

Implementasi dan Optimasi Sistem Manajemen Proyek dan Sumber Daya Manusia Berbasis Laravel Menggunakan Metode Scrum

Fadhil Raihan¹, Ramadhan Renaldy²

¹Program Studi Informatika, Universitas Persatuan Guru Republik Indonesia Semarang, Kota Semarang

²Program Studi Informatika, Universitas Persatuan Guru Republik Indonesia Semarang, Kota Semarang

*Email: eyhanfadi555@gmail.com

Abstract

The development of an internal information system using Laravel 10 and MySQL aimed to digitize project management and human resource processes. Key features included project creation, task assignment based on user roles, role-based access control, and real-time notifications to improve team collaboration. The development process covered requirement analysis, UI design, data processing, debugging, and system testing. Performance optimization was achieved through clean code principles, lazy loading, frontend asset compression, and efficient querying with Eloquent ORM. Enhancements such as refactoring Blade templates into partial components and implementing session handling improved system reliability. Additionally, the integration of GitLab with CI/CD pipelines supported an efficient and collaborative DevOps workflow. The result demonstrated that Laravel can deliver a secure, responsive, and scalable solution, with future development potential in online payment, API integration, sensitive configuration encryption, and comprehensive technical documentation.

Keywords: Laravel; Information System; MySQL; Project Management; CI/CD.

Abstrak

Pengembangan sistem informasi internal berbasis web menggunakan Laravel 10 dan MySQL bertujuan untuk mendigitalisasi proses manajemen proyek dan sumber daya manusia. Fitur utama yang diimplementasikan meliputi pembuatan proyek, penugasan tugas berdasarkan peran pengguna, kontrol akses berbasis *role*, serta notifikasi *real-time* untuk meningkatkan kolaborasi tim. Proses pengembangan mencakup analisis kebutuhan, perancangan antarmuka, pengolahan data, *debugging*, dan pengujian sistem. Optimasi performa dilakukan melalui penerapan prinsip *clean code*, *lazy loading*, kompresi aset *frontend*, serta efisiensi *query* menggunakan *Eloquent ORM*. Peningkatan keandalan sistem juga didukung dengan refaktor *Blade* template menjadi komponen parsial dan penerapan session handling. Selain itu, integrasi *GitLab* dengan *pipeline CI/CD* memperkuat praktik *DevOps* yang efisien dan kolaboratif. Hasil akhirnya menunjukkan bahwa Laravel mampu memberikan solusi yang aman, responsif, dan *scalable*, dengan potensi pengembangan ke depan seperti pembayaran daring, integrasi *API*, enkripsi konfigurasi sensitif, serta dokumentasi teknis yang komprehensif.

Kata kunci: Laravel; Sistem Informasi; MySQL; Manajemen Proyek; CI/CD.

1. Pendahuluan

Dalam pengelolaan proyek berskala kecil maupun besar, Efisiensi komunikasi tim dan pemantauan aktivitas menjadi faktor krusial dalam keberhasilan manajemen proyek. Namun, banyak organisasi masih menggunakan sistem manual atau terfragmentasi dalam pengelolaan proyek dan sumber daya manusia (SDM), yang menimbulkan kendala seperti duplikasi data, keterlambatan validasi, serta kurangnya transparansi. Kondisi ini semakin kompleks dalam lingkungan kerja hybrid seperti WFO dan WFH [2] [3]. Berbagai penelitian telah menawarkan solusi melalui sistem manajemen berbasis web menggunakan metode Waterfall atau Agile, dan memanfaatkan framework Laravel. Namun, kebanyakan

implementasi masih belum mengoptimalkan fitur penting seperti notifikasi real-time, dashboard status progres, atau integrasi CI/CD untuk efisiensi kolaboratif [3] [5].

Artikel ini mengusulkan pengembangan sistem manajemen proyek dan SDM berbasis Laravel 10 dengan pendekatan Scrum. Fokus utama diarahkan pada otomatisasi proses kerja, monitoring berbasis Kanban, pengelolaan hak akses berbasis peran, serta notifikasi real-time menggunakan Notify.js. Sistem juga didukung pipeline GitLab CI/CD untuk memastikan pengembangan berjalan cepat, aman, dan terstruktur. Tujuan dari penelitian ini adalah membangun sistem terintegrasi yang responsif terhadap kebutuhan kerja modern dan sekaligus memberikan kontribusi praktis maupun akademik dalam pengembangan sistem menggunakan Laravel dan metode Scrum.

2. Metode

Penelitian ini menggunakan pendekatan *Scrum*, yaitu kerangka kerja pengembangan perangkat lunak yang bersifat iteratif dan adaptif. *Scrum* memungkinkan pengembangan dilakukan secara bertahap melalui siklus bernama *Sprint*, yang pada proyek ini masing-masing berdurasi empat minggu. Proses diawali dengan penyusunan *Product Backlog*, yang berisi daftar kebutuhan sistem dan fitur inti. Item-item dalam *backlog* ditentukan berdasarkan hasil observasi internal perusahaan, diskusi bersama tim teknis, serta masukan dari stakeholder. Tiap item kemudian diprioritaskan dan dianalisis tingkat kompleksitasnya untuk dibagi ke dalam *Sprint* yang terstruktur [12].

2.1 Struktur Tim dan Proses Scrum

Pengembangan sistem dilakukan oleh tim kecil yang terdiri dari *Product Owner* (juga merangkap *Project Manager*), *Scrum Master*, *UI/UX Designer*, serta *Developer*. Setiap *Sprint* melibatkan empat tahapan utama: *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective*. Selama proses berlangsung, penulis mulai terlibat sejak *Sprint* ke-3, dan ikut berkontribusi langsung dalam pengembangan modul *dashboard*, fitur *notifikasi*, serta pengaturan profil pengguna. *Scrum* terbukti mendukung kolaborasi lintas peran serta adaptasi terhadap perubahan kebutuhan pengguna yang dinamis.

2.2 Backlog dan Rencana Sprint

Daftar *Product Backlog* berfungsi sebagai fondasi utama pengembangan. Beberapa fitur inti yang dikembangkan di antaranya adalah sistem autentikasi pengguna berbasis peran, manajemen proyek, manajemen tugas harian, *dashboard* interaktif, sistem cuti, hingga *notifikasi* berbasis Notify.js.

Tabel 1. Ringkasan Item Backlog dan Estimasi

No	Fitur Sistem	Deskripsi Fitur Utama	Estimasi (hari)
1	Autentikasi Pengguna	Login & logout berdasarkan role	5
2	Dashboard Utama	Visualisasi proyek & tugas terkini	9
3	Manajemen Proyek	Tambah, ubah, dan hapus data proyek	9
4	Manajemen Tugas	Penugasan & update progres tugas	13
5	Pengajuan Cuti	Sistem cuti dinamis dengan validasi admin	10
6	Notifikasi	Peringatan real-time via Notify.js	10
7	Pengaturan Profil	Update dan manajemen informasi pribadi pengguna	12

Setiap fitur dibagi ke dalam *Sprint* yang berbeda, disesuaikan dengan prioritas dan beban kerja tim. Estimasi waktu dihitung berdasarkan diskusi *Sprint Planning* menggunakan pendekatan story point.

2.3 Sprint Planning dan Distribusi Tugas

Salah satu Sprint paling produktif terjadi pada Sprint ke-2, yang fokus pada pembangunan fitur inti sistem. Perencanaan tugas dilakukan secara terstruktur dan melibatkan semua anggota tim.

Tabel 2. Sprint Planning Sprint 2

Fitur	Task Utama	Estimasi (Hari)
Manajemen Tugas	Desain UI, implementasi CRUD, testing	6
Manajemen User	Desain UI, CRUD pengguna, validasi input	5
Dashboard Proyek	Desain dan integrasi visual data progres	3

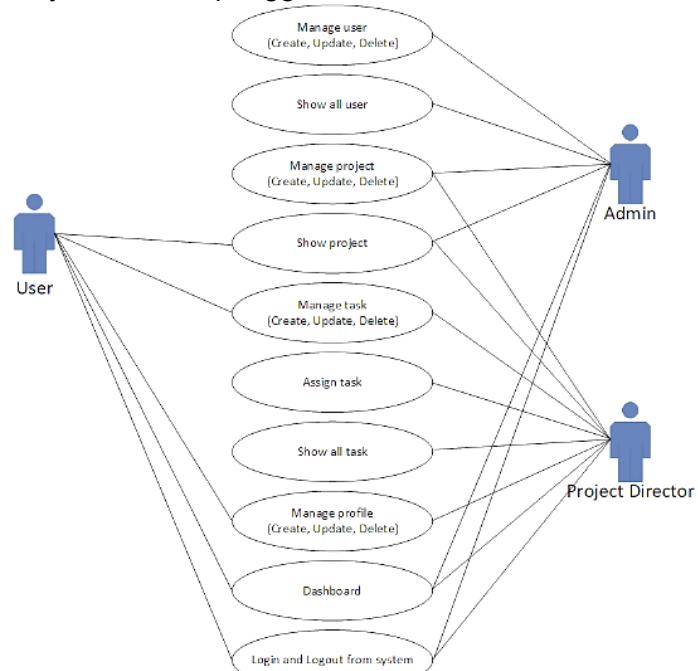
Total estimasi pekerjaan selama *Sprint* ke-2 mencapai 14 hari aktif. Tim juga menjalankan *Daily Scrum* untuk memastikan alur kerja tetap sesuai rencana dan hambatan dapat segera diatasi.

2.4 Perancangan Sistem

Perancangan sistem merupakan tahap krusial dalam proses pengembangan perangkat lunak karena menjadi dasar visualisasi terhadap bagaimana sistem akan bekerja, bagaimana setiap komponen saling berinteraksi, dan bagaimana alur data mengalir di dalamnya. Dalam proyek pengembangan sistem manajemen proyek dan sumber daya manusia (SDM) berbasis *web* ini, pendekatan yang digunakan adalah *Unified Modeling Language* (UML), yang menyediakan berbagai jenis diagram sebagai alat bantu visual untuk menggambarkan struktur serta perilaku sistem secara menyeluruh dan terstruktur [2] [11]. Adapun diagram yang digunakan dalam proses ini meliputi:

A. Use Case Diagram

Use Case Diagram berfungsi untuk memetakan hubungan antara aktor (pengguna sistem) dan fungsionalitas utama dari sistem. Dalam sistem ini, terdapat tiga aktor utama: Admin, *Project Director*, dan *User* (karyawan) [2] [11]. Gambar 1 menunjukkan representasi visual dari seluruh interaksi antara pengguna dan sistem. Dalam diagram tersebut, panah asosiasi antara aktor dan fungsi mencerminkan aksi yang dilakukan oleh masing-masing pengguna. Fungsi seperti *login*, *logout*, akses *dashboard*, pengajuan cuti, pengelolaan proyek, pengelolaan *user*, hingga pemberitahuan *notifikasi* merupakan bagian dari *use case* yang menggambarkan skenario nyata selama penggunaan sistem.

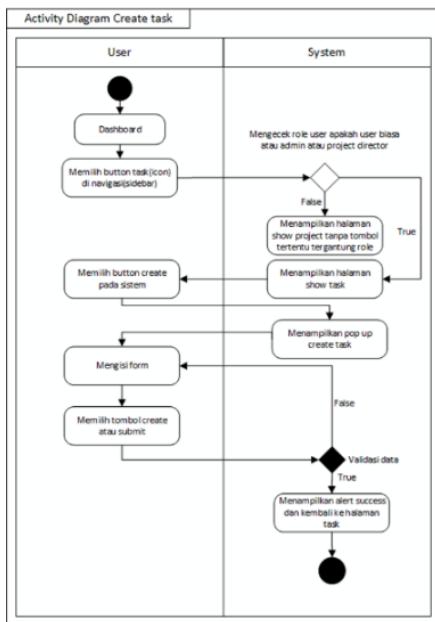


Gambar 1. Use Case Diagram.

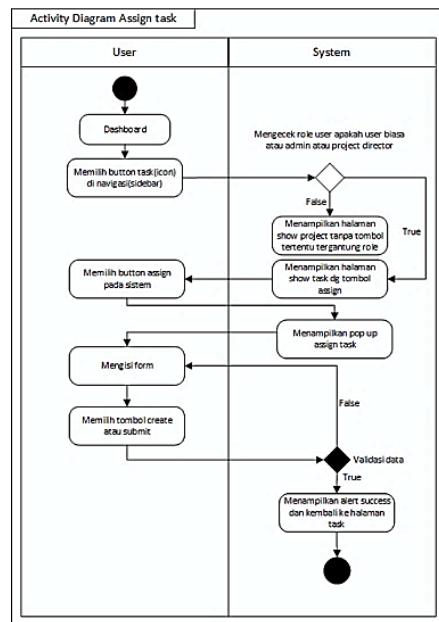
B. Activity Diagram

Digunakan untuk memetakan alur kerja (*workflow*) dari beberapa fitur utama dalam sistem. Diagram ini memvisualisasikan langkah-langkah prosedural pengguna dalam

menjalankan fungsi-fungsi sistem, sekaligus memperlihatkan kondisi percabangan logika dalam sistem [2] [11].



Gambar 2. Activity Diagram Create Task.

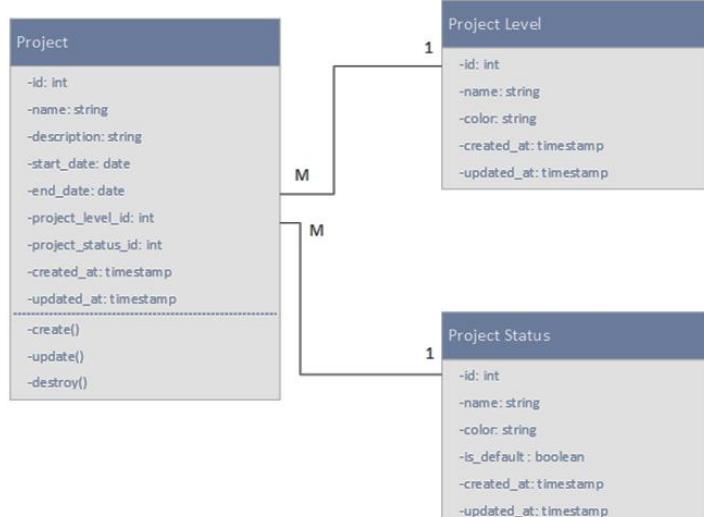


Gambar 3. Activity Diagram Assign Task.

Pada Gambar 2. ini menunjukkan alur proses *create* atau pembuatan data *task* yang bisa dilakukan oleh seluruh user dimulai dari halaman *show all task*. Data ditambahkan dengan memilih tombol *create task* kemudian mengisi form dan memilih tombol *create* untuk menambahkan data. Jika data berhasil ditambahkan akan muncul *alert success* dan Kembali ke halaman *show all task*. Pada Gambar 3 ini menunjukkan alur proses *assign* atau pemindahan data *task* yang bisa dilakukan oleh seluruh *user* dimulai dari halaman *show all task*. Data dapat dipindahkan dengan memilih tombol *assign task* kemudian memilih *task* yang akan dipindah lalu mengisi form dan melakukan permindahan kemudian memilih tombol *assign* untuk merubah data. Jika data berhasil diubah akan muncul *alert success* dan kembali ke halaman *show all task*.

C. Class Diagram

Class Diagram digunakan untuk menggambarkan struktur statis sistem, termasuk entitas data utama, atribut, serta relasi antar kelas. Dalam sistem manajemen proyek ini, class diagram dirancang untuk mencerminkan arsitektur basis data dan modularisasi logika bisnis berbasis pendekatan *MVC (Model–View–Controller)* [5].



Gambar 1. Class Diagram Project

Pada Gambar 4 ini menunjukkan merepresentasikan struktur data *class diagram project* yang berisi 3 tabel yang dimulai dari tabel *Project*, *Project Level*, dan *Project Status*. Tabel ini memiliki fungsi yang saling berhubungan yaitu *Create Read Update Delete (Crud)*.

3. Hasil dan Pembahasan

3.1 Penyajian Hasil

Sistem manajemen proyek dan sumber daya manusia (SDM) yang dikembangkan dalam proyek magang ini berhasil membentuk sebuah platform berbasis web yang terintegrasi. Sistem ini dirancang untuk mendukung kolaborasi tim, automasi tugas administratif, serta visibilitas aktivitas proyek secara *real-time*. Beberapa modul utama yang berhasil diimplementasikan, antara lain:

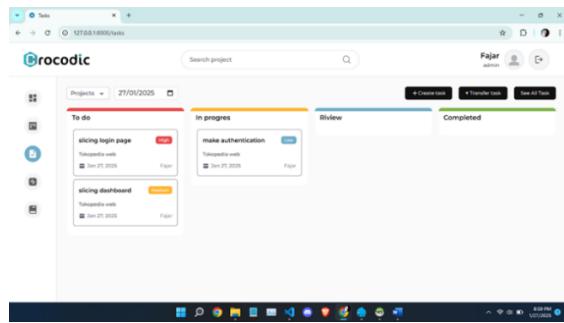


Gambar 5. Halaman Login.

Project Name	Start Date	Deadline	Project Director	Project Level	Project Status
Web 1-Awening SMP	10 February 2025	24 February 2025	Alfan	High	Not Started
E-Store	01 February 2025	18 March 2025	Rico	Medium	Not Started
Seni Web	26 February 2025	12 March 2025	Rico	Medium	Not Started
E-Tikang	04 March 2025	10 March 2025	Rico	Medium	Not Started
E-Keuangan	24 February 2025	31 March 2025	Rico	Medium	Not Started
jurnal	06 March 2025	23 March 2025	Rico	Medium	Not Started
coba alert	03 March 2025	15 March 2025	Rico	Medium	Not Started
coba coba	03 March 2025	15 March 2025	Rico	Medium	Not Started
coba aje	24 February 2025	01 March 2025	Alfan	Medium	Not Started
Re	01 March 2025	01 April 2025	Rico	Medium	Not Started

Gambar 6. Halaman Show Project.

Modul Autentifikasi Pengguna Berbasis Peran (*Role-Based Access Control*). Pada gambar 5 menunjukkan saat pengguna memasukkan nama dan kata sandi pada halaman ini, sistem akan memverifikasi informasi yang diberikan. Jika data yang dimasukkan sesuai, pengguna akan segera dialihkan ke halaman *dashboard*, tempat berbagai informasi dan fitur utama tersedia sesuai dengan kebutuhan aksesnya. Modul Manajemen Proyek, Pada gambar 6 menampilkan halaman utama daftar seluruh proyek yang telah ditambahkan ke sistem. Sebagian pengguna hanya memiliki akses untuk melihat informasi proyek yang tersedia tanpa dapat melakukan perubahan apa pun, sedangkan pengguna tertentu, seperti admin atau manajer proyek, memiliki hak untuk menambahkan dan memperbarui data proyek.

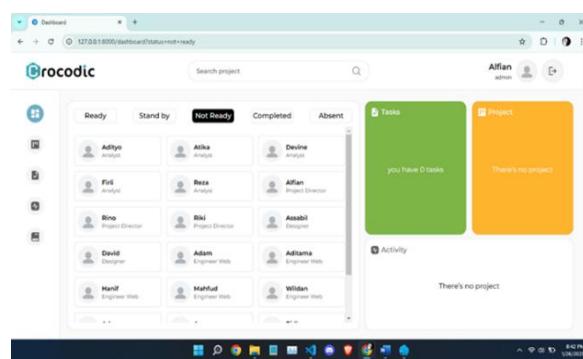


Gambar 7. Halaman Show Task.

Name Request	Kategori Cuti	Jumlah Hari Cuti	Tanggal Mulai	Tanggal Akhir	Memberi Laporan	Daftar Pihak
1 Admin	Cuti normal	5 Hari	21 Februari 2025	26 Februari 2025	Ya	Ya
2 Admin	Cuti normal	5 Hari	21 Februari 2025	26 Februari 2025	Ya	Ya
3 Rico	Cuti	6 Hari	18 March 2025	24 March 2025	Ya	Ya
4 Firdi Halim	Cuti non karyaw	30 Hari	10 March 2025	10 April 2025	Ya	Ya
5 Rico	Cuti normal	21 Hari	07 March 2025	21 Maret 2025	Ya	Ya
6 Firdi	Cuti karyaw	21 Hari	10 March 2025	08 April 2025	Ya	Ya
7 Admin	Cuti normal	7 Hari	19 Februari 2025	26 Februari 2025	Ya	Tidak
8 Firdi Halim	Cuti non karyaw	10 Hari	20 Februari 2025	29 Februari 2025	Ya	Ya
9 Admin	Cuti karyaw	14 Hari	20 Februari 2025	06 April 2025	Ya	Ya

Gambar 8. Halaman Leave Submision

Modul Manajemen Tugas, Pada gambar 7 menunjukkan halaman Daftar Tugas dengan tampilan *kanban*. Di halaman ini, pengguna bisa melihat detail tugas seperti nama, proyek, penanggung jawab, tingkat kesulitan, status, dan tanggal pembuatan. Fitur Pengajuan dan Validasi Cuti, Pada gambar 8 menunjukkan data pengajuan cuti, memiliki fitur yang disediakan agar karyawan dapat mengajukan permohonan cuti secara mandiri dan terdokumentasi secara sistematis. Melalui halaman ini, pengguna dapat memilih jenis cuti, menentukan tanggal mulai dan selesai cuti, serta menyertakan alasan atau catatan tambahan. Setelah formulir diajukan, sistem akan secara otomatis mengirimkan notifikasi kepada manajer proyek atau admin untuk melakukan proses validasi



Gambar 9. Dashboard

Dashboard Interaktif Berbasis Status Tugas, Pada gambar 9 menunjukkan *Dashboard* interaktif ini dirancang untuk menyajikan informasi menyeluruh terkait keterlibatan dan status aktivitas sumber daya manusia (SDM) dalam proyek. Melalui pembagian status yang jelas, manajemen dapat dengan mudah melakukan pemantauan dan pengambilan keputusan berbasis data aktual. Implementasi kelima modul ini membentuk fondasi utama dari sistem yang dibangun, sekaligus menjawab kebutuhan akan efisiensi, transparansi, dan akurasi dalam pengelolaan proyek serta SDM di lingkungan kerja modern. Sistem ini telah diuji dan menunjukkan performa yang stabil, baik dalam lingkungan kerja tatap muka (*Work from Office*) maupun jarak jauh (*Work from Home*). Efisiensi sistem ditunjukkan dari hasil uji performa berikut:

3.2 Pembahasan

Berdasarkan hasil implementasi, sistem yang dikembangkan berhasil menjawab kebutuhan pengguna dalam mengelola proyek dan SDM secara terintegrasi. Penggunaan pendekatan modular dan metodologi Scrum memungkinkan proses pengembangan yang adaptif terhadap masukan stakeholder serta mendukung pembaruan sistem secara berkelanjutan.

Tabel 2. Perbandingan kinerja sistem sebelum dan sesudah implementasi.

Metrik	Sebelum Implementasi	Setelah Implementasi
Waktu koordinasi tugas	2–3 hari	< 1 jam
Akurasi pelaporan progres	± 68%	> 95%
Waktu muat dashboard	± 4,2 detik	± 1,8 detik
Tingkat respons notifikasi	± 3 detik	Real-time (<1 detik)

Implementasi *Role-Based Access Control (RBAC)* membantu mengurangi kesalahan akses data oleh pengguna yang tidak berwenang. Sementara itu, tampilan task board secara visual mempermudah anggota tim dalam melacak tugas dan progres, menjadikan sistem lebih komunikatif dan kolaboratif. *Notifikasi* berbasis *Notify.js* berperan besar dalam mempercepat alur informasi antar pengguna. Setiap perubahan data strategis langsung diberitahukan, sehingga mempercepat proses pengambilan keputusan. Selain itu, dashboard berbasis status memungkinkan manajer proyek melihat keterlibatan anggota tanpa harus membuka tiap detail proyek satu per satu.

4. Kesimpulan

Pengembangan sistem manajemen proyek dan sumber daya manusia (SDM) berbasis web dalam proyek ini telah berhasil diwujudkan melalui penerapan framework Laravel dan metode Scrum. Proses pengembangan dilakukan secara iteratif dalam tiga siklus Sprint yang efektif, kolaboratif, dan adaptif terhadap perubahan kebutuhan. Sistem yang dihasilkan terbukti mampu menjawab tantangan dalam pengelolaan proyek dan SDM, khususnya dalam hal integrasi fitur, efisiensi koordinasi, serta transparansi pelaporan. Dengan menggabungkan autentikasi berbasis peran, manajemen proyek dan tugas, pengajuan cuti digital, dashboard interaktif, serta notifikasi real-time, sistem ini membentuk sebuah platform yang responsif terhadap dinamika kerja modern, baik untuk lingkungan WFO maupun WFH. Penerapan

Laravel memberikan kemudahan dalam pengelolaan basis data dan pengembangan antarmuka, sementara pendekatan Scrum memungkinkan tim pengembang menjaga arah kerja dan kualitas fitur di setiap iterasinya.

Secara keseluruhan, sistem ini berkontribusi dalam meningkatkan akurasi operasional, mempercepat alur kerja tim, serta mendukung pengambilan keputusan yang lebih cepat dan berbasis data. Keberhasilan implementasi ini menunjukkan potensi besar pemanfaatan teknologi berbasis Laravel dan metode Scrum dalam mendukung transformasi digital yang terstruktur dan berkelanjutan di lingkungan kerja profesional.

5. Referensi

- [1] Ralahallo, Ralahallo, F. N., Jaya, F. H., & Tukimun. (2024). *Manajemen proyek*. Sulur Pustaka
- [2] D. H. Mustofa, "Analisis Sistem dan Pembuatan Aplikasi Manajemen Proyek Berbasis Web di PT Taman Media Indonesia Sub Crocodic," Laporan Kerja Praktik, Program Studi Teknik Informatika, Fakultas Teknik dan Informatika, Universitas AKI, Semarang, Indonesia, 2024.
- [3] M. P. Rasidin and H. Ma'sum, "ANALISIS SISTEM INFORMASI PENDATAAN KEPENDUDUKAN KAMPUNG ARENG DENGAN MENGGUNAKAN LARAVEL," vol. 9, no. 1, 2024.
- [4] Suradi, S. (2022). *Pemodelan sistem (sebuah pengantar)*. CV. Tohar Media.
- [5] Eka, A. (2025). *Laporan Magang: PT Taman Media Indonesia – Crocodic Studio*. Laporan tidak dipublikasikan. Program Studi Informatika, Universitas AKI Semarang.
- [6] Robin. Nixon, *Learning PHP, MySQL & JavaScript, 5th Edition*. O'Reilly Media, Inc., 2018.
- [7] eSkripsi Universitas Semarang. (n.d.). *Contoh Laporan Skripsi G.231.17.0089*.
- [8] Sumirat, L. P., et al. (2023). *Dasar-dasar rekayasa perangkat lunak* (Edisi pertama). Madza Media.
- [9] Duckett, J. (2011). *HTML and CSS: Design and build websites*. Wiley.
- [10] F. Zampetti, S. Geremia, G. Bavota, and M. Di Penta, "CI/CD Pipelines Evolution and Restructuring: A Qualitative and Quantitative Study." [Online]. Available: <https://madnight.github.io/githut/>
- [12] Anindita, P. P. (2024). Laporan Magang: PT Campus Digital Indonesia. Laporan tidak dipublikasikan. Program Studi Informatika, Universitas PGRI Semarang.
- [13] Trisanto, D., Rismawati, N., Izzatillah, M., & Mulya, M. F. (2023). Analisis dan perancangan sistem informasi e-learning menggunakan metode Scrum berbasis framework laravel dan bootstrap. *Journal of Information System, Applied, Management, Accounting and Research*, 7(2), 225-232.