

RANCANG BANGUN APLIKASI VILLAGE SERVICE BERBASIS WEB MENGGUNAKAN LARAVEL FILAMENT DI DESA CANTIGI KULON

Tohadi^{1*}, Aji Primajaya², Carudin³

^{1,2,3}Universitas Singaperbangsa Karawang; Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur Karawang, Indonesia; (0267) 641177

Keywords:

Filament;
Laravel;
Prototype;
Village Service.

Correspondent Email:

toadiasley243@gmail.com

Abstrak. Pelayanan administrasi desa yang masih bersifat konvensional mendorong perlunya digitalisasi layanan publik, sejalan dengan tren global transformasi digital dalam sektor pemerintahan. Penelitian ini bertujuan untuk merancang dan membangun aplikasi *Village Service* berbasis web menggunakan Laravel Filament, guna mempermudah pengelolaan surat, pengaduan warga, dan penyampaian informasi. Metode yang digunakan adalah *prototype*, yang memungkinkan pengembangan sistem secara bertahap berdasarkan masukan pengguna. Pengujian dilakukan melalui *black-box testing* untuk validasi fungsionalitas, serta *User Acceptance Test* (UAT) untuk menilai kepuasan pengguna. Hasil menunjukkan bahwa aplikasi berhasil dibangun sesuai kebutuhan pelayanan di Desa Cantigi Kulon, dengan nilai UAT dari pengguna umum sebesar 91,82%. Dengan demikian, penggunaan Laravel Filament dan metode *prototype* terbukti efektif dalam mendukung sistem pelayanan desa yang fungsional dan efisien, serta mendukung agenda digitalisasi pelayanan publik secara berkelanjutan.



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract. Conventional administrative services in villages highlight the need for digitalization of public services, in line with the global trend of digital transformation in the government sector. This study aims to design and develop a web-based Village Service application using the Laravel Filament framework to facilitate the management of letters, citizen complaints, and information dissemination. The research employs a prototype method, allowing for iterative system development based on user feedback. Testing was carried out through black-box testing to validate system functionality and User Acceptance Testing (UAT) to assess user satisfaction. The results show that the application was successfully developed according to the service needs of Cantigi Kulon Village, with a UAT score of 91.82% from general users. Thus, the use of Laravel Filament and the prototype method has proven effective in supporting a functional and efficient village service system, while contributing to the ongoing digitalization of public services.

1. PENDAHULUAN

Dalam era digital yang terus berkembang, pemanfaatan teknologi informasi pada layanan publik telah menjadi kebutuhan utama untuk

meningkatkan efisiensi, transparansi, dan aksesibilitas bagi masyarakat. Pemerintah, termasuk di tingkat kelurahan dan desa, didorong untuk menghadirkan pelayanan yang

lebih praktis dan mudah dijangkau. Salah satu bentuk penerapan teknologi dalam sektor pemerintahan adalah konsep *e-government*, yaitu penggunaan teknologi informasi dan komunikasi (TIK) dalam menjalankan fungsi pemerintahan secara digital. Melalui *e-government*, layanan dapat diakses masyarakat kapan saja dan dari mana saja, sekaligus mendorong partisipasi publik dan transparansi. [1]

Pemerintah Indonesia turut mendukung implementasi *e-government* melalui Peraturan Presiden Nomor 95 Tahun 2018 tentang Sistem Pemerintahan Berbasis Elektronik (SPBE), yang bertujuan menyederhanakan birokrasi, mempersingkat waktu pelayanan, dan meningkatkan kualitas pelayanan publik. Penerapan sistem digital ini juga diharapkan menjangkau hingga ke tingkat desa sebagai ujung tombak pemerintahan yang langsung bersentuhan dengan masyarakat.

Desa Cantigi Kulon, yang terletak di Kecamatan Cantigi, Kabupaten Indramayu, merupakan contoh desa yang masih menerapkan sistem pelayanan administratif secara konvensional. Berdasarkan hasil observasi dan wawancara dengan perangkat desa, ditemukan bahwa masyarakat masih harus datang langsung ke kantor desa untuk mengurus keperluan seperti pembuatan surat atau pengaduan. Meskipun penyampaian informasi mulai dilakukan melalui media sosial, namun proses pelayanan secara umum masih manual dan belum terdigitalisasi sepenuhnya. Hal ini menyebabkan proses pelayanan menjadi lambat dan tidak efisien.

Beberapa penelitian sebelumnya telah menunjukkan bahwa digitalisasi pelayanan desa dapat meningkatkan efisiensi dan kualitas layanan. [2] menggunakan metode *Extreme Programming* dalam mengembangkan sistem layanan terpadu di Desa Rejosari, dan menghasilkan tingkat kelayakan sistem sebesar 96,42%. Penelitian oleh [3] mengembangkan sistem informasi administrasi desa berbasis web menggunakan metode RAD, dan memperoleh hasil UAT yang memuaskan. [4] juga membangun website desa untuk promosi potensi desa menggunakan metode *prototype*, yang menunjukkan efektivitas pendekatan iteratif dalam pengembangan sistem.

Namun, meskipun telah banyak dilakukan penelitian serupa, sebagian besar masih

berfokus pada aspek layanan terpadu umum atau promosi potensi desa, bukan secara spesifik pada sistem pengelolaan pelayanan administratif seperti pengajuan surat, pengaduan warga, dan penyampaian informasi secara terpusat dan terpadu. Selain itu, belum banyak yang menggunakan *framework modern* seperti Laravel Filament, yang menawarkan antarmuka admin siap pakai, pengembangan cepat, dan efisiensi tinggi dalam pengelolaan data administratif berbasis web.

Berdasarkan hal tersebut, penelitian ini memiliki gap berupa kebutuhan akan sistem pelayanan administrasi desa yang spesifik, terintegrasi, dan mudah digunakan, yang dibangun menggunakan *framework modern* dan metode iteratif. Kebaruan (*novelty*) dari penelitian ini terletak pada penerapan Laravel Filament dalam membangun aplikasi *Village Service* yang disesuaikan dengan kebutuhan nyata desa, serta pendekatan *prototype* yang mengakomodasi masukan pengguna dalam setiap tahap pengembangan.

Tujuan dari penelitian ini adalah untuk merancang dan membangun aplikasi *Village Service* berbasis web menggunakan *framework* Laravel Filament yang dapat digunakan untuk mempermudah pengelolaan surat, pengaduan warga, dan penyampaian informasi di Desa Cantigi Kulon. Penelitian ini juga bertujuan untuk menguji kelayakan sistem melalui pendekatan *black-box testing* dan *User Acceptance Test (UAT)* guna memastikan bahwa sistem dapat diterima dan digunakan secara efektif oleh pengguna desa.

2. TINJAUAN PUSTAKA

2.1. Desa

Desa, menurut Kamus Besar Bahasa Indonesia (KBBI), adalah suatu wilayah yang dihuni oleh banyak keluarga dan dipimpin oleh seorang kepala desa. Desa juga merujuk pada kelompok rumah di luar kota di mana sebagian besar penduduk tinggal sebagai petani. Dalam hal ini, desa juga dapat didefinisikan sebagai kampung atau dusun yang berfungsi sebagai pusat kehidupan sosial dan budaya komunitas pedesaan.

2.2. Pelayanan Publik

Pemerintah membangun sistem pelayanan publik untuk memenuhi kebutuhan

dan kepentingan rakyat. Pemerintah diharapkan dapat memberikan pelayanan publik yang profesional, akuntabel, dan optimal untuk memenuhi hak rakyat. Pelayanan publik yang baik adalah pelayanan yang membuat masyarakat merasa puas dengan apa yang mereka dapatkan. [5]

2.3. Rancang Bangun

Rancang bangun adalah teknik pembuatan sistem untuk membuat sistem baru yang dapat menggantikan atau membenahi sistem saat ini, baik secara keseluruhan maupun hanya sebagian. [6]

2.4. Website

Website adalah sekumpulan halaman yang dapat menampilkan informasi berupa teks, data, gambar, animasi, suara, video, dan berbagai jenis konten lainnya. *Website* dapat beroperasi baik secara statis maupun dinamis dengan membangun serangkaian pondasi dalam yang saling berhubungan, yang memungkinkan interkoneksi antar halaman (*hyperlink*). [6]

2.5. PHP

PHP adalah bahasa pemrograman *server-side* yang banyak digunakan untuk membuat program yang dinamis dan interaktif. PHP populer karena *syntax* nya yang mudah dipahami. Selain itu, PHP memungkinkan pembuatan halaman web dinamis dengan menyisipkan skrip PHP di dalam HTML. Dengan cara ini, PHP dapat diintegrasikan secara efektif dalam struktur dokumen HTML, yang memungkinkan pengelolaan konten dinamis menjadi lebih fleksibel. [7]

2.6. Framework

Framework atau kerangka kerja ialah kumpulan intruksi-intruksi yang disatukan dalam kelas dan fungsi-fungsi dengan masing-masing fungsi. Fungsi-fungsi tersebut memungkinkan pengembang untuk menggunakannya tanpa perlu menulis ulang sintaks program yang sama, sehingga dapat menghemat waktu. [8]

2.7. Laravel

Laravel adalah kerangka kerja pengembangan web berbasis *Model View Controller* (MVC) yang dimaksudkan untuk mengurangi biaya pengembangan dan

pemeliharaan perangkat lunak sambil meningkatkan kualitasnya. [9]

2.8. Filament

Filament adalah *admin panel* yang dirancang untuk mempermudah pembuatan kerangka kerja UI *full-stack* berbasis TALL stack (Tailwind CSS, Alpine.js, Laravel, Livewire). Kerangka ini menyediakan berbagai komponen interaktif yang sudah siap pakai, sehingga sangat cocok untuk membangun CMS atau area administrasi. [10]

2.9. MySQL

MySQL merupakan salah satu perangkat lunak yang termasuk dalam kategori *Database Management System* (DBMS) dan bersifat *open source*. Sebagai DBMS, MySQL menyediakan beragam fitur untuk menyimpan dan mengelola data dalam sebuah basis data, serta mendukung pelaksanaan *query* dan berbagai operasi lainnya. MySQL telah menjadi pilihan populer dalam pengembangan aplikasi web, sistem informasi bisnis, serta berbagai sistem lain yang memerlukan pengelolaan data yang andal dan mampu menangani skala besar. [7]

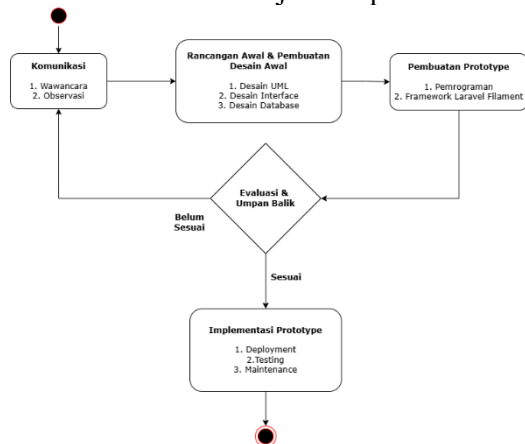
2.10. Prototype

Metode *prototype* adalah teknik pengembangan perangkat lunak yang memungkinkan pengembang untuk membuat model awal dari sistem yang akan dibangun. Metode ini digunakan pada tahap awal pengembangan untuk membangun dan menguji model awal sistem untuk menjelaskan persyaratan dan memastikan bahwa desain sistem akan berfungsi dengan baik sebelum dilanjutkan ke tahap pengembangan yang lebih lanjut. [11]

3. METODE PENELITIAN

Penelitian ini menggunakan metode pengembangan *prototype* sebagai pendekatan dalam perancangan sistem. Model ini diawali dengan pengumpulan kebutuhan dari pengguna terkait perangkat lunak yang akan dibuat. Prototipe yang dihasilkan kemudian dievaluasi secara terus-menerus oleh pengguna hingga tercapai spesifikasi yang sesuai dengan keinginan mereka. Pemilihan model *prototype* didasarkan pada sifat kebutuhan pengguna yang dinamis, serta peran aktif pengguna dalam

proses pengembangan yang memungkinkan identifikasi kesalahan sejak tahap awal.



Gambar 1. Metode *Prototype*

Berikut adalah penjelasan tahapan dari metode *prototype*:

3.1. Komunikasi

Pada tahap ini, pengembang sistem harus berbicara dengan pengguna untuk mengetahui kebutuhan dan batasan perangkat lunak. Wawancara, observasi atau diskusi langsung dapat digunakan untuk mendapatkan informasi ini. Selanjutnya, data yang dikumpulkan diperiksa untuk mendapatkan informasi yang dibutuhkan

3.2. Rancangan Awal dan Pembuatan Desain Awal

Pada tahap ini, pengembang menentukan kebutuhan sistem berdasarkan komunikasi dengan pengguna, seperti perangkat desa dan masyarakat, lalu merancang sistem sesuai kebutuhan tersebut. Tahapan meliputi perancangan desain sistem melalui pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* untuk menetapkan elemen penting dalam aplikasi. Selanjutnya, dilakukan pembuatan desain antarmuka melalui *prototyping* menggunakan *tools* seperti Figma untuk menguji konsep dan mendapatkan umpan balik awal. Selain itu, dirancang pula struktur database yang mencakup tabel, relasi, fitur, dan tipe data, guna memastikan integritas, efisiensi, dan skalabilitas sistem.

3.3. Pembuatan Prototype

Pengkodean sudah mulai dilakukan pada tahap pembuatan prototype. Pada tahap ini, desain tampilan dan struktur aplikasi serta

pengembangan kode untuk penerapan fitur dasar dilakukan. Pengkodean dilakukan untuk memastikan bahwa aplikasi berfungsi sesuai dengan rancangan awal.

3.4. Evaluasi & Umpan Balik Prototype

Pada tahap ini, prototype yang telah dirancang dievaluasi bersama dengan perangkat desa dan perwakilan masyarakat. Evaluasi dilakukan melalui demonstrasi prototype di mana fitur dan fungsi utama prototype diperkenalkan secara langsung. Selain itu, *user* diminta untuk memberikan masukan melalui *feedback form* yang dirancang untuk menangkap opini, saran, dan tanggapan mereka terhadap prototype.

3.5. Implementasi Prototype

Pada tahap implementasi, pengembang membangun aplikasi versi lengkap berdasarkan prototype yang telah diuji dan mendapat masukan. Fitur seperti sistem administrasi, informasi, dan pengaduan disesuaikan dengan kebutuhan pengguna. Pengujian dilakukan menggunakan *black-box testing* untuk memvalidasi fungsi sistem, serta *User Acceptance Testing* (UAT) untuk menilai kepuasan pengguna akhir. Setelah aplikasi dinyatakan layak, dilanjutkan dengan tahap pemeliharaan dan pembaruan guna menjaga kinerja sistem dan memperbaiki masalah yang muncul.

4. HASIL DAN PEMBAHASAN

4.1. Komunikasi

4.1.1. Observasi

Tahap komunikasi merupakan tahapan untuk mencari kebutuhan mengenai sistem yang akan dikembangkan dengan cara melakukan observasi dan wawancara serta menentukan indentifikasi kebutuhan pengguna, dan menentukan kebutuhan sistem.

4.1.2. Wawancara

Pada tahap ini, peneliti melakukan observasi di Desa Cantigi Kulon, yang merupakan subjek penelitian sesuai dengan topik yang diangkat. Desa ini terletak di Kecamatan Cantigi, Kabupaten Indramayu.

4.1.3. Analisis Kebutuhan Pengguna

Berdasarkan hasil obeservasi dan wawancara dengan narasumber, indentifikasi Kebutuhan pengguna sebagai berikut:

1. Sistem pelayanan desa yang mudah di akses dan digunakan oleh masyarakat.
2. Fitur pengajuan surat online.
3. Fitur pengaduan untuk masyarakat yang mudah di akses.
4. Fitur informasi dari desa untuk masyarakat seperti berita, foto kegiatan dan profil desa.
5. Fitur kelola surat untuk admin.
6. Fitur kelola pengaduan untuk admin.
7. Fitur informasi yang dapat dikelola dengan mudah untuk admin.

4.1.4. Analisis Kebutuhan Sistem

1. Functional Requirements

Kebutuhan fungsional yang diperlukan pada aplikasi *village service* berbasis web adalah sebagai berikut:

Tabel 1. Kebutuhan Fungsional

No	Kebutuhan
1	Login User
2	Register
3	Halaman Beranda
4	Halaman Mengajukan Surat
5	Halaman Mengajukan Pengaduan
6	Halaman Lihat Profil Desa
7	Halaman Lihat Kabar Desa
8	Halaman Login Admin
9	Halaman Dashboard
10	Halaman Kelola Surat
11	Halaman Kelola Pengaduan
12	Halaman Kelola Galeri Desa
13	Halaman Kelola Kabar Desa
13	Logout

2. Non-Functional Requirements

Kebutuhan non-fungsional merupakan kebutuhan teknis yang diperlukan dalam pengembangan aplikasi, pengujian aplikasi, dan penggunaan aplikasi yang dikembangkan. Berikut adalah tabel kebutuhan non-fungsional pada Tabel 2.

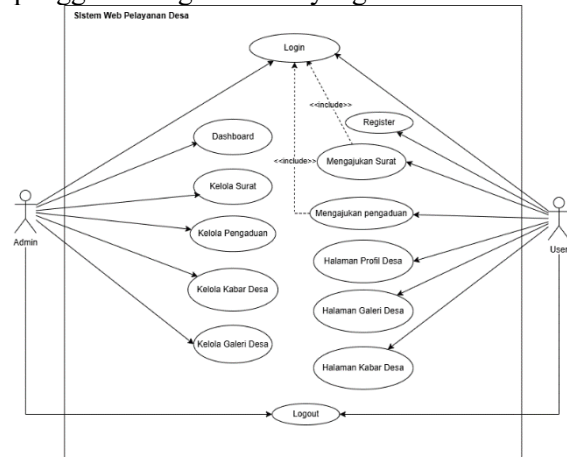
Table 2. Kebutuhan Non Fungsional

No	Nama Perangkat	Jenis
1	Visual Studio Code	Perangkat Lunak
2	Laragon	Perangkat Lunak
3	PHP	Perangkat Lunak
4	Laravel	Perangkat Lunak
5	Filament	Perangkat Lunak
6	Tailwind CSS	Perangkat Lunak
7	Google Chrome	Perangkat Lunak
8	Figma	Perangkat Lunak
9	Draw.io	Perangkat Lunak

4.2. Rancangan Awal & Pembuatan Desain Awal

4.2.1. Use Case Diagram

Use Case Diagram merupakan suatu model yang menggambarkan hubungan interaksi antara aktor dan sistem sehingga bisa mendeskripsikan jenis interaksi antara pengguna dengan sistem yang dibuat.

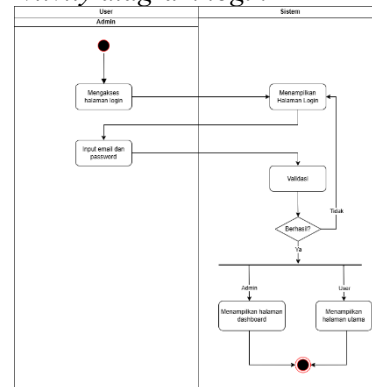


Gambar 2. Use Case Diagram

4.2.2. Activity Diagram

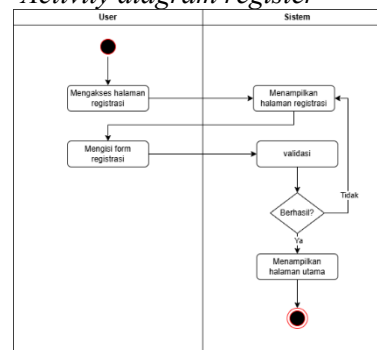
Berikut adalah *activity diagram* pada aplikasi *Village Service* berbasis web:

1. Activity diagram login



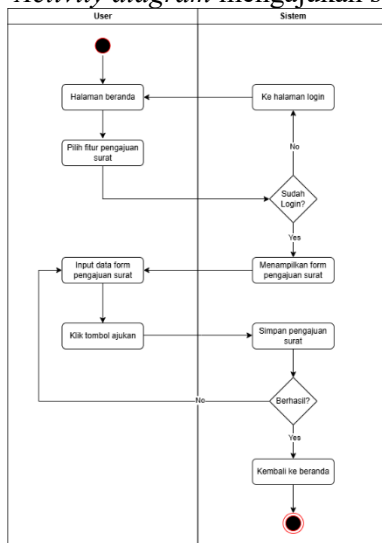
Gambar 3. Activity Diagram Login

2. Activity diagram register



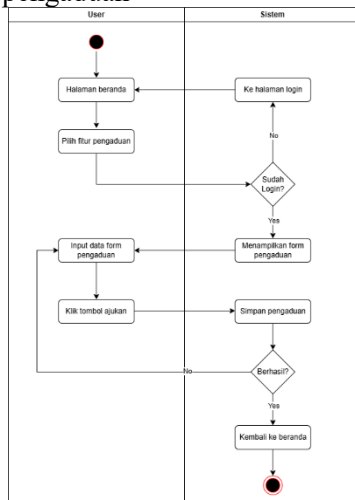
Gambar 4. Activity Diagram Register

3. Activity diagram mengajukan surat



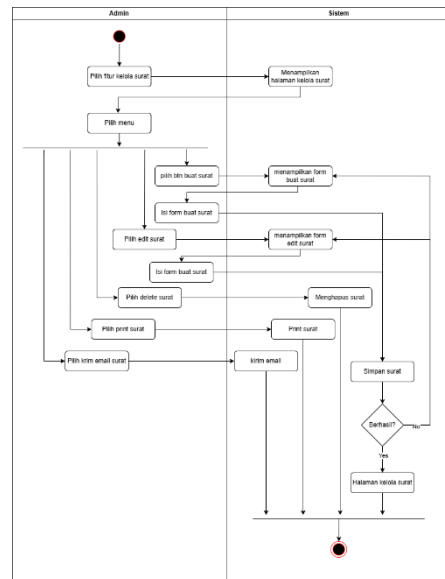
Gambar 5. Activity Diagram Mengajukan Surat

4. Activity diagram mengajukan pengaduan



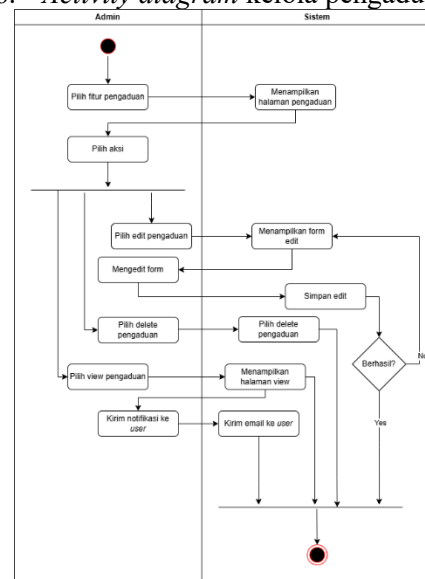
Gambar 6. Activity Diagram Mengajukan Pengaduan

5. Activity diagram kelola surat



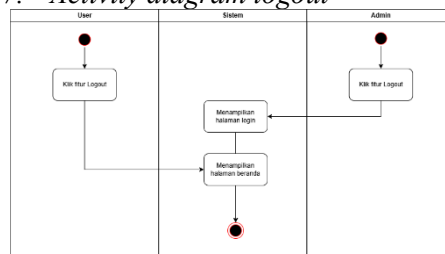
Gambar 7. Activity Diagram Kelola Surat

6. Activity diagram kelola pengaduan



Gambar 8. Activity Diagram Kelola Pengaduan

7. Activity diagram logout

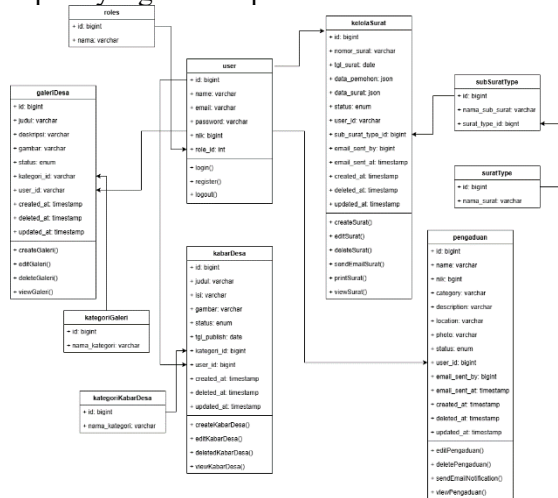


Gambar 9. Activity Diagram Logout

4.2.3. Class Diagram

Struktur sistem dalam *website Village Service* memfasilitasi interaksi antara berbagai

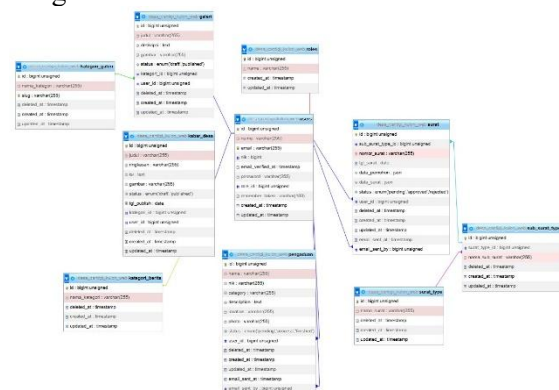
objek yang saling terkait, yang dapat diilustrasikan melalui sebuah *class diagram* seperti yang terlihat pada di bawah ini



Gambar 10. *Class Diagram*

4.2.4. Database Design

Database design adalah jenis perancangan yang terdiri dari pembuatan tabel yang terdiri dari *field* data dan *field* kunci yang disesuaikan dengan masalah awal.



Gambar 11. *Database Design*

4.3. Pembuatan Prototype

Pada tahap ini, *prototype* sistem dikembangkan menggunakan Laravel sebagai *framework* utama, Filament untuk *admin panel*, dan HTML, CSS (menggunakan *framework* Tailwind CSS), serta *JavaScript* untuk antarmuka pengguna dan telah mengintegrasikan *frontend* dan *backend*. Sistem ini sudah mencakup fungsionalitas penuh, dengan integrasi antara *frontend* dan *backend*, serta penggunaan *database* MySQL untuk penyimpanan data. *Prototype* ini memungkinkan pengujian fungsionalitas dasar dan interaksi antara antarmuka pengguna dan

sistem backend yang terhubung secara langsung.

4.4. Evaluasi & Umpan Balik *Prototype*

Untuk menguji sistem secara menyeluruh, tahap evaluasi dan *feedback* dilakukan setelah *prototype* selesai dan terintegrasi. Pengguna atau pemangku kepentingan akan memiliki kemampuan untuk menguji sistem dan memberikan umpan balik tentang fungsionalitas, kinerja, dan tampilan aplikasi. Evaluasi ini berfokus pada menemukan masalah atau kekurangan yang dapat diperbaiki, serta memberikan rekomendasi untuk peningkatan sistem. Berdasarkan umpan balik ini, Pengembang akan melakukan perbaikan dan penyesuaian pada aplikasi sebelum implementasi penuh dimulai.

4.5. Implementasi *Prototype*

4.5.1. Tampilan Akhir

Berikut adalah beberapa halaman yang sudah diimplementasikan dengan menerapkan model *prototype*:

1. Halaman *login*

Gambar 12. Halaman *Login*

- ## 2. Halaman registrasi



Pendaftaran Akun

Nama

Email

Nomor Induk Kependudukan

Kata Sandi

Konfirmasi Kata Sandi

DAFTAR

[Sudah terdaftar?](#)

Gambar 13. Halaman *Register*

3. Halaman mengajukan surat

Gambar 14. Halaman Mengajukan Surat

4. Halaman mengajukan pengaduan

Gambar 15. Halaman Mengajukan Pengaduan

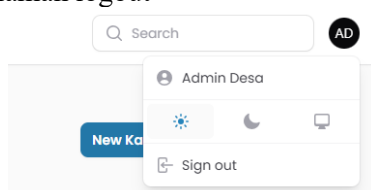
5. Halaman kelola surat

Gambar 16. Halaman Kelola Surat

6. Halaman kelola pengaduan

Gambar 17. Halaman Kelola Pengaduan

7. Halaman logout



Gambar 18. Halaman Logout

4.5.2. Pengujian Sistem

1. Black Box Testing

Menurut [12], Metode pengujian perangkat lunak yang dikenal sebagai *black-box* memungkinkan penguji untuk memeriksa fungsionalitas aplikasi tanpa mengetahui struktur kode internalnya. *Black-box* digunakan untuk menemukan kesalahan dan mendemonstrasikan fungsional aplikasi saat dioperasikan, apakah *input* diterima dengan benar dan *output* yang dihasilkan telah sesuai dengan yang diharapkan.

Tabel 3. Hasil Pengujian *Black-box Testing*

<i>Input</i>	Hasil yang diharapkan	Hasil Pengujian
Halaman Register		
Memasukkan <i>form</i> registrasi	Pengguna baru akan tersimpan datanya dan akan diarahkan ke halaman beranda.	Sesuai harapan
Halaman Login		
Memasukkan <i>form</i> login	Masuk ke halaman Beranda atau <i>Dashboard Admin</i>	Sesuai harapan
Halaman Mengajukan Surat		
Memasukkan <i>form</i> pengajuan surat	Data akan tersimpan dan akan diarahkan ke halaman beranda.	Sesuai harapan
Halaman Mengajukan Pengaduan		
Memasukkan <i>form</i> pengajuan pengaduan	Data akan tersimpan dan akan diarahkan ke halaman beranda.	Sesuai harapan
Halaman Kelola Surat		
Mengelola surat masuk	Berhasil membuat, mengedit, menghapus surat dan mengirim email	Sesuai harapan
Halaman Kelola Pengaduan		
Mengelola pengaduan	Berhasil mengedit, menghapus, mengirim email.	Sesuai harapan
Halaman Logout		

Menekan tombol <i>logout</i>	Menampilkan halaman beranda atau <i>login</i>	Sesuai harapan
------------------------------	---	----------------

2. User Acceptance Testing (UAT)

User acceptance testing (UAT) adalah prosedur pengujian oleh pengguna akhir untuk mengevaluasi perangkat lunak. Hasil pengujian ini menghasilkan dokumen yang menunjukkan bahwa perangkat lunak telah memenuhi kebutuhan yang ditentukan dan dapat digunakan. [13]

Berdasarkan hasil pengujian *black-box*, aplikasi telah berfungsi dengan baik dan sesuai harapan. Untuk memastikan bahwa prototipe yang dikembangkan dalam penelitian ini dapat diterima oleh pengguna, dilakukan evaluasi terhadap aplikasi melalui kuesioner. Evaluasi ini menggunakan teknik rating scale dengan melibatkan 30 responden. Menurut [14] menegaskan bahwa 30 responden sering dijadikan *golden rule* dalam pengembangan dan validasi *instrument* penelitian, karena sering kali jumlah yang lebih kecil menjadi alasan penolakan publikasi. Data dari kuesioner dianalisis menggunakan skala Likert, yang terdiri dari skala 1 hingga skala 4. Skala Likert adalah metode pengukuran yang digunakan dalam penelitian untuk menilai sejauh mana seorang responden setuju atau tidak setuju terhadap suatu pernyataan, penggunaan skala 1-4 untuk mendorong kecenderungan responden dalam pengambilan sikap yang tidak netral. Berikut ini adalah proses dan hasil dari evaluasi kuesioner tersebut.

Merujuk pada penelitian [15] pertanyaan-pertanyaan ini dirancang berdasarkan standar ISO/IEC 9126 yang merupakan kerangka internasional untuk pengukuran kualitas perangkat lunak. Dalam penelitian ini, kualitas sistem dianalisis melalui empat variabel utama, yaitu fungsionalitas sistem, kinerja sistem, pengalaman & tampilan antarmuka, serta efisiensi. Pemilihan indikator dalam setiap variabel mengacu pada dimensi kualitas yang relevan agar dapat mencerminkan persepsi pengguna secara objektif dan sistematis.

Berdasarkan hasil pengujian *User Acceptance Testing* (UAT), menunjukkan bahwa aplikasi memiliki persentase kelayakan sebesar 91,82%. Angka ini didapat dari empat variabel pertanyaan dimana hasil dari tiap

variabel yaitu: fungsionalitas sistem 89,15%, kinerja sistem 95%, pengalaman & tampilan antarmuka sistem 95,62%, efisiensi & produktivitas sistem 87,5%. Hasil ini mengindikasikan bahwa sebagian besar pengguna akhir merasa bahwa aplikasi ini telah memenuhi kebutuhan dan fungsionalitas yang diharapkan.

5. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan pada bab sebelumnya mengenai aplikasi *village service* berbasis web menggunakan *framework* Laravel filament, dapat disimpulkan bahwa:

1. Dengan memanfaatkan Laravel Filament dan metode *prototype*, aplikasi *Village Service* berbasis web berhasil dikembangkan secara responsif dan sesuai kebutuhan pengguna. Sistem ini mendukung fitur utama seperti pengelolaan galeri, pengaduan, surat, kabar desa, serta pengajuan surat dan pengaduan secara online.
2. Hasil pengujian menunjukkan sistem berfungsi dengan baik secara teknis dan diterima positif oleh pengguna. *Black-box testing* memastikan fitur berjalan sesuai rancangan, sementara *User Acceptance Test* (UAT) menghasilkan nilai kelayakan rata-rata sebesar 91,82%.

Sebagai rekomendasi, dalam pengembangan selanjutnya, dapat dilakukan:

1. Disarankan menambahkan fitur “forgot password” untuk memungkinkan pengguna mengatur ulang kata sandi yang terlupakan.
2. Direkomendasikan untuk menambahkan fitur kelola user, kelola page utama, agar tampilan halaman utama bisa diubah kapan saja jika ingin memperbaharui nya.
3. Disarankan menambahkan fitur “tracking status surat” untuk mengetahui sejauh mana proses surat yang diajukan, pengembang dapat membuat halaman dashboard user untuk memuat fitur tersebut agar rapih.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada kedua orang tua yang selalu memberikan dukungan baik doa, dan semangat yang tiada henti. Dan juga terima kasih kepada dosen pembimbing yang telah membimbing hingga penelitian ini selesai.

DAFTAR PUSTAKA

- [1] S. Afriyah, "The Impact of E-Government Services, Citizen Participation, and Transparency on Public Trust in Government," *Global International Journal of Innovative Research*, vol. 2, no. 6, pp. 1246–1261, Jun. 2024, doi: 10.59613/global.v2i6.200.
- [2] E. Supriyati and M. Azrino Gustalika, "Extreme Programming Method for Integrated Service System Website Development in Rejosari Village," *Jurnal Riset Informatika*, vol. 5, no. 3, pp. 533–538, Jun. 2023, doi: 10.34288/jri.v5i3.527.
- [3] K. F. Siman and J. Wiratama, "A Web-based Village Administrative Information Systems for Improvement Quality of Service towards Smart Village Concept," *G-Tech: Jurnal Teknologi Terapan*, vol. 7, no. 4, pp. 1519–1528, Oct. 2023, doi: 10.33379/gtech.v7i4.3196.
- [4] R. R. Marlina, W. Sejati, W. A. Nisa, U. Pujayanti, R. Sopian, and W. Noergana, "Rancang Bangun Website Desa Citengah untuk Pengembangan Promosi Potensi Desa," *JPPM (Jurnal Pengabdian dan Pemberdayaan Masyarakat)*, vol. 6, no. 1, p. 193, Jun. 2022, doi: 10.30595/jppm.v6i1.7667.
- [5] D. Permatasari, "Kualitas E-Government Dalam Pelayanan Tiket Wisata Berbasis Website Di Dinas Kebudayaan, Kepemudaan, Dan Olahraga Serta Pariwisata Kota Surabaya," *Publika*, vol. 11, no. 3, pp. 2303–2314, 2023.
- [6] P. Anjelita and E. Rosiska, "Rancang Bangun Sistem Informasi E-learning Pada SMK Negeri 3 Batam," 2019.
- [7] Y. A. Dewantara, A. Primajaya, and E. H. Nurkifli, "RANCANG BANGUN WEBSITE PENYEWAAN LAPANGAN MENGGUNAKAN FRAMEWORK LARAVEL (STUDI KASUS: GOR JAYA ABADI)," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3, Aug. 2024, doi: 10.23960/jitet.v12i3.4762.
- [8] A. F. Sallaby and I. Kanedi, "Perancangan Sistem Informasi Jadwal Dokter Menggunakan Framework Codeigniter," *Jurnal Media Infotama*, vol. 16, no. 1, Feb. 2020.
- [9] N. Naofal, M. R. D. Ulhaq, and C. Prianto, "Development of E-Commerce Information System at Az-Zahra Shop Using Laravel Framework," *JOMLAI: Journal of Machine Learning and Artificial Intelligence*, vol. 1, no. 1, pp. 95–106, Mar. 2022, doi: 10.55123/jomlai.v1i1.176.
- [10] Filament, "Introducing Filament Build a Beautiful & Powerful Laravel Admin Panel In Minutes.," <https://filamentphp.com/use-cases/admin-panel#:~:text=Laravel%20Admin%20Panel%20In%20Minutes,next%20CMS%20or%20administration%20area>.
- [11] Indriani and F. Suhana S, "Rancang Bangun Sistem Informasi Pelayanan Administrasi Desa Berbasis Website Menggunakan Metode Prototype Di Desa Ciomas Kabupaten Ciamis," *Computatio: Journal of Computer Science and Information Systems*, vol. 7, no. 2, pp. 203–210, 2023.
- [12] T. D. Capote, "A Comparative Study of Black Box and White Box Testing Techniques in Modern Software Development," *Frontiers in Engineering and Technology*, vol. 5, no. 1, pp. 1–7, 2024, [Online]. Available: https://iaeme.com/Home/journal/FET1ID:FET_05_01_001Availableonlineathttps://iaeme.com/Home/issue/FET?Volume=5&Issue=1
- [13] Elsen, "Analisis Dan Perancangan Sistem Informasi Penyewaan Forklift Berbasis Web Pada Cv Jaya Abadi Forklift Dengan Metode Pengujian User Acceptance Test (UAT)," 2022. Accessed: Dec. 15, 2024. [Online]. Available: <https://repositori.buddhidharma.ac.id/1453/>
- [14] J. Gunawan, C. Marzilli, and Y. Aunguroch, "Establishing appropriate sample size for developing and validating a questionnaire in nursing research," *Belitung Nurs J*, vol. 7, no. 5, pp. 356–360, Oct. 2021, doi: 10.33546/bnj.1927.
- [15] D. Ayu Agung Indra Swari, D. Panji Agustino, N. Luh Komang Irma Wulandari, Y. Vony Sora Zalogo, and J. Lau, "Quality Analysis of the Registration Information System Website using ISO/IEC 9126 Standard," *Journal of Applied Informatics and Computing (JAIC)*, vol. 9, no. 3, p. 674, 2025, [Online]. Available: <http://jurnal.polibatam.ac.id/index.php/JAIC>