

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian ini didasarkan pada berbagai penelitian terdahulu yang membahas pengelolaan pengaduan masyarakat dan pengembangan sistem informasi pelayanan publik.

Penelitian yang dilakukan oleh Wumu dkk. (2023) dengan judul “Pengelolaan Layanan Pengaduan Masyarakat di Dinas Kependudukan dan Pencatatan Sipil Kota Manado” berfokus pada analisis mekanisme pengelolaan pengaduan masyarakat yang masih bersifat manual serta dampaknya terhadap proses penanganan dan dokumentasi pengaduan. Penelitian tersebut belum mengembangkan solusi berupa sistem informasi berbasis *website* untuk mendukung pengelolaan pengaduan secara digital. Hal tersebut menjadi salah satu perbedaan dengan penelitian ini yang mengembangkan sistem informasi pengaduan berbasis *web* untuk mendukung pengelolaan dan pemantauan pengaduan layanan administrasi kependudukan [10].

Penelitian yang dilakukan oleh Nimas Woro Pangesti (2022) dengan judul “Sistem Informasi Pengaduan Masyarakat Kelurahan Borangan Manisrenggo Klaten Jawa Tengah” membahas pengembangan sistem informasi pengaduan masyarakat untuk membantu mengelola pengaduan terkait kondisi sarana dan prasarana di Kelurahan Borangan. Sistem yang

dikembangkan memungkinkan masyarakat menyampaikan pengaduan secara *online* serta menyediakan fitur tanggapan dari pihak kelurahan. Pengembangan sistem menggunakan metode *waterfall* yang meliputi analisis kebutuhan, desain, pemrograman, pengujian, dan pemeliharaan. Penelitian tersebut berfokus pada pengaduan masyarakat terkait sarana dan prasarana kelurahan, sedangkan penelitian ini berfokus pada pengaduan layanan administrasi kependudukan dengan melibatkan koordinasi antara kelurahan dan Disdukcapil [11].

Penelitian yang dilakukan oleh Susanto dkk. (2024) dengan judul “Implementasi Metode RAD pada Sistem Pengaduan Masyarakat (SIPMAS) di Desa Logung Menggunakan Framework Laravel” membahas pengembangan sistem pengaduan masyarakat dengan menggunakan metode *Rapid Application Development* (RAD). Metode tersebut menekankan proses pengembangan sistem secara cepat dan iteratif melalui pembuatan *prototipe*. Penelitian ini memiliki perbedaan dengan penelitian yang dilakukan, yaitu pada metode pengembangan sistem yang digunakan. Penelitian ini menggunakan metode *waterfall* yang memiliki tahapan pengembangan secara berurutan dan terstruktur sehingga setiap tahap dapat dilakukan secara sistematis [12].

Berdasarkan hasil kajian terhadap penelitian-penelitian tersebut, dapat disimpulkan bahwa penelitian terdahulu telah membahas pengelolaan pengaduan masyarakat, baik melalui analisis proses pengaduan maupun pengembangan sistem informasi berbasis *web*. Namun, penelitian-penelitian

tersebut belum secara spesifik membahas pengembangan sistem pengaduan yang mengintegrasikan pihak kelurahan dengan Disdukcapil dalam pengelolaan pengaduan layanan administrasi kependudukan. Oleh karena itu, penelitian ini diarahkan untuk mendesain dan membangun Sistem Informasi E-Aduan Layanan Administrasi Kependudukan di Kelurahan Kota Tegal berbasis *web* dengan menggunakan *framework* Laravel. Sistem tersebut diharapkan dapat meningkatkan efektivitas dalam pengelolaan pengaduan terkait pelayanan administrasi kependudukan di Kota Tegal.

2.2 Landasan Teori

2.2.1 Sistem

Sistem merupakan sekumpulan komponen atau elemen yang saling berhubungan dan bekerja secara terpadu untuk mencapai tujuan tertentu. Setiap komponen memiliki fungsi masing-masing, tetapi tetap saling mendukung sehingga proses yang berlangsung dapat berjalan secara efektif. Keberadaan sistem membantu organisasi dalam mengelola aktivitas secara lebih terstruktur, meningkatkan efisiensi kerja, menjaga konsistensi proses, serta mempermudah pencapaian tujuan yang telah ditetapkan[13].

2.2.2 Informasi

Informasi adalah hasil pengolahan data yang telah diorganisasi sehingga memiliki makna dan dapat digunakan sebagai dasar dalam pengambilan keputusan. Informasi yang baik harus akurat, relevan,

mudah dipahami, dan tersedia pada waktu yang dibutuhkan agar mampu memberikan manfaat bagi penerimanya[14].

2.2.3 Sistem Informasi

Sistem informasi merupakan kombinasi antara manusia, perangkat keras (*hardware*), perangkat lunak (*software*), basis data (*database*), jaringan komunikasi, *data warehouse*, serta prosedur kerja yang saling terintegrasi untuk mengolah data menjadi informasi yang bermanfaat. Penerapan sistem informasi dibuat agar pengelolaan dan penyimpanan data menjadi lebih mudah, sehingga dapat menghasilkan informasi yang benar dan konsisten.[15].

2.2.4 Pelayanan Administratif

Pelayanan administratif merupakan salah satu bentuk pelayanan publik yang diselenggarakan oleh instansi pemerintah dalam rangka memenuhi kebutuhan administrasi masyarakat sesuai dengan ketentuan peraturan perundang-undangan. Pelayanan ini meliputi berbagai proses administrasi yang bertujuan memberikan kemudahan, kepastian, serta keteraturan dalam penyelenggaraan layanan kepada masyarakat [16].

2.2.5 Layanan Pengaduan

Layanan pengaduan adalah sarana yang disediakan oleh lembaga atau organisasi untuk mengumpulkan laporan, keluhan, dan masukan dari para pengguna layanan. Data yang diperoleh melalui layanan keluhan dimanfaatkan sebagai dasar untuk menilai dan

memperbaiki mutu pelayanan, meningkatkan pengaruh terhadap isu yang muncul, serta meningkatkan tanggung jawab penyelenggara layanan terhadap publik.

2.2.6 Website

Website merupakan media berbasis internet yang menyajikan berbagai informasi dalam bentuk halaman-halaman yang saling terhubung dan dapat diakses melalui *web browser*. Informasi yang disajikan dapat berupa teks, gambar, suara, video, maupun kombinasi dari berbagai elemen multimedia. Berdasarkan karakteristiknya, *website* dibedakan menjadi tiga jenis, yaitu *website* statis, *website* dinamis, dan *website* interaktif. Setiap jenis memiliki karakteristik serta tingkat interaksi yang berbeda sehingga penggunaannya dapat disesuaikan dengan kebutuhan pengembang maupun pengguna [17].

2.2.7 Visual Studio Code

Visual Studio Code merupakan sebuah alat pengedit kode yang dibuat oleh Microsoft. Perangkat lunak gratis dan *open-source* yang dirancang untuk membantu para pengembang dalam menulis dan mengedit kode dengan berbagai bahasa pemrograman. Selain melakukan pengeditan dan menemukan kesalahan dalam kode, Visual Studio menawarkan *kompilator*, alat untuk menyelesaikan kode, manajemen sumber, *ekstensi*, dan berbagai fitur lainnya untuk meningkatkan setiap langkah dalam proses pengembangan perangkat lunak. Visual Code dapat dipakai untuk membuat serta mengedit

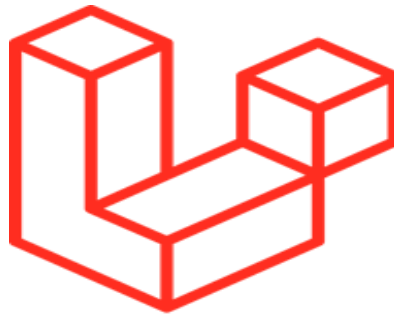
source code dari berbagai bahasa pemrograman Contohnya, seperti Html, CSS, Dan JavaScript[18].



Gambar 2. 1 Logo Visual Studio Code [19]

2.2.8 Framework Laravel

Framework atau kerangka kerja menjadi wadah bagi pengembangan sebuah *website*. Dengan menggunakan *framework*, proses pembuatan *website* menjadi lebih efisien karena waktu pengerjaan lebih singkat, dan perbaikan dapat dilakukan dengan lebih mudah [20]. Laravel juga merupakan salah satu *framework* pemrograman berbasis *open source* yang populer di kalangan *developer* di seluruh dunia. *Framework* ini dibuat untuk mendukung para pengembang dalam memanfaatkan PHP untuk membangun situs *web*. Laravel pertama kali dirilis oleh Taylor Otwell pada tahun 2011 sebagai alternatif untuk *framework CodeIgniter*, dengan fokus pada fitur yang lebih modern seperti autentikasi dan *routing* [21].



Gambar 2. 2 Logo Laravel [22]

2.2.9 PHP

Hypertext Preprocessor (PHP) merupakan salah satu teknologi utama dalam pengembangan aplikasi *web* yang banyak digunakan untuk membangun sistem berbasis server. Bahasa pemrograman ini memiliki karakteristik fleksibel, mudah dipelajari, dan mampu berintegrasi dengan berbagai basis data sehingga mendukung pembuatan aplikasi yang dinamis. Dengan hadirnya kerangka kerja modern seperti *Laravel*, efisiensi pengembangan aplikasi berbasis PHP meningkat secara signifikan, bahkan mencapai sekitar 70% karena struktur kode yang lebih rapi, fitur bawaan yang lengkap, serta pendekatan arsitektur yang memudahkan proses pemeliharaan dan pengembangan sistem.[21]



Gambar 2. 3 Logo PHP [19]

2.2.10 MySQL

Aplikasi pengelola basis data merupakan perangkat lunak yang digunakan untuk menyimpan, mengelola, dan memproses data secara terstruktur. *My Structured Query Language* (MySQL) adalah salah satu aplikasi *Database Management System* (DBMS) yang sering digunakan untuk mengelola basis data dengan kemampuan keamanan yang cukup andal. MySQL juga dapat diintegrasikan dengan berbagai bahasa pemrograman, seperti PHP. Selain itu, MySQL didukung oleh komunitas pengguna yang luas sehingga memudahkan pengguna dalam memperoleh dokumentasi, pembaruan sistem, serta solusi terhadap berbagai permasalahan yang dihadapi. Sebagai perangkat lunak yang bersifat *open source*, MySQL terus dikembangkan sehingga banyak digunakan dalam pengembangan aplikasi berbasis *web*.



Gambar 2. 4 Logo MySQL [22]

2.2.11 CSS

Penggunaan *Cascading Style Sheets* (CSS) murni *custom* CSS memungkinkan perancang mengatur gaya visual *website* secara bebas tanpa bergantung pada *framework*. *Developer* dapat menyesuaikan

tata letak, *tipografi*, warna, dan *responsivitas* sesuai kebutuhan proyek, menghasilkan kode yang lebih ringkas, terkontrol, dan bebas dari kelas tidak perlu. Pendekatan ini juga mempermudah penyesuaian antar elemen serta meningkatkan performa halaman karena hanya memuat kode yang diperlukan. CSS *custom* sangat cocok untuk proyek yang menuntut tampilan unik dan optimal serta memberikan fleksibilitas maksimal dalam pengelolaan gaya tampilan *web*. [23]



Gambar 2. 5 Logo CSS [24]

2.2.12 Bootstrap

Bootstrap merupakan *framework* berbasis CSS yang bersifat *open source* dan banyak digunakan dalam pengembangan antarmuka *website*. *Framework* ini menyediakan berbagai komponen siap pakai seperti tombol, tabel, *form*, navigasi, *card*, serta sistem *grid* yang membantu pengembang dalam menyusun tampilan *website* secara lebih cepat dan terstruktur. Dengan hadirnya Bootstrap, proses desain antarmuka menjadi lebih efektif karena para pengembang tidak perlu membangun semua elemen CSS dari nol. Lebih dari itu, Bootstrap juga mengadopsi prinsip desain *web* responsif sehingga tampilan situs

web dapat disesuaikan dengan ukuran layar di perangkat desktop maupun mobile.

Bootstrap diciptakan pertama kali oleh Mark Otto dan Jacob Thornton pada tahun 2011 di dalam lingkungan Twitter. Di samping mempercepat tahapan pengembangan antarmuka, penggunaan Bootstrap juga memudahkan penyesuaian tampilan situs *web* pada berbagai ukuran perangkat agar sistem dapat diakses dengan lebih nyaman [25].

2.2.13 Laragon

Laragon adalah perangkat lunak yang bersifat *open source* (terbuka) yang dapat mendukung berbagai sistem operasi, di mana Laragon bertugas sebagai *server virtual* atau yang sering disebut sebagai localhost. Aplikasi ini sangat efektif untuk manajemen aplikasi yang berbasis *web*. Laragon adalah software yang serupa dengan Cross-Platform, Apache, MySQL/MariaDB, PHP, dan Perl (XAMPP), tetapi dikembangkan khusus untuk memenuhi kebutuhan pengembang PHP yang memanfaatkan *framework* Laravel. *Service* yang *include* dalam *Laragon* antara lain Apache, MySQL, PHP Server, Memcached, Redis, Composer, Xdebug, PhpMyAdmin, Cmder, dan layanan lainnya. Aplikasi ini sangat cocok digunakan oleh *developer* PHP yang menggunakan *framework* Laravel, karena dapat mempermudah proses pengembangan aplikasi [21].



Gambar 2. 6 Logo Laragon [26]

2.2.14 Unified Modeling Language UML

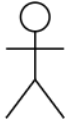
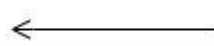
Unified Modeling Language (UML) adalah suatu bahasa visual yang digunakan untuk merepresentasikan, menggambarkan, memberi spesifikasi, dan mendokumentasikan sistem perangkat lunak yang berbasis objek dengan cara yang terorganisir. *UML* menawarkan kumpulan diagram standar seperti diagram *use case*, diagram kelas, diagram urutan, dan diagram aktivitas yang berfungsi untuk menjelaskan alur proses serta keterkaitan antara berbagai komponen dalam sistem. Menurut jurnal *Tools Rekayasa Perangkat Lunak dalam Membuat Pemodelan Desain Menggunakan Unified Modeling Language (UML)*, penggunaan UML mempermudah pengembang dalam menganalisis kebutuhan, merancang arsitektur sistem, serta memastikan desain sistem terdokumentasi dengan baik sehingga mengurangi risiko kesalahan pada tahap pengembangan [27].

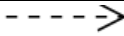
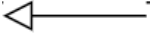
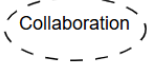
Terdapat beberapa diagram UML yang sering digunakan dalam pengembangan sebuah sistem yaitu:

1. Use Case Diagram

Use Case Diagram adalah representasi dari kemampuan yang diharapkan dari sistem yang menunjukkan hubungan antara aktor dan sistem. Di dalam use case terdapat aktor yang menggambarkan entitas berupa individu atau sistem yang menjalankan tugas dalam sistem tersebut.

Tabel 2. 1 Use Case Diagram

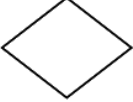
No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menentukan Sekumpulan peran yang dijalankan oleh pengguna saat berinteraksi dengan <i>use case</i> .
2.		<i>Dependency</i>	Sebuah hubungan di mana perubahan yang dialami oleh elemen mandiri (<i>independent</i>) akan berdampak pada elemen yang tergantung padanya, yaitu elemen yang tidak mandiri (<i>dependent</i>).
3.		<i>Generalization</i>	Sebuah hubungan di mana objek turunan (<i>descendent</i>) memiliki berbagai tingkah laku dan struktur data yang berasal dari objek di atasnya, yaitu objek induk (<i>ancestor</i>).
4.		<i>Include</i>	Mensyaratkan bahwa <i>use case</i>

No	Gambar	Nama	Keterangan
			sumber dijelaskan dengan jelas.
5.		<i>Extend</i>	Merepresentasikan bahwa <i>use case</i> target meningkatkan perilaku dari <i>use case</i> sumber pada saat tertentu.
6.		<i>Association</i>	Yang mengaitkan satu objek dengan objek lainnya.
7.		<i>System</i>	Menerangkan paket yang menunjukkan sistem dengan batasan tertentu.
8.		<i>Use Case</i>	Menggambarkan serangkaian tindakan yang menghasilkan keluaran yang berarti bagi aktor.
9.		<i>Collaboration</i>	Turnya interaksi antara elemen-elemen atau aturan untuk mencapai fungsi tertentu.
10.		<i>Note</i>	Catatan tambahan yang menjelaskan bagian tertentu dari diagram.

2. Activity Diagram

Sebuah diagram aktivitas UML yang menunjukkan pergerakan dinamis dari sebuah sistem atau *segmen* dari sistem dengan menampilkan aliran pengendalian di antara tindakan yang dijalankan oleh sistem.

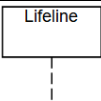
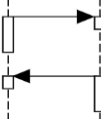
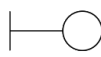
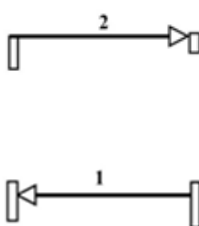
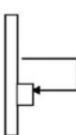
Tabel 2. 2 Activity Diagram

No	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Sekelompok action yang membentuk satu proses lengkap. Dapat terdiri dari beberapa action di dalamnya.
2.		<i>Action</i>	Langkah tunggal / aksi sederhana yang dilakukan sistem. Simbol: persegi panjang dengan sudut tumpul.
3.		<i>Initial Node</i>	Titik awal aktivitas; digambarkan sebagai lingkaran hitam penuh.
4.		<i>Final Node</i>	Titik akhir aktivitas; digambarkan sebagai lingkaran dengan lingkaran lain di dalamnya.
5.		<i>Fork Node</i>	Memecah satu alur menjadi beberapa alur paralel; simbol garis tebal.
6.		<i>Decision</i>	Titik percabangan berdasarkan kondisi tertentu; simbol belah ketupat.
7.		<i>Rake</i>	Menunjukkan adanya dekomposisi
8.		<i>Send</i>	Tanda pengiriman

3. Sequence Diagram

Menggambarkan interaksi antar objek di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* umumnya digunakan untuk memvisualisasikan situasi atau tahapan yang diambil sebagai reaksi terhadap suatu peristiwa untuk menghasilkan hasil.

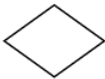
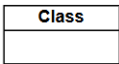
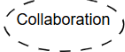
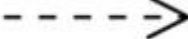
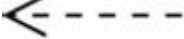
Tabel 2. 3 Sequence Diagram

No	Gambar	Nama	Kesimpulan
1.		<i>Lifeline</i>	Objek <i>entity</i> , antara muka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktivitas yang terjadi.
3.		<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan system.
4.		<i>Boundary Class</i>	Menggambarkan penggambaran dari form.
5.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.
6.		<i>Control Class</i>	Menggambarkan penghubung antara Boundary dengan label.
7.		Activation	Sebagai sebuah objek yang akan melakukan sebuah aksi.
8.		Message	Mengindikasikan komunikasi antar objek dengan objek.
9.		Self Message	Mengindikasikan komunikasi kembali kedalam sebuah objek itu sendiri.

4. Class Diagram

Class Diagram adalah representasi dari struktur dan penjelasan mengenai *class*, *package*, serta objek yang memiliki hubungan satu sama lain, termasuk di dalamnya pewarisan, asosiasi, dan berbagai bentuk lainnya.

Tabel 2. 4 Class Diagram

No	Gambar	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan di mana objek anak (<i>descendant</i>) mewarisi perilaku dan struktur dari objek induk (<i>ancestor</i>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagai atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi yang ditampilkan dalam sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Dependency</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan Dimana perubahan yang terjadi pada suatu

No	Gambar	Nama	Keterangan
			elemen (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.