

BAB II
LANDASAN TEORI

2.1. Tinjauan Pustaka

Didalam suatu penelitian diperlukan dukungan hasil – hasil dari studi terdahulu untuk mendukung penelitian ini yang berhubungan dengan Aplikasi Administrasi Surat Menyurat. Berikut adalah daftar literatur terdahulu :

Tabel 2. 1 Daftar Literatur

No	Nama Peneliti	Tahun	Judul
1	Lili Andraini dan Cintyha Bella	2022	Pengelolaan Surat Menyurat Dengan Sistem Informasi (Studi Kasus: Kelurahan Gunung Terang)
2	Nurul Ikhsan Dan Siti Ramadhani	2020	Sistem Informasi Administrasi Surat Menyurat Kantor Wilyaha Kementrian Agama Provinsi Riau
3	Eva Oktaviani, Rodianto, Sari Noviana, Dan Nawassyarif	2020	Rncang Bangun Sistem Infromasi Untuk Meningkatkan Tata Kelola Administrasi Surat Menyurat
4	Mohammad Sholikhul Huda Dan Nanik Susanti	2021	Sistem Informasi Pelayanan Surat Keterangan Pada Kantor Desa Honggosoco (SI SUKET)
5	Didin Agus Priyadi Dan Endah Wiji Lestari	2018	Perancangan Sistem Infromasi Pelayanan Surat Menyurat Pada Kantor Desa Tanjungsari Kutowinangun Kebumen Berbasis Desktop

2.1.1. Literatur 1

Oleh Lili Andriani dan Cinthya Bella (2022) dari program studi Manajemen dengan judul Pengelolaan Surat Menyurat Dengan Sistem Informasi (Studi Kasus: Kelurahan Gunung Terang). Dimana dalam penelitian ini menjelaskan Kantor Kelurahan Gunung Terang memiliki masalah yaitu mengeluhkan lamanya proses pembuatan surat sehingga memakan waktu yang tidak memiliki batasan waktu yang jelas sehingga peneliti mengembangkan sistem informasi surat menyurat berbasis web menggunakan metode pengembangan sistem metode *prototype* dan perencanaan sistem menggunakan UML. Implementasi sistem ini menggunakan software *sublime text* dan *database* menggunakan *MySQL* serta pengujian sistem menggunakan ISO 9126. Hasil dari penelitian ini yaitu menghasilkan laporan surat menyurat yang memiliki persentase keberhasilan dengan total rata – rata 96,61% dengan 10 responden untuk menguji kualitas kelayakan perangkat lunak menggunakan sistem pengujian ISO 9126.

2.1.2. Literatur 2

Oleh Nurul Ikhsan dan Siti Ramadhani (2020) dari Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim, Riau, Indonesia dengan judul Sistem Informasi Administrasi Surat Menyurat Kantor Wilayah Kementerian Agama Provinsi Riau. Dalam penelitian ini menjelaskan tentang Kantor Wilayah Kementerian Agama Provinsi Riau yang memiliki sebuah bagian penting dalam informasi pelayanan publik yaitu bagian Pusat Pelayanan Terpadu Satu (PTSP) yaitu sebuah bagian yang mengumpulkan segala bentuk informasi yang berhubungan dengan kegiatan dalam bidang informasi surat Di Kantor Wilayah

Kementrian Agama Provinsi Riau. Pegawai PTSP mengumpulkan dan mengolah data surat masuk dan surat keluar yang masih menggunakan cara konvensional. Maka dari itu peneliti akan membangun sebuah sistem informasi surat yang dapat memudahkan pegawai PTSP dalam mengelola data surat serta mengurangi kendala yang ada dan meningkatkan kinerja para pegawai. Sistem pengujian di dalam sistem ini menggunakan *black box* yang menghasilkan sistem tersebut berjalan dengan baik serta pengujian untuk pengguna di kantor wilayah kementrian agama provinsi riau menggunakan pengujian *User Acceptance Test (UAT)* dengan hasil keseluruhan adalah sistem tersebut mudah dipahami dan sesuai dengan apa yang diharapkan peneliti.

2.1.3. Literatur 3

Oleh Eva Oktaviani, Rodianto, Sari Oviana dan Nawassyarif (2020) mahasiswa Informatika Universitas Teknologi Sumbawa dengan judul Rancang Bangun Sistem Informasi Untuk Meningkatkan Tata Kelola Administrasi Surat Menyurat. Penelitian ini menjelaskan bahwa Kantor Kelurahan Uma Sima masih dikatakan kurang maksimal dan efisien dalam proses pembuatan surat yang dimana permasalahannya biodata pembuat surat sebelumnya harus di hapus terlebih dahulu lalu mengubah dan memasukkan biodata baru yang akan membuat surat keterangan tersebut sehingga banyak memakan waktu dan memperlambat kinerja karyawan. Dengan permasalahan tersebut, peneliti menawarkan sebuah sistem informasi untuk meningkatkan tata kelola administrasi di Kantor Kelurahan Uma Sima Kabupaten Sumbawa yang berbasis web. Sistem informasi tersebut dapat membantu dalam pembuatan surat, data surat keluar dan data surat masuk. Metode yang digunakan dalam penelitian ini menggunakan *waterfall* model dan sistem ini dibangun

menggunakan *Visual Studio Code* dengan bahasa pemrograman PHP dan MySQL sebagai *database*. Kemudian pengujian perangkat lunak menggunakan pengujian *black box*. Hasil akhir dari penelitian ini adalah sebuah sistem informasi administrasi surat menyurat yang berguna untuk membantu dalam pembuatan surat serta pengolahan surat masuk dan surat keluar.

2.1.4. Literatur 4

Oleh Muhammad Sholikhul Huda dan Nanik Susanti (2021) dari Program Studi Sistem Informasi Fakultas Teknik Universitas Muria Kudus dengan judul Sistem Informasi Pelayanan Surat Keterangan pada Kantor Desa Hanggosoco (SI SUKET). Pada penelitian ini membahas tentang surat menyurat yang menjadi salah satu jenis pelayanan yang terpenting yang ada di setiap kantor desa. Masalah yang terjadi pada penelitian ini para Kaur umum harus mengetik isi surat secara manual dan hal tersebut menghambat kinerja pelayanan surat keterangan di balai desa dan menyebabkan penumpukan antrian warga yang akan membuat surat keterangan. Dan terdapat masalah lain yaitu pengarsipan secara manual dengan buku beresiko terjadinya kehilangan data. Dengan permasalahan tersebut maka peneliti mengembangkan sistem informasi berbasis web yang dapat memberikan solusi terutama pada kegiatan pelayanan surat menyurat pada kantor desa. Metode yang digunakan dalam penelitian ini menggunakan metode *System development life cycle* (SDLC). Dengan tahapan *investigate, analyze, desain, implementasi* dan perawatan system. Hasil dari penelitian ini adalah aplikasi SI SUKET yang mampu membantu pihak kantor desa dalam menyediakan pelayanan pembuatan surat keterangan secara mandiri dan berbasis web (*online*).

2.1.5. Literatur 5

Oleh Didin Agus Priyadi dan Endah Wiji Lestari (2018) merupakan mahasiswa dari Akademik Manajemen Informatika Dan Komputer Bina Sarana Informatika dengan Judul Perancangan Sistem Informasi Pelayanan Surat Menyurat Pada Kantor Desa Tanjungsari Kutowinangun Kebumen Berbasis Desktop. Penelitian ini membahas tentang Kantor Desa Tanjungsari yang membutuhkan sistem yang mendukung dan memberikan pelayanan yang memuaskan bagi warga yang mana di Kantor Desa Tanjungsari sampai saat ini belum terkomputerisasi yang menyebabkan saat proses pelayanan terjadi kesalahan dalam pencatatan, ketidaktepatan laporan yang dibuat dan keterlambatan dalam pencarian data yang diperlukan. Sehingga peneliti merancang sistem yang terkomputerisasi dan akan memberikan solusi yang terbaik untuk mengatasi permasalahan tersebut. Metode pengembangan perangkat lunak yang digunakan dalam penelitian ini adalah menggunakan metode *waterfall* dengan lima tahapan yaitu analisa kebutuhan perangkat lunak, desain, pembuatan kode program, pengujian, dan pendukung (*support*) atau pemeliharaan (*maintenance*). Metode penelitian ini menggunakan jenis penelitian deskriptif. Hasil dari penelitian ini adalah aplikasi yang telah dirancang dapat digunakan untuk menangani proses pembuatan surat menyurat, *backup* data dan pencarian data serta pelaporan data penduduk berserta surat – surat.

2.2. Rancang Bangun

Rancang bangun merupakan suatu kegiatan yang dijalankan untuk memperoleh hasil analisa yang terbentuk di dalam perangkat lunak untuk

menciptakan suatu sistem baru atau menambah sejumlah fitur dari sistem sebelumnya (Bangun *et al.*, 2019).

2.3. Surat Menyurat

Menurut (RMDA, 2018) surat adalah sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lain. Informasi tersebut bisa berupa pemberitahuan, laporan, pernyataan, permintaan dan sebagainya.

2.4. Aplikasi

Aplikasi adalah suatu program yang berwujud perangkat lunak yang berjalan pada sistem tertentu yang berguna untuk membantu banyak kegiatan yang dilakukan oleh pengguna (Huda and Priyatna, 2019)

2.5. Sistem

Menurut (Marakas and O'Brien, 2017) sistem didefinisikan sebagai seperangkat komponen yang saling terkait, dengan batasan yang jelas, yang bekerja sama untuk mencapai tujuan tertentu dengan menerima masukan dan menghasilkan keluaran dalam proses transformasi yang terorganisasi. Sistem memiliki tiga fungsi dasar :

1. Masukan, mencakup pengambilan dan penyusunan elemen yang masuk ke dalam sistem untuk diproses
2. Pemrosesan, mencakup proses transformasi yang mengubah masukan menjadi keluaran.

3. Keluaran, mencakup mentransfer elemen yang telah diproduksi oleh proses transformasi menuju tujuan akhirnya.

2.6. Informasi

Menurut (Romney and Steinbart, 2016) informasi adalah data yang dikelola dan di proses untuk memberikan arti dan memperbaiki proses pengambilan keputusan.

2.7. Sistem Informasi

Menurut (Marakas and O'Brien, 2017) sistem informasi adalah kombinasi dari orang – orang, perangkat keras, perangkat lunak, jaringan komunikasi , sumber daya data, dan kebijakan serta prosedur dalam menyimpan, mendapatkan kembali, mengubah, dan menyebarkan informasi dalam suatu organisasi.

2.8. Kantor

Menurut (Zulkarnain and Sumarsono, 2018) kantor adalah serangkaian kegiatan atau aktivitas untuk menangani informasi, mulai dari menerima, mengumpulkan, mengolah, menyimpan sampai dengan menyalurkan.

2.9. Desa

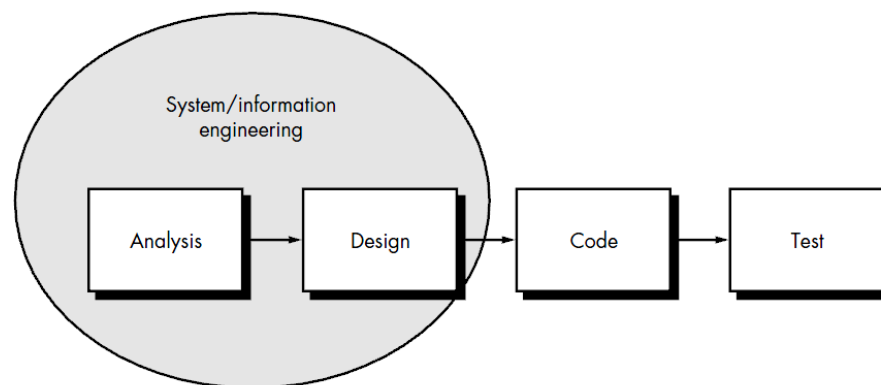
Menurut undang – undang nomor 6 tahun 2004 desa adalah kesatuan masyarakat hukum yang memiliki batas wilayah yang berwenang untuk mengatur dan mengurus urusan pemerintahan, kepentingan masyarakat setempat berdasarkan prakarsa masyarakat , hak asal usul, dan/atau tradisional yang diakui dan dihormati dalam sistem pemerintahan negara Kesatuan Republik Indonesia(Ramly *et al.*, 2018)

2.10. Administrasi

Menurut (Maisaroh and Danuri, 2020) administrasi adalah suatu kegiatan atau rangkaian kegiatan yang berupa proses pengelolaan usaha kerjasama sekelompok manusia yang tergabung dalam suatu organisasi untuk mencapai suatu tujuan bersama yang telah di tetapkan sebelumnya agar efektif dan efisien.

2.11. Metode pengembangan *Waterfall*

Menurut (A.S and Shalahuddin, 2018) model SDLC memiliki beberapa model dalam penerapan tahapan proses salah satu nya adalah model SDLC air terjun (*waterfall*) atau yang sering disebut dengan model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun ini dijalankan dengan terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung. Dibawah ini merupakan gambar tahapan dari model waterfall.



Gambar 2. 1 Ilustrasi proses metode waterfall

Sumber : A.S Rosa & Shalahuddin, 2018

2.11.1. Analisis kebutuhan perangkat lunak

Pada tahap ini dilakukan proses pengumpulan data yang dilakukan dengan cara mendetail untuk memspezifikasikan kebutuhan perangkat lunak agar mudah dipahami. Dan pada tahap ini perangkat lunak perlu untuk didokumentasikan

2.11.2. Desain

Pada tahap ini proses yang dilakukan yaitu mendesain perangkat lunak yang di dalamnya terdapat struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Dan pada tahap ini juga terdapat proses pemindahan kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program selanjutnya, dan proses ini perlu didokumentasikan juga.

2.11.3. Pembuatan kode program

Desain yang sudah dibuat sebelumnya harus dipindahkan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program yang dibuat sama dengan desain yang telah dirancang sebelumnya pada tahap desain.

2.11.4. Pengujian

Tahap pengujian difokuskan dari segi logik dan fungsional dan memastikan semua bagian sudah diuji karena pada tahap ini untuk memastikan tidak terdapat kesalahan dan memastikan hasil sesuai dengan yang diinginkan.

2.11.5. Pendukung (*Support*) atau pemeliharaan (*maintenance*)

Pada setiap perangkat memungkinkan terjadinya perubahan ketika sudah dikirimkan ke pengguna, dan perubahan tersebut bisa saja terjadi kesalahan yang muncul dan tidak terdeteksi pada saat pengujian sistem. Maka dari itu pada tahap ini pendukung atau pemeliharaan dapat mengulangi proses pengembangan dimulai dari analisis spesifikasi tapi tidak untuk perangkat lunak yang baru.

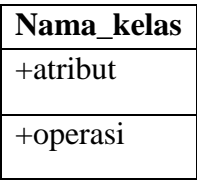
2.12. Unified Modelling Language (UML)

Menurut (A.S and Shalahuddin, 2018) *Unified Modelling Language* (UML) adalah sebuah teknik pemrograman berorientasi objek dengan standarisasi bahasa pemodelan untuk membangun perangkat lunak. UML hadir karena perangkat lunak membutuhkan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan mendokumentasikan sistem tersebut. UML adalah sebuah bahasa pemrograman untuk pemodelan dan komunikasi tentang sistem dengan menggunakan pendukung seperti diagram dan teks – teks. Serta UML paling banyak digunakan untuk metodologi berorientasi objek.

2.12.1. Class Diagram

Menurut (A.S and Shalahuddin, 2018) *class* diagram merupakan suatu pendefinisian kelas yang menggambarkan suatu struktur sistem yang dimana tujuan dari pendefinisian tersebut untuk membangun suatu sistem. Kelas diagram mempunyai atribut dan metode atau operasi. Di bawah ini merupakan simbol – simbol yang terdapat pada *class* diagram dapat dilihat pada tabel 2.1

Tabel 2. 2 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>association</i> 	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependency</i> 	Relasi antarkelas dengan makna keberantungan antarkelas
Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

Sumber : (A.S Rosa & Shalahuddin, 2018)

2.12.2. Use Case Diagram

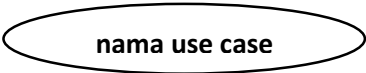

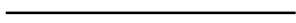
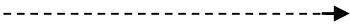
Menurut (A.S and Shalahuddin, 2018) *use case* atau *diagram use case* merupakan pemodelan untuk perilaku sistem informasi yang akan dibuat. *Use case* menjelaskan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Syarat untuk penamaan *use case*

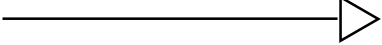
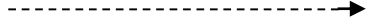
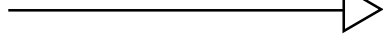
dijelaskan sesederhana mungkin agar mudah di mengerti. Ada dua hal utama dalam penamaan *use case*, yaitu :

- Aktor adalah orang, proses, atau sistem lain yang berkorelasi dengan sistem informasi yang akan dibuat diluar sistem informasi itu sendiri.
- *Use case* merupakan suatu upaya yang disediakan oleh sistem untuk unit - unit yang saling berhubungan antar unit atau aktor.

Berikut merupakan simbol – simbol yang ada pada diagram use case, dapat di lihat pada tabel 2.2

Tabel 2. 3 Simbol *Use Case* Diagram

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>suatu upaya yang disediakan oleh sistem untuk unit - unit yang saling berhubungan antar unit atau aktor</p>
<p>Aktor /actor</p> 	<p>orang, proses, atau sistem lain yang berkorelasi dengan sistem informasi yang akan dibuat diluar sistem informasi itu sendiri</p>
<p>Asosiasi/ <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi/<i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama</p>

	dengan <i>use case</i> yang ditambahkan
<p>Generalisasi/<i>generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah <i>use case</i> dimana fungsi yang lebih umum dari lainnya.
<p>Menggunakan/ <i>include/uses</i></p> <p style="text-align: center;"><<<i>include</i>>></p>  <p style="text-align: center;"><<<i>uses</i>>></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankannya <i>use case</i> ini


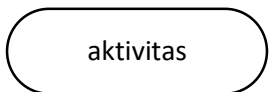
Sumber : (A.S Rosa & Shalahuddin, 2018)

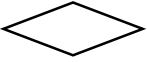

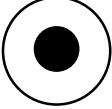
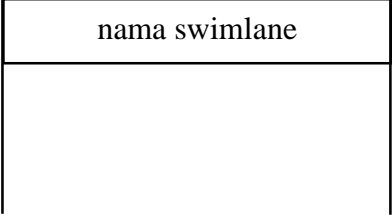
2.12.3. Activity Diagram

Menurut (A.S and Shalahuddin, 2018) *activity* diagram merupakan aliran kerja dari sebuah sistem yang ada pada perangkat lunak. Aktivitas disini menggambarkan sebuah aktivitas sistem bukan yang dilakukan oleh aktor melainkan di lakukan oleh sistem itu sendiri.

Berikut ini adalah simbol- simbol yang ada pada *activity* diagram dapat dilihat pada tabel 2.3:

Tabel 2. 4 Simbol *Activity* Diagram

Simbol	Deskripsi
<p>Status awal</p> 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
<p>Aktivitas</p> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja





Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakuka sistem, sebuah diagram aktivitas memiliki sebuah status akhir
swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktiivtas yang terjadi



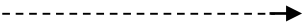
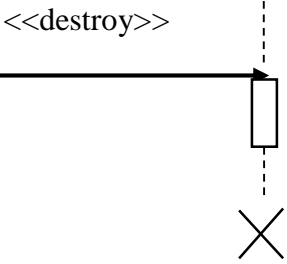
Sumber : (A.S Rosa & Shalahuddin, 2018)

2.12.4. Sequence Diagram

Menurut (A.S and Shalahuddin, 2018) Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek –objek yang terlibat dalam sebuah *use case* beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Berikut merupakan simbol – simbol yang ada pada diagram sekuen dapat dilihat pada tabel 2.5 :

Tabel 2. 5 Simbol Sequence Diagram

Simbol	Deskripsi
<p>aktor</p>  <p>nama_aktor</p> <p>Atau</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">nama_aktor</div> <p>Tanpa waktu aktif</p>	<p>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 aktor</p>
<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama objek : nama kelas</div>	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya</p>
<p>Pesan tipe create</p> <p><<create>></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>

<p>Pesan tipe <i>call</i></p> <p>1 : nama_metode() </p>	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>
<p>Pesan tipe <i>send</i></p> <p>1 : masukan </p>	<p>Menyatakan bahwa suatu objek mengirimkan data/asukan.informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> <p>1 : keluaran </p>	<p>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</p>
<p>Pesan tipe <i>destroy</i></p> <p><<destroy>> </p>	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhir, sebaiknya jika ada create maka ada <i>destroy</i></p>

Sumber : (A.S and Shalahuddin, 2018)

2.13. Black box testing

Menurut (A.S and Shalahuddin, 2018) *black box testing* yaitu sistem pengujian perangkat lunak yang secara fungsional tanpa menguji desain dan kode program. Pengujian ini bertujuan untuk melihat apakah fungsi, *input*, *output* dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan

spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji yang dibuat adalah :

1. Jika pengguna login dengan username dan password yang benar.
2. Jika pengguna login dengan username benar dan password yang salah atau sebaliknya.

Adapun kerangka yang akan digunakan untuk melakukan pengujian dapat dilihat pada tabel 2.5

Tabel 2. 6 Kerangka Pengujian

Kasus dan Hasil Uji			
Data Masukan	Data Diharapkan	Pengamatan	Kesimpulan
			Diterima () Ditolak ()
			Diterima () Ditolak ()

Rumus dari kuisioner pengujian black box menggunakan analisa deskriptif ditunjukkan sebagai berikut :

$$\% \text{ Skor} = \text{skor aktual/skor ideal} \times 100\% \dots\dots\dots(1)$$

Ket :

Skor aktual : Jawaban diterima responden

Skor ideal : Total jumlah butir soal yang telah diujikan kepada responden

Adapun kriteria hasil perhitungan kelayakan sistem dari pengujian ini dapat dilihat pada tabel 2.7

Tabel 2. 7 Kriteria Skor Tanggapan Responden

Jumlah Skor	Kriteria
0.00 – 36.00	Tidak layak
36.01 – 52.00	Kurang layak
52.01 – 68.00	Cukup layak
68.01 – 84.00	Layak
84.01 - 100	Sangat Layak

2.13. XAMPP

XAMPP merupakan singkatan dari *Multi Platform (X)*, *Apache (A)*, *MySQL (M)*, *PHP (P)*, *Perl (P)*. XAMPP merupakan aplikasi yang simple, ringan dan sangat mudah untuk dioperasikan sebagai *developer web* untuk membuat *web server* lokal dengan berbagai macam kebutuhan. XAMPP bisa berjalan di berbagai macam OS seperti *windows*, *Linux*, maupun *Mac OS*. Dibawah ini merupakan komponen yang terdapat pada XAMPP.

- a. *Apache*, dibuat dan di *maintenance* oleh *Apache Software Foundation* dan dapat digunakan dengan gratis. *Apache* sebagai sebuah *web server* yaitu, memroses permintaan dari *client* dan meneruskannya ke halaman yang dituju melalui *protocol HTTP*.
- b. *MySQL* adalah salah satu aplikasi *DBMS(Database Management System)* yang sangat populer dan gratis
- c. *PHP* adalah singkatan dari *Hypertext Preprocessor*, *PHP* adalah bahasa pemrograman web yang berjalan pada sisi *server*, sangat *powerfull* bila

dikombinasikan dengan MySQL, PHP juga sangat populer di dunia dan dipakai oleh beberapa perusahaan besar seperti *wordpress* dan *facebook*.

- d. *Perl*, komponen yang sangat jarang disentuh. *Perl* juga sebuah bahasa pemrograman tingkat tinggi seperti PHP dan bisa juga dipakai untuk membuat aplikasi *web*. Tokopedia adalah salah satu perusahaan yang memakai *perl* untuk aplikasi *web*-nya
- e. *PhpMyAdmin*, *FileZilla*, Dll merupakan komponen lain yang tak kalah penting adalah *PhpMyAdmin* yang bisa kita gunakan sebagai mengelola *database*. *FileZilla* juga bisa digunakan untuk membuat sebuah *file server*.

2.14. *CodeIgniter*

Menurut (Supono and Putratama, 2018) *Codeigniter* adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. *Codeigniter* memudahkan pengembang untuk membuat aplikasi web dengan cepat dan mudah dibanding harus membuat dari awal. Terdapat tiga jenis komponen dari *codeigniter* untuk membangun sebuah *website* dinamis, diantaranya :

- Model, berhubungan langsung dengan *database* untuk memanipulasi data (*insert, update, delete, search*), menangani validasi, namun tidak berhubungan langsung dengan *view*.
- *View*, merupakan bagian yang menangani *presentation logic*. Biasanya didalam suatu web bagian ini berupa file template HTML, yang di atur oleh *controller*. *View* berfungsi untuk menerima dan

merepresentasikan data kepada *user*. Bagian ini tidak memiliki akses langsung terhadap model.

- *Controller*, merupakan bagian yang mengatur hubungan antara bagian model dan bagian model dan bagian *view*, *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan di proses oleh aplikasi

Terdapat beberapa kelebihan CodeIgniter (CI), diantaranya :

- a. Performa sangat cepat, menjadi salah satu alasan menggunakan *CodeIgniter* dibandingkan dengan *framework* yang lain
- b. Konfigurasi yang sangat minim, tentu saja untuk menyesuaikan dengan *database* dan keleluasaan *routing* tetap diizinkan melakukan konfigurasi seperti *database.php* atau *autoload.php*, dalam menggunakan CI hanya perlu sedikit mengubah *setting* standar saja pada folder *config*.
- c. Memakai konsep MVC, dengan adanya MVC pengerjaan antara logika dengan *layout* telah terpisahkan, sehingga antara *programmer* dan *designer* dapat santai melakukan tugasnya.

2.15. PHP

Menurut (Supono and Putratama, 2018) PHP (*Hypertext Preprocessor*) adalah sebuah bahasa pemrograman yang berguna untuk menerjemahkan kode program menjadi sebuah bahasa mesin yang dimengerti oleh komputer yang

bersifat *server-side* yang ditambahkan ke dalam HTML. PHP memiliki kelebihan sebagai bahasa pemrograman. Berikut adalah kelebihan yang dimiliki PHP :

- a. PHP adalah bahasa *multiplatform* yang artinya dapat berjalan di berbagai mesin dan sistem operasi dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah – perintah sistem lainnya.
- b. Bersifat *opensource* yang berarti dapat digunakan oleh siapa saja secara gratis.
- c. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
- d. Dapat mendukung banyak *database*, seperti MySQL, Oracle, MS-SQL, dst.
- e. *Web server* yang mendukung PHP dapat ditemukan di mana-mana dari mulai *apache*, IIS, *Lightpad*, *nginx*, hingga *Xitami* dengan konfigurasi yang *relative* mudah dan tidak berbelit-belit, bahkan yang membuat dalam bentuk paket.
- f. Dalam sisi pengembangan lebih mudah, karena banyaknya *milis – milis*, komunitas dan *developer* yang siap membantu dalam pengembangan.
- g. Banyak bertebaran aplikasi dan Program PHP yang gratis dan siap pakai seperti *Wordpress*, *Pertasshop*, dan lain- lain.

PHP juga memiliki beberapa kekurangan didalamnya, yaitu :

- a. PHP tidak mengenal *package*
- b. Jika tidak di-*encoding*, maka kode PHP dapat dibaca semua orang dan untuk meng-*encoding*-nya dibutuhkan tool dari Zend yang mahal sekali biayanya.

- c. PHP memiliki kelemahan keamanan, jadi *programmer* harus jeli dan berhati-hati dalam melakukan pemrograman dan konfigurasi PHP

2.16. MySQL

Menurut (Supono and Putratama, 2018) MySQL adalah manajemen *database* SQL yang bersifat *open source* yang mendukung beberapa fitur seperti *multithreaded*, *multi user*, dan *SQL database* manajemen sistem (DBMS). *Database* ini dibuat dengan tujuan untuk keperluan sistem *database* yang cepat, terpercaya, dan mudah digunakan.

2.17. Database

Menurut (A.S and Shalahuddin, 2018) *database* atau bisa juga disebut dengan basis data merupakan sebuah sistem yang terkomputerisasi yang bertujuan untuk memelihara data yang sudah di olah atau informasi dan membuat informasi tersedia saat dibutuhkan. Yang pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

2.18. Web

Menurut (Elgamor, 2020) *Web* atau bisa juga disebut dengan *website* merupakan sebuah media yang memiliki banyak halaman yang saling terhubung, dimana *website* memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara dan animasi atau penggabungan dari semuanya.

2.19. User Acceptance Test

Pengujian User Acceptance Test dilakukan dengan cara memberi beberapa pertanyaan atau kuisisioner kepada pengguna akhir dan pada pengujian user

acceptance dilakukan pada akhir ketika sistem telah selesai dibangun(Pujianto *et al.*, 2020). Tujuan dari sistem pengujian ini adalah untuk mengetahui apakah sistem ini dapat memenuhi kebutuhan pengguna berdasarkan permasalahan yang ada di latar belakang penelitian ini(Hady, Haryono and Rahayu, 2020). Untuk penilaian terhadap pertanyaan terdapat bobot nilai untuk setiap jawaban adalah sebagai berikut :

Tabel 2. 8 Penilaian Bobot Terhadap Jawaban

Skala jawaban	Keterangan	Skor	Persentase
SS	Sangat setuju	5	81% - 100%
S	Setuju	4	61% - 80%
C	Cukup	3	41% - 60%
TS	Tidak setuju	2	21% - 40%
STS	Sangat tidak setuju	1	0% - 20%

Sumber : (Pratama, 2019)