

BAB II

LANDASAN TEORI

2.1 Tinjauan Terhadap Penelitian Terkait

Berikut beberapa penelitian terkait referensi pada penelitian ini :

Diana Grace, Mu'amar S. Tanciga, dan Nurdin Nurdin (2018) dalam penelitiannya yang berjudul **“Sistem Informasi Letak Geografis Penentuan Jalur Tercepat Rumah Sakit Di Kota Palu Menggunakan Algoritma Greedy Berbasis Web”**

Teknologi membuat manusia dapat mengembangkan pola pikir mereka menjadi lebih baik dari sebelumnya. Suatu lokasi yang dulunya ditempuh dengan waktu yang lama kini mereka sudah bisa ditempuh dengan waktu yang singkat. Ini disebabkan oleh pola pikir mereka yang semakin berkembang pula. Pemanfaatan berbagai jalur untuk mencapai tujuan merupakan salah satu perkembangan pola pikir manusia.

Jalur menuju rumah sakit dengan jalan tercepat merupakan suatu bentuk kebutuhan manusia. Ini dikarenakan kondisi pasien untuk mendapat pertolongan juga harus cepat dan tepat, sehingga tidak menjadikan hal-hal yang tidak diinginkan terjadi. Letak dari rumah sakit ditiap kota berbeda-beda sehingga jika ingin menuju ke rumah sakit ada banyak pilihan yang harus dipilih, sesuai dengan kebutuhan dari pasien. Jarak dari pusat kota menuju rumah sakit memiliki perbedaan waktu tempuh. Terkadang ada orang yang mengabaikan waktu tempuh untuk menuju rumah sakit dengan jarak terdekat. Oleh karena itu penerapan algoritma Greedy ini sangat diperlukan dalam menyelesaikan kasus ini.

Badrudin Hadibrata, dan Saeful Maudin (2020) dalam penelitiannya yang berjudul **“Pencarian Rute Terpendek Menuju Tempat Wisata Menggunakan Metode Algoritma Greedy Pada Dinas Pemuda Olahraga Kebudayaan Dan Pariwisata Kota Cirebon”**

Terjebak dalam kemacetan pastinya membuat seseorang menjadi jenuh dan bosan, sehingga mereka akan berusaha mencari cara agar perjalanan mereka menjadi tidak terhambat dan sampai ke tempat tujuan dengan waktu yang singkat dan cepat. Salah satu solusinya yaitu dengan mengetahui rute alternatif tercepat agar terhindar dari kemacetan. Rute alternatif tercepat ini berguna untuk mendapatkan jarak dan waktu tempuh yang optimal dari tempat asal menuju tempat tujuan tanpa terhambat oleh kemacetan yang disebabkan melonjaknya wisatawan.

Dengan mempermudah wisatawan untuk mengakses tempat wisata dengan jalur yang cepat maka penerapan Sistem Informasi Geografis (SIG) berbasis web merupakan salah satu langkah atau cara untuk membantu para wisatawan dalam mengakses tempat-tempat wisata yang ada di Kota Cirebon. Sistem ini nantinya akan dirancang agar pengguna dapat mengakses dan memperoleh navigasi rute tempat-tempat wisata di kota Cirebon, melalui perangkat Smartphone serta bantuan web dengan mengimplementasi Algoritma Greedy, sehingga mempercepat pengaksesan tempat wisata.

Algoritma Greedy merupakan algoritma yang membentuk solusi langkah per langkah. Pada setiap langkah tersebut akan dipilih keputusan yang paling optimal. Sebuah metode yang dapat memecahkan persoalan optimasi dengan cara bertahap melalui alur yang selalu berkembang hingga pemecahan permasalahan

dapat teratasi. Algoritma Greedy dapat memilih sebuah rute yang diambil dahulu untuk memberikan suatu alternatif lokal agar menghasilkan alternatif optimal dengan menyeluruh sehingga diperoleh rute tercepat dari lokasi awal hingga lokasi yang dituju.

Indra Riksa Herlambang, Mohamad Nurkamal Fauzan, dan Rd. Nuraini Siti Fathonah³ (2021) dalam penelitiannya yang berjudul **“Penentuan Rute Terpendek Pendistribusian Barang Menggunakan Algoritma Floyd-Warshall”**

Ketepatan waktu merupakan hal yang penting dalam pendistribusian kantong pos atau barang di PT Pos Logistik Indonesia (Jakarta Pusat). Berdasarkan survei yang telah dilakukan, tingkat keterlambatan dalam pendistribusian kantong pos dari lokasi awal ke lokasi tujuan masih cukup besar dan masalah tersebut menjadi hal yang harus diperbaiki. Untuk mengoptimalkan jarak tempuh perusahaan dalam melakukan proses pendistribusian barang menuju lokasi yang menjadi tujuan dan juga dapat mengurangi tingkat keterlambatan dalam proses pendistribusian barang pendistribusian yang ada di PT Pos Logistik Indonesia (Jakarta Pusat) maka perlu diterapkan pencarian rute terpendek menggunakan algoritma Floyd-Warshall agar dapat mengefisiensi waktu, jarak, dan biaya. Permasalahan pada jalur terpendek adalah untuk menemukan jalur antara dua simpul sehingga jumlah bobot dari busur penyusunnya dapat seminimal mungkin.

Permasalahan jalur transportasi dan penentuan rute terpendek bisa diatasi dengan menggunakan teori Graf. Graf yaitu kumpulan-kumpulan titik yang dihubungkan dengan garis yang memiliki bobot, dapat dikatakan graf merupakan

kumpulan dari beberapa simpul yang dihubungkan oleh sisi-sisi. Berdasarkan pada teori Graf, permasalahan pada rute terpendek dapat diartikan sebagai suatu permasalahan untuk mendapatkan lintasan antara dua buah simpul pada graf berbobot yang mempunyai gabungan nilai dari total bobot pada sisi graf yang dilalui dengan jumlah yang paling minimum. Persoalan mencari jalur terpendek pada graf ialah salah satu persoalan optimasi. Graf yang digunakan dalam pencarian jalur terpendek adalah graf berbobot (weighted graph).

Aprilia Divi Yustita, Siska Aprilia Hardiyanti, dan Ika Yuniwati (2018) dalam penelitiannya yang berjudul **“Algoritma Floyd-Warshall Untuk Penentuan Rute Terpendek Model Jaringan Pariwisata Kabupaten Banyuwangi”**.

Perkembangan pariwisata yang dilakukan di Banyuwangi mengakibatkan terus bertambahnya jumlah objek wisata yang ada. Objek wisata di Banyuwangi saat ini memiliki jumlah yang cukup banyak, baik wisata alam maupun buatan. Terdapat sebanyak 128 objek wisata tersebar di Banyuwangi yang tercatat di Dinas Kebudayaan dan Pariwisata Kabupaten Banyuwangi pada tahun 2018, dimana 19 diantaranya merupakan objek wisata populer saat ini. Sementara itu, penambahan jumlah objek wisata yang terjadi mengakibatkan semakin banyaknya jumlah wisatawan yang tertarik untuk mengunjungi Banyuwangi. Hal tersebut dibuktikan dengan adanya kenaikan jumlah wisatawan yang terjadi pada setiap tahunnya, baik wisatawan domestik maupun mancanegara. Pada tabel berikut ditunjukkan kenaikan jumlah kunjungan wisatawan ke Banyuwangi mulai tahun 2013 sampai dengan tahun 2017 menurut catatan Dinas Kebudayaan dan Pariwisata Kabupaten Banyuwangi tahun 2018.

Semakin banyaknya jumlah objek wisata yang ada di Banyuwangi mengakibatkan semakin besar pula kebutuhan wisatawan akan informasi yang ingin diketahui tentang objek wisata yang akan dikunjungi. Salah satu jenis informasi yang dibutuhkan wisatawan adalah mengenai rute menuju beberapa objek wisata yang populer, serta rute terpendek untuk menuju objek wisata tersebut. Informasi tersebut diperlukan untuk membantu wisatawan merencanakan perjalanan yang akan dilakukan hingga kembali ke tempat asal maupun ke tempat tinggal sementara sehingga mengefisienkan waktu, jarak, tenaga, dan biaya yang dikeluarkan (Widya & Andrasto, 2016).

Pada (Widya & Andrasto, 2016) telah dibahas mengenai penerapan algoritma Floyd Warshall dalam menentukan rute terpendek pemodelan jaringan pariwisata di Kota Semarang. Hasil dari penelitian tersebut adalah diperolehnya aplikasi untuk pencarian jalur terpendek pada jaringan pariwisata Kota Semarang. Berdasarkan uraian yang telah dijelaskan, maka pada penelitian ini dibahas mengenai analisis penentuan rute terpendek jaringan pariwisata di Kabupaten Banyuwangi dengan menerapkan algoritma Floyd Warshall. Input dari algoritma ini adalah matriks persegi yang merupakan hasil representasi dari graf berarah berbobot dari peta objek wisata di Banyuwangi. Algoritma Floyd-Warshall dimungkinkan memiliki waktu proses yang lama dikarenakan pembentukan matriks sesuai dengan banyaknya iterasi yang dilakukan, akan tetapi masih sering digunakan dalam pencarian lintasan terpendek karena memiliki algoritma yang sederhana dan mudah diimplementasikan (Widya & Andrasto, 2016). Selain itu, sebagai salah satu algoritma untuk penentuan shortest path, algoritma ini mampu membandingkan semua kemungkinan lintasan pada graph untuk semua sisi dan

semua simpul yang ada (Jayanti, 2014). Dengan demikian, algoritma ini baik untuk diterapkan dalam penghitungan jalur terpendek khususnya pada jaringan pariwisata Kabupaten Banyuwangi

2.2 Definisi Jarak

Jarak terpendek merupakan teori graf salah satu persoalan optimasi. graf yang digunakan dalam pencarian lintasan terpendek adalah graf berbobot (weighted graph), yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot. Bobot pada sisi graf dapat menyatakan jarak antar kota, waktu pengiriman pesan, ongkos pembangunan, dan sebagainya. Asumsi yang digunakan di sini adalah bahwa semua bobot bernilai positif. Lintasan terpendek adalah jalur yang dilalui dari suatu node ke node lain dengan besar atau nilai pada sisi yang jumlah akhirnya dari node awal ke node akhir paling kecil. Lintasan terpendek adalah lintasan minimum yang diperlukan untuk mencapai suatu tempat dari tempat lain. Lintasan minimum yang dimaksud dapat dicari dengan menggunakan graf. Graf yang digunakan adalah graf yang berbobot yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot (Yonny Danies Mahendra, Nuryanto, dan Auliya Burhanuddin, 2019).

2.3 Algoritma Sistem

2.3.1 Travelling Salesman Problem (TSP) & Greedy

Travelling Salesman Problem yaitu permasalahan optimasi yang sering terjadi sekitar kita. Permasalahan ini menarik perhatian para peneliti. Para ilmuwan komputer sering terlibat riset permasalahan TSP, karena kasus TSP mudah didefinisikan namun sulit diselesaikan. Permasalahan TSP mengambil contoh kasus permasalahan dari seorang sales yang harus melewati beberapa kota tujuan. Dimana setiap kota tujuan hanyalah dikunjungi satu kali dan selanjutnya

kembali ke kota asal dan total jarak yang ditempuh haruslah seminimum mungkin. Permasalahan TSP ini akan dicoba dianalisa dan dieksplorasi menggunakan algoritma Particle Swarm Optimization (PSO) dan juga pengembangan algoritma PSO, yaitu algoritma Quantum Particle Swarm Optimization (QPSO). Optimasi yang dilakukan untuk menentukan urutan paling optimal alamat-alamat pengiriman kurir. Solusi yang dihasilkan berupa probabilitas yang dapat dijadikan acuan kurir (Khadafi, 2016).

William Rowan Hamilton dan Thomas Penyngton merupakan orang yang pertama kali menemukan travelling salesman. Permainan Icosian Hamilton merupakan asal mula ditemukannya TSP. Dalam permainan tersebut diminta untuk menyelesaikan perjalanan dari 20 titik berbeda yang melewati jalur-jalur tertentu. TSP merupakan permasalahan yang dialami salesman dalam mencari rute terpendek yang dapat dilalui dengan mengunjungi beberapa tempat tanpa harus mengulang untuk melewati tempat yang telah dilewati artinya salesman tidak melewati tempat atau titik yang sama lebih dari satu kali tetapi salesman dapat menyelesaikannya sampai kembali ke tempat atau titik awal (Applegate dkk., 2016).

Dalam menyelesaikan permasalahan TSP terdapat beberapa algoritma yang digunakan, salah satunya adalah algoritma *Greedy*. Algoritma *Greedy* merupakan suatu metode yang digunakan untuk memecahkan permasalahan optimasi dimana setiap langkah akan memilih kota atau tempat yang belum dikunjungi, kota atau tempat yang belum dikunjungi dipilih berdasarkan jarak kota atau tempat mana yang terdekat dari kota atau tempat sebelumnya (Muhammad Z. U., dkk, 2018).

Berikut adalah cara perhitungan dengan menggunakan algoritma Greedy:

1. Tentukan kota atau tempat awal dimulainya perjalanan.
2. Kemudian kota awal menuju kota selanjutnya dimana kota selanjutnya tersebut berada paling dekat dari kota awal.
3. Kemudian lanjut ke kota berikutnya yang terdekat hingga kota tujuan selesai.
4. Selanjutnya balik ke kota awal dan jumlahkan seluruh jarak tersebut.

Berdasarkan definisi diatas, maka dapat disimpulkan bahwa persyaratan dalam menyelesaikan TSP yaitu (Muhammad Z. U., dkk, 2018) :

1. Tidak dapat melewati setiap kota lebih ataupun kurang dari satu kali
2. Setiap kota harus dapat dikunjungi dan diselesaikan dalam satu kali perjalanan atau rute
3. Salesman harus kembali ke titik atau kota awal untuk mengakhiri perjalanan

Parameter yang digunakan dalam TSP adalah d sebagai jarak dari kota i ke kota j . Berikut merupakan model persamaan TSP : (Muhammad Z. U., dkk, 2018)

$$x_{ij} = \begin{cases} 1, & \text{jika kota } j \text{ dicapai dari kota } i \\ 0, & \text{jika sebaliknya} \end{cases}$$

Fungsi tujuan : $Min z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}, d_{ij} = \infty$ untuk semua $i = j$

$$\sum_{n=1}^n x_{ij} = 1 \quad i = 1, \dots, n \quad (1)$$

$$\sum_{n=1}^n x_{ij} = 1 \quad j = 1, \dots, n \quad (2)$$

$$x_{ij} = (0,1) \quad (3)$$

2.3.2 Floyd-Warshall

Algoritma Floyd-Warshall digunakan untuk pencarian jalur terpendek menuju suatu objek wisata tertentu dari posisi awal wisatawan berada sehingga dapat diperoleh suatu jalur alternatif yang dapat ditempuh. Algoritma Floyd-Warshall adalah algoritma sederhana yang dapat digunakan untuk menghitung jarak terpendek antara seluruh pasangan titik yang dihubungkan oleh garis dalam suatu graf berarah yang memiliki bobot (weighted directed graph). Algoritma FloydWarshall bekerja dengan membandingkan semua kemungkinan lintasan pada graf untuk setiap sisi dari semua titik dan memberikan perkiraan nilai optimal (pemilihan jalur terpendek) untuk setiap dua titik (Aprilia Divi Yustita, Siska Aprilia Hardiyanti, dan Ika Yuniwati, 2018).

Algoritma Floyd-Warshall memiliki input graf berarah dan berbobot (V,E) , dimana V adalah himpunan semua titik dan E adalah himpunan semua sisi. Bobot garis e dapat diberi simbol $w(e)$. Jumlah bobot sisi-sisi pada sebuah jalur adalah total bobot jalur tersebut. Matriks keterhubungan W yang digunakan untuk menyatakan graf berarah berbobot sama dengan matriks yang digunakan untuk menyatakan graf berbobot, yaitu elemen-elemen dalam matriks menyatakan besar bobot garis atau jarak setiap dua titik.

Pada prinsipnya, algoritma Floyd-Warshall yaitu jika penjumlahan dari nilai titik awal dengan nilai titik akhir lebih kecil dibandingkan dengan nilai jarak sebenarnya maka ganti nilai jarak yang sebenarnya dengan jumlah dari nilai titik awal dengan titik akhir. Algoritma Floyd-Warshall untuk mencari lintasan terpendek adalah sebagai berikut : (Sudaria, Arman Syah Putra, dan Yosua

Novembrianto. 2021).

1) $W = W_0; Z = Z_0$

2) Untuk $k = 1$ hingga n , lakukan:

 Untuk $i = 1$ hingga n , lakukan:

 Untuk $j = 1$ hingga n , lakukan:

 Jika $W[i,j] > W[i,k] + W[k,j]$ maka

 a. Tukar $W[i,j]$ dengan $W[i,k] + W[k,j]$.

 b. Ganti $Z_{i,j}$ dengan $Z_{i,k}$

3) $W^* = W$.

Keterangan:

W_0 = matriks keterhubungan graf berarah berbobot awal

W^* = matriks keterhubungan minimal

$W[i,j]$ = lintasan terpendek dari titik v_i ke v_j

2.1.1 Haversine

Metode Haversine digunakan untuk menghitung jarak antara titik di permukaan bumi menggunakan garis lintang (longitude) dan garis bujur (latitude) sebagai variabel inputan. Haversine formula adalah persamaan penting pada navigasi, memberikan jarak lingkaran besar antara dua titik pada permukaan bola (bumi) berdasarkan bujur dan lintang. Dengan mengasumsikan bahwa bumi berbentuk bulat sempurna dengan jari-jari = $R = 6.371$ km, dan lokasi dari 2 titik di koordinat bola (lintang dan bujur) masing-masing adalah lon_1 , lat_1 , dan lon_2 , lat_2 , maka rumus Haversine dapat ditulis dengan persamaan berikut ini.

Rumus Haversine :

$$x = (\text{lon2} - \text{lon1}) * \cos ((\text{lat1} + \text{lat2})/2)$$

$$y = (\text{lat2} - \text{lat1})$$

$$d = \text{sqrt}(x*x + y*y) * R$$

Keterangan:

x = Longitude (Lintang)

y = Latitude (Bujur)

d = Jarak

R= Radius Bumi =6371 km

1 derajat = 0.0174532925 radian

Haversine formula adalah metode perhitungan jarak antara dua titik di bumi berdasarkan panjang garis lurus antara dua titik tanpa mengabaikan kelengkungan bumi .

Berikut adalah persamaan Haversine :

$$a = \sin^2 (\Delta \text{lat} / 2) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2 (\Delta \text{long} / 2)$$

$$d = 2r \cdot \text{arc sin} (\sqrt{a})$$

Keterangan :

d = Jarak

r = Jari-jari bumi

Δlat = besaran perubahan latitude = lat2 – lat1

Δlong = besaran perubahan longitude = long2 – long1

2.2 Pemograman Aplikasi

2.2.1 Internet

Internet merupakan sistem global dari seluruh jaringan komputer yang saling terhubung. Internet adalah jaringan global yang menghubungkan komputer-komputer seluruh dunia, dengan internet sebuah komputer bisa mengakses data yang terdapat pada komputer lain di benua yang berbeda (Sudaria , Arman Syah Putra , dan Yosua Novembrianto, 2021).

2.2.2 Definisi Sistem

Sistem adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasanya berbagi dalam sub sistem yang lebih kecil yang mendukung sistem yang lebih besar. Sistem adalah kumpulan atau rangkaian komponen-komponen yang berhubungan, bekerja sama dan saling berinteraksi untuk mencapai suatu tujuan dengan melalui tiga tahapan input (masuk, proses dan output (keluar). Menyatakan bahwa sistem bisa diartikan sebagai kumpulan sub sistem, komponen yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan output yang sudah ditentukan sebelumnya (Sudaria, Arman Syah Putra, dan Yosua Novembrianto. 2021).

2.2.3 Definisi Informasi

Informasi (information) adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan. Sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Informasi adalah data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat. Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa informasi adalah sekumpulan fakta-fakta yang telah diolah yang telah diolah menjadi bentuk data, sehingga dapat menjadi lebih berguna dan dapat digunakan oleh siapa saja yang membutuhkan data-data tersebut sebagai pengetahuan ataupun dapat digunakan dalam pengambilan keputusan (Sudaria , Arman Syah Putra , dan Yosua Novembrianto, 2021).

2.2.4 Web

World Wide Web (WWW atau web) merupakan sistem informasi terdistribusi yang berbasis hypertext. World Wide Web (WWW). Informasi WWW ini disimpan pada web server untuk dapat diakses dari jaringan browser terlebih dahulu, seperti Internet Explorer atau Mozilla Firefox (Sudaria , Arman Syah Putra , dan Yosua Novembrianto, 2021).

2.2.5 HTML

HTML (Hyper Text Markup Language) adalah di kembangkan pertama kali oleh Tim Berners-Lee bersamaan dengan protokol HTTP (Hypertext Transfer Protocol) Tujuan utama pengembangan HTML adalah untuk menghubungkan

setiap laman web dengan laman web lainnya (Rani Hormati, Sartina Yusuf , dan Muhdar Abdurahman, 2021).

2.2.6 CSS (Cascading Style Sheet)

CSS atau Cascading Style Sheets adalah memiliki arti gaya menata halaman bertingkat ,yang artinya setiap satu elemen yang telah di format dan di memiliki anak dan telah di format ,maka dari elemen tersebut secara otomatis mengikuti format elemen induknya (Rani Hormati, Sartina Yusuf , dan Muhdar Abdurahman, 2021).

2.2.7 PHP (Hypertext Processor)

PHP merupakan salah satu bahasa pemograman berbasis web yang di tulis untuk mengembang web dan PHP sering digunakan pada sisi server sebuah web (Rani Hormati, Sartina Yusuf , dan Muhdar Abdurahman, 2021).

2.2.8 JavaScript

Javascript adalah bahasa yang berbentuk kumpulan skrip yang fungsinya digunakan untuk menambahkan interaksi antara halaman web dengan pengguna halaman web. Javascript dijalankan pada sisi klien yang akan memberikan kemampuan fiturfitur tambahan halaman web yang lebih baik dibandingkan fitur – fitur yang terdapat pada HTML (Andy Febrianto, Sentot Achmadi, dan Agung Panji Sasmito, 2021).

2.2.9 MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal. MySQL menggunakan bahasa SQL untuk mengakses databasenya. Lisensi Mysql adalah FOSS License Exception dan ada juga yang versi komersial nya. MySQL tersedia untuk beberapa platform, di antara nya adalah untuk versi windows dan

versi linux. Untuk melakukan administrasi secara lebih mudah terhadap Mysql, anda dapat menggunakan software tertentu, diantaranya adalah phpmyadmin dan mysql yog (Andy Febrianto, Sentot Achmadi, dan Agung Panji Sasmito, 2021).

2.2.10 Database

Basis data merupakan komponen terpenting dalam pembangunan SI, karena menjadi tempat untuk menampung dan mengorganisasikan seluruh data yang ada dalam sistem, sehingga dapat dieksplorasi untuk menyusun informasi-informasi dalam berbagai bentuk. Basis data merupakan himpunan kelompok data yang saling berkaitan. Basis data (database) adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi atau perusahaan yang diorganisir atau dikelola dan disimpan secara terintegritasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakainya (Sudaria , Arman Syah Putra , dan Yosua Novembrianto, 2021).

2.3 Pemodelan Unifed Modeling Language (UML)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada

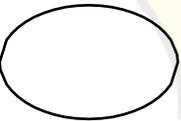
kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Rani Hormati, Sartina Yusuf, dan Muhdar Abdurahman, 2021).

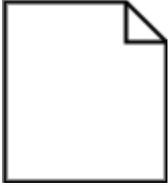
2.3.1 Use Case Diagram

Use case atau diagram use case 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. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rani Hormati, Sartina Yusuf, dan Muhdar Abdurahman, 2021).

Berikut adalah symbol-simbol yang ada pada Use case Diagram :

GAMBAR	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>usecase</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri(<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).

	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>usecase</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i>	Menspesifikasikan bahwa <i>usecase</i> target memperluas perilaku dari <i>usecase</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Usecase</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).

	<p><i>Note</i></p>	<p>Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi</p>
---	--------------------	--

Gambarl 2.1 Use Case Diagram (Nugroho, 2015).

2.3.2 Activity Diagram

Diagram aktivitas menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Irianto , Sudarmin , dan Afrisawati, 2021).

Berikut adalah symbol-simbol yang ada pasa Activity Diagram :

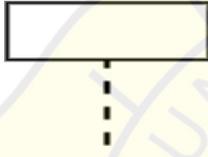
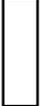
Simbol	Penjelasan
	<p><i>Initial State</i></p> <p>Mempresentasikan dimulainya alur kerja suatu sistem dalam <i>activity diagram</i>.</p>
	<p><i>Fork</i></p> <p>Adanya percabangan paralel dari aktivitas</p>
	<p><i>Final State</i></p> <p>Mempresentasikan bahwa telah diakhirinya alur suatu sistem dalam <i>activity diagram</i>.</p>

Gambar 2.2 Activity Diagram (Nugroho, 2015)

2.3.3 Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dengan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek (Rani Hormati, Sartina Yusuf, dan Muhdar Abdurahman, 2021).

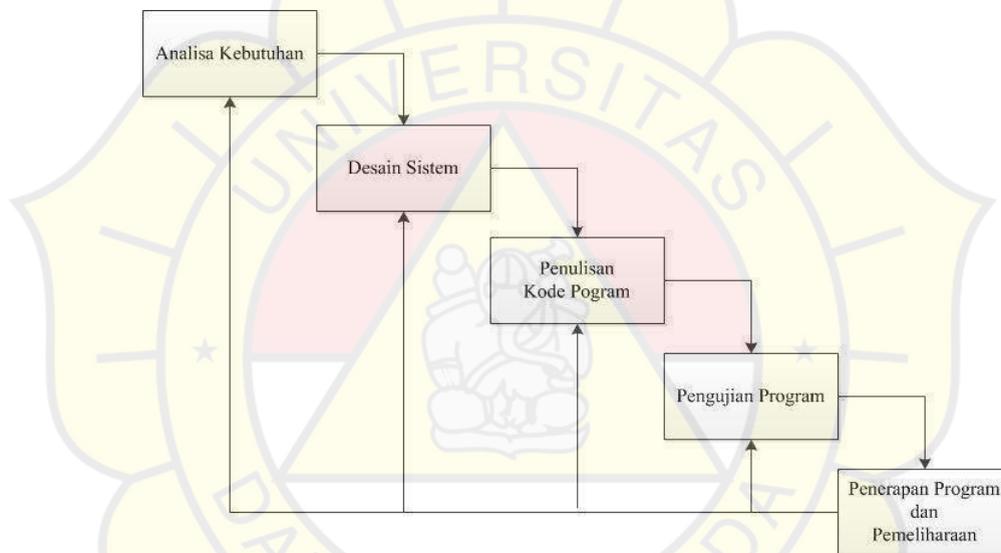
Berikut symbol – symbol yang ada pada Sequence Diagram :

Simbol	Nama Elemen	Keterangan
	Objek	Komponen yang menjadikan sebuah objek dalam membuat diagram
	<i>Stimulus</i>	Untuk menandakan hubungan komunikasi antarobjek
	<i>Self Stimulus</i>	Fungsi sama dengan <i>Stimulus</i> , tetapi pesan yang disampaikan dikirimkan untuk objek itu sendiri
	<i>Focus Control</i>	Sebagai tempat untuk hasil input atau output dari sebuah proses yang dilakukan oleh objek ataupun aktor yang ada dalam sistem

Gambar 2.3 Sequence Diagram (Nugroho, 2015)

2.4 Metodologi Pengembangan Sistem

Metode penelitian yang diterapkan pada penelitian ini adalah dengan pengembangan metode waterfall. Metode waterfall merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode Waterfall adalah suatu proses perangkat lunak yang berurutan, dipandang sebagai terus mengalir kebawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi dan pengujian.



Tahapan -Tahapan dalam Metode Waterfall :

1. Analisa Kebutuhan

Sebelum melakukan pengembangan perangkat lunak, seorang pengembang harus mengetahui dan memahami bagaimana informasi kebutuhan pengguna terhadap sebuah perangkat lunak. Metode pengumpulan informasi ini dapat diperoleh dengan berbagai macam cara diantaranya, diskusi, observasi, survei, wawancara, dan sebagainya. Informasi yang diperoleh kemudian diolah dan dianalisa sehingga didapatkan data atau

informasi yang lengkap mengenai spesifikasi kebutuhan pengguna akan perangkat lunak yang akan dikembangkan.

2. Desain Sistem

Informasi mengenai spesifikasi kebutuhan dari tahap Analisa ebutuhan selanjutnya di analisa pada tahap ini untuk kemudian diimplementasikan pada desain pengembangan. Perancangan desain dilakukan dengan tujuan membantu memberikan gambaran lengkap mengenai apa yang harus dikerjakan. Tahap ini juga akan membantu pengembang untuk menyiapkan kebutuhan hardware dalam pembuatan arsitektur sistem perangkat lunak yang akan dibuat secara keseluruhan.

3. Penulisan Kode Program / Implementasi

Tahap Penulisan Kode Program ini merupakan tahap pemrograman. pembuatan perangkat lunak dibagi menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Disamping itu, pada fase ini juga dilakukan pengujian dan pemeriksaan terhadap fungsionalitas modul yang sudah dibuat, apakah sudah memenuhi kriteria yang diinginkan atau belum.

4. Pengujian Program

Setelah seluruh unit atau modul yang dikembangkan dan diuji di tahap implementasi selanjutnya diintegrasikan dalam sistem secara keseluruhan. Setelah proses integrasi selesai, selanjutnya dilakukan pemeriksaan dan pengujian sistem secara keseluruhan untuk mengidentifikasi kemungkinan adanya kegagalan dan kesalahan sistem.

5. Penerapan dan Pemeliharaan Program

Pada tahap terakhir dalam Metode Waterfall, perangkat lunak yang sudah jadi dioperasikan pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang untuk melakukan perbaikan atas kesalahan yang tidak terdeteksi pada tahap-tahap sebelumnya. Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit sistem, dan peningkatan dan penyesuaian sistem sesuai

