

Transformasi Ekosistem Digital melalui Implementasi *Super App* untuk Meningkatkan Administrasi Kantor

Y D Setiawan¹, O Karnalim²

¹⁻²Program Studi Teknik Informatika, Universitas Kristen Maranatha

E-mail: 2172003@maranatha.ac.id¹, oscar.karnalim@it.maranatha.edu²

Abstrak. Manajemen kantor sangat penting untuk kelancaran operasional organisasi. Untuk mengatasi kompleksitas peran ini, penelitian ini telah mengembangkan aplikasi super yang mengintegrasikan berbagai fitur administrasi kantor. Dibangun di atas kerangka kerja Laravel, aplikasi super ini memastikan stabilitas, keamanan, dan skalabilitas untuk integrasi fitur-fitur yang kompleks. Proses desain dan validasi prototipe melibatkan pengujian pengguna dan umpan balik dari para pemangku kepentingan, sehingga menghasilkan solusi inovatif yang divalidasi oleh para pemangku kepentingan yang secara signifikan memberikan kontribusi terhadap manajemen administrasi perkantoran. Solusi ini meningkatkan efisiensi dan efektivitas administrasi perkantoran, membuka jalan untuk aplikasi serupa di masa depan.

Kata kunci: aplikasi super; integrasi kerangka kerja Laravel; efisiensi administrasi perkantoran; peningkatan produktivitas; validasi pemangku kepentingan

Abstract. Office management is vital for seamless organizational operations. To tackle the complexities of this role, we've developed a comprehensive super application that integrates various office administration features. Built on the Laravel framework, this super application ensures stability, security, and scalability for seamless integration of complex features. Implementing this application is expected to reduce administrative task time and increase productivity. The design and prototype validation process involved user testing and feedback from stakeholders, resulting in an innovative, stakeholder-validated solution that significantly contributes to office administration management. This solution enhances office administration efficiency and effectiveness, paving the way for future similar applications.

Keywords: comprehensive super application; Laravel framework integration; office administration efficiency; productivity enhancement; stakeholder-validated solution

1. Pendahuluan

Efisiensi administrasi kantor merupakan peran utama bagi operasi organisasi yang lancar dan produktif. Dalam era globalisasi dan teknologi yang terus berkembang, tantangan dalam mengelola administrasi kantor semakin kompleks [1]. Untuk mengatasi kompleksitas peran ini, perlu adanya solusi yang inovatif dan terintegrasi secara menyeluruh. Dalam konteks ini, pengembangan aplikasi super komprehensif telah menjadi fokus penelitian dan pengembangan [2]. Aplikasi ini, yang dibangun di atas kerangka kerja Laravel [3], menjanjikan stabilitas, keamanan, dan skalabilitas untuk integrasi fitur-fitur administrasi kantor yang kompleks.

Pengembangan aplikasi super untuk administrasi kantor tidak terlepas dari beberapa permasalahan kunci [4]. Pertama, bagaimana cara terbaik untuk mengintegrasikan berbagai fitur administrasi kantor menjadi satu kesatuan yang kohesif? Kedua, bagaimana memastikan bahwa aplikasi tersebut tetap stabil, aman, dan dapat disesuaikan dengan kebutuhan pengguna di masa mendatang? Bagaimana mengukur dampak implementasi aplikasi ini terhadap efisiensi dan produktivitas dalam tugas-tugas administratif? Terakhir, bagaimana memperhitungkan masukan dan kebutuhan dari berbagai pihak terkait (*stakeholder*) dalam proses pengembangan aplikasi ini?

Aplikasi super memberikan pengalaman yang menyeluruh bagi pengguna dalam mengelola administrasi kantor. Karyawan dapat dengan mudah menggunakan fitur presensi kehadiran untuk mencatat aktivitas harian [5]. Selanjutnya, aplikasi ini memfasilitasi pengajuan cuti karyawan, pencatatan serta peminjaman inventaris dengan proses yang efisien. Aplikasi super juga menyediakan sistem manajemen proyek [6] yang memungkinkan tim untuk berkolaborasi secara lebih terstruktur dan fitur pembuatan resume karyawan secara otomatis memberikan kemudahan dalam memonitor dan melacak pengembangan karier setiap individu. Melalui *user-friendly dashboard*, karyawan dan manajemen dapat mengakses informasi secara *real-time*, meningkatkan transparansi, dan memastikan akuntabilitas dalam menjalankan tugas sehari-hari serta memantau kemajuan proyek secara efektif.

Dalam konteks pengembangan aplikasi, kerangka kerja Laravel telah menjadi pilihan yang populer karena kemampuannya dalam menyediakan struktur yang kokoh dan skalabilitas yang tinggi [7]. Dengan fitur-fitur bawaan seperti sistem manajemen basis data yang kuat dan kemampuan untuk membangun komunikasi dalam sistem dengan mudah, Laravel memungkinkan pengembang untuk fokus pada logika bisnis inti tanpa harus menghabiskan waktu berlebihan untuk konfigurasi teknis. Berbagai studi telah menunjukkan bahwa penggunaan kerangka kerja Laravel dapat mempercepat proses pengembangan dan meningkatkan kualitas aplikasi [8]. Melalui komunitas yang besar dan aktif di sekitar Laravel menyediakan sumber daya dan dukungan yang berkelanjutan, termasuk dokumentasi yang kaya dan berbagai paket ekstensi, yang menjadikannya pilihan yang menarik bagi pengembang di berbagai tingkat keahlian.

Evaluasi terhadap dampak implementasi aplikasi administrasi terhadap efisiensi dan produktivitas dalam tugas-tugas administratif merupakan aspek penting dalam mengukur keberhasilan aplikasi tersebut [9]. Partisipasi aktif dari berbagai pihak terkait (*stakeholder*) dalam proses pengembangan aplikasi ini adalah kunci untuk memastikan bahwa aplikasi dapat mengakomodasi berbagai kebutuhan dan persyaratan yang ada. Selain itu, penggunaan teknologi terkini dan metodologi pengembangan yang terstruktur juga dapat berkontribusi dalam mencapai tujuan efisiensi dan efektivitas aplikasi administrasi. Perancangan aplikasi administrasi yang terintegrasi secara menyeluruh akan membantu meningkatkan transparansi, efisiensi, dan akuntabilitas dalam menjalankan tugas sehari-hari serta memantau kemajuan proyek secara efektif.

Memastikan bahwa aplikasi tetap stabil dan dapat disesuaikan dengan kebutuhan pengguna di masa mendatang adalah tantangan yang harus diatasi dalam pengembangan aplikasi administrasi kantor [10]. Oleh karena itu, pengembang perlu memperhatikan tidak hanya kebutuhan saat ini tetapi juga perkembangan teknologi dan perubahan kebutuhan pengguna untuk memastikan aplikasi tetap relevan dan berkelanjutan. Dalam konteks ini, pengujian reguler dan pemeliharaan yang berkala juga diperlukan untuk menjaga kualitas dan performa aplikasi secara keseluruhan. Pentingnya memastikan aplikasi super untuk administrasi kantor tidak hanya menawarkan fitur yang komprehensif tetapi juga memperhatikan aspek keamanan data, yang merupakan elemen krusial dalam pengembangan aplikasi berbasis web saat ini [11]. Keamanan data menjadi semakin penting seiring dengan meningkatnya volume dan sensitivitas informasi yang disimpan dan diakses oleh aplikasi tersebut. Salah satu aspek yang harus diperhatikan adalah penggunaan *JSON Web Tokens (JWT)* [12]. Dengan pendekatan yang holistik dan kolaboratif, pengembangan aplikasi super untuk administrasi kantor dapat menjadi solusi proaktif dalam meningkatkan efisiensi dan produktivitas di era digital.

Representational State Transfer [13] atau yang dikenal dengan singkatan REST merupakan gaya arsitektur perangkat lunak yang menjelaskan prinsip pembuatan layanan web yang terukur. REST

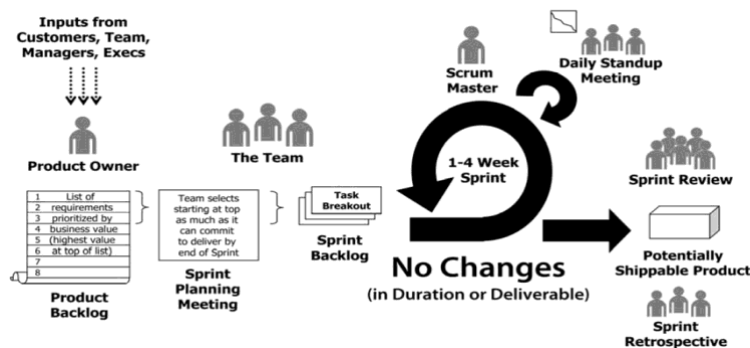
Application Programming Interface (REST API) memungkinkan berbagai sistem untuk berkomunikasi dan mengirim atau menerima data dengan cara yang sangat sederhana. *JavaScript Object Notation Web Token* (JSON Web Token atau JWT)[14] adalah standar terbuka (RFC 7519) yang secara ringkas mendefinisikan cara untuk mengirimkan informasi secara aman antar pihak sebagai objek JSON.

Metodologi *Agile* [15] adalah metode pengembangan perangkat lunak yang berfokus pada kolaborasi, komunikasi, dan iterasi. *Agile* memecah proyek menjadi siklus pendek (disebut *sprint*) yang memungkinkan tim beradaptasi dengan cepat terhadap perubahan dan umpan balik. Metodologi *agile* yang umum mencakup *Scrum*, *Kanban*, *Lean*, dan *Extreme Programming* (XP)[16].

Pengujian API merupakan tahap krusial dalam pengembangan perangkat lunak yang bertujuan untuk memeriksa apakah API berfungsi sesuai spesifikasinya, termasuk aspek fungsionalitas, keandalan, kinerja, dan keamanan [17]. Postman merupakan alat uji API yang menyederhanakan proses pembuatan, pengelolaan, dan pengujian permintaan HTTP. Postman mendukung simulasi kondisi lingkungan dan pemantauan kinerja API.

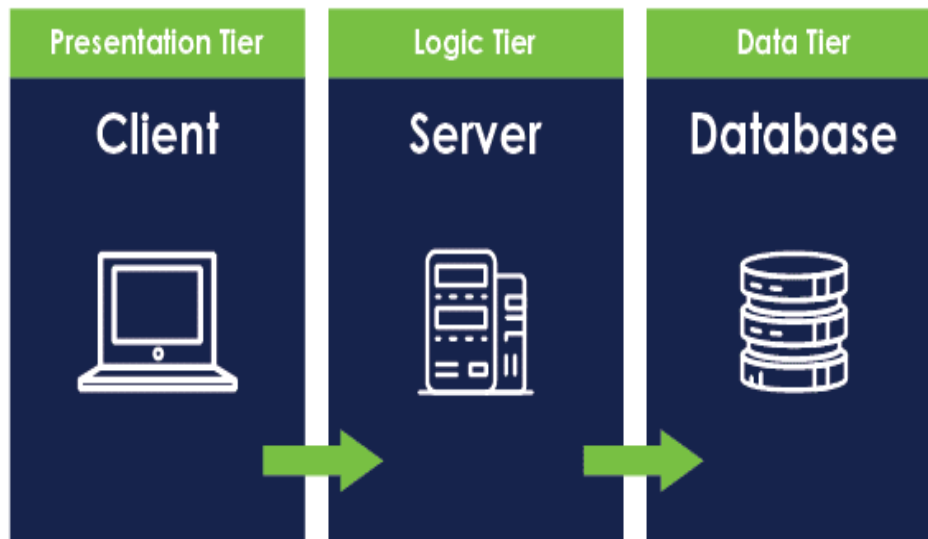
2. Metode

Penelitian ini mengadopsi pendekatan pengembangan perangkat lunak yang menggabungkan prinsip-prinsip *Agile* dengan metodologi *The Majestic Monolith* untuk membangun aplikasi super. Desain ini dipilih karena kesederhanaannya dalam pengembangan, pemeliharaan, dan pemahaman kode, yang sesuai dengan skala proyek ini. Langkah-langkah utama dalam penelitian ini termasuk analisis kebutuhan, desain konseptual, pengembangan prototipe, pengujian pengguna, dan validasi pemangku kepentingan. Metodologi *Agile*, termasuk *Scrum*, menawarkan pendekatan iteratif dan inkremental dalam pengembangan perangkat lunak. Gambar 1 menunjukkan metodologi *Scrum*, di mana pengembangan proyek dibagi menjadi iterasi terpisah yang disebut "*sprint*". Setiap *sprint* biasanya berlangsung selama satu hingga empat minggu, dengan tujuan pengiriman potongan fungsionalitas yang dapat digunakan kepada pengguna akhir.



Gambar 1. Visualisasi Proses Scrum (Agile)

Proses penelitian berikutnya adalah merancang sebuah arsitektur yang dapat menjembatani aplikasi yang akan dibuat. Aplikasi yang dibuat akan menggunakan *three-tier architecture* seperti yang ditunjukkan pada Gambar 2. *Three-tier architecture* merupakan sebuah arsitektur yang membagi antara presentation tier, *application layer*, serta *database tier*. *Presentation tier* akan dibuat menggunakan basis *website*. Pertimbangan tersebut didasarkan bahwa sebagian besar karyawan menggunakan komputer kantor. *Application tier* akan dibuat menggunakan kerangka kerja Laravel. Seluruh informasi dari aplikasi web akan diproses oleh *tier* ini. Laravel dipilih karena merupakan kerangka kerja yang cukup stabil disertai proses pengembangan yang cepat. Proses komunikasi antara *presentation tier* dengan *application tier* menggunakan *Application Programming Interface*(API). MySQL dipilih sebagai *database tier* memiliki struktur yang baik serta dapat melakukan pemrosesan data dengan cepat.

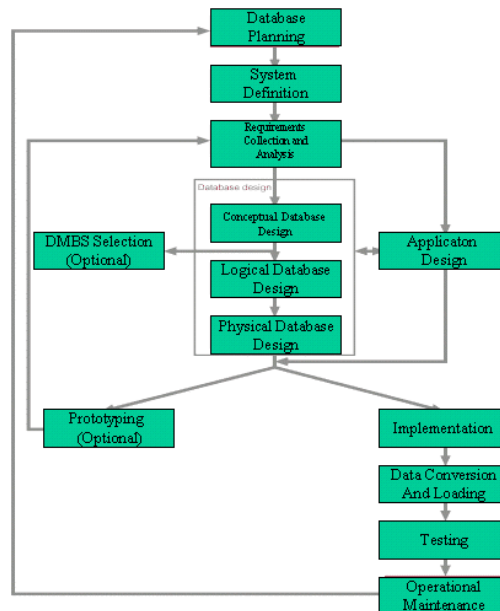


Gambar 2. Rancangan *Three-Tier Architecture* Aplikasi Super

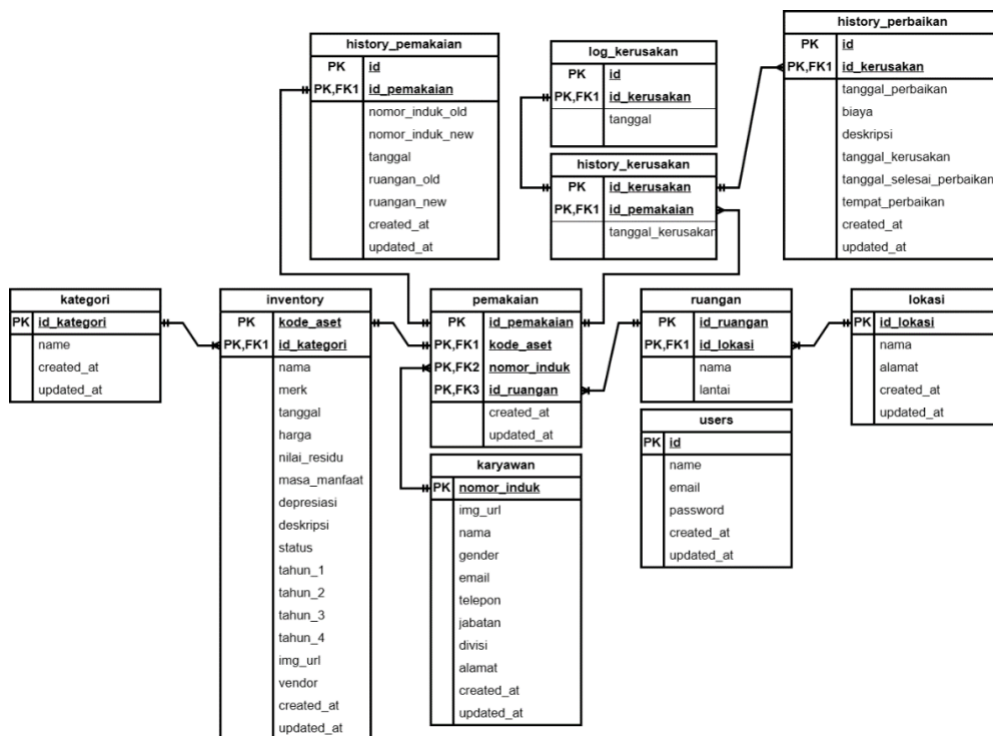
3. Hasil dan Pembahasan

3.1. Konsep Perancangan Basis Data

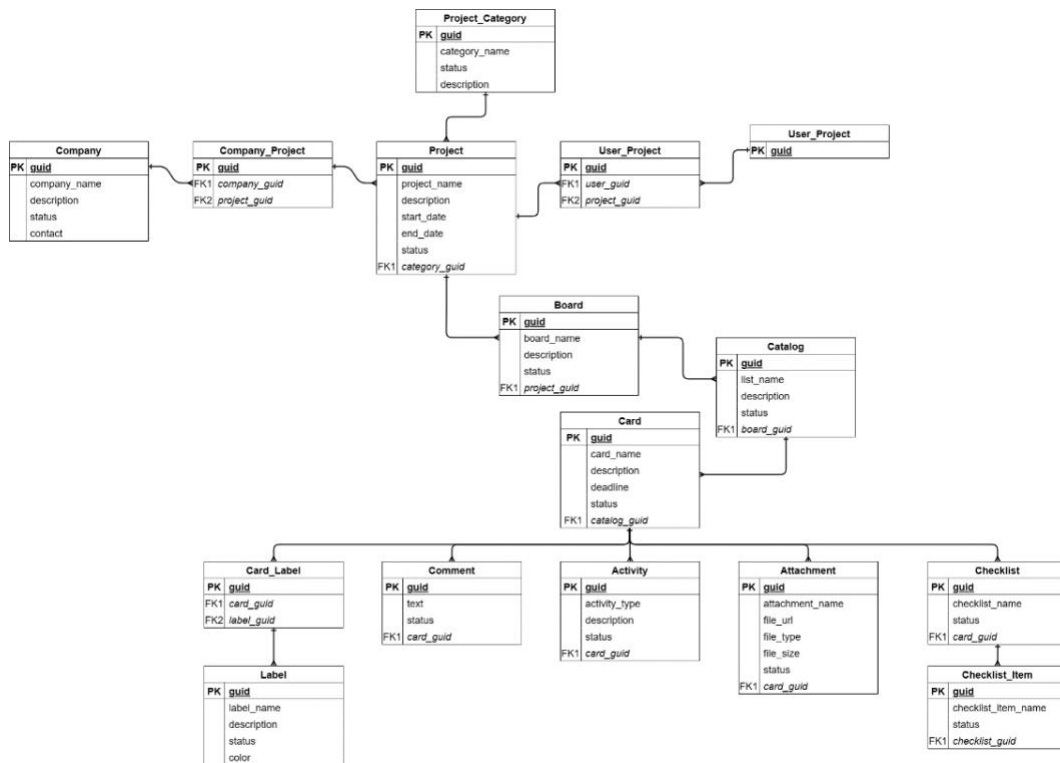
Pada tahap ini, dilakukan perancangan basis data yang menjadi hal utama dari aplikasi super. Konsep perancangan basis data ini mencakup struktur, hubungan antar-tabel, serta pemodelan entitas dan atribut. Basis data dibangun dengan mempertimbangkan kebutuhan akan keamanan, kecepatan akses, dan skalabilitas. Struktur basis data yang baik akan memastikan efisiensi dalam penyimpanan dan pengambilan data, serta memfasilitasi integrasi antarmuka pengguna dengan *backend* aplikasi. Metode pengembangan ERD saat ini adalah dengan menggunakan DBSDLC (*Database Systems Development Lifecycle*). Dalam DBSDLC yang ditunjukkan pada Gambar 3, terdapat beberapa tahapan yang harus diselesaikan untuk dapat mengembangkan basis data yang sesuai dengan kebutuhan. Dimulai dengan perencanaan basis data, kemudian definisikan sistem, cari tahu kebutuhan pengguna sistem. Basis data ini dirancang untuk mendukung semua fitur administrasi kantor yang terintegrasi dalam aplikasi, mulai dari manajemen inventaris (Gambar 4), sistem manajemen proyek (Gambar 5), dan sistem pembuatan CV otomatis (Gambar 6). Dengan implementasi konsep perancangan basis data yang matang, aplikasi dapat beroperasi dengan stabil, menyediakan akses data yang cepat dan akurat, serta mendukung pengembangan fitur-fitur tambahan di masa depan. Selain itu, basis data yang dirancang dengan baik juga memberikan keamanan yang tinggi terhadap informasi sensitif yang disimpan. Dengan adanya landasan yang kuat ini, diharapkan aplikasi dapat terus berkembang dan memberikan manfaat yang signifikan bagi pengguna dan organisasi di masa mendatang. Berikut diagram relasi pada sistem yang dibuat:



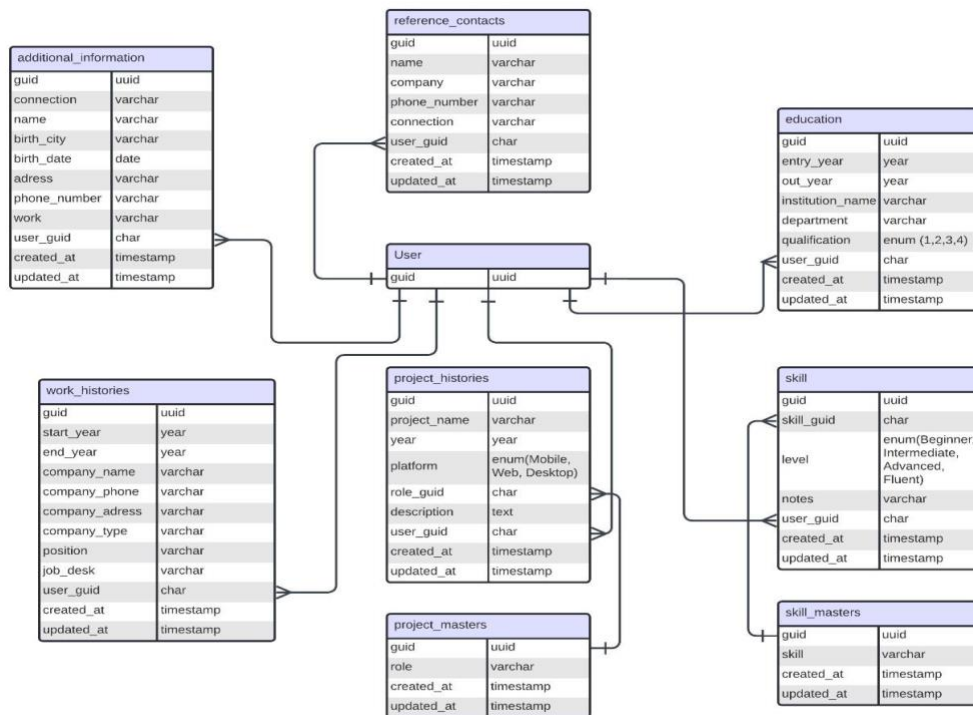
Gambar 3. Lifecycle Pengembangan Sistem Basis Data, Connolly and Begg



Gambar 4. Diagram Relasi Entitas Sistem Pengelolaan Inventaris



Gambar 5. Diagram Relasi Entitas Sistem Manajemen Proyek

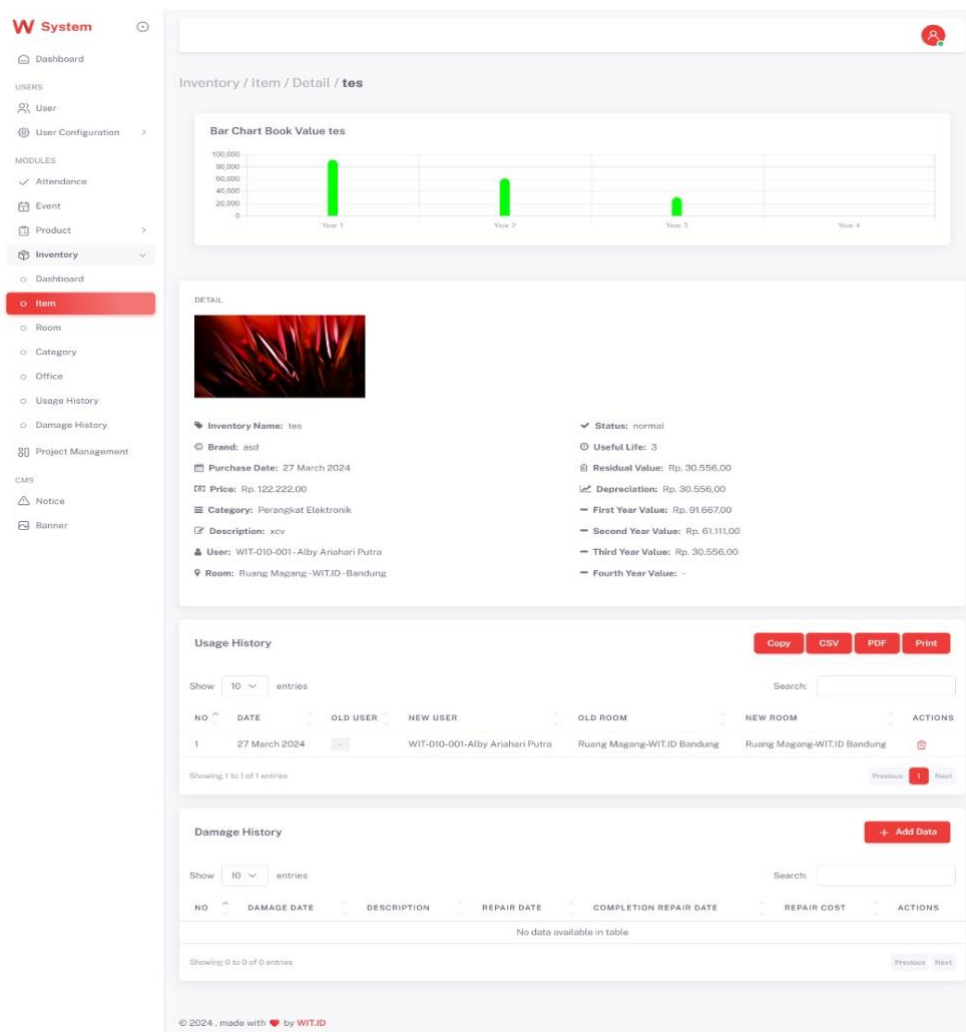


Gambar 6. Diagram Relasi Entitas Sistem Pembuatan CV Otomatis

3.2. Hasil Implementasi (User Interface)

Hasil implementasi mencakup antarmuka pengguna (*user interface*) dari aplikasi super untuk administrasi kantor. Antarmuka pengguna ini dirancang dengan memperhatikan prinsip-prinsip desain yang mencakup kegunaan (*usability*), keindahan (*aesthetics*), dan keterbacaan (*readability*). Melalui antarmuka pengguna yang *user-friendly*, karyawan dan manajemen dapat dengan mudah mengakses berbagai fitur aplikasi, melacak aktivitas, mengelola proyek, serta mengakses informasi secara *real-time*.

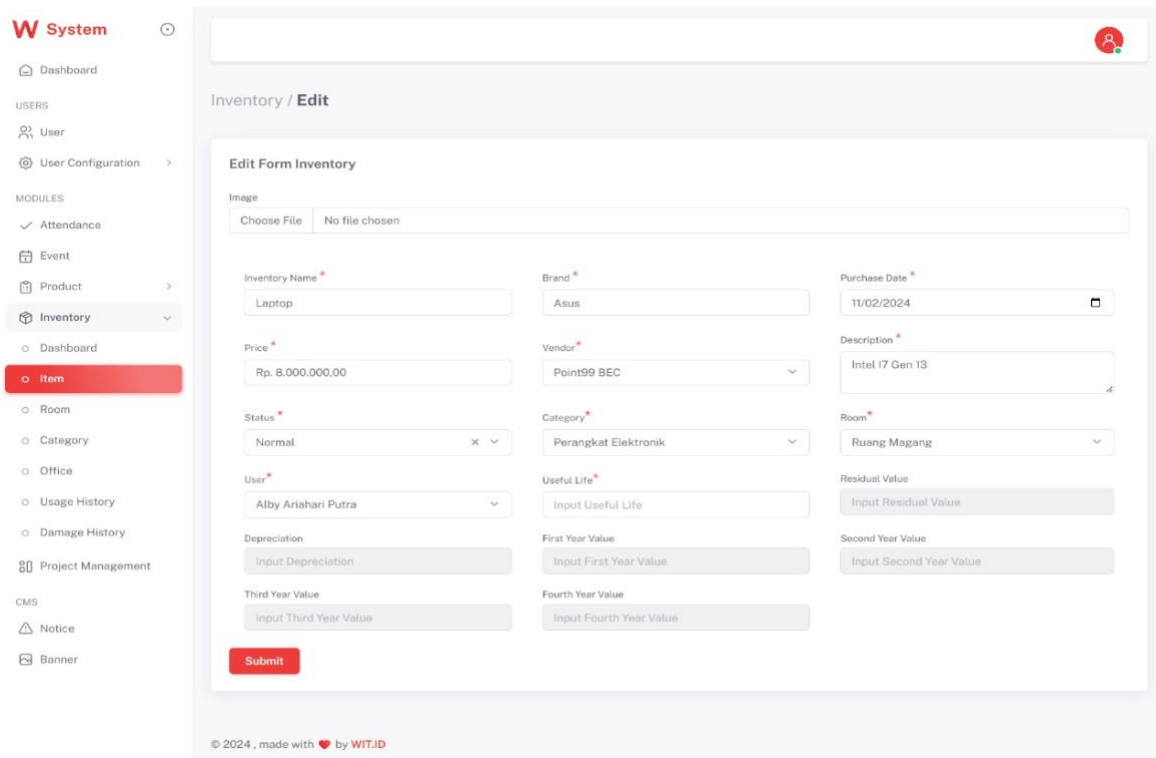
Aplikasi ini menampilkan antarmuka manajemen inventaris kantor (Gambar 7, Gambar 8, dan Gambar 9), yang memungkinkan pengguna untuk mengelola inventaris dengan efisien melalui tampilan yang intuitif dan informatif, memudahkan pemeriksaan ketersediaan dan pelacakan penggunaan barang. Gambar 7 memperlihatkan antarmuka untuk manajemen inventaris kantor yang mempermudah pengguna dalam mengelola inventaris.



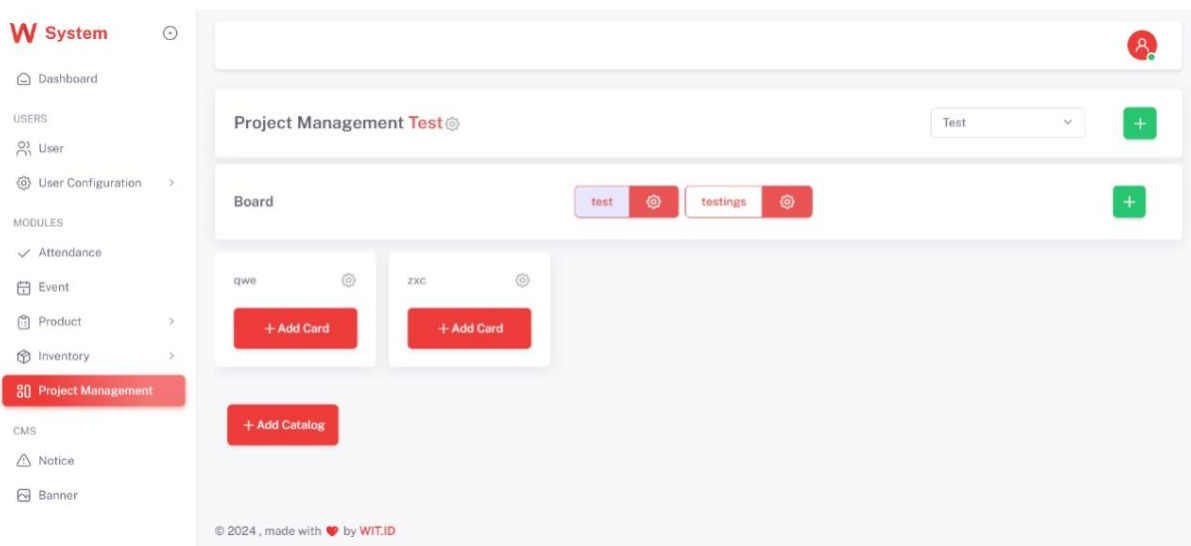
Gambar 7. User Interface Manajemen Inventaris Kantor

Gambar 8 menunjukkan antarmuka untuk penambahan data manajemen inventaris kantor yang menyediakan kemudahan bagi pengguna dalam memperbarui informasi inventaris dengan menambah, menghapus, atau mengedit data, memastikan bahwa semua informasi selalu akurat dan terkini. Gambar 9 menggambarkan antarmuka pengguna untuk manajemen proyek untuk pengguna dapat mengelola proyek, menetapkan tugas, memantau kemajuan, dan berkolaborasi dengan tim secara efektif. Integrasi ketiga fitur

ini dalam aplikasi super mendukung efisiensi operasional dan akurasi manajemen inventaris serta proyek kantor secara keseluruhan.



Gambar 8. User Interface Update Data Manajemen Inventaris Kantor



Gambar 9. User Interface Manajemen Proyek Kantor

3.3. Black Box Testing

Black box testing dilakukan sebagai bagian penting dari proses pengujian aplikasi. Metode ini menguji fungsionalitas aplikasi tanpa memperhatikan detail implementasi internalnya. Dengan menggunakan pendekatan ini, setiap fitur dan fungsi dalam aplikasi dievaluasi untuk memastikan bahwa aplikasi beroperasi sesuai dengan spesifikasi yang diinginkan.

Tujuan dari *black box testing* adalah untuk mengidentifikasi dan memperbaiki *bug* serta kelemahan fungsional dalam aplikasi. Proses ini melibatkan simulasi penggunaan aplikasi oleh pengguna akhir, dengan memasukkan berbagai *input* dan mengevaluasi *output* yang dihasilkan. Hasil dari *black box testing* akan memberikan gambaran tentang kualitas dan kinerja keseluruhan dari aplikasi super untuk administrasi kantor. *Black Box Testing* terhadap aplikasi super memverifikasi fungsionalitas aplikasi dari perspektif pengguna akhir tanpa perlu memperhatikan detail implementasi internal. Hasil pengujian menunjukkan bahwa fitur-fitur kunci seperti presensi kehadiran, pengajuan cuti, dan manajemen proyek beroperasi sesuai yang diharapkan, sementara *feedback* dari pengujian telah memberikan wawasan berharga untuk meningkatkan *usability* dan kualitas keseluruhan aplikasi. Dengan demikian, *Black Box Testing* telah membuktikan dirinya sebagai langkah yang krusial dalam memastikan bahwa aplikasi dapat memenuhi standar kualitas yang diharapkan oleh pengguna.

4. Kesimpulan

Penelitian ini berhasil menghasilkan sebuah sistem administrasi kantor berbasis *website* menggunakan REST API dan kerangka kerja Laravel, dengan fokus pada efisiensi administrasi sebagai kunci utama dalam menjaga kelancaran operasional organisasi di era globalisasi dan teknologi. Aplikasi yang dikembangkan tidak hanya menyediakan fitur-fitur dasar seperti presensi kehadiran dan manajemen cuti, tetapi juga mengintegrasikan manajemen inventaris, sistem manajemen proyek, dan pembuatan resume karyawan, yang berpotensi meningkatkan efisiensi operasional serta produktivitas karyawan secara menyeluruh. Dengan demikian, penelitian ini memberikan kontribusi yang signifikan dalam memperbaiki efisiensi administrasi kantor melalui solusi teknologi yang terintegrasi dan adaptif.

5. Ucapan Terima Kasih

Ucapan terima kasih yang sebesar-besarnya diberikan kepada Universitas Kristen Maranatha dan Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Kristen Maranatha atas dukungan moril dan dana untuk penelitian ini. Tidak lupa juga bagi Fakultas Teknologi Informasi Universitas Kristen Maranatha yang telah bersedia untuk menjadi tempat panduan dalam melakukan penelitian bagi mahasiswanya.

Referensi

- [1] S. Melián-González and J. Bulchand-Gidumal, "Information technology and front office employees' performance," *International Journal of Contemporary Hospitality Management*, vol. 29, no. 8, 2017, doi: 10.1108/IJCHM-10-2015-0585.
- [2] M. Steinberg, R. Mukherjee, and A. Punathambekar, "Media power in digital Asia: Super apps and megacorps," *Media Cult Soc*, vol. 44, no. 8, 2022, doi: 10.1177/01634437221127805.
- [3] N. Yadav, D. S. Rajpoot, and S. K. Dhakad, "LARAVEL: A PHP Framework for E-Commerce Website," in *Proceedings of the IEEE International Conference Image Information Processing*, 2019. doi: 10.1109/ICIIP47207.2019.8985771.
- [4] J. Sutticherchart and S. Rakthin, "Determinants of digital wallet adoption and super app: A review and research model," *Management and Marketing*, vol. 18, no. 3, 2023, doi: 10.2478/mmcks-2023-0015.
- [5] S. Budi *et al.*, "IBAtS - Image based attendance system: A low cost solution to record student attendance in a classroom," in *Proceedings - 2018 IEEE International Symposium on Multimedia, ISM 2018*, 2018. doi: 10.1109/ISM.2018.00037.

- [6] P. Svejvig and P. Andersen, "Rethinking project management: A structured literature review with a critical look at the brave new world," *International Journal of Project Management*, vol. 33, no. 2, 2015, doi: 10.1016/j.ijproman.2014.06.004.
- [7] K. I. Bagwan and S. D. Ghule, "A Modern Review on Laravel-PHP Framework," *IRE Journals: Iconic Research and Engineering Journals*, vol. 12, no. 2, 2019.
- [8] A. R. Fachrudin and G. Saputri, "Rancang Bangun Sistem Manajemen Banner Periklanan pada PT. Merah Putih Media Menggunakan Framework Laravel Berbasis Web dan Metode RAD," *SAINSTECH: JURNAL PENELITIAN DAN PENGKAJIAN SAINS DAN TEKNOLOGI*, vol. 33, no. 2, 2023, doi: 10.37277/stch.v33i2.1583.
- [9] R. M. Ferrer-Dávalos, "Influence of technology adoption on organizational performance: evidence from Paraguayan microenterprises," *South Florida Journal of Development*, vol. 4, no. 2, 2023, doi: 10.46932/sfjdv4n2-007.
- [10] I. C. Oghenerukevwe and N. E. Kayii, "Utilization of Cloud Technology Applications for Efficient Office Administration among Secretaries in Rivers State Universities," *International Journal of Information Systems and Informatics*, vol. 4, no. 1, 2023, doi: 10.47747/ijisi.v4i1.1059.
- [11] F. Susanto, F. Fauziah, and A. Andrianingsih, "Lecturer Attendance System using Face Recognition Application an Android-Based," *Journal of Computer Networks, Architecture and High Performance Computing*, vol. 3, no. 2, 2021, doi: 10.47709/cnahpc.v3i2.981.
- [12] L. V. Jánoky, J. Levendovszky, and P. Ekler, "An analysis on the revoking mechanisms for JSON Web Tokens," *Int J Distrib Sens Netw*, vol. 14, no. 9, 2018, doi: 10.1177/1550147718801535.
- [13] B. Costa, P. F. Pires, F. C. Delicato, and P. Merson, "Evaluating a Representational State Transfer (REST) architecture: What is the impact of REST in my architecture?," in *Proceedings - Working IEEE/IFIP Conference on Software Architecture 2014, WICSA 2014*, 2014. doi: 10.1109/WICSA.2014.29.
- [14] N. Rasyada, "SHA-512 Algorithm on Json Web Token for Restful Web Service-Based Authentication," *Journal of Applied Data Sciences*, vol. 3, no. 1, 2022, doi: 10.47738/jads.v3i1.51.
- [15] T. Dingsøyr, S. Nerur, V. Balijepally, and N. B. Moe, "A decade of agile methodologies: Towards explaining agile software development," *Journal of Systems and Software*, vol. 85, no. 6. 2012. doi: 10.1016/j.jss.2012.02.033.
- [16] K. Conboy, "Agility from first principles: Reconstructing the concept of agility in information systems development," *Information Systems Research*, vol. 20, no. 3, 2009, doi: 10.1287/isre.1090.0236.
- [17] A. Ehsan, M. A. M. E. Abuhaliqa, C. Catal, and D. Mishra, "RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions," *Applied Sciences (Switzerland)*, vol. 12, no. 9. 2022. doi: 10.3390/app12094369.