

# Sistem Informasi Manajemen Karyawan di Salah Satu Perusahaan Software Development di Indonesia: Studi R&D

Fery Holanda

Teknik Informatika, Politeknik Negeri Batam  
[feryholanda1@gmail.com](mailto:feryholanda1@gmail.com)

## Article Info

### Article history:

Received ...

Revised ...

Accepted ...

### Keyword:

Manajemen Karyawan, Metode Waterfall, Research and Development, Sistem Informasi

## ABSTRACT

This study aims to develop a web-based employee management system for a software development company in Indonesia. The system was developed using ReactJS, C#, and SQL Server to provide structured and efficient employee data management. The development process applied the Waterfall method, which consists of sequential stages including analysis, design, implementation, testing, and maintenance. The system's functionality was tested using Black Box Testing to ensure that all features operate as intended. The results show that the system improves the efficiency of the Human Resources Department in managing employee data and facilitates better workflow in administrative processes. The implementation of this system is expected to provide a more effective and reliable solution for employee data management within the company.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

## I. PENDAHULUAN

Kemajuan teknologi di zaman digital saat ini telah membawa pengaruh yang positif di berbagai bidang, khususnya dalam hal pengelolaan data karyawan [1]. Setiap perusahaan seharusnya bisa memanfaatkan inovasi teknologi untuk mengumpulkan, menyimpan dan mengelola data karyawan secara akurat dan efisien [2]. Karyawan merupakan sumber daya yang krusial yang berkontribusi pada peningkatan produktivitas dan performa perusahaan. Dengan demikian pemanfaatan teknologi informasi memberikan keuntungan tersendiri bagi perusahaan dalam meningkatkan efisiensi saat mengelola data karyawan.

Pengelolaan data karyawan, khususnya berkaitan dengan gaji masih dilakukan secara tradisional menggunakan Excel [3]. Namun, dengan metode manual ini, terdapat kemungkinan kesalahan dalam perhitungan hari kerja karyawan, jumlah gaji dan ada risiko pemalsuan data. Selain itu, proses permohonan cuti, lembur dan perubahan data pribadi karyawan masih dilaksanakan dengan cara konvensional menggunakan formulir kertas.

Hal ini menyebabkan penggunaan kertas yang berlebihan, tidak efisien, menambah beban kerja *Human Resource Department*, serta mengurangi efektivitas pengelolaan informasi pegawai. Proses absensi juga merupakan salah

satu tantangan yang di hadapi. Sebelumnya, pencatatan dilakukan menggunakan mesin absensi, lalu informasi tersebut diolah secara manual oleh bagian bagian Human Resource Department dan dimasukkan ke dalam file Excel. Dengan data tersebut, proses pembayaran gaji selanjutnya dilaksanakan. Namun cara ini dianggap tidak efisien karena belum ada integrasi langsung antara sistem absensi dan sistem penggajian.

Berdasarkan masalah yang telah diidentifikasi, telah dirancang suatu sistem informasi manajemen karyawan yang bertujuan untuk menjalankan proses absensi secara langsung berdasarkan lokasi. Data absensi yang dicatat akan segera disimpan ke dalam sistem, sehingga dapat diakses oleh karyawan maupun *Human Resource Department* dan selanjutnya diproses oleh HRD untuk keperluan administrasi dan penggajian. Selain itu, pengajuan cuti, lembur, serta perubahan data pribadi karyawan dilakukan melalui sistem yang berbasis web. Proses penggajian juga terintegrasi dengan data absensi dan lembur yang telah disimpan dalam sistem. Kehadiran sistem ini diharapkan dapat meningkatkan efisiensi dan efektivitas dalam pengelolaan informasi pegawai, mengurangi penggunaan kertas, meminimalkan kesalahan dalam perhitungan gaji, serta mengurangi beban kerja di bagian HRD.

Pengembangan sistem ini mencakup tiga pengguna utama, yaitu Admin, *Human Resource Department* dan karyawan. Penerapan sistem ini dilaksanakan dengan memanfaatkan ReactJS dan C# sebagai bahasa pemrograman, serta menggunakan SQL Server untuk menyimpan dan pengelolaan data. Pilihan kombinasi teknologi ini di ambil untuk mendukung penciptaan fitur fitur sistem yang efisien, terorganisir dan memiliki kinerja optimal.

## II. METODE

### A. Metode Penelitian

Metode *Research and Development* (R&D) adalah pendekatan yang menitikberatkan pada penciptaan serta pengembangan produk melalui proses penelitian yang terstruktur. Menurut Chen et al. [4], upaya R&D mencakup alokasi sumber daya manusia, biaya, dan peralatan yang berkaitan dengan aspek penelitian dan pengembangan, yang berfungsi sebagai pendukung utama untuk pertumbuhan yang berkelanjutan dan inovasi dalam suatu organisasi. Dalam konteks studi ini, metode R&D diadopsi untuk menganalisis kebutuhan pengguna selama fase penelitian dan menghasilkan produk yang berbasis sistem informasi pada fase pengembangan.

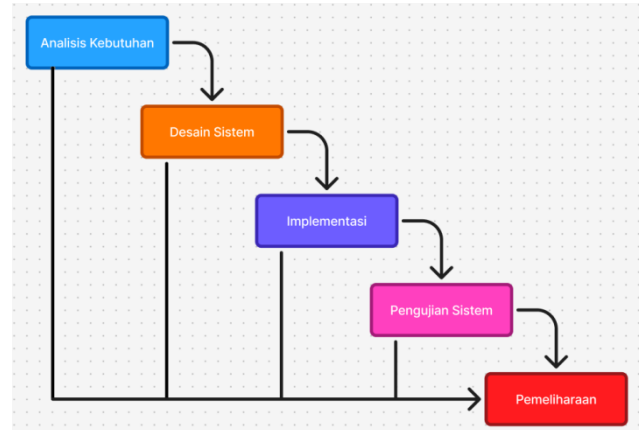
Dalam pendekatan penelitian ini, selanjutnya digunakan pendekatan pengembangan perangkat lunak, yang dikenal dengan metode pengembangan *Waterfall*.

### B. Metode Pengembangan Perangkat Lunak

Metode *Waterfall* adalah salah satu model paling awal dalam pengembangan perangkat lunak yang digunakan di sektor rekayasa perangkat lunak. Konsep ini pertama kali diluncurkan oleh Herbert D. Benington pada tahun 1956 selama presentasinya di *Symposium on Advanced Programming Methods for Digital Computers*, kemudian disebarluaskan dalam *Annals of the History of Computing* pada tahun 1983 [5].

Model tersebut menggambarkan langkah-langkah dalam pengembangan sistem yang dilakukan secara berurutan dan terstruktur, di mana setiap fase harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Langkah-langkah dalam metode *Waterfall* biasanya mencakup seperti pada Gambar 1, analisis kebutuhan, desain sistem, implementasi, pengujian sistem, hingga pemeliharaan [6].

Kelebihan utama dari kerangka kerja ini berada pada proses yang sistematis dan tercatat dengan jelas, yang memungkinkan pengawasan proyek serta pemantauan perkembangan dengan lebih mudah. Meski demikian, model ini juga menghadapi beberapa kelemahan, terutama dikarenakan ketidakmampuannya untuk beradaptasi dengan pergeseran kebutuhan pengguna selama fase pengembangan [5].



Gambar 1. Waterfall

#### 1. Analisis Kebutuhan

Tahap analisis kebutuhan merupakan langkah awal dalam pendekatan Waterfall, yang bertujuan untuk mengenali kebutuhan dari pengguna serta sistem itu sendiri [6]. Dalam fase ini, pengumpulan informasi dilakukan melalui observasi dan wawancara dengan pihak *Human Resource Department*. Keluaran dari tahap ini adalah daftar kebutuhan fungsional seperti pengelolaan data karyawan, absensi, pengajuan cuti, lembur, perubahan data diri dan penggajian, serta kebutuhan non fungsional seperti keamanan data dan kemudahan penggunaan sistem.

Secara lebih mendetail, sistem informasi manajemen karyawan ini memiliki tiga pengguna utama, yaitu Admin, *Human Resource Department* dan karyawan yang masing masing memiliki kebutuhan fungsional sebagai berikut:

Tabel 1. *Functional Requirement*

Kode	Functional Requirement
F001	Admin dapat mengelola data pengguna (user login).
F002	Pengguna dapat melakukan proses login untuk mengakses sistem.
F003	HRD dapat mengelola data karyawan.
F004	Karyawan dapat melakukan absensi.
F005	Karyawan dapat melihat riwayat absensi.
F006	HRD dapat melakukan proses absensi.
F007	Karyawan dapat melakukan pengajuan cuti, lembur dan perubahan data diri.
F008	Karyawan dapat melihat riwayat pengajuan cuti, lembur dan perubahan data diri.
F009	HRD dapat melakukan melakukan proses persetujuan atau penolakan terhadap pengajuan cuti, lembur dan perubahan data diri.
F010	HRD dapat melakukan proses penggajian berdasarkan hasil absensi dan lembur karyawan.

Berdasarkan pada Tabel 1, fungsi-fungsi sistem mencakup kemampuan Admin untuk mengatur informasi pengguna serta hak akses sistem, HRD dalam mengelola data karyawan, melakukan proses absensi, dan menghitung gaji berdasarkan data absensi dan lembur karyawan. Di sisi lain, karyawan diperbolehkan untuk absensi, melihat riwayat absensi, mengajukan permohonan cuti, lembur, dan perubahan data pribadi, serta memonitor status atau riwayat permohonan mereka. Selain itu, HRD juga diberi hak untuk menyetujui atau menolak setiap permintaan yang dimasukkan oleh karyawan. Semua fitur tersebut dirancang dengan tujuan untuk memastikan bahwa pengelolaan data karyawan berjalan dengan cara yang efisien, efektif, dan terstruktur.

Tabel 2. *Non Functional Requirement*

Kriteria	Non Functional Requirement
Usability	Sistem mudah digunakan oleh pengguna dengan tampilan antarmuka yang sederhana.
Reliability	Sistem dapat berfungsi secara stabil tanpa sering mengalami gangguan.
Kinerja	Sistem mampu memproses dan menampilkan data dengan cepat.
Compability	Sistem dapat diakses melalui berbagai perangkat dan browser
Keamanan	Sistem menjamin keamanan data dengan autentikasi dan pembatasan hak akses user.

Berdasarkan pada Tabel 2, menjelaskan aspek sistem yang meliputi *Usability*, *reliability*, kinerja, *compability* dan keamanan. Aspek aspek ini bertujuan untuk memastikan sistem dapat digunakan dengan mudah oleh pengguna, beroperasi secara stabil, memiliki performa yang cepat, dapat diakses melalui berbagai perangkat, serta menjaga keamanan data agar sistem berjalan optimal.

## 2. Desain Sistem

Setelah seluruh kebutuhan telah dikumpulkan, pengembang bergerak ke fase perancangan sistem sebelum memulai proses pengkodean. Pada fase ini, perhatian tertuju pada pengembangan struktur informasi dan desain tampilan perangkat lunak, yang meliputi pembuatan *Use Case Diagram*, *Activity Diagram*, serta *Entity Relationship Diagram* [7].

## 3. Implementasi

Setelah menyelesaikan tahap desain sistem, tahapan selanjutnya adalah menerapkan hasil desain ke dalam sistem yang dapat dioperasikan dengan bahasa pemrograman yang tepat. Pada fase ini, semua desain seperti *Use Case Diagram*, *Activity Diagram*, dan *Entity Relationship Diagram* diubah menjadi kode program yang sesuai dengan kebutuhan fungsional dan non fungsional yang telah ditetapkan sebelumnya [8].

## 4. Pengujian Sistem

Tahap pengujian sistem adalah fase yang dilaksanakan setelah developer menyelesaikan tahap implementasi, dan ini bertujuan untuk memastikan bahwa aplikasi yang telah dirancang berfungsi dengan baik tanpa adanya kesalahan (*error*) sebelum digunakan oleh pengguna [7]. Hasil dari proses pengujian juga bisa digunakan sebagai pedoman untuk melakukan perbaikan terhadap sistem.

Tabel 3. Data Demografi Partisipan Pengujian Sistem

Kode Partisipan	Jabatan	Pendidikan	Pengalam Kerja
P1	Manager	S1 Teknik Informatika	17 Tahun
P2	Karyawan	D3 Teknik Informatika	5 Tahun

## 5. Pemeliharaan

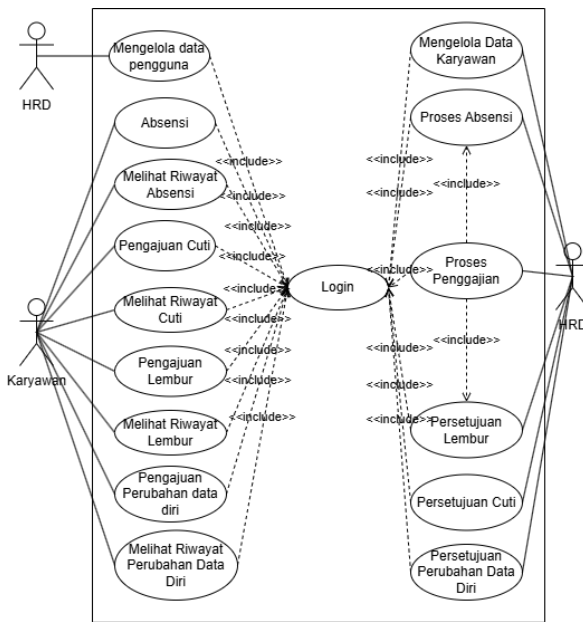
Tahap penutup dalam metode *Waterfall* adalah pemeliharaan sistem. Setelah seluruh pengujian sistem dilaksanakan dan dipastikan bebas dari kesalahan (*error*), program tersebut siap digunakan oleh para pengguna. Sistem harus dipelihara dan diperbaiki jika muncul masalah saat digunakan oleh pengguna [7].

## III. HASIL DAN PEMBAHASAN

### A. Rancangan Arsitektur Sistem

#### 1. Use Case Diagram

Use Case Diagram adalah salah satu elemen penting dalam *Unified Modeling Language* (UML) yang bertujuan untuk menggambarkan interaksi antara aktor luar dengan sistem perangkat lunak. Diagram ini berperan dalam memodelkan kebutuhan fungsional sistem dengan cara menggambarkan perilaku sistem yang diinginkan dari sudut pandang pengguna [9]. Use case juga menggambarkan serangkaian tindakan yang menghasilkan output yang terlihat oleh aktor, dan diagram ini berfungsi untuk menjelaskan koneksi antara aktor dan berbagai proses yang ada dalam sistem [10].



Gambar 2. Use Case Diagram

2. Activity Diagram

Activity Diagram adalah salah satu jenis diagram dalam Unified Modeling Language (UML) yang digunakan untuk memvisualisasikan rangkaian aktivitas atau proses bisnis di dalam sebuah sistem. Diagram ini menggambarkan urutan tindakan yang dilakukan oleh entitas dan sistem untuk mencapai tujuan tertentu [9].

Saat memanfaatkan Activity Diagram, pengembang dapat memperoleh pemahaman mengenai logika alur kerja dari setiap proses, dari awal hingga akhir, serta dapat mengevaluasi keputusan yang bercabang dan proses yang dilakukan secara bersamaan.

Activity Diagram juga berperan penting dalam menganalisis dan memodelkan proses bisnis sebelum dilaksanakan tahap implementasi, sehingga pengembang dapat mendeteksi potensi masalah dalam proses lebih awal.

3. Entity Relationship Diagram

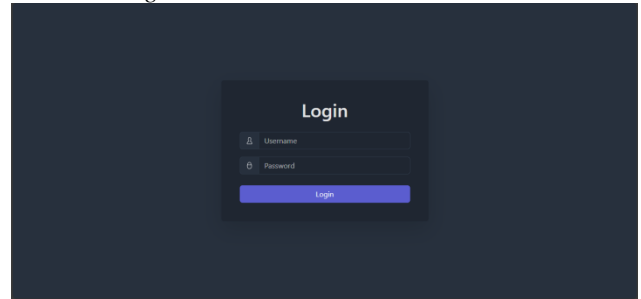
Entity Relationship Diagram (ERD) merupakan suatu gagasan yang dirancang untuk menunjukkan keterkaitan antar entitas dalam database. Entity Relationship Diagram dapat memberikan dukungan kepada pengembang dalam menciptakan struktur data dengan efisiensi tinggi [11]. Dalam fase ini, Entity Relationship Diagram berperan untuk mempresentasikan keterkaitan antara absensi, cuti, lembur serta data karyawan.

B. Implementasi

Tahap implementasi adalah langkah penerapan hasil desain sistem ke dalam aplikasi yang dapat digunakan. Dalam fase ini, semua desain yang telah disusun sebelumnya seperti Use Case Diagram, Activity Diagram dan Entity Relationship Diagram dikonversi menjadi kode program

menggunakan bahasa pemrograman ReactJS untuk sisi Front-End dan C# untuk sisi Back-End, dengan SQL Server sebagai basis datanya.

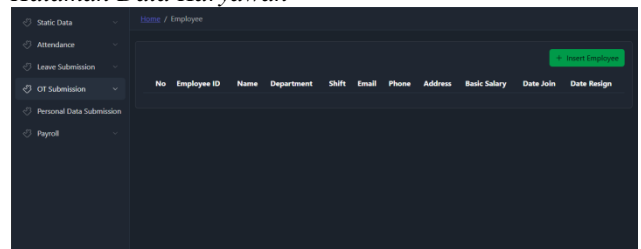
1. Halaman Login



Gambar 3. Form Login

Halaman login adalah komponen pertama yang perlu dijangkau oleh pengguna sebelum memasuki sistem. Pada halaman ini, pengguna diminta untuk mengisi informasi rahasia yang terdiri dari nama pengguna dan kata sandi.

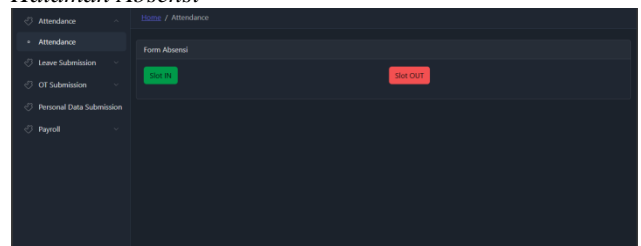
2. Halaman Data Karyawan



Gambar 4. Halaman Data Karyawan

Halaman ini digunakan untuk mempresentasikan informasi karyawan yang telah disimpan dalam sistem, termasuk detail seperti Employee ID, Employee Name, Department, Shift, Email, Phone, Address, Basic Salary, Date Join dan Date Resign.

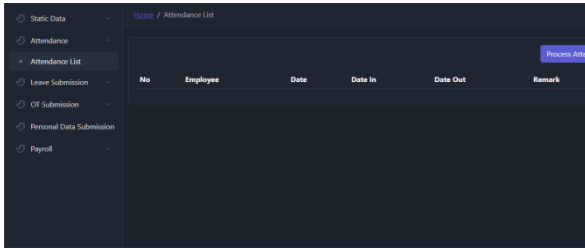
3. Halaman Absensi



Gambar 5. Halaman Absensi

Gambar 5 menampilkan tampilan bagaimana karyawan akan melakukan absensi, ada dua pilihan tombol yang tersedia, Slot In dan Slot Out.

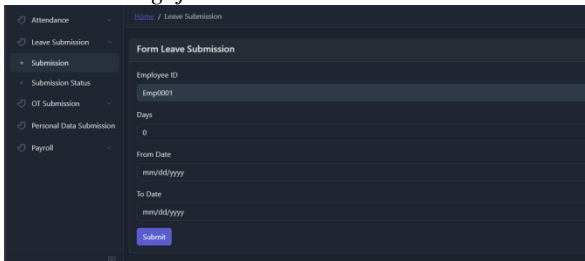
4. Halaman Proses Absensi



Gambar 6. Halaman Proses Absensi

Gambar 6 menampilkan halaman proses absensi yang digunakan oleh *Human Resource Department*. Terdapat tombol *Process Attendance*, setelah tombol itu ditekan, sistem akan menghasilkan sebuah data absensi yang kemudian akan ditampilkan ke dalam sebuah tabel, data absensi tersebut juga digunakan untuk proses penggajian.

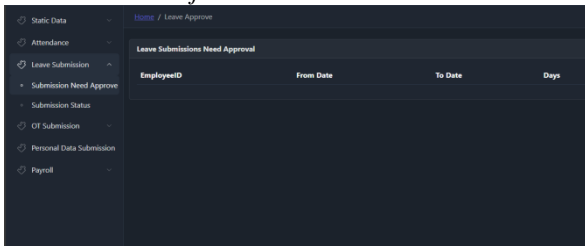
### 5. Halaman Pengajuan Cuti



Gambar 7. Halaman Pengajuan Cuti

Gambar 7 merupakan tampilan dari pengajuan cuti, halaman ini digunakan oleh karyawan untuk mengajukan cuti, karyawan diwajibkan mengisi berapa hari akan cuti, dimulai dari tanggal berapa sampai tanggal berapa. Kemudian menekan tombol submit, yang mana itu akan masuk ke dalam list persetujuan cuti di menu *Human Resource Department*.

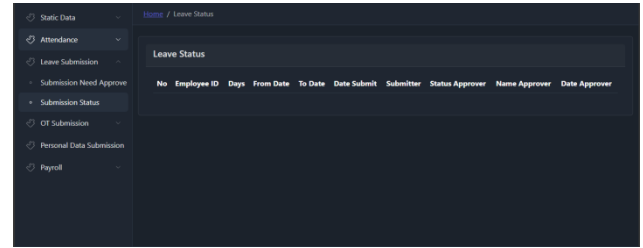
### 6. Halaman Persetujuan Cuti



Gambar 8. Halaman Persetujuan Cuti

Gambar 8 memperlihatkan halaman persetujuan cuti, dimana *Human Resource Department* akan melakukan persetujuan atau penolakan cuti yang telah diajukan oleh karyawan.

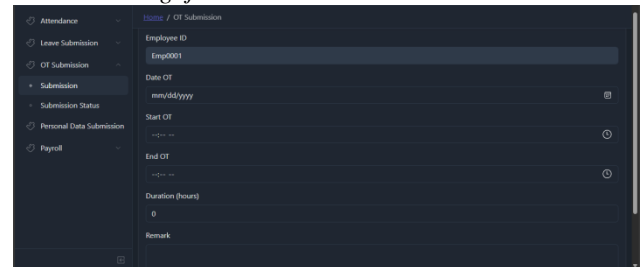
### 7. Halaman Riwayat Cuti



Gambar 9. Halaman Riwayat Cuti

Gambar 9 memperlihatkan halaman riwayat cuti, dimana halaman ini berisi informasi riwayat pengajuan cuti karyawan tersebut. Pada halaman ini berisi informasi kapan pengajuan itu disetujui atau ditolak dan status pengajuan cuti.

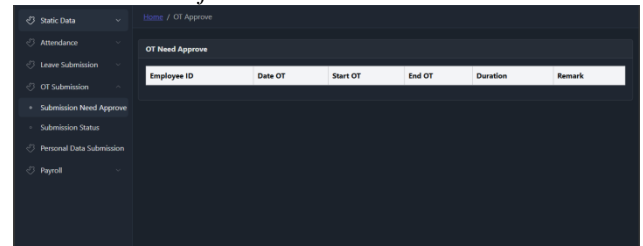
### 8. Halaman Pengajuan Lembur



Gambar 10. Halaman Pengajuan Lembur

Gambar 10 merupakan tampilan dari pengajuan lembur, halaman ini digunakan oleh karyawan untuk mengajukan lembur, karyawan diwajibkan mengisi tanggal berapa akan lembur, dimulai dari jam berapa sampai jam berapa melakukan lembur, karyawan juga disarankan untuk mengisi *Remark*, agar *Human Resource Department* mengetahui alasan lembur. Kemudian menekan tombol submit, yang mana akan masuk ke dalam list persetujuan lembur di menu *Human Resource Department*.

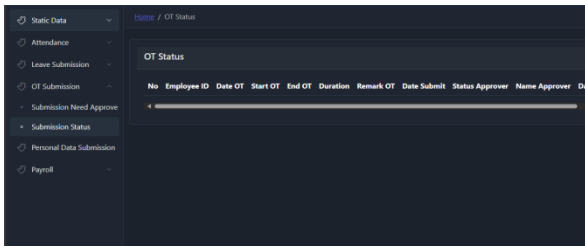
### 9. Halaman Persetujuan Lembur



Gambar 11. Halaman Persetujuan Lembur

Gambar 11 memperlihatkan halaman persetujuan lembur, dimana *Human Resource Department* akan melakukan persetujuan atau penolakan lembur yang telah diajukan oleh karyawan.

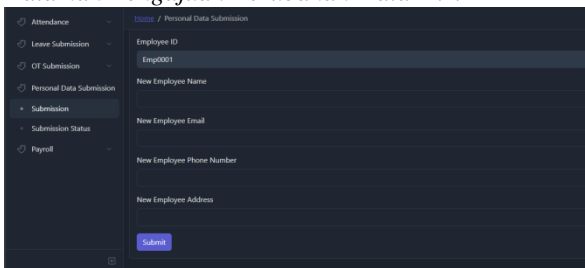
### 10. Halaman Riwayat Lembur



Gambar 12. Halaman Riwayat Lembur

Gambar 12 memperlihatkan halaman riwayat lembur, dimana halaman ini berisi informasi riwayat pengajuan lembur karyawan tersebut. Pada halaman ini berisi informasi kapan pengajuan itu disetujui atau ditolak dan status pengajuan lembur.

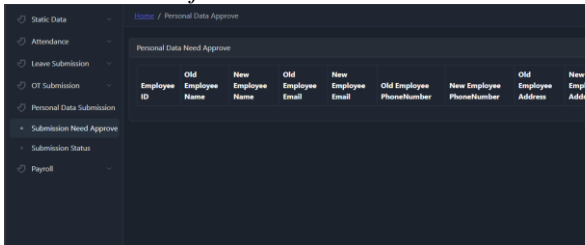
### 11. Halaman Pengajuan Perubahan Data Diri



Gambar 13. Halaman Pengajuan Perubahan Data Diri

Gambar 13 merupakan tampilan dari pengajuan perubahan data diri, halaman ini digunakan oleh karyawan untuk melakukan pengajuan perubahan data pribadi.

### 12. Halaman Persetujuan Perubahan Data Diri



Gambar 14. Halaman Persetujuan Perubahan Data Diri

### C. Hasil Pengujian

Pada tahap ini, dilakukan pengujian terhadap sistem yang telah diterapkan untuk memastikan bahwa semua fungsi beroperasi sesuai dengan spesifikasi dan rencana yang telah ditentukan. Pengujian ini dilakukan menggunakan metode *Black Box Testing*, yang merupakan strategi pengujian perangkat lunak yang menekankan pada analisis fungsi sistem berdasarkan masukan dan keluaran tanpa mempertimbangkan struktur internal kode program [12].

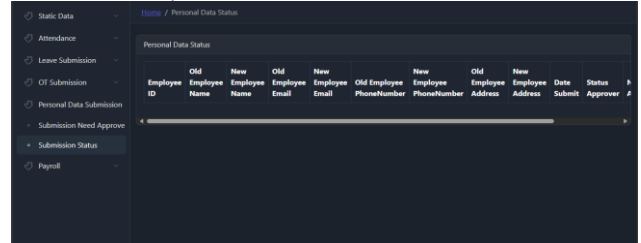
Konsep black box sendiri pertama kali dikenalkan dalam bidang *cybernetics* oleh Norbert Wiener pada tahun 1948 dan kemudian dikembangkan lebih lanjut oleh W. Ross Ashby pada tahun 1956 sebagai cara untuk memahami sistem melalui hubungan *input-output* tanpa perlu memahami mekanisme internalnya [12], [13]. Dalam dunia rekayasa perangkat lunak, prinsip ini diadaptasi menjadi metode *Black Box Testing* untuk memastikan bahwa sistem bekerja sesuai dengan kebutuhan pengguna. Dengan demikian, pengujian ini dilakukan dari sudut pandang pengguna untuk menilai kecocokan, keandalan, dan kesesuaian sistem dengan spesifikasi yang telah ditentukan. Hasil pengujian dapat dilihat pada tabel 3.

Tabel 3. Hasil Pengujian

No	Nama Fungsional	Skenario	Input	Output yang Diharapkan	Hasil Pengujian
----	-----------------	----------	-------	------------------------	-----------------

Gambar 14 memperlihatkan halaman persetujuan perubahan data diri, dimana *Human Resource Department* akan melakukan persetujuan atau penolakan perubahan data diri yang telah diajukan oleh karyawan.

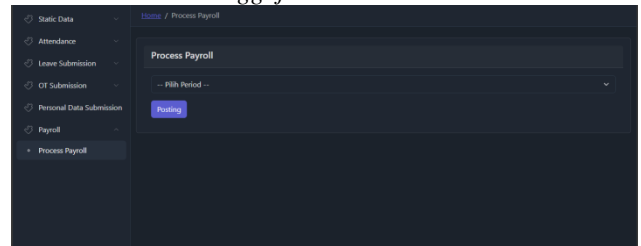
### 13. Halaman Riwayat Perubahan Data Diri



Gambar 15. Halaman Riwayat Perubahan Data Diri

Gambar 15 memperlihatkan halaman riwayat perubahan data diri, dimana halaman ini berisi informasi riwayat pengajuan perubahan data diri karyawan tersebut. Pada halaman ini berisi informasi perubahan data diri apa saja yang berubah, kapan pengajuan itu disetujui atau ditolak dan status pengajuan data diri.

### 14. Halaman Proses Penggajian



Gambar 16. Halaman Proses Penggajian

Gambar 16 memperlihatkan halaman proses penggajian, dimana halaman ini *Human Resource Department* akan melakukan proses penggajian dengan memilih periode penggajian.

1.	Login	Pengguna memasukkan username dan password yang benar	Username: admin, Password: 12345	Sistem menampilkan halaman utama	Berhasil
2.	Login	Pengguna memasukkan username dan password yang salah	Username: komet, Password: alat	Sistem menampilkan pesan "Username atau password salah"	Berhasil
3.	Login	Pengguna memasukkan username yang benar dan password yang salah	Username: admin, Password: 9999	Sistem menampilkan pesan "Password salah."	Berhasil
4.	Pengajuan Cuti	Karyawan mengisi form pengajuan cuti dengan detail cuti	Days : 1, FromDate : 20/10/2025, EndDate : 20/10/2025	Sistem menyimpan pengajuan dan menampilkan pesan "Leave submission berhasil diajukan!"	Berhasil
5.	Persetujuan Cuti	HRD memilih salah satu data cuti, kemudian menekan tombol Approve	Klik tombol Approve	Cuti berhasil di approve, dan sistem menampilkan pesan "Leave submission approved!"	Berhasil
6.	Riwayat Cuti	Karyawan membuka halaman riwayat cuti	Klik menu "Leave Submission" kemudian "Submission Status"	Sistem menampilkan riwayat cuti yang sudah diajukan	Berhasil
7.	Pengajuan Lembur	Karyawan mengisi form pengajuan lembur dengan detail lembur	Tanggal lembur : 20/10/2025, Start OT: 18:00, End OT: 19:00, Duration : 1, Remark : perbaikan	Sistem menyimpan pengajuan dan menampilkan pesan "OT submission berhasil diajukan!"	Berhasil
8.	Persetujuan Lembur	HRD memilih salah satu data lembur, kemudian menekan tombol Approve	Klik tombol Approve	Lembur berhasil di approve, dan sistem menampilkan pesan "OT submission approved!"	Berhasil
9.	Riwayat Lembur	Karyawan membuka halaman riwayat lembur	Klik menu "OT Submission" kemudian "Submission Status"	Sistem menampilkan riwayat lembur yang sudah diajukan	Berhasil
10.	Pengajuan Perubahan Data Diri	Karyawan mengisi form pengajuan perubahan data diri dengan detail data yang mau diubah	New Employee Name: Adi, New Employee Email : <a href="mailto:adi@gmail.com">adi@gmail.com</a> , New Phone Number: 081290901234, New Address : Batu aji	Sistem menyimpan pengajuan dan menampilkan pesan "Personal data submission berhasil dikirim!"	Berhasil
11.	Persetujuan Perubahan Data Diri	HRD memilih salah satu data perubahan data diri, kemudian menekan tombol Approve	Klik tombol Approve	Pengajuan perubahan data diri berhasil di approve, dan sistem menampilkan pesan "Approved!"	Berhasil
12.	Riwayat Perubahan Data Diri	Karyawan membuka halaman riwayat lembur	Klik menu "Personal Data Submission" kemudian "Submission Status"	Sistem menampilkan riwayat perubahan data diri yang sudah diajukan	Berhasil
13.	Absensi	Karyawan memilih tipe absensi Slot In	Karyawan menekan tombol Slot In	Sistem memverifikasi lokasi dan menyimpan data absensi	Berhasil
14.	Proses Absensi	HRD menekan tombol "Process Attendance" di menu "Attendance List"	Tekan tombol "Process Attendance"	Sistem akan mengolah data absensi karyawan untuk menentukan jam masuk dan jam keluar	Berhasil
15.	Proses Penggajian	HRD memilih periode penggajian	Klik tombol "Posting"	Sistem akan menghitung gaji karyawan berdasarkan hari kerja dan lembur pada periode yang dipilih	Berhasil

#### IV. KESIMPULAN

Berdasarkan hasil dari studi dan pengembangan yang telah dilakukan, Sistem Informasi Manajemen Karyawan yang dirancang dengan menggunakan pendekatan *Research and Development* (R&D) serta model pengembangan *Waterfall* ini berhasil menyediakan solusi yang efektif untuk membantu proses pengelolaan data karyawan dalam perusahaan yang bergerak di bidang pengembangan perangkat lunak. Sistem ini mencakup fitur-fitur utama seperti *login*, absensi, pengajuan cuti, lembur, perubahan data pribadi, persetujuan cuti, lembur, dan pengubahan data diri, serta riwayat permohonan dan pengajian.

Pelaksanaan sistem ini memberikan kemudahan bagi karyawan dalam melakukan pengajuan secara *online*, dan juga mempermudah departemen HRD dalam proses persetujuan serta pemantauan data secara langsung. Selain itu, sistem ini berkontribusi dalam meningkatkan efisiensi dan akurasi dalam pengelolaan administrasi karyawan.

Hasil dari pengujian dengan menggunakan metode *Black Box Testing* menunjukkan bahwa semua fungsi sistem berfungsi sesuai dengan kebutuhan fungsional yang telah ditetapkan. Dengan demikian, sistem ini dinyatakan layak digunakan sebagai alat pendukung dalam manajemen karyawan di lingkungan perusahaan.

Secara keseluruhan, studi ini berhasil menciptakan sistem informasi yang terintegrasi, efisien, dan dapat dipercaya untuk mendukung kegiatan operasional perusahaan, terutama dalam meningkatkan efektivitas pengelolaan data dan proses administrasi karyawan.

#### DAFTAR PUSTAKA

- [1] M. Shahiduzzaman, "Digital Maturity in Transforming Human Resource Management in the Post-COVID Era: A Thematic Analysis," *Administrative Sciences*, vol. 15, no. 2, p. 51, 2025.
- [2] M. Fakhri, M. S. J. Irmawan, A. S. Alwi, I. Febriani Asril, N. Q. Ridha, and D. Fadhillatunisa, "Perancangan Sistem Informasi Manajemen Karyawan Berbasis Website dengan Metode Waterfall," *Jurnal MediaTIK: Jurnal Media Pendidikan Teknik Informatika dan Komputer*, vol. 6, no. 3, pp. 35–44, Sep. 2023.
- [3] A. F. P. Jeremy Panjaitan, "Perancangan Sistem E-reporting Menggunakan ReactJS dan Firebase," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. VII, no. 1, pp. 20-34, 2021.
- [4] S. Chen and D. Yu, "The impact of R&D effort on business model innovation: Evaluating chain mediation through collaboration breadth and depth," *PLOS ONE*, vol. 18, no. 6, p. e0286715, Jun. 2023, doi: 10.1371/journal.pone.0286715 [Diakses: 23 Oktober 2025].
- [5] H. D. Benington, "Production of Large Computer Programs," *Annals of the History of Computing*, vol. 5, no. 4, pp. 350–361, Oct. 1983.
- [6] W. W. Royce, "Managing the Development of Large Software Systems," in *Proceedings of IEEE WESCON*, Los Angeles, CA, USA, 1970, pp. 1–9.
- [7] Y. Bassil, "A Simulation Model for the Waterfall Software Development Life Cycle," arXiv preprint, arXiv:1205.6904, May 2012.
- [8] A. Saravanos and M. X. Curinga, "Simulating the Software Development Lifecycle: The Waterfall Model," arXiv preprint, arXiv:2308.03940, Aug. 2023.
- [9] M. Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3rd ed. Boston: Addison-Wesley, 2004.
- [10] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language User Guide*, 2nd ed. Boston: Addison-Wesley, 2005.
- [11] K. Afifah, Z. Azzahra, and A. Anggoro, "Analisis Teknik Entity-Relationship Diagram dalam Perancangan Database: Sebuah Literature Review," *INTECH*, vol. 3, pp. 18–22, 2022.
- [12] N. Wiener, *Cybernetics: Or Control and Communication in the Animal and the Machine*, 2nd ed., Cambridge, MA: MIT Press, 1961.
- [13] W. R. Ashby, *An Introduction to Cybernetics*, London, U.K.: Chapman & Hall Ltd., 1956.