

***EMPLOYEE PAYROLL SYSTEM DESIGN USING EXTREME PROGRAMMING ALGORITHM***

***PERANCANGAN SISTEM PENGGAJIAN KARYAWAN MENGGUNAKAN ALGORITMA EXTREME PROGRAMMING***

**Muhammad Luthfi Hamzah<sup>\*</sup>, Daffa Takratama Savra, Della Harmutika, Rahma Sani Nahampun, Ali Alamuddin Muzaffar, Anjasy Syahrani, Abdul Hamid**  
Universitas Islam Negeri Sultan Syarif Kasim Riau<sup>123456</sup>  
muhammad.luthfi@uin-suska.ac.id<sup>\*</sup>

Received: 15 June 2025, Revised: 23 November 2025, Accepted: 02 December 2025

*\*Corresponding Author*

**ABSTRACT**

*The employee payroll system was built to make it easier for companies, especially those that still handle payroll manually, to manage employee payroll data. Company X still records employee information, absence information, and salary information manually in a book as part of its employee payroll process. The design of the employee payroll system in this study uses the Extreme Programming method. The Extreme Programming approach is used in the design of the employee payroll system in this study. This study also tests the system. The system is tested using White-box and Black-box testing methodologies, with the aim of testing the functional and non-functional requirements of the system. The findings of this study lead to the creation of an employee payroll information system that Company X can use to facilitate the payroll process for its employees.*

**Keywords :** *Black-box, Extreme Programming, Employee Payroll ,White-box*

**ABSTRAK**

Sistem penggajian karyawan dibangun untuk memudahkan perusahaan khususnya yang masih menangani penggajian secara manual, untuk mengelola data penggajian karyawan. Perusahaan X masih mencatat informasi karyawan, informasi absensi, dan informasi gaji secara manual dalam sebuah buku sebagai bagian dari proses penggajian karyawannya. Perancangan sistem Penggajian karyawan pada penelitian ini menggunakan metode Extreme Programming. Pendekatan Extreme Programming digunakan dalam perancangan sistem penggajian karyawan pada penelitian ini. Penelitian ini juga menguji sistem. Sistem diuji menggunakan metodologi pengujian White-box dan Black-box, dengan tujuan menguji persyaratan fungsional dan non-fungsional sistem. Temuan studi ini mengarah pada pembuatan sistem informasi penggajian karyawan yang dapat digunakan Perusahaan X untuk memudahkan proses penggajian bagi karyawannya.

**Kata Kunci :** *Black-box, Extreme Programming, Penggajian Karyawan ,White-box*

**1. Introduction**

Perusahaan X adalah sebuah perusahaan yang proses penggajian masih dilakukan secara manual, dan data karyawan, data absensi, dan data penggajian semuanya disimpan dalam satu buku besar yang sama. Jika ada kesalahan butuh waktu lama untuk menghitung kembali gaji karyawan. Kemudian, karena disimpan dalam tumpukan kertas yang masih berbentuk dokumen, informasi penting bisa saja hilang. Sistem akuntansi kepegawaian dibutuhkan sebagai akibat dari masalah yang dihadapi. Tujuan pengembangan sistem penggajian karyawan adalah untuk mempermudah perusahaan mengelola data penggajian karyawan sehingga informasi tidak bertambah atau hilang karena penggunaan kertas. Dan bahkan di perusahaan X, semua karyawan masih meminta gaji langsung dari karyawan, sehingga pengisian gaji memakan waktu lama.

Menurut (Wibowo et al., 2013) yang melakukan studi literatur pada (Beck, 2000) menjelaskan *Extreme Programming (XP)* adalah metodologi yang menggunakan pendekatan berorientasi objek untuk membentuk kelompok kecil hingga menengah. Metode ini juga bisa dihadapkan pada kebutuhan pengguna yang tidak jelas atau permintaan yang tiba-tiba berubah.

Dan pada penelitian (Supriyatna & Informatika, 2018) yang merujuk pada penelitian (Wibowo et al., 2013) *Extreme Programming* adalah teknik pengembangan *software* yang paling umum digunakan. Tujuan *Extreme Programming* ini untuk mengatasi persyaratan yang tidak jelas dan persyaratan dengan sangat cepat. *Extreme Programming* adalah teknik yang sangat mudah beradaptasi untuk menghasilkan *prototype* saat dibutuhkan. (Fruhling et al., 2005) (Silalahi et al., 2021; Abrar et al., 2025; Pramudya et al., 2025).

Menurut penelitian yang dilakukan oleh (Asbar & Saptari, 2017) menyimpulkan keberadaan metode *pieces* dapat memberikan sebuah Figan pada perusahaan untuk mengamati karyawan secara langsung, meningkatkan perbaikan setiap orang secara individu dan mempermudah perusahaan. Penelitian (Supriyatna & Maria, 2017) yang menganalisis Sistem Informasi DJP *Online* menggunakan *Framework* PIECES, Sistem Informasi DJP online dapat memuaskan pengguna. Pendekatan PIECES digunakan dalam penelitian oleh (Supriyatna, 2015) yang menyimpulkan bahwa implementasi sistem ujian online kampus BSI Jatiwaringin berhasil dan disukai oleh pengguna. PIECES dapat menelaah sebuah sistem secara menyeluruh dan mencermati dengan seksama agar dapat ditentukan kelebihan dan kekurangan sistem tersebut dan dijadikan pedoman bagi perusahaan kedepannya. (Supriyatna, 2015; Sholikin et al., 2023; Fauzi et al., 2025).

Model analisis *Extreme Programming* dan *PIECES Framework* akan digunakan untuk mengembangkan sistem penggajian karyawan berdasarkan kesimpulan dari permasalahan yang telah dibahas dalam penelitian ini (Fajar et al., 2025; Luthfiya et al., 2025; Pandit et al., 2025).

## 2. Literature Review

### **Framework PIECES**

PIECES adalah sebuah *framework* yang dipakai untuk mendefinisikan *problem* desain sistem. Pengimplementasian *framework* ini menghasilkan sesuatu yang segar agar dibuat untuk pengembangan sistem dan diperhitungkan. (Tullah & Hanafri, 2014)(Supriyatna & Maria, 2017).

1. *Performance*, untuk menentukan seberapa baik atau buruknya suatu sistem berfungsi
2. *Information*, digunakan untuk menilai kejelasan informasi yang dihasilkan
3. *Economics*, yang mengkaji keuangan dan pengeluaran yang dikeluarkan untuk menentukan apakah sistem tersebut dapat diterapkan pada suatu instansi.
4. *Efficiency*, melakukan penelitian untuk menentukan apakah sistem yang dikembangkan dapat beroperasi secara efektif dan menghasilkan *output* yang baik dan memuaskan dengan input yang minimal
5. *Service*, layanan yang menilai seberapa besar keamanan yang diterapkan.

### **Algoritma Extreme Programming (XP)**

*Extreme Programming* sangat responsif pada perubahan (Sulianta, 2019) (Kustiawan et al., 2022). *Extreme Programming* dimulai dari *planning*, *design*, *coding*, dan *testing* (Suryantara et al., 2017).

1. *Planning*, melaksanakan rencana dengan mengidentifikasi masalah dengan sistem saat ini dan kemudian menilai kebutuhan pengguna untuk sistem yang dikembangkan
2. *Design*, desain dan pemodelan sistem berdasarkan analisis kebutuhan
3. *Coding*, membuat sistem untuk penggajian karyawan menggunakan *Framework Codeigniter*.
4. *Testing*, menguji sistem yang dibuat untuk mengetahui apakah sudah berfungsi dengan baik atau tidak. Pengujian *Black-Box Testing* dan *White-Box Testing* Teknik yang digunakan.

### **White-Box**

Pengujian transparan adalah nama lain dari pengujian *white-box* (Irawan, 2017). Untuk menemukan masalah pemrograman, digunakan *White-box* (Nidhra & Dondeti, 2012)(Paembonan, 2018). *White-box* memeriksa *code* pemrograman apakah input dan output memenuhi persyaratan yang diperlukan (Cholifah et al., 2018). Dalam pendekatan ini diperlukan

prosedur pengujian untuk membuat *flow graph* diagram program dengan tujuan menganalisis keakuratan struktur program yang dibuat (Rahayuda & Santiari, 2017).

### Black-Box

Penggunaan teknik *black-box*, data diperoleh dari kebutuhan fungsional yang telah ditentukan sebelumnya tanpa memperhitungkan struktur program akhir. (MZ, 2016). Pendekatan ini memberikan penekanan yang kuat pada pelaksanaan tugas dan memeriksa *input* dan *output* (Howden, 1986)(MZ, 2016).

### 3. Research Methods

Metode penelitian merupakan tahap-tahap yang dilakukan di penelitian ini, yang dimulai dari perencanaan sampai implementasi sistem. Disusun pada Fig 1.

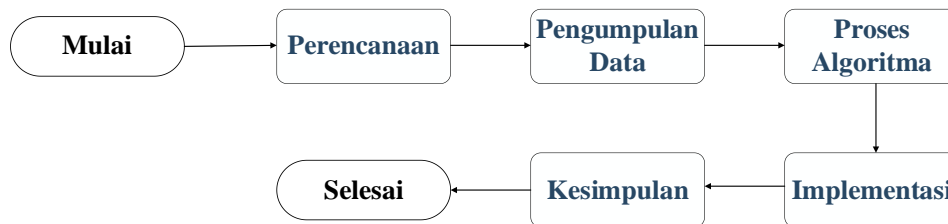


Figure 1. Metodologi

### 4. Results and Discussions

#### Planning

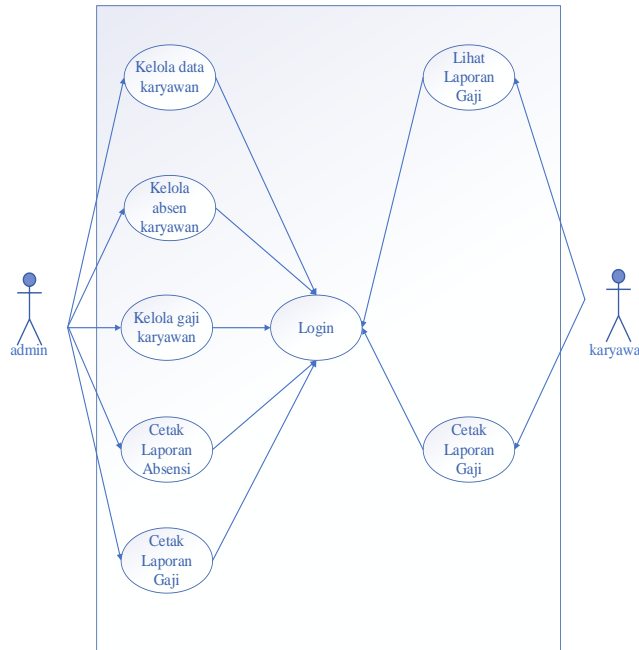
Pada tahap *planning* ini dilakukan analisa terhadap perusahaan X dengan menggunakan *Framework* PIECES. Disusun pada Table 1.

Table 1 – PIECES

No.	Analisis masalah	Permasalahan	Solusi
1.	<i>Performance</i>	Data dikumpulkan dengan cara mencatatnya dalam sebuah buku.	Merancang Sistem Penggajian Karyawan untuk memudahkan <i>admin</i> dan karyawan.
2.	<i>Information</i>	Karyawan sangat lambat mendapatkan informasi gaji karena harus menanyakan kepada <i>admin</i> terlebih dahulu dan membutuhkan waktu yang lama.	Sistem Penggajian Karyawan dirancang untuk memudahkan karyawan memperoleh slip gaji dengan cepat dan memungkinkan <i>admin</i> dapat <i>create</i> , <i>update</i> , dan <i>delete</i> data karyawan, jabatan, gaji, absensi.
3.	<i>Economy</i>	Perusahaan menyimpan informasi dalam sebuah buku. Perusahaan perlu membeli buku baru untuk memasukkan data setelah yang sekarang diisi. Oleh karena itu, perusahaan menghabiskan lebih banyak uang untuk buku baru.	Perusahaan ini memiliki simpanan yang awalnya digunakan untuk membeli buku. Dalam metode ini, perusahaan hanya menggunakan dana untuk menyewa hosting dan nama domain, yang lebih murah daripada membeli buku secara teratur.
4.	<i>Efficiency</i>	Input data tidak efisien karena mereka terus menggunakan buku besar, dan manajer kehilangan banyak waktu untuk melacak informasi, kehadiran, dan gaji karyawan. Karyawan juga membuang waktu mengantri untuk mendapatkan gaji mereka dari manajer.	Tujuan dari sistem penggajian karyawan adalah untuk mempercepat tugas-tugas administrasi dan mempermudah karyawan untuk menerima gaji tanpa mengantri.
5.	<i>Service</i>	Untuk mendapatkan slip gaji, karyawan harus bertanya kepada <i>admin</i> dan mengantri.	Ketersediaan sistem informasi penggajian pegawai ini dapat mempermudah pekerjaan <i>admin</i> dan pegawai.

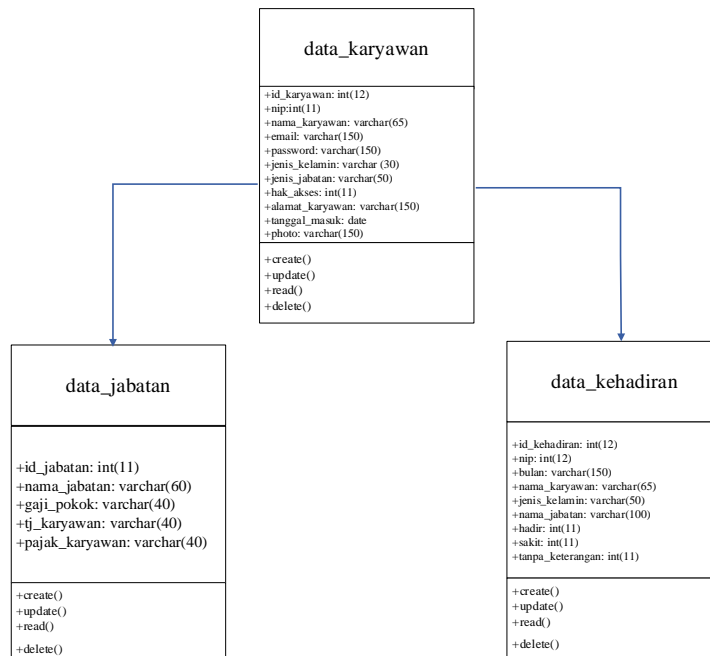
**Design**

Pada tahap ini dilakukan pembuatan *Use Case Diagram*. Berikut rancangan *Use Case Diagram* nya pada Fig 2.



**Figure 2. Use Case Diagram**

Peneliti menggambar *class diagram* untuk menggambarkan basis data sistem setelah mengidentifikasi *use case*. Tabel data pegawai, tabel data absensi, dan tabel data jabatan merupakan tiga tabel yang peneliti gunakan dalam sistem ini. *Class Diagram* ini disusun pada Fig dibawah.

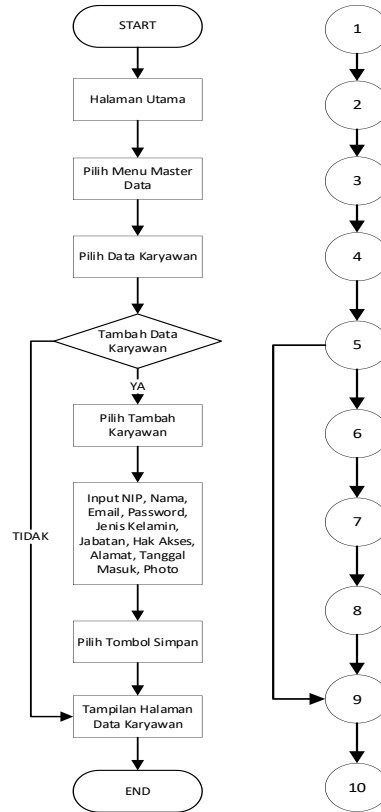


**Figure 3. Class Diagram**

**Testing**

Pengujian (*testing*) menggunakan metode *White-Box* dan *Black-box*. Berikut *White-Box* yang telah dirancang.

**White-Box  
Data Karyawan**



**Figure 4.** Flow Graph Input Data Karyawan

Kompleksitas siklomatis pada Fig 4. Flow Graph Data Karyawan kompleksitas siklomatis dihitung menggunakan cara:

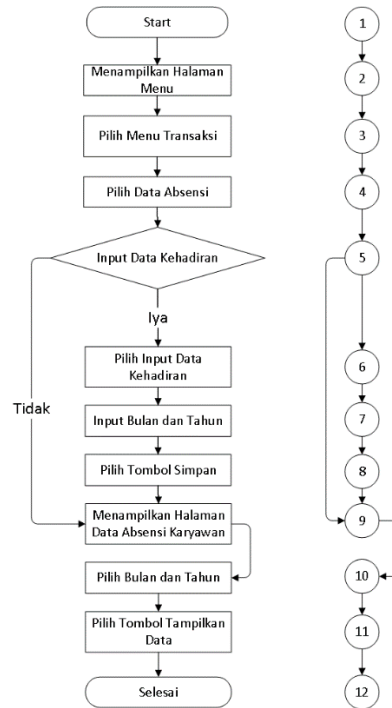
- V (G) = E - N + 2 = 10 - 10 + 2 = 2
- path 1 : 1-2-3-4-5-6-7-8-9-10
- path 2 : 1-2-3-4-5-9-10

**Table 2 – Test Case Data Karyawan**

Path	1
Jalur	1-2-3-4-5-6-7-8-9-10
Skenario	1. Mulai 2. Halaman Utama 3. Pilih menu Master data 4. Pilih data karyawan 5. Tambahkan data Karyawan? 6. Jika ya, pilih tambah data karyawan 7. Input NIP, Nama, Email, Password, Jenis Kelamin, Jabatan, Hak Akses, Alamat, Tanggal Masuk, Photo 8. Pilih tombol simpan 9. Menampilkan halaman data karyawan. 10. Selesai
Hasil pengujian	Berhasil
Path	2

Jalur	1-2-3-4-5-9-10
Skenario	1. Mulai 2. Halaman utama 3. Pilih menu Master data 4. Pilih data karyawan 5. Tambahkan data karyawan? 9. Jika tidak, menampilkan halaman data karyawan 10. Selesai
Hasil Pengujian	Berhasil

**Data Absensi Karyawan**



**Figure 5. Flow Graph Data Absensi Karyawan**

Kompleksitas siklomatis pada Fig 5. *Flow Graph Data Absensi Karyawan* kompleksitas siklomatis dihitung menggunakan cara:

$$V(G) = E - N + 2 = 10 - 10 + 2 = 2$$

Path = 1 ; 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12

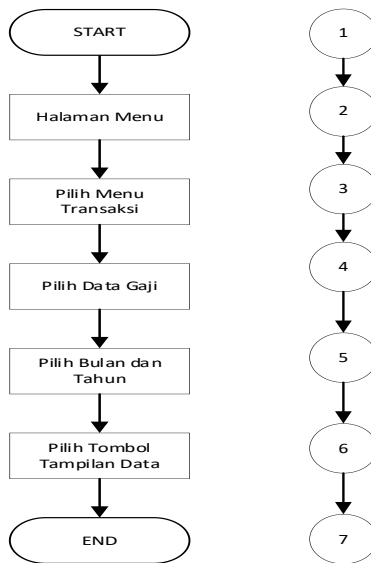
Path = 2 ; 1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 12

**Table 3 – Test Case Data Absensi Karyawan**

Path	1
Jalur	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 - 12
Skenario	1. Mulai 2. Menampilkan halaman menu 3. Pilih menu transaksi 4. Pilih data absensi 5. Input data kehadiran ? 6. Jika iya, pilih input data kehadiran 7. Input bulan dan tahun 8. Pilih tombol simpan 9. Menampilkan halaman data absensi karyawan 10. Pilih bulan dan tahun 11. Pilih tombol tampilkan data 12. Selesai
Hasil Pengujian	Berhasil

Path	<b>2</b>
Jalur	1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 12
Skenario	<ol style="list-style-type: none"> <li>1. Mulai</li> <li>2. Menampilkan halaman menu</li> <li>3. Pilih menu transaksi</li> <li>4. Pilih data absensi</li> <li>5. Input data kehadiran ?</li> <li>9. Jika tidak, menampilkan halaman data absensi karyawan</li> <li>10. Pilih bulan dan tahun</li> <li>11. Pilih tombol tampilkan data</li> <li>12. Selesai</li> </ol>
Hasil Pengujian	Berhasil

**Data Gaji Karyawan**



**Figure 6.** Flow Graph Data Gaji Karyawan

Kompleksitas siklomatis pada Fig 6. Flow Graph Data Gaji Karyawan kompleksitas siklomatis dihitung menggunakan cara:

$V(G) = E - N + 2 = 8 - 7 + 2 = 3$   
 path 1 : 1-2-3-4-5-6-7

**Table 4 – Test Case Data Gaji Karyawan**

Path	<b>1</b>
Jalur	1-2-3-4-5-6-7
Skenario	<ol style="list-style-type: none"> <li>1. Mulai</li> <li>2. Halaman menu</li> <li>3. Pilih menu transaksi</li> <li>4. Pilih data gaji</li> <li>5. Pilih bulan &amp; tahun</li> <li>6. Pilih tombol tampilan data</li> <li>7. Selesai</li> </ol>
Hasil pengujian	Berhasil

**Cetak Laporan Gaji Karyawan**



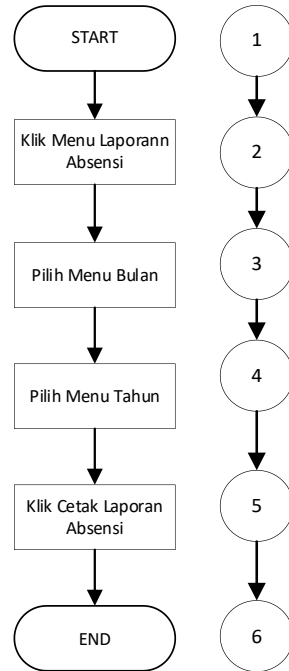
**Figure 7.** Flow Graph Cetak Laporan Gaji Karyawan

Kompleksitas siklomatis pada Fig 7. Flow Graph Cetak Laporan Gaji Karyawan kompleksitas siklomatis dihitung menggunakan cara:

$V(G) = E - N + 2 = 6 - 5 + 2 = 3$   
 Path = 1 ; 1 - 2 - 3 - 4 - 5 - 6

Path	1
Jalur	1 - 2 - 3 - 4 - 5 - 6
Skenario	1. Mulai 2. Klik menu laporan gaji 3. Pilih menu bulan 4. Pilih menu tahun 5. Klik cetak laporan gaji 6. End
Hasil Pengujian	Berhasil

**Cetak Laporan Absensi Karyawan**



**Figure 8.** Flow Graph Cetak Laporan Absensi Karyawan

Kompleksitas siklomatis pada Fig 3. Flow Graph Laporan Absensi Karyawan kompleksitas siklomatis dihitung menggunakan cara:

**Table 7 – Test Case Cetak Laporan Absensi Karyawan**

Jalur	1 – 2 – 3 – 4 – 5 – 6
Skenario	1. Start 2. Klik menu laporan absensi 3. Pilih menu bulan 4. Pilih menu tahun 5. Klik cetak laporan absensi 6. End
Hasil Pengujian	Berhasil

**Black-Box**

**Table 8 – Black-box**

Form atau Halaman	Hasil yang diharapkan	Hasil Aktual (Keluaran)	Keterangan
<b>Login</b>	Masuk ke halaman utama/dashboard sesuai hak akses user	Sistem menampilkan halaman utama/dashboard sesuai hak akses user	Berhasil
<b>Dashboard</b>	Menampilkan halaman data admin, data karyawan, data jabatan dan data absensi	Sistem menampilkan halaman data admin, data karyawan, data jabatan dan data absensi	Berhasil
<b>Data Karyawan</b>	Menambahkan data karyawan, update dan menghapus data absensi	Sistem dapat menambahkan data karyawan, update dan menghapus data absensi	Berhasil
<b>Data Jabatan</b>	Menambahkan data jabatan, update, dan menghapus data jabatan	Sistem dapat menambahkan data jabatan, update, dan menghapus data jabatan	Berhasil

<b>Data Absensi</b>	Menginput dan menampilkan data kehadiran	Sistem dapat menginput dan menampilkan data kehadiran	Berhasil
<b>Potongan Gaji</b>	Menambah dan menghapus data potongan gaji	Sistem dapat menambah dan menghapus data potongan gaji	Berhasil
<b>Data Gaji Karyawan</b>	Menampilkan data gaji karyawan	Sistem dapat menampilkan data gaji karyawan	Berhasil
<b>Laporan gaji</b>	Mencetak laporan gaji karyawan	Sistem dapat mencetak laporan gaji karyawan	Berhasil
<b>Laporan Absensi</b>	Mencetak laporan Absensi	Sistem dapat mencetak laporan absensi	Berhasil
<b>Ubah Password</b>	Mengubah password	Sistem dapat mengubah password	Berhasil
<b>Log out</b>	Keluar dari sistem	Sistem dapat keluar (log out) dari sistem	Berhasil

## 5. Conclusion

Berdasarkan pembahasan penelitian diatas, kesimpulannya adalah sistem informasi penggajian karyawan yang dirancang pada perusahaan X mampu meningkatkan kinerja pada perusahaan karena sistem ini memudahkan dalam pengelolaan dan proses penggajian karyawan. Hasil dari pengujian sistem informasi penggajian menggunakan metode blackbox dan whitebox adalah semua fitur yang ada pada sistem dapat berfungsi dengan baik. Dengan demikian, sistem ini dapat digunakan untuk membantu perusahaan x dalam pengelolaan serta proses penggajian karyawan.

## References

- Abrar, M. F., Alferaidi, A., Almurayziq, T. S., Saqib, M., Khan, W., Khan, Z., & Alsaffar, M. (2025). Enhancing Extreme Programming (XP) Adoption through SAMAM: A Scalable Agile Maturity Assessment Model Based on Industry Best Practices. *IEEE Access*.
- Asbar, Y., & Saptari, M. A. (2017). Analisa dalam mengukur kualitas pelayanan terhadap kepuasan konsumen menggunakan metode PIECES. *Jurnal Visioner & Strategis*, 6(2).
- Beck, K. (2000). *Extreme programming explained: embrace change*. addison-wesley professional.
- Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2018). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 3(2), 206–210.
- Fajar, F. S., Samsudin, S., & Ilyas, I. (2025). Analysis of Zoom Meeting User Satisfaction Using the Pieces Framework at Islamic University of Indragiri. *Digital Informatics*, 1(2), 65-74.
- Fauzi, I. M., Nugroho, A. A., & Khoirunisa, A. (2025). Evaluation of SAPA WARGA Application System and User Satisfaction: Public Service Technology Management Perspective Based on PIECES Framework. *JURNAL SISFOTEK GLOBAL*, 15(1), 34-41.
- Fruhling, A., Tyser, K., & De Vreede, G.-J. (2005). Experiences with extreme programming in telehealth: Developing and implementing a biosecurity health care application. *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, 151b-151b.
- Howden, W. E. (1986). *Functional program testing and analysis*. McGraw-Hill, Inc.
- Irawan, Y. (2017). Pengujian Sistem Informasi Pengelolaan Pelatihan Kerja UPT BLK Kabupaten Kudus dengan Metode Whitebox Testing. *Senra Penelitian Engineering Dan Edukasi*, 9(3), 59–63.

- Kustiawan, D., Cholifah, W. N., Destriana, R., & Heriyani, N. (2022). Rancang Bangun Sistem Informasi Akuntansi Pengelolaan Koperasi Menggunakan Metode Extreme Programming. *Jurnal Teknologi Dan Informasi*, 12(1), 78–92.
- Luthfiya, P. C., Nugroho, A. B., Maulana, R. R., & Priatna, R. (2025, September). Evaluating the Performance of “Sitata”(Our Land Validation) Application Using the Pieces Framework at the Bandung City Land Office. In *Proceeding International Conference on Economics, Business and Information Technology* (Vol. 6, No. 1, pp. 705-713).
- MZ, M. M. K. (2016). Pengujian perangkat lunak metode black-box berbasis equivalence partitions pada aplikasi sistem informasi sekolah. *MIKROTIK: Jurnal Manajemen Informatika*, 6(1).
- Nidhra, S., & Dondeti, J. (2012). Black box and white box testing techniques-a literature review. *International Journal of Embedded Systems and Applications (IJESA)*, 2(2), 29–50.
- Paembonan, S. (2018). Implementasi White Box Pada Informasi Akademik Smp Negeri 1 Bupon. *PENA TEKNIK: Jurnal Ilmiah Ilmu-Ilmu Teknik*, 3(2), 177–184.
- Pandit, D. P., Gajjam, N. S., Sangu, V. S., Chandra, S., & Chandrasatheesh, C. (2025). Optimizing Efficiency and Delivering Quality for Lean and Extreme Programming (XP) in Agile Business Methodologies. In *Impact of Digital Transformation on Business Growth and Performance* (pp. 547-578). IGI Global Scientific Publishing.
- Pramudya, B., Ramadhani, D. C. P., Mujaddidah, H. N., & Pradini, R. S. (2025). Implementation of Extreme Programming (XP) in the Development of Dental Clinic Information Systems. *Journal of Enhanced Studies in Informatics and Computer Applications*, 2(1), 20-28.
- Rahayuda, I. G. S., & Santiari, N. P. L. (2017). Basis Path Testing of Iterative Deepening Search and Held-Karp on Pathfinding Algorithm. *Jurnal Ilmiah Kursor*, 9(2).
- Sholikin, M., Sedyono, E., & Hendry, H. (2023). Application of the PIECES Framework Method in E-Report Evaluation. *Jurnal Penelitian Pendidikan IPA*, 9(SpecialIssue), 438-447.
- Silalahi, M. A., Madaniyah, S. S., Lisnawati, L., & Adnin, S. S. (2021). Model Extreme Programming: Rancang Bangun Sistem Informasi Penggajian Karyawan PT. Gemilang Lestari Teknindo. *Jurnal Sistem Informasi Akuntansi (JASIKA)*, 1(2), 126–138.
- Sulianta, F. (2019). *Strategi Merancang Arsitektur Sistem Informasi Masa Kini*. Elex Media Komputindo.
- Supriyatna, A. (2015). Analisis dan evaluasi penerapan aplikasi ujian berbasis web dengan metode pieces framework. *Swabumi*, 3(1), 1–15.
- Supriyatna, A., & Informatika, M. (2018). Metode Extreme Programming pada pembangunan WEB aplikasi seleksi peserta pelatihan kerja. *Jurnal Teknik Informatika*, 11(1), 1–18.
- Supriyatna, A., & Maria, V. (2017). Analisa tingkat kepuasan pengguna dan tingkat kepentingan penerapan sistem informasi djp online dengan kerangka pieces. *Khazanah Informatika: Jurnal Ilmu Komputer Dan Informatika*, 3(2), 88–94.
- Suryantara, I. G. N., Kom, S., & Kom, M. (2017). *Merancang Aplikasi dengan Metodologi Extreme Programming*. Elex Media Komputindo.
- Tullah, R., & Hanafri, M. I. (2014). Evaluasi penerapan sistem informasi pada Politeknik LP3I Jakarta dengan metode PIECES. *Jurnal Sisfotek Global*, 4(1).
- Wibowo, S. A., Sholiq, S., & Muqtadiroh, F. A. (2013). Rancang Bangun Aplikasi Web Informasi Eksekutif Pada Pemerintah Kabupaten XYZ. *Jurnal Teknik ITS*, 2(3), A476–A480.