

PEMBUATAN APLIKASI WEB MANAJEMEN CUTI IZIN SAKIT (CIS) DI CV AVATAR SOLUTIONS

IPGMA Putra¹, IKG Suhartana², dan IMW Wirawan³

ABSTRAK

Kinerja karyawan dapat dilihat dari beberapa parameter. Kehadiran karyawan di tempat kerja merupakan salah satunya. Karyawan dengan komitmen tinggi biasanya memiliki catatan kehadiran, kinerja, dan produktivitas yang baik. Komitmen karyawan menunjukkan tingkat kesetiaan pada kebijakan perusahaan, kepuasan dan motivasi kerja yang tinggi, serta memiliki *turnover rate* lebih rendah. Salah satu metode yang dapat dilakukan untuk melakukan pemantauan kehadiran karyawan adalah aplikasi manajemen cuti, izin, dan sakit (CIS). Melalui aplikasi ini, ketidakhadiran karyawan serta alasannya dapat diketahui. Aplikasi manajemen CIS diimplementasikan di perusahaan yang bergerak di bidang ICT, yaitu Avatar Solutions. Aplikasi ini dibangun menggunakan SDLC bertipe *agile*. Pendekatan metode ini memungkinkan sistem dikembangkan dengan lebih fleksibel sesuai dengan *sprint-sprint*-nya. Karena aplikasi yang dibuat adalah aplikasi web, maka aplikasi yang dibuat dapat dijalankan di banyak *platform* asalkan terkoneksi internet. Luaran Aplikasi Manajemen CIS Avatar Solution adalah *flow* pengajuan CIS yang *seamless*, seperti pengajuan CIS dapat dilakukan pada sistem sentral dan limitasi pengajuan CIS langsung diatur oleh sistem sehingga untuk mengurangi kelonggaran.

Kata kunci : Manajemen CIS, Website, Aplikasi, Agile, Karyawan

ABSTRACT

The performance of employees can be assessed using several parameters, one of which is the attendance of employees in the workplace. Employees commitment typically maintain commendable records in attendance, performance, and productivity. Such high commitment is indicative of loyalty to company policies, elevated job satisfaction, strong work motivation, and a reduced turnover rate. One effective method for monitoring employee attendance is the leave management system. Through this application, instances of employee absenteeism and their causes can be tracked. The leave management application is implemented by companies operating within the ICT domain, such as Avatar Solutions. This application is constructed utilizing an agile Software Development Life Cycle (SDLC) approach. This method allows for a more flexible system development in line with its individual sprints. The output of this implementation is a web-based application, it works across multiple platforms if they are internet-enabled. The output of the Avatar Solution Leave Management Application manifests as a seamless leave submission process, whereby submissions can be made through the central system, with system-defined limitations set to reduce flexibility.

Keywords: Leave Management System, Website, Application, Agile, Employee

¹ Program Studi Informatika, Fakultas MIPA, Universitas Udayana, Bukit Jimbaran, 80361, Badung-Indonesia, gedemahardika@student.unud.ac.id

² Program Studi Informatika, Fakultas MIPA, Universitas Udayana, Bukit Jimbaran, 80361, Badung-Indonesia, ikg.suhartana@unud.ac.id

³ Program Studi Informatika, Fakultas MIPA, Universitas Udayana, Bukit Jimbaran, 80361, Badung-Indonesia, made_widhi@unud.ac.id

Submitted: 7 Oktober 2024

Revised: 26 Oktober 2024

Accepted: 26 Oktober 2024

1. PENDAHULUAN

Kehadiran karyawan merupakan salah satu parameter yang dapat dijadikan acuan kinerja karyawan. Komitmen karyawan juga terlihat dari kehadirannya di tempat kerja. Karyawan yang berkomitmen tinggi umumnya memiliki catatan kehadiran, kinerja, dan produktivitas yang baik. Komitmen yang tinggi juga menunjukkan tingkat kesetiaan pada kebijakan perusahaan, kepuasan dan motivasi kerja yang tinggi, serta memiliki *turnover rate* lebih rendah (Sembiring & Winarto, 2020). Berdasarkan fakta yang ada, maka pemantauan terhadap kehadiran karyawan merupakan agenda penting yang harus dijalankan perusahaan. Mempunyai sebuah sistem sentral yang dapat mendata kehadiran karyawan merupakan bagian penting untuk perusahaan. Selain itu, sistem sentral juga menjadi bagian dari bukti tercatat kehadiran atau pun ketidakhadiran karyawan di waktu tertentu.

Sistem manajemen cuti, izin, dan sakit (CIS) adalah salah satu implementasi dari pencatatan kehadiran karyawan. Jika pencatatan kehadiran mencatatkan kehadiran karyawan, sebaliknya sistem manajemen CIS mencatatkan ketidakhadiran karyawan. Tingkat kehadiran dapat dilihat dari jumlah ketidakhadiran (Ronal dkk., 2020). Sistem manajemen CIS pun harus memiliki beberapa kemampuan, seperti dapat berjalan secara *cross-platform* mengingat di perusahaan, terutama perusahaan teknologi, tidak semua karyawan menggunakan ekosistem atau *platform* yang sama. Pendekatan *cross-platform* memiliki keuntungan dalam penerapannya. Keuntungan ini antara lain, pendistribusian kode yang sama untuk tiap *platform* sehingga dapat mengurangi duplikasi kode. Keuntungan lain yang didapatkan adalah konsep pengembangan aplikasi yang sama di tiap *platform*, tidak perlu memikirkan kompatibilitas di *platform* yang berbeda (Blanco & Lucrédio, 2021).

Avatar Solutions adalah salah satu perusahaan yang bergerak di bidang pengembangan *software* dan *information and communication technology* (ICT) di Bali (Avatar Solutions, 2023). Sebagai salah satu *software house* terkemuka di Bali, perusahaan ini tentunya akan menjaga kualitas produk dan jasanya. Kualitas ini dapat dilihat dari kinerja karyawannya, mengingat Avatar Solutions memiliki berbagai tipe karyawan, seperti karyawan tetap (*full-time*), karyawan paruh waktu (*part time*), *freelance*, dan pekerja magang. Oleh karena itu, aplikasi manajemen CIS dapat digunakan sebagai sistem sentral yang mendata ketidakhadiran karyawan. *Output* berupa kinerja dan performa karyawan pun dapat di-*tracking* melalui aplikasi manajemen CIS.

2. METODE PELAKSANAAN

2.1 Pelaksanaan Kegiatan

Praktik Kerja Lapangan (PKL) merupakan sebuah pendekatan pembelajaran berbasis proyek atau *project-based learning*. Melalui PKL, pembelajaran difokuskan pada bidang praktisi dengan menerapkan pengetahuan akademik yang telah didapatkan. PKL bertujuan untuk memecahkan masalah nyata pada masyarakat, seperti pada perusahaan, organisasi, dan pemerintah. PKL pun memberikan luaran berupa peningkatan *hard-skill* dan *soft-skill*. Peningkatan *hard-skill* berupa kemampuan teknis didapatkan dari pembangunan proyek-proyek yang dikerjakan selama PKL, sedangkan *soft-skill* didapatkan dari proses pengerjaan proyek melalui kolaborasi tim di tempat PKL.

2.2 Kebutuhan Fungsional dan Nonfungsional Sistem

Kebutuhan fungsional dan nonfungsional wajib didefinisikan terlebih dahulu sebelum memulai pengembangan aplikasi. Kebutuhan fungsional menentukan fitur-fitur utama dan kemampuan yang harus dapat dilakukan oleh sistem, sedangkan kebutuhan nonfungsional meliputi batasan kualitas, seperti keamanan, kinerja, dan kegunaan yang tidak terikat langsung ke fitur sistem (Becker dkk.,

2019). Aplikasi Manajemen CIS untuk Avatar Solutions memiliki beberapa kebutuhan fungsional agar sistem dapat melakukan fungsi dan memenuhi ekspektasi pengguna terhadap aplikasi. Berikut adalah kebutuhan fungsional dari sistem yang dikembangkan.

Tabel 2.1 Kebutuhan Fungsional

No	Kebutuhan Fungsional
1	Sistem dapat menangani proses <i>input</i> data CIS dari karyawan
2	Sistem dapat menangani proses <i>delete</i> data CIS dari karyawan dan admin
3	Sistem memiliki fitur memberikan <i>acceptance</i> atau <i>rejection</i> untuk CIS yang dimasukkan
4	Sistem dapat menampilkan daftar CIS yang diajukan
5	Sistem dapat menampilkan detail CIS yang diajukan
6	Sistem dapat melakukan limitasi otomatis terhadap kuantitas CIS yang diajukan

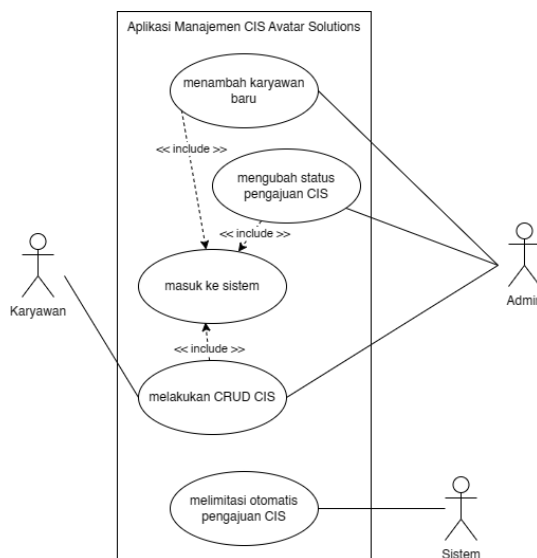
Kebutuhan nonfungsional juga harus ditentukan agar pengembangan aplikasi berjalan lancar dan aplikasi memiliki performa yang memadai untuk digunakan. Berikut kebutuhan nonfungsional aplikasi manajemen CIS Avatar Solutions.

Tabel 2.2 Kebutuhan Nonfungsional

No	Aspek	Kebutuhan Nonfungsional
1	Operasional	Sistem dapat diakses melalui <i>smartphone</i> dan komputer yang terkoneksi internet
2	Waktu respons	Sistem dapat merespons masukan pengguna di bawah 5 detik
3	Keamanan	a. Hanya admin yang dapat mengubah status CIS menjadi <i>accepted</i> atau <i>rejected</i> b. Hanya admin yang dapat melihat pengajuan CIS dari semua karyawan c. Hanya admin yang dapat mendaftarkan karyawan baru d. Karyawan bisa membuat pengajuan CIS baru e. Karyawan bisa mengubah detail CIS f. Karyawan bisa menghapus CIS g. Karyawan hanya bisa melihat daftar dan detail pengajuan CIS miliknya
4	Teknologi sistem	a. Komputer server dapat menjalankan Docker b. Bahasa pemrograman: Go (<i>back end</i>), JavaScript (<i>front end</i>) c. Library/framework: Gin (<i>back end</i>), ReactJS (<i>front end</i>) d. DBMS: MySQL

2.3 Perancangan Sistem

Perancangan sistem adalah langkah krusial sebelum masuk ke tahap pengkodean sistem. Secara garis besar, sistem dapat digambarkan menggunakan *use case diagram* yang merupakan bagian dari diagram *unified modeling language* (UML). *Use case* menggambarkan interaksi antara pengguna dengan sistem yang dibangun. Diagram memiliki dua *metadata*, yaitu *metadata* properti dan *metadata* relasi. *Metadata* properti terdiri atas nama aktor dan nama *use case*, sedangkan *metadata* relasi dapat berupa informasi relasi antaraktor, relasi antara aktor dan *use case*, dan relasi antar-*use case* (Fauzan dkk., 2019). Berikut merupakan diagram Aplikasi Manajemen CIS Avatar Solutions.



Gambar 2.1. Diagram Use Case

Pada Gambar 2.1. terdapat tiga aktor pada diagram *use case* Aplikasi Manajemen CIS Avatar Solutions. Aktor-aktor ini adalah admin, karyawan, dan sistem. Admin dapat menambah data karyawan baru, mengubah status pengajuan CIS dan melakukan *create*, *read*, *update*, dan *delete* (CRUD) CIS. Lebih terbatas, karyawan hanya dapat melakukan CRUD pada CIS yang diajukan oleh dirinya sendiri. Tentunya, admin dan karyawan harus masuk ke sistem terlebih dahulu untuk dapat melakukan *use case* yang dimaksud. Aktor terakhir adalah sistem yang melakukan otomatisasi. Sistem secara otomatis melimit pengajuan cuti dan izin jika karyawan telah menggunakan semua haknya untuk cuti dan izin.

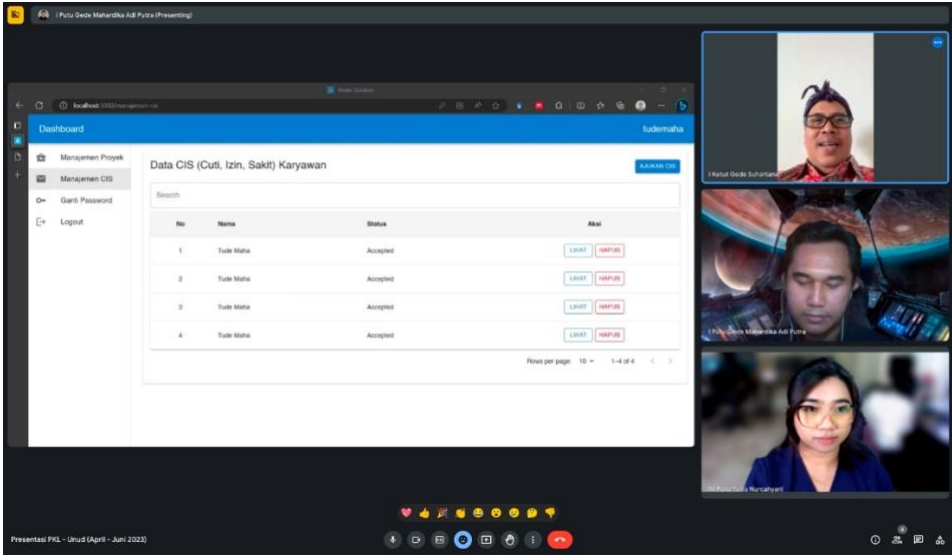
3. HASIL DAN PEMBAHASAN

Aplikasi dikembangkan menggunakan *software development life cycle* (SDLC) bertipe *agile*. SDLC ini termasuk yang independen, prosesnya dapat berjalan tanpa menunggu proses sebelumnya selesai. *Agile* menjadi tren yang banyak digunakan karena memadai untuk menghadapi kebutuhan pasar yang fleksibel (Harjanto dkk., 2020). *Agile* dibangun berdasarkan peningkatan berulang pada tiap prosesnya. Keuntungan yang dapat dirasakan ketika menggunakan metode *agile* adalah dapat beradaptasi pada perubahan-perubahan yang terjadi pada masa pengembangan dengan lebih cepat (Mehta & Sood, 2023). Metode ini juga menawarkan kemudahan bekerja sama pada tiap-tiap divisi pengembangan perangkat lunak, seperti divisi *back end* dan *front end* yang dapat bekerja bersama-sama.

Sprint pertama pada metode *agile* yang diterapkan adalah membuat versi awal aplikasi. Proses ini dimulai dengan *plan and design*. Perencanaan dimulai dengan membuat skema basis data yang digunakan untuk sistem manajemen CIS. Skema dibuat dalam bentuk *entity relationship diagram* (ERD). Pembuatan ERD dimulai dari menentukan tabel-tabel yang terlibat, atribut masing-masing tabel, serta relasinya. Pada tahap perencanaan juga dibuat desain tampilan aplikasi web. Proses selanjutnya adalah *development*. Pada proses ini, koding-koding aplikasi mulai dibuat. Koding dimulai dari bagian *back end*. *Object relational mapping* (ORM) digunakan untuk menghubungkan basis data ke *back end*. ORM berfungsi untuk mempermudah proses penyimpanan dan pengambilan data ke dan dari RDBMS dalam bentuk objek (Gorodnichev dkk., 2020).

Koding selanjutnya adalah pembuatan *application programming interface* (API) di *back end* dan tampilan web di *front end*. Komunikasi menggunakan API dipilih karena dapat meningkatkan kolaborasi antarpengembang, meningkatkan keamanan sistem, dan mendukung pengembangan inovasi (IBM, 2023). Pengembangan *back end* dan *front end* dilakukan hingga aplikasi yang dibuat minimal dapat berfungsi sesuai dengan kebutuhan sistem yang telah didefinisikan. Tahap selanjutnya adalah *testing*, *deployment*, dan *review*. Pada proses ini, pengguna aplikasi, yaitu Avatar Solutions. Tahap ini memungkinkan aplikasi dites, kemudian di-*hosting*, dan di-*review*.

Sprint-sprint selanjutnya dilakukan untuk menyempurnakan aplikasi yang dibuat. *Sprint* lanjutan dilakukan seperti langkah awal pengembangan perangkat lunak. *Sprint* ini dilakukan hingga didapatkan *acceptance* dari mitra. Saat aplikasi sudah disetujui fitur-fiturnya, dilanjutkan dengan sosialisasi pada mitra seperti yang terlihat pada Gambar 3.1.



Gambar 3.1. Sosialisasi dan Demo Aplikasi Kepada Mitra

Dengan dibangunnya Aplikasi Manajemen CIS, maka terdapat beberapa dampak pada *flow* pengajuan CIS di Avatar Solutions. Hasil dan dampak yang dimaksud dapat dilihat pada tabel di bawah.

Tabel 3.1. Tabel Hasil Kegiatan

Indikator	Sebelum Pengembangan Aplikasi	Setelah Pengembangan Aplikasi
Pengajuan CIS	Melalui aplikasi perpesanan, seperti WhatsApp	Dilakukan melalui sistem sentral Aplikasi Pengajuan CIS
Limitasi pengajuan CIS	Limitasi dicek secara manual melalui media penyimpanan	Limitasi ditentukan oleh sistem, karyawan otomatis tidak dapat mengajukan jika melebihi limit
Tabulasi data CIS	Tabulasi pada media penyimpanan manual, seperti <i>sheet</i>	Tabulasi telah disediakan di sistem, detail juga dapat dilihat pada bagian “ <i>details</i> ”

4. KESIMPULAN

Aplikasi Manajemen CIS Avatar Solutions penting untuk dikembangkan. Aplikasi ini dapat bermanfaat untuk operasional di internal Avatar Solutions. Sistem yang dibangun dapat membantu manajemen CIS yang dibuat oleh karyawan pada suatu tempat sentral. *Flow* yang lebih *seamless* juga didapatkan dari pengembangan sistem ini. Keuntungan yang didapat, yaitu pengajuan CIS dapat dilakukan pada sistem sentral yang telah dibangun, limitasi pengajuan CIS langsung diatur oleh sistem sehingga bisa mengurangi kelonggaran, dan terakhir adalah mempermudah tabulasi data CIS karyawan. Tabulasi ini pula dapat dilihat langsung detailnya dengan sekali klik. Manfaat lain yang dapat dirasakan nanti saat pengembangan aplikasi lebih lanjut adalah mudahnya pengembangan aplikasi karena menggunakan komunikasi API dan telah disiapkan kebutuhan sistem. Aplikasi yang berbentuk *web app* juga mempermudah aplikasi diakses dari banyak *platform* dan di mana saja selama terkoneksi dengan jaringan internet.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih atas kepercayaan yang diberikan oleh CV Avatar Solutions untuk bisa berkembang dan berproses selama proses PKL. Proses ini sangat bermanfaat untuk penulis bisa merasakan pengalaman bekerja di bidang *software house* secara langsung. Penulis juga mengucapkan terima kasih untuk Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana yang memfasilitasi proses pengabdian yang merupakan bagian dari Tri Dharma Perguruan Tinggi melalui kegiatan PKL. Kegiatan ini memberikan banyak *insight* positif untuk penulis bisa menjalankan Tri Dharma dan mendapatkan pengalaman di bidang praktisi selama menjalankan pengabdian.

DAFTAR PUSTAKA

- Avatar Solutions. (2023). Avatar Solutions — Software Development, Mobile Application, Web Design. <https://avatarsolution.com/about.html>
- Becker, P., Tebes, G., Peppino, D., & Olsina, L. (2019). Applying an Improving Strategy that embeds Functional and Non-Functional Requirements Concepts. *Journal of Computer Science and Technology*, 19(2), e15. <https://doi.org/10.24215/16666038.19.E15>
- Blanco, J. Z., & Lucrédio, D. (2021). A holistic approach for cross-platform software development. *Journal of Systems and Software*, 179, 110985. <https://doi.org/10.1016/j.jss.2021.110985>
- Fauzan, R., Siahaan, D., Rochimah, S., & Triandini, E. (2019). Use Case Diagram Similarity Measurement: A New Approach. 2019 12th International Conference on Information & Communication Technology and System (ICTS), 3–7. <https://doi.org/10.1109/ICTS.2019.8850978>
- Gorodnichev, M., Moseva, M., Poly, K., Dzhabrailov, K., & Gematudinov, R. (2020). Exploring Object-Relational Mapping (ORM) Systems and How to Effectively Program A Data Access Model. *PalArch's Journal of Archaeology of Egypt / Egyptology*, 17(3), 615–627. <https://doi.org/10.48080/JAE.V17I3.141>
- Harjanto, K. A., Ariance, M., & Pakereng, I. (2020). Perancangan Sistem Pengajuan Cuti Izin Sakit (CIS) Menggunakan Platform Pega System. *Jurnal Informatika*, 7(2), 131–140. <https://doi.org/10.31294/JI.V7I2.8232>
- IBM. (2023). What is an Application Programming Interface (API)? <https://www.ibm.com/topics/api>
- Mehta, K., & Sood, V. M. (2023). Agile Software Development in the Digital World – Trends and Challenges. Dalam *Agile Software Development* (hlm. 1–22). Wiley. <https://doi.org/10.1002/9781119896838.ch1>
- Ronal, O. :, Sihalohe, D., Siregar, H., Sisingamangaraja, U., Tapanuli, X., Kunci, K., Kerja, L., & Karyawan, K. (2020). Pengaruh Lingkungan Kerja terhadap Kinerja Karyawan pada PT. Super Setia Sagita Medan. *JURNAL ILMIAH SOCIO SECRETUM*, 9(2), 273–281. <http://jurnal.darmaagung.ac.id/index.php/socio/article/view/413>
- Sembiring, R., & Winarto, W. (2020). Pengaruh Budaya Kerja dan Komitmen Terhadap Kinerja Karyawan (Studi Kasus pada Perawat di Rumah Sakit Milik Pemerintah). *Jurnal Ilmiah METHONOMI*, 6(1), 21–30. <https://doi.org/10.46880/METHONOMI.VOL6NO1.PP21-30>