

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. Kajian Teoritis**

##### **1. Sistem Informasi**

Sistem informasi merupakan kesatuan yang menghubungkan manusia, perangkat keras, perangkat lunak, basis data, dan prosedur kerja. Setiap komponen memiliki fungsi yang saling mendukung dalam menjalankan kegiatan organisasi. Seluruh komponen tersebut bekerja bersama untuk mengolah data menjadi informasi yang dapat digunakan dalam kegiatan operasional, pengawasan, dan pengambilan keputusan (Ikhwan & Putri Lubis, 2023:1).

Perkembangan teknologi membuat sistem informasi tidak hanya digunakan untuk menyimpan data, tetapi juga meningkatkan efisiensi proses kerja. Sistem yang saling terhubung dapat menghasilkan informasi yang akurat, relevan, dan tersedia secara real-time. Penerapannya dapat mengurangi kesalahan pencatatan, mempercepat penyajian data, serta meningkatkan kualitas pelayanan kepada pengguna (Setiawan et al., 2023:986).

Sistem informasi berperan sebagai sarana untuk mengelola data secara terstruktur agar menghasilkan informasi yang berguna. Dalam penelitian ini, sistem informasi digunakan untuk mendukung pengelolaan pengaduan lingkungan, mulai dari penerimaan hingga penyelesaian laporan.

Penerapan sistem informasi diharapkan dapat membuat pelayanan pengaduan menjadi lebih cepat, teratur, dan transparan.

## **2. Website**

Website merupakan kumpulan halaman digital yang saling terhubung dan dapat diakses melalui internet menggunakan perangkat peramban. Berdasarkan pengelolaan kontennya, website dibedakan menjadi website statis dan dinamis. Website statis memerlukan perubahan kode untuk memperbarui konten. Sementara itu, website dinamis memungkinkan data diperbarui melalui sistem tanpa harus mengubah kode program secara langsung (Sinlae et al., 2024).

Dalam pelayanan publik, website memberikan kemudahan kepada masyarakat untuk mengakses informasi dan layanan pemerintah tanpa dibatasi tempat dan waktu. Bagi instansi, website dapat digunakan untuk mengelola dan mendokumentasikan data secara terpusat. Pemanfaatan tersebut mendukung penyelenggaraan pelayanan yang lebih terbuka, efektif, dan dapat dipertanggungjawabkan (Setiaji et al., 2024:42).

Website berperan sebagai media digital yang mendukung penyampaian informasi dan pelayanan secara luas. Website dinamis dipilih dalam penelitian ini karena memungkinkan masyarakat mengirimkan pengaduan dan petugas mengelola laporan secara langsung. Penggunaannya juga mendukung pembaruan status penanganan laporan secara lebih cepat dan mudah.

### **3. Pengaduan Masyarakat**

Pengaduan masyarakat merupakan fasilitas yang diberikan kepada warga untuk menyampaikan keluhan, laporan, atau aspirasi kepada instansi pemerintah. Pengaduan tersebut dapat berkaitan dengan pelayanan publik maupun pelanggaran yang terjadi di lingkungan masyarakat. Selain menjadi sarana penyampaian keluhan, pengaduan juga berfungsi sebagai bentuk pengawasan sosial dan partisipasi masyarakat dalam penyelenggaraan pemerintahan (Halid & Baijuri, 2025).

Dalam bidang lingkungan hidup, pengaduan masyarakat dapat berupa laporan pencemaran udara, pencemaran air, pembuangan sampah ilegal, dan bentuk kerusakan lingkungan lainnya. Dinas Lingkungan Hidup sebagai instansi yang berwenang memiliki kewajiban untuk menerima, memeriksa, dan menindaklanjuti laporan tersebut secara cepat dan terstruktur. Penggunaan sistem pengaduan digital terbukti dapat mempercepat respons terhadap laporan masyarakat (Suharsono, 2022:509).

Pengaduan masyarakat menjadi penghubung antara warga dan instansi pemerintah dalam menangani permasalahan lingkungan. Sistem pengaduan digital diperlukan agar masyarakat dapat menyampaikan laporan dengan lebih mudah. Sistem tersebut juga membantu petugas menangani pengaduan secara teratur sekaligus meningkatkan partisipasi masyarakat dalam pengawasan lingkungan.

#### **4. Tracking Status**

Tracking status merupakan fitur yang digunakan untuk mengetahui perkembangan suatu proses secara bertahap. Dalam sistem pengaduan, fitur ini memberikan informasi kepada pelapor mengenai posisi laporan, mulai dari penerimaan, verifikasi, proses penanganan, hingga dinyatakan selesai. Informasi tersebut membantu masyarakat mengetahui tindakan yang telah dilakukan terhadap laporan yang disampaikan (Zuliyanto & Asriningtias, 2025).

Fitur pemantauan status juga membantu instansi dalam mencatat, mengelompokkan, dan menyalurkan laporan kepada petugas yang bertanggung jawab. Setiap perubahan status dapat disimpan dalam riwayat terpusat sehingga proses penanganan pengaduan dapat dipantau dan dievaluasi secara berkala. Hasil evaluasi tersebut dapat digunakan untuk meningkatkan kualitas pelayanan kepada masyarakat (Liesnaningsih et al., 2021:276).

Tracking status menjadi fitur penting dalam meningkatkan keterbukaan pelayanan pengaduan. Penerapan fitur ini memungkinkan masyarakat memperoleh kepastian mengenai perkembangan laporan yang disampaikan. Selain itu, instansi dapat mengawasi setiap tahap penanganan pengaduan secara lebih sistematis dan berkelanjutan.

#### **5. Pengujian Perangkat Lunak (IEEE 29119)**

IEEE 29119 merupakan standar internasional untuk pengujian perangkat lunak yang diterbitkan oleh Institute of Electrical and Electronics

Engineers. Standar ini memberikan pedoman mengenai konsep, proses, dokumentasi, dan teknik pengujian perangkat lunak. Penerapannya bertujuan agar kegiatan pengujian dilakukan secara sistematis, terukur, dan sesuai dengan kebutuhan fungsional maupun nonfungsional sistem (Rachmadi et al., 2025:847).

IEEE 29119 digunakan untuk menguji seluruh fungsi sistem informasi pengaduan lingkungan, termasuk fitur tracking status. Tahapan pengujian meliputi perencanaan pengujian, penyusunan kasus uji, pelaksanaan pengujian, dan penyusunan laporan hasil pengujian. Penggunaan standar tersebut membantu memastikan bahwa setiap fitur dapat berjalan sesuai dengan kebutuhan pengguna (Mulya et al., 2025:53).

IEEE 29119 berfungsi sebagai pedoman untuk menjaga kualitas perangkat lunak melalui proses pengujian yang terarah. Tahapan pengujian yang terstruktur memungkinkan kesalahan pada sistem ditemukan dan diperbaiki sebelum digunakan. Dengan demikian, sistem yang diserahkan kepada Dinas Lingkungan Hidup Kota Madiun diharapkan telah memenuhi kebutuhan dan memiliki tingkat keandalan yang baik.

## **6. Framework Laravel**

Laravel merupakan framework pengembangan aplikasi web berbasis PHP yang menerapkan pola Model, View, dan Controller atau MVC. Pola tersebut memisahkan pengelolaan data, tampilan pengguna, dan logika aplikasi. Pemisahan ini membuat struktur kode lebih teratur, mempermudah

pemeliharaan, dan mendukung proses pengembangan aplikasi secara lebih cepat (Sinlae et al., 2024:120).

Dalam penelitian ini, Laravel digunakan sebagai framework pada bagian backend sistem. Laravel menangani autentikasi pengguna, pengelolaan data pengaduan, perubahan tracking status, dan komunikasi dengan basis data PostgreSQL melalui Application Programming Interface. Laravel dipilih karena memiliki dukungan keamanan, skalabilitas, dan ekosistem pengembangan yang lengkap (Sinlae et al., 2024:123).

Laravel menjadi framework yang sesuai untuk membangun bagian backend sistem informasi pengaduan lingkungan. Struktur MVC dan berbagai fitur yang tersedia membantu pengembang menghasilkan sistem yang terorganisasi, aman, serta mudah dipelihara. Penggunaan Laravel juga mendukung pengelolaan proses bisnis dan pertukaran data antara frontend dengan basis data.

## **7. Framework Vue.js**

Vue.js merupakan framework JavaScript progresif yang digunakan untuk membangun antarmuka pengguna berbasis web. Framework ini menerapkan konsep reactive data binding dan component-based architecture. Konsep tersebut memungkinkan pengembang membuat tampilan dinamis dalam bentuk komponen yang terstruktur dan dapat digunakan kembali. Vue.js juga dikenal ringan, fleksibel, dan relatif mudah dipelajari (Rahman et al., 2024:598).

Vue.js digunakan untuk mengembangkan bagian frontend sistem pengaduan lingkungan. Kemampuan reaktivitas Vue.js dimanfaatkan untuk menampilkan perubahan status laporan secara real-time. Dengan demikian, pengguna dapat melihat perkembangan penanganan pengaduan tanpa harus memuat ulang halaman secara manual (Rahman et al., 2024).

Vue.js berperan sebagai framework yang mendukung pembuatan tampilan sistem yang responsif dan interaktif. Penggunaan Vue.js membantu memberikan pengalaman yang lebih baik ketika masyarakat mengirimkan pengaduan dan memantau status laporan. Framework ini juga mempermudah pengembang dalam mengelola komponen antarmuka agar tetap terstruktur dan mudah dipelihara.

## **8. PostgreSQL**

PostgreSQL merupakan sistem manajemen basis data relasional yang bersifat open-source dan mendukung standar SQL. PostgreSQL memiliki kemampuan dalam mengelola data kompleks, menjaga kestabilan transaksi, mendukung berbagai tipe data, serta menangani pertumbuhan jumlah data. Mekanisme ACID yang diterapkan juga membantu menjaga keamanan dan konsistensi data di dalam sistem (Setiawan et al., 2023:986).

Dalam sistem informasi pengaduan lingkungan, *PostgreSQL* digunakan untuk menyimpan data pengguna, laporan pengaduan, riwayat perubahan status, dan data administrasi lainnya. Kemampuannya dalam mengelola data transaksional menjadikan PostgreSQL sesuai untuk sistem yang memerlukan integritas dan ketepatan data tinggi. Selain itu, dukungan

keamanan PostgreSQL membantu melindungi data yang tersimpan dari akses yang tidak sesuai (Setiawan et al., 2023).

PostgreSQL menjadi basis data yang andal untuk mendukung sistem informasi pengaduan lingkungan. Penggunaannya membantu memastikan bahwa seluruh data tersimpan secara terstruktur, konsisten, aman, dan dapat diakses sesuai kebutuhan sistem. Kemampuan tersebut menjadikan PostgreSQL sesuai untuk mendukung pengelolaan data pengaduan yang terus bertambah.

## 9. Metode Agile Scrum

*Agile Scrum* merupakan metode pengembangan perangkat lunak yang menerapkan pendekatan iteratif dan inkremental. Proses pengembangan dibagi menjadi beberapa periode kerja yang disebut sprint dan umumnya berlangsung selama dua hingga empat minggu. Pada setiap sprint, tim menghasilkan bagian produk yang dapat diperiksa dan dievaluasi oleh pemangku kepentingan sehingga perubahan kebutuhan dapat ditangani dengan lebih cepat (Santoso et al., 2022:209).

Scrum memiliki tiga peran utama, yaitu Product Owner, Scrum Master, dan Development Team. Product Owner bertanggung jawab menentukan dan memprioritaskan kebutuhan produk. Scrum Master memastikan proses Scrum berjalan dengan baik, sedangkan Development Team melaksanakan pengembangan sistem. Kegiatan Scrum meliputi sprint planning, daily scrum, sprint review, dan sprint retrospective (Santoso et al., 2022:210).

Agile Scrum menjadi metode yang memungkinkan sistem dikembangkan dan dievaluasi secara bertahap. Dalam penelitian ini, Scrum membantu pengembang menyesuaikan sistem informasi pengaduan lingkungan dengan kebutuhan pengguna. Penyesuaian tersebut dilakukan melalui proses perencanaan, pembangunan, pengujian, dan evaluasi pada setiap sprint.



Gambar 2. 1 Metode Agile Scrum

Sumber : Website binar mengenali konsep agile scrum dan sprint

## 10. *Flowchart*

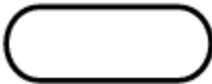
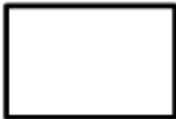
Flowchart merupakan diagram yang digunakan untuk menggambarkan urutan kegiatan atau alur kerja dalam suatu sistem. Diagram ini menggunakan simbol-simbol standar untuk menunjukkan hubungan antaraktivitas dan urutan logis sebuah proses. Penyajian secara visual dapat membantu pengembang dan pemangku kepentingan memahami rancangan sistem sebelum tahap implementasi dilakukan (Syani et al., 2020).

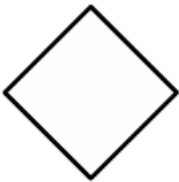
Beberapa simbol yang umum digunakan dalam flowchart meliputi terminator untuk menunjukkan awal dan akhir proses, process untuk

menggambarkan pengolahan data, decision untuk menunjukkan percabangan, serta input/output untuk menggambarkan data masukan dan keluaran. Penggunaan simbol yang tepat membantu pengembang menjelaskan alur sistem secara sistematis. Alur yang jelas dapat mengurangi kesalahan dalam proses perancangan sistem (Smrti et al., 2023:56).

Flowchart berperan sebagai alat bantu visual untuk menjelaskan alur kerja sistem secara jelas dan berurutan. Dalam penelitian ini, flowchart digunakan untuk menggambarkan proses pengaduan lingkungan, mulai dari laporan dikirimkan hingga selesai ditangani. Penggunaan flowchart membantu pengembang dan pihak instansi memahami setiap tahapan yang terdapat dalam proses pengaduan (Huda, 2023) .

Tabel 2.1 Simbol-simbol Flowchart

| Simbol                                                                              | Nama                             | Fungsi                                                                                                    |
|-------------------------------------------------------------------------------------|----------------------------------|-----------------------------------------------------------------------------------------------------------|
|  | Aliran Data /<br><i>Flowline</i> | Simbol garis alir (flow lines symbol) digunakan untuk menunjukkan arus dari proses                        |
|  | Terminal /<br><i>Terminator</i>  | Simbol Titik Terminal (terminal point symbol) Digunakan untuk menunjukkan awal dan akhir dan suatu proses |
|  | Proses / <i>Process</i>          | Proses Simbol proses digunakan untuk mewakili suatu proses                                                |

|                                                                                     |                             |                                                                                                                                                                  |
|-------------------------------------------------------------------------------------|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Keputusan / <i>Decision</i> | Simbol keputusan (decision) Digunakan untuk penyeleksian kondisi di dalam program                                                                                |
|    | Konektor                    | Simbol penghubung (connector symbol) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman yang berbeda |
|  | <i>Input/Output</i>         | Input/Output Simbol digunakan untuk mewakili data input/output                                                                                                   |

Sumber : (Smrti et al., 2023)

## 11. UML (Unified Modeling Language)

Sebagai instrumen pemodelan visual, UML (*Unified Modeling Language*) berfungsi menjembatani komunikasi teknis dalam rekayasa perangkat lunak berbasis objek. Kehadiran bahasa pemodelan standar ini mempermudah penggambaran spesifikasi, struktur internal, dan dinamika operasional suatu sistem informasi. Skema penulisan rancangan yang sistematis ini memastikan bahwa seluruh elemen sistem, keterkaitan antar-komponen, dan logika proses bisnis terdokumentasi dengan baik, sehingga meminimalkan risiko miskomunikasi selama siklus hidup pengembangan perangkat lunak berlangsung (Ramdany et al., 2024:31).

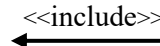
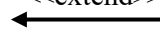
### a. Use Case Diagram

Dalam memvisualisasikan arsitektur fungsional perangkat lunak, use case diagram dimanfaatkan untuk merepresentasikan kebutuhan sistem yang telah dirumuskan pada tahap analisis. Struktur diagram ini memperlihatkan korelasi antara entitas eksternal (aktor) dengan rangkaian layanan yang disiapkan oleh sistem, sehingga memberikan pemahaman yang komprehensif mengenai batasan ruang lingkup aplikasi dan mekanisme interaksi di dalamnya. Kehadiran diagram ini membantu pihak pengembang untuk memproyeksikan peran administrator maupun user umum terhadap seluruh fitur yang ada. Akibatnya, dokumen visual ini berfungsi sebagai panduan baku yang mengarahkan proses perancangan dan implementasi kode program (Ramdany et al., 2024:4).

*Use Case Diagram* berfungsi memvisualisasikan interaksi aktor dengan sistem guna memperjelas batasan dan ruang lingkup kerja platform yang dirancang. Diagram ini memaparkan daftar modul operasional beserta hak akses yang dimiliki oleh tiap pengguna. Komponen penyusunnya terdiri atas dua elemen primer, yaitu aktor (pengguna, proses, atau sistem luar) dan *use case* (layanan yang disediakan). Untuk menggambarkan logika keterkaitan antar-fitur secara spesifik, digunakan beberapa bentuk relasi baku, di antaranya *include*, *extend*, serta *generalization* sesuai dengan spesifikasi rancangan arsitektur aplikasi (Ramdany et al., 2024).

Use case diagram berfungsi untuk menjelaskan fitur dan hak akses setiap pengguna. Dalam sistem pengaduan lingkungan, diagram ini menggambarkan interaksi masyarakat dan petugas dengan fitur yang tersedia. Fitur tersebut meliputi pengiriman laporan, verifikasi, perubahan status, dan pemantauan pengaduan.

Tabel 2.2 Simbol Use Case Diagram

| Simbol                                                                              | Elemen       | Keterangan                                                                                                                                          |
|-------------------------------------------------------------------------------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Aktor        | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri    |
|   | Use Case     | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor                                            |
|  | Association  | Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.                               |
|  | Generalisasi | Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
|  | Include      | Include atau uses berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan.                                        |
|  | Extend       | Menunjukkan bahwa tambahan akan selalu melakukan pengecekan apakah use case yang ditambahkan telah dijalankan sebelum use case tambahan dijalankan. |

Sumber : (Ramdany et al., 2024:34)

## b. Sequence Diagram

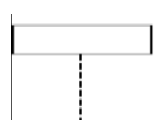
Sequence Diagram adalah bagian dari permodelan UML yang digunakan untuk menggambarkan interaksi antar-objek dalam sistem melalui pertukaran pesan yang terstruktur berdasarkan urutan waktu. Diagram ini memaparkan alur komunikasi logis dari setiap objek dalam mengeksekusi suatu proses bisnis mulai dari pengiriman data hingga penerimaan pesan balik sesuai dengan skenario pada *use case*. Penyusunan *sequence diagram* memerlukan identifikasi objek-objek yang terlibat beserta operasi atau metode yang dimiliki sehingga alur interaksi dapat divisualisasikan secara jelas dan menjadi acuan dalam memahami proses kerja sistem (Syahrizal, 2025:164). .

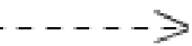
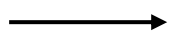
Struktur penyusunan *sequence diagram* melibatkan beberapa elemen utama yaitu *actor*, *lifeline*, *object*, serta *message*. Peran *actor* di sini mewakili pihak pengguna atau sistem eksternal yang menginisiasi interaksi. Selanjutnya, *lifeline* bertindak sebagai penunjuk durasi keaktifan suatu entitas dalam rentang waktu tertentu. Untuk komponen *object*, fungsinya adalah memodelkan entitas internal sistem yang ikut serta dalam menerima atau mengirimkan instruksi. Terakhir, *message* dimanfaatkan sebagai representasi visual dari sinyal komunikasi dan perpindahan data yang berlangsung di antara elemen-elemen tersebut saat sistem beroperasi. Selain itu, *sequence diagram* juga mengenal beberapa jenis pesan, seperti *return* yang menunjukkan hasil dari suatu operasi, *send* yang menggambarkan proses pengiriman data atau informasi ke objek lain, dan

*call* yang menunjukkan pemanggilan operasi atau metode pada objek lain maupun pada objek itu sendiri (Syahrizal, 2025).

Sequence diagram berfungsi untuk menjelaskan urutan komunikasi antarbagian sistem secara terperinci. Dalam penelitian ini, sequence diagram membantu menggambarkan proses pengiriman pengaduan, pemeriksaan laporan, dan perubahan status. Diagram tersebut juga memperlihatkan proses penyampaian informasi perkembangan pengaduan kepada pelapor.

Tabel 2.3 Simbol-simbol Sequence Diagram

| Simbol                                                                                                                                                                                     | Elemen          | Keterangan                                                                                                                      |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|---------------------------------------------------------------------------------------------------------------------------------|
|                                                                                                         | Aktor           | Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri. |
|                                                                                                         | <i>LifeLine</i> | Objek entity, antar muka yang saling berinteraksi                                                                               |
| <div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama Objek</div><br><div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama Kelas</div> | Objek           | Menyatakan objek yang berinteraksi oleh pesan.                                                                                  |
|                                                                                                         | <i>Message</i>  | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi                          |

|                                                                                   |                        |                                                                                                                                                                                       |
|-----------------------------------------------------------------------------------|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | Pesan tipe<br>return   | 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. |
|  | Pesan tipe send        | Menyatakan bahwa suatu objek mengirim data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.                                                       |
|  | Pesan tipe <i>call</i> | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.                                                                                        |

Sumber : (Syahrizal, 2025)

### c. Activity Diagram

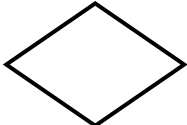
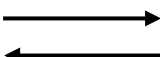
Sebagai bagian dari metodologi pemodelan visual, *activity diagram* berfungsi menggambarkan urutan kronologis dari aktivitas-aktivitas yang membentuk suatu proses sistem. Karakteristik utama dari diagram ini terletak pada fokus visualisasinya yang mengarah pada tindakan-tindakan logis yang dilakukan oleh sistem itu sendiri. Hal ini mempermudah perancang untuk melacak titik awal sebuah alur, percabangan aktivitas di dalam sistem, hingga parameter penutup yang menandakan berakhirnya siklus proses tersebut. *Activity diagram* membantu menggambarkan aliran kerja sistem secara sistematis sehingga lebih mudah dipahami dalam tahap analisis maupun perancangan perangkat lunak (Syahrizal, 2025:158).

*Activity diagram* menggunakan berbagai simbol untuk merepresentasikan alur aktivitas yang terjadi dalam suatu sistem. Elemen-elemen tersebut meliputi *activity* untuk menggambarkan tindakan logis sistem, serta *control flow* yang mengarahkan urutan perpindahan dari satu tindakan ke tindakan berikutnya. Aliran data maupun dokumen yang bergerak di antara proses-proses tersebut diwakili oleh *object flow*, sementara batas awal dan akhir dari skenario proses ditandai melalui *start point* dan *end point*. Di samping itu, untuk memfasilitasi proses yang berjalan secara *concurrent*, digunakan mekanisme *fork* untuk membagi alur dan *join* untuk menggabungkannya kembali. Adapun penentuan arah alur berdasarkan parameter atau pengujian logika tertentu direpresentasikan dengan notasi *decision* (Syahrizal, 2025).

Activity diagram berfungsi sebagai alat untuk memvisualisasikan tahapan dan keputusan dalam suatu proses. Pada sistem pengaduan lingkungan, diagram ini digunakan untuk menggambarkan alur pengaduan dari proses pelaporan hingga selesai. Penggunaan diagram tersebut membantu memperjelas aktivitas verifikasi, penanganan, dan pembaruan status laporan.

Tabel 2.4 Simbol Activity Diagram

| Simbol                                                                              | Elemen          | Keterangan                                                                              |
|-------------------------------------------------------------------------------------|-----------------|-----------------------------------------------------------------------------------------|
|  | <i>Activity</i> | Menyatakan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.. |

|                                                                                     |                          |                                                                                             |
|-------------------------------------------------------------------------------------|--------------------------|---------------------------------------------------------------------------------------------|
|    | <i>Control Flow</i>      | Menunjukkan Urutan Eksekusi.                                                                |
|    | <i>Object Flow Start</i> | Menunjukkan aliran objek dari sebuah action atau activity ke action                         |
|    | <i>Start Point</i>       | Menyatakan bahwa objek dibentuk atau diawali. sebuah Menyatakan                             |
|    | <i>End Point</i>         | Menyatakan bahwa sebuah objek dibentuk atau diakhiri.                                       |
|  | <i>Decision</i>          | Menunjukkan penggambaran suatu keputusan/tindakan yang harus diambil pada kondisi tertentu. |
|  | <i>Line Connector</i>    | Digunakan untuk menghubungkan satu simbol dengan simbol lainnya.                            |

Sumber : (Syahrizal, 2025:158)

#### d. Class Diagram

Class Diagram adalah bagian dari permodelan UML yang berfungsi untuk menggambarkan komponen struktural dan hubungan statis antar-kelas di dalam sistem. Diagram ini memaparkan data di mana setiap kelasnya tersusun atas atribut sebagai representasi karakteristik data, serta metode sebagai perwujudan fungsi operasionalnya. Kehadiran diagram ini membantu pengembang dalam memetakan relasi dan struktur

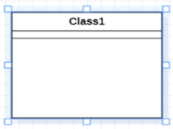
basis data yang akan diimplementasikan pada aplikasi. Penyusunan *class diagram* bertujuan memberikan gambaran mengenai rancangan struktur sistem sehingga dapat dijadikan pedoman bagi pengembang dalam mengimplementasikan perangkat lunak sesuai dengan desain yang telah direncanakan, dokumentasi perancangan tetap selaras dengan hasil implementasi sistem (Syahrizal, 2025:159).

Sebagai salah satu pilar pemodelan UML, *class diagram* memaparkan rancangan statis sistem lewat pemetaan kelas, atribut, metode, dan jalinan relasinya. Kehadiran diagram ini memberikan pemahaman mendalam terkait sifat dasar dan peran tiap entitas yang menjadi fondasi dalam siklus pengembangan aplikasi; wujud fisiknya digambarkan lewat bidang persegi yang menampung nama kelas beserta fungsi dan karakteristik datanya (Ramdany et al., 2024). Di samping itu, pemodelan ini menggunakan ragam notasi untuk mengekspresikan interaksi antar-kelas, seperti *interface*, *association*, *generalization*, *dependency*, dan *aggregation*. Tiap-tiap komponen visual tersebut membawa peran fungsional yang berbeda dalam menerjemahkan aspek keterkaitan, konsep pewarisan (inheritance), hubungan ketergantungan, hingga struktur bagian-keseluruhan (whole-part relation) di dalam arsitektur perangkat lunak (Syahrizal, 2025).

Class diagram berfungsi sebagai pedoman dalam merancang struktur data dan hubungan antarobjek. Dalam penelitian ini, class diagram digunakan untuk memodelkan kelas pengguna, pengaduan, kategori

pengaduan, riwayat status, tanggapan, dan data pendukung lainnya. Pemodelan tersebut membantu pengembang mengimplementasikan struktur basis data dan logika sistem sesuai dengan rancangan.

Tabel 2.5 Simbol Class Diagram

| Simbol                                                                              | Elemen                | Keterangan                                                                                                                                                                                   |
|-------------------------------------------------------------------------------------|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <i>Activity</i>       | Blok-blok pembangunan pada pemrograman berorientasi objek.                                                                                                                                   |
|    | <i>Association</i>    | Sebuah hubungan yang menunjukkan adanya interaksi antar class. Hubungannya                                                                                                                   |
|  | <i>Aggregation</i>    | Mengindikasikan keseluruhan bagian relationship dan biasanya disebut relasi.                                                                                                                 |
|  | <i>Composition</i>    | Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi Composition terhadap class tempat dia bergantung tersebut. |
|  | <i>Generalization</i> | Sebuah hubungan antar class yang bersifat dari khusus ke umum.                                                                                                                               |
|  | <i>Dependency</i>     | Menunjukkan operasi pada suatu class yang menggunakan class yang lain. Sebuah                                                                                                                |

Sumber : (Ramdany et al., 2024:7)

## B. Kajian Empiris

### 1. Penelitian Terdahulu

Berbagai penelitian terdahulu mengenai sistem informasi pengaduan berbasis *website* telah dikaji untuk mendukung pemodelan dalam penelitian ini. Peninjauan literatur ini dilakukan guna memetakan persamaan, perbedaan, serta menegaskan posisi dan kontribusi penelitian ini di tengah perkembangan riset yang ada. Berikut adalah ringkasan studi terdahulu yang diangkat sebagai acuan komparatif:

(Sembiring et al., 2025) melakukan penelitian berjudul “Sistem Pelayanan Pengaduan Masyarakat Berbasis Web Di Kecamatan Medan Tuntungan” yang dipublikasikan dalam Jurnal Armada Informatika. Penelitian ini berfokus pada rancang bangun sistem informasi pelayanan pengaduan masyarakat berbasis *web* guna mengoptimalkan efisiensi serta transparansi tindak lanjut laporan di lingkup pemerintahan kecamatan. Titik temu atau persamaan dengan penelitian yang sedang dilakukan terletak pada kesamaan orientasi, yakni pengembangan platform pengaduan publik berbasis digital yang ditujukan bagi instansi birokrasi pemerintah demi mengakselerasi mutu pelayanan kepada masyarakat. Perbedaannya, penelitian tersebut berfokus pada pelayanan umum tingkat kecamatan, sedangkan penelitian ini spesifik pada bidang lingkungan hidup di DLH Kota Madiun dengan penambahan fitur tracking status real-time dan standar pengujian IEEE 29119.

(Padillah & Syahputra, 2024) melakukan penelitian berjudul “Sistem Informasi Layanan Pengaduan Berbasis Web Dalam Peningkatan Pelayanan Publik Informasi Kecamatan Lubuk Begalung” yang dipublikasikan dalam Jurnal Sains dan Teknologi Informatika. Penelitian ini berfokus pada pengembangan sistem informasi layanan pengaduan berbasis *web* dalam rangka mengoptimalkan kualitas pelayanan publik di tingkat kecamatan. Keterkaitan atau persamaan mendasar dengan riset yang sedang dilakukan terletak pada kesamaan pemodelan platform, yakni pemanfaatan teknologi informasi berbasis *website* yang diimplementasikan pada instansi pemerintahan dengan orientasi yang sejalan, yaitu memacu transparansi serta efisiensi tata kelola birokrasi bagi masyarakat. Adapun perbedaannya, penelitian tersebut tidak dilengkapi fitur tracking status real-time dan tidak menggunakan metode Agile Scrum maupun standar pengujian IEEE 29119 seperti yang diterapkan dalam penelitian ini.

Hasa et al., (2023) melakukan penelitian berjudul “Sistem Pengaduan Masyarakat Terhadap Masalah Lingkungan” yang dipublikasikan dalam JIKA (Jurnal Informatika) Universitas Muhammadiyah Tangerang. Penelitian ini membangun sistem pengaduan berbasis digital yang secara khusus menangani permasalahan lingkungan hidup. Persamaan dengan penelitian ini adalah sama-sama membangun sistem informasi pengaduan berbasis digital yang berfokus pada permasalahan lingkungan hidup sebagai domain utama. Perbedaannya, penelitian tersebut belum menerapkan fitur tracking status real-time dan tidak menggunakan stack teknologi Laravel,

Vue.js, serta PostgreSQL maupun metode pengembangan Agile Scrum seperti pada penelitian ini.

Kistyawati & Wijayanti, (2022) melakukan penelitian berjudul “Sistem Informasi Pengaduan Masyarakat Berbasis Web (Studi Kasus: Kantor Balai Desa Karangrowo)” yang dipublikasikan dalam Indonesian Journal of Technology, Informatics and Science (IJTIS). Penelitian ini membangun sistem informasi pengaduan masyarakat berbasis web untuk instansi pemerintah tingkat desa guna meningkatkan efisiensi pengelolaan laporan. Persamaan dengan penelitian ini adalah sama-sama membangun sistem informasi pengaduan masyarakat berbasis website untuk instansi pemerintah dengan tujuan meningkatkan efisiensi dan transparansi pelayanan. Perbedaanannya, penelitian tersebut berfokus pada lingkup pemerintah desa dan tidak dilengkapi fitur tracking status real-time, sedangkan penelitian ini berfokus pada pengaduan lingkungan hidup di DLH Kota Madiun dengan fitur tracking status real-time dan standar pengujian internasional IEEE 29119.

Wijayanti et al., (2022) melakukan penelitian berjudul “Rancang Bangun Sistem Manajemen Pengelolaan Pengaduan Masyarakat Di Kabupaten Kudus” yang dipublikasikan dalam Journal of Computer and Information Systems Ampera. Penelitian ini membangun sistem manajemen pengelolaan pengaduan masyarakat tingkat kabupaten secara terstruktur dan terdokumentasi. Persamaan dengan penelitian ini adalah sama-sama membangun sistem informasi untuk mengelola pengaduan masyarakat secara

terstruktur pada instansi pemerintah daerah. Perbedaannya, penelitian tersebut memiliki cakupan pengaduan umum dan belum memiliki fitur tracking status real-time, sedangkan penelitian ini difokuskan pada pengaduan lingkungan hidup dengan fitur tracking status yang terintegrasi dan dibangun menggunakan teknologi Laravel, Vue.js, serta PostgreSQL.

Dari uraian diatas, memperlihatkan bahwa studi ini mempunyai kebaruan (novelty) berupa penggabungan fitur tracking status real-time khusus domain lingkungan hidup, pengembangan dengan metode Agile Scrum, penggunaan stack teknologi Laravel, Vue.js, dan PostgreSQL, serta penerapan standar pengujian internasional IEEE 29119 dalam satu sistem pengaduan lingkungan yang dikhususkan untuk Dinas Lingkungan Hidup Kota Madiun. Kombinasi seluruh aspek tersebut belum ditemukan pada penelitian-penelitian sebelumnya.

## **2. Keaslian Penelitian**

Keaslian penelitian ini ditunjukkan melalui *State of The Art* (SOTA) yang mempertegas kebaruan dan orisinalitas penelitian yang dilakukan. Dari uraian studi sebelumnya yang relevan, belum ditemukan penelitian yang secara khusus membangun sistem informasi pengaduan lingkungan berbasis website dengan fitur tracking status real-time untuk Dinas Lingkungan Hidup Kota Madiun menggunakan kombinasi framework Laravel, Vue.js, dan PostgreSQL dengan metode pengembangan Agile Scrum serta standar pengujian IEEE 29119.

Penelitian Suharsono (2022) membangun sistem pengaduan lingkungan berbasis website namun belum dilengkapi fitur tracking status real-time. (Ikhwan & Putri Lubis, 2023) mengembangkan sistem pengaduan masyarakat berbasis web untuk instansi pemerintah namun tidak berfokus pada bidang lingkungan hidup. (Setiawan et al., 2023) membangun sistem aduan lingkungan hidup untuk Kota Magelang namun tidak menerapkan metode Agile Scrum maupun standar pengujian internasional. (Halid & Baijuri, 2025) merancang sistem pengaduan berbasis web namun cakupannya mencakup layanan umum, bukan spesifik lingkungan hidup. (Liesnaningsih et al., 2021) membangun sistem monitoring pengaduan namun tidak dilengkapi fitur tracking status yang terintegrasi dengan notifikasi real-time kepada pelapor.

Kebaruan (*novelty*) penelitian ini terletak pada tiga aspek utama. Pertama, sistem yang dibangun secara spesifik dirancang untuk kebutuhan Dinas Lingkungan Hidup Kota Madiun dengan alur penanganan pengaduan yang disesuaikan dengan prosedur operasional instansi tersebut. Kedua, fitur tracking status mencakup tiga tahapan yang terstruktur yaitu menunggu verifikasi, sedang diproses, dan selesai yang memungkinkan pelapor memantau perkembangan laporannya secara real-time. Ketiga, penelitian ini menggabungkan metode pengembangan *Agile Scrum* dengan standar pengujian internasional IEEE 29119, sehingga menghasilkan sistem yang dikembangkan secara iteratif sekaligus diuji secara sistematis dan terstandar.

Kombinasi ketiga aspek kebaruan tersebut menjadikan penelitian ini orisinal dan berbeda dari penelitian-penelitian sebelumnya.

### 3. Gambaran Umum Studi Kasus dan Metode Penelitian

Penelitian ini dilaksanakan di Dinas Lingkungan Hidup (DLH) Kota Madiun, instansi pemerintah yang berwenang dalam pengelolaan dan pengendalian lingkungan hidup di wilayah Kota Madiun. Mekanisme pelaporan lingkungan yang masih konvensional melalui surat, telepon, dan kunjungan langsung menyebabkan pengelolaan data aduan menjadi kurang terorganisasi serta membatasi transparansi informasi bagi masyarakat. Untuk mengatasi keterbatasan tersebut, dibangun sistem informasi pengaduan lingkungan berbasis *website*. Melalui penambahan fitur *tracking status*, aplikasi ini menghubungkan secara terintegrasi dua aktor utama sistem, yaitu masyarakat Kota Madiun sebagai pelapor dan petugas DLH Kota Madiun selaku pengelola laporan.

Pengembangan sistem dilakukan menggunakan metode *Agile Scrum* yang bersifat iteratif dan adaptif, dengan pembagian pengerjaan ke dalam beberapa *sprint* berdasarkan prioritas fitur. Tahapan dalam setiap siklus *sprint* meliputi:

1. Product Backlog & Sprint Planning, yakni penyusunan daftar kebutuhan sistem (product backlog) berdasarkan hasil observasi dan wawancara di DLH Kota Madiun, kemudian diprioritaskan dan direncanakan dalam setiap sprint.

2. Sprint Execution, meliputi perancangan Use Case Diagram, ERD, antarmuka pengguna, serta implementasi kode program berbasis Laravel, Vue.js, dan PostgreSQL secara iteratif sesuai backlog yang dipilih untuk setiap sprint.
3. Sprint Review & Retrospective, yaitu evaluasi hasil pengerjaan setiap sprint bersama pemangku kepentingan (*stakeholder*) DLH Kota Madiun, serta perbaikan proses untuk sprint berikutnya.
4. Pengujian, dilakukan menggunakan standar IEEE 29119 yang mencakup perencanaan, desain kasus uji, eksekusi, dan pelaporan hasil pengujian terhadap seluruh fungsionalitas sistem dari perspektif pengguna.
5. Deployment & Pemeliharaan, mencakup penerapan sistem ke lingkungan produksi DLH Kota Madiun, perbaikan bug, optimasi performa, dan penyusunan dokumentasi sistem.

### **C. Kerangka Berpikir**

Permasalahan yang dihadapi Dinas Lingkungan Hidup Kota Madiun adalah belum tersedianya sistem pengaduan lingkungan berbasis digital. Mekanisme pengaduan yang masih konvensional menyebabkan pengelolaan laporan tidak terstruktur dan berpotensi terjadinya kehilangan data. Selain itu, masyarakat tidak dapat memantau perkembangan pengaduan sehingga transparansi pelayanan belum optimal.

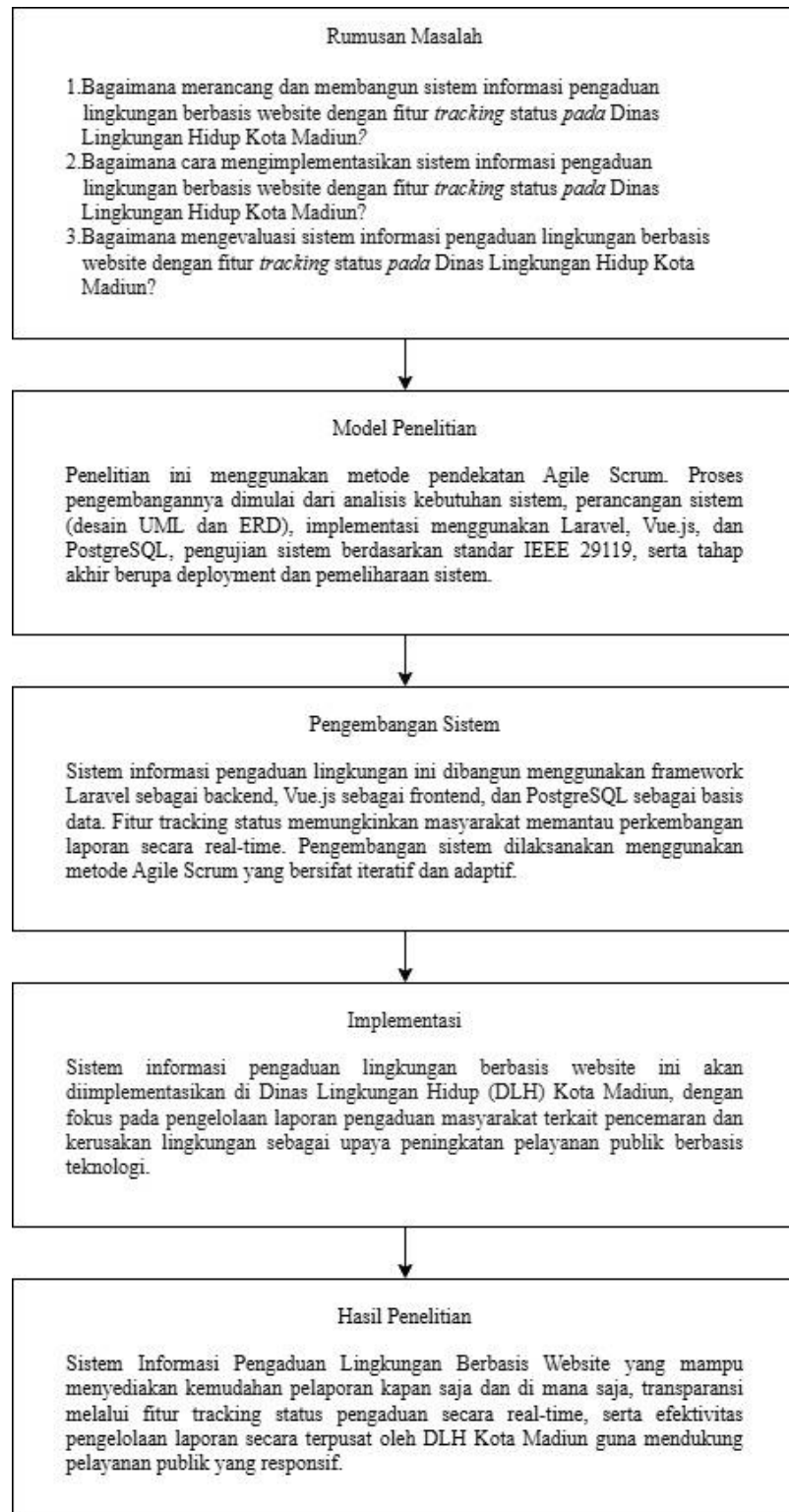
Berdasarkan kajian teori dan penelitian terdahulu, sistem informasi berbasis website terbukti mampu meningkatkan efisiensi dan kualitas pelayanan publik. Namun, sebagian besar sistem tersebut belum dilengkapi dengan fitur

*tracking status* secara real-time. Kondisi ini menunjukkan adanya kesenjangan yang perlu diselesaikan untuk meningkatkan transparansi layanan.

Solusi yang diajukan dalam penelitian ini adalah pembangunan Sistem Informasi Pengaduan Lingkungan Berbasis *Website* yang menyertakan fitur *tracking status*. Dalam proses pengerjaannya, metode *Agile Scrum* digunakan sebagai panduan pengembangan yang bersifat iteratif dan fleksibel terhadap kebutuhan sistem. Landasan teknologi yang diimplementasikan mencakup Laravel untuk arsitektur sistem, Vue.js untuk pengelolaan antarmuka, serta PostgreSQL sebagai basis data pendukungnya.

Tahapan penelitian meliputi analisis kebutuhan, perancangan sistem menggunakan UML, implementasi, serta pengujian. Pengujian dilakukan berdasarkan standar IEEE 29119 untuk memastikan kualitas sistem yang dikembangkan. Dengan tahapan tersebut, sistem diharapkan dapat berjalan sesuai kebutuhan pengguna.

Hasil penelitian yang diharapkan adalah sistem informasi pengaduan lingkungan berbasis website yang fungsional dan terintegrasi. Sistem ini mampu mempermudah masyarakat dalam menyampaikan pengaduan serta memantau status secara real-time. Selain itu, sistem diharapkan dapat meningkatkan efektivitas dan transparansi pengelolaan pengaduan di Dinas Lingkungan Hidup Kota Madiun.



Gambar 2. 2 Kerangka Berfikir