

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Menurut Aries Nur Arifin dan Eni Heni Hermaliani (2024), penerapan sistem informasi berbasis web dalam pelayanan administrasi kependudukan mampu meningkatkan efektivitas dan efisiensi pelayanan melalui digitalisasi data serta integrasi pengelolaan dokumen secara terpusat. Sistem informasi yang terkomputerisasi memungkinkan proses verifikasi data menjadi lebih cepat, akurat, dan terdokumentasi dengan baik sehingga dapat meminimalkan kesalahan pencatatan. Penelitian ini menggunakan pendekatan perancangan sistem informasi berbasis web dengan pengujian sistem untuk memastikan sistem dapat berjalan sesuai kebutuhan pengguna. Namun, penelitian ini masih berfokus pada pengelolaan layanan administrasi secara umum dan belum mengintegrasikan seluruh alur pelayanan dalam satu sistem terpadu yang mendukung layanan jemput bola secara langsung, sehingga masih terdapat peluang pengembangan sistem pelayanan yang lebih komprehensif [13].

Menurut Ulfiya Shalika Sakinah dan Saipul (2024), program layanan jemput bola dalam administrasi kependudukan terbukti efektif dalam meningkatkan kepemilikan dokumen kependudukan masyarakat, khususnya pada wilayah dengan keterbatasan akses terhadap kantor pelayanan. Penelitian ini menggunakan metode analisis efektivitas layanan melalui pendekatan observasi dan survei terhadap masyarakat penerima layanan.

Hasil penelitian menunjukkan bahwa layanan jemput bola mampu memperluas jangkauan pelayanan publik serta meningkatkan kemudahan masyarakat dalam memperoleh dokumen kependudukan. Namun, penelitian ini masih berfokus pada implementasi layanan secara operasional dan belum memanfaatkan sistem informasi digital terintegrasi untuk mengelola proses layanan secara otomatis, sehingga masih diperlukan pengembangan sistem berbasis web untuk mendukung layanan jemput bola secara lebih optimal [14].

Menurut Riansyah dkk. (2025), pengembangan sistem informasi pelayanan publik pada pemerintah daerah dapat membantu meningkatkan efisiensi, transparansi, dan kualitas pelayanan kepada masyarakat. Penelitian ini menghasilkan sistem informasi pelayanan publik yang dilengkapi dengan pengelolaan data terintegrasi, pelacakan permohonan secara *real-time*, serta notifikasi otomatis untuk mempermudah proses pelayanan. Hasil penelitian menunjukkan bahwa sistem informasi yang dikembangkan mampu mendukung pelayanan publik agar lebih terstruktur, cepat, dan mudah diakses oleh masyarakat. Namun, penelitian tersebut masih berfokus pada sistem pelayanan publik pemerintah daerah secara umum dan belum secara khusus membahas layanan jemput bola administrasi kependudukan yang mencakup pengajuan layanan, penjadwalan kegiatan lapangan, pencatatan hasil pelayanan, *monitoring*, dan pelaporan dalam satu *platform* terpadu [15].

Menurut Anto, Mappasere, Abdi, dan Alyas (2025), penerapan *e-government* memiliki peran penting dalam meningkatkan efisiensi

administrasi dan transparansi pemerintah dalam pelayanan publik. Pemanfaatan teknologi informasi dalam layanan pemerintah dapat mempermudah akses masyarakat, mempercepat proses administrasi, serta meningkatkan kualitas pelayanan yang diberikan. Hasil penelitian menunjukkan bahwa *e-government* berpotensi mendukung tata kelola pemerintahan yang lebih efektif, transparan, dan akuntabel. Namun, penelitian tersebut masih membahas penerapan *e-government* secara umum dan belum secara khusus mengarah pada pengembangan sistem informasi layanan jemput bola administrasi kependudukan berbasis web [16].

Berdasarkan penelitian-penelitian tersebut, dapat disimpulkan bahwa pemanfaatan teknologi informasi dan konsep layanan jemput bola mampu meningkatkan kualitas pelayanan publik. Namun, sebagian besar penelitian masih berfokus pada digitalisasi layanan secara umum dan belum mengembangkan sistem informasi terintegrasi yang mendukung seluruh proses layanan jemput bola, mulai dari pengajuan layanan, penjadwalan kegiatan lapangan, pencatatan hasil pelayanan, hingga *monitoring* dan pelaporan dalam satu *platform* berbasis *web*. Oleh karena itu, penelitian ini mengembangkan rancang bangun Sistem Informasi JEBOL (Jemput Bola *Online*) berbasis *web* untuk mendukung transformasi digital pelayanan administrasi kependudukan di Dinas Kependudukan dan Pencatatan Sipil Kota Tegal.

2.2. Landasan Teori

2.2.1. Sistem

Sistem merupakan jaringan proses kerja yang terhubung satu sama lain dan berkumpul untuk mencapai sebuah tujuan serta melakukan suatu kegiatan tertentu yang sebagian sistem terdiri dari komponen yang lebih kecil yang berfungsi untuk mendukung yang lebih besar [17]. Sistem berperan penting untuk dikembangkan karena memiliki beberapa fungsi penting yaitu meningkatkan efisiensi, meningkatkan kualitas, meningkatkan keamanan, meningkatkan kemampuan serta meningkatkan keterkaitan.

2.2.2. Sistem Informasi

Sistem Informasi merupakan sistem yang dimanfaatkan banyak orang dalam mengambil keputusan karena menyediakan berbagai informasi. Selain itu dapat mengurangi kesalahan yang terjadi sehingga dapat meningkatkan kinerja agar lebih efisien dan kecepatan dalam operasional. Sistem informasi dirancang untuk mempermudah pengolahan dan penyimpanan data untuk menghasilkan data yang tepat dan akurat [18]. Komponen – komponen dalam sistem informasi antara lain *hardware*, *software*, *telekomunikasi*, *database* dan *data warehouse*, serta sumber daya manusia.

2.2.3. E-Government

E-Government merupakan pemanfaatan teknologi informasi dan komunikasi oleh instansi pemerintah untuk meningkatkan kualitas

pelayanan publik kepada masyarakat. Konsep e-government bertujuan menciptakan pelayanan yang lebih transparan, akuntabel, cepat, dan efisien melalui penggunaan sistem digital dalam proses administrasi pemerintahan. Penerapan *e-government* memungkinkan pelayanan publik dapat diakses secara elektronik sehingga tidak terbatas oleh waktu dan tempat. Selain itu, pemanfaatan e-government juga mampu mempercepat proses pelayanan, meningkatkan keterbukaan informasi, serta mendukung efektivitas penyelenggaraan pelayanan administrasi public [19].

2.2.4. Pelayanan Administrasi Kependudukan

Pelayanan publik merupakan kegiatan yang dilakukan oleh instansi pemerintah untuk memenuhi kebutuhan masyarakat sesuai dengan ketentuan peraturan perundang-undangan. Pelayanan publik harus dilaksanakan secara profesional, transparan, serta berorientasi pada kepuasan masyarakat. Administrasi kependudukan di Indonesia diatur dalam Undang-Undang Nomor 24 Tahun 2013 tentang Administrasi Kependudukan yang mengatur penyelenggaraan pencatatan dan penerbitan dokumen kependudukan seperti KTP, Kartu Keluarga, akta kelahiran, dan dokumen lainnya. Undang-undang tersebut menekankan bahwa pelayanan administrasi kependudukan harus dilakukan secara tertib, cepat, tepat, dan akurat sehingga dapat meningkatkan kualitas pelayanan kepada masyarakat [20].

2.2.5. Website

Website merupakan kumpulan halaman informasi yang saling terhubung dan dapat diakses melalui jaringan internet menggunakan *browser* seperti *Google Chrome*, *Mozilla Firefox*, dan sejenisnya. *Website* digunakan sebagai media penyampaian informasi dalam bentuk teks, gambar, audio, maupun video yang dapat diakses secara global tanpa batasan geografis. *Website* mempermudah penyebaran informasi serta mendukung penyediaan layanan secara daring sehingga meningkatkan kemudahan akses bagi pengguna [21]. Dalam pengembangan sistem informasi berbasis web, *website* berperan sebagai media utama dalam penyampaian layanan, pengelolaan data, serta interaksi antara pengguna dengan sistem secara lebih efisien dan terstruktur.

2.2.6. Visual Studio Code

Visual Studio Code merupakan editor kode sumber yang dikembangkan oleh Microsoft. Perangkat lunak gratis dan bersifat *open source* yang dirancang untuk membantu pengembang dalam menulis dan mengedit kode dengan berbagai bahasa pemrograman. Selain pengeditan dan penelusuran kesalahan kode, *Visual Studio Code* menyediakan kompilator, alat penyelesaian kode, kontrol sumber, ekstensi, serta berbagai fitur untuk mendukung setiap tahap proses pengembangan perangkat lunak. *Visual Studio Code* dapat

digunakan untuk membuat dan mengedit *source code* dari berbagai bahasa pemrograman seperti *HTML*, *CSS*, dan *JavaScript* [22].



Gambar 2. 1 Logo Visual Studio Code [23]

2.2.7. 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 [24]. *Laravel* juga merupakan salah satu *framework* pemrograman berbasis *open source* yang populer di kalangan *developer* di seluruh dunia. *Framework* ini dirancang untuk membantu pengembang dalam penggunaan *PHP* untuk pengembangan *website*. *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* [25].



Gambar 2. 2 Logo Laravel [26]

2.2.8. PHP

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 [27].



Gambar 2. 3 Logo PHP [28]

2.2.9. MySQL

MySQL merupakan salah satu aplikasi *Database Management System* (DBMS) yang sering digunakan untuk mengelola basis data dengan tingkat keamanan yang cukup handal. *MySQL* mampu terhubung dengan berbagai aplikasi seperti *PHP* untuk mendukung pengembangan aplikasi berbasis web. *MySQL* memiliki komunitas

pengguna yang luas sehingga berbagai kendala penggunaan dapat dicari solusinya melalui forum diskusi khusus *MySQL*. Perangkat lunak ini terus diperbarui, bersifat gratis, serta mempermudah proses instalasi dan pengelolaan basis data dalam pengembangan sistem informasi [29].



Gambar 2. 4 MySQL [30]

2.2.10. CSS

Penggunaan *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 [31].



Gambar 2. 5 Logo CSS [32]

2.2.11. Laragon

Laragon merupakan perangkat lunak yang bersifat *open source* dan mendukung berbagai sistem operasi. *Laragon* berfungsi sebagai *server* virtual atau *localhost* yang digunakan untuk pengelolaan aplikasi berbasis *website*. *Laragon* memiliki konsep yang mirip dengan *XAMPP*, namun lebih difokuskan pada kebutuhan pengembang *PHP* yang menggunakan *framework Laravel*. Layanan yang tersedia dalam *Laragon* meliputi *Apache*, *MySQL*, *PHP Server*, *Memcached*, *Redis*, *Composer*, *Xdebug*, *PhpMyAdmin*, *Cmder*, serta berbagai layanan pendukung lainnya. *Laragon* cocok digunakan oleh pengembang *PHP* dengan *framework Laravel* karena dapat mempermudah proses pengembangan aplikasi [33].



Gambar 2. 6 Logo Laragon [34]

2.2.12. JavaScript

JavaScript merupakan bahasa pemrograman yang digunakan dalam pengembangan *website* untuk membuat halaman menjadi lebih interaktif dan dinamis. Dalam sistem informasi berbasis web, *JavaScript* berfungsi untuk mengatur mekanisme interaktif seperti validasi *form*, pemrosesan input, aksi tombol, serta komunikasi data antara pengguna dan sistem. *JavaScript* juga dapat meningkatkan kenyamanan pengguna karena proses interaksi pada halaman *website* menjadi lebih cepat, responsif, dan mudah digunakan. Penggunaan *JavaScript* bersama *CSS* dalam pengembangan sistem informasi berbasis web dapat mendukung tampilan yang responsif serta meningkatkan kualitas layanan digital [35].



Gambar 2. 7 Logo JavaScript [36]

2.2.13. Unified Modeling Language

Unified Modeling Language (UML) merupakan bahasa pemodelan visual yang digunakan untuk memodelkan, memvisualisasikan, menspesifikasikan, serta mendokumentasikan

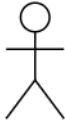
sistem perangkat lunak berorientasi objek secara terstruktur. *UML* menyediakan seperangkat diagram standar seperti *Use Case Diagram*, *class diagram*, *sequence diagram*, dan *Activity Diagram* yang membantu menggambarkan alur proses dan hubungan antar komponen dalam sistem. Menurut jurnal *Tools Rekayasa 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 [37].

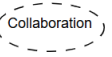
Terdapat beberapa diagram UML yang sering digunakan dalam pengembangan sebuah sistem yaitu:

1. Use Case Diagram

Use Case Diagram merupakan gambaran dari fungsionalitas yang diharapkan dari sistem yang merepresentasikan sebuah interaksi antara aktor dan sistem. Di dalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem. Rincian *Use Case Diagram* ini disajikan pada Tabel 2. 1 berikut.

Tabel 2. 1 Use Case Diagram

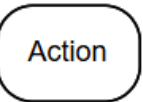
No	Gambar	Nama	Keterangan
1.		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2.		Dependency	Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
3.		Generalization	Hubungan di mana objek anak (<i>descendant</i>) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk(ancestor).
4.		Include	Menspesifikasikan bahwa use case sumber secara eksplisit.
5.		Extend	Merepresentasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
6.		Association	Apa yang menghubungkan antar objek satu dengan objek lainnya.
7.		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

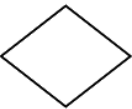
No	Gambar	Nama	Keterangan
8.		Use Case	Mendesripsikan urutan aksi yang memberikan hasil bernilai bagi aktor.
9.		Collaboration	Interaksi antar elemen atau aturan untuk mencapai suatu fungsi tertentu.
10.		Note	Catatan tambahan untuk menjelaskan bagian tertentu dari diagram.

2. Activity Diagram

Sebuah diagram aktivitas UML yang menggambarkan perilaku dinamis dari suatu sistem atau bagian dari sistem melalui aliran kontrol antara aksi yang dilakukan sistem. Rincian *Activity Diagram* ini disajikan pada Tabel 2. 2 berikut.

Tabel 2. 2 Activity Diagram

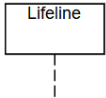
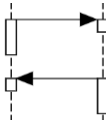
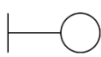
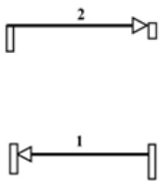
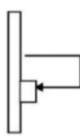
No	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Sekelompok action yang membentuk satu proses lengkap. Dapat terdiri dari beberapa action di dalamnya.
2.		<i>Action</i>	Langkah tunggal / aksi sederhana yang dilakukan sistem. Simbol: persegi panjang dengan sudut tumpul.
3.		<i>Initial Node</i>	Titik awal aktivitas; digambarkan sebagai lingkaran hitam penuh.

No	Gambar	Nama	Keterangan
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* biasa digunakan untuk menggambarkan skenario atau langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan *output*. Rincian *Sequence Diagram* ini disajikan pada Tabel 2.3 berikut.

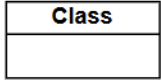
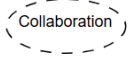
Tabel 2. 3 Sequence Diagram

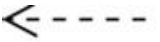
No	Gambar	Nama	Kesimpulan
1.		<i>Lifeline</i>	Objek entity, antara muka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktivitas yang terjadi.
3.		<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan system.
4.		<i>Boundary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

4. Class Diagram

Class Diagram merupakan gambaran struktur dan deskripsi dari *class*, *package*, dan objek yang saling keterkaitan seperti diantaranya pewarisan, asosiasi, dan lainnya. Rincian *Class Diagram* ini disajikan pada Tabel 2.4 berikut.

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.

No	Gambar	Nama	Keterangan
5.		<i>Dependency</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan di mana perubahan yang terjadi pada suatu 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.