:161), “UML (*Unified Modeling Language*) merupakan pengganti dari metode analisis berorientasi object dan design berorientasi object (*OOAD / Objectoriented analysis and design*) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an”.

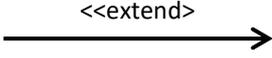
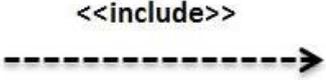
2.2.2 Pengertian *Usecase*

Menurut Hanafri et al., (2019:88) “*Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Usecase* mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi fungsi itu”.

Tabel 2. 1 Simbol-simbol *Usecase Diagram*

| No | Simbol | Keterangan |
|----|------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | Usecase  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase Nama <i>Usecase</i> . |
| 2. | Aktor/ <i>Actor</i>  | 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. |
| 3. | Asosiasi / <i>Association</i>  | Komunikasi antar aktor dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan aktor. |
| 4. | Ekstensi / <i>Extend</i>  | Relasi usecase tambahan ke sebuah usecase dimana usecase yang ditambahkan dapat berdiri sendiri walau tanpa usecase tambahan itu; mirip dengan prinsip inheritance objek; ditambahkan, misal arah panah mengarah pada usecase yang ditambahkan; biasanya usecase yang menjadi extend-nya merupakan jenis yang sama dengan usecase yang menjadi induknya. |
| 5. | Generalisasi/ <i>Generalization</i>  | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah usecase dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
| 6. | Menggunakan/ <i>include/ uses</i>  | Relasi usecase tambahan ke sebuah usecase dimana usecase yang ditambahkan memerlukan usecase ini untuk menjalankan fungsing atau sebagai syarat dijalankan usecase. |

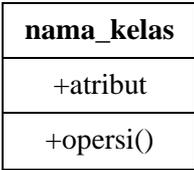
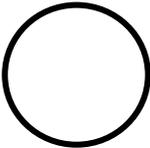
Sumber: Rosa and Shalahuddin (2019:156)



2.2.3 Pengertian *Class Diagram*

Menurut Anggraini et al., (2020:67) “Class diagram merupakan model yang menggambarkan struktur dan deskripsi class serta dapat menghubungkan antara class yang lain”.

Tabel 2. 2 Simbol-simbol *Class Diagram*

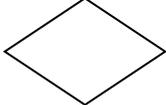
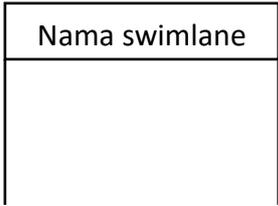
| No | Simbol | Keterangan |
|----|---------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Kelas  | Kelas pada struktur sistem. |
| 2 | Antarmuka / <i>Interface</i>  | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasiobjek. |
| 3 | Asosiasi / <i>association</i>  | Relasi antar kelas dengan makna umum asosiasi biasanya juga disertai dengan <i>multiplicit</i> . |
| 4 | Asosiasi berarah / <i>directed association</i>  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| 5 | Agregasi / <i>aggregation</i>  | Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>). |
| 6 | Generalisasi  | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus). |
| 7 | Kebergantungan / <i>dependecy</i>  | Relasi antar kelas dengan makna kebergantungan antar kelas. |

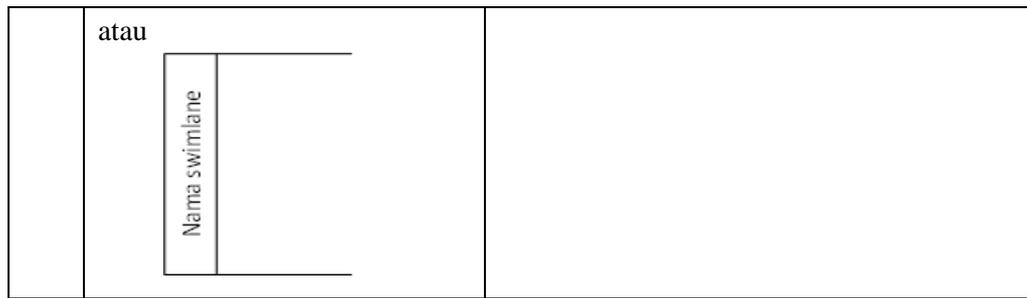
Sumber: Rosa and Shalahuddin (2019:146)

2.2.4 Pengertian *Activity Diagram*

Novitasari (dalam Kurniawan et al., 2021:162), “*Activity diagram* adalah pemodelan yang dilakukan pada suatu sistem dan menggambarkan aktivitas sistem berjalan. *Activity diagram* di gunakan sebagai penjelelasan aktivitas program tanpa melihat koding atau tampilan”.

Tabel 2. 3 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 / 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  | 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. |

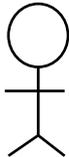
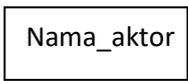


Sumber: Rosa and Shalahuddin (2019:162)

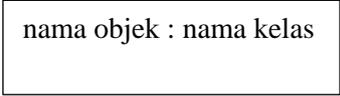
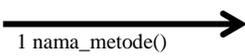
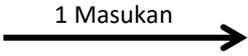
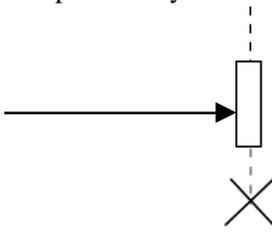
2.2.5 Pengertian *Sequence Diagram*

Menurut Sukamto & Shalahudin (dalam Afifah and Setyantoro 2021:112), “Sequence Diagram digunakan untuk menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima oleh objek. Banyaknya diagram sekuen yang kan dibangun sesuai dengan pendefenisian use case yang memiliki proses sendiri”.

Tabel 2. 4 Simbol-simbol *Sequence Diagram*

| No | Simbol | Deskripsi |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | Aktor  Atau  | 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 actor |
| 2. | Garis hidup/ <i>lifeline</i>  | Menyatakan kehidupan suatu objek |



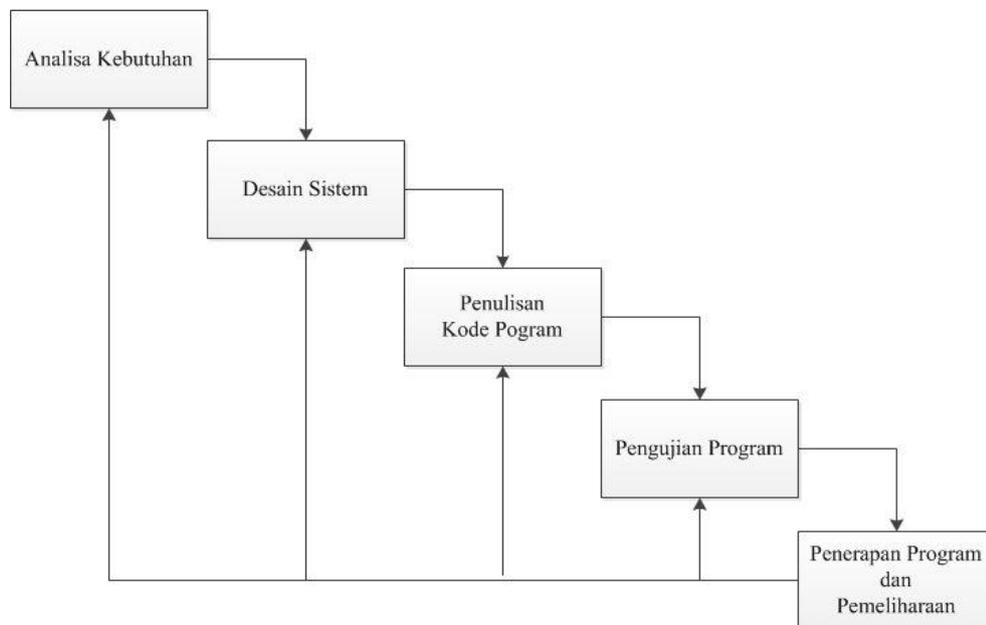
| | | |
|----|-----------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3. | Objek  | Menyatakan objek yang berinteraksi pesan |
| 4 | Waktu Aktif  | Menyatakan objek dalam keadaan aktif dan berinteraksi pesan. |
| 5 | Pesan tipe Create  | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| 6 | Pesan tipe Call  | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri. |
| 7 | Pesan tipe Send  | Menyatakan bahwa satu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim. |
| 8 | Pesan tipe Return  | 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 destroy  | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy. |

Sumber: Rosa and Shalahuddin (2019:165)

2.2.6 Pengertian Metode 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 jatuh satu per satu, model ini menggunakan penyelesaian tahap sebelumnya baru kemudian dapat dilanjutkan ke tahap berikutnya.

Dapat disimpulkan bahwa *Metode Waterfall* adalah suatu proses atau metode pengembangan sistem secara berurutan yang mengalir ke bawah (seperti air terjun). Tahapan metode waterfall dapat dilihat pada gambar di bawah ini:



Berdasarkan gambar di atas, pengembangan sistem dengan menggunakan *metode waterfall* memiliki beberapa tahapan yang berurut yaitu: *Requirement Analysis, Design System, Implementation, Testing, dan Maintenance*. Tahapan-tahapan dari *metode waterfall* adalah sebagai berikut:

a) *Requirement Analisis* (Analisis Kebutuhan)

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* (Desain Sistem)

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* (Pengujian Kode Program)

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) *Testing* (Pengujian Program)

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* (Penerapan & Pemeliharaan)

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.

2.2.7 Pengertian Kamus Data (*Data Dictionary*)

Kurniawan (dalam Premana 2019:56) “Kamus data adalah penjabaran dari aliran-aliran data yang ada di dalam sebuah data flow diagram. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.”



2.3 Teori judul

2.3.1 Pengertian Aplikasi

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”.

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

Berdasarkan pendapat diatas maka dapat disimpulkan bahwa aplikasi adalah suatu program yang siap digunakan untuk mengolah data dan melaksanakan suatu fungsi dengan sasaran tertentu sehingga menyajikan atau menampilkan hasil yang diinginkan.

2.3.2 Pengertian Website

Romadhon et al., (2021:31), “*Website* adalah kumpulan informasi/kumpulan page yang biasa diakses lewat jalur internet. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara online di jaringan internet. Secara teknis, *website* adalah kumpulan dari page, yang tergabung kedalam suatu domain atau subdomain tertentu”.

Novendri (2019:49) “Pengertian Web atau Situs Website atau situs dapat diartikan sebagai kumpulan halaman halaman yang digunakan untuk menampilkan informasi, teks, gambar diam atau bergerak, animasi, suara, dan atau gabungan dari semuanya itu, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling berkait dimana masing masing dihubungkan dengan jaringan jaringan halaman/hyperlink”.

Berdasarkan pendapat diatas maka dapat disimpulkan bahwa *website* adalah halaman yang terdapat dalam sebuah domain yang berisikan informasi yang memiliki topik saling terhubung.



2.3.3 Pengertian Aplikasi Peminjaman Mobil Dinas Pada PT. Bukit Asam, Tbk Berbasis *Website*

Aplikasi Peminjaman Mobil Dinas Pada PT. Bukit Asam, Tbk Berbasis *Website* adalah sebuah aplikasi yang digunakan untuk mempermudah proses peminjaman mobil dinas pada PT. Bukit Asam, Tbk sehingga proses peminjaman mobil dinas tersebut tidak harus mendatangi kantor bagian kendaraan dan staff tidak perlu mencatat pada buku peminjaman.

2.4 Teori Program

2.4.1 Pengertian MySQL



Sumber : <https://www.mysql.com/>

Gambar 2. 1 Logo MySQL

Novendri (2019:48) “MySQL disebut juga SQL yang merupakan singkatan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. SQL pertama kali didefinisikan oleh *American National Standards Institute* (ANSI) pada tahun 1986. MySQL adalah sebuah sistem manajemen database yang setbersifat *open source*”.

Menurut Purnamasari (dalam Winanjar dan Susanti, 2021:99), “MySQL adalah salah satu jenis *database server* yang sangat populer, hal ini disebabkan karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses data basenya. MySQL bersifat *Open Source, Software* ini dilengkapi dengan *Source code* (kode yang dipakai untuk membuat MySQL)”.



2.4.2 Pengertian PHP



Sumber : <https://www.php.net/>

Gambar 2. 2 Logo PHP

Menurut Sahi (2020:121), “PHP merupakan singkatan dari *Hypertext Preprocessor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan *website* yang disisipkan pada dokumen HTML. PHP merupakan *software open source* yang disebarakan dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya”.

2.4.3 Pengertian HTML



Sumber : <https://www.nurulfikri.com/article/mengenal-html/>

Gambar 2. 3 Logo HTML

Menurut Vandenberg (dalam Steven 2019:4) HTML adalah standar teks untuk mencari gambar, memformat sebuah teks, dan menempatkan referensi ataupun tautan ke dokumen, selain itu HTML berguna sebagai struktur dokumen pada tingkat lebih atas”.



2.4.4 Pengertian XAMPP

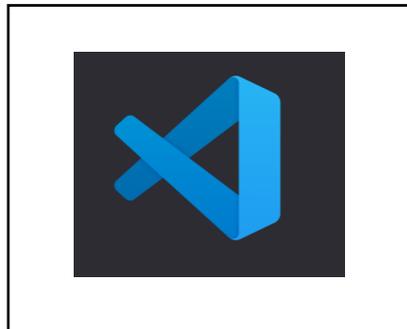


Sumber : <https://www.apachefriends.org/>

Gambar 2. 4 Logo XAMPP

Menurut Putra dan Nita (2019:82), “XAMPP merupakan software server apache di mana memiliki banyak keuntungan seperti mudah untuk digunakan, tidak memerlukan biaya serta mendukung pada instalasi Windows dan Linux. Hal ini juga didukung karena dengan instalasi yang di lakukan satu kali tersedia MySQL, apache web server, Database server PHP support”.

2.4.5 Pengertian Visual Studio Code



Sumber : <https://code.visualstudio.com/>

Gambar 2. 5 Logo Visual Studio Code

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



2.4.6 Pengertian CSS

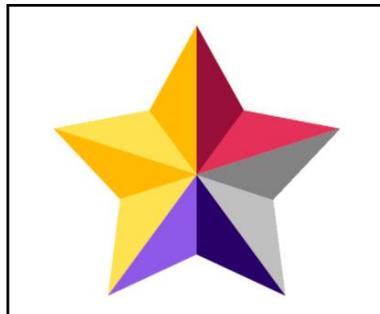


Sumber : <https://www.freepik.com/>

Gambar 2. 6 Logo CSS

Menurut Sengkey (dalam Fatimah and Samsudin 2019:36), “Cascading Style Sheet (CSS) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman. Pada umumnya CSS dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa HTML dan XHTML.”

2.4.7 Pengertian StarUML



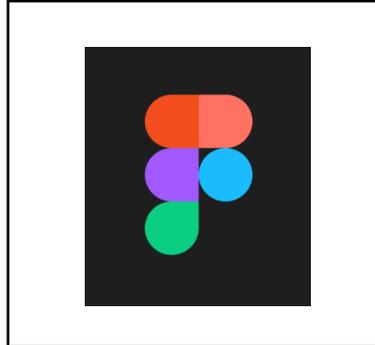
Sumber : <https://staruml.io/>

Gambar 2. 7 Logo StarUML

Menurut (Novi Ika Sari, Dedi Wirasasmita 2020:73), “StarUML adalah sebuah proyek open source untuk pengembangan secara cepat, fleksibel, extensible, featureful, dan bebas-tersedia. Tujuan dari proyek StarUML adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat UML komersial seperti Rational Rose, Together dan sebagainya.”



2.4.8 Pengertian Figma



Sumber : <https://www.figma.com/>

Gambar 2. 8 Logo Figma

Menurut Kurniawan dan Romzi (2022:3) “Figma adalah aplikasi desain UI dan UX berbasis browser, dengan desain yang sangat baik, prototyping, dan alat pembuatan kode. Saat ini (bisa dibilang) alat desain antarmuka terkemuka di industri, dengan fitur-fitur canggih yang mendukung tim yang bekerja pada setiap fase proses desain.”