

## Implementasi Algoritma A\* pada Pengantaran Paket Barang Berdasarkan Rute Terpendek di Perusahaan Ekspedisi

Hendri Pratama<sup>1)</sup>, Felicia Sylviana<sup>2)</sup>, Abertun Sagit Sahay<sup>3)</sup>

<sup>1)2)3)</sup>Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya  
Jalan Hendrik Timang, Kota Palangka Raya, Kalimantan Tengah, Indonesia

<sup>1)</sup>hendriprstama447@gmail.com

<sup>2)</sup>felicia.upr@it.upr.ac.id

<sup>3)</sup>abertun@it.upr.ac.id

### Abstrak

Layanan ekspedisi barang menuntut efisiensi dan kecepatan dalam proses pengiriman untuk menjaga kualitas layanan dan daya saing perusahaan. Namun, kurir sering kali harus menentukan rute pengantaran secara manual, yang dapat menyebabkan ketidakefisienan, peningkatan biaya operasional, dan keterlambatan pengiriman.

Penelitian ini bertujuan untuk mengembangkan sistem pengantaran paket dengan integrasi algoritma A\* untuk pencarian rute terpendek dan metode klusterisasi K-Means untuk pembagian wilayah kerja kurir. Sistem dibangun menggunakan pendekatan pengembangan perangkat lunak Waterfall, melalui tahapan analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan.

Algoritma A\* diterapkan untuk menghasilkan rute optimal berdasarkan data lokasi pengiriman harian, sementara K-Means digunakan untuk mengelompokkan tujuan pengiriman berdasarkan kedekatan geografis. Sistem ini dikembangkan dengan bahasa pemrograman PHP dan JavaScript serta basis data MySQL, dan didukung dokumentasi teknis berupa Flowchart, DFD, dan ERD. Hasil pengujian menggunakan metode BlackBox menunjukkan bahwa sistem berfungsi sesuai dengan harapan. Secara keseluruhan, sistem telah berhasil diimplementasikan dan mampu meningkatkan efisiensi proses pengiriman serta pengelolaan tugas kurir dalam perusahaan ekspedisi.

**Kata kunci:** Algoritma A\*, K-Means, Optimasi Rute, Kurir, Ekspedisi

### Abstract

*The efficiency and speed of package delivery are critical factors in maintaining service quality and competitiveness in the logistics industry. However, couriers often determine delivery routes manually, leading to inefficiencies, increased operational costs, and delayed shipments.*

*This research aims to develop a package delivery system integrating the A\* algorithm for shortest path finding and K-Means clustering for courier area division. The system was developed using the Waterfall software development model, which includes requirement analysis, design, implementation, testing, and maintenance stages.*

*The A\* algorithm is applied to generate optimal delivery routes based on daily delivery location data, while K-Means is used to group delivery points geographically. The system was built using PHP and JavaScript, with MySQL as the database, and supported by technical documentation such as Flowcharts, DFD, and ERD. Testing using the BlackBox method showed that the system functions as expected. Overall, the system has been successfully implemented and improves the efficiency of delivery processes and task management for couriers in logistics companies.*

**Keywords:** A\* Algorithm, K-Means, Route Optimization, Courier, Logistics

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Ekspedisi pengiriman barang merupakan layanan yang bertujuan untuk mendistribusikan paket dari pengirim ke penerima dalam jangka waktu tertentu. Layanan ini mencakup berbagai proses, mulai dari penerimaan barang, penyortiran, hingga pengantaran ke alamat tujuan. Dalam operasionalnya, perusahaan ekspedisi mengandalkan kurir untuk mengantarkan paket ke berbagai lokasi yang tersebar di suatu wilayah.

Dalam industri logistik dan ekspedisi, kecepatan dan efisiensi dalam pengiriman barang menjadi faktor utama yang memengaruhi kualitas layanan. Setiap harinya, kurir bertugas mengantarkan paket ke berbagai lokasi yang tersebar di suatu wilayah. Namun, dalam praktiknya, banyak kurir masih harus menentukan rute perjalanan mereka secara manual, mengandalkan pengalaman pribadi atau menggunakan peta sederhana tanpa sistem navigasi yang optimal.

Metode penyusunan rute yang kurang terstruktur ini sering kali menyebabkan ketidakefisienan dalam perjalanan, seperti memilih jalur yang lebih jauh atau mengunjungi lokasi dalam urutan yang tidak optimal. Akibatnya, proses pengiriman menjadi lebih lama, konsumsi bahan bakar meningkat, dan biaya operasional pun bertambah. Dalam skala besar, masalah ini dapat berdampak pada keterlambatan pengiriman, kepuasan pelanggan yang menurun, serta penurunan daya saing perusahaan ekspedisi.

Untuk mengatasi permasalahan ini, diperlukan sistem yang dapat secara otomatis mencari dan merekomendasikan rute tercepat dan paling efisien bagi kurir. Salah satu algoritma yang dapat digunakan adalah Algoritma A\* (A Star), yang dikenal sebagai metode pencarian jalur terpendek dengan pendekatan heuristik. Algoritma ini tidak hanya mempertimbangkan jarak yang telah ditempuh, tetapi juga memperkirakan jarak menuju tujuan dengan cara yang lebih optimal dibandingkan metode pencarian rute lainnya.

Dengan menerapkan algoritma A\* dalam sistem pengelolaan ekspedisi, kurir dapat memperoleh rute terbaik secara real-time tanpa harus menyusunnya secara manual. Hal ini akan meningkatkan efisiensi waktu tempuh, mengurangi biaya operasional, serta memastikan bahwa barang sampai ke pelanggan dengan lebih cepat dan tepat waktu. Oleh karena itu, penelitian ini berfokus pada penerapan algoritma A\* untuk membantu kurir dalam menemukan rute pengiriman yang paling optimal, sehingga proses distribusi barang menjadi lebih efektif dan terorganisir.

A\* adalah algoritma pencarian best-first (terbaik-pertama) yang pseudokodenya ditunjukkan pada Algoritma 1. Algoritma ini menyimpan dua himpunan simpul, yaitu daftar OPEN dan daftar CLOSED. Daftar CLOSED berisi simpul-simpul yang telah dikembangkan, yaitu simpul yang penerus-penerusnya telah dihasilkan. Sementara itu, daftar OPEN berisi simpul-simpul yang telah dihasilkan, tetapi masih menunggu untuk dikembangkan. Pada setiap iterasi dari perulangan utama (while loop) dalam algoritma, A\* akan memilih satu simpul dari daftar OPEN yang memiliki nilai  $f$  terkecil untuk dikembangkan. Nilai  $f$  dari sebuah simpul  $n$  didefinisikan sebagai  $f(n) = g(n) + h(n)$ , di mana  $g(n)$  adalah biaya dari jalur terbaik yang diketahui dari simpul awal  $s_0$  hingga simpul  $n$ , dan  $h(n)$  adalah estimasi (heuristik) biaya dari simpul  $n$  menuju simpul tujuan terdekat. Dengan demikian,  $f(n)$  memberikan estimasi terhadap total biaya solusi terpendek yang melalui simpul  $n$  [1].

Namun, selain tantangan dalam menentukan rute perjalanan kurir, masih terdapat permasalahan lain yang berkontribusi pada ketidakefisienan pengantaran paket, yaitu metode penugasan paket oleh operator. Dalam sistem yang masih bersifat manual, operator harus menugaskan paket satu per satu kepada masing-masing kurir tanpa adanya sistem otomatisasi yang dapat mengoptimalkan distribusi tugas. Pendekatan ini mengakibatkan beban kerja operator semakin tinggi dan memperlambat proses distribusi paket, terutama saat volume pengiriman meningkat tajam.

Proses penugasan manual ini juga sering kali menyebabkan sebagian besar paket yang diberikan kepada seorang kurir bisa jadi tidak berada dalam jalur pengantaran yang optimal. Hal

ini dapat mengakibatkan kurir harus menempuh perjalanan yang lebih jauh, membuang waktu di jalan, dan pada akhirnya menyebabkan keterlambatan dalam pengiriman.

Untuk mengatasi permasalahan ini, penulis mengusulkan solusi dengan menerapkan algoritma klasterisasi K-Means dalam proses penugasan paket kepada kurir. Algoritma ini akan mengelompokkan paket berdasarkan lokasi geografisnya sehingga setiap kurir hanya menangani paket dalam area tertentu yang telah diklasterkan. Dengan pendekatan ini, distribusi tugas menjadi lebih merata dan setiap kurir memiliki rute pengantaran yang lebih terstruktur.

K-Means clustering ini merupakan metode klasterisasi partisi yang paling umum digunakan. Prosesnya dimulai dengan memilih K titik perwakilan sebagai centroid awal. Setiap data kemudian dikelompokkan ke dalam klaster berdasarkan centroid terdekat, dengan menggunakan ukuran kedekatan tertentu. Setelah klaster terbentuk, centroid dari masing-masing klaster diperbarui. Algoritma ini kemudian mengulangi proses tersebut secara iteratif hingga centroid tidak lagi berubah atau hingga terpenuhi kriteria konvergensi lainnya yang lebih longgar [2].

Klasterisasi K-Means bekerja dengan membagi sekumpulan titik (lokasi paket) ke dalam beberapa kelompok (klaster) berdasarkan kedekatan geografisnya. Dengan menerapkan metode ini, operator tidak lagi perlu menugaskan paket secara manual satu per satu. Sebaliknya, sistem akan secara otomatis menentukan kelompok paket yang harus diantarkan oleh masing-masing kurir, sehingga perjalanan menjadi lebih efisien dan waktu pengiriman dapat dipersingkat.

Dengan menggabungkan klasterisasi K-Means untuk pembagian tugas serta algoritma A\* untuk pencarian rute optimal, sistem ekspedisi dapat meningkatkan efisiensi operasional secara signifikan. Solusi ini tidak hanya mengurangi beban kerja operator, tetapi juga memastikan bahwa setiap kurir mendapatkan tugas yang optimal dengan rute tercepat. Pada akhirnya, implementasi solusi ini akan berdampak positif pada peningkatan kecepatan pengiriman, pengurangan biaya operasional, serta kepuasan pelanggan yang lebih baik.

Oleh sebab itu penulis akan memberikan judul "Implementasi Algoritma A Star Pada Pengantaran Paket Barang Berdasarkan Rute Terpendek Di Perusahaan Ekspedisi".

### 1.1.1 Subbagian dengan Heading 3

Subbagian ini merupakan contoh subbagian dengan level tiga (menggunakan Heading 3). Aturan penulisan judul subbagian sama dengan aturan pada subbagian sebelumnya.

### Subbagian dengan heading 4

Subbagian ini merupakan contoh subbagian dengan level empat (menggunakan Heading 4). Aturan penulisan judul hanya huruf depan judul saja yang menggunakan huruf besar dan penulisan judul tanpa disertai *numbering*.

## 2. TINJAUAN PUSTAKA

Dalam penelitian "Implementasi Algoritma A Star Pada Pengantaran Paket Barang Berdasarkan Rute Terpendek Di Perusahaan Ekspedisi" beberapa studi sebelumnya menjadi acuan. Sandy Purnama dan Dyah Ayu Megawaty (2018) menciptakan aplikasi pencarian jarak terdekat menuju wisata kuliner di kota bandar lampung [3]. Mukhtar et al. (2021) menerapkan A Star untuk pencarian rute terpendek (*Shortest Path Problem*) pada pencarian kantor pos di Kota Pekanbaru [4]. Syihabuddin et al. (2022) melakukan penelitian efektifitas A Star dalam menentukan rute terpendek untuk wisata di Kota Malang [5]

## 3. METODE PENELITIAN

Untuk membangun website rute terpendek, proses pengembangannya menggunakan metode Waterfall. Metode *Waterfall* adalah model pengembangan perangkat lunak yang memperlihatkan proses pengembangan secara berurutan dan linier. Dalam pendekatan ini, setiap fase harus diselesaikan sepenuhnya sebelum melanjutkan ke fase berikutnya. Metode ini menyiratkan bahwa setiap langkah dalam pengembangan software bersifat

tidak terbalik, sehingga jika ada perubahan di tengah proses, dapat sangat sulit dan mahal untuk diterapkan [6].

### 3.1 Analisis Kebutuhan

Pada tahap ini, pengembang dan pemangku kepentingan melakukan dialog untuk mengumpulkan semua kebutuhan sistem. Hal ini mencakup pemahaman tentang apa yang diinginkan dan dibutuhkan oleh pengguna.

### 3.2 Desain Sistem

Pada tahap desain, arsitektur sistem dan desain komponen spesifik ditentukan berdasarkan kebutuhan yang telah diidentifikasi. Ini termasuk desain antarmuka pengguna, database, dan sistem yang lebih besar.

### 3.3 Implementasi

Setelah desain selesai, pengembang mulai menulis kode untuk mengimplementasikan desain yang telah dibuat. Ini adalah tahap di mana coding dilakukan, dan selanjutnya perangkat lunak dibangun menjadi sebuah sistem yang berfungsi.

### 3.4 Pengujian

Pada tahap ini, perangkat lunak diuji untuk mendeteksi dan memperbaiki kesalahan. Pengujian dapat mencakup berbagai jenis, seperti pengujian unit, integrasi, sistem, dan penerimaan.

### 3.5 Pemeliharaan

Setelah perangkat lunak diimplementasikan dan diserahkan kepada pengguna, tahap pemeliharaan dimulai. Ini mencakup perbaikan bug, pembaruan, dan penambahan fitur baru berdasarkan umpan balik dari pengguna.

## 4. PEMBAHASAN

### 4.1 Analisis Kebutuhan

Pada tahap ini mengidentifikasi masalah, kebutuhan pengguna, dan kebutuhan sistem.

#### 4.1.1 Identifikasi Masalah

Permasalahan yang timbul saat ini adalah belum adanya sistem penugasan semua paket barang ke kurir dan pencarian rute terpendek untuk kurir yang melakukan pengantaran, yang dimana sekarang operator masih menugaskan paket barang satu persatu kepada kurir yang tersedia dan kurir juga masih melakukan pengantaran mengandalkan membaca alamat satu persatu dan kemudian menyusun rute untuk dilakukan pengantaran, hal ini menyebabkan keterlambatan dalam penugasan paket barang kepada kurir dan berkurangnya efisiensi dalam pengantaran paket.

#### 4.1.2 Kebutuhan Pengguna

##### Operator

Operator sudah memiliki *username* berupa *format email* dan *password*, dapat menambahkan data operator baru, memiliki hak akses penuh dalam *website*, dan Memiliki hak akses untuk mengelola paket dan kurir.

##### Kurir

Kurir dibuatkan *username format email* dan *password* oleh operator untuk *Login* ke *website*, dapat *Login* menggunakan *Username* dan *password*, dan dapat melihat detail tugas paket yang harus diantar.

### 4.1.3 Kebutuhan Sistem

Operator dapat mengelola tugas kurir dan dapat menambahkan Akun Kurir baru. Setiap kurir memiliki atribut atribut : nama kurir, username, password, dan nomor telepon. Setiap paket memiliki atribut : resi paket, nama pengirim, nama penerima, alamat penerima, nomor telepon penerima, deskripsi paket, status paket, Update terakhir paket, waktu Update, tanggal pembuatan, dan titik koordinat paket. Paket yang ditugaskan kepada kurir akan dibagi menjadi beberapa kluster tergantung jumlah kurir. Kurir dapat melakukan pencarian rute terpendek menuju alamat penerima berdasarkan lokasi nya sekarang. Kurir dapat melakukan pencarian rute keseluruhan tugasnya hanya dengan satu kali permintaan. Kurir dapat mengubah status paket jika paket tersebut berhasil diantar ke penerima. Terdapat 3 status paket yaitu : Paket Sedang Di Proses, Paket Menuju Alamat Pengantaran, dan Paket Diterima Oleh Pembeli

## 4.2 Desain Sistem

### 4.2.1 Konsep Algoritma *Klustering K-Means*

#### 1) Penyiapan Data

Sistem mengambil data koordinat geografis (Latitude & Longitude) dari paket yang sedang dalam status "Paket Sedang Diproses" pada tanggal saat ini dan data resi dari paket juga disimpan untuk pelacakan identitas.

#### 2) Inisialisasi *Centroid* Awal (*K-Means*)

Pilih satu titik data secara acak sebagai centroid pertama. Untuk setiap titik data lainnya, hitung jarak ke centroid yang sudah dipilih. Pilih titik yang paling jauh untuk menjadi centroid berikutnya. Ulangi hingga sebanyak k centroid diperoleh.

#### 3) Proses Klasterisasi

Setiap paket dihitung jaraknya ke semua centroid. Paket dikelompokkan ke kluster dengan centroid terdekat. Sistem menghitung centroid baru dari hasil klasterisasi sementara. Proses ini berulang hingga tidak ada perubahan dalam pembagian kluster (konvergen).

#### 4) Pemerataan Kluster (*Rebalancing*)

Jika terdapat kluster yang jumlahnya kurang dari rata-rata minimal, maka: Sistem mencari kluster yang kelebihan (overload). Paket dari kluster overload dipindahkan ke kluster yang kekurangan, berdasarkan jarak terdekat ke centroid tujuan.

#### 5) Optimasi Kluster (*Evaluate and Swap*)

Sistem mengevaluasi efisiensi distribusi dengan membandingkan jarak sebelum dan sesudah pertukaran dua paket dari kluster yang berbeda. Jika hasil pertukaran bisa mengurangi total jarak tempuh, maka sistem melakukan swap (pertukaran kluster) antara kedua paket.

#### 6) Redistribusi Kluster

Jika setelah rebalancing masih ada kluster yang kelebihan jumlah paket, maka : Paket dari kluster tersebut dialihkan ke kluster terdekat yang masih dalam batas wajar. Proses ini dilakukan berdasarkan prioritas jarak antar centroid.

#### 7) Hasil Akhir Klasterisasi

Setelah seluruh proses rebalancing dan redistribusi selesai, sistem kembali menghitung ulang centroid baru dari masing-masing kluster.

### 4.2.2 Konsep Algoritma *A Star*

#### 1) Inisialisasi nilai awal

Algoritma mulai dari simpul awal. Untuk setiap simpul:  $g(n)$  (biaya dari awal ke simpul n) diset ke tak hingga ( $\infty$ ).  $f(n)$  (total estimasi biaya:  $g(n) + h(n)$ ) juga diset ke  $\infty$ .

Untuk simpul awal:  $g(\text{Start}) = 0$ .  $f(\text{Start}) = h(\text{Start})$ , yaitu perkiraan jarak dari Start ke goal menggunakan fungsi heuristik haversine.

Simpul awal dimasukkan ke dalam antrian prioritas (openSet), yang akan digunakan untuk menentukan simpul mana yang dievaluasi selanjutnya.

#### 2) Memilih simpul dengan nilai $f(n)$ terkecil

Algoritma akan memilih simpul dari openSet yang memiliki nilai  $f(n)$  paling kecil. Simpul ini dianggap sebagai simpul yang paling menjanjikan untuk menuju tujuan (karena total biaya perjalanannya diperkirakan paling kecil).

3) Mengevaluasi tetangga dari simpul saat ini

Untuk setiap simpul tetangga dari simpul saat ini: Hitung perkiraan biaya perjalanan aktual dari Start ke tetangga, yaitu: Tambahkan jarak dari simpul awal ke simpul saat ini ( $g(\text{current})$ ) Ditambah jarak langsung dari simpul saat ini ke tetangganya Hasilnya adalah nilai  $g(n)$  yang baru untuk tetangga Bandingkan nilai  $g(n)$  baru ini dengan nilai sebelumnya: Jika lebih kecil, berarti ditemukan jalur yang lebih baik Maka: Simpan jalur terbaik dengan mencatat bahwa simpul tetangga berasal dari simpul saat ini (cameFrom) Perbarui nilai  $g(n)$  untuk tetangga Hitung nilai  $f(n)$  untuk tetangga dengan:  $f_n = g(n) + h(n)$

di mana:  $g(n)$  adalah biaya aktual yang baru ditemukan  $h(n)$  adalah estimasi jarak dari tetangga ke tujuan dan Masukkan simpul tetangga ke dalam antrian openSet agar diperiksa nanti

4) Menandai simpul yang sudah diperiksa

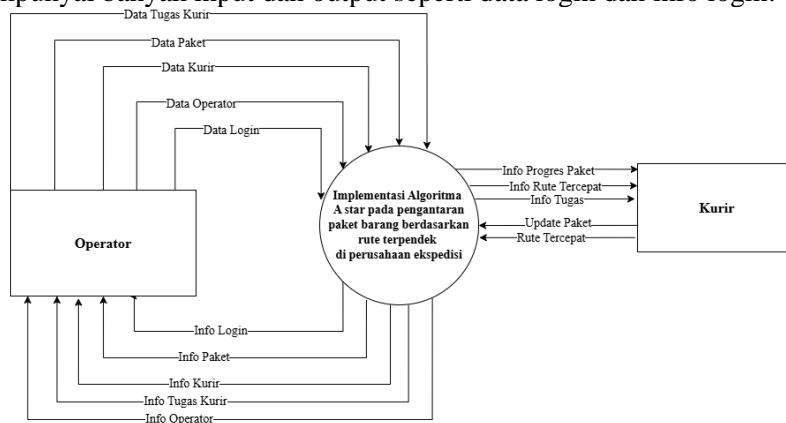
Setelah semua tetangga diperiksa, simpul saat ini dianggap sudah selesai dan tidak akan diperiksa lagi dan simpul ini dimasukkan ke dalam kumpulan closedSet (simpul yang sudah selesai diproses).

5) Menyusun kembali jalur dari tujuan ke awal

Setelah simpul tujuan ditemukan, algoritma akan menyusun kembali jalur dari tujuan ke awal. Ini dilakukan dengan melihat data cameFrom, yaitu catatan dari mana setiap simpul berasal. Jalur disusun mundur (dari goal ke Start), lalu dibalik agar urutannya menjadi dari Start ke goal.

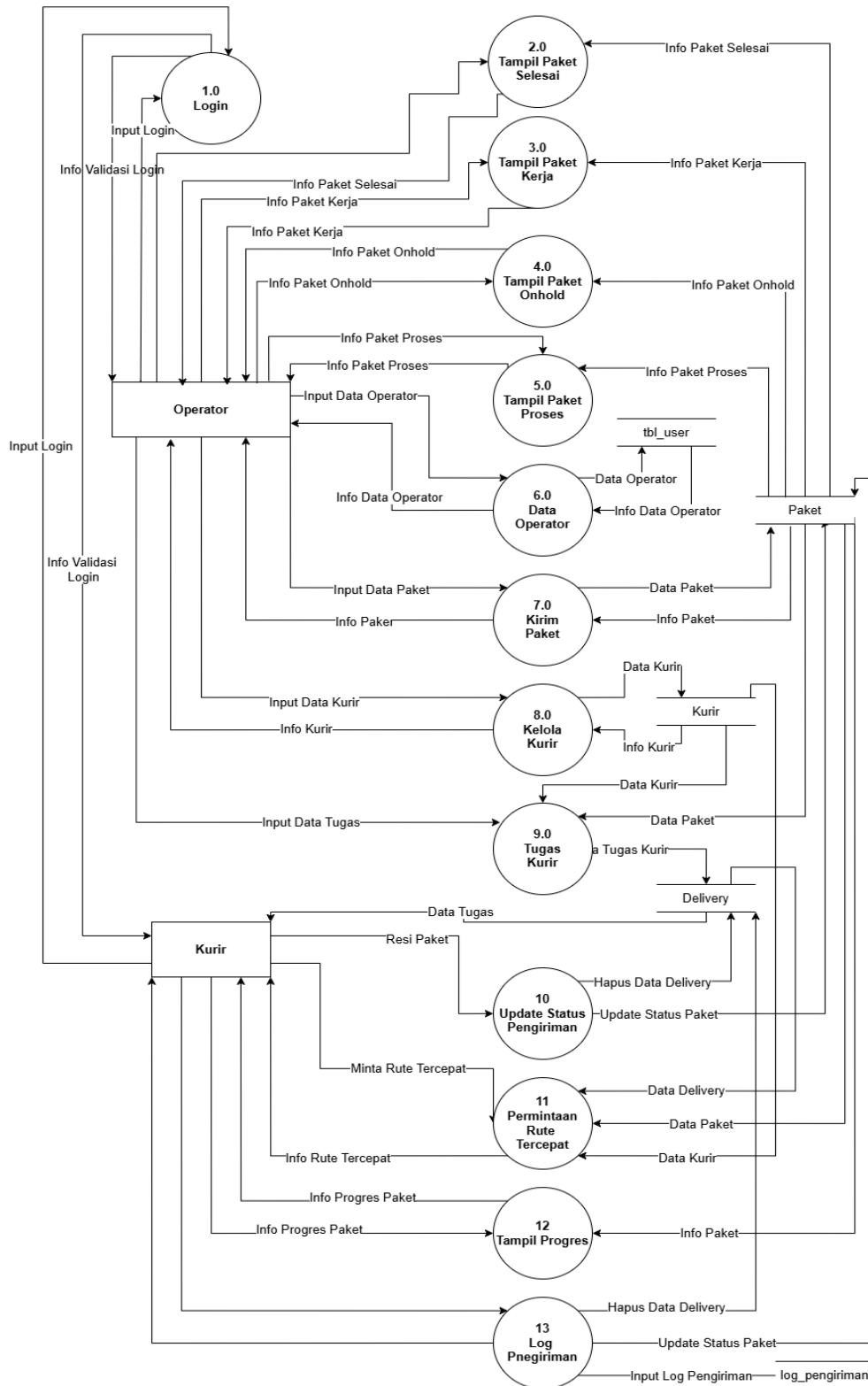
### 4.2.3 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan Diagram yang memvisualisasikan aliran data dalam suatu sistem. DFD menunjukkan pergerakan data dari satu proses ke proses lainnya serta bagaimana data disimpan dalam sistem. Diagram ini sering dimanfaatkan dalam analisis dan perancangan sistem untuk memahami cara informasi diproses dalam sebuah organisasi atau aplikasi. Pada gambar 1 merupakan DFD level 0 terdapat dua entitas luar yaitu Operator dan Kurir lalu mempunyai banyak input dan output seperti data login dan info login.



Gambar 1. Data Flow Diagram Level 0

Pada gambar 2 terdapat lanjutan setelah DFD level 0 yaitu DFD level 1 yang terdapat keseluruhan input dan output dari keseluruhan sistem.

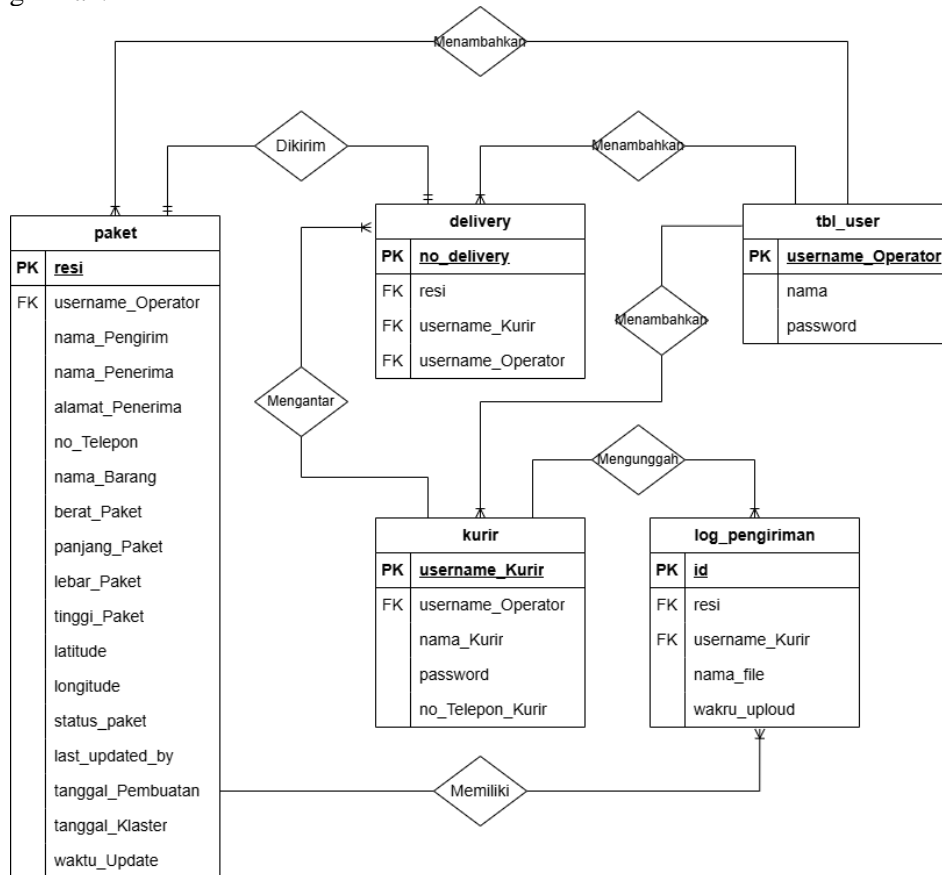


Gambar 2. Data Flow Diagram Level 1

#### 4.2.4 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah Diagram yang digunakan untuk menggambarkan relasi antara entitas dalam suatu sistem. Diagram ini menunjukkan entitas yang ada, atribut yang melekat padanya, dan hubungan antar entitas. ERD sangat berguna dalam perancangan basis

data untuk memberikan gambaran yang jelas tentang struktur data. Pada gambar 3 merupakan ERD dari sistem yang terdapat lima entitas yaitu paket, delivery, tbl\_user, kurir, dan log\_pengiriman.



Gambar 3. Entity Relationship Diagram

**4.2.5 Desain Tabel Database**

Dalam pembuatan website Implementasi Algoritma A\* Pada Pengantaran Paket Barang Berdasarkan Rute Terpendek Di Perusahaan Ekspedisi ini terdapat 5 tabel yaitu : tabel user, tabel kurir, tabel paket, tabel log\_pengiriman, dan tabel delivery. Pada tabel 1 merupakan tabel user yang terdapat nama, username\_Operator, dan password.

Tabel 1. Tabel User

No	Field	Tipe Data	Lebar	Keterangan
1	Nama	Varchar	50	Not Null
2	Username_Operator	Varchar	50	Primary Key
3	Password	Varchar	10	Not Null

Tabel 2. Tabel Paket

No	Field	Tipe Data	Lebar	Keterangan
1	resi	Integer	50	Primary Key
2	Username_Operator	Varchar	50	Foreign Key (Tabel User)
3	nama_Pengirim	Varchar	50	Not Null
4	nama_Penerima	Varchar	50	Not Null
5	alamat_Penerima	Varchar	255	Not Null
6	no_Telepon	Varchar	20	Not Null
7	nama_Barang	Varchar	50	Not Null
8	berat_Paket	Integer	10	Not Null
9	panjang_Paket	Integer	10	Not Null
10	lebar_Paket	Integer	10	Not Null
11	tinggi_Paket	Integer	10	Not Null
12	Latitude	double	-	Not Null
13	Longitude	double	-	Not Null
14	status_paket	Varchar	50	Not Null
15	last_Updated_by	Varchar	50	Not Null
16	tanggal_pembuatan	Date	-	Not Null
17	tanggal_Klaster	Date	-	Not Null
18	waktu_Update	Datetime	-	Not Null

Tabel 3. Tabel Kurir

No	Field	Tipe Data	Lebar	Keterangan
1	Username_Kurir	Varchar	50	Primary Key
2	Username_Operator	Varchar	50	Foreign Key (Tabel user)
3	nama_Kurir	Varchar	50	Not Null
4	password	Varchar	50	Not Null
5	no_Telepon_Kurir	Varchar	50	Not Null

Tabel 4. Tabel Delivery

No	Field	Tipe Data	Lebar	Keterangan
1	no_delivery	Integer	50	Primary Key
2	resi	Integer	50	Foreign Key (Tabel paket)
3	username_Kurir	Varchar	50	Foreign Key (Tabel kurir)
4	Username_Operator	Varchar	50	Foreign Key (Tabel user)

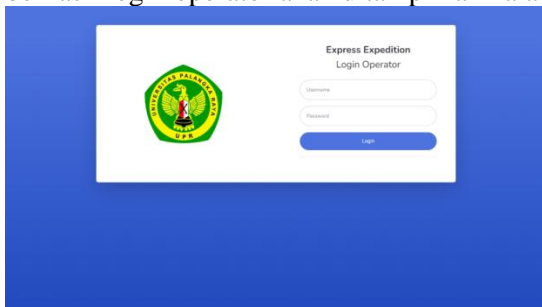
Tabel 5. Tabel Log Pengiriman

No	Field	Tipe Data	Lebar	Keterangan
1	id	Integer	10	Primary Key
2	resi	Integer	20	Foreign Key (Tabel paket)
3	nama_file	Varchar	50	Not null
4	waktu_upload	Datetime	-	Not null
	uploaded_by	Varchar	50	Not null

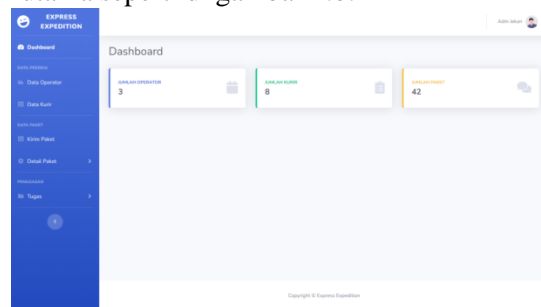
### 4.3 Implementasi

#### 4.3.1 Implementasi Antar Muka Operator

Saat operator menggunakan website, halaman pertama yang ditampilkan adalah seperti gambar 4.a terdapat form untuk *input username* dan *password* lalu ada tombol login. Setelah berhasil login operator akan ditampilkan halaman utama seperti di gambar 4.b.

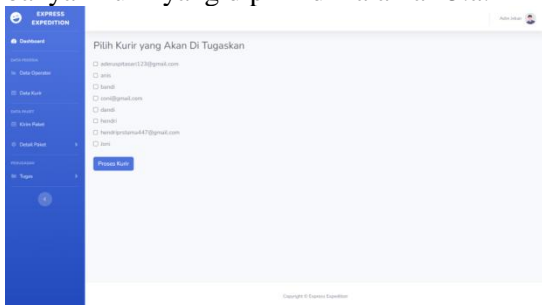


Gambar 4.a Halaman Login Operator

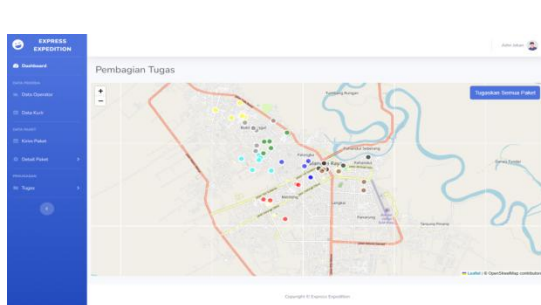


Gambar 4.b Halaman Utama

Dibagian operator terdapat proses pemilihan jumlah kurir untuk di klusterisasi pada gambar 5.a lalu proses kluster nya di gambar 5.b. Hasil jumlah kluster yang didapat tergantung seberapa banyak kurir yang dipilih di halaman 5.a.



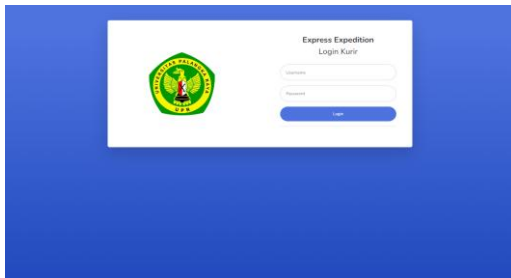
Gambar 5.a Halaman Pilih Kurir



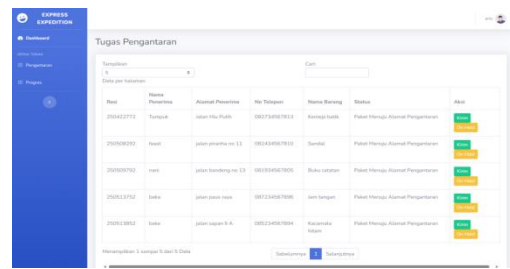
Gambar 5.b Halaman Hasil Kluster

#### Kurir

Saat kurir pertama kali mengakses *website* maka halaman pertama yang dikunjungi kurir adalah halaman login seperti pada gambar 6.a lalu setelah kurir melakukan login terdapat halaman utama pada gambar 6.b.



Gambar 6.a Halaman Login Kurir



Gambar 6.b Halaman Utama Kurir

Dihalaman kurir juga terdapat proses Algoritma *A Star* yang prosesnya akan dijalankan jika kurir mengklik tombol mulai cari rute seperti pada gambar 7.



Gambar 7. Halaman Proses Cari Rute

#### 4.4 Pengujian

Tabel 6. Tabel *Blackbox* Pengujian Operator

No	Kondisi Awal	Kondisi yang diinginkan	Kondisi hasil
1	Melakukan login dan klik tombol login	Login berhasil lalu menuju ke halaman utama operator	Login berhasil lalu menuju ke halaman utama operator
2	Pilih semua kemungkinan kurir yang bisa	Inputan sesuai dan hasil klaster yang ditampilkan sesuai dengan jmlah yang diinputkan	Inputan sesuai dan hasil klaster yang ditampilkan sesuai dengan jmlah yang diinputkan
3	Terdapat data paket yang belum ditugaskan lalu mengklik tombol tugaskan semua paket	Semua data paket yang belum ditugaskan akan masuk ke tabel deivery didatabase dan status paket akan berubah menjadi Paket Menuju Alamat Pengantaran	Semua data paket yang belum ditugaskan akan masuk ke tabel deivery didatabase dan status paket akan berubah menjadi Paket Menuju Alamat Pengantaran

Tabel 7. Tabel *Blackbox* Pengujian Kurir

No	Kondisi Awal	Kondisi yang diinginkan	Kondisi hasil
1	Melakukan login dan klik tombol login	Login berhasil lalu menuju ke halaman utama operator	Login berhasil lalu menuju ke halaman utama operator
2	Klik tombol mulai cari	Ditampilkan rute menuju	Ditampilkan rute menuju

---

rute	ke semua tujuan berdasarkan tujuan paling dekat dari lokasi saat ini	ke semua tujuan berdasarkan tujuan paling dekat dari lokasi saat ini
------	--	--

---

#### 4.5 Pemeliharaan

Setelah perangkat lunak diimplementasikan dan diserahkan kepada pengguna, tahap pemeliharaan dimulai. Ini mencakup perbaikan bug, pembaruan, dan penambahan fitur baru berdasarkan umpan balik dari pengguna.

### 5. KESIMPULAN

Berdasarkan penelitian dan pengembangan yang telah dilakukan, sistem pengantaran paket berbasis algoritma A\* dan klusterisasi K-Means telah berhasil dibangun dan berfungsi sesuai dengan tujuan. Implementasi algoritma A\* dalam pencarian rute pengantaran memungkinkan sistem menampilkan rute terpendek secara efisien berdasarkan data lokasi pengiriman harian, sehingga memudahkan kurir dalam melakukan pengantaran. Namun, perlu dicatat bahwa algoritma A\* yang digunakan saat ini hanya mempertimbangkan jarak tempuh terpendek tanpa memperhitungkan kondisi atau kualitas jalan yang dilalui.

Sementara itu, penggunaan K-Means dalam pembagian tugas kurir memungkinkan pendistribusian paket berdasarkan kedekatan geografis, yang secara langsung mendukung efisiensi kerja di lapangan. Sistem ini dirancang dengan pendekatan pengembangan perangkat lunak Waterfall dan telah melalui tahapan analisis kebutuhan, desain, implementasi, pengujian, serta pemeliharaan. Desain sistem ditunjang dengan representasi Diagram seperti Flowchart, DFD, dan ERD, serta dibangun menggunakan PHP, JavaScript, dan MySQL. Hasil pengujian menggunakan metode BlackBox menunjukkan bahwa fungsi-fungsi utama berjalan lancar dan sesuai dengan peran masing-masing pengguna, baik operator maupun kurir.

Secara keseluruhan, sistem ini telah menunjukkan performa yang stabil dan memenuhi kebutuhan operasional perusahaan ekspedisi, baik dari sisi pengelolaan tugas maupun efisiensi pengantaran, sehingga dapat disimpulkan bahwa sistem ini telah berhasil diimplementasikan dengan baik.

### DAFTAR PUSTAKA

- [1] A. Fukunaga, A. Botea, Y. Jinnai, and A. Kishimoto, "A Survey of Parallel A\*," pp. 1–25, 2017, [Online]. Available: <http://arxiv.org/abs/1708.05296>
- [2] C. K. Reddy and B. Vinzamuri, *A Survey of Partitional and Hierarchical Clustering Algorithms*. 2019. doi: 10.1201/9781315373515-4.
- [3] Y. F. Sandy Purnama, Dyah Ayu Megawaty, "Jarak Terdekat Wisata Kuliner Di Kota Bandarlampung," *Teknologi*, vol. 12, no. 1, pp. 28–32, 2018.
- [4] H. Mukhtar, Y. Hendri, and S. Soni, "Implementasi Algoritma a Star Dalam Pencarian Rute Terpendek (Shortest Path Problem) Pada Sistem Pencarian Kantor Pos Di Kota Pekanbaru," *J. Softw. Eng. Inf. Syst.*, vol. 2, no. 1, pp. 111–119, 2021, doi: 10.37859/seis.v2i1.3313.
- [5] R. F. Syihabuddin, M. N. Jauhari, M. Khudzaifah, and H. Fahmi, "Implementasi Algoritma A-Star dalam Menentukan Rute Terpendek Destinasi Wisata Kota Malang," *J. Ris. Mhs. Mat.*, vol. 1, no. 5, pp. 236–245, 2022.
- [6] I. Sommerville, *Ninth Edition*. 2011.