



# JURNAL MAHASISWA SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

## SISTEM INFORMASI LOGISTIK KARGO MENGGUNAKAN METODE AGILE (EXTREME PROGRAMMING) DI PT. XYZ

Hendra Hendriana<sup>1\*</sup>, Tuti Rohayati<sup>2</sup>, Dadan Mulyana<sup>3</sup>

<sup>1,2,3</sup>Program Studi Sistem Informasi, Fakultas Teknik, Universitas Galuh  
Email: <sup>1</sup>hendra\_hendriana@student.unigal.ac.id, <sup>2</sup>tutirohayati@unigal.ac.id,  
<sup>3</sup>dadan@unigal.ac.id

### ABSTRACT

*The development of the logistics industry requires an integrated information system to facilitate the management of goods delivery. PT. XYZ, as a cargo logistics service company, needs a digital solution to minimize recording errors, accelerate tracking, and improve customer service. This study aims to develop a cargo logistics information system using the Extreme Programming (XP) methodology, which emphasizes flexibility and rapid development through the stages of planning, design, coding, and testing. The system was developed using the PHP programming language with the Laravel framework and a MySQL database, and was tested using Blackbox Testing to ensure its functionality. The implementation results show that the system is capable of managing waybills, tariffs, manifests, and reporting in an integrated manner, accelerating the input and tracking processes, and reducing potential data errors, thereby supporting the company's operational needs.*

*Keywords: Information Systems, Cargo Logistics, Extreme Programming, Laravel.*

### ABSTRAK

Perkembangan industri logistik memerlukan sistem informasi terintegrasi untuk mempermudah pengelolaan pengiriman barang. PT. XYZ sebagai perusahaan jasa logistik kargo membutuhkan solusi digital guna meminimalkan kesalahan pencatatan, mempercepat pelacakan, serta meningkatkan pelayanan pelanggan. Penelitian ini bertujuan membangun sistem informasi logistik kargo dengan metode *Extreme Programming* (XP) yang menekankan fleksibilitas dan kecepatan pengembangan melalui tahap *planning, design, coding, dan testing*. Sistem dibangun dengan bahasa pemrograman PHP, *framework* Laravel dan basis data MySQL, serta diuji menggunakan *Blackbox Testing* untuk memastikan fungsionalitas berjalan dengan baik. Hasil implementasi menunjukkan bahwa sistem mampu mengelola data resi, tarif, *manifest*, dan pelaporan secara terintegrasi, mempercepat proses input dan pelacakan pengiriman, serta mengurangi potensi kesalahan data, sehingga mendukung kebutuhan operasional perusahaan.

Kata Kunci: Sistem Informasi, Logistik Kargo, *Extreme Programming*, Laravel.

### PENDAHULUAN

Penerapan sistem informasi telah memberikan dampak yang besar terhadap berbagai proses bisnis, termasuk dalam sektor logistik (Ardana, 2024). Perusahaan logistik harus menunjukkan kompetensi dalam pengelolaan barang yang cepat, tepat, dan terkoordinasi untuk memenuhi ekspektasi konsumen yang terus berkembang. Penelitian (Sahara & Romadona, 2024) menegaskan bahwa penerapan sistem informasi logistik yang terintegrasi

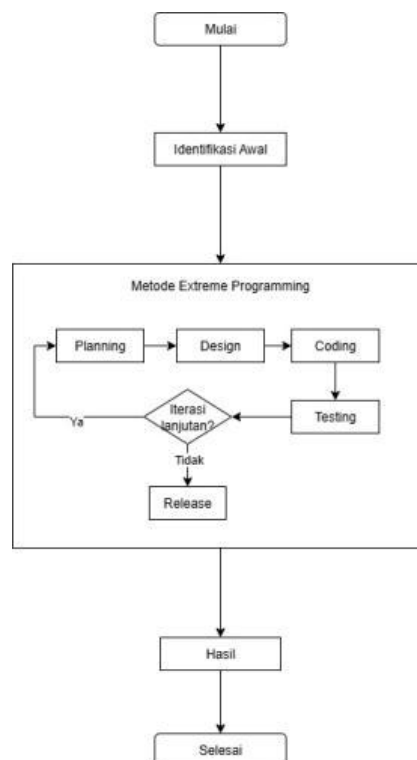
mampu meningkatkan efisiensi serta responsivitas terhadap kebutuhan pasar.

PT. XYZ sebagai perusahaan logistik kargo menghadapi tantangan operasional seiring meningkatnya permintaan pengiriman barang melalui jalur darat, laut, dan udara. Volume pengiriman yang semakin besar berpotensi menghambat distribusi, memperlambat layanan, dan menurunkan kepuasan pelanggan. Upaya perusahaan dalam mengadopsi sistem informasi komersial masih menemui kendala karena keterbatasan dalam menyesuaikan kebutuhan spesifik, khususnya terkait manajemen tarif pengiriman yang fleksibel, pengelolaan *manifest*, dan pelaporan. Kondisi ini menunjukkan adanya kesenjangan antara kebutuhan operasional perusahaan dengan kemampuan sistem informasi yang tersedia.

Berdasarkan permasalahan tersebut, penelitian ini bermaksud untuk membangun sistem informasi logistik kargo yang terintegrasi menggunakan metode *Extreme Programming* (XP). Pendekatan XP dipilih karena kemampuannya dalam mendukung pengembangan sistem secara iteratif, adaptif terhadap perubahan kebutuhan, serta melibatkan kolaborasi erat dengan pengguna. Sistem yang dibangun diharapkan mampu mengelola data resi, tarif, *manifest*, dan pelaporan secara terintegrasi, sehingga dapat mempercepat proses pengelolaan pengiriman barang dan mengurangi potensi kesalahan data.

### METODE

Penelitian ini menggunakan metode *Extreme Programming* (XP) sebagai model pengembangan sistem karena metode ini bersifat iteratif dan sesuai dalam pengembangan sistem dengan kebutuhan yang dapat berubah sewaktu-waktu (Purnama, 2023). Adapun Penelitian dilaksanakan di PT. XYZ yang bergerak di bidang jasa logistik kargo, dengan waktu penelitian berlangsung mulai dari Mei hingga Juli 2025.



Gambar 1. Bagan Alir Penelitian



# JURNAL MAHASISWA

## SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

---

Adapun penjelasan dari bagan alir tersebut, sebagai berikut:

1. Identifikasi Awal

Tahap awal penelitian dimulai dengan melakukan wawancara dengan pihak perusahaan. Tujuannya adalah untuk mengetahui bagaimana bisnis beroperasi, masalah apa yang dihadapi dan apa yang dibutuhkan pengguna. Kemudian mencari referensi yang relevan melalui studi literatur. Tahap ini memberikan informasi yang akan digunakan untuk merancang sistem yang akan dibangun.

2. Pengembangan Sistem dengan metode *Extreme Programming*.

Setelah kebutuhan dan masalah teridentifikasi, pengembangan sistem dilakukan dengan pendekatan *Agile*, khususnya metode *Extreme Programming* (XP). Metode ini bersifat iteratif dan adaptif terhadap perubahan kebutuhan pengguna. Berikut tahapan dalam XP:

a. *Planning* (Perencanaan)

Pada tahap ini, dilakukan perencanaan berdasarkan hasil identifikasi sebelumnya. Fitur-fitur sistem ditentukan dalam bentuk *user stories* yang menggambarkan kebutuhan pengguna. Prioritas pengembangan juga ditentukan di tahap ini.

b. *Design* (Perancangan)

Tahap ini merupakan proses perancangan awal sistem, yaitu membuat struktur sistem menggunakan UML, mulai dari *Use Case*, *Activity*, *Class*, dan *Sequence Diagram*.

UML merupakan alat bantu pemodelan visual pada pengembangan sistem berorientasi objek. Bahasa ini membantu pengembang membuat rancangan standar yang mudah dipahami serta mendukung komunikasi desain antar tim (Munawar, 2021)

c. *Coding* (Pengkodean)

Setelah desain selesai, proses pengembangan aplikasi dilakukan menggunakan pendekatan pemrograman berorientasi objek dengan framework Laravel berbasis PHP. XAMPP sebagai *webserver*, MySQL sebagai *database*, Visual Studio Code sebagai IDE (*Integrated Development Environment*) dalam proses pengkodean.

d. *Testing* (Pengujian)

Pengujian dilakukan untuk memastikan bahwa sistem bekerja sebagaimana mestinya sesuai dengan kebutuhan. Testing dilakukan menggunakan *Blackbox Testing*. Tujuannya testing adalah untuk mendeteksi kesalahan pada sistem dan memastikan sistem stabil.

3. Hasil

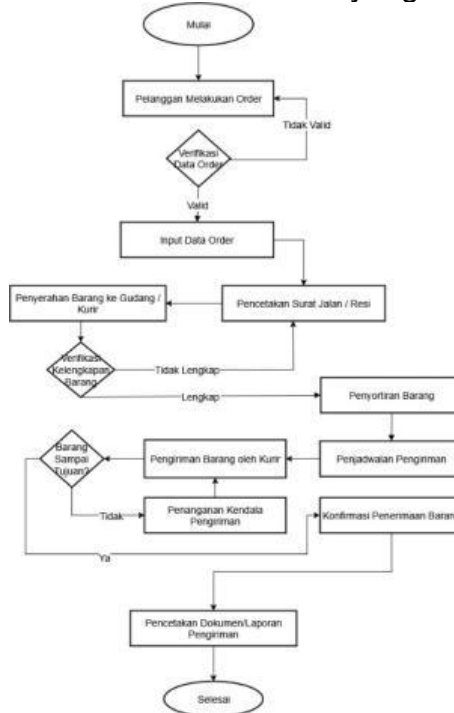
Pada tahap akhir, sistem informasi logistik kargo yang telah dibangun menghasilkan fitur-fitur yang sesuai dengan kebutuhan pengguna.

## HASIL DAN PEMBAHASAN

### Planning

#### 1. Analisis Sistem Berjalan

Gambar 2 memperlihatkan hasil analisis sistem yang berjalan.



**Gambar 2. Analisis Sistem Berjalan**

#### 2. User Story

*User story* diperoleh melalui proses wawancara dengan pengguna (pihak terkait). Proses wawancara dilakukan untuk menggali kebutuhan, harapan, serta permasalahan yang dihadapi pengguna terhadap sistem yang akan dibangun.

##### a. Iterasi 1

Pengembangan awal difokuskan pada fitur pengelolaan data pengiriman dan data tarif.

**Tabel 1. User Story**

| Aktor              | User Story                                                                                                                                                                 | Value |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| Superadmin & Admin | Sebagai <i>Superadmin/Admin</i> , saya ingin mengelola data pengiriman, sehingga tidak lagi memakai sistem lama.                                                           | 10    |
|                    | Sebagai <i>Superadmin/Admin</i> , saya ingin mengelola data tarif, termasuk rute, estimasi dan servis yang digunakan sehingga memudahkan pemilihan tarif dalam pengiriman. | 10    |

##### b. Iterasi 2

Pada iterasi kedua, ditambahkan fitur *manifest* sehingga *superadmin* atau *admin* dapat mengelola *manifest* pengiriman.

**Tabel 2. User Story Iterasi 2**

| Aktor              | User Story                                                                                                                                                                                               | Value |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| Superadmin & Admin | Sebagai <i>Superadmin/Admin</i> , saya ingin mengelola <i>manifest</i> pengiriman agar memudahkan penggabungan beberapa resi ke dalam satu data <i>barcode</i> yang sama, sehingga ketika <i>QR code</i> | 10    |

tersebut dipindai, status pengiriman *terupdate*  
otomatis menjadi *MANIFESTED*.

c. Iterasi 3

Pada iterasi ketiga, fokus pada kebutuhan *Customer* dengan menyediakan fitur *request pickup* agar dapat langsung mengajukan penjemputan paket.

**Tabel 3. User Story Iterasi 3**

| Aktor    | User Story                                                                                                                                                | Value |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| Customer | Sebagai <i>Customer</i> , saya ingin melakukan permintaan ( <i>request</i> ) <i>pickup</i> , sehingga saya dapat mengajukan permintaan penjemputan paket. | 9     |

d. Iterasi 4

Sistem dilengkapi dengan fitur laporan pengiriman untuk mendukung analisis performa, identifikasi tren, dan pengambilan keputusan operasional.

**Tabel 4. User Story Iterasi 4**

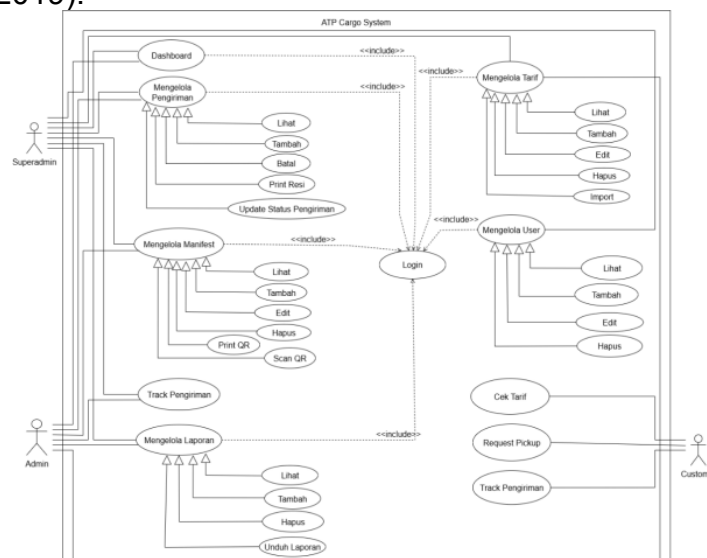
| Aktor              | User Story                                                                                                                                                                                        | Value |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------|
| Superadmin & Admin | Sebagai <i>Superadmin/Admin</i> , saya ingin melihat laporan pengiriman sehingga memudahkan dalam analisis performa, identifikasi tren, dan pengambilan keputusan terkait operasional pengiriman. | 8     |

## Design

### 1. Use Case Diagram

Sistem ini melibatkan tiga aktor, yaitu *Superadmin*, *Admin* dan *Customer*. Fungsi dari masing-masing aktor dapat dilihat pada Gambar 3.

*Use Case Diagram* adalah gambaran kelakuan (*behavior*) sistem informasi yang akan dibuat. Diagram ini menampilkan hubungan interaksi antara satu atau lebih aktor dengan sistem, serta mempermudah dalam mengidentifikasi fungsi-fungsi yang terdapat di dalam sistem dan siapa saja yang berhak menggunakannya (Rosa & Shalahuddin, 2019).



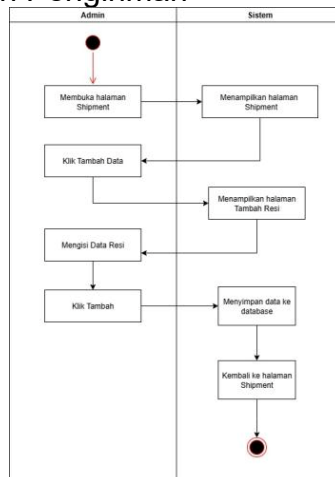
**Gambar 3. Use Case Diagram Keseluruhan**

**Tabel 5. Deskripsi Aktor**

| Aktor      | Deskripsi Aktor                                                                                                                                                                                               |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Superadmin | Superadmin dapat melakukan <i>login</i> ke sistem. Superadmin dapat mengelola pengiriman, mengelola tarif pengiriman, mengelola <i>manifest</i> pengiriman dan mengelola laporan, dan mengelola <i>user</i> . |
| Admin      | Admin dapat melakukan <i>login</i> ke sistem, Admin dapat mengelola pengiriman, mengelola tarif pengiriman, mengelola <i>manifest</i> pengiriman dan mengelola laporan.                                       |
| Customer   | Customer dapat melakukan permintaan ( <i>request</i> ) <i>pickup</i> , cek tarif pengiriman, dan melacak pengiriman ( <i>tracking</i> ).                                                                      |

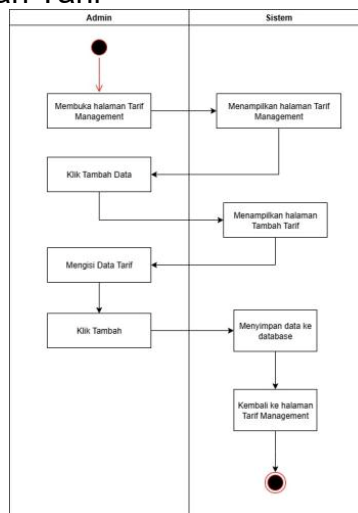
## 2. Activity Diagram

### a. Activity Diagram Tambah Pengiriman



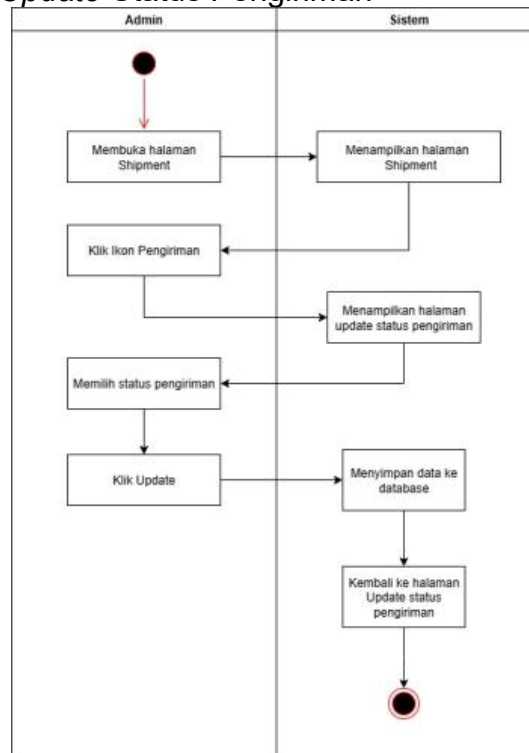
**Gambar 4. Activity Diagram Tambah Pengiriman**

### b. Activity Diagram Tambah Tarif



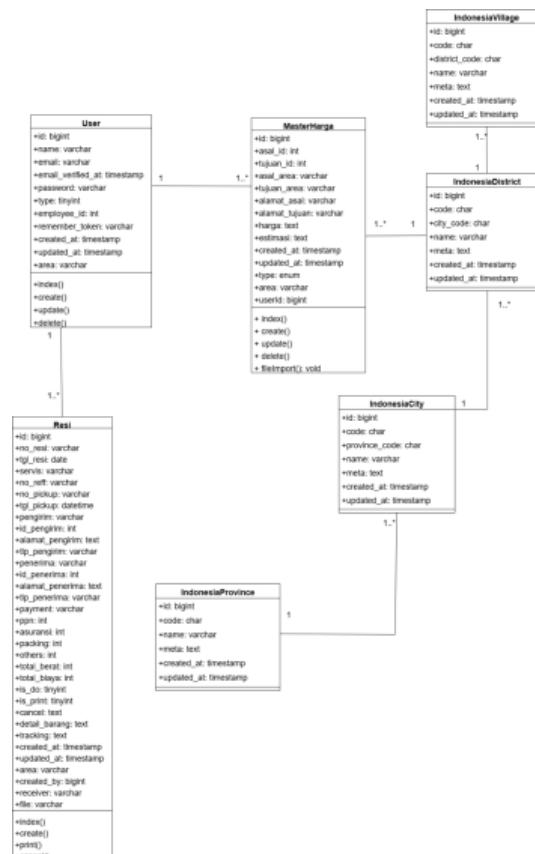
**Gambar 5. Activity Diagram Tambah Tarif**

## c. Activity Diagram Update Status Pengiriman



Gambar 6. Activity Diagram Update Status Pengiriman

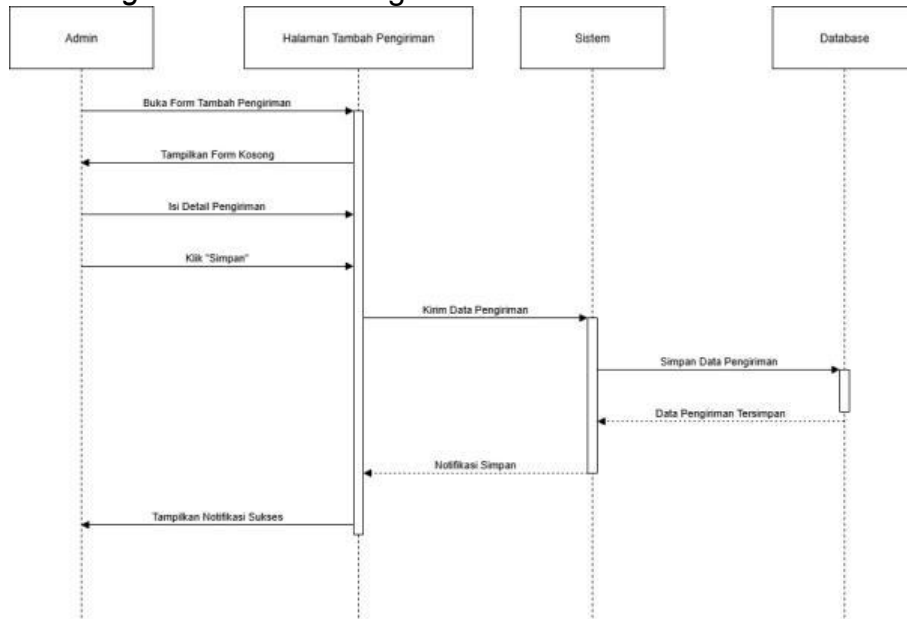
## 3. Class Diagram



Gambar 7. Class Diagram Keseluruhan

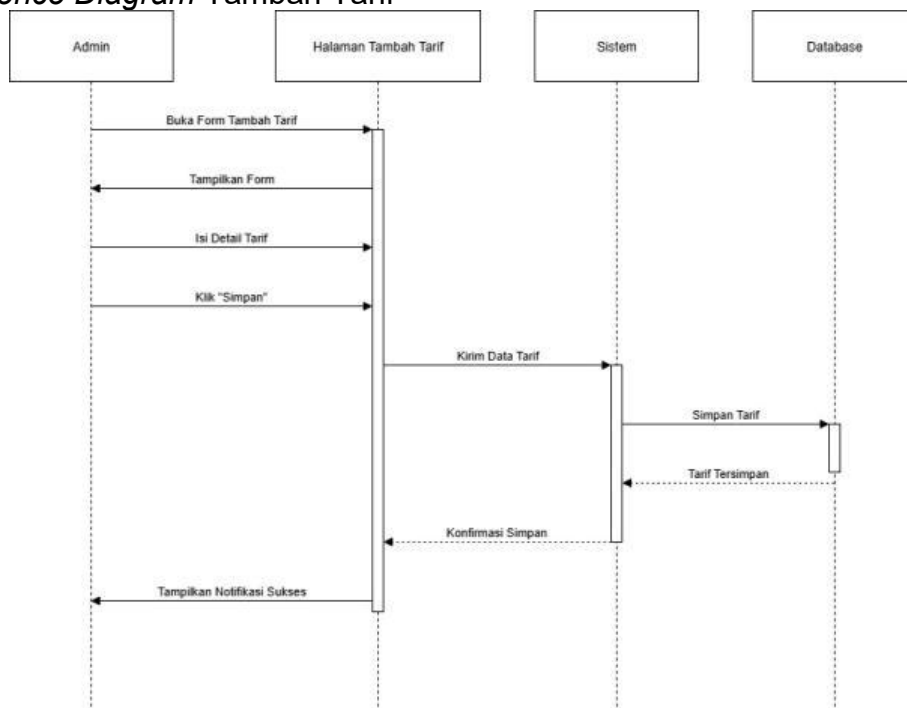
## 4. Sequence Diagram

### a. Sequence Diagram Tambah Pengiriman



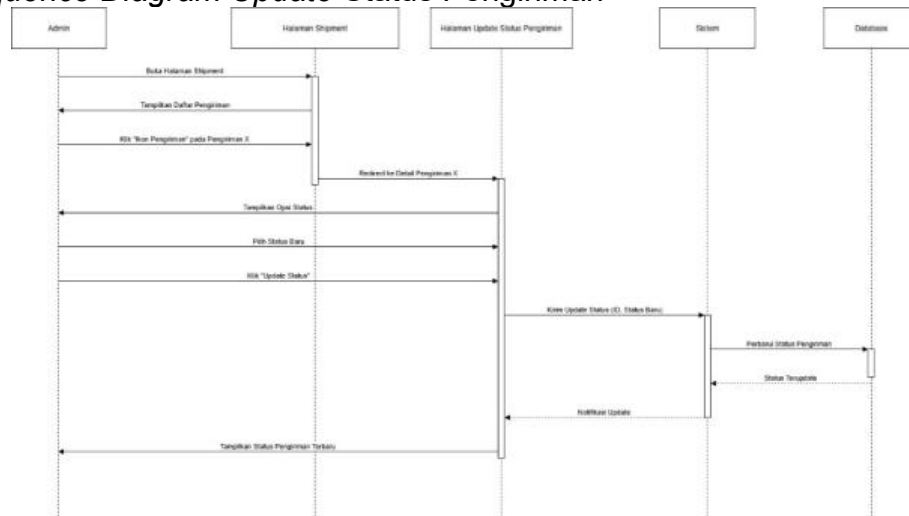
**Gambar 8. Sequence Diagram Tambah Pengiriman**

### b. Sequence Diagram Tambah Tarif



**Gambar 9. Sequence Diagram Tambah Tarif**

## c. Sequence Diagram Update Status Pengiriman

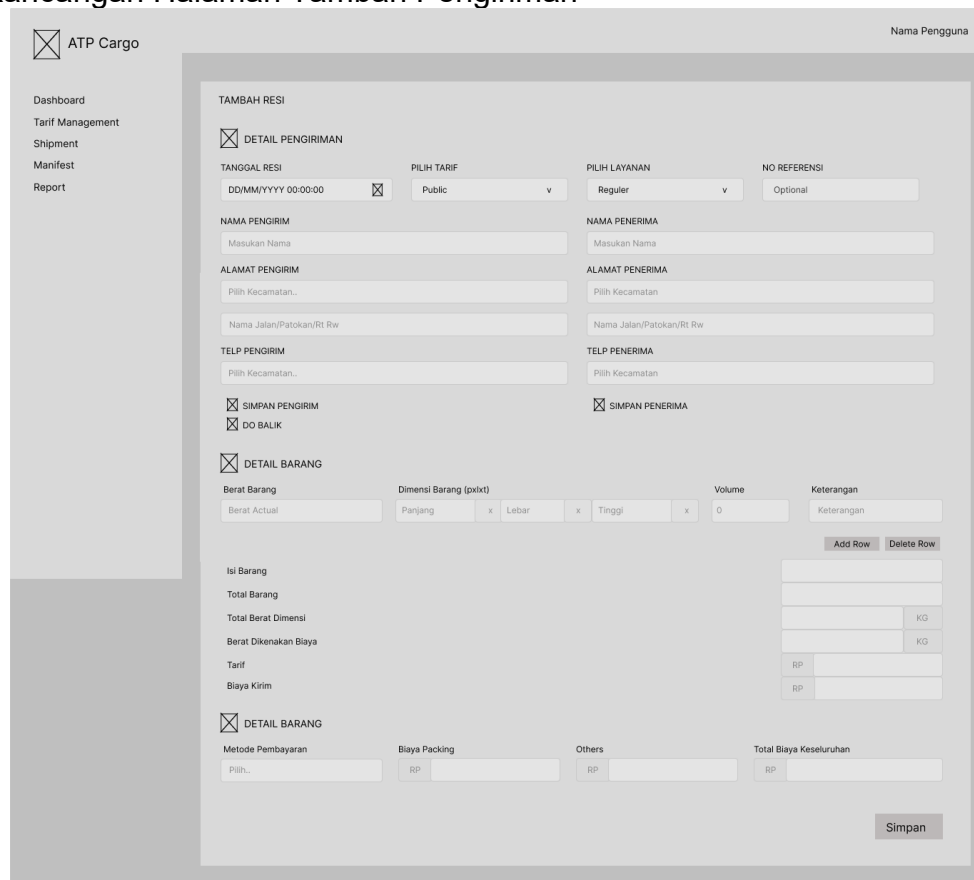


Gambar 10. Sequence Diagram *Update Status Pengiriman*

## 5. Rancangan Halaman Antarmuka

Pada tahap ini membuat rancangan halaman antarmuka sebelum tahap *coding* dilakukan.

### a. Rancangan Halaman Tambah Pengiriman



ATP Cargo Nama Pengguna

Dashboard  
Tariff Management  
Shipment  
Manifest  
Report

### TAMBAH RESI

☒ DETAIL PENGIRIMAN

TANGGAL RESI: DD/MM/YYYY 00:00:00 ☒ PILIH TARIF: Public v PILIH LAYANAN: Regular v NO REFERENSI: Optional

NAMA PENGIRIM: Masukkan Nama ALAMAT PENGIRIM: Pilih Kecamatan... Nama Jalan/Patokan/Rt Rw TELP PENGIRIM: Pilih Kecamatan...

☒ SIMPAN PENGIRIM ☒ DO BALIK

☒ DETAIL BARANG

Berat Barang: Berat Actual Dimensi Barang (pxxt): Panjang x Lebar x Tinggi x Volume: 0 Keterangan:

Isi Barang: Total Barang: Total Berat Dimensi: Berat Dikenakan Biaya: Tarif: Biaya Kirim:

☒ DETAIL BARANG

Metode Pembayaran: Pilih... Biaya Packing: RP Others: RP Total Biaya Keseluruhan: RP

Simpan

Gambar 11. Rancangan Halaman Tambah Pengiriman



# JURNAL MAHASISWA SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

## b. Rancangan Halaman Tambah Tarif

ATP Cargo Nama Pengguna

Dashboard  
Tarif Management  
Shipment  
Manifest  
Report

TAMBAH TARIF

☒ RUTE

ASAL  
DKI JAKARTA, KOTA JAKARTA TIMUR, CAKUNG

KODE AREA ASAL

TUJUAN  
JAWA BARAT, BANDUNG, CIMENYAN

KODE AREA TUJUAN

☒ ESTIMASI, SERVIS, TARIF

| KODE SERVIS | NAMA SERVIS | LEADTIME                  | TARIF                   |
|-------------|-------------|---------------------------|-------------------------|
| REG         | REGULER     | <input type="text"/> Hari | Rp <input type="text"/> |
| KIL         | KILAT       | <input type="text"/> Hari | Rp <input type="text"/> |
| EKO         | EKONOMI     | <input type="text"/> Hari | Rp <input type="text"/> |

Kembali Simpan

Gambar 12. Rancangan Halaman Tambah Tarif

## c. Rancangan Halaman *Update* Status Pengiriman

ATP Cargo Nama Pengguna

Dashboard  
Tarif Management  
Shipment  
Manifest  
Report

UPDATE LACAK PENGIRIMAN

ATPXXXX00000 Cari

Tracking Timeline

15 Juni 2025 10:10 AM ☒ PICK UP  
Mas Agung akan menjemput paketmu, pastikan nomor telepon dapat dihubungi dan paketmu sudah siap.

Update Status

Pilih...

Update

Gambar 13. Rancangan Halaman *Update* Status Pengiriman



# JURNAL MAHASISWA

## SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

### Coding

Selanjutnya, dilakukan tahap pengkodean (*coding*) berdasarkan desain atau rancangan sistem yang telah dirancang pada tahap sebelumnya. Tahap coding ini merupakan realisasi dari semua perancangan, baik perancangan UML, maupun rancangan antarmuka pengguna.

#### 1. Halaman Tambah Pengiriman

Gambar 14. Halaman Tambah Pengiriman

#### 2. Halaman Tambah Tarif

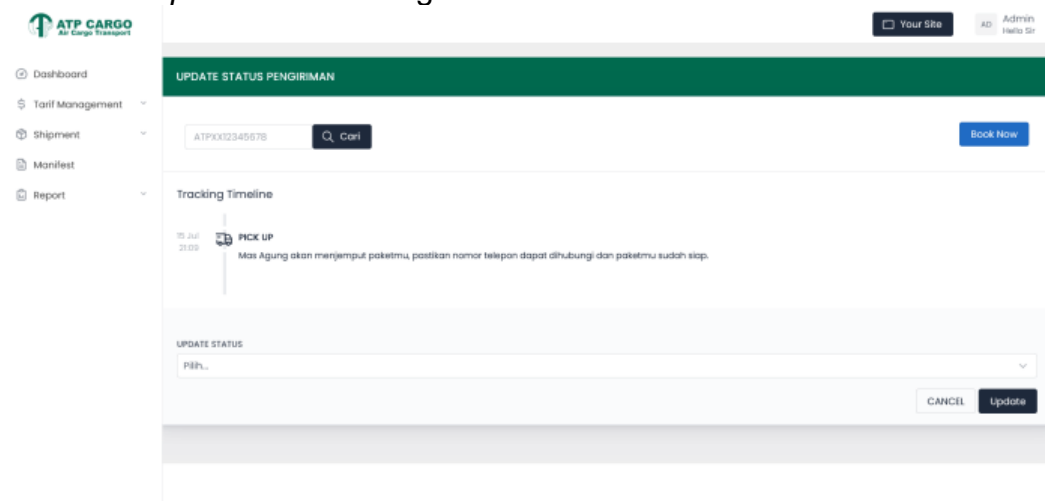
Gambar 15. Halaman Tambah Tarif



# JURNAL MAHASISWA SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

## 3. Halaman *Update Status Pengiriman*



**Gambar 16. Halaman *Update Status Pengiriman***

### Testing

Pengujian (*testing*) merupakan fase yang bertujuan untuk memastikan kualitas dan kinerja suatu sistem sebelum dirilis ke pengguna akhir. Pengujian akan dilakukan menggunakan *Blackbox Testing* guna menguji fungsionalitas sistem.

*Blackbox Testing* digunakan untuk mengidentifikasi kesalahan fungsional dari sudut pandang pengguna. Tujuan dari *testing* (pengujian) ini yaitu untuk mengurangi terjadinya *error* (kesalahan) serta memastikan bahwa sistem yang dikembangkan telah berfungsi sesuai dengan yang diharapkan (Arima et al., 2024).

**Tabel 6. *Blackbox Testing***

| Skenario Pengujian           | Test Case                                                                                                                                    | Hasil yang diharapkan                                                                                              | Hasil                                      | Status |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|--------------------------------------------|--------|
| Login Berhasil               | Memasukkan <i>username</i> dan <i>password</i> yang sesuai kemudian menekan tombol " <i>Login</i> ".                                         | Sistem menerima input dan masuk ke <i>database</i> .                                                               | Sistem menerima input.                     | Valid  |
| Login Gagal                  | Memasukkan <i>username</i> dan <i>password</i> yang tidak sesuai kemudian menekan tombol " <i>Login</i> ".                                   | Sistem tidak menerima input dan menampilkan pesan <i>error</i> .                                                   | Sistem menampilkan pesan <i>error</i> .    | Valid  |
| Menambah Pengiriman          | Pada menu <i>Shipment</i> klik " <i>Tambah Data</i> ", kemudian mengisi data pengiriman yang sesuai lalu klik " <i>Simpan</i> ".             | Sistem menampilkan form tambah pengiriman dan sistem menerima input penambahan data pengiriman.                    | Sistem berhasil menyimpan data pengiriman. | Valid  |
| Menambah Tarif               | Pada menu <i>Tarif Management</i> klik " <i>Tambah Data</i> ", kemudian mengisi data tarif yang sesuai lalu klik " <i>Simpan</i> ".          | Sistem menampilkan form penambahan tarif baru dan sistem menerima input penambahan data tarif.                     | Sistem berhasil menyimpan data tarif.      | Valid  |
| Mengupdate status pengiriman | Pada menu <i>Shipment</i> klik " <i>Ikun Pengiriman</i> " pada entri data pengiriman, memilih status pengiriman lalu klik " <i>Update</i> ". | Sistem menampilkan halaman <i>Update status pengiriman</i> dan sistem berhasil menerima <i>update pengiriman</i> . | Sistem berhasil memperbarui pengiriman.    | Valid  |



# JURNAL MAHASISWA SISTEM INFORMASI GALUH (JMSIG)

Volume 2. Nomor 1, Oktober 2025  
ISSN : 3089-3577

## SIMPULAN

Dari penelitian ini diperoleh kesimpulan bahwa sistem informasi logistik kargo berhasil dibangun sesuai dengan kebutuhan, dimana alur bisnis utama mulai dari pengelolaan pengiriman, pengelolaan tarif, manajemen *manifest* hingga pelaporan telah terintegrasi dalam satu sistem. Implementasi fitur-fitur tersebut mampu meminimalkan proses manual dan mempercepat akses informasi. Penerapan metode *Extreme Programming* terbukti membantu dalam proses pengembangan karena sifatnya yang iteratif memungkinkan penyesuaian fitur secara cepat berdasarkan masukan pengguna.

Sejalan dengan hasil penelitian ini, disarankan adanya pengembangan lebih lanjut pada modul laporan, seperti penambahan laporan kinerja kurir, penambahan lapisan keamanan sistem seperti autentikasi dua faktor untuk meningkatkan perlindungan data, serta pengembangan aplikasi mobile.

## DAFTAR PUSTAKA

- Ardana, A. R. (2024). Pengembangan Sistem Informasi Manajemen Logistik Berbasis Code Igniter. *Journal of Informatics and Computer Science (JOINCOS)*, 1(2), 1–7. <https://www.jurnal.stkippgritulungagung.ac.id/index.php/joincos/article/view/8105>
- Arima, S. P., Putri, U. M., & Bangsa, M. T. A. (2024). PERANCANGAN SISTEM ADMINISTRASI PEMBAYARAN SUMBANGAN PEMBINAAN PENDIDIKAN (SPP) BERBASIS WEBSITE PADA PESANTREN QORYATULHUFFAZH MENGGUNAKAN METODE EXTREME PROGRAMMING. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(6).
- Munawar. (2021). *Analisis Perancangan Sistem Berorientasi Objek dengan UML (Unified Modeling Language)*. Informatika Bandung.
- Purnama, I. (2023). Clinical Information System Using Extreme Programming Method. *International Journal of Science, Technology & Management*, 4(5), 1229–1235. <https://doi.org/10.46729/IJSTM.V4I5.931>
- Rosa, A. S., & Shalahuddin, M. (2019). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Informatika Bandung.
- Sahara, S., & Romadona, F. A. (2024). Pengaruh Sistem Informasi Logistik Terhadap Efisiensi Pengiriman Barang (Studi Kasus Pada PT XYZ). *Public Service and Governance Journal*, 5(1), 05–15. <https://doi.org/10.56444/PSGJ.V5I1.1213>