

# Penerapan Extreme Programming dalam Rancang Bangun Sistem Informasi Inventaris Berbasis Website pada Dinas Komunikasi dan Informatika Kabupaten Kerinci

Eastyanda Putra Kesuma<sup>1</sup>, Ulfa Khaira<sup>2</sup>, Muhammad Razi A<sup>3</sup>

<sup>1,2,3</sup>Information System, Universitas Jambi, Indonesia

[eastyandaputra@gmail.com](mailto:eastyandaputra@gmail.com)<sup>1</sup>, [ulfakhaira@unja.ac.id](mailto:ulfakhaira@unja.ac.id)<sup>2</sup>, [muhammadrazi@unja.ac.id](mailto:muhammadrazi@unja.ac.id)<sup>3</sup>

---

## Article Info

### Article history:

Received Jan 13, 2025

Revised Nov 01, 2025

Accepted Nov 23, 2025

---

### Keywords:

Inventory Management System  
Web-Based Application  
Extreme Programming  
User Acceptance Testing  
Performance Testing

---

## ABSTRACT

Inventory management at the Communication and Informatics Department (Diskominfo) of Kerinci Regency was constrained by centralized processes and the functional limitations of the desktop-based Regional Management Information System (SIMDA). These issues compromised operational efficiency, data accuracy, and real-time accessibility. To address this, a web-based Inventory Information System was developed using the Extreme Programming (XP) methodology. This system provides a distributed platform enabling remote data access and features better aligned with user needs. Development proceeded in two iterations: the first covered basic inventory tracking, while the second implemented asset condition reporting, data validation, and comprehensive reporting, capabilities previously lacking in SIMDA. User Acceptance Testing (UAT) and performance testing confirmed that the system meets requirements and significantly improves inventory management efficiency. Consequently, this implementation enhances data transparency, operational efficiency, and public service quality at Diskominfo Kerinci.

Copyright © 2025 Universitas Indraprasta PGRI.  
All rights reserved.

---

## Corresponding Author:

Ulfa Khaira,  
Program Studi Sistem Informasi,  
Universitas Jambi,  
Jl. Jambi – Muara Bulian No.KM. 15, Mendalo Darat, Kec. Jambi Luar Kota, Kabupaten Muaro Jambi,  
Jambi  
Email: [ulfakhaira@unja.ac.id](mailto:ulfakhaira@unja.ac.id)

---

## 1. PENDAHULUAN

Pengelolaan inventaris di Dinas Komunikasi dan Informatika (Diskominfo) Kabupaten Kerinci menghadapi berbagai kendala yang menghambat kinerja organisasi. Masalah utama mencakup pengelolaan data yang masih terpusat pada Badan Pengawasan Keuangan dan Pembangunan (BPKP) dan penggunaan aplikasi Sistem Informasi Manajemen Daerah (SIMDA) yang berbasis desktop yang terbukti tidak fleksibel dan tidak memenuhi kebutuhan operasional Organisasi Perangkat Daerah (OPD) secara spesifik. Situasi ini berdampak langsung pada ketidakefisienan proses input data, keterlambatan penyajian laporan, dan kesulitan dalam memantau aset secara real-time. Oleh karena itu, Diskominfo Kabupaten Kerinci membutuhkan solusi sistem informasi inventaris berbasis web yang dirancang khusus untuk mengatasi tantangan tersebut dan menjadi pelopor digitalisasi pengelolaan aset di tingkat OPD.

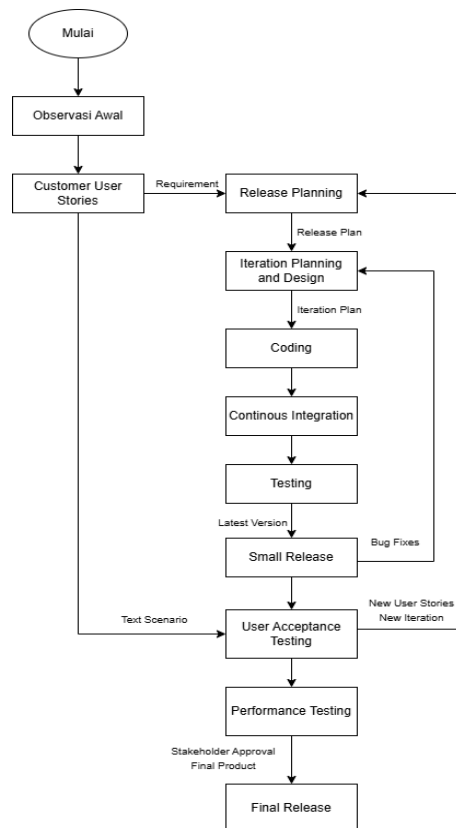
Untuk menghasilkan sistem yang adaptif, sesuai kebutuhan pengguna, dan dapat dikembangkan secara

cepat, penelitian ini mengadopsi model Extreme Programming (XP), yang merupakan bagian dari metodologi Agile. XP dipilih karena keunggulannya dalam menjamin keterlibatan user secara intensif dalam pengembangan dan menekankan pengujian terus-menerus melalui pendekatan iterasi. Pendekatan ini sangat cocok untuk memastikan bahwa perangkat lunak yang dihasilkan memiliki kualitas tinggi dan secara langsung memenuhi kepuasan serta kebutuhan pengguna akhir di Diskominfo [1, 2].

Penelitian-penelitian sebelumnya telah menunjukkan efektivitas metode Agile, khususnya XP, dalam pengembangan sistem informasi berbasis website. Misalnya, penelitian [3] dan [4] membuktikan keberhasilan penggunaan XP dalam mempercepat pengolahan data inventaris dan secara signifikan mengurangi tingkat kesalahan pencatatan di sektor usaha. Studi lain [5, 6, 7] juga menegaskan bahwa XP mampu meningkatkan efisiensi pengelolaan layanan, mempercepat pencarian data secara real-time, dan menghasilkan sistem yang fleksibel serta terintegrasi. Namun, penelitian-penelitian terdahulu tersebut umumnya berfokus pada sektor usaha kecil, manajemen barang tertentu, atau institusi non-pemerintahan. Oleh karena itu, terdapat gap riset terkait penerapan model pengembangan Agile (XP) secara spesifik dalam konteks pengelolaan inventaris berbasis website di lingkungan lembaga pemerintahan daerah (OPD), yang memiliki kompleksitas regulasi dan jenis aset yang berbeda. Penelitian ini hadir untuk menjembatani gap tersebut.

Berdasarkan latar belakang dan identifikasi gap riset, tujuan utama dari penelitian ini adalah merancang dan membangun Sistem Informasi Inventaris berbasis website yang fungsional menggunakan metode Extreme Programming (XP), serta mengevaluasi fungsionalitas sistem dalam meningkatkan efisiensi operasional, optimalisasi persediaan, dan transparansi data inventaris pada Dinas Komunikasi dan Informatika Kabupaten Kerinci. Kontribusi signifikan dari penelitian ini terletak pada hasil rancang bangun sistem yang tidak hanya meningkatkan efisiensi operasional internal Diskominfo, tetapi juga memastikan transparansi dalam pengelolaan aset publik. Dengan keberhasilan implementasi ini, Diskominfo Kabupaten Kerinci diharapkan dapat menjadi pelopor pengelolaan inventaris berbasis website yang dapat direplikasi oleh OPD lain di Kabupaten Kerinci. Implementasi sistem ini pada akhirnya bertujuan untuk mendukung optimalisasi sumber daya, mengurangi beban administrasi manual, dan meningkatkan kualitas pelayanan publik yang lebih akuntabel.

**2. METODE**



Gambar 1. Kerangka Kerja Penelitian

## A. Observasi Awal

Pada penelitian ini, peneliti terlebih dahulu melakukan observasi ke tempat penelitian. Observasi terhadap permasalahan pada proses manajemen pengelolaan barang-barang inventaris yang dihadapi Dinas Komunikasi dan Informatika Kabupaten Kerinci. Selanjutnya dilakukan wawancara terhadap Kepala Dinas dan Staf Pegawai untuk menggali informasi lebih dalam seputar permasalahan tersebut.

B. *Planning*

*Planning* merupakan tahapan pertama untuk mendengarkan dan mengumpulkan kebutuhan aktifitas suatu sistem yang memungkinkan pengguna agar memahami proses bisnis yang berjalan terhadap sistem yang dibuat dan mendapatkan gambaran yang jelas mengenai fitur utama, fungsionalitas dan keluaran yang diinginkan [8]. Penulis membuat perancangan terkait fitur-fitur yang akan diintegrasikan ke dalam aplikasi. Proses ini melibatkan analisis menyeluruh terhadap kebutuhan pengguna dan cara aplikasi dapat digunakan secara optimal. Fitur-fitur ini mencakup segala aspek, mulai dari antarmuka pengguna hingga fungsionalitas aplikasi. Fungsi aplikasi juga menjadi fokus utama pada tahap ini. Penulis berusaha memahami bagaimana setiap fitur berinteraksi satu sama lain dan bagaimana keseluruhan aplikasi akan beroperasi secara menyeluruh.

C. *Design*

*Design* merupakan tahapan dimana pada tahapan ini dilakukan kegiatan pemodelan yang dimulai dari pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan basis data [9]. Tahap ini akan penulis mulai dengan perancangan *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram*, lalu terakhir perancangan *design high fidelity* dari sistem yang berupa desain *User Interface* dan *User Experience* (UI/UX).

1. *Use Case Diagram*: Diagram yang menggambarkan hubungan antara sistem dan pengguna, atau aktor, dalam sebuah perangkat lunak untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu [10].
2. *Class Diagram*: Hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem [11].
3. *Activity Diagram*: Diagram yang menggambarkan berbagai aksi yang dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari aksi tersebut [12].
4. *Sequence Diagram*: Diagram yang memperlihatkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu [13].
5. *High Fidelity UI/UX Design*: Desain tampilan dari *website* yang akan dibangun, mulai dari pewarnaan, *layout*, dan lain sebagainya.

D. *Coding*

*Coding* merupakan tahapan dimana rancangan yang sudah dibuat diimplementasikan menjadi sebuah kode-kode pemrograman [14]. Pada tahap ini, Penulis melakukan proses pengkodean fitur-fitur sesuai dengan kebutuhan yang telah diidentifikasi. Melibatkan penerapan konsep-konsep XP seperti *Continuous Integration* untuk memastikan kualitas dan konsistensi kode.

E. *Testing*

*Testing* merupakan tahapan untuk menguji fungsionalitas sistem untuk mengetahui apakah sistem sudah sesuai dengan yang diharapkan oleh pengguna, dan untuk mengetahui apakah ada kesalahan lain yang timbul pada sistem ketika dijalankan [15]. Pada tahap ini, penulis menggunakan 2 pengujian yaitu:

- *User Acceptance Testing* (UAT)

UAT merupakan sebuah bentuk pengujian yang dilakukan oleh *end user*, dimana *end user* adalah pengguna yang akan langsung berinteraksi menggunakan sistem dan melakukan verifikasi terkait dengan fungsi apakah sudah berjalan dengan baik dan sesuai dengan fungsi beserta kebutuhan [16]. Pada pengujian ini, akan dilakukan dengan memberikan kuesioner yang berisi 20 pertanyaan dengan 5 kriteria jawaban dari Sangat Setuju – Sangat Tidak Setuju yang dimana setiap jawaban akan memiliki bobot nilai, kepada pegawai di Dinas Komunikasi dan Informatika Kabupaten Kerinci.

Setelah *user* mengisi kuesioner pengujian, jawaban yang ada akan dihitung untuk mendapatkan persentase nilai tersebut dengan rumus:

$$\text{Nilai rata-rata} = \frac{\text{jumlah bobot nilai responden}}{\text{total responden}}$$

$$\text{Persentase} = \frac{\text{jumlah bobot nilai responden}}{\text{total responden}} \times 100\%$$

- *Performance Testing*

*Performance testing* merupakan teknik pengujian untuk mengukur performansi *software* berdasarkan *performance metric*, yaitu *respon time*, *load data*, *traffic*, ketahanan dan stabilitas ketika diakses secara bersama oleh banyak *request* dan *resource* [17]. *Performance testing* ini nantinya akan menggunakan *tools* yang bernama JMeter karena JMeter dapat digunakan pada banyak sistem operasi dan kompatibel dalam menangani berbagai macam protokol. Skenario yang akan dilakukan adalah:

- *Normal Load Test*
  - *Number of threads*: 100,300,500,1000
  - *Ramp-up Period*: 30 detik
  - *Loop Count*: 10
- *Spike Test*
  - *Number of threads*: 50, 500
  - *Ramp-up Period*: 10, 0 detik
  - *Loop Count*: 1
- *Soak Test*
  - *Number of threads*: 100
  - *Ramp-up Period*: 30 detik
  - *Loop Count*: Forever
  - *Duration*: 2 jam

F. *Continuous Integration*

*Continuous Integration* (CI) adalah praktik pengembangan perangkat lunak di mana kode baru yang dikembangkan oleh para pengembang secara otomatis digabungkan ke dalam repositori utama secara berkala dengan tujuan untuk mendeteksi masalah integrasi lebih awal dengan mengotomatiskan proses *build*, *testing*, dan verifikasi kode [18]. Pada tahap ini, penulis menggunakan Github untuk melakukan proses integrasi.

G. *Small Release*

*Small Release* dalam metode *Extreme Programming* (XP) mengacu pada praktik merilis perangkat lunak dalam interval waktu yang pendek dan teratur. Dalam XP, tim pengembangan berfokus pada pengiriman produk perangkat lunak yang berfungsi kepada pelanggan atau pengguna dalam periode waktu yang singkat.

H. *Final Release*

*Final Release* merupakan rilis akhir dari perangkat lunak yang dihasilkan dalam siklus pengembangan. Setelah melalui serangkaian *small release* dan menerima feedback serta melakukan perbaikan, pengembangan mencapai titik di mana produk dianggap cukup matang dan memenuhi persyaratan yang ditetapkan sehingga menghasilkan rilis versi final yang menandai akhir dari iterasi pengembangan dan perangkat lunak siap untuk digunakan oleh klien atau pengguna secara penuh.

### 3. HASIL DAN PEMBAHASAN

A. *User Stories*

*User stories* adalah kegiatan penulis untuk mendapatkan kebutuhan pengguna dalam pengembangan sistem ini. Dimana dimulai dengan pengumpulan data melalui wawancara dengan tujuan mendengarkan kebutuhan pengguna. Dalam penelitian ini, penulis menemui dan melakukan wawancara pada pemangku kepentingan dari aplikasi yaitu Ibu Kepala Bidang Layanan E-Government Diskominfo Kabupaten Kerinci, Ibu Olra Nofratilova, S.Psi. Pertemuan wawancara ini juga merupakan salah satu cara mengimplementasikan *value* dari metode *extreme programming* yaitu *communication*. Hasil kebutuhan pengguna yang didapat dari wawancara akan dibagi menjadi beberapa iterasi.

B. Tahapan Iterasi

1. *Planning*

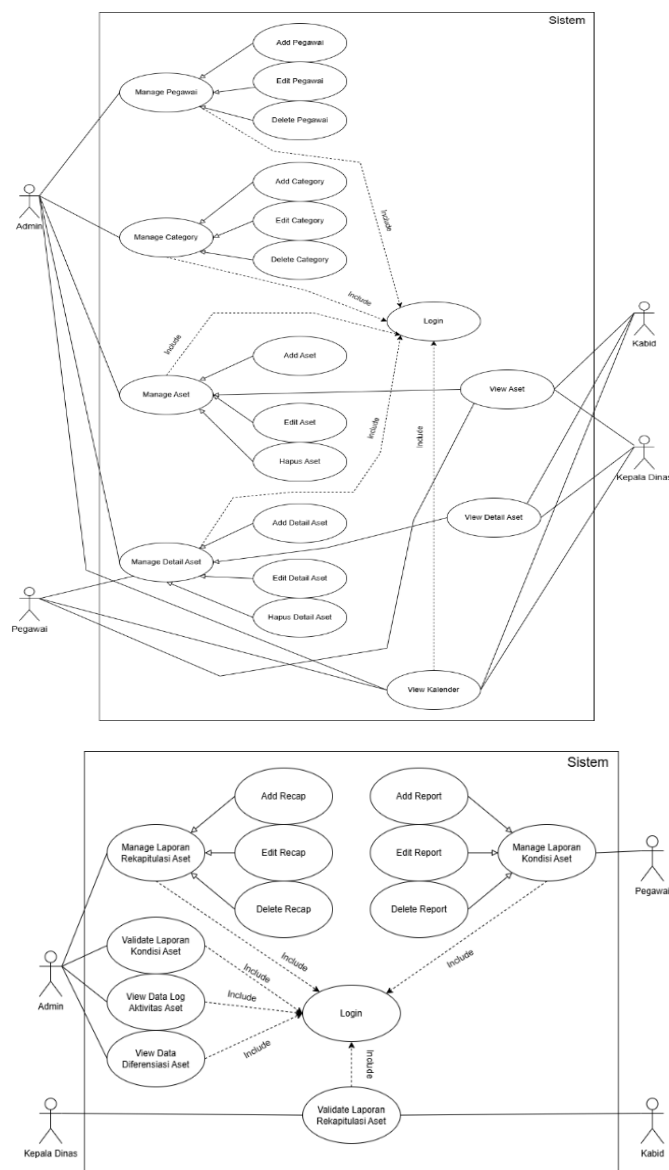
Berdasarkan tahapan pengumpulan data melalui wawancara mendalam dengan staf di Dinas Komunikasi dan Informatika Kabupaten Kerinci, serangkaian kebutuhan fungsional sistem berhasil diidentifikasi. Guna memastikan pengembangan dapat dilakukan secara adaptif dan terfokus sesuai prinsip *Extreme Programming* (XP), keseluruhan kebutuhan pengguna ini kemudian dipisahkan menjadi dua (2) iterasi pengembangan utama. Pemisahan ini didasarkan pada prioritas fungsionalitas dan ketergantungan antar-modul. Rincian lengkap mengenai kebutuhan pengguna pada setiap iterasi, yang menjadi dasar perancangan sistem, disajikan pada Tabel 1.

Tabel 1. Iterasi

Iterasi I	Iterasi II
<ul style="list-style-type: none"> <li>• Login</li> <li>• Kelola data kategori</li> <li>• Kelola detail barang</li> <li>• Logout</li> <li>• Kelola data barang inventaris</li> <li>• Kalender Aset</li> <li>• Kelola data pegawai</li> </ul>	<ul style="list-style-type: none"> <li>• Depresiasi aset</li> <li>• Pelaporan kondisi aset</li> <li>• Log aktivitas aset</li> <li>• Pelaporan rekapitulasi aset</li> </ul>

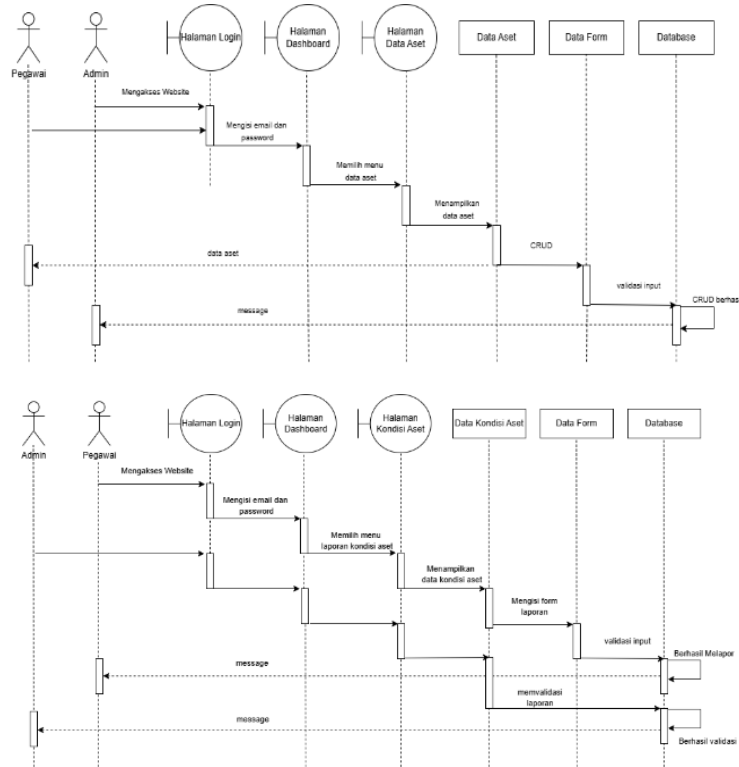
2. *Design*

Setelah dilakukan *planning*, selanjutnya membuat rancangan berdasarkan analisa kebutuhan fungsional yang telah dilakukan. Hasil rancangan berupa *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram*, dan *high fidelity ui/ux*.



Gambar 2. Use Case Diagram Iterasi-1 dan Iterasi-2

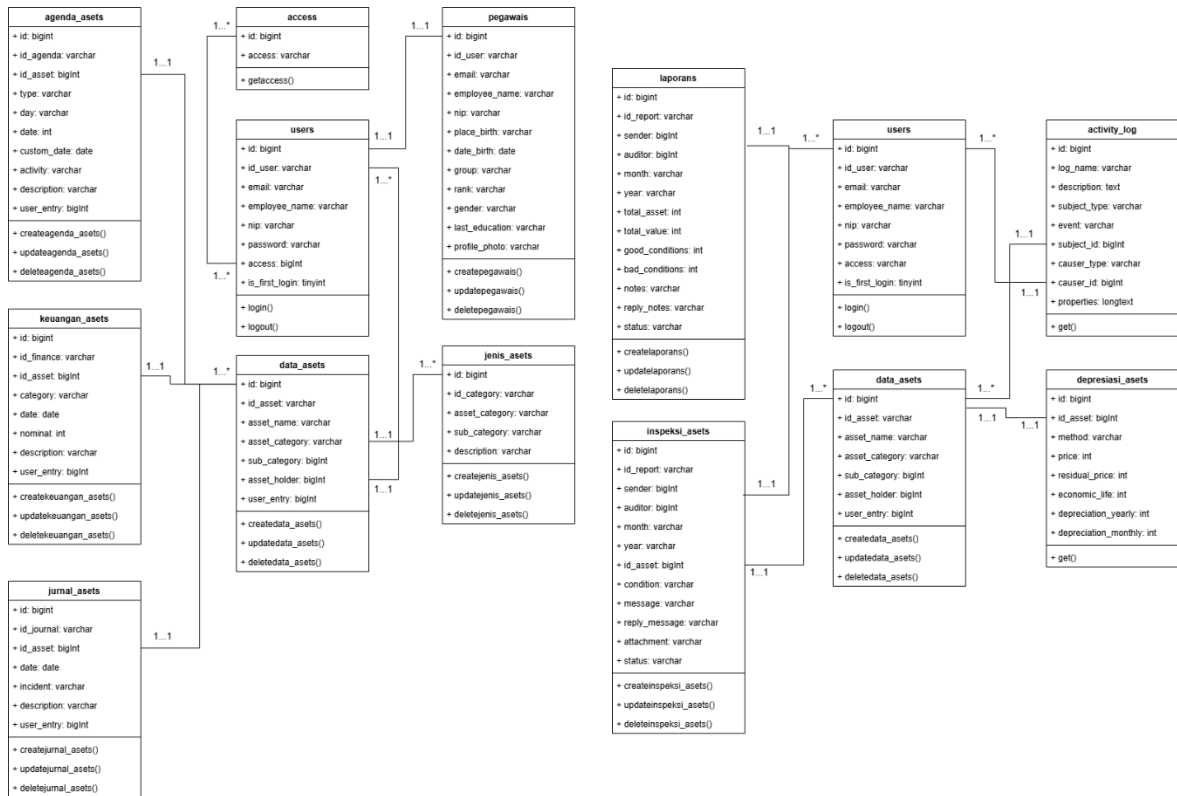
Pada sistem ini terdapat 4 aktor: Admin, Pegawai, Kabid, dan Kepala Dinas. **Iterasi 1:** Admin dapat mengelola data pegawai, kategori, barang aset, dan detail aset, sementara aktor lainnya hanya dapat melihat data tersebut. **Iterasi 2:** Pegawai dapat melaporkan kondisi aset untuk divalidasi oleh Admin. Admin juga dapat melaporkan rekapitulasi aset yang akan divalidasi oleh Kabid dan Kepala Dinas.



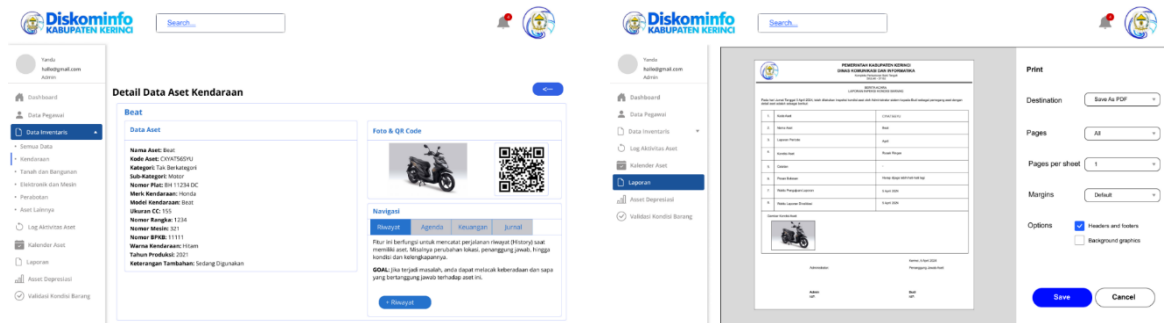
Gambar 3. Sequence Diagram Iterasi-1 dan Iterasi-2

Diagram sequence ini menggambarkan interaksi pengguna dengan sistem inventaris berdasarkan hak akses. Admin memiliki kontrol penuh untuk melakukan proses CRUD (*create, read, update, delete*), memungkinkan pengelolaan data barang inventaris secara menyeluruh, termasuk menambah, mengubah, dan menghapus data. Pengguna lain, seperti Pegawai, hanya memiliki akses untuk melihat data (*read*) tanpa izin untuk melakukan perubahan. Selain itu, Pegawai dapat melaporkan kondisi aset, termasuk informasi keadaan dan statusnya. Laporan tersebut kemudian divalidasi oleh Admin untuk memastikan akurasi dan kelengkapan data. Diagram ini menekankan pentingnya pembatasan akses dan validasi untuk menjaga integritas dan keamanan data.

Class diagram ini mencakup tabel-tabel utama yang saling terhubung untuk mendukung pengelolaan aset secara menyeluruh. Tabel **users** menyimpan data akun pengguna, sementara tabel **pegawai** menyimpan profil pegawai. **Data\_aset** menyimpan informasi detail aset dan berelasi dengan tabel seperti **jenis\_aset**, **depresiasi\_aset** (mengelola data penyusutan aset), **inspeksi\_aset** (hasil pemeriksaan kondisi aset), **activity\_log** (catatan aktivitas sistem), dan **laporan** (rekapitulasi informasi, termasuk keuangan dan penyusutan). Tabel **agenda\_aset** mencatat kegiatan pemeliharaan, **jurnal\_aset** mencatat kejadian terkait aset, sedangkan **keuangan\_aset** mencatat transaksi terkait aset. **Access** mengatur hak akses pengguna dalam sistem. Hubungan ini memastikan pengelolaan, pemantauan, dan pelaporan aset yang terstruktur.



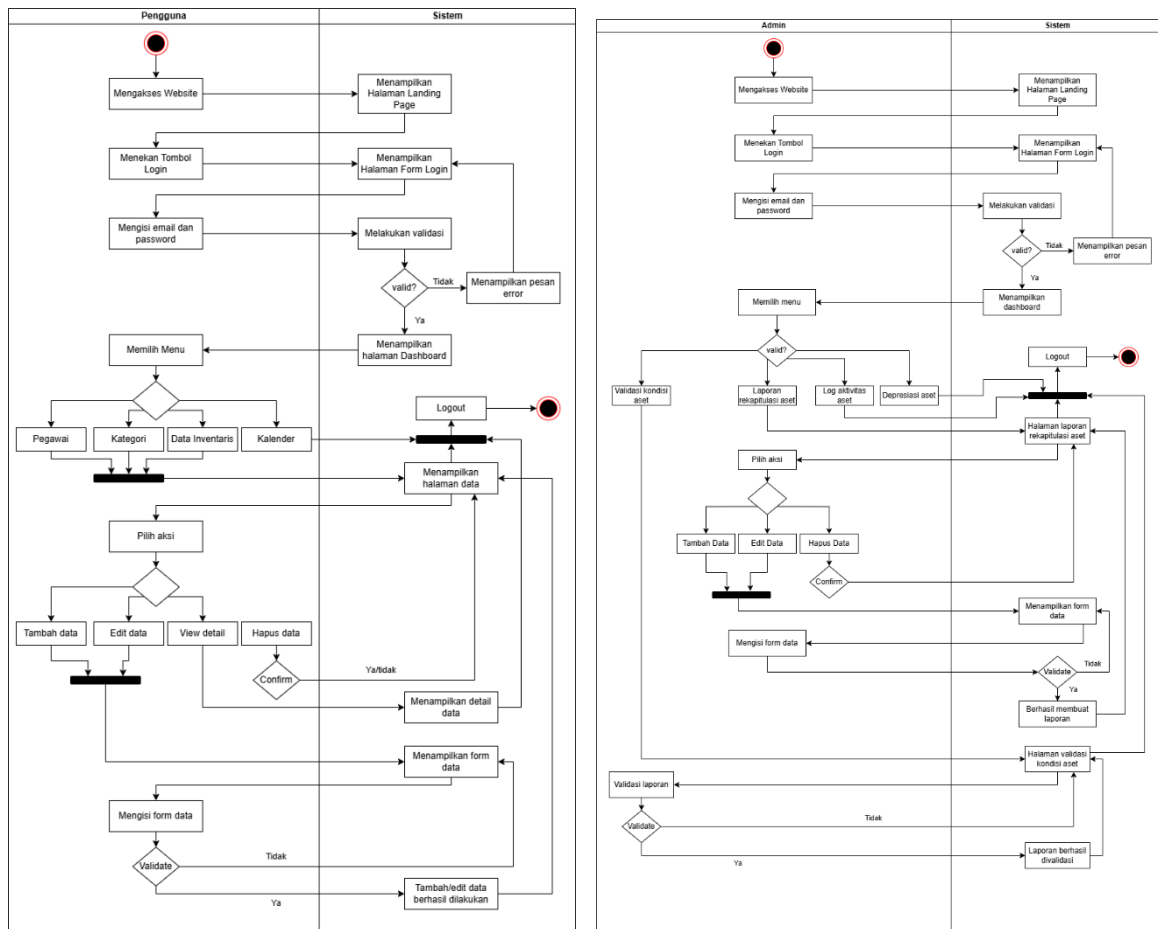
Gambar 4. Class Diagram Iterasi-1 dan Iterasi-2



Gambar 5. Salah Satu High-Fidelity UI Iterasi-1 dan Iterasi-2

Desain high-fidelity UI ini mencakup dua fitur utama dalam Iterasi-1 dan Iterasi-2. Pada Iterasi-1, UI dirancang untuk menampilkan detail lengkap aset dengan informasi terperinci seperti jenis, kategori, status, dan kondisi, sehingga pengguna dapat mengelola data aset dengan mudah dan efisien. Sementara pada Iterasi-2, desain UI memungkinkan pengguna untuk mencetak laporan kondisi aset yang terstruktur dan profesional, mencakup semua informasi penting untuk analisis atau pelaporan lebih lanjut. Kedua desain ini bertujuan untuk memberikan pengalaman pengguna yang optimal dan memudahkan pengelolaan serta pelaporan data aset.

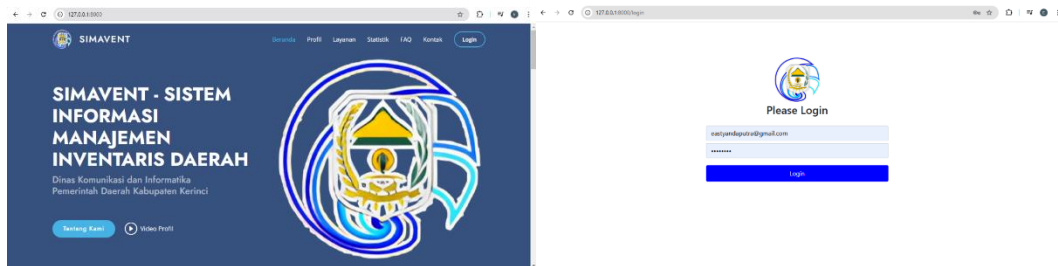
Diagram ini menggambarkan alur aktivitas dalam iterasi pertama, dimulai dari login, dilanjutkan dengan dashboard untuk mengelola data pegawai, kategori, dan barang melalui proses CRUD. Sistem juga menyediakan fitur kalender aset untuk mempermudah pengelolaan aset. Proses berakhir dengan *logout*. Di sisi lain, Pegawai melaporkan kondisi aset yang kemudian divalidasi oleh Admin. Admin juga membuat laporan rekapitulasi aset yang akan diverifikasi dan divalidasi oleh Kepala Bidang (Kabid) dan Kepala Dinas untuk memastikan keakuratan dan kelengkapan data sebelum disetujui. Diagram ini merangkum alur aktivitas dari *login* hingga *logout*, serta interaksi data dalam sistem.



Gambar 6. Activity Diagram Iterasi-1 dan Iterasi-2

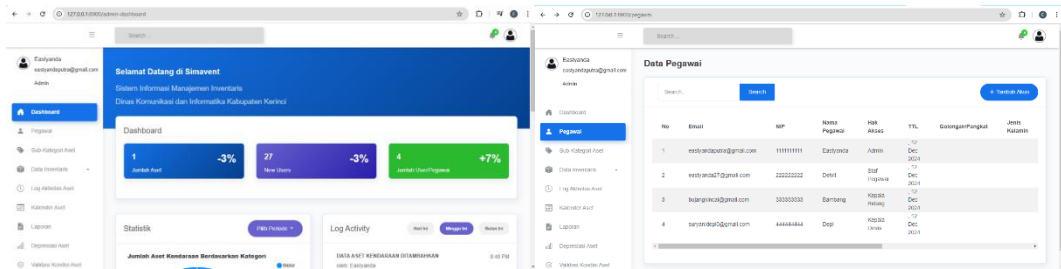
3. Coding

Selanjutnya sistem diimplementasikan ke dalam sebuah database dan kode. Penulis menggunakan *framework Laravel* dan *Bootstrap* untuk membantu pengembangan. Implementasi dimulai dari implementasi rancangan *database* yang sebelumnya dirancang di *class diagram* ke *phpMyAdmin*, lalu diimplementasikan rancangan kebutuhan sebelumnya ke kode menggunakan *Visual Studio Code*. Setelah dibuat database, lalu dilakukan proses *coding* dengan hasil tampilan *website* sebagai berikut:



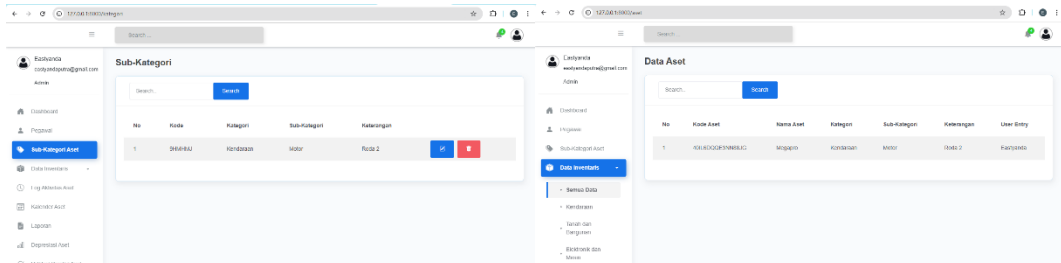
Gambar 7. Halaman Landing Page dan Login

Pada iterasi pertama, beberapa halaman utama telah berfungsi dengan baik, dimulai dari landing page yang memperkenalkan sistem dan menyediakan navigasi ke fitur lainnya. Halaman login memungkinkan pengguna memasukkan *email* dan *password*, yang kemudian divalidasi dengan data di *database*. Jika data valid, pengguna diarahkan ke *dashboard* sebagai pusat kontrol. *Dashboard* ini dirancang sesuai dengan peran dan hak akses masing-masing pengguna.



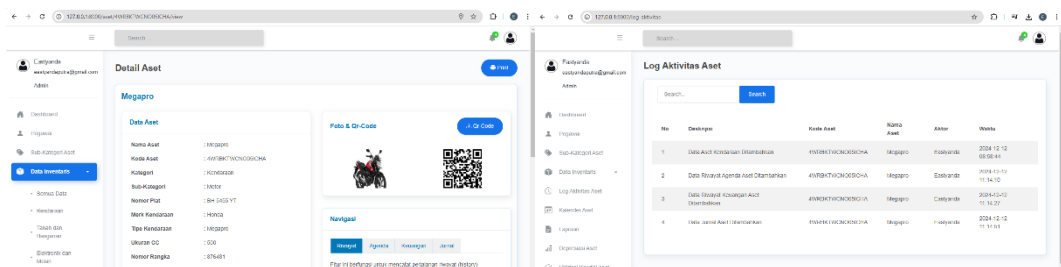
Gambar 8. Halaman Dashboard dan Pegawai

Halaman *dashboard* adalah halaman pertama yang muncul setelah pengguna berhasil *login*, menampilkan statistik dan rekapitulasi data dari database untuk memberikan gambaran kondisi inventaris secara *real-time*. Informasi ini membantu pengguna memantau status inventaris dan mendukung pengambilan keputusan yang lebih efektif. Sementara itu, halaman pegawai menampilkan data lengkap tentang pegawai yang terdaftar dalam sistem, seperti email, nama, jabatan, dan informasi lainnya, yang hanya dapat diakses oleh Admin dengan hak akses penuh. Halaman ini dirancang untuk mempermudah pengelolaan dan pembaruan data pegawai guna mendukung operasional sistem.



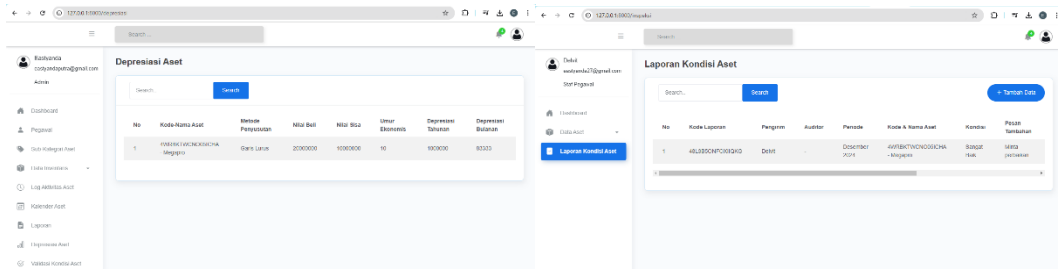
Gambar 9. Halaman Kategori dan Data Aset

Halaman kategori menyajikan data tentang kategori aset dalam sistem, seperti kode kategori, nama, sub-kategori, dan keterangan, yang hanya dapat diakses dan dikelola oleh Admin. Halaman ini mempermudah pengaturan dan pembaruan kategori aset, memastikan data selalu terstruktur dengan baik. Pada halaman-halaman terkait aset, Admin dapat melakukan proses CRUD (*create, read, update, delete*), sedangkan pengguna lain hanya dapat melihat data yang tersedia. Data yang berhasil diinput ditampilkan dalam tabel yang terorganisir untuk memudahkan pengelolaan dan navigasi.



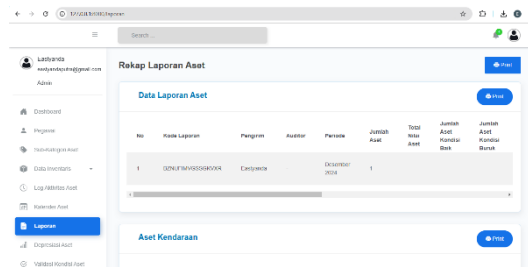
Gambar 10. Halaman Detail Aset dan Log Aktivitas Aset

Halaman ini menampilkan detail lengkap dari data aset, termasuk gambar dan QR-Code yang terhubung dengan aset tersebut. Semua pengguna, seperti Admin, Kepala Bidang, Kepala Dinas, dan Pegawai, dapat mengakses halaman ini. Namun, Admin memiliki hak akses tambahan untuk menambahkan informasi terkait aset, seperti riwayat, agenda, keuangan, dan jurnal. Pengguna dapat mengunduh dan mencetak data serta *QR-Code*, yang akan ditempelkan pada aset fisik untuk memudahkan akses detail melalui pemindaian. Selain itu, halaman log aktivitas aset mencatat setiap perubahan yang terjadi pada data aset, seperti penambahan, pembaruan, atau penghapusan, yang dapat dipantau oleh Admin untuk memastikan transparansi dan pelacakan perubahan dalam sistem.



Gambar 11. Halaman Depresiasi dan Kondisi Aset

Halaman depresiasi aset menampilkan informasi tentang penurunan nilai barang seiring bertambahnya umur, termasuk perhitungan depresiasi yang membantu Admin dalam memantau kondisi finansial aset dan merencanakan pengelolaan keuangan. Sementara itu, halaman laporan kondisi aset memungkinkan Pegawai untuk melaporkan kondisi aset yang menjadi tanggung jawabnya, yang kemudian divalidasi oleh Admin. Kedua halaman ini mendukung Admin dalam mengelola, memantau, dan memelihara aset dengan efektif.



Gambar 12. Halaman Rekapitulasi Aset

Halaman laporan rekapitulasi aset menampilkan rekapitulasi data aset pada waktu tertentu, disusun oleh Admin dan divalidasi oleh Kepala Bidang (Kabid) dan Kepala Dinas. Laporan ini memberikan gambaran keseluruhan kondisi aset secara periodik untuk evaluasi dan pengambilan keputusan pengelolaan aset.

4. *Testing*

- *User Acceptance Testing (UAT)*

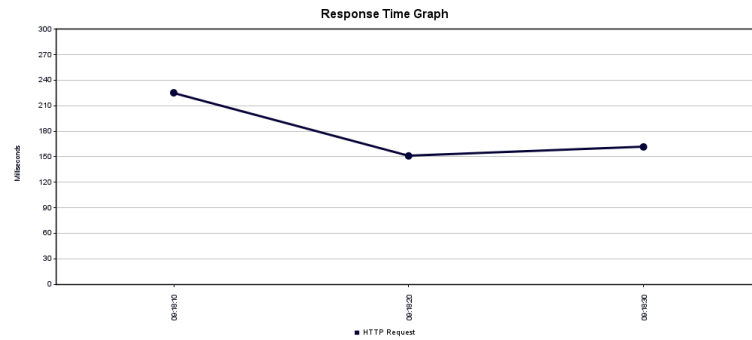
Pengujian UAT melibatkan 7 responden dari pegawai Diskominfo Kabupaten Kerinci. Pengujian dilakukan melalui kuesioner Google Form (<https://forms.gle/RGTHnmKhbmXPUQA78>) setelah pelatihan penggunaan sistem informasi. Responden diminta memberikan jawaban dengan skala 1-5 (sangat tidak setuju hingga sangat setuju). Hasil pengujian akan dianalisis berdasarkan jawaban yang diberikan. Berdasarkan hasil pengujian UAT tersebut, peneliti dapat menyimpulkan bahwa Sistem Informasi Manajemen Inventaris sangat layak untuk digunakan dalam pengelolaan barang inventaris, dengan nilai rata-rata akhir sebesar 4,1 atau persentase 82%.

- *Performance Testing*

Performance testing akan menggunakan JMeter, alat open-source untuk menguji performa, beban, dan stres pada aplikasi web, database, dan API. JMeter akan mensimulasikan berbagai skenario pengguna secara bersamaan untuk mengukur kinerja sistem di bawah beban tertentu.

a. *Normal Load Test*

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
HTTP Request	9977	165	0	1399	93.03	97.22%	316.4/sec	996.96	2.24	3226.9
TOTAL	9977	165	0	1399	93.03	97.22%	316.4/sec	996.96	2.24	3226.9



Gambar 13. Normal Load Test

Dari hasil *normal load test*, sistem masih mampu mengelola sampai maksimal 500 user yang mengakses secara serentak, dimana dibuktikan dengan ketika sistem dicoba akses dengan 1000 user, tingkat *error* meningkat dari 0% menjadi 97,22 %.

b. *Spike Test*

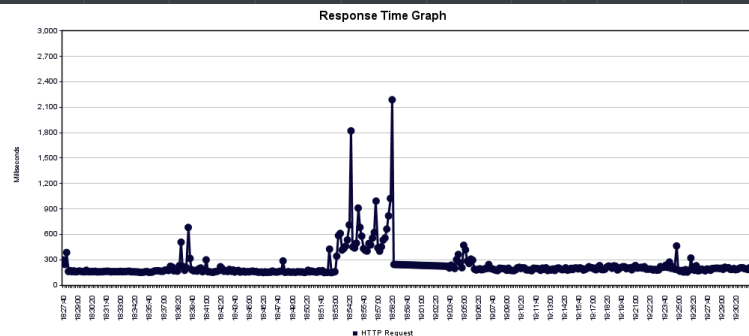
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
HTTP Request	550	353	140	1228	187.49	87.27%	7.5/sec	49.96	0.51	6853.2
TOTAL	550	353	140	1228	187.49	87.27%	7.5/sec	49.96	0.51	6853.2

Gambar 14. Spike Test

Dari hasil *spike test*, sistem masih tidak mampu untuk menampung lonjakan user dalam jumlah besar secara mendadak Dimana dibuktikan ketika skenario pertama diakses 50 user sistem berjalan lancar dengan 0%, namun ketika datang 500 user secara tiba-tiba, persentase *error* langsung meningkat ke 87,27%.

c. *Soak Test*

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughp...	Received ...	Sent KB/s...	Avg. Bytes
HTTP Req...	268709	2687	0	130284	13006.60	9.18%	36.7/sec	726.87	8.32	20275.9
TOTAL	268709	2687	0	130284	13006.60	9.18%	36.7/sec	726.87	8.32	20275.9



Gambar 15. Soak Test

Dari hasil soak test, sistem masih mampu tetap stabil meskipun diakses oleh user dalam waktu yang lama. Dimana dibuktikan ketika JMeter disimulasikan selama 2 jam nonstop, tingkat *error* yang terjadi hanya meningkat dari 0% menjadi 9,18%.

C. *Final Release*

Pada tahap final pengembangan, website telah berhasil disempurnakan sebelum diserahkan kepada Diskominfo Kabupaten Kerinci untuk digunakan dalam pengelolaan inventaris. Keputusan ini diambil dengan tujuan memastikan bahwa website telah disesuaikan secara optimal sesuai dengan persyaratan yang ditetapkan, sehingga mampu membantu dalam menyediakan *platform* dalam mengelola barang-barang inventaris. Final Release untuk saat ini masih menggunakan domain sendiri yaitu simavent.site yang menggunakan layanan *hosting* dari Hostinger sembari menunggu pemberian server dari Dinas Komunikasi dan Informatika Kabupaten Kerinci.

#### 4. PENUTUP

Berdasarkan hasil dan pembahasan, sistem informasi manajemen inventaris berbasis web pada Dinas Komunikasi dan Informatika Kabupaten Kerinci berhasil dirancang menggunakan metode *Extreme Programming* (XP), yang berfokus pada kebutuhan pengguna melalui pendekatan interaktif dengan komunikasi terbuka. Proses pengembangan dilakukan dalam dua iterasi, disertai pengujian *performance testing* dan *user acceptance testing*. Hasil pengujian menunjukkan bahwa sistem mendapatkan nilai 4,1 atau 82% pada *user acceptance testing*, yang mengindikasikan sistem dapat diterima oleh pengguna. Pada *performance testing*, sistem mampu menampung hingga 500 pengguna dalam waktu singkat pada normal load test, namun masih mengalami kendala pada spike test. Meskipun demikian, hasil *soak test* menunjukkan stabilitas sistem saat digunakan selama 2 jam tanpa henti. Dengan jumlah pengguna yang relatif sedikit, dapat disimpulkan sistem tetap dapat digunakan tanpa terganggu oleh keterbatasan performa.

#### DAFTAR PUSTAKA

- [1] Kundang K Juman, "Agile Development Methods," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2018, [Online]. Available: <https://sis.binus.ac.id/2017/05/08/agile-development-methods/>
- [2] N. A. Septiani and F. Y. Habibie, "Penggunaan Metode Extreme Programming Pada Perancangan Sistem Informasi Pelayanan Publik," *J. Sist. Komput. dan Inform.*, vol. 3, no. 3, p. 341, 2022, doi: 10.30865/json.v3i3.3931.
- [3] H. Handayani, K. U. Faizah, A. M. Ayulya, M. F. Rozan, D. Wulan, and M. L. Hamzah, "Perancangan Sistem Informasi Inventory Barang Berbasis Web Menggunakan Metode Agile Software Development," *J. Test. dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 29–40, 2023.
- [4] T. Ardiansah, "Perancangan Sistem Persediaan Menggunakan Metode Extreme Programming," *J. Ilm. Inform. dan Ilmu Komput.*, vol. 1, no. 1, pp. 1–6, 2022, doi: 10.58602/jima-ilkom.v1i1.1.
- [5] F. Wicaksono and O. S. Baswara, "Design and Implementation of Web-Based Helpdesk Information Systems Using Extreme Programming Methods," *ITEJ (Information Technol. Eng. Journals)*, vol. 5, no. 2, pp. 88–96, 2020, doi: 10.24235/itej.v5i2.44.
- [6] A. Fauzi, N. Indriyani, and A. B. Hasta Yanto, "Implementasi Sistem Informasi Inventory Berbasis Web (Studi Kasus: Cv. Sinar Abadi Cemerlang)," *J. Teknol. Dan Open Source*, vol. 3, no. 2, pp. 144–157, 2020, doi: 10.36378/jtos.v3i2.781.
- [7] I. G. N. Suryantara and J. F. Andry, "Development of Medical Record With Extreme Programming SDLC," *Int. J. New Media Technol.*, vol. 5, no. 1, pp. 47–53, 2018, doi: 10.31937/ijnmt.v5i1.706.
- [8] E. B. Pratama and A. Hendini, "Implementasi Extreme Programming Pada Perancangan SIMRS (Sistem Informasi Manajemen Rumah Sakit)," *J. Khatulistiwa Inform.*, vol. 10, no. 2, pp. 107–112, 2022, doi: 10.31294/jki.v10i2.14159.
- [9] S. Romney, "Penerapan Metode Extreme Programming Dalam Perancangan Aplikasi Perhitungan Kuota Sks Mengajar Dosen," *IKRA-ITH Inform.*, vol. 3, no. 1, pp. 106–113, 2015.
- [10] A. Hendini, "PEMODELAN UML SISTEM INFORMASI MONITORING PENJUALAN DAN STOK BARANG (STUDI KASUS: DISTRO ZHEZHA PONTIANAK)," *Commun. ACM*, vol. 25, no. 1, pp. 27–47, 2016, doi: 10.1145/358315.358387.
- [11] M. Syarif and E. B. Pratama, "Analisis Metode Pengujian Perangkat Lunak Blackbox Testing Dan Pemodelan Diagram Uml Pada Aplikasi Veterinary Services Yang Dikembangkan Dengan Model Waterfall," *J. Tek. Inform. Kaputama*, vol. 5, no. 2, pp. 253–258, 2021.
- [12] S. Alfarizi, A. R. Mulyawan, and H. Basri, "Rancang Bangun Sistem Informasi Penjualan Berbasis Web Dengan Pemanfaatan Uml (Unified Modelling Language) Pada Cv Harum Catering Karawang," *Ijns.org Indones. J. Netw. Secur.*, vol. 7, no. 4, pp. 2302–2700, 2018.
- [13] T. Abdulghani and B. P. Sati, "Pengenalan Rumah Adat Indonesia Menggunakan Teknologi Augmented Reality Dengan Metode Marker Based Tracking Sebagai Media Pembelajaran," *Media J. Inform.*, vol. 11, no. 1, p. 43, 2020, doi: 10.35194/mji.v11i1.770.
- [14] A. Gustiawan and C. Trisianto, "Perancangan Sistem Informasi Penggajian Karyawan Berbasis Web Menggunakan Metode Extreme Programming Pada Pt. Pradana Energi Gemilang," *J. Ilmu Komput. JIK*, vol. V, no. 01, p. 2022, 2022, [Online]. Available: <https://jurnal.pranataindonesia.ac.id/index.php/jik/article/view/110/63>
- [15] D. Andriansyah and L. Nulhakim, "Extreme Programming Dalam Perancangan Sistem Informasi Jasa Fotografi," *ICIT J.*, vol. 7, no. 1, pp. 10–19, 2021, doi: 10.33050/icit.v7i1.1442.
- [16] P. Pujianto, M. Mujito, D. Prabowo, and B. H. Prasetyo, "Pemilihan Warga Penerima Bantuan Program Keluarga Harapan (PKH) Menggunakan Metode Simple Additive Weighting (SAW) dan User Acceptance Testing (UAT)," *J. Inform. Univ. Pamulang*, vol. 5, no. 3, p. 379, 2020, doi: 10.32493/informatika.v5i3.6671.
- [17] B. Wisnuadhi, A. Maspupah, S. R. Wulan, M. R. Sholahuddin, S. D. Setiarini, and L. H. Firdaus, "Studi Komparatif Fungsionalitas, Performance dan User experience pada Website Geopark," *J. Inf. Syst. Res.*, vol. 3, no. 4, pp. 579–589, 2022, doi: 10.47065/josh.v3i4.1879.
- [18] A. W. Giovanni Harrius, Jong Jek Siang, "Implementasi Continuous Integration dan Otomatisasi Monitoring Job Server Dengan Jenkins pada Compute Engine," vol. 6, no. 4, pp. 704–714, 2024.