

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Aplikasi surat masuk dan surat keluar telah banyak dibangun dan diimplementasikan. Salah satu penelitian yang menerapkan aplikasi surat masuk dan surat keluar yang dilakukan oleh penelitian terdahulu diantaranya.

Menurut penelitian Khairunnissa dan Afriyadi (2023) dengan judul *Perancangan Aplikasi Pengelolaan Surat Berbasis Web di Badan Pengelola Pajak dan Retribusi Daerah Kabupaten Sarolangun*. Berdasarkan pengamatan dari penelitian yang dilakukan perlu adanya suatu aplikasi yang dapat membantu kinerja dalam proses pengelolaan surat. dengan tujuan untuk meminimalisir permasalahan kehilangan data, juga dapat digunakan sebagai backup penyimpanan manual. Penyimpanan data surat bisa dilakukan di database pada sistem pengelolaan surat. Penelitian ini menggunakan metode Waterfall, sistem yang dikembangkan berbasis website dengan menggunakan MySQL, HTML, database, bahasa pemrograman PHP.

Menurut penelitian Aرسالina dan Marlinda (2022) dengan judul *Aplikasi Pengelolaan surat masuk dan surat keluar berbasis web di PDAM Tirta Musi Palembang*. Berdasarkan dari penelitian yang dilakukan pada PDAM Tirta Musi Palembang penyimpanan surat masuk dan surat keluar masih tergolong kurang baik, dikarenakan berkas menumpuk menjadi satu membuat pencarian data memakan waktu yang cukup lama, dengan adanya aplikasi pengolahan surat masuk dan surat keluar berbasis web dapat meminimalisir waktu dalam pencarian data. Data juga terjaga dan terjamin keberadaannya. Penelitian ini menggunakan metode Waterfall, sistem yang dikembangkan berbasis website dengan menggunakan MySQL dan bahasa pemrograman PHP.

Menurut Penelitian Sugiartika dkk., 2020) dengan judul *“Aplikasi pengelolaan surat masuk dan surat keluar di kantor Dinas Komunikasi dan Informatika Kabupaten Ciamis”* berdasarkan dari penelitian yang dilakukan pada Dinas Komunikasi dan Informatika Kabupaten Ciamis, perlu dibangun suatu Aplikasi Pengelolaan Surat Masuk dan Surat Keluar yang dapat memfasilitasi petugas untuk mengelola surat dengan mudah tanpa

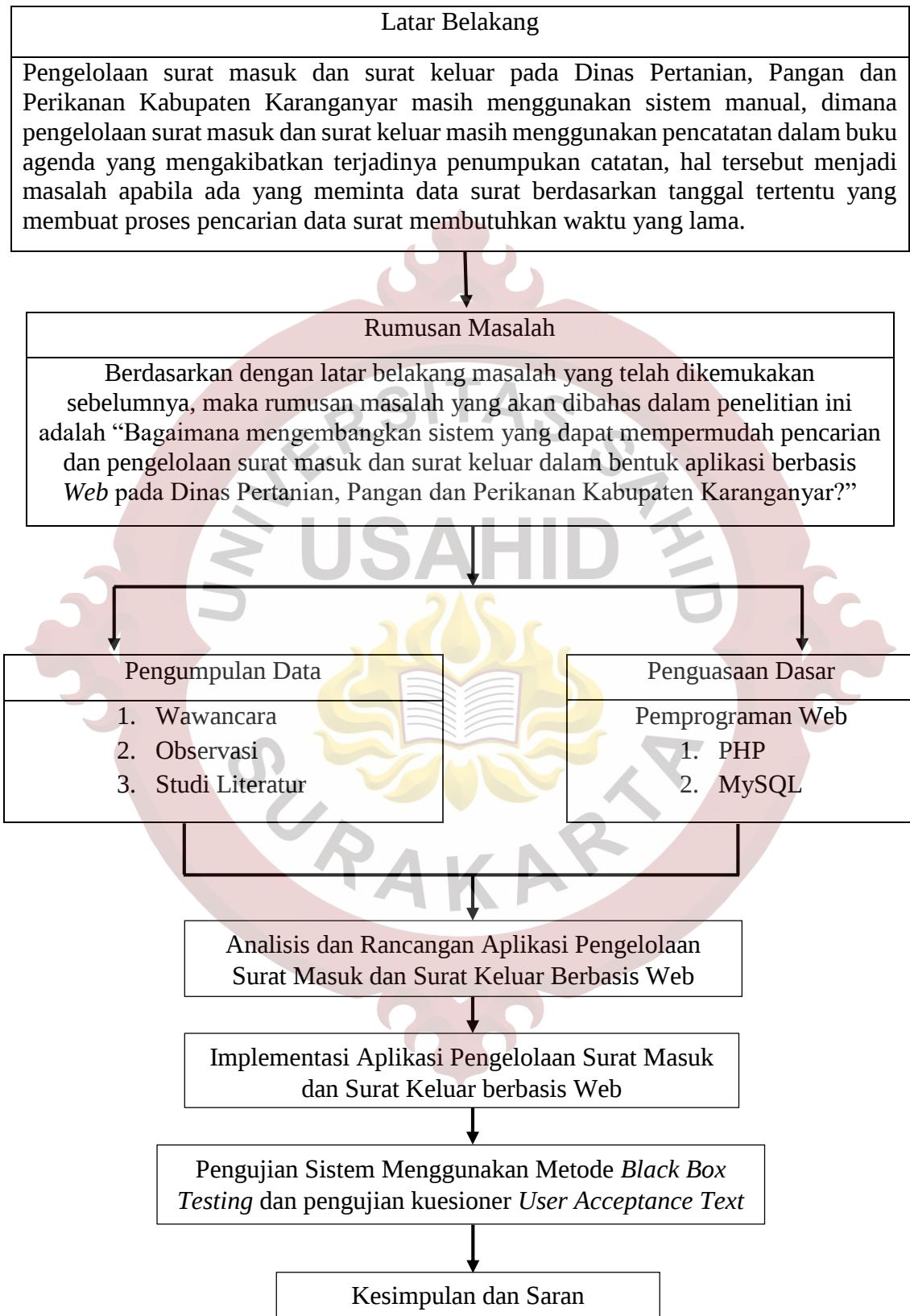
adanya hambatan seperti pada pengelolaan surat secara manual. Metodologi yang digunakan pada aplikasi ini adalah metode waterfall. Bahasa pemrograman yang digunakan untuk mengembangkan aplikasi ini menggunakan Bahasa HTML, Database MySQL, dan Framework Codeigniter. Pengujian sistem menggunakan *User Acceptance Test* dengan *Black Box Testing*

Menurut penelitian Prayoga dkk., (2024) dengan judul “Aplikasi Surat Masuk dan Surat Keluar Berbasis Web pada Badan Pengelola Keuangan Aset Daerah Kota Palangka Raya”. Berdasarkan dari penelitian yang dilakukan pengelolaan surat masuk dan surat keluar masih menggunakan cara manual. Metode yang digunakan dalam penelitian ini adalah metode *SDLC*. Desain proses menggunakan *Unified Modeling Language* (UML), menggunakan bahasa pemrograman PHP dan menggunakan MySQL sebagai pengolahan databasenya. Hasil penelitian menunjukkan bahwa dengan sistem berbasis web, risiko kehilangan atau kerusakan dokumen fisik dapat dikurangi secara signifikan dan surat-surat yang diarsipkan dalam format digital akan lebih aman.

Menurut Penelitian Pritasari dkk., (2020) dengan judul “Aplikasi arsip surat berbasis web pada Dinas Kesehatan Kabupateng Pandeglang” Proses pengelolaan arsip surat masih menggunakan cara manual, mulai dari pencatatan, disposisi hingga letak penyimpanan arsip. Tujuan dilakukannya penelitian ini adalah membuat aplikasi arsip surat berbasis web untuk memudahkan pengguna dalam mengelola data surat, juga untuk mewujudkan visi Pandeglang satu data. Penelitian ini dilakukan dengan menggunakan metode perancangan sistem *Waterfall*, sedangkan pemodelan sistem menggunakan *Unified Modeling Language* (UML).

2.2 Kerangka Berpikir

Aplikasi pengelolaan surat masuk dan surat keluar berbasis web merupakan aplikasi yang dapat memberi kemudahan pada pegawai Sub Bagian Umum dan Kepegawaian dalam proses pengelolaan surat masuk dan surat keluar, yang dimana sebelumnya Dinas Pertanian, Pangan dan Perikanan Kabupaten Karanganyar masih menggunakan pencatatan manual pada buku. Adapun kerangka berpikir dari rancang bangun aplikasi surat masuk dan surat keluar dapat dilihat pada Gambar 2.1 dibawah ini.



Gambar 2. 1 Kerangka Berfikir

2.3 Teori Pendukung

2.3.1 Aplikasi

Menurut Yulisman, (2020) Aplikasi adalah suatu perangkat untuk mempermudah pekerjaan yang dilakukan oleh individu, organisasi maupun institusi seperti untuk pengolahan data dan penyimpanan data.

2.3.2 Aplikasi Berbasis Web

Menurut Suryawinata, (2019) Aplikasi berbasis web adalah sebuah program computer yang memanfaatkan web browser serta teknologi web untuk melakukan tugas-tugas melalui internet. Aplikasi berbasis web menggunakan kombinasi *server-side script* seperti aspek maupun PHP untuk menangani penyimpanan dan mendapatkan informasi.

2.3.3 Website

Menurut Widia & Asriningtias, (2021) dalam jurnalnya “Website adalah kumpulan dokumen berupa halaman web yang berisi teks dalam format *Hyper Text Marckup Language* (HTML). Website disimpan di server hosting yang dapat diakses menggunakan *Browser* dengan jaringan internet melalui alamat internet berupa *Uniform Resource Locator* (URL)”.

2.3.4 Pengertian Surat

Menurut Aرسالina & Marlinda, (2022) dalam jurnalnya “Surat adalah alat komunikasi penyampaian informasi secara tertulis dari satu pihak ke pihak lain. Informasi yang disampaikan tersebut dapat berupa pemberitahuan, pernyataan, pertanyaan, permintaan, laporan, pemikiran, sanggahan, kritikan dan sebagainya. Bila suatu surat mendapat jawaban atau balasan maka terjadilah hubungan surat menyurat yang sering disebut korespondensi. Peranan surat menyurat sangat penting dalam suatu organisasi, baik organisasi kecil, sedang, ataupun besar. Surat berfungsi sebagai alat informasi tertulis mengenai suatu hal tertentu secara jelas dan tertulis dan harus terperinci”.

2.3.5 Pengertian Surat Masuk

Menurut penelitian Rachman, (2022) “Surat masuk adalah surat yang diterima oleh suatu Instansi atau organisasi yang berasal dari seseorang atau dari suatu instansi. Pengelolaan surat masuk adalah semua kegiatan yang dilakukan mulai dari penerimaan surat masuk, pengelolaan, hingga surat tersebut disimpan atau diarsipkan. Prosedur pengelolaan surat masuk meliputi penerimaan, pengarahannya, penilaian, pencatatan, penyimpanan dan penyampaian”.

2.3.6 Pengertian Surat Keluar

Menurut Rachman, (2022) dalam jurnalnya “Surat keluar adalah surat yang dikirimkan sebagai jawaban atau tanggapan atas isi surat masuk yang diterima dari organisasi atau instansi agar terjalin komunikasi dua arah agar saling menguntungkan dari kedua belah pihak. Dalam definisi lain surat keluar surat yang dibuat dan dikirimkan kepada suatu instansi atau lembaga manapun yang berupa informasi yang ingin disampaikan. Prosedur pengelolaan surat keluar meliputi penyampaian, pengelompokan, pencatatan, penyampaian, pengiriman dan pendistribusikan”.

2.3.7 PHP

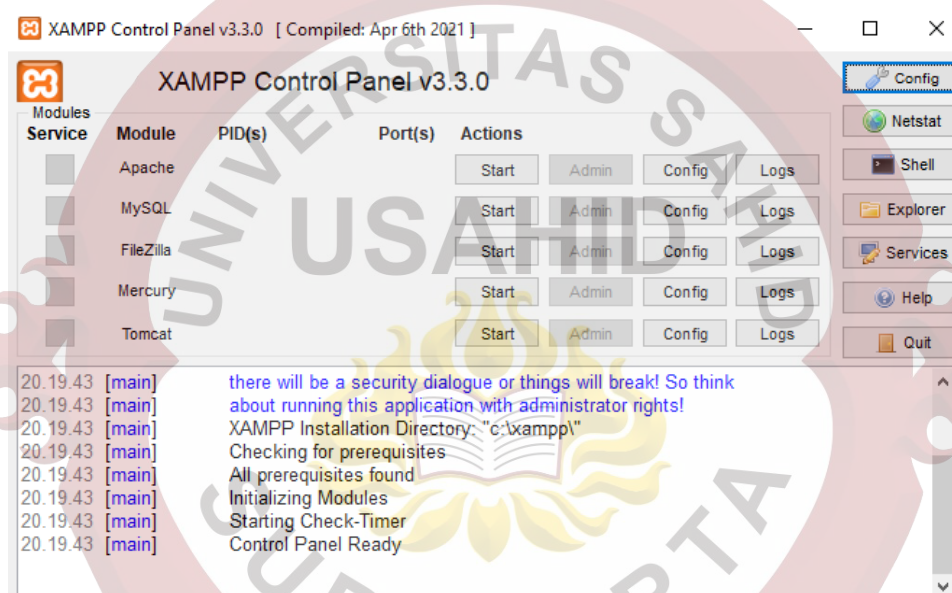
Menurut Hidayat dkk., (2019) dalam jurnalnya “PHP merupakan bahasa server-side yang menyatu dengan html, untuk membuat halaman web yang dinamis, salah satu fungsinya adalah untuk menerima dan mengolah dan menampilkan data ke sebuah situs, data yang diterima akan diolah disebuah program databases server, untuk kemudian hasilnya ditampilkan kembali ke layar browser sebuah situs”.

2.3.8 MySql

Menurut Siregar dkk., (2023) dalam jurnalnya “MySQL merupakan sistem manajemen basis data SQL bersifat Open Source. Dengan kata lain MySql dapat didefinisikan sebagai manajemen database. Database sendiri merupakan struktur penyimpanan data untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah database computer diperlukan sistem manajemen database seperti MySql server)”.

2.3.9 XAMPP

Menurut Abdullah dkk., (2020) dalam jurnalnya “Xampp merupakan singkatan dari Apache, MySQL, PHP, dan Perl sedangkan huruf “X” adalah suatu perangkat lunak yang dapat dijalankan di empat OS utama seperti windows, Mac OS, Linux dan Solaris. Istilah ini seringkali disebut dengan cross platform (*Software* multi OS). Dalam arti lain XAMPP adalah sebuah software yang berperan sebagai web server pada komputer atau Istilah ini sering disebut localhost”. Dapat dilihat pada Gambar XAMPP dapat ditampilkan pada Gambar 2.2



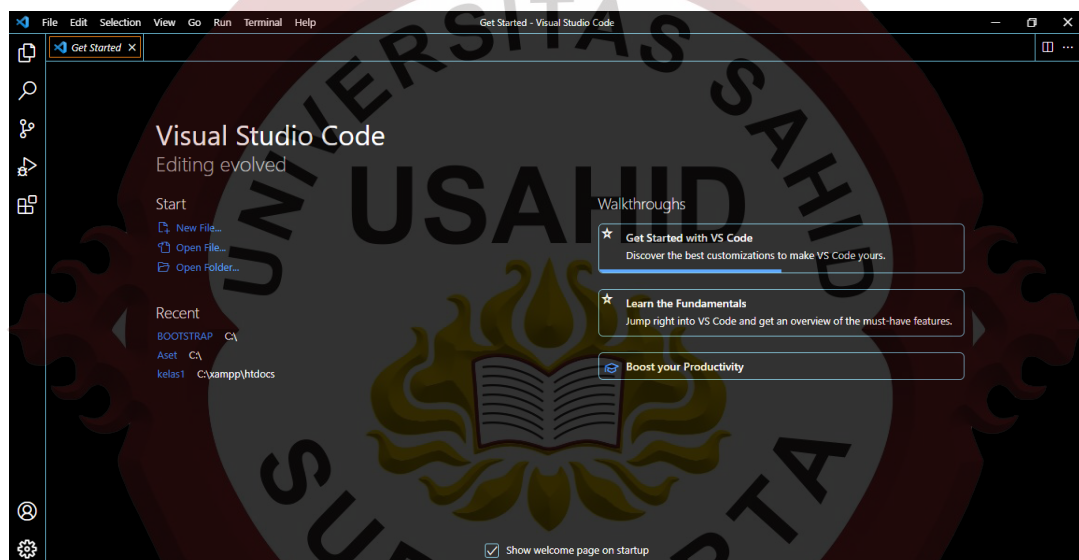
Gambar 2. 2 XAMPP

2.3.10 Bootstrap

Menurut Abdulloh, (2022) dalam jurnalnya “Bootstrap adalah salah satu 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. Bootstrap juga membuat proses pengaturan desain menjadi lebih cepat karena tidak perlu lagi banyak menulis CSS, bahkan hampir tidak perlu kecuali jika memerlukan pengaturan desain yang berbeda dengan *style* Bootstrap. Bootstrap telah didukung oleh hampir semua *browser* baik desktop maupun *mobile*”.

2.3.11 Visual Studio Code

Menurut Salamah, (2021) “*Visual Studio Code* adalah sebuah text editor ringan yang dibuat oleh Microsoft untuk sistem operasi *Multiplatform*, artinya tersedia juga untuk versi Linux, MacOS dan Windows. Teks editor mendukung banyak bahasa pemrograman seperti JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang di *Visual Studio Code* seperti C++, C#, Python, Go, dan java. *Visual Studio Code* bersifat *open source*”. Dapat dilihat pada Gambar *Visual Studio Code* dapat ditampilkan pada Gambar 2.3



Gambar 2.3 Visual Studio Code

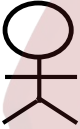
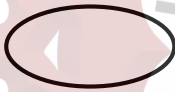
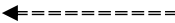
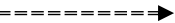
2.3.12 Unified Modeling Language (UML)

Menurut Nugroho dalam jurnal Ferdika & Kuswara, (2017) mengemukakan bahwa, ”UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma, berorientasi objek. Pemodelan (*modeling*) digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Beberapa diagram yang digunakan diantaranya.

2.3.12.1 Use Case Diagram

Menurut Sukamto & Shalahudin, (2018) “*Use Case Diagram* merupakan pemodelan untuk kelakuan (*Bahavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem informasi yang akan dibuat. *Use Case* digunakan untuk mengetahui fungsi atau proses apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu”. Simbol *Use Case* dapat dilihat pada Tabel 2.1

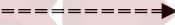
Tabel 2. 1 Simbol Use Case Diagram

Simbol	Nama	Keterangan
	<i>Actor</i>	Simbol Ini Mengidentifikasi pengguna sistem untuk melakukan sesuatu.
	<i>Use Case</i>	Simbol ini mengidentifikasi interaksi antara <i>actor</i> dengan sistem
	<i>Assosiation</i>	Simbol ini digunakan untuk menghubungkan <i>actor</i> dengan <i>use case</i>
	<i>Generalization</i>	Simbol ini mendefinisikan hubungan antara dua <i>actor</i> atau <i>Use Case</i>
<< Include >> 	<i>Include</i>	Simbol ini menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>usecase</i> lainnya
<< Extend >> 	<i>Extend</i>	Simbol ini menunjukan bahwa suatu <i>use case</i> merupakan tambahan fungsionalitas dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

2.3.12.2 Class Diagram

Menurut Sukanto & Shalahudin, (2018) “*Class Diagram* atau kelas diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas”. Simbol *Class Diagram* dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Simbol Class Diagram

Simbol	Nama	Keterangan
	<i>Association</i>	Menghubungkan antara objek satu dengan objek yang lainnya.
	<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas
	<i>Generalization</i>	Relasi antar kelas dimana objek anak (<i>Descendent</i>) berbagi perilaku dan struktur data objek yang ada di atasnya objek induk (<i>Ancestor</i>)
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaoration</i>	Deskripsi diri urutan aksi-aksi yang telah ditampilkan oleh sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.

2.3.12.3 Activity Diagram

Menurut Sukamto & Shalahudin, (2018) “ *Activity Diagram* atau aktivitas diagram menggambarkan *Workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. Simbol *activity diagram* dapat dilihat pada Tabel 2.3.

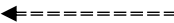
Tabel 2.3 Simbol Activity Diagram

Simbol	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Process</i>	Pilihan untuk mengambil keputusan.
	<i>Start State</i>	Bagaimana objek dibentuk dan diawali.
	<i>End State</i>	Bagaimana objek dibentuk dan diakhiri.
	<i>Fork Node</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama

2.3.12.4 Sequence Diagram

Menurut Sukanto & Shalahudin, (2018) “*Sequence Diagram* menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek dan *Message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *Sequence Diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *Use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat *Sequence Diagram* juga dibutuhkan untuk melihat skenario yang ada pada *Use Case*”. Simbol *Sequence Diagram* dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Simbol Sequence Diagram

Simbol	Nama	Keterangan
	<i>Objek lifeline</i>	Simbol ini mengidentifikasi objek apa saja yang terlibat
	<i>Actor</i>	Simbol ini mengidentifikasi hubungan antara objek dengan <i>message</i>
	<i>Activation</i>	Simbol ini mengidentifikasi hubungan antara <i>actor</i> yang terlibat
	<i>Message (call)</i>	Simbol ini mengidentifikasi alur <i>message</i> yang merupakan kejadian objek pengirim <i>lifeline</i> ke objek penerima <i>lifeline</i> .
	<i>Message (return)</i>	Symbol ini mengidentifikasi alur pengambilan <i>message</i> ke objek pemanggil dan tanda bahwa objek penerima telah menyelesaikan prosesnya.

2.3.12.5 Deployment Diagram

Dikutip dari penelitian Widialvana, (2022) Deployment Diagram merupakan suatu diagram yang menunjukkan suatu konfigurasi dari sebuah komponen dalam proses menjalankan suatu aplikasi dan memperlihatkan bagian-bagian pada perangkat lunak (*Software*) yang berjalan pada bagian perangkat keras *Hardware*. Deployment Diagram juga dapat digunakan sebagai permodelan lain seperti sistem tambahan dan sistem *client* atau server Deployment Diagram dapat dilihat pada Tabel 2.5

Tabel 2. 5 Simbol Deployment Diagram.

Simbol	Nama	Keterangan
	Package	Package merupakan sebuah bungkus dari satu atau lebih node
	Node	Node merupakan sebuah unit yang dapat membentuk proses
	Dependency	Merupakan sebuah garis yang berfungsi menunjukkan operasi pada suatu component yang menggunakan component yang lain.
	Link	Merupakan suatu garis yang berfungsi menghubungkan obyek satu dengan obyek lain.

2.3.12.6 Pengujian Sistem Black Box Testing

Menurut Mintarsih, (2023) Pengujian *Black Box Testing* Pengujian *Black Box Testing* adalah pengujian yang memverifikasi hasil eksekusi aplikasi berdasarkan masukan yang diberikan (data uji) untuk memastikan fungsional dari aplikasi sudah sesuai dengan persyaratan. Pengujian *Black Box Testing* ialah pengujian yang berfokus pada interface atau tampilan dan pengujian fungsional yang terdapat pada aplikasi, serta kesesuaian pada alur fungsi yang dibutuhkan oleh pengguna. Pengujian *Black Box Testing* tidak menguji berdasarkan *source code* program. Pengujian *Black Box Testing* dilakukan mengikuti tahapan berikut ini:

- 1 Membuat *Test Case* untuk pengujian fungsi-fungsi yang terdapat pada aplikasi

- 2 Membuat *Test Case* untuk pengujian kesesuaian *Flow* atau alur dari kerja suatu fungsi pada program cocok dengan apa yang dibutuhkan dan permintaan dari pengguna
- 3 Mencari *bugs/error* berdasarkan tampilan (*Interface*) pada aplikasi Dalam melakukan pengujian harus memilih teknik yang tepat, yaitu teknik yang dapat menemukan kesalahan yang belum terdeteksi sehingga dapat meningkatkan kualitas *Software*.

2.3.12.7 User Acceptance Test (UAT)

Menurut Naufal, (2023) dalam penelitiannya “*User Acceptance Test (UAT)* merupakan proses pengujian oleh pengguna untuk menghasilkan sebuah aplikasi yang telah dibangun, dapat diterima oleh pengguna. Proses pengujian UAT menggunakan alat kuesioner untuk menentukan tingkat keberhasilan sistem yang telah dibangun sudah sesuai dengan rancangan dan kebutuhan dari pengguna. Metode perhitungan pada pengujian UAT ini menggunakan metode skala likert. Pengujian dengan metode ini dapat diaplikasikan pada penilaian yang berdasarkan pendapat responden terhadap pertanyaan yang diajukan demi kepentingan penelitian. Pertanyaan tersebut pada umumnya telah ditentukan berdasarkan fenomena sosial yang dibahas oleh peneliti, yang kemudian akan dijadikan sebagai variabel penelitian”. Berikut adalah rumus menghitung skor pengujian *User Acceptance Test (UAT)* berdasarkan rumus perhitungan menggunakan *skala likert* dan kriteria skor yang dijelaskan pada Tabel 2.6

$$P = \frac{S}{\text{Skor Ideal}} \times 100\%$$

Keterangan :

P = Nilai persentase yang dicari

S = Jumlah frekuensi dikalikan nilai yang ditetapkan jawaban

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel

Tabel 2.6 Tabel Kriteria Skala likert

Skala Jawaban	Skor	keterangan	Nilai
SS	5	Sangat Setuju	81% - 100%
S	4	Setuju	61% - 80%
CS	3	Cukup Setuju	41% - 60%
TS	2	Tidak Setuju	21% - 41%
STS	1	Sangat Tidak Setuju	0% - 20%