

Penerapan Konsep MVC dalam Pembangunan Aplikasi Sikerabat Menggunakan Codeigniter dan PostgreSQL

^{1,*}Dimas Dwi Randa, ²Dhella Amelia

¹ Jurusan Sains Data, FS, Institut Teknologi Sumatera, Lampung Selatan, Indonesia

² Jurusan Ilmu Komputer, FMIPA, Universitas Lampung, Bandar Lampung, Indonesia

Abstrak — Perkembangan teknologi informasi yang telah pesat telah mendorong transformasi pada layanan publik serta mendorongnya inisiatif *e-Government* untuk lingkup pemerintahan. Penelitian ini membahas tentang digitalisasi kerjasama antara lembaga pemerintahan khusus di Biro Pemerintahan Provinsi Sumatera Barat, sebuah aplikasi berbasis website yang memfasilitasi koordinasi, pengajuan, verifikasi pengajuan dan pengelolaan data kerjasama. Metode dalam merancang aplikasi ini menggunakan RAD (Rapid Application Development) karena sangat cocok dalam penerapannya. Aplikasi ini memberikan kontribusi positif dalam mengoptimalkan pengelolaan datanya dan mengurangi hambatan administrasi antar lembaga pemerintahan. Dari temuan ini menunjukkan bahwa platform digital seperti Sikerabat mendukung peran strategis dan meningkatkan efisiensi tata kelola pada sektor publik. Setiap lembaga pemerintahan yang ingin mengajukan PKS dan MOU bisa melalui aplikasi Sikerabat. Pengajuan tersebut kemudian di verifikasi oleh Biro Pemerintahan dan ditindaklanjuti, sehingga rekapan hasil pengajuannya teroganisir dengan sistem digital. Hasil evaluasi awal menunjukkan dengan menggunakan framework mvc codeigniter ini dalam pembangunan aplikasi bisa memberikan kontribusi positif dalam mengoptimalkan koordinasi.

Kata Kunci: E-Government; Framework; MVC; Platform; RAD.

Abstract — The rapid development of information technology has driven transformation in public services and encouraged *e-Government* initiatives within the scope of government. This study discusses the digitalization of collaboration between government institutions, specifically within the Government Bureau of West Sumatra Province, through a web-based application that facilitates coordination, submission, verification of submissions, and data management of collaborations. The method used to design the application is RAD (Rapid Application Development), which is highly suitable for its implementation. This application contributes positively by optimizing data management and reducing administrative barriers between government institutions. The findings indicate that digital platforms like sikerabat support strategic roles and enhance governance efficiency in the public sector. Any government institution wishing to submit a Cooperation Agreement (PKS) or Memorandum of Understanding (MoU) can do so through the sikerabat application. These submissions are then verified by the Government Bureau and followed up accordingly, resulting in an organized digital system for submission records. Initial evaluation results show that using the MVC CodeIgniter framework in developing the application can positively contribute to optimizing coordination.

Keywords: E-Government; Framework; MVC; Platform; RAD.

* Corresponding author :
Dimas Dwi Randa
Program Studi Sains Data, Lampung Selatan, Indonesia
dimas.randa@sd.itera.ac.id

1. PENDAHULUAN

Kerjasama pemerintahan di Indonesia mengalami peningkatan yang cukup signifikan, terutama pada era digital saat ini. Perkembangan teknologi informasi telah memberikan dampak yang sangat positif terhadap transformasi sistem pemerintahan berbasis teknologi *website*, yang mencakup berbagai sektor secara luas. *Website* tidak hanya berfungsi sebagai media penyampaian informasi, tetapi juga menjadi sarana penting dalam proses pengolahan dan manajemen data secara elektronik. Melalui sistem berbasis web, masyarakat dan instansi pemerintah dapat melakukan proses entri data, verifikasi, rekapitulasi, serta penyimpanan data hingga tahap pelaporan. Proses ini jauh lebih efisien dibandingkan metode manual yang dilakukan pada masa lalu, di mana pencatatan dan pencarian data membutuhkan waktu

lama serta rentan terhadap kesalahan manusia. Salah satu permasalahan umum yang dihadapi pada sistem manual adalah kesulitan dalam melakukan rekap data tahunan, terutama dalam pencarian kembali data lama yang tidak terdokumentasi dengan baik. Sebelum adanya sistem informasi terintegrasi, setiap Organisasi Perangkat Daerah (OPD) yang akan mengajukan kerja sama melalui Memorandum of Understanding (MoU), Perjanjian Kerja Sama (PKS), atau addendum, harus mendatangi langsung Biro Pemerintahan. Proses verifikasi usulan tersebut masih dilakukan secara manual oleh petugas biro, sehingga OPD hanya dapat menunggu informasi status pengajuan tanpa kepastian waktu. Kondisi ini menunjukkan perlunya sebuah sistem informasi yang mampu mengelola proses pengajuan, verifikasi, dan pelaporan secara cepat, transparan, dan terintegrasi. Menjawab permasalahan tersebut, peneliti merancang dan membangun aplikasi sikerabat berbasis web yang digunakan oleh Biro Pemerintahan Provinsi Sumatera Barat. Sistem ini memungkinkan OPD yang telah memiliki akun untuk melakukan proses pengajuan dan addendum secara digital. Akses pengguna pada aplikasi ini terbagi menjadi tiga level, yaitu pengguna publik, pengguna OPD (privat), dan admin biro pemerintahan. Hanya pengguna yang memiliki akun resmi (*username* dan *password*) yang dapat melakukan pengajuan kerja sama secara *daring*, sehingga keamanan dan keabsahan data lebih terjamin. Dalam proses pengembangannya, sistem sikerabat menggunakan metode *Rapid Application Development (RAD)* yang dikenal mampu mempercepat proses pembangunan perangkat lunak melalui iterasi cepat dan keterlibatan aktif pengguna [1]. Pendekatan RAD terbukti efektif digunakan dalam berbagai penelitian pengembangan sistem informasi, seperti sistem informasi keuangan desa [2] dan sistem perpustakaan sekolah [3]. Untuk sisi teknis, aplikasi ini dibangun menggunakan bahasa pemrograman *PHP* dengan *framework CodeIgniter* sebagai kerangka kerja utama. *Framework CodeIgniter* banyak digunakan dalam pengembangan aplikasi web karena mendukung arsitektur *Model-View-Controller (MVC)* yang memudahkan pemisahan antara logika bisnis dan tampilan [4]. Sejumlah penelitian juga menunjukkan keunggulan *CodeIgniter* dalam hal kemudahan implementasi dan performa yang baik dibanding *framework* lain seperti *Laravel* [5]. Untuk pengelolaan data, digunakan basis data *PostgreSQL* yang memiliki kemampuan tinggi dalam menangani *query* kompleks dan mendukung integrasi sistem berskala besar [6]. *PostgreSQL* terbukti handal dalam membaca dan mengeksekusi data dalam jumlah besar dengan stabilitas yang baik, sehingga cocok digunakan untuk sistem pemerintahan yang membutuhkan integritas data tinggi. Dengan penerapan metode *RAD*, *framework CodeIgniter*, dan *PostgreSQL* sebagai sistem basis data, diharapkan aplikasi mampu meningkatkan efisiensi, transparansi, dan kecepatan layanan kerja sama antar instansi pemerintahan di Provinsi Sumatera Barat.

2. METODOLOGI PENELITIAN

Metode penelitian adalah tahapan atau langkah yang dilakukan oleh peneliti dalam rangka untuk mengumpulkan informasi atau data serta melakukan investigasi pada data yang telah didapatkan. Pada Bagian metode ini akan memandu tahapan perencanaan dan perancangan sistem agar mencapai hasil yang maksimal. Ada beberapa tahapan dalam merancang sistem yang terdiri dari:

2.1 Perencanaan Kebutuhan (Requirement Planning)

Pada tahap ini dilakukan pengumpulan kebutuhan sistem melalui:

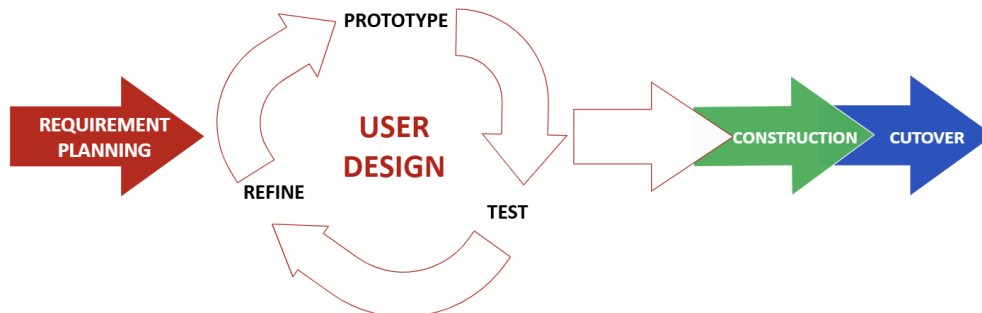
- a. Studi pustaka terkait sistem kerja pemerintahan dan pengelolaan MoU/PKS,
- b. Observasi terhadap proses bisnis Biro Pemerintahan,
- c. Wawancara dengan pihak yang akan menggunakan sistem (admin atau petugas).

Tujuan tahap ini adalah untuk merumuskan kebutuhan fungsional dan non-fungsional dari aplikasi Sikerabat.

2.2 Model Pengembangan Sistem

Penelitian ini menggunakan pendekatan *Rapid Application Development (RAD)* dalam membangun rancangan aplikasi sistem informasi kerjasama (Sikerabat). RAD adalah model pengembangan perangkat lunak yang menekankan pada kecepatan dan iterasi melalui pembuatan prototipe, keterlibatan pengguna secara aktif, dan pengujian berulang [7]. Model ini dipilih karena mendukung pengembangan aplikasi berbasis online secara cepat dengan feedback langsung dari pengguna/stakeholder, serta

memungkinkan implementasi konsep **Model-View-Controller (MVC)** secara bertahap dan modular menggunakan *framework codeIgniter* [8]. Alasan menggunakan pendekatan *RAD* adalah karena memiliki kelebihan, di antaranya siklus pengembangan yang lebih cepat, fleksibilitas tinggi, melibatkan user secara langsung, serta mengurangi kesalahan data.



Gambar 1. *Rapid Application Development (RAD) Sikerabat*

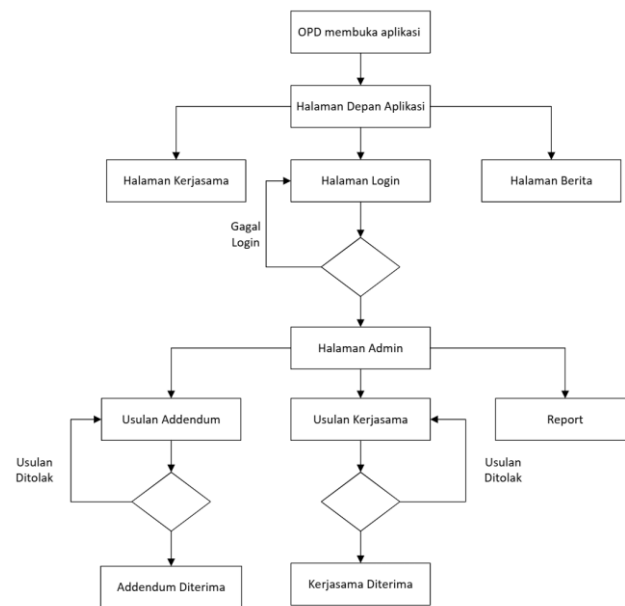
2.3 Model View Controller (MVC)

Metode *MVC (Model, View, Controller)* merupakan pola arsitektur perangkat lunak yang memisahkan tanggung jawab aplikasi menjadi tiga komponen berbeda agar sistem menjadi modular dan mudah dikembangkan. Model bertanggung jawab terhadap pengelolaan data dan logika bisnis, *View* menangani penyajian antarmuka pengguna (UI), sedangkan *Controller* berfungsi sebagai pengatur komunikasi antara *Model* dan *View*, menerima input pengguna dan memutuskan tindakan yang akan dilakukan. Penerapan *MVC* telah terbukti efektif dalam pengembangan aplikasi web modern, mempercepat waktu pemeliharaan, meningkatkan keterbacaan kode, serta mendukung kerja tim paralel antara *front-end* dan *back-end* [9]. Selain itu, penelitian *Empirical Assessment of the Quality of MVC Web Applications* menunjukkan bahwa peran *Controller* dan Model memiliki metrik arsitektural yang berbeda dan penting untuk menjaga kualitas aplikasi *MVC* secara keseluruhan [10]. Terdapat tiga komponen didalam metode *MVC*, yaitu

- Model mengelola basis data (RDBMS), dan salah satu basis data yang digunakan adalah *PostgreSQL* [11]. Data dikelola melalui *class* yang berfungsi untuk membuat *CRUD (Create, Read, Update, Delete)*.
- View* merupakan bagian yang akan menampilkan data yang telah diinputkan dan *view* dapat berupa *html, css, javascript* dan lainnya.
- Controller* adalah jembatan yang menghubungkan antara model dan *view* agar pemrosesan *logic* sistemnya dapat berjalan dengan baik berupa *class* dan fungsi.

2.4 Activity Diagram User Level Opd/Instansi

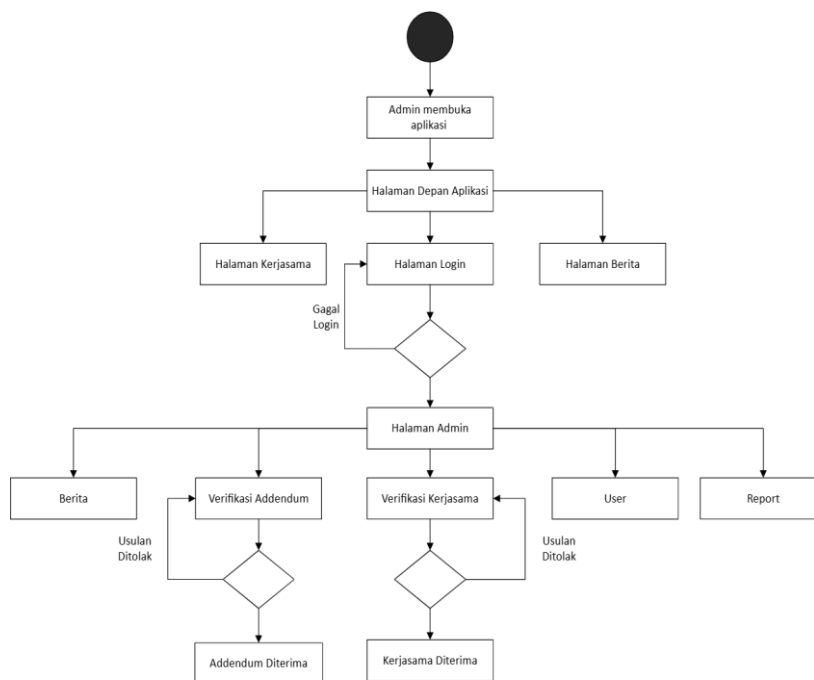
Activity diagram Adalah alur penggunaan yang dilakukan didalam sistem yang sedang berjalan [12]. Aktivitas pengguna (dalam hal ini OPD atau instansi) saat mengakses aplikasi. Proses dimulai ketika OPD membuka aplikasi dan diarahkan ke halaman depan, yang menyediakan beberapa menu utama seperti halaman kerjasama, halaman *login*, dan halaman berita. Untuk mengajukan usulan kerjasama, pengguna harus melakukan login terlebih dahulu melalui halaman Login. Jika proses login gagal, maka pengguna tidak dapat melanjutkan.



Gambar 2. *Activity Diagram* Berjalan OPD

2.5 Activity Diagram User Level Opd/Instansi

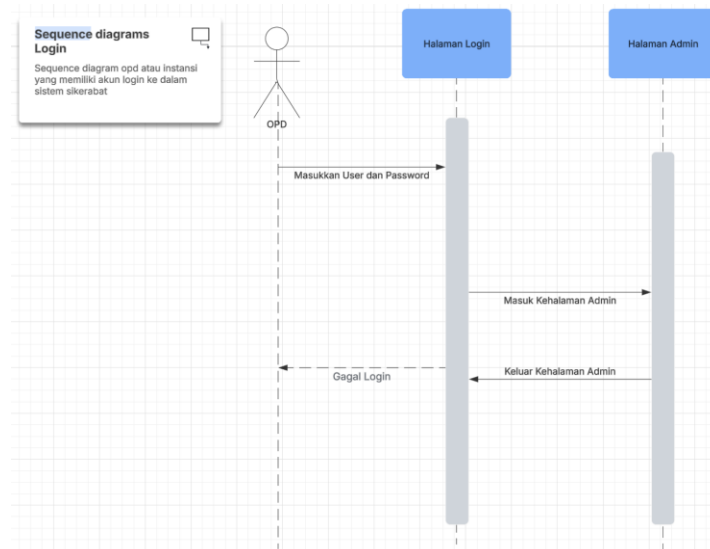
Alur aktivitas yang dilakukan oleh admin ketika mengakses aplikasi. Proses dimulai saat admin membuka aplikasi dan diarahkan ke halaman depan, yang menyediakan beberapa pilihan menu seperti halaman kerjasama, halaman *login*, dan halaman berita. Untuk mengakses fitur-fitur administrasi, admin harus melalui proses login terlebih dahulu. Jika proses login gagal, maka sistem akan menolak akses dan pengguna tidak dapat melanjutkan. Namun jika login berhasil, admin akan diarahkan ke halaman admin sebagai pusat kontrol utama dalam sistem. Di halaman admin, terdapat berbagai fitur yang dapat diakses oleh admin, antara lain verifikasi kerjasama, verifikasi addendum, manajemen *user*, *report*, dan pengelolaan berita. Admin memiliki kewenangan untuk memverifikasi usulan kerjasama yang telah diajukan oleh instansi atau OPD.



Gambar 3. *Activity Diagram* Berjalan Admin

2.6 Sequence Diagram Login

Sequence diagram proses *login* yang dilakukan oleh pengguna dari pihak OPD (Organisasi Perangkat Daerah) ke dalam sistem sikerabat. Proses dimulai ketika user OPD membuka halaman *login* dan memasukkan *username* dan *password* sebagai kredensial akses. Sistem akan memproses data yang dimasukkan. Apabila *username* atau *password* tidak sesuai, maka sistem akan memberikan respon berupa “Gagal Login”, dan pengguna tidak dapat melanjutkan ke halaman berikutnya. Namun jika data *login valid*, maka pengguna akan diarahkan menuju halaman admin, yang merupakan halaman utama untuk melakukan berbagai aktivitas seperti mengajukan kerjasama, melihat status usulan, atau mengelola data lainnya.



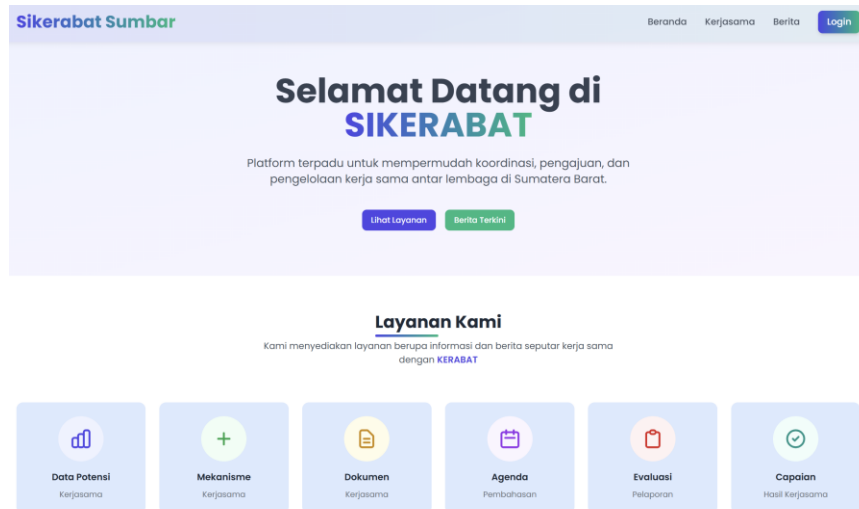
Gambar 4. *Sequence Diagram Login User*

3. HASIL DAN PEMBAHASAN

Berdasarkan hasil penelitian yang telah dilakukan, diperoleh sebuah sistem informasi kerjasama daerah (Sikerabat) yang dirancang untuk membantu Biro Pemerintahan Provinsi Sumatera Barat dalam mengelola data kerjasama antarinstansi secara lebih efektif. Sistem ini berbasis *website* sehingga mudah diakses oleh dinas atau OPD terkait kapan saja dan di mana saja. Selain mendukung efisiensi birokrasi, sistem juga menerapkan sanitasi data untuk mencegah kesalahan input serta mengurangi risiko duplikasi dan inkonsistensi. Teknik sanitasi tersebut meliputi validasi input sisi server, pemeriksaan format data, serta pembersihan data dari karakter berbahaya untuk menjaga keamanan koneksi dan basis data [13].

3.1 Halaman utama aplikasi sikerabat

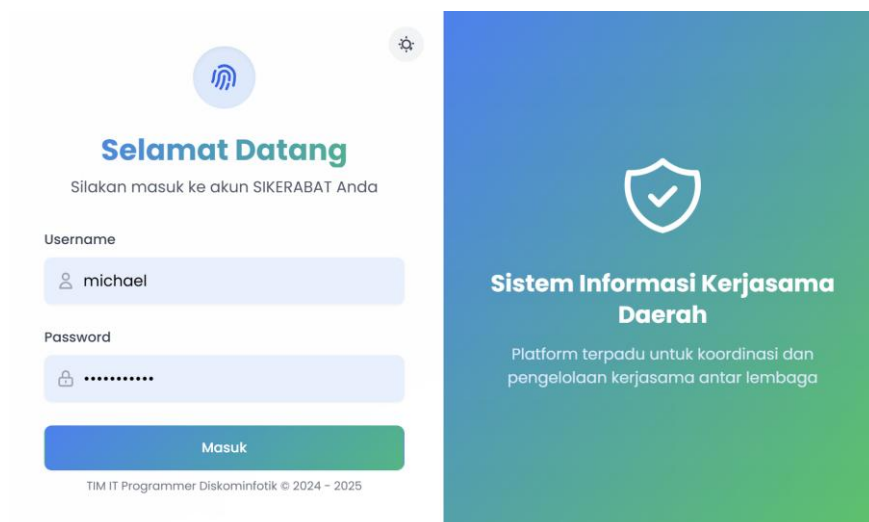
Halaman ini menampilkan informasi layanan yang ada pada aplikasi sikerabat pada Biro Pemerintahan Provinsi Sumatera Barat. Dan beberapa icon yang ada di bawah menampilkan data potensi kerjasama, mekanisme kerjasama, dokumen kerjasama, agenda, evaluasi dan capaian hasil kerjasama.



Gambar 5. Halaman utama aplikasi sikerabat

3.2 Halaman Login

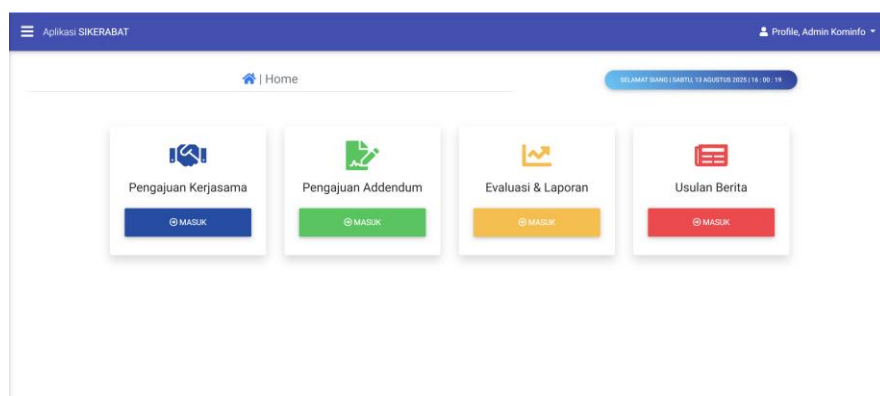
Halaman ini merupakan akses utama kedalam aplikasi sikerabat bagi opd atau instansi yang telah memiliki akun yang telah di tambahkan oleh admin biro pemerintahan.



Gambar 6. Halaman *login* aplikasi sikerabat

3.3 Halaman *Dashboard Login OPD*

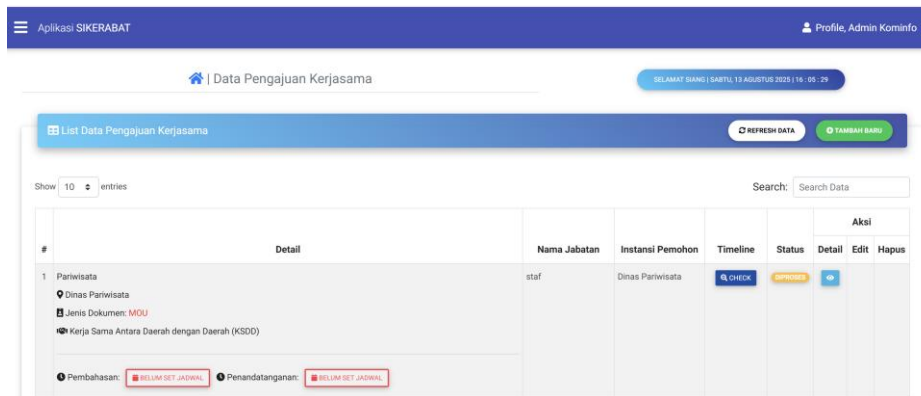
Pada halaman ini opd bisa melihat beberapa fitur ada pada aplikasi sikerabat yang terdiri dari pengajuan kerjasama, pengajuan addendum, evaluasi dan laporan, serta usulan berita.



Gambar 7. Halaman dashboard aplikasi sikerabat

3.4 Halaman List View Pengajuan Kerjasama

Pada halaman ini merupakan list *view* informasi kerjasama yang telah dilakukan pengajuan terutama informasi *timeline* untuk melihat sudah sampai mana surat tersebut berada.



Gambar 8. Halaman list data pengajuan kerjasama

3.5 Form Input Usulan Kerjasama

Halaman ini merupakan form pengentrian data usulan kerjasama yang dilakukan oleh opd atau instansi yang memiliki beberapa kolom entri yang terdiri dari kategori permohonan, jenis permohonan, tentang, nama pemohon, instansi pemohon, jabatan pemohon, jenis urusan, kolom khusus PKS, kolom khusus upload dokumen, dan upload dokumen.

Gambar 9. Halaman input usulan kerjasama

4. KESIMPULAN

Aplikasi **Sikerabat** terbukti sangat membantu OPD atau instansi dalam melakukan pengajuan kerjasama secara online. Dengan penerapan metode *Rapid Application Development (RAD)*, sistem ini dapat dikembangkan secara lebih cepat, iteratif, dan melibatkan pengguna secara aktif dalam proses perancangan. Hal ini meminimalkan kesalahan, meningkatkan fleksibilitas, serta memungkinkan perbaikan berkelanjutan berdasarkan masukan langsung dari pengguna. Selain itu, penerapan *framework CodeIgniter* dengan pendekatan *Model-View-Controller (MVC)* memberikan struktur pengembangan yang modular dan terorganisir, sehingga memudahkan proses pemeliharaan, pembaruan fitur, serta pengelolaan data dalam aplikasi. Model ini juga memisahkan logika bisnis, antarmuka pengguna, dan basis data secara jelas sehingga sistem lebih efisien dan terukur. Penggunaan *PostgreSQL* sebagai basis

data utama semakin memperkuat stabilitas dan performa aplikasi, terutama dalam mengelola data kerjasama yang cukup besar dan kompleks. Basis data ini mendukung transaksi secara andal, menjaga integritas data, serta mampu menangani kebutuhan operasional dengan baik. Secara keseluruhan, aplikasi Sikerabat mampu meningkatkan efisiensi birokrasi, mempercepat koordinasi antarinstansi, serta mendukung transparansi proses pengajuan kerjasama. Dengan adanya sistem ini, pengambilan keputusan dapat dilakukan lebih tepat karena berbasis data yang valid, terstruktur, dan mudah diakses.

DAFTAR PUSTAKA

- [1] S. Aswati, M. S. Ramadhan, A. U. Firmansyah, and K. Anwar, "STUDI ANALISIS MODEL RAPID APPLICATION DEVELOPMENT DALAM PENGEMBANGAN SISTEM INFORMASI STUDI ANALISIS MODEL RAPID APPLICATION DEVELOPMENT DALAM PENGEMBANGAN SISTEM INFORMASI 1," vol. 16, no. 2, p. 2017.
- [2] S. A. Sholikhatin, A. L. Munawaroh, and R. A. Ramadhan, "RESISTOR Journal | 131 Penerapan Metode RAD dan Framework Codeigniter Pada Web Keuangan Desa: Studi Kasus Desa Melung", [Online]. Available: <https://s.id/jurnalresistor>
- [3] A. I. Anshori, M. N. Falah, S. Aisah, and U. M. Wulandari, "Implementasi Metode Rapid Application Development (RAD) pada Sistem Informasi Perpustakaan Berbasis Web di SMK Nusantara 1 Comal," *RIGGS: Journal of Artificial Intelligence and Digital Business*, vol. 4, no. 3, pp. 1648–1654, Aug. 2025, doi: 10.31004/riggs.v4i3.2208.
- [4] H. Zul Fahmi *et al.*, "IMPLEMENTASI MVC DENGAN CODEIGNITER PADA SISTEM ADMIN PEGAWAI PERKANTORAN."
- [5] W. Muthia Kansha, "Analisis Perbandingan Struktur dan Performa Framework Codeigniter dan Laravel dalam Pengembangan Web Application".
- [6] D. Aji Bayu Prasetyo and Y. Alfa Susetyo, "Implementasi Information Schema Database Pada Postgre SQL Untuk Pembuatan Tabel Informasi Dengan Menggunakan Python Di PT XYZ," 2022. [Online]. Available: <http://jurnal.mdp.ac.id>
- [7] S. Informasi, P. Jasa, P. Berbasis, W. Pada, P. Karya, and S. Jaya, "Penerapan Metode RAD Dalam Pengembangan," *Remik: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 8, no. 3, 2024, doi: 10.33395/remik.v8i3.13944.
- [8] M. Ridwan, T. H. Sinaga, and M. Elsera, "PENERAPAN FRAMEWORK CODEIGNITER DALAM PERANCANGAN APLIKASI MANAJEMEN IURAN PERUMAHAN GRIYA MANDIRI," 2022.
- [9] S. Necula, "Exploring The Model-View-Controller (MVC) Architecture: A Broad Analysis of Market and Technological Applications," Apr. 29, 2024. doi: 10.20944/preprints202404.1860.v1.
- [10] G. Paolone, R. Paesani, M. Marinelli, and P. Di Felice, "Empirical Assessment of the Quality of MVC Web Applications Returned by xGenerator," 2021, doi: 10.3390/computers.
- [11] W. N. Suliyanti, "STUDI LITERATUR BASIS DATA SQL DAN NOSQL," vol. 8, no. 1, 2019.
- [12] P. DI SISTEM INFORMASI PEMBANGUNAN JALAN PERDESAAN KABUPATEN SERANG Widyawati, D. Ratnasari, and I. Gunawan, "02 JURNAL OF INNOVATION AND FUTURE TECHNOLOGY."
- [13] J. Lee *et al.*, "A Novel Data Sanitization Method Based on Dynamic Dataset Partition and Inspection Against Data Poisoning Attacks," *Electronics (Switzerland)*, vol. 14, no. 2, Jan. 2025, doi: 10.3390/electronics14020374.