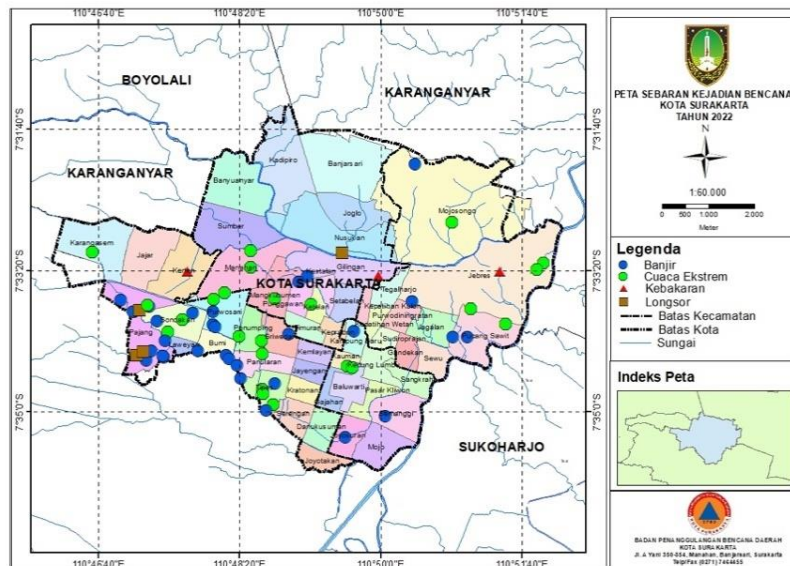


BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Cuaca ekstrem, hujan yang berkepanjangan serta meluapnya sungai dan bendungan menyebabkan sering terjadi banjir di kota-kota besar yang ada di Indonesia, salah satunya di Kota Surakarta. Banjir merupakan masalah global yang menyebabkan banyak kerugian baik dalam bentuk perekonomian, kesehatan, harta benda serta memakan banyak korban jiwa (Ramadhan, 2024).



Gambar 1.1 Peta Kejadian Bencana

Dilansir dari web resmi bpbd.surakarta.go.id Penyebab banjir di Kota Surakarta dipengaruhi oleh dua faktor yaitu faktor yang disebabkan oleh alam dan faktor yang disebabkan oleh manusia, curah hujan yang tinggi dan rusaknya *drainase*, terjadinya penumpukan sampah di saluran pembuangan air turut memperburuk situasi. Data Badan Penanggulangan Bencana Daerah (BPBD) tahun

2022 untuk Kota Surakarta, tercatat diantaranya ada lima lokasi titik masuk zona rawan banjir seperti tersebar di wilayah Kecamatan Banjarsari, Jebres, Laweyan, Serengan dan Pasar Kliwon.

Koordinasi dan penyebaran informasi tentang daerah rawan banjir masih kurang efektif. Banyak warga yang tidak memiliki akses terhadap informasi terbaru mengenai daerah rawan banjir, sehingga mereka tidak dapat mengambil langkah-langkah pencegahan yang tepat. Selain itu, pemerintah sering kesulitan dalam mengumpulkan dan mengelola data terkait banjir, yang menyebabkan respon terhadap bencana menjadi kurang optimal. Oleh karena itu, dibutuhkan sebuah sistem informasi pemetaan yang dapat menyediakan informasi spasial secara real-time dan akurat mengenai daerah rawan banjir. Sistem ini akan membantu pemerintah dan masyarakat dalam mengidentifikasi dan mengelola risiko banjir, serta meningkatkan kesiapsiagaan dan mitigasi bencana. Dengan adanya sistem informasi ini, diharapkan kerugian akibat banjir dapat diminimalisir dan keselamatan warga dapat lebih terjamin.

Melalui penerapan pemetaan sistem informasi geografis banjir dapat dikelola secara efektif dengan memuat informasi daerah rawan banjir di kota Surakarta. Penerapan sistem informasi pemetaan daerah rawan banjir mendukung pengelolaan informasi menjadi sangat penting untuk mengumpulkan, mengelola, menyediakan dan menyebarkan informasi daerah rawan banjir oleh pemerintah kepada masyarakat Kota Surakarta. Berdasarkan kondisi yang terjadi pada saat itu juga, dapat mempengaruhi keakuratan informasi data terkini.

Pemetaan daerah rawan banjir sangat penting untuk membantu meningkatkan kesadaran masyarakat terhadap daerah yang berpotensi banjir di Kota Surakarta. Dalam proses perancangan pemetaan dilakukan dengan memanfaatkan *Geographic Information System (GIS)* menggunakan *Google Maps Api*. *Google Maps API* adalah serangkaian antarmuka pemrograman aplikasi yang disediakan oleh *Google* untuk mengintegrasikan peta, data geospasial, dan berbagai layanan *Google Maps* ke dalam aplikasi atau situs web (Anam, 2024).

Melalui Direktorat Jendral Sumber Daya Air, Kementerian Pekerjaan Umum dan Perumahan Rakyat terus berupaya melakukan penanganan banjir yang terjadi di seluruh wilayah sungai Indonesia. Di Kota Surakarta, Kementerian PUPR melalui Balai Besar Wilayah Sungai (BBWS) Pompengan Jeneberang melakukan monitoring dan peninjauan lapangan terkait penanganan banjir di Kota Surakarta.

Pada tahun 2021, Kementerian Pekerjaan Umum dan Perumahan Rakyat (PUPR) melalui BBWS Pompengan Jeneberang mengalokasikan anggaran sebesar Rp. 44 Miliar untuk operasi dan pemeliharaan seluruh sarana pengelolaan Sumber Daya Air termasuk revitalisasi pengendalian banjir di Kota Surakarta.

Peninjauan lapangan dilakukan untuk memastikan kondisi aliran sungai di kota Surakarta jelang musim hujan sebagai bentuk upaya mengurangi risik bencana banjir, dimana titik pertama yang ditinjau adalah aliran sungai yang dapat terjadi penyempitan sungai akibat penumpukan limbah sampah baru di area sungai. Adanya normalisasi masih terhambat dikarenakan penyempitan jembatan di daerah perumahan.

Berdasarkan latar belakang yang dijelaskan, penulis bertujuan untuk membuat suatu sistem informasi pemetaan untuk daerah rawan banjir di Kota Surakarta. Hasil yang diharapkan dari penelitian ini adalah akan mempermudah penyajian informasi spasial khususnya daerah rawan banjir dalam mengidentifikasi daerah mana saja yang sering terkena banjir sehingga dapat mengurangi jumlah kerugian dan korban jiwa akibat banjir.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka permasalahan yang dirumuskan adalah, sebagai berikut:

Bagaimana merancang dan membuat sistem pengaduan daerah rawan banjir berbasis web di Kota Surakarta untuk warga yang masih belum memiliki akses terhadap informasi mengenai daerah rawan banjir?

1.3 Batasan Masalah

Berdasarkan rumusan masalah yang telah diuraikan, maka batasan yang dipakai pada penulisan ini, sebagai berikut:

1. Data hanya memuat data untuk cakupan wilayah Kota Surakarta.
2. Sistem dibuat hanya mencakup rekap pengaduan titik banjir.
3. Menggunakan bahasa pemrograman *PHP* dan *database MySQL*.
4. Menggunakan *Google Maps API* sebagai layanan penyedia digital peta.
5. Level *users* yaitu untuk melaporkan titik banjir sedangkan level *admin* yaitu mengelola titik banjir, mengelola laporan masuk laporan banjir dan mengelola verifikasi *users*.

1.4 Tujuan Penulisan

Berdasarkan rumusan masalah yang telah diuraikan, maka tujuan pada penulisan ini yaitu sebagai sarana dalam merancang sistem informasi pemetaan daerah rawan banjir di Kota Surakarta berbasis *Geographic Information System*.

1.5 MANFAAT PENULISAN

Manfaat yang diharapkan dalam penulisan ini adalah sebagai berikut:

1.5.1 Manfaat Bagi Badan Penanggulangan Bencana Daerah

Untuk mengetahui penyebaran dan pertukaran informasi serta meningkatkan kesadaran masyarakat lebih siap dalam menghadapi bencana banjir dan dapat meminimalisir kerugian akibat banjir

1.5.2 Manfaat Bagi Masyarakat

Penulisan ini dapat memberikan kontribusi yang signifikan dalam penanggulangan bencana banjir. Dengan memahami bagaimana aplikasi pengaduan banjir dapat membantu masyarakat dan pemerintah dalam mengatasi masalah ini, penelitian ini dapat menjadi langkah awal dalam upaya mitigasi bencana.

1.5.3 Manfaat Bagi Penulis

Menambah pengetahuan dalam pembuatan sistem informasi pemetaan berbasis *Geographic System Information*.

1.5.4 Manfaat Bagi Kampus STMIK AMIKOM Surakarta

Hasil yang didapatkan dari penulisan ini diharapkan menjadi bahan referensi selanjutnya bagi yang berminat dalam bidang pemetaan.

1.6 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penyusunan tugas akhir ini mengandung uraian tentang metode yang digunakan untuk mengumpulkan data. Contoh di bawah silakan disesuaikan dengan kebutuhan penulis.

1.6.1 Metode Observasi

Melakukan observasi langsung ke lokasi-lokasi yang rawan banjir di Kota Surakarta pada Kecamatan Pasar Kliwon, Jebres, Laweyan, Banjarsari dan Serengan. Mengamati kondisi topografi, sistem drainase, kerentanan bangunan dan infrastruktur, serta pola aliran air yang berpotensi menyebabkan banjir.

1.6.2 Metode Wawancara

Merangkum hasil wawancara warga yang telah diterbitkan oleh berbagai media berita terkait, seperti yang bersumber dari perwakilan masyarakat setempat yaitu Camat Serengan Bapak Agung Wijayanto, Kepala BPBD Kota Surakarta Bapak Nico Agus Putranto, dan pihak media Kompas.id yaitu Bapak Nino Citra Anugrahanto.

1.6.3 Studi Pustaka

Pengumpulan data rawan banjir melalui literatur yaitu diantaranya “Pemetaan Kerawanan Banjir Di Kota Surakarta, 2024” yang bersifat ilmiah yang memiliki keterkaitan dengan materi pembahasan pemetaan daerah rawan banjir di Kota Surakarta dari Jurnal terkait dan media berita.

1.7 Teori Yang Digunakan

Berikut adalah teori-teori yang digunakan dalam penulisan ini untuk mendukung pemahaman tentang berbagai informasi mengenai Sistem Pengaduan Titik Rawan Banjir di Kota Surakarta Berbasis Web.

1.7.1 Bencana Banjir

Banjir merupakan bencana alam yang sering terjadi di Indonesia. Definisi banjir adalah keadaan dimana suatu daerah tergenang oleh air dalam jumlah yang besar. Kedatangan banjir dapat diprediksi dengan memperhatikan curah hujan dan aliran air. Namun kadang kala banjir dapat datang tiba-tiba akibat dari angin badai atau kebocoran tanggul yang biasa juga disebut banjir bandang.

Kerentanan banjir merupakan penilaian daerah yang mungkin menjadi sasaran banjir. Dataran banjir biasanya ditemukan didaerah datar, dekat sungai, cekungan dan dataran banjir. Bentuk lahan dataran banjir biasanya ditemukan di daerah yang telah mengalami banjir berulang, dan daerah ini biasanya memiliki kelembaban tanah yang lebih tinggi daripada daerah lain yang kurang banjir. Kelembaban tanah yang tinggi ini disebabkan oleh bentuk lahan, yang terdiri dari materi halus yang diendapkan oleh banjir dan kondisi drainase yang buruk, yang membuat daerah tersebut rentan terhadap genangan air (Anggraini et al., 2020)

Kata banjir terkadang identik dengan genangan bagi sebagian orang, sehingga dalam penyampaian informasi banjir di suatu daerah tidak cukup akurat. Genangan air adalah luapan air yang hanya terjadi dalam beberapa jam setelah hujan mulai turun. Genangan yang disebabkan oleh luapan air hujan di saluran pembuangan, menyebabkan air terkumpul dan tertahan di suatu area dengan ketinggian air 5 sampai lebih dari 20 cm, sedangkan banjir adalah luapan air hujan deras dengan ketinggian air 30 sampai lebih dari 200 cm (Ramadhan, 2024)

Penyebab banjir mencakup curah hujan tinggi, permukaan tanah lebih rendah dibanding muka air laut, wilayah terletak pada suatu cekungan yang dikelilingi perbukitan dengan sedikit resapan air, pendirian bangunan di sepanjang bantaran sungai, aliran sungai tidak lancar akibat terhambat oleh sampah, serta kurangnya tutupan lahan di daerah hulu sungai.

1.7.1.1 Klasifikasi Banjir

Pada umumnya, banjir yang sering terjadi di Indonesia dapat diklasifikasikan menjadi tiga, yaitu:

1. Banjir akibat meluapnya sungai

Terjadinya banjir ini dikarenakan kapasitas saluran atau sungai tidak mampu menampung volume air sehingga air menjadi meluap keluar melewati batas tanggul sungai. Biasanya daerah yang sering terkena banjir ini adalah daerah yang berada di sekitar aliran sungai yang letaknya cukup rendah atau merupakan dataran banjir. Jika di daerah perkotaan banjir biasanya disebabkan oleh kapasitas *drainase* atau saluran air yang tidak mampu menampung lagi air hujan atau tersumbatnya saluran air akibat penumpukan sampah.

2. Banjir Lokal

Banjir lokal adalah banjir yang disebabkan oleh tingginya curah hujan dalam periode waktu tertentu (intensitas hujan) yang dapat menggenangi suatu daerah yang relatif lebih rendah. Jenis banjir ini dapat terjadi pada daerah ledakan atau cekungan fluvial yang memiliki kelembaban tanah yang tinggi sehingga pada waktu terjadi hujan lebat, peresapan air ke dalam tanah sangat kecil.

3. Banjir yang disebabkan oleh pasang surut air laut

Jenis ini terjadi pada daerah dataran aluvial pantai yang letaknya cukup rendah atau berupa cekungan dan terdapat muara sungai dengan anak-anak sungainya sehingga jika terjadi pasang dari laut atau 'rob' maka air laut atau sungai akan menggenangi daerah tersebut. Banjir ini tidak disebabkan oleh hujan sehingga walaupun pada saat musim kemarau banjir ini bisa terjadi.

1.7.1.2 Kriteria Parameter Kerawanan Banjir

Kerawanan banjir adalah keadaan yang menggambarkan mudah atau tidaknya suatu daerah terkena banjir dengan didasarkan pada faktor-faktor atau kriteria yang mempengaruhi banjir antara:

1. Kemiringan Lahan/Kelerengan

Kelerengan atau kemiringan lahan merupakan perbandingan persentase antara jarak vertikal (tinggi lahan) dengan jarak horizontal (Panjang lahan datar). Semakin landai kemiringan lerengnya maka semakin berpotensi mengalami banjir, begitu pula sebaliknya. Semakin terjal kemiringannya, maka semakin aman akan bencana banjir (Pratama et al., 2020).

2. Ketinggian Lahan/Elevasi

Ketinggian (Elevasi) Lahan adalah ukuran tinggi daerah di atas permukaan laut. Ketinggian dapat mempengaruhi terjadinya banjir. Semakin rendah suatu daerah maka semakin berpotensi mengalami banjir, begitu pula sebaliknya semakin tinggi suatu daerah, maka semakin aman akan bencana banjir (Pratama et al., 2020).

3. Jenis Tanah

Jenis tanah memiliki peran penting dalam mempengaruhi risiko dan dampak banjir. Tanah yang berstruktur baik dengan kemampuan drainase yang tinggi, seperti tanah berpasir atau tanah lempung berpasir, biasanya lebih efektif dalam menyerap air hujan dan mengurangi aliran permukaan. Sebaliknya, tanah liat atau tanah dengan kandungan lempung tinggi memiliki sifat yang cenderung lebih padat dan tidak mudah menyerap air, yang mengakibatkan air cenderung menggenang di permukaan dan meningkatkan risiko banjir. Selain itu, tanah yang sudah jenuh air akibat curah hujan yang tinggi dalam jangka waktu lama juga dapat memperburuk kondisi banjir karena tidak mampu menyerap lebih banyak air (Latue et al., 2023).

4. Curah Hujan

Merupakan banyaknya air hujan yang turun pada suatu daerah dalam kurun waktu tertentu. Curah hujan rata-rata di seluruh wilayah bersangkutan diperlukan perancangan pengendalian banjir, bukan hanya curah hujan pada suatu titik. Curah hujan tersebut biasa disebut curah hujan wilayah atau daerah (Pratama et al., 2020).

5. Penggunaan Lahan

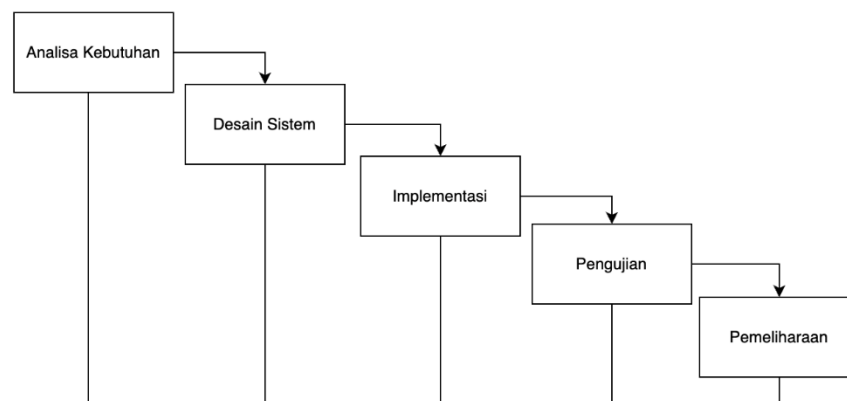
Penggunaan lahan dapat mempengaruhi kerentanan suatu wilayah terhadap banjir, dan penggunaan lahan berperan dalam volume banjir yang terjadi akibat curah hujan yang meningkatkan infiltrasi. Lahan bervegetasi menyerap lebih banyak air hujan dan membutuhkan waktu lebih lama untuk mencapai sungai, sehingga banjir lebih sedikit daripada di area bervegetasi. Kandungan bahan organik pada kebun di lereng memiliki nilai bobot.

6. Kerapatan Sungai

Kerapatan aliran sungai menggambarkan kapasitas penyimpanan air permukaan dalam cekungan – cekungan seperti danau, rawa dan badan sungai yang mengalir di suatu daerah aliran sungai (DAS). Berdasarkan Peraturan Direktur Jendral NOMOR : P.3/V-SET/2013 dalam pedoman identifikasi Karakteristik Daerah Aliran Sungai Kerapatan Sungai adalah suatu angka indeks yang menunjukkan banyaknya anak sungai di dalam suatu DAS (Pamuji et al., 2022).

1.7.2 Pengembangan Sistem Metode *Waterfall*

Pada penelitian ini digunakan metode *Waterfall* dalam pengembangan sistem, adapun tahapannya adalah sebagai berikut (Wahid, 2020):



Gambar 1.2 Metode *Waterfall*

1. *Requirement*

Pada tahap ini, pengembang sistem melakukan komunikasi untuk memahami perangkat lunak yang diharapkan oleh pengguna serta batasannya. Informasi dikumpulkan melalui wawancara, diskusi, atau survei langsung, kemudian dianalisis untuk memperoleh data yang dibutuhkan oleh pengguna.

2. *Design*

Di tahap ini, pengembang membuat desain sistem yang mencakup penentuan perangkat keras dan persyaratan sistem. Desain ini juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. *Implementation*

Pada tahap ini, sistem dikembangkan dalam bentuk unit-unit kecil yang terintegrasi pada tahap selanjutnya. Setiap unit dikembangkan dan diuji fungsionalitasnya.

4. *Verification*

Sistem diuji dan diverifikasi untuk memastikan bahwa sistem memenuhi persyaratan yang telah ditetapkan. Pengujian dilakukan dalam dengan cara uji *black box*.

5. *Maintenance*

Ini adalah tahap akhir dari metode waterfall. Perangkat lunak yang telah jadi dijalankan dan dipelihara, termasuk memperbaiki kesalahan yang mungkin tidak terdeteksi pada langkah sebelumnya.


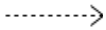



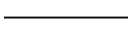


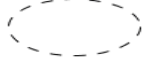
1.7.3 Perancangan Sistem *Unified Modeling Language* (UML)

Unified Modeling Language (UML) merupakan diagram yang masing-masing memiliki aturan-aturan tertentu dalam penyusunan. Diagram - diagram tersebut tersusun atas sejumlah elemen grafik yang saling membentuk satu kesatuan dalam pemodelan *software*. Masing-masing diagram UML mempresentasikan berbagai sudut pandangan terhadap sistem dan mendefenisikan apa yang dikerjakan oleh sistem, bukan bagaimana sistem berkerja (Rosa, 2018).


1.7.3.1 *Use Case Diagram*

Use case adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Secara singkat *use case* adalah serangkaian skenario yang digabungkan bersama-sama oleh tujuan umum pengguna. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel 1.1 *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Actor</i>	Tentukan set peran yang dimainkan pengguna saat berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</i>	Suatu hubungan di mana perubahan-perubahan yang terjadi pada suatu unsur yang berdiri sendiri akan mempengaruhi unsur-unsur yang bergantung
3.		<i>Generalization</i>	Objek anak (Keturunan) berbagi perilaku dan hubungan struktur data objek di atas induknya (nenek moyang).
4.		<i>Include</i>	Secara eksplisit mendefinisikan kasus penggunaan sumber daya.
5.		<i>Extend</i>	Menentukan perilaku kasus penggunaan target memperluas kasus penggunaan sumber pada titik tertentu
6.		<i>Association</i>	Menghubungkan suatu objek dengan objek lainnya
7.		<i>System</i>	Menentukan paket dengan sistem terbatas
8.		<i>Use Case</i>	Deskripsi urutan Tindakan yang dilakukan oleh sistem yang mengarah pada hasil yang terukur untuk aktor.
9.		<i>Collaboration</i>	Interaksi aturan dan elemen lain yang bersama-sama memberikan perilaku yang lebih besar dari jumlah dan elemennya (sinergi)


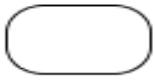


Tabel 1.1 Lanjutan

NO	GAMBAR	NAMA	KETERANGAN
10.		<i>Note</i>	Elemen fisik yang ada saat aplikasi berjalan dan mewakili sumber daya komputer.


1.7.3.2 Activity Diagram

Activity diagram adalah Teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa.

Tabel 1.2 Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.

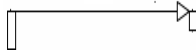
Tabel 1.2 Lanjutan

NO	GAMBAR	NAMA	KETERANGAN
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.

1.7.3.3 Sequence Diagram

Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu, Diagram ini secara khusus berasosiasi dengan use-case. Sequence diagram memperlihatkan tahap demi tahap apa yang harus terjadi untuk menghasikan suatu didalam *use-case diagram*.


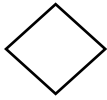
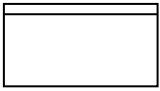


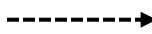

Tabel 1.3 Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek t

1.7.3.4 Class Diagram

Class diagram merupakan *diagram* yang selalu ada di pemodelan sistem berorientasi obyek. *Class diagram* menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan. *Class diagram* umumnya tersusun dari elemen *class*, yaitu:

Tabel 1.4 Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Generalization</i>	Hubungan di mana objek anak (keturunan) berbagi perilaku dan struktur data objek di atas orang tua (leluhur).
2.		<i>Nary Assosiation</i>	Cobalah untuk menghindari bergaul dengan lebih dari 2 objek.
3.		<i>Class</i>	Kumpulan objek yang memiliki property dan operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari serangkaian Tindakan yang dilakukan oleh sistem. Ini memberikan subjek hasil yang terukur.
5.		<i>Realization</i>	Apa yang sebenarnya dilakukan objek.
6.		<i>Dependency</i>	Suatu hubungan dimana perubahan – perubahan yang terjadi pada suatu elemen independent akan mempengaruhi elemen – elemen yang bergantung pada elemen – elemen yang tidak independen
7.		<i>Association</i>	Apa yang menghubungkan satu makhluk dengan benda lainnya

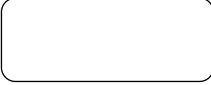



1.7.4 Flowchart

Flowchart adalah alat visual yang sangat berguna dalam memvisualisasikan dan memahami alur kerja suatu proses. Selain itu, *flowchart* juga memungkinkan untuk melakukan analisis terhadap proses tersebut, seperti mengidentifikasi titik-titik yang memungkinkan terjadinya *bottleneck* atau perbaikan efisiensi. Dengan




menggunakan *flowchart*, tim atau individu dapat dengan mudah berkolaborasi dalam merancang atau memperbaiki suatu proses, karena *flowchart* menyediakan gambaran yang jelas tentang urutan langkah-langkah yang diperlukan.

Salah satu keuntungan utama dari penggunaan *flowchart* adalah kemampuannya untuk menyederhanakan proses yang kompleks menjadi bagian-bagian yang lebih kecil dan lebih mudah dimengerti. Dengan memecah proses menjadi langkah-langkah yang lebih sederhana, *flowchart* membantu menghilangkan kebingungan dan meningkatkan pemahaman terhadap proses tersebut. Hal ini sangat bermanfaat dalam situasi di mana terdapat banyak variabel atau langkah-langkah yang terlibat, seperti dalam pengembangan sistem informasi seperti pemetaan daerah rawan banjir di Kota Surakarta.

Tabel 1.5 Simbol *Flowchart*

SIMBOL	NAMA	KETERANGAN
	Terminator Awal / Akhir Program	Simbol untuk memulai dan mengakhiri suatu program
	Dokumen	Menunjukkan dokumen berupa dokumen input dan output pada proses manual dan proses berbasis komputer
	Proses Manual	Menunjukkan kegiatan proses yang dilakukan secara manual
	Proses Komputer	Menunjukkan kegiatan proses yang dilakukan secara komputerisasi

Tabel 1.5 Lanjutan

SIMBOL	NAMA	KETERANGAN
	Arah Aliran Data	Menunjukkan arah aliran dokumen antar bagian yang terkait pada suatu system
	Penyimpanan Manual	Menunjukkan media penyimpanan data / informasi secara manual
	Data	Simbol input/output digunakan untuk mewakili data input/output

1.8 Perangkat Keras (*Hardware*) Dan Perangkat Lunak (*Software*) Yang Digunakan

Berikut perangkat keras dan perangkat lunak yang digunakan dalam mendukung jalannya penelitian ini:

1.8.1 Perangkat Keras (*Hardware*)

Tabel 1.6 Perangkat Keras

No.	Perangkat Keras	Unit	Spesifik
1.	Laptop	1	<i>Processors inter i3 core, SSD 512, RAM 4 GB</i>
2.	<i>Smartphone</i>	1	RAM 6 GB

1.8.2 Perangkat Lunak (*Software*)

Tabel 1.7 Perangkat Lunak

No.	Perangkat Lunak	Unit	Spesifik
1.	Sistem Operasi	1	Windows 11, Android 11
2.	Text Editor	1	Visual Studio Code
3.	Hosting Server	1	Basic Hosting Server

1.9 Pengujian Sistem Metode *Black Box*

Pengujian *black box* adalah metode pengujian perangkat lunak yang fokus pada pengujian fungsionalitas aplikasi tanpa mempertimbangkan struktur internal atau implementasi kode. Dalam pengujian ini, tester bertindak sebagai pengguna akhir dan menguji sistem berdasarkan spesifikasi dan kebutuhan fungsional yang telah ditentukan. Tujuan utamanya adalah memastikan bahwa aplikasi bekerja sesuai dengan persyaratan yang telah ditetapkan, meskipun tester tidak memiliki pengetahuan tentang bagaimana fungsi-fungsi tersebut diimplementasikan dalam kode (Mintarsih, 2023).

Salah satu keuntungan utama dari pengujian *black box* adalah kemampuannya untuk mendeteksi masalah pada antarmuka pengguna dan integrasi antara berbagai komponen sistem. Dengan berfokus pada fungsionalitas dan hasil *output* dari berbagai *input*, pengujian ini dapat mengidentifikasi kesalahan yang mungkin tidak terlihat dalam pengujian unit atau pengujian sistem internal. Hal ini sangat penting untuk memastikan bahwa aplikasi berfungsi dengan benar dari perspektif pengguna dan memenuhi kebutuhan yang telah ditentukan.

Pengujian *black box* juga memiliki keterbatasan. Karena penguji tidak melihat kode sumber atau logika internal, sulit untuk mengetahui apakah semua jalur kode telah diuji atau jika ada kondisi tepi yang tidak tercover. Ini dapat menyebabkan pengujian menjadi tidak lengkap jika tidak ada dokumentasi spesifikasi yang memadai atau jika penguji tidak memiliki pemahaman yang cukup tentang fungsionalitas yang diuji.

Untuk memaksimalkan efektivitas pengujian *black box*, penting untuk menyusun kasus uji yang komprehensif berdasarkan spesifikasi fungsional dan kebutuhan pengguna. Pengujian harus mencakup berbagai kondisi input, termasuk kasus batas dan input yang tidak valid, untuk memastikan bahwa sistem dapat menangani semua kemungkinan scenario dengan benar. Dengan pendekatan ini, pengujian *black box* dapat memberikan gambaran yang jelas tentang apakah aplikasi memenuhi ekspektasi pengguna dan berfungsi sesuai dengan yang diharapkan.

1.10 Jadwal Kegiatan

Untuk menyelesaikan tugas akhir ini penulis telah menyiapkan rencana kegiatan yang berfungsi agar semua kegiatan penelitian dapat berjalan sesuai dengan yang penulis harapkan dan selesai dengan tepat waktu.

