



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Komputer adalah sebuah alat elektronik yang memiliki kemampuan untuk melakukan pengolahan data baik berupa teks, gambar maupun suara untuk menghasilkan informasi yang kita kehendaki (Harahap *et al.*, 2022).

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

2.1.2 Pengertian Perangkat Lunak (*Software*)

Software atau perangkat lunak dapat diartikan sebagai sekumpulan data elektronik yang disimpan dan diatur oleh komputer itu dapat berupa program atau instruksi yang akan menjalankan suatu perintah (Habibullah, *et al.*, 2023).

Software adalah sebuah program komputer yang menjadi jembatan penghubung antara pengguna komputer (*user*) dengan komputer. Dengan kata lain, *software* merupakan penerjemah antara *user* sebagai pemberi intruksi dan komputer sebagai pihak yang menerima intruksi, *software* tidak dapat disentuh dengan indra peraba secara fisik. Namun, *software* sangat penting dalam pengoperasian komputer (Rahma, 2020).

2.1.3 Pengertian Basis Data (*Database*)

Basis data adalah kumpulan terpadu dari elemen data logis yang saling berhubungan. Basis data mengonsolidasi banyak catatan yang sebelumnya disimpan dalam *file* terpisah (Sudarso, 2022).

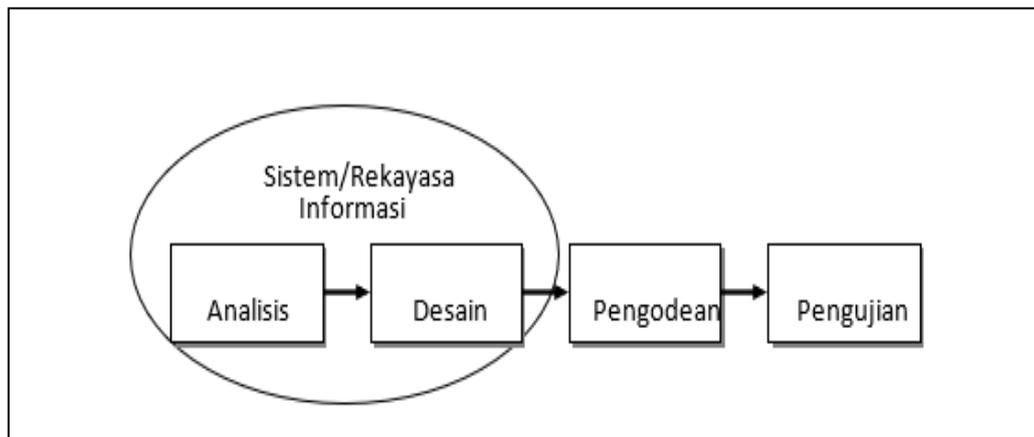
Basis data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi berbagai kebutuhan (Andriaty, 2019).



2.1.4 Metode Pengembangan Sistem *Waterfall*

Menurut Sukamto dan Shalahuddin (2018), “Model *Waterfall* adalah model yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, dan pengujian”.

Metode *waterfall* adalah konsep pengembangan yang menekankan pada langkah sistematis. Sehingga, proses penciptaan sebuah sistem harus dilakukan secara berurutan, mulai dari tahapan identifikasi kebutuhan sampai ke proses perawatan.



Sumber : Sukamto dan Shalahuddin (2018)

Gambar 2.1 Model *Waterfall*

Tahapan-tahapan dalam model *waterfall* menurut Sukamto dan Shalahuddin (2018) adalah sebagai berikut:

a. Analisis Kebutuhan

Tahap analisis kebutuhan, 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.

b. Desain

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



c. Pengkodean

Sistem terintegrasi dengan sintaksis sehingga sistem informasi dapat digunakan sesuai kebutuhan, yang akan diintegrasikan pada tahap selanjutnya.

d. Pengujian

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

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

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 (Wijayanti, 2022).

Aplikasi adalah suatu program komputer yang siap untuk digunakan dan bertujuan untuk mengerjakan tugas dari *user* (Palupi, *et al.*, 2023).

2.2.2 Pengertian Pendataan

Menurut Kamus Besar Bahasa Indonesia (KBBI), arti kata pendataan adalah proses, cara, perbuatan mendata. Arti lainnya dari pendataan adalah pengumpulan data.

Data adalah angka dan fakta yang tidak sedang digunakan pada proses keputusan dan biasanya berbentuk catatan historis yang dicatat dan di arsipkan tanpa maksud untuk segera di ambil kembali dalam rangka pengambilan keputusan (Suprianur, 2022).

2.2.3 Pengertian Penduduk

Penduduk adalah salah satu kesatuan dari manusia yang diwakili oleh suatu nilai statistik tertentu (Santoso, *et al.*, 2022).



Penduduk adalah jumlah orang yang bertempat tinggal di suatu wilayah pada waktu tertentu dan merupakan hasil dari proses-proses demografi yaitu fertilitas, mortalitas, dan migrasi (Hilmi, *et al.*, 2022).

2.2.4 Pengertian Website

Sebuah situs *web* atau *website* adalah sebutan bagi sekelompok halaman *web* (*web page*), yang umumnya merupakan bagian dari suatu domain (*domain name*) atau sub domain di *world wide web* (WWW) di internet (Budianto, *et al.*, 2022).

Website adalah sekumpulan halaman *web* yang telah dipublikasikan dan mempunyai domain/URL (*Uniform Resource Locator*) yang bisa dilihat dan diakses semua pengguna internet dengan cara menuliskan alamatnya. Dengan adanya teknologi *world wide web* (Mustaib, *et al.*, 2022).

2.2.5 Pengertian Aplikasi Pendataan Penduduk Pada Dinas Kependudukan Dan Pencatatan Sipil Kota Kayuagung Kabupaten Ogan Komering Ilir Berbasis Website

Aplikasi Pendataan Penduduk Pada Dinas Kependudukan Dan Pencatatan Sipil Kota Kayuagung Kabupaten Ogan Komering Ilir Berbasis *Website* adalah aplikasi yang dibuat dengan tujuan untuk mempermudah dalam pendataan data penduduk yang meliputi penduduk lahir, penduduk meninggal, penduduk pindah, dan penduduk datang.

2.3 Teori Khusus

2.3.1 Pengertian *Unified Modeling Language* (UML)

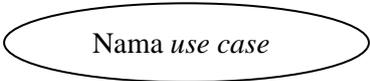
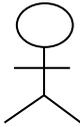
Unified Modeling Language (UML) adalah sebuah patokan bahasa untuk menulis kerangka kerja terperinci dari sebuah perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan sebuah sistem perangkat lunak dapat berupa informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak dapat berupa model, deskripsi dari sistem perangkat lunak seperti pemodelan bisnis.



2.3.2 Pengertian *Use Case Diagram*

Menurut Sukamto dan Shalahuddin (2018:155), “*Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat dapat dikatakan *use case* digunakan”.

Tabel 2.1 Simbol-simbol *Use Case Diagram*

No.	Simbol	Deskripsi
1	<p><i>Use Case</i></p> 	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
2	<p>Aktor/ <i>actor</i></p>  <p>Nama aktor</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang dibuat itu sendiri.
3.	<p>Asosiasi / <i>association</i></p> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<p>Ekstensi / <i>extend</i></p>  <p><<extend>></p>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri tanpa <i>use case</i> tambahan itu, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.
5.	<p><<include>></p> 	menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.

Sumber : Sukamto dan Shalahuddin (2018:155)



2.3.3 Pengertian Class Diagram

Menurut Sukamto dan Shalahuddin (2018:141), “Diagram kelas atau *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang sebut atribut dan *method* atau operasi”.

Tabel 2.2 Simbol-simbol *Class* Diagram

No	Simbol	Deskripsi
1.	<p>Kelas</p>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2.	<p>Antarmuka / <i>interface</i></p> <p>Nama_ <i>interface</i></p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	<p>Asosiasi / <i>association</i></p>	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai <i>multiplicity</i> .
4.	<p>Asosiasi berarah / <i>directed association</i></p>	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	<p>Agregasi / <i>aggregation</i></p>	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

Lanjutan Tabel 2.2 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
6.	Generalisasi/ <i>generalization</i> 	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum khusus).
7.	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).

Sumber : Sukamto dan Shalahuddin (2018:146)

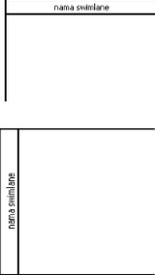
2.3.4 Pengertian *Activity Diagram*

Menurut Sukamto dan Shalahuddin (2018:161), “*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Tabel 2.3 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	Status awal 	Bagaimana objek dibentuk atau diawali.
2.	Aktivitas 	Memperlihatkan bagaimana kelas antarmuka saling berinteraksi satu sama lain.
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu

Lanjutan Tabel 2.3 Simbol-simbol *Activity Diagram*

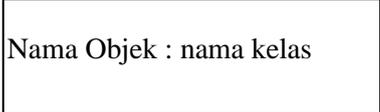
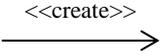
No	Simbol	Deskripsi
4.	Status akhir 	Bagaimana objek dibentuk dan dihancurkan.
5.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber : Sukamto dan Shalahuddin (2018:162)

2.3.5 Pengertian *Sequence Diagram*

Menurut Sukamto dan Shalahuddin (2018:165), “*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar objek dan 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”.

Tabel 2.4 Simbol-simbol *Sequence Diagram*

No	Simbol	Deskripsi
1.	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.
2.	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.
3.	Objek/ <i>object</i> 	Menyatakan objek yang berinteraksi pesan.
4.	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
5.	Pesan tipe <i>send</i> 	Menyatakan suatu objek mengirim data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.

Lanjutan Tabel 2.4 Simbol-simbol *Sequence Diagram*

No	Simbol	Deskripsi
6.	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.
7.	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah.

Sumber : Sukamto dan Shalahuddin (2018:165-167)

2.3.6 Kamus Data

Menurut Sukamto dan Shalahuddin (2018:73-74), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*Input*) dan keluaran (*Output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Tabel 2.5 Simbol Kamus Data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau...
{ } n	n kali diulang/bernilai banyak
()	Data opsional
..	Batas komentar

Sumber : Sukamto dan Shalahuddin (2018:74)



2.4. Teori Progam

2.4.1 Pengertian *Hypertext Markup Language* (HTML)

HTML (*Hyper Text Mark Up Language*) adalah bahasa yang digunakan untuk mendeskripsikan struktur halaman *web*. HTML berfungsi untuk mempublikasi dokumen online (Noviantoro, *et al.*, 2022).



Sumber:salmangayo.blogspot.com

Gambar 2.2 Lambang HTML

HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar *web* yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website* (Dody, *et al.*, 2021).

2.4.2 Pengertian *Hypertext Preprocessor* (PHP)

Sistem manajemen *database* yang sering digunakan bersama PHP adalah Mysql namun, PHP juga mendukung sistem manajemen *Database Oracle*, *Microsoft Access*, *Interbase*, *d-Base* dan *PostgreSQL* (Utami, 2022).



Sumber:pngimg.com

Gambar 2.3 Lambang PHP

PHP adalah salah satu bahasa pemrograman *open source* yang cocok atau



dikhususkan untuk pengembangan. Bahasa PHP dapat menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server-side*, dimana pemrosesan datanya dilakukan pada sisi server (Noviantoro, *et al.*, 2022).

2.4.3 Pengertian *Cascading Style Sheets* (CSS)

CSS adalah singkatan dari *Cascading Style Sheets* merupakan bahasa pengkodean yang memberikan tampilan dan tata letak situs *web*. Seiring dengan HTML, CSS adalah dasar untuk desain *web*. Tanpa itu, situs *web* akan tetap menjadi teks biasa dengan latar belakang putih sehingga tentunya tidak menarik dan terkesan datar atau monoton. CSS atau singkatan dari *Cascading Style Sheet* adalah suatu aturan untuk mengatur tampilan dari *website* sehingga tampilan dalam *web* lebih terstruktur (Sari, *et al.*, 2022).



Sumber: gudrilogo.blogspot.com

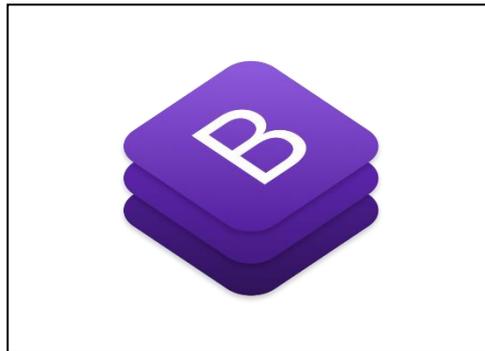
Gambar 2.4 Lambang CSS

Cascading Style Sheet (CSS) 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 (Noviantoro, *et al.*, 2022).



2.4.4 Pengertian *Bootstrap*

Bootstrap adalah salah satu kerangka kerja dari CSS yang digunakan untuk membantu *developer* dalam mempermudah, mempercepat dan memperindah tampilan *website* tersebut. *Bootstrap* ini sangat berguna dalam sisi membantu *front – end* atau disebut juga *client – side* (Firmansyah, *et al.*, 2021).



Sumber: www.kindpng.com

Gambar 2.5 Lambang *Bootstrap*

Bootstrap adalah kerangka kerja (*framework*) *front-end open-source* yang digunakan untuk membangun tampilan situs *web* dan aplikasi *web* responsif (*responsive web design*) yang dapat diakses melalui perangkat apapun, seperti desktop, tablet, atau *smartphone* (Febriyani, *et al.*, 2023).

Bootstrap menyediakan serangkaian gaya CSS (*Cascading Style Sheets*), JavaScript, dan komponen UI (*User Interface*) yang dapat digunakan untuk mempercepat proses pembuatan tampilan situs web. *Bootstrap* memiliki banyak fitur yang membuatnya populer di kalangan pengembang *web*, seperti *grid sistem*, *typography*, *form*, *button*, *icon*, *navbar*, *modal*, *carousel*, dan lain sebagainya. Selain itu, *Bootstrap* juga memiliki dokumentasi yang lengkap dan komunitas yang aktif untuk membantu pengembang *web* dalam menggunakan dan memahami kerangka kerja ini.



2.4.5 Pengertian MySQL

My Structured Query Language (MySQL) adalah suatu RDBMS (*Relational Data Base Management System*), yaitu aplikasi sistem yang menjalankan pengolahan data (Rapita, *et al.*, 2022).



Sumber: www.w3resource.com

Gambar 2.6 Lambang MySQL

My Structured Query Language (MySQL) adalah suatu sistem basis data *relation* atau *Relational Data Base Management System (RDBMS)* yang mampu bekerja secara cepat dan mudah digunakan (Renaldy, *et al.*, 2020).

2.4.6 Pengertian Sublime Text

Sublime text adalah *text editor* yang sangat populer karena fiturnya sangat lengkap. *Text editor* ini merupakan salah satu *text editor* yang canggih dan *powerfull*. *Sublime text* dapat digunakan diberbagai *platform* yang berbeda seperti *linux, mac os, dan windows* (Syauqi, *et al.*, 2022).



Sumber: www.nesabamedia.com

Gambar 2.7 Lambang *Sublime Text*



Sublime text merupakan perangkat lunak teks editor yang digunakan untuk membuat atau mengedit suatu aplikasi. *Sublime text* adalah teks editor berbasis *Python*, sebuah teks editor yang elegan, kaya fitur, *cross platform*, mudah dan *simple* yang cukup terkenal dikalangan *developer* (pengembang), penulis dan *desainer*. Para programmer biasanya menggunakan *sublime text* untuk menyunting *source code* yang sedang ia kerjakan (Louis, et al., 2022).

2.4.7 Pengertian XAMPP

XAMPP kependekan dari *X Apache MySQL PHP Perl*, X adalah sistem operasi (Windows, Linux, Unix), merupakan paket *software* yang terdiri dari *server web* (Apache), *database* (MySQL MariaDB), dan pengembangan aplikasi (PHP dan Perl) disebut juga sebagai *software stack*. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada *Windows* dan *Linux* (Sahdilla, 2021).



Sumber:logosandtypes.com

Gambar 2.8 Lambang XAMPP

XAMPP adalah *software* yang di dalamnya terdapat *server MySQL* dan didukung oleh PHP sebagai bahasa pemrograman untuk membuat *website* dinamis serta terdapat *web server apache* yang dapat dijalankan di beberapa *platform* seperti *OS X, Windows, Linux, Mac dan Solaris*. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada *Windows* dan *Linux*. Keuntungan lain yang didapatkan adalah hanya dengan melakukan instalasi cukup satu kali kemudian didalamnya tersedia



MySQL, Apache web server, Database server PHP dan beberapa modul lainnya (Sumenep, et al., 2023).

2.4.8 Pengertian *PhpMyAdmin*

PhpMyAdmin adalah aplikasi *web* yang dibuat oleh *phpmyadmin.net*. *PhpMyAdmin* digunakan untuk administrasi *database MySQL*, sebuah *software* berbasis pemrograman PHP yang dipergunakan sebagai administrator *MySQL* melalui *browser (web)* yang digunakan untuk *management database* (Abraham, et al., 2023).



Sumber: www.pngall.com

Gambar 2.9 Lambang *phpMyAdmin*

PhpMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *MySQL* melalui *world wide web* (Yanuar, et al., 2022).

PhpMyAdmin adalah perangkat lunak gratis (*freeware*) yang ditulis menggunakan bahasa pemrograman PHP, dengan dimaksudkan untuk menangani administrasi *database MySQL* melalui *interface Web*. *PhpMyAdmin* mendukung berbagai operasi di *database MySQL* dan *MariaDB*. Operasi yang sering digunakan seperti mengelola *database*, tabel, kolom, relasi, indeks, pengguna, izin, dan lainnya, dapat kita lakukan melalui antarmuka pengguna, sementara itu kita juga masih bisa menulis perintah SQL secara langsung untuk operasi pengelolaan *database*-nya.