

Aplikasi Pemandu Kurir Pengiriman Paket untuk Menentukan Tujuan Terdekat dengan Metoda Dijkstra Berbasis Android

Sugeng¹, Gregorius Christierl Bree Kumoro²

Program Studi Sistem Komputer,
Fakultas Teknik dan Ilmu Komputer,
Universitas Komputer Indonesia
¹sugeng@email.unikom.ac.id, ²gregorius@gmail.com

Abstrak

Perkembangan usaha jasa kurir atau pengiriman paket merupakan salah satu bisnis dibidang layanan pengantaran atau pengiriman jasa. Semakin meningkatnya perkembangan usaha maka diperlukan juga jasa pengiriman barang, salah satu faktor ketidakpuasan konsumen disebabkan oleh keterlambatan pengiriman barang. Jasa pengiriman barang ini adalah jasa yang dapat memenuhi kebutuhan manusia akan sebuah efisiensi. Jasa pengiriman paket juga sangat menguntungkan masyarakat karena tingkat efektif yang tinggi dimana penerima hanya menunggu di rumah tanpa harus datang ke toko. Namun demikian dalam prosesnya sering kali terjadinya masalah yang kerap ditemui pada saat kurir hendak mengirimkan paket menuju konsumen. Masalah yang sering muncul adalah sulitnya kurir untuk dapat mengetahui rute terdekat menuju alamat penerima paket. Pada penelitian ini, penulis mengangkat tema tentang aplikasi pemandu kurir pengiriman paket untuk menentukan rute perjalanan yang terdekat menggunakan metoda dijkstra. Tujuan pembuatan aplikasi ini yaitu untuk dapat memberikan solusi terhadap masalah kurir dalam menentukan jalur tercepat mencapai lokasi penerima paket dari lokasi kurir saat ini. Aplikasi yang dikembangkan dibuat menggunakan bahasa program java dalam bentuk aplikasi android. Metoda yang digunakan untuk mencari rute tercepat dari posisi kurir menuju lokasi penerima paket adalah menggunakan metoda dijkstra. Metoda dijkstra diimplementasikan dalam aplikasi dengan bantuan map, dimana dalam map tersebut dibuat node-node yang menjadi titik acuan perhitungan metoda dijkstra. Aplikasi ini dicoba juga dalam mencari rute terdekat penerima paket yang ada di dalam jalan sempit atau area jalur gang perkampungan. Berdasarkan pengujian yang dilakukan maka didapatkan hasil pengujian yang tepat dalam mencari rute terpendek dengan tepat sesuai dengan algoritma yang dibuat.

Kata kunci: Pencarian rute terdekat, Algoritma Dijkstra, Google maps API

Abstract

The development of a courier service or package delivery business is one of the businesses in the field of delivery services or delivery services. With increasing business development, goods delivery services are also needed, one of the factors of consumer dissatisfaction is caused by delays in the delivery of goods. This freight forwarding service is a service that can fulfill human needs for efficiency. Package delivery services also greatly benefit the community because of the high effective rate where the recipient just waits at home without having to come to the store. However, in the process, problems often occur when couriers

want to send packages to consumers. The problem that often arises is the difficulty for couriers to find out the closest route to the package recipient's address. In this research, the author raised the theme of the package delivery courier guide application to determine the closest travel route using the Dijkstra method. The purpose of making this application is to be able to provide a solution to the courier's problem in determining the fastest path to reach the package recipient's location from the current courier's location. The developed application is made using the Java programming language in the form of an Android application. The method used to find the fastest route from the courier's position to the package recipient's location is the Dijkstra method. The Dijkstra method is implemented in an application with the help of a map, where in the map nodes are created which become the reference point for calculating the Dijkstra method. This application was also tried to find the closest route to the recipient of the package in a narrow road or village alley area. Based on the tests carried out, the right test results are obtained in finding the shortest route exactly according to the algorithm that was made.

Keywords: Nearest route search, Dijkstra Algorithm, Google maps API

1. Pendahuluan

Pencarian jalur terpendek adalah sebuah permasalahan yang sangat sering diperbincangkan untuk menentukan lamanya jarak tempuh. Tingginya minat masyarakat untuk melakukan berbelanja online sudah pasti meningkatkan pula jasa pengiriman barang. Bagi pengelola suatu jasa pengiriman barang sudah pasti akan saling berlomba-lomba untuk dapat mempersempit layanan yang paling terbaik [9]. Beberapa permasalahan yang dihadapi kurir baru di area pemukiman padat penduduk adalah kendala untuk mencari lokasi penerima paket. Kurir tidak memiliki pengetahuan tentang rute daerah yang dilewati, kadang kala kurir tidak melengkapi diri dengan bantuan teknologi maps, pada akhirnya kurir akan tersesat dan pengiriman menjadi terlambat.

Dalam proses pengiriman paket terdapat kendala dalam mencari rute tercepat, salah satunya adalah keadaan jalan yang mungkin kecil seperti gang di dalam perkampungan, jalan yang terlalu macet serta kondisi lain yang tidak terduga. Terbatasnya sebuah informasi tentu akan menjadi kendala bagi para kurir yang sedang dalam proses perjalanan mengirimkan paket. Pengetahuan tentang kondisi dan rute jalan sangat diperlukan, khususnya bagi kurir baru, dimana kurir belum begitu banyak mengetahui informasi lokasi tujuan pengiriman melalui rute terdekat. Dengan minimnya pengalaman yang dimiliki oleh kurir terkait rute daerah tersebut maka mungkin sekali terjadi kesalahan dalam menentukan rute menuju lokasi. Beberapa kesalahan lain yang mungkin terjadi adalah kurir harus kembali kelokasi yang sebelumnya pernah dilewati. Kondisi ini sering terjadi dimana seharusnya paket dapat terlebih dahulu dikirimkan sesuai posisinya saat ini. Karena kurir tidak memiliki pengetahuan tentang posisi terdekat terhadap posisi penerima selanjutnya maka bisa jadi kurir harus menghabiskan banyak waktu untuk mencari dan kembali lagi keposisi awal. Padahal bisa jadi posisi penerima selanjutnya dari pemilik paket ada di area sekitaran kurir.[4][5]

Dengan adanya permasalahan tersebut maka penulis mencoba melakukan penelitian dengan merancang sebuah aplikasi yang dapat memandu kurir dalam menentukan rute tercepat menuju lokasi penerima paket. Serta dalam aplikasi tersebut diberikan informasi terkait titik-titik mana saja yang berada pada jangkauan kurir. Pembuatan aplikasi penentuan rute terdekat ini dikembangkan menggunakan metoda dijkstra dan *location based services*. Penelitian terkait yang pernah dikerjakan oleh Taufik Rahman [10]

menjelaskan bahwa penggabungan cara kerja *location based services* dan dijkstra mampu dipergunakan untuk proses menganalisa dan merancang suatu aplikasi yang terkait dengan penentuan rute dalam aplikasi android..

2. Tinjauan Pustaka

Terdapat beberapa tinjauan pustaka yang akan dijabarkan antara lain adalah penelitian terkait yang pernah diteliti sebelumnya, kemudian penjelasan singkat tentang Google Maps API, Sistem operasi android, Sistem informasi geografis, dan LBS, untuk lengkapnya bisa dilihat penjelasan di bawah ini:

2.1. Penelitian Sebelumnya

Penjelasan dari apa yang telah diteliti oleh S.C. Nugroho, O.D Nurhayati dan E.D. Widiyanto menjelaskan bahwa terkait penelitian menggunakan LBS menghasilkan sebuah pencarian rute dalam menentukan posisi perguruan tinggi berbasis LBS dapat memudahkan pengguna dalam mendapatkan informasi. Selain itu juga dapat menentukan rute dengan baik karena terdapat data informasi dan letak lokasi-lokasi pada aplikasi yang dibangun[8]

Sementara pada penelitian yang dilakukan oleh A..Cantona, dkk, menjelaskan bahwa implementasi algoritma dijkstra untuk mencari rute terpendek menuju museum Jakarta dapat digunakan untuk mencari rute terpendek sehingga dapat mempersingkat waktu serta hasilnya efektif sebagai penunjuk arah menggunakan kendaraan roda empat atau mobil.[2]

Pada penelitian yang dilakukan oleh M.N Parapat, dkk. dijelaskan bahwa algoritma dijkstra yang diterapkan pada aplikasi android untuk pengiriman barang dapat mempermudah pegawai pos menuju lokasi sesuai jarak terdekat. [12]

Implementasi algoritma dijkstra juga diterapkan oleh R. Sholikhin dalam pencarian lintasan terpendek lokasi rumah sakit, hotel serta lokasi terminal di Malang berbasis WEB. Dari penelitian tersebut dihasilkan sebuah algoritma dijkstra yang dapat disimpan dan dimuat dalam sebuah situs web untuk mencari lokasi hotel, rumah sakit serta terminal di kota Malang.[11]

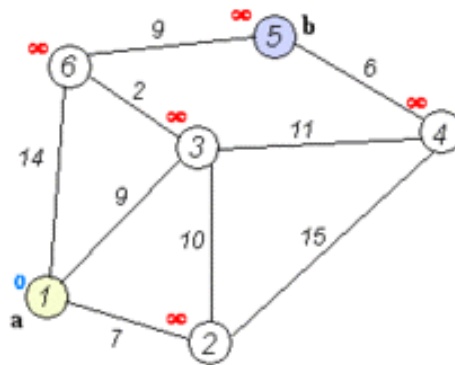
2.2. Google Map API

Google Maps merupakan fasilitas peta gratis yang disediakan oleh perusahaan Google yang begitu populer. Google maps merupakan peta digital yang dapat dengan bebas digunakan dalam mencari suatu daerah dengan lengkap, data peta yang diambil menggunakan satelit. Selain itu pengguna juga dapat menggunakan fitur dengan cara menambahkan Google maps di dalam website yang telah pengguna buat atau mungkin pengguna juga bisa menambahkan pada blog pribadi seperti wordpress. Penggunaan Google maps API ini ada yang berbayar ataupun gratis.

Sebagai contoh penggunaan API Google maps adalah memberi *marker* pada peta android, penentuan jarak antara suatu titik menggunakan *direction* API dan lain-lain. [6]

2.3. Algoritma Dijkstra

Algoritma dijkstra adalah sebuah algoritma yang dipakai dalam mencari rute. Algoritma dijkstra juga dipakai untuk menuntaskan mencari lintasan terpendek [1][2]. Ilustrasi pencarian rute terpendek terlihat pada Gambar 1



Gambar. 1. Ilustrasi Node Pada Metoda Dijkstra

Persoalan tentang rute yang terpendek dari suatu tempat ke tempat lain adalah sebuah masalah yang sangat banyak dialami dalam melakukan percobaan apakah sebuah algoritma yang digunakan berhasil atau tidak. Dengan mengetahui rute terpendek dari sebuah lintasan tentu akan mempercepat waktu tempuh. Dalam metoda dijkstra cara yang digunakan adalah dengan menjumlahkan seluruh *edge* yang akan dilalui, namun demikian metoda dijkstra juga mempunyai banyak pilihan solusi untuk menentukan rute terpendek. Berikut adalah tahapan alur logika dari algoritma dijkstra:

1. Berikan nilai dari suatu jarak pada satu titik ke titik lainnya, setelah itu berikan nilai kosong (0) pada node awal dan berikan nilai tak hingga di node lain yang masih belum dilewati.
2. Langkah selanjutnya adalah memberi tanda "belum dilewati" dan berikan node awal sebagai node awal.
3. Selanjutnya adalah menghitung dari node awal menuju titik akhir melalui node mana saja yang dilewati. Sebagai contoh misalnya titik berangkat dari E-F memiliki bobot jarak 2 dan dari F-G memiliki bobot jarak 4, maka dari titik E ke titik G dengan melewati titik F didapatkan hasil $2+4 = 6$. Jika data jarak yang barusan dihitung lebih kecil daripada jarak sebelumnya maka kita harus menghapus data lama dan kita dapat menyimpan dengan data jarak yang lebih terbaru yang telah dihitung.
4. Langkah selanjutnya adalah set node belum dilewati dari node awal tadi.

2.4. Sistem Informasi Geografis

Sistem Informasi Geografis (SIG) adalah sebuah sarana informasi yang terdapat di dalam *software* yang berbasis komputer dan dibuat untuk menyimpan, mengumpulkan dan melakukan olah data. Sistem informasi geografis juga dapat digunakan untuk menganalisa data yang akan dicari dan mendapatkan suatu tampilan data dari sebagian objek yang berhubungan dengan tata letak gambaran di bumi. Suatu elemen SIG antara lain adalah perangkat lunak serta perangkat keras, dan data pengguna aplikasi [3].

2.5. LBS (Location Based Services)

Location Based Services (LBS) adalah layanan berbasis lokasi yang dipergunakan untuk menemukan lokasi perangkat yang sedang digunakan [7]. LBS juga merupakan implementasi dari GIS Mobile yang berfungsi untuk menampilkan objek seperti navigasi,

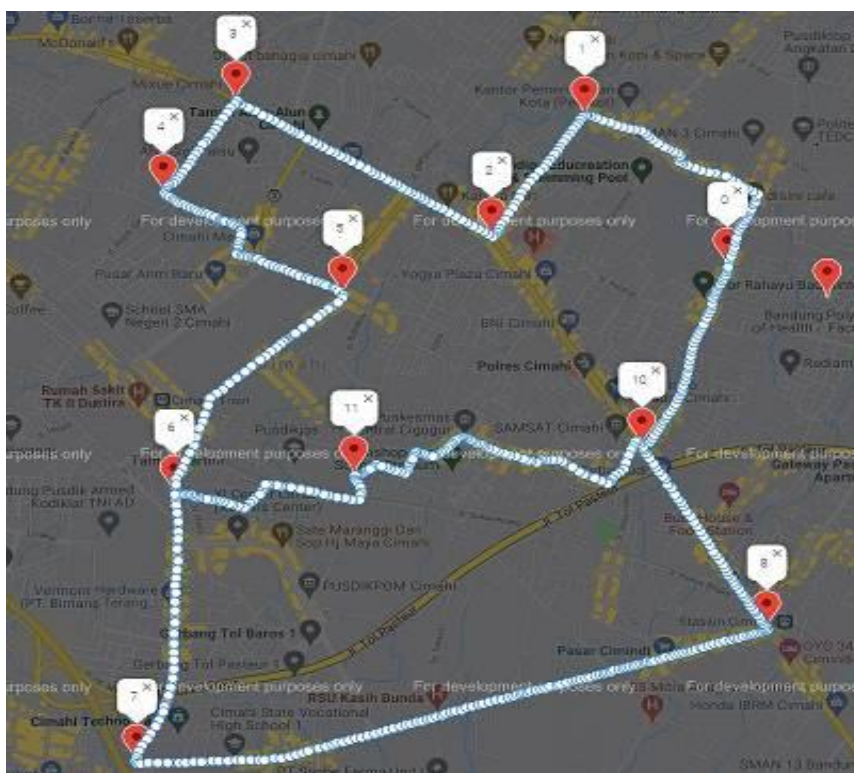
kota, pencarian suatu alamat. Layanan informasi Location Based Services ini biasanya dapat memberi interaksi dua arah antar pengguna [8].

3. Metoda Penelitian

Beberapa metoda untuk dapat menyelesaikan penelitian ini diantaranya adalah.

3.1 Pembuatan Graf

Untuk mengimplementasikan algoritma pada aplikasi android, hal pertama yang dilakukan adalah membuat graf. Proses pembuatan graf ini dilakukan untuk membuat jalur yang akan digunakan untuk menghitung jarak terdekat tersebut dengan menetapkan node dan lokasi yang telah di tetapkan. Gambar graf yang telah dibuat untuk diimplementasikan ke dalam aplikasi bisa dilihat pada Gambar 2



Gambar 2 Metode Penentuan Graf

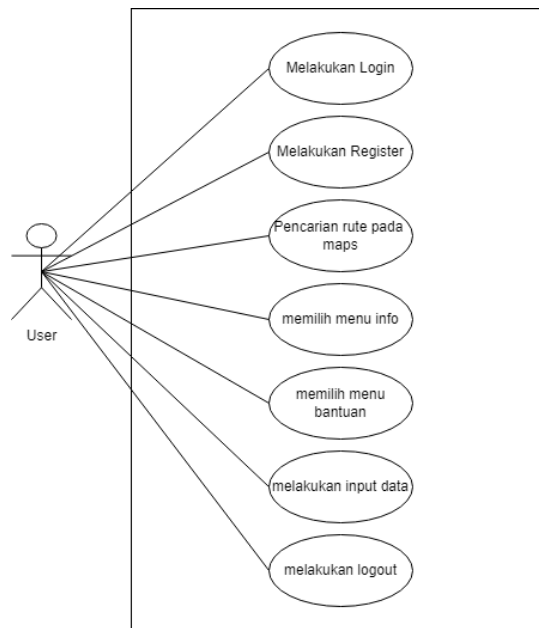
Setelah membuat graf maka langkah selanjutnya adalah membangkitkan jalur yang berbentuk JSON. Lalu setelah mendapatkan data jalur JSON tadi kita bisa langsung memasukan data tersebut ke dalam database SQLite. Setelah itu database yang telah dibuat tadi siap digunakan untuk perhitungan pada aplikasi pencarian rute. Kemudian Pada Tabel 1 dijelaskan bahwa jarak antar nodes yang saling bersambungan adalah berbeda-beda.

Tabel 1. Jarak Antar Node

No	Nodes	Jarak
1.	0-1	1312.2 m
2.	1-2	759.1 m
3.	2-3	1343 m
4.	3-4	585.2 m
5.	4-5	1053.5 m
6.	5-6	1330.7 m
7.	6-7	1445.8 m
8.	7-8	2906 m
9.	8-10	1105.2 m
10.	10-0	711 m
11.	10-11	1486.9 m
12.	11-6	1195 m

3.2 Use Case Diagram

Langkah yang dilakukan dalam analisis ini untuk membangun sistem berbasis objek yang terdapat pada Gambar 3.



Gambar. 3. Rancangan Tampilan Menu dan Maps

Selanjutnya, pada tahap ini akan dijelaskan beberapa *usecase* mulai dari *login*, memilih lokasi, mendapatkan rute, menggunakan menu *logout* dan *input data*

3.3 Perancangan Antarmuka

1. Menu Login dan Register

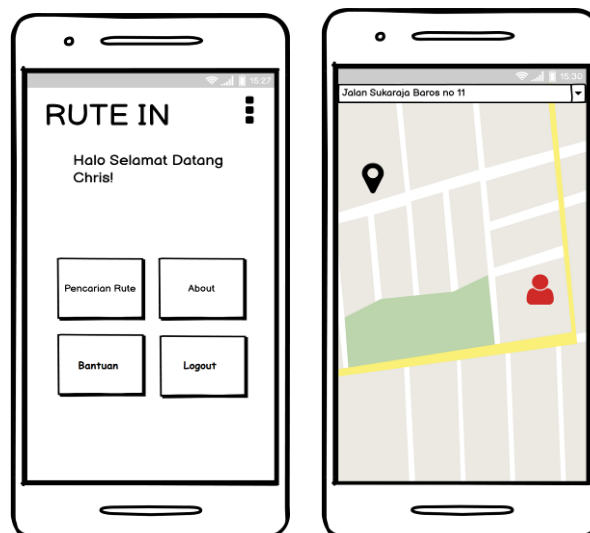
Pada Gambar 4 berikut adalah rancangan aplikasi yang digunakan *user* untuk dapat melakukan *Login* dan *Register* ke dalam aplikasi

2. Rancangan Tampilan Menu dan Maps

Pada Gambar 5 dapat dilihat bahwa ini adalah tampilan *login* dan *register* yang akan didesain.



Gambar 4 Rancangan Tampilan Login dan Register



Gambar 5 Rancangan Tampilan Menu dan Maps

4. Hasil dan Pembahasan

4.1 Pengujian BlackBox

Pengujian *blackbox* adalah metode pengujian pada piranti lunak yang hanya fokus untuk memonitor pada segi fungsi *input*, dan juga *output* pada suatu aplikasi.

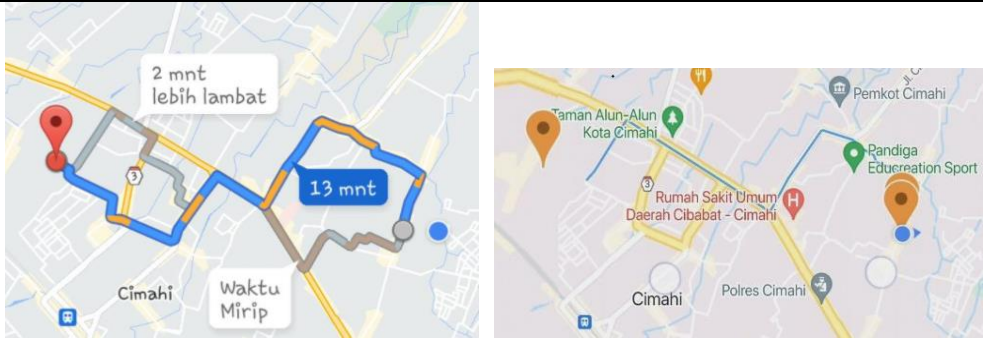
Tabel 2. Skenario Uji *Blackbox*

No	Skenario Uji	Harapan	Hasil
1	Klik icon apliasi	Menampilkan Menu Login	Berhasil, sistem menampilkan menu utama
2	menu login	Tampil form <i>login</i>	Berhasil, sistem menampilkan menu login
3.	Menu Register	Tampil form Register	Berhasil, sistem menampilkan menu registrasi
4.	Menu Maps	Menampilkan Maps	Berhasil, sistem menampilkan menu maps
5.	Menu Info	Tampil form Info	Berhasil, sistem menampilkan menu info
6.	Bantuan Pengguna	Tampil Form Bantuan	Berhasil, sistem menampilkan menu pengguna
7.	<i>Input data</i>	Tampil menu <i>Input data</i>	Berhasil, sistem menampilkan menu <i>input data</i> dan jika paket telah selesai diantar maka user dapat menghapus data tersebut
8.	<i>Logout</i>	Keluar dari Aplikasi	Berhasil, <i>user</i> dapat keluar dari aplikasi

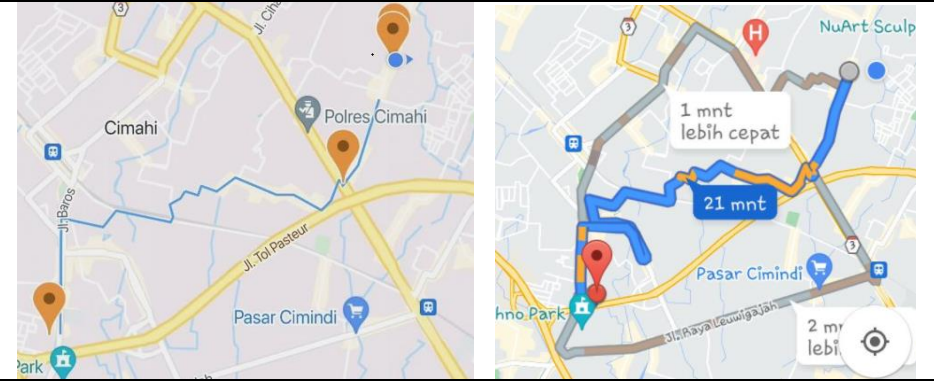
4.2 Pengujian Rute

Berikut ini adalah pengujian untuk mendapatkan rute yang dijalankan pada aplikasi Rute In dan Google Maps. Pada pengujian 1 didapatkan hasil *polyline* seperti pada Tabel 3, Tabel 4 dan Tabel 5. Dimana sistem diuji untuk dapat mencari rute tercepat dariposisi koordinat tertentu menuju koordinat yang ditentukan.

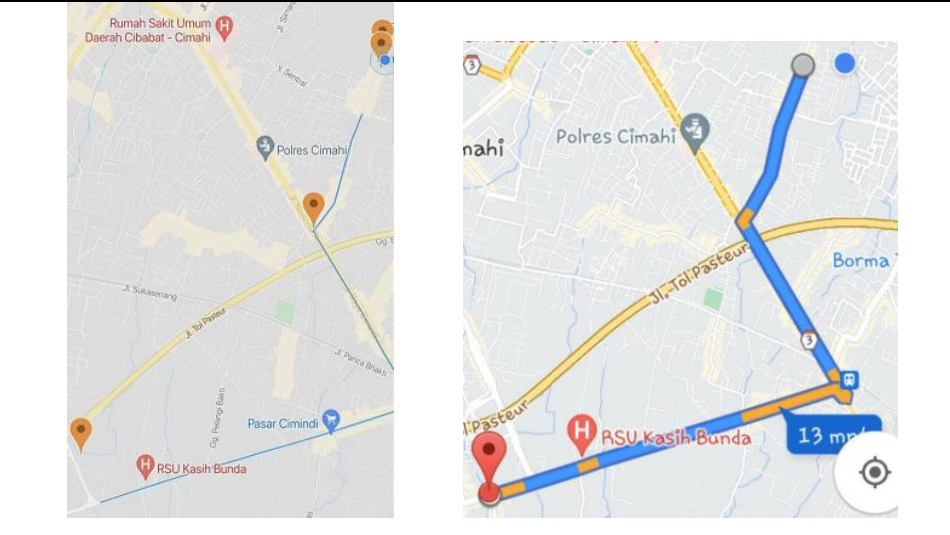
Tabel 3. Pengujian 1 dari Titik Nol ke Titik Empat

Pengujian 1	
Koordinat Awal: -6.879503, 107.558887	
Koordinat Tujuan: -6.8742588713575055, 107.5351982240194	
	

Tabel 4. Pengujian 2 dari Titik Nol ke Titik Tujuh

Pengujian 2
Koordinat Awal: -6.879503, 107.558887
Koordinat Tujuan: -6.897293035365197, 107.53799945470969


Tabel 5. Pengujian 3 dari Titik Nol ke Titik Delapan

Pengujian 3
Koordinat Awal: -6.879503, 107.558887
Koordinat Tujuan: -6.9001987390100075, 107.5439183377541


4.3. Analisis Algoritma Dijkstra

Pada Tabel 6 (Analisa 1) dengan keberangkatan dari koordinat: -6.879503, 107.558887 sampai dengan lokasi tujuan: -6.8742588713575055, 107.5351982240194 didapatkan hasil perhitungan relasi vertex yang paling dekat yaitu Relasi Vertex 1 dengan jarak tempuh 3.999,5 m dan hasil yang didapat sesuai dengan perhitungan algoritma. Sedangkan untuk perhitungan jarak oleh Google maps didapatkan hasil perhitungan 4.300 m.

Tabel 6. Analisa Dijkstra Pengujian 1

No	RV	Analisa Pengujian 1
1	1	$0-1-2-3-4 = (1312,2 + 759,1 + 1343,0 + 585,2) = 3.999,5 \text{ m}$
	2	$0-10-11-6-5-4 = (711 + 1486,9 + 1038,7 + 1330,7 + 1053,5) = 5.620,8 \text{ m}$
	3	$0-10-8-7-6-5-4 = (1025,5 + 1105,2 + 2906,0 + 1445,8 + 1330,7 + 1053,5) = 7.813,2$
		RELASI VERTEX 1

Pada Tabel 7 (Analisa 2) dengan keberangkatan dari koordinat: -6.879503, 107.558887 sampai dengan lokasi tujuan: -6.897293035365197, 107.53799945470969 didapatkan hasil perhitungan relasi vertex yang paling dekat yaitu Relasi Vertex 2 dengan jarak tempuh yaitu 4.838,7 m dan hasil yang didapat tidak sesuai dengan perhitungan algoritma. Sedangkan untuk perhitungan jarak oleh Google maps didapatkan hasil perhitungan 6.300 m.

Tabel 7. Analisa Dijkstra Pengujian 2

No	RV	Analisa Pengujian 2
2	1	$0-1-2-3-4-5-6-7 = (1312,2 + 759,1 + 1343,0 + 585,2 + 1053,5 + 1330,7 + 1445,8) = 7.829,5$
	2	$0-10-11-6-7 = (711 + 1486,9 + 1195,0 + 1445,8) = 4.838,7$
	3	$0-10-8-7 = (1025,5 + 1105,2 + 2906,0) = 5.036,7$
		RELASI VERTEX 2

Pada Tabel 8 (Analisa 3) dengan keberangkatan dari koordinat: -6.879503, 107.558887 sampai dengan lokasi tujuan: -6.9001987390100075, 107.5439183377541 didapatkan hasil perhitungan relasi vertex yang paling dekat yaitu Relasi Vertex 3 dengan jarak tempuh yaitu 1.816,2 m dan hasil yang didapat sesuai dengan perhitungan algoritma. Sedangkan untuk perhitungan jarak oleh Google maps didapatkan hasil perhitungan 2.200 m.

Tabel 8. Analisa Dijkstra Pengujian 3

No	RV	Analisa Pengujian 3
3	1	$0-1-2-3-4-5-6-7-8 = (1312,2 + 759,1 + 1343,0 + 585,2 + 1053,5 + 1330,7 + 1445,8 + 2906,0) = 10.735,5 \text{ m}$
	2	$0-10-11-6-7-8 = (711 + 1486,9 + 1195,0 + 1445,8 + 2906,0) = 7.774,7 \text{ m}$
	3	$0-10-8 = (711 + 1105,2) = 1.816,2$
		RELASI VERTEX 3

5. Kesimpulan

Berdasarkan hasil pengujian perangkat lunak pada aplikasi pemandu kurir pengiriman paket untuk menentukan tujuan terdekat berbasis android telah diperoleh kesimpulan yaitu jalur yang dipilih oleh algoritma dijkstra merupakan jalur terpendek sesuai dengan jalur sesungguhnya pada Google map. dengan adanya aplikasi ini diharapkan dapat membantu pengguna dalam mengetahui lokasi pengiriman terdekat. Seluruh hasil pengujian berdasarkan pengujian satu sampai dengan pengujian tiga dan analisisnya menunjukkan hasil yang tepat dengan berhasil seluruhnya mencari rute terpendek. Kemudian perbandingan aplikasi *Rute In* dengan Google maps tidak sepenuhnya sama.

Daftar Pustaka

- [1] Makiolor, A. A. (2017). Rancang Bangun Pencarian Rumah Sakit, Puskesmas Dan Dokter Praktek Terdekat Di Wilayah Manado Berbasis Android. *Jurnal Teknik Informatika*, 10(1).
- [2] Cantona, A., Fauziah, F., & Winarsih, W. (2020). Implementasi Algoritma Dijkstra Pada Pencarian Rute Terpendek Ke Museum Di Jakarta. *Jurnal Teknologi Dan Manajemen Informatika*, 6(1), 27-34.
- [3] Rahmah, S., Sari, A. P., & Nasution, R. (2018). Aplikasi SIG Mobile Finder Kampus BSI-Nusa Mandiri Jakarta Berbasis Android. *SNIT 2018*, 1(1), 174-179.
- [4] Pambudi, A. (2013). Implementasi Model Perangkat Lunak Pelayanan Informasi Kegiatan Belajar Mengajar Tingkat SLTA Dengan Berbasis Operating System Android. *Jurnal Ilmu Komputer*, 9(2), 108-120.
- [5] Palabiran, M., Cahyadi, D., & Arifin, Z. (2016). Sistem Informasi Geografis Kuliner, Seni Dan Budaya Kota Balikpapan Berbasis Android. *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, 10(1), 54-57.
- [6] Asmar, A. A., Setyaningsih, S., & Karlitasari, L. (2016). Penerapan Direction Api Untuk Mencari Rute Fastfood Di Kota Bogor Berbasis Android. *Jurnal Online Mahasiswa (JOM) Bidang Ilmu Komputer/Informatika*, 1(1).
- [7] S. Br Ginting, Y. Ginting, And T. Prabudi, "Penerapan Algoritma Best-Path Planning Untuk Aplikasi Pencarian Rute Transportasi Publik Berbasis Android", *Jati*, Vol. 5, No. 2, Pp. 49-62, Sep. 2015.
- [8] Nugroho, S. C., Nurhayati, O. D., & Widiyanto, E. D. (2015). Aplikasi Pencarian Rute Perguruan Tinggi Berbasis Android Menggunakan Location Based Service (LBS) Di Kota Semarang. *Jurnal Teknologi Dan Sistem Komputer*, 3(2), 311-319.
- [9] Mulia, M. R., Rismawan, T., & Hidayati, R. Aplikasi Pengelola Data Pada Jasa Pengiriman Barang Berbasis Android. *Coding Jurnal Komputer Dan Aplikasi*, 8(1).
- [10] Rahman, T., Nuha, M. U., Kuswanto, H., & Handono, F. W. (2020). Android-Based Go-Course Application With Location Based Services Method. *Journal Of Informatics And Telecommunication Engineering*, 4(1), 23-32.
- [11] Oktoviana, Mohamad, Yasindan Lucky, T., & Sholichin, R. (2017). Implementasi Algoritma Dijkstra Dalam Pencarian Lintasan Terpendek Lokasi Rumah Sakit, Hotel Dan Terminal Kota Malang Berbasis Web. *Jurnal Informatika*, 4(3), 3993-4000.
- [12] M N Parapat, Deddy Kusbianto, C. R. (2017). Rancang Bangun Aplikasi Pencarian Rute Terpendek Jasa Kiriman Barang Berbasis Mobile Dengan Metode Algoritma Dijkstra. *Informatika Polinema*, 1(2), 15-19.

This Page Intentionally Left Blank