



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Ramadhan dkk (2022:4), “Komputer adalah sebuah perangkat elektronik yang dapat menerima, menyimpan, memproses, dan menghasilkan informasi dengan menggunakan rangkaian instruksi atau program yang telah ditentukan”.

Menurut Manurung (2020:54), “Komputer merupakan mesin penghitung elektronik yang cepat dan dapat menerima informasi input digital, kemudian memprosesnya sesuai dengan program yang tersimpan di memorinya, dan menghasilkan output berupa informasi”.

Berdasarkan definisi diatas, dapat disimpulkan bahwa komputer merupakan alat elektronik yang dapat menerima, memproses, serta menyimpan data *input-an* yang diolah yang dioperasikan oleh manusia.

2.1.2 Pengertian Perangkat Lunak

Menurut Roihan (2018;1), “Perangkat lunak adalah bagian yang tidak berwujud dalam sebuah perangkat elektronika yang dimasukkan ke dalam perangkat keras (*hardware*) berfungsi sebagai pelaksana tugas dari instruksi yang diberikan oleh pengguna (*user*)”.

Menurut Hafipah, dkk (2022:159), “Perangkat lunak adalah aplikasi yang terbuat dari berbagai instruksi berupa *code*, yang berfungsi untuk mengolah data yang diterima menjadi suatu informasi, kemudian informasi tersebut disimpan secara digital pada komputer, perlu diketahui bahwa aplikasi yang dibuat berdasarkan kebutuhan suatu organisasi atau perusahaan”.

Dari pendapat di atas dapat disimpulkan bahwa perangkat lunak adalah instruksi-instruksi yang ditujukan kepada komputer dan kumpulan data untuk melakukan proses pengelolaan data.



2.1.3 Pengertian Perangkat Keras

Menurut Barokah, dkk (2021:365), “Perangkat keras (*hardware*) merupakan salah satu komponen dari sebuah komputer yang ciri alatnya dapat dirasakan oleh indra peraba serta dicermati secara langsung atau dapat dikatakan bahwa bentuknya nyata yang berfungsi untuk mendukung proses komputerisasi”.

Menurut Kustina, dkk, (2022:26), “Perangkat keras (*hardware*) adalah suatu perangkat komputer yang terdiri dari komponen-komponen elektronik berupa benda”.

Dari pendapat di atas dapat disimpulkan bahwa perangkat keras adalah komponen fisik komputer yang bisa dilihat dan dicermati secara langsung.

2.1.4 Pengertian Sistem

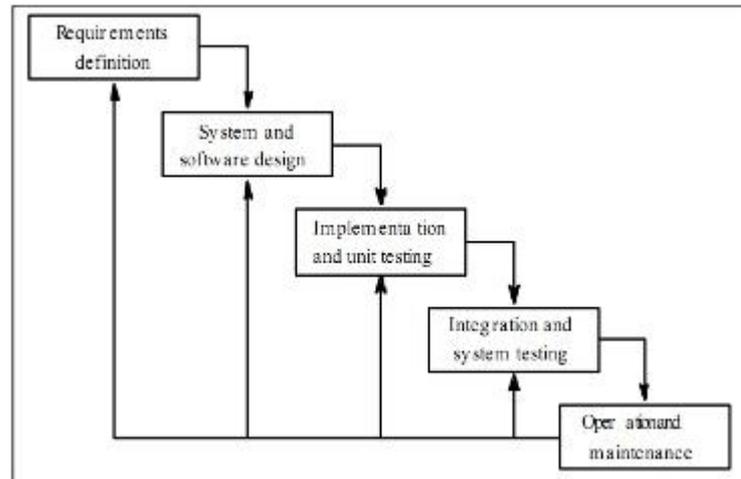
Menurut Hall dalam (Purnama, 2021:27) menuliskan, “Sistem adalah sekelompok dari dua atau lebih subsistem yang mempunyai hubungan dan memiliki suatu tujuan yang sama.”

Menurut Huda Misbahul (2020:6), “Sistem merupakan suatu yang terdiri dari sejumlah unsur atau komponen yang saling mempengaruhi dan terkait satu sama lain oleh satu atau beberapa asas”.

Dapat disimpulkan bahwa sistem adalah kumpulan atau himpunan dari suatu unsur yang terorganisasi serta saling tergantung satu sama lain dan terpadu

2.1.5 Metode Pengembangan Sistem Waterfall

Menurut Fathoroni, Annisa dkk, (2020:19-20), “Metode air terjun atau yang sering sering dinamakan siklus hidup klasik (*classic life cycle*), nama model ini sebenarnya adalah “*Linear Sequential Model*”, 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*), pemodelan (*modelling*), konstruksi (*construction*), serta penyerahan system ke para pengguna (*develoyment*), yang diakhiri dengan dukungan pada perangkat lunak yang dihasilkan”.



Gambar 2.1 Tahapan dalam Model *Waterfall*

Tahapan-tahapan dalam model ini menurut Fathoroni dkk adalah sebagai berikut:

a. *Requirement Analisis*

Tahap ini pengembangan sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dimulai dapat diperoleh melalui wawancara.

b. *System Design*

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain disiapkan. Desain system membantu dalam menentukan perangkat keras (hardware) dan system persyaratan dan juga membantu dalam mendefinisikan arsitektur system secara keseluruhan.

c. *Implementation*

Pada tahap ini, system 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 sitem setelah pengujian yang dilakukan masing-masing *unit*. Setelah integrasi seluruh system diuji unruk mengecek setiap kegagalan kesalahan.



e. Operation & Maintenance

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. Perbaikan implementasi *unit* sistem dan peningkatan jasa sebagai kebutuhan baru.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Menurut Pane, Syafrial Fachri dkk (2020:53), “Aplikasi adalah suatu perangkat lunak (*software*) atau program komputer yang beroperasi pada sistem tertentu yang diciptakan dan dikembangkan untuk melakukan perintah tertentu”.

2.2.2 Pengertian Penerimaan

Menurut Sujalu dkk (2021:129), “Penerimaan (*revenue*) adalah seluruh pendapatan yang diterima dari hasil penjualan barang pada tingkat harga tertentu”.

2.2.3 Pengertian Retribusi

Menurut Arafat, Yasser dkk (2021:5), “Retribusi adalah pungutan daerah sebagai pembayaran atas jasa atau pemberian izin tertentu yang khusus disediakan dan/atau diberikan oleh pemerintah daerah untuk kepentingan pribadi atau badan”.

2.2.4 Pengertian Parkir Tepi Jalan Umum

Menurut Peraturan Daerah Kabupaten Mamuju Nomor 16 Tahun 2011 Tentang Retribusi Pelayanan Parkir Di Tepi Jalan Umum Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, yang dimaksud dengan parkir tepi jalan umum adalah perbuatan/tindakan memarkir kendaraan bermotor di tepi jalan umum yang dibolehkan dan ditentukan oleh Pemerintah Daerah sesuai dengan ketentuan Peraturan Undang-Undang tentang Jalan dan Undang-undang tentang Lalu lintas dan Angkutan Jalan.



2.2.5 Pengertian QR Code

Menurut Agastya (2019:77), “*Quick Response (QR) Code* merupakan barcode dua dimensi yang dapat menyimpan berbagai informasi dan cara mengakses atau membuka qr code adalah dengan cara melakukan scan dengan smartphone yang dimiliki”.

2.2.6 Pengertian Website

Menurut Abdulloh dalam (Sa’ad, 2019:3), “*Website* atau web adalah sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet.

2.2.7 Pengertian Aplikasi Penerimaan Retribusi Parkir Tepi Jalan Umum Menggunakan QR Code Berbasis Website Pada Dinas Perhubungan Kota Palembang

Aplikasi Penerimaan Retribusi Parkir Tepi Jalan Umum Menggunakan QR Code Berbasis Website adalah aplikasi yang digunakan untuk mempermudah dalam mengidentifikasi juru parkir serta membantu dalam pengelolaan penerimaan retribusi parkir tepi jalan umum.

2.3 Teori Khusus

2.3.1 *Unified Modeling Language (UML)*

Menurut Hutabri dan Putri (2019:58), “*Unified Modeling Language (UML)* merupakan bahasa pemodelan perangkat lunak yang digunakan untuk perancangan system yang berorientasi objek. UML dapat menggambarkan perangkat lunak yang dirancang secara visual, mendokumentasi, serta menspesifikasikannya agar mudah dipahami oleh programmer dan user”.

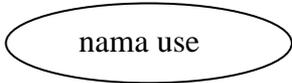
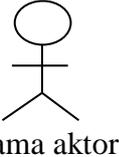


2.3.2 Jenis-Jenis Diagram UML

2.3.2.1 Pengertian Use Case Diagram

Menurut Hutabri dan Putri (2019:58), “*Use Case Diagram* atau diagram use case adalah diagram untuk memodelkan perilaku suatu system yang akan dirancang dengan menggambarkan interaksi antara satu atau lebih aktor yang akan menggunakan sistem”.

Tabel 2.1 Simbol-simbol *Use Case Diagram*

No.	Simbol	Deskripsi
1	<p><i>Use Case</i></p> 	Merupakan fungsi atau kegunaan sistem untuk unit atau sistem pertukaran pesan antara unit dan aktor.
2	<p>Aktor</p> 	Merupakan aktor atau orang, sistem atau proses yang berperan dan memiliki hubungan dengan system informasi yang dibuat.
3	<p>Asosiasi / <i>Association</i></p> 	Merupakan interaksi aktor dengan <i>use case</i> atau komunikasi antara actor dengan <i>use case</i>
4	<p>Ekstensi / <i>Extend</i></p> 	Merupakan relasi tambahan <i>use case</i> dengan <i>use case</i> lain. Tanpa <i>use case</i> yang ditambahkan. Arah dari panah ke arah <i>use case</i> yang di tambahkan.
5	<p>Uses / <i>Include</i></p> 	Merupakan relasi dua <i>use case</i> , <i>use case</i> yang ditambahkan dengan <i>use case</i> tambahan. <i>Use case</i> yang ditambahkan memerlukan <i>use case</i> tambahan untuk menjalankan kegunaan atau fungsi sebagai syarat menjalankan <i>use case</i>



Lanjutan Tabel 2.1 Simbol-simbol *Use Case Diagram*

6	<p><i>Generalization / Generalisasi</i> (Menggabungkan)</p> <p style="text-align: center;">.....→</p>	<p>Merupakan hubungan umum-khusus (generalisasi dan spesialisasi) untuk dua <i>use case</i>. Satu <i>use case</i> memiliki fungsi yang lebih general (umum).</p>
---	-----------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------

Sumber : Hutabri dan Putri(2019:59)

2.3.2.2 Pengertian *Activity Diagram*

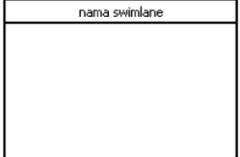
Menurut Hutabri dan Putri (2019:59), “Diagram aktivitas (*activity diagram*) mendeskripsikan aliran kerja (*workflow*) atau aktivitas system atau proses bisnis atau menu yang terdapat di dalam system atau perangkat lunak”.

Tabel 2.2 Simbol-simbol *Activity Diagram*

No.	Simbol	Deskripsi
1	<p>Status awal</p> 	Merupakan status awal keadaan dari system, setiap diagram aktivitas memiliki satu status awal.
2	<p>Aktivitas</p> 	Merupakan kegiatan yang dilakukan system, sering dimulai dengan kata kerja.
3	<p><i>Decision</i> (Percabangan)</p> 	Merupakan hubungan percabangan untuk keputusan aktivitas yang memiliki lebih dari satu pilihan.
4	<p><i>Join</i> (Penggabungan)</p> 	Merupakan hubungan penggabungan jika satu atau lebih aktivitas menjadi menjadi satu
5	<p>Status akhir</p> 	Bagaimana objek dibentuk dan dihancurkan.



Lanjutan Tabel 2.2 Simbol-simbol *Activity Diagram*

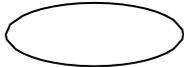
6	<p><i>Swimlane</i></p> 	<p>Merupakan yang memisahkan organisasi bisnis. Memiliki tanggung jawab untuk aktivitas yang terjadi.</p>
---	----------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------

Sumber : Hutabri dan Putri (2019:59)

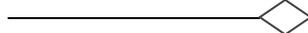
2.3.2.3 Pengertian *Class Diagram*

Menurut Hutabri dan putri (2019:60), “*Class Diagram* atau Diagram kelas digunakan untuk membuat system dengan mendeskripsikan struktur system dari sisi pendefinisian kelas yang dibuat”.

Tabel 2.3 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1	<p>Kelas</p> 	<p>Merupakan kelas yang ada pada stuktur system. Memiliki atribut dan operasi dalam kelas</p>
2	<p>Antarmuka / <i>interface</i></p> 	<p>Merupakan kemiripan dengan kelas tetapi memiliki metode yang di deklarasikan tanpa isi dan tanpa atribut kelas.</p>
3	<p>Asosiasi</p> 	<p>Merupakan relasi antarkelas (umum), biasanya dilengkapi dengan <i>multiplicity</i></p>

Lanjutan Tabel 2.3 Simbol-simbol *Class Diagram*

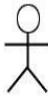
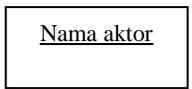
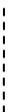
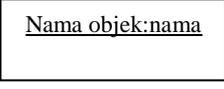
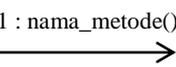
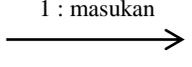
No	Simbol	Deskripsi
4	Asosiasi berarah / directed association 	Relasi <i>antarclass</i> bermakna satu kelas yang digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus).
6	Ketergantungan / <i>dependency</i> 	Merupakan relasi ketergantungan (<i>dependency</i>) antar kelas.
7	Agregasi / <i>Aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

Sumber : Hutabri dan Putri (2019:60)

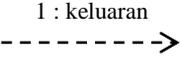
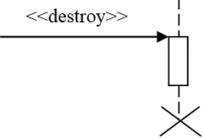
2.3.2.4 Pengertian Sequence Diagram

Menurut Hutabri dan Putri (2019:60), "Diagram Sekuen (*Sequence Diagram*) mendeskripsikan suatu perilaku objek pada *use case* dengan menjelaskan alur waktu hidup dari objek dan pesan atau *message* yang diterima dan dikirm antar objek".


Tabel 2.4 Simbol-simbol *Sequence Diagram*

No	Simbol	Deskripsi
1	<p>Aktor</p>  <p>Atau</p> 	Merupakan orang atau system lain, atau proses diluar sistem yang dibuat, yang berhubungan dengan sistem yang dibuat. Sistem yang dibuat belum tentu orang, walaupun memiliki symbol bergambar orang.
2	<p>Lifeline (Garis hidup)</p> 	Merupakan garis hidup objek yang menerangkan kehidupan objek.
3	<p>Objek</p> 	Merupakan objek yang melakukan interaksi pesan.
4	<p>waktu aktif</p> 	Merupakan symbol yang menyatakan bahwa objek dalam keadaan berinteraksi dan keadaan aktif, dan semua yang berhubungan dengan waktu aktif merupakan tahapan yang dilakukan selama dalam keadaan aktif
6	<p>Pesan tipe call</p> 	Merupakan pernyataan satu objek memanggil metode atau operasi lain atau diri sendiri.
7	<p>Pesan tipe send</p> 	Merupakan pernyataan bahwa objek mengirimkan informasi atau masukan atau data ke objek lain.

Lanjutan Tabel 2.4 Simbol-simbol *Sequence Diagram*

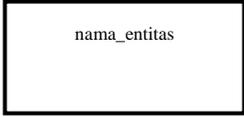
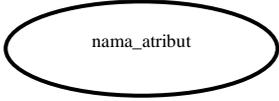
8	Pesan tipe return 	Merupakan pernyataan bahwa objek menjalankan metode operasi yang memberi hasil suatu pengembalian atau keluaran ke objek tertentu.
9	Pesan tipe destroy 	Merupakan pernyataan bahwa satu objek mengakhiri hidup dari objek lain, jika ada create lebih baik ada destroy.

Sumber : Hutabri dan Putri (2019:60)

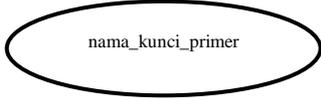
2.3.3 Pengertian ERD (*Entity Relationship Program*)

Menurut Sukamto dan Shalahuddin (2018:50), “ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD”.

Tabel 2.5 Simbol-simbol *Entity Relationship Program*

No	Simbol	Deskripsi
1	Entitas/entiry 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih kekata benda dan belum merupakan nama tabel.
2	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.

Lanjutan Tabel 2.5 Simbol-simbol *Entity Relationship Program*

3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa <i>id</i> ; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4.	Atribut Multinilai / Multivalue 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6.	Asosiasi/ <i>Association</i> 	Penghubung antara relasi dan entitas di mana dikedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

Sumber : Sukanto dan Shalahudin (2018:50-51)



2.4. Teori Program

2.4.1 Pengertian 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 Habibi (2020:77) “Visual Studio Code (VS Code) adalah teks editor ringan dan handal yang dibuat oleh Microsoft untuk system operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows”.

2.4.2 Pengertian HTML

Setyawan (2018:5), “*Hyper Text Markup Language* (HTML) merupakan mengimplementasikan konsep hypertext dalam suatu naskah atau dokumen. Namun HTML bukanlah sebuah bahasa pemrograman, melainkan sebuah standar yang digunakan secara luas untuk menampilkan halaman web”.

Menurut Aziz dkk (2019:5), “HTML atau *Hyper Text Markup Language* merupakan standar bahasa pemrograman semi terstruktur yang dibuat dalam bentuk tag-tag yang menyusun setiap elemen *website* yang dapat diakses atau ditampilkan menggunakan *browser*”.

2.4.3 Pengertian CSS

Menurut Aziz dkk (2019:36) “CSS (*Cascading Style Sheet*) adalah sebuah dokumen yang terdiri dari kode program yang digunakan untuk membuaelegan tampilan dari tampilan *website* yang dibuat”.

Rahman dkk (2020:47) menuliskan, “CSS (*Cascading Style Sheet*) adalah suatu bahasa desain web (*style sheet language*) yang mengontrol format tampilan



sebuah halaman web yang ditulis dengan menggunakan penanda (*markup language*)”.

2.4.4 Pengertian Bootstrap

Menurut Abdulloh (2022:10), “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

Menurut Azis Abdul, dkk (2019:7), “PHP merupakan bahasa pemrograman untuk membuat aplikasi berbasis *website*, PHP juga sering dikenal dengan *Hypertext Preprocessor* yang merupakan sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML”.

Menurut Sibero dalam (Supono, 2020:3), “ PHP (*Personal Home Page*) adalah pemrograman (interpreter) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan”.

2.4.6 Pengertian Laravel

Menurut Harianto dkk, (2019:14), “Laravel merupakan sebuah *framework PHP* yang akan membantu web developer untuk menciptakan aplikasi istimewa secara simple, kode yang bersih dan menyenangkan. Dalam penelitian ini framework laravel digunakan untuk mempermudah perancangan *PHP*”.

Menurut Habibi, Roni, dkk (2019:5) “Laravel adalah PHP open source framework yang dibangun dengan model view controller dan dibekali berbagai macam sintaks”.



2.4.7 Pengertian Javascript

Menurut Abdullah (2022:10), “Javascript merupakan bahasa pemrograman web yang pemrosesannya dilakukan di sisi client. Karena berjalan di sisi client. Javascript dapat dijalankan hanya dengan menggunakan browser”.

Menurut Kusumawardani dkk (2023:61), “Javascript adalah bahasa pemrograman berbasis teks yang digunakan baik di sisi klien dan sisi server memungkinkan seseorang membuat halaman web menjadi interaktif”.

2.4.8 Pengertian MySQL

Menurut Harianto dkk (2019:13-14), “MySQL (*My Structure Query Language*) adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang database sebagai sumber dan pengelolaan datanya”.

Menurut Setyawan dan Dinda (2019:52), “MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. Database management system (DBMS) MYSQL multi pengguna dan multi alur ini sudah dipakai lebih dari 6 juta pengguna di seluruh dunia”.

2.4.9 Pengertian XAMPP

Menurut Andhara, Andre dkk (2022:14), “XAMPP adalah perangkat lunak (free Software) bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, yang ditulisdengan bahasa pemrograman PHP dan Perl.”

Menurut Harefa, Kecitaan (2022:5), “XAMPP merupakan perangkat lunak berbasis web server yang bersifat open source (bebas), serta mendukung di berbagai sistem operasi, baik Windows, Linux, atau Mac OS. XAMPP digunakan sebagai standaloner server (berdiri sendiri) atau biasa disebut dengan localhost Hal tersebut memudahkan dalam proses pengeditan, desain, dan pengembangan aplikasi”.