

# BAB 1

## USULAN GAGASAN

### 1.1 Latar Belakang Masalah

Saat bepergian kesuatu destinasi, tidak jarang mengalami kendala pada perjalanan. Pengendara kerap kali mengalami kendala pada saat menggunakan rute yang umum khususnya daerah Kota Bandung sehingga kendala seperti kemacetan, jalan ditutup sementara, ataupun cuaca yang tidak mendukung[1]. Dikarenakan daerah perkotaan khususnya Kota Bandung memiliki jalan utama yang sering kali mengalami kepadatan lalu lintas khususnya pada kemacetan, beberapa pengendara motor maupun mobil kerap mencari jalan alternatif untuk mencari rute perjalanan yang lebih cepat. Kemacetan dapat di prediksi berdasarkan waktu, cuaca, kondisi lalu lintas, dan suhu [2]. Akan tetapi, pengendara khususnya mobil kerap tidak mengetahui apakah jalan alternatif tersebut dapat dilalui oleh mobil atau idealnya hanya pengguna motor.

Untuk mengatasi hal tersebut, diperlukan suatu sistem untuk mencarikan pengguna rute terbaik hingga rute alternatif namun tidak mengubah angka kepadatan lalu lintas. Oleh karena itu, dengan merancang sistem rute rekomendasi yang interaktif ini diharapkan dapat memberikan solusi bagi pengguna khususnya pengendara umum untuk mendapatkan rute tercepat untuk mencapai tujuan. *Interactive Route Recommendation System* merupakan sistem yang dimana pengguna di berikan rute yang terbaik bagi pengendara.

### 1.2 Informasi Pendukung Masalah

Permasalahan pada rute rekomendasi tidak hanya cara untuk merekomendasi sebuah rute, melainkan keragaman jenis kendaraan oleh pengendara khususnya di Indonesia. Di Indonesia di dominasikan oleh 77.5% oleh pengendara motor sementara pada jenis lainnya seperti mobil pribadi, bus dan truk. Hal ini dikarenakan lalu lintas yang heterogen dan mempengaruhi kondisi dalam beberapa aspek seperti kenyamanan berkendara, atau kemacetan[2]. Bandung merupakan salah satu kota terpadat di Asia yaitu peringkat ke-14. Bandung juga menjadi kota termacet pertama di Indonesia [3]. Agar pengendara mobil dan motor mendapatkan rute terbaik ke tujuan, dibutuhkan sistem pencari rute terbaik yang menggunakan *multi-criteria decision making* (MCDM). MCDM digunakan agar pengendara mendapatkan rute terbaik dikarenakan telah dikalkulasikan keputusan yang akan di ambil [4]. Menggunakan Algoritma A\* maupun

Algoritma *Dijkstra* menjadi opsi dalam pembuatan *Interactive Route Recommendation System* ini. Algoritma *Dijkstra* merupakan algoritma yang digunakan untuk menyelesaikan masalah jarak terpendek (*shortest path problem*) pada sebuah graf yang terarah (*directed graph*) [5] sedangkan Algoritma A\* merupakan pencarian rute terpendek yang merupakan perbaikan dari Algoritma *Best First Search* (BFS) dengan memodifikasi fungsi heuristiknya [6]. Algoritma A\* memiliki kapabilitas akurasi 30 arc/sec yang dimana baik untuk mengimplementasikannya dengan GPS [7]. Hal ini dikarenakan Algoritma A\* menggunakan *Heuristic Estimate* untuk mendapatkan rute terbaiknya sedangkan Algoritma *Dijkstra* lebih mengeksplorasi berbagai rute sehingga hasil pencarian rute tercepatnya lebih pelan dibandingkan Algoritma A\* [8]. Pembuatan *Route Recommendation* ini juga menggunakan *Dynamic Graph*. *Dynamic Graph* ini difungsikan untuk membuat node dan struktur data [9]. Dengan mempresentasikan *learning method* pada *Dynamic Graph*, proses mendapatkan informasi akan lebih efisien [10]. Pengimplementasian *Dynamic Graph* ini juga digunakan untuk membuat *Route Planning*. *Route planning* ini membutuhkan pencarian rute terbaik dari posisi pengguna saat ini sampai tujuan dengan perhitungan waktu tempuh dari setiap jalan dengan berbagai kemungkinan [11].

### **1.3 Analisis Umum**

#### 1.3.1. Aspek Manufakturabilitas

Produk ini memiliki tingkat kompleksitas relatif menengah, dikarenakan untuk membuatnya memerlukan berbagai penerapan seperti menerapkan algoritma, *database* dan API tomtom untuk memandu pengguna ke rute terbaik, membuat desain berbagai kemungkinan pada rute terbaik dan rute alternatif.

#### 1.3.2. Aspek Penggunaan (*usability*)

Aspek penggunaan dalam mengoperasikan sistem, pengguna menginput lokasi awal dan akhir. Sistem akan menampilkan jalur optimal dan memulai perjalanan. Jika selama perjalanan terjadi penutupan jalan atau kemacetan, sistem akan mengarahkan pengguna ke jalur alternatif.

#### 1.3.3. Aspek Keberlanjutan (*sustainability*)

Pembuatan sistem rekomendasi rute ini akan membantu pengguna dalam menemukan rute optimal selama perjalanan, sehingga dapat menghemat waktu, menghindari dari kemacetan dan meningkatkan efisiensi perjalanan.

## 1.4 Kebutuhan yang Harus Dipenuhi

Berdasarkan analisis yang telah dilakukan, kebutuhan yang harus dipenuhi dalam pembuatan sistem yaitu :

- a. Sistem dapat ditampilkan dan diakses pada tampilan website.
- b. Sistem dapat menampilkan rute terbaik.
- c. Program dirancang untuk menampilkan rute yang dapat berubah sesuai keadaan (dinamis)
- d. Program dirancang dengan tujuan untuk merekomendasikan rute terbaik, guna membantu pengguna untuk menghindari kondisi kemacetan.

## 1.5 Solusi Sistem yang Diusulkan

### 1.5.1. Karakteristik Produk

- 1) Fitur Utama:
  - a) Menampilkan rute perjalanan : menyajikan rute perjalanan yang akan ditempuh, dengan menggunakan inisial node awal ke node tujuan.
  - b) Pencarian jalur rute : memungkinkan pengguna untuk mencari jalur rute dengan cara menginput rute awal dan tujuan yang diinginkan.
- 2) Fitur Dasar:
  - a) Sistem dapat merekomendasikan rute terbaik, dapat menampilkan informasi setiap jalan, berdasarkan variabel yang akan digunakan pada sistem yaitu jarak, *density*, *popularity*, tipe jalan, ruas jalan, dan kondisi jalan.
- 3) Fitur Tambahan:
  - a) Di implementasikan pada *platform* Website
  - b) Jika rute yang telah ditentukan terputus, maka sistem akan secara otomatis mencari opsi rute alternatif yang dapat tempuh.
  - c) Fitur Tutup dan Buka Jalan memungkinkan simulasi perubahan rute dinamis, apabila rute yang ditempuh ada penutupan jalan, maka rute akan berubah.
- 4) Sifat solusi yang diharapkan:
  - a) Sistem mudah diakses dikarenakan diimplementasikan melalui *platform* website
  - b) Rute alternatif tetap mengarahkan pada *opsi* rute yang dapat diakses oleh berbagai jenis kendaraan.

## 1.5.2. Usulan Solusi

### 1.5.2.1 Solusi 1

Sistem akan diimplementasikan pada Website. Algoritma yang digunakan yaitu algoritma *Dijkstra* dikarenakan algoritma tersebut banyak digunakan berbagai penelitian pencarian jalur terpendek dari suatu titik ke titik dengan cara memperbarui nilai dari titik terdekat secara berurutan, sehingga jarak terpendek ke setiap titik akan ditemukan. Akan tetapi algoritma *Dijkstra* hanya berfungsi dengan baik pada graf yang berbobot positif dan tidak ada graf yang berbobot negatif.

Skenario penggunaan produk:

1. Saat pertama kali membuka *website*, diharapkan pengguna dapat menuliskan *node* awal dan tujuan yang diinginkan pada kolom pencarian rute utama.
2. Setelah menuliskan rute, akan memberikan rute kepada pengguna
3. Jika pada rute yang diberikan jalannya terputus, maka akan memberikan jalan yang terbaik bagi pengguna yang dapat dilalui oleh mobil maupun motor.
4. Setelah rute perjalanan selesai di tempuh, maka akan memberitahu pengguna telah sampai ditujuan.

### 1.5.2.2 Solusi 2

Untuk penggunaan produknya masih sama dengan solusi satu namun yang membedakan yaitu algoritma yang digunakan. Algoritma yang digunakan yaitu algoritma  $A^*$  (*A-star*). Algoritma  $A^*$  adalah algoritma pencarian yang digunakan untuk menemukan jalur terpendek dari suatu titik ke titik yang lainnya pada peta dengan menggunakan fungsi heuristik untuk mempercepat menemukan jalur terpendek. Prinsip algoritma  $A^*$  adalah melakukan perhitungan satu per satu dalam setiap titik untuk menemukan rute terpendek. Algoritma akan menghitung jarak lintasan, kemudian menyimpannya dan akan menghitung setiap node yang terbuka.

Algoritma  $A^*$  membutuhkan waktu yang lebih cepat untuk menemukan rute terpendek dari pada algoritma *Dijkstra* dengan menggunakan nilai tujuan akhir (*estimated cost to goal*) dan menggunakan pendekatan dengan mmempertimbangkan nilai  $f(n) = g(n) + h(n)$ , dimana  $g(n)$  adalah nilai dari titik awal ke titik yang sedang

diproses sedangkan  $h(n)$  adalah nilai *heuristic* dari titik yang sedang diproses ke titik tujuan.

## 1.6 Solusi yang Dipilih

Berdasarkan latar belakang dan kebutuhan, diputuskan bahwa solusi terbaik adalah menggunakan algoritma A\*. Pemilihan ini didasarkan pada pertimbangan bobot kriteria yang telah ditetapkan dalam sistem rute rekomendasi. Algoritma A\* dipilih karena dapat menyediakan waktu pemrosesan yang lebih cepat, sehingga data dan tampilan peta dapat sesuai. Keputusan ini diambil berdasarkan fakta bahwa algoritma A\* memiliki kemampuan untuk menghasilkan hasil yang akurat dengan efisiensi yang tinggi, terutama dalam menemukan rute terpendek. Dalam hal ini, algoritma Dijkstra juga merupakan pilihan yang kuat untuk menemukan rute terpendek.

Namun, algoritma A\* memiliki keunggulan dalam menggabungkan informasi biaya (*cost*) sejauh ini ( $g(n)$ ) dan estimasi biaya yang tersisa hingga tujuan ( $h(n)$ ). Ini memungkinkan A\* untuk fokus pada rute yang paling berpotensi mengarah langsung ke tujuan, yang dapat mengurangi jumlah simpul yang harus dieksplorasi dan menghasilkan solusi lebih cepat. Oleh karena itu, berdasarkan pertimbangan efisiensi dan akurasi, penggunaan algoritma A\* adalah pilihan yang lebih tepat dalam konteks sistem rute rekomendasi ini.

Tabel 1. 1 Perbandingan Alternatif Solusi

Alternatif Solusi	Aspek Manufakturabilitas	Aspek Penggunaan	Aspek Keberlanjutan
Algoritma <i>Dijkstra</i>	<p><b>Kelebihan:</b> Bagus untuk melakukan routing dan dapat menghasilkan tree rute terpendek</p> <p><b>Kekurangan:</b> Kurang cocok untuk penerapan <i>dynamic graph</i></p>	<p><b>Kelebihan:</b> Dari segi pengalaman pengguna, pengguna dapat disuguhkan ruteterpendek dengan akurasi lebih baik daripada Algoritma A*</p> <p><b>Kekurangan:</b> Saat memuat rute, akan membutuhkan waktu yang lebih lama daripada Algoritma A*</p>	<p><b>Kelebihan:</b> Mengimplementasikannya pada Website mendukung sistem rute rekomendasi ini menjadi lebih interaktif.</p> <p><b>Kekurangan:</b> Dalam segi performa, aplikasi native.</p>
Algoritma A*	<p><b>Kelebihan:</b> Bagus digunakan untuk mencari rute tercepat dengan lebih cepat dan optimal dibandingkan Algoritma <i>Dijkstra</i></p> <p><b>Kekurangan:</b> Disamping kelebihannya, ada hal yang harus dikurangi yaitu keakuratan.</p>	<p><b>Kelebihan:</b> Dari segi pengalaman pengguna, pengguna akan disuguhkan ruteterpendek lebih cepat dan optimal dibandingkan Algoritma <i>Dijkstra</i></p> <p><b>Kekurangan:</b> Meskipun lebih cepat, dalam segi keakuratannya masih lebih baik menggunakan Algoritma <i>Dijkstra</i></p>	<p><b>Kelebihan:</b> Mengimplementasikannya pada Website mendukung sistem rute rekomendasi ini menjadi lebih interaktif.</p> <p><b>Kekurangan:</b> Dalam segi performa, aplikasi native.</p>

## 1.7 Kesimpulan dan Ringkasan BAB I

*Interactive Route Recommendation System* dapat membantu pengendara kendaraan untuk mendapatkan rute terbaik dari rute asal sampai tujuan pengendara. Pada *Capstone Design* ini akan membuat sebuah sistem yang merekomendasi rute perjalanan dengan interaktif.

Menggunakan metode *Interactive Route* untuk menampilkan rute nya sedangkan metode algoritma A\* untuk rute rekomendasinya. Dengan menganalisis kebutuhan apa saja yang harus dipenuhi untuk membuat sistem yaitu menggunakan API *tomtom* untuk mengimplementasikan data kecepatan *realtime traffic data*, *database Density*, membuat rute jalan menggunakan algoritma A\* dan menambahkan beberapa fitur dan menganalisa algoritma mana yang akan digunakan untuk merancang sistem rute rekomendasi yang interaktif dengan membandingkan dengan beberapa aspek pada algoritma A\* dan algoritma *Dijkstra*.