



E-Travel Riau Berbasis Mobile Menggunakan Metode Dijkstra

Prina Marni*, Hadi Asnal, Susi Erlinda, Agustin

Teknik Informatika, STMIK AMIK RIAU, Riau, Indonesia

Email: ^{1,*}prinamarnii@email.com, ²hadiasnal@stmik-amik-riau.ac.id, ³susierlinda@stmik-amik-riau.ac.id, ⁴agustin@sar.ac.id

Email Penulis Korespondensi: prinamanii@email.com

Abstrak—Travel merupakan salah satu transportasi yang sering digunakan oleh study tour atau pariwisata di Provinsi Riau. Travel tidak hanya bergerak dalam bidang pemesanan tetapi juga di bidang pengiriman barang. Tetapi yang sering terjadi pada travel yaitu pemesanan yang masih dilakukan secara manual. Proses Pengorderan dilakukan melalui telephon, lalu admin mencatat alamat penumpang yang akan dijemput dan admin langsung mengkonfirmasi ke driver untuk menjemput penumpang. Tujuan dari penelitian ini adalah untuk membantu penumpang dalam pemesanan travel secara online dan driver dapat memantau dan menentukan lokasi calon penumpang. Pada penelitian ini dibuat sebuah aplikasi pemesanan travel online berbasis android menggunakan algoritma Dijkstra. Algoritma dijkstra adalah sebuah algoritma yang dipakai dalam memecahkan permasalahan jarak terpendek (Shortest path problem) untuk sebuah graf berarah (directed graph) dengan bobot-bobot sisi (edge weight) yang bernilai tak negative. Algoritma ini digunakan oleh driver untuk menentukan rute tercepat dalam proses penjemputan calon penumpang. Kelebihan metode Dijkstra ini adalah dapat mencari rute terdekat dari titik awal menuju titik akhir dengan membandingkan nilai terkecil antar titik yang akan dijadikan rute yang nantinya akan dilewati oleh driver travel agar sampai lebih cepat menuju tujuan. Hasil dari penelitian ini adalah sebuah aplikasi travel yang telah diuji coba oleh 1 calon penumpang yang memudahkan pengguna dalam pemesanan travel serta memudahkan driver dalam menentukan rute tercepat dalam penjemputan penumpang.

Kata Kunci: Travel; Aplikasi; Android; Algoritma Dijkstra.

Abstract—Travel is one of the transportation that is often used by study tours or tourism in Riau Province. Travel is not only engaged in ordering but also in the field of delivery of goods. But what often happens to travel is that bookings are still done manually. The ordering process is carried out by telephone, then the admin records the address of the passenger to be picked up and the admin immediately confirms to the driver to pick up the passenger. The purpose of this study is to assist passengers in ordering travel online and drivers can monitor and determine the location of prospective passengers. In this study, an android-based online travel booking application was created using the Dijkstra algorithm. The dijkstra algorithm is an algorithm used to solve the shortest path problem for a directed graph with non-negative edge weights. This algorithm is used by drivers to determine the fastest route in the process of picking up prospective passengers. The advantage of this Dijkstra method is that it can find the closest route from the starting point to the end point by comparing the smallest value between points that will be used as a route that will be passed by the travel driver in order to get to the destination faster. The results of this study are a travel application that makes it easier for users to book travel and makes it easier for drivers to determine the fastest route in picking up passengers.

Keywords: Travel; Application; Android; Dijkstra's Algorithm

1. PENDAHULUAN

Travel merupakan salah satu pilihan yang sering digunakan oleh pengguna untuk berpergian dari desa ke kota. Dan salah satu solusinya adalah travel merupakan salah satu transportasi yang saat ini banyak digunakan. Travel sudah menjadi pilihan pengguna karena biayanya terjangkau dan keamanan sangat baik dan aman hingga ketujuan. Dibandingkan menggunakan transportasi lainnya[1].

Permasalahan yang terjadi pada travel ialah proses pemesanan yang masih dilakukan secara manual dan belum terkomputerisasi. Proses penjemputan penumpang, *driver* dibantu dengan informasi tentang rute terdekatnya sehingga *driver* dapat memilih rute terbaik. Hal tersebut dapat mengoptimalkan proses penjemputan penumpang sehingga waktu menjadi singkat dan biaya yang dibutuhkan dapat dikurangi. Hasil pengelompokan tempat tujuan penumpang berdasarkan rute terdekat dapat diinformasikan setiap *driver* suatu armada. Strategi ini dapat diterapkan oleh *driver* armada dalam upaya menjaga kepuasan pelayanan terhadap penumpang. Salah satu algoritma yang digunakan dalam penentuan jarak terpendek adalah algoritma dijkstra.

Algoritma Dijkstra adalah merupakan salah satu bentuk algoritma greedy. Algoritma ini termasuk algoritma pencarian graf yang digunakan untuk menyelesaikan masalah lintasan terpendek dengan satu sumber pada sebuah graf yang tidak memiliki cost sisi negatif, dan menghasilkan sebuah pohon lintasan terpendek[2]. Algoritma ini lebih efisien dan cepat dalam penentuan rute terpendek[3]. Algoritma Dijkstra ini sendiri ditujukan untuk menemukan jalur terpendek berdasarkan bobot terkecil dari titik keberangkatan ke titik lainnya[4].

Pada penelitian sebelumnya membahas tentang implementasi algoritma dijkstra dalam menentukan rute terpendek untuk pendistribusian barang pada aplikasi android. Tujuan dari penelitian ini adalah dapat memberikan informasi urutan toko menurut jarak terdekat sesuai tujuan yang telah dipilih. Dan untuk hasil pengujian jarak yang telah dilakukan, diketahui bahwa persentase rata-rata selisih jarak antara aplikasi yang dibuat (menggunakan algoritma Dijkstra) dan haversine sebesar 0.648%. Sehingga dapat disimpulkan bahwa aplikasi distribusi barang dapat menghitung jarak dengan cukup baik[5].

Penelitian yang dilakukan [6] tentang Penerapan Algoritma Dijkstra Untuk Menentukan Rute Terbaik Pada Mobile E-Parking Berbasis Sistem Informasi Geografis. Penelitian ini mengenai posisi suatu tempat dan



menemukan rute terbaik untuk menuju sebuah tempat tujuan termasuk hal yang cukup rumit terutama mencari rute dalam menemukan tempat parkir. Hasil penelitian adalah sistem yang dirancang untuk menyediakan informasi parkir berbasis Sistem Informasi Geografis pada platform android, dan menyediakan navigasi rute terbaik dengan menerapkan Algoritma Dijkstra pada fitur Google Maps.

Penelitian yang dilakukan[7] tentang Penerapan Algoritma Dijkstra Untuk Sistem Pendukung Keputusan Bagi Penentuan Jalur Terpendek Pengiriman Paket Barang Pada Travel. Dalam penelitian ini proses pengiriman barang. Pemborosan waktu dan biaya operasional juga merupakan efek utama yang terjadi dari proses pengiriman barang berdasarkan urutan daftar kirim. Hasil dari penelitian adalah alternatif rute yang efektif dengan memperhatikan Jenis Arah, Tingkat Kemacetan, Lebar Jalan, Kondisi Jalan, Volume Jalan, Jarak, Jumlah Simpangan.

Penelitian yang dilakukan[8] tentang Implementasi Algoritma Dijkstra Pada Aplikasi Wisata Kuningan Berbasis Android. Dalam penelitian ini Kabupaten kuningan adalah salah satu tujuan wisata yang cukup populer di kalangan wisatawan. Kuningan memiliki banyak objek wisata yang potensial untuk wisata alam, sejarah. Hasil dari penelitian Aplikasi berbasis mobile diperlukan dalam menemukan lokasi objek wisata dan menyediakan deskripsi wisata, dan fasilitas yang ada di objek wisata.

Penelitian yang dilakukan[9] tentang Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi). Dalam penelitian ini Kebun raya purwodadi memiliki koleksi tanaman sejumlah 2002 jenis/spesies, 178 suku/family, 962 marga/genus dan 11.669 specimen. Dengan jumlah tanaman yang begitu banyak, dibutuhkan aplikasi yang dapat menunjukkan jalan dari lokasi pengguna ke lokasi tanaman yang dituju. Hasil dari penelitian ini implementasi algoritma dijkstra dalam menentukan jarak terpendek dari lokasi pengguna ke tanaman yang di tuju menghasilkan nilai usability sebesar 70.916%, yang diambil dari 30 responden. Nilai tersebut berada di rentang interval 60% - 80% yang dapat dikategorikan sebagai aplikasi yang memuaskan.

Penelitian yang dilakukan[10] tentang Implementasi Algoritma Dijkstra untuk Pencarian Rute Terpendek Menuju Rumah Sakit pada Aplikasi C-Hos (Covid-19 Hospital) Dalam penelitian ini Tepat pada tanggal 2 Maret 2020 corona virus (Covid-19) masuk ke Indonesia. Di Jakarta sudah terdapat 445.302 kasus positif Covid-19 per tanggal 12 Juni 2021. Pemerintah sendiri telah menunjuk wisma atlet sebagai rumah sakit darurat corona dan juga terdapat beberapa rumah sakit di Jakarta untuk menjadi rumah sakit rujukan Covid-19. Akan tetapi banyak masyarakat yang belum mengetahui rumah sakit rujukan covid-19. Hal ini sangat disayangkan, karena jika sewaktu-waktu masyarakat atau warga yang sudah menunjukkan gejala covid-19 bisa langsung datang ke rumah sakit rujukan Covid-19 terdekat. Hasil dari penelitian pengujian aplikasi C-Hos ini dapat membantu masyarakat mengetahui rumah sakit mana saja yang menjadi rumah sakit rujukan covid-19 dan penerapan Algoritma Dijkstra pada aplikasi C-Hos sangat membantu dalam menentukan rute terpendek menuju rumah sakit rujukan Covid-19.

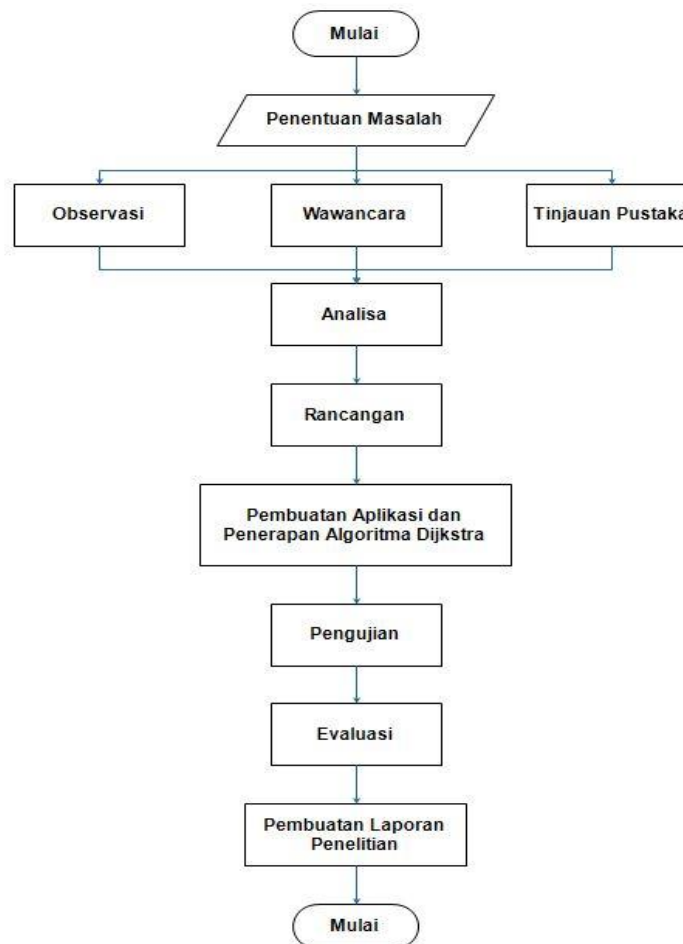
Penelitian yang dilakukan[11] tentang Algoritma Dijkstra untuk Penentuan Jarak Tempuh Terpendek Pengantaran Katering Pabrik. Dalam penelitian ini Algoritma Dijkstra digunakan untuk menemukan jalur terpendek antara titik pada graf dan persamaan Haversine digunakan untuk mengukur jarak dari lokasi awal menuju lima lokasi tujuan yang mana lokasi tersebut merupakan pabrik yang berada di kota Cikarang dan lokasinya ada di sekitaran penyedia rumah catering. Hasil dari penelitian Sistem mampu menampilkan prediksi jarak dan waktu tempuh untuk rekomendasi dari urutan lima pengantaran dengan penerapan metode algoritma Dijkstra dimana proses yang dilakukan sistem adalah memperhitungkan jarak menggunakan Haversine Formula, sehingga didapatkan waktu tempuh berdasarkan parameter kemacetan. selain itu API mampu memvisualisasikan rute setiap tujuan dari titik lokasi katering.

permasalahan diatas peneliti menerapkan algoritma dijkstra dalam menentukan rute terpendek untuk *driver* travel dalam penjemputan penumpang. Tujuan dari penelitian ini adalah membuat sebuah aplikasi yang diharapkan dapat membantu *driver* dalam menentukan rute tercepat dalam penjemputan penumpang. Dengan memanfaatkan fitur GPS yang di sediakan pada smartphone mobile dan menggunakan algoritma Dijkstra sebagai penentuan jalur terpendek.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Pada penelitian digunakan sebagai pedoman dalam pelaksanaan penelitian agar hasil yang dicapai tidak menyimpang dari tujuan yang telah dilakukan. Berikut ini adalah metodologi yang digunakan dalam pengumpulan data dengan melakukan observasi dan wawancara di loket travel kota pekanbaru. Seperti yang terlihat pada gambar 1.



Gambar 1. Tahapan penelitian

Tahapan yang dilakukan dapat dijelaskan sebagai berikut:

1. **Penentuan Masalah**
Penentuan masalah dapat kita ambil dari kekurangan suatu sistem yang telah berjalan serta dipergunakan untuk suatu instansi atau badan usaha.
2. **Pengumpulan Data**
Pengumpulan data yang dilakukan pertama adalah dengan melakukan observasi ke tempat penelitian yaitu salah satu loket travel. Serta melakukan wawancara terhadap staff terkait dalam hal ini adalah staff yang menangani transaksi pemesanan tiket serta *driver* dari mobil travel. Dilanjutkan mencari referensi – referensi sebagai acuan yang berhubungan dengan penelitian yang dilakukan.
3. **Analisa**
Analisa sistem dengan komunitas pemakai sistem (end-user), dimana untuk mendapatkan tanggapan dan kepercayaan pemakai, sehingga mendapat partisipasi yang baik seperti analisa kebutuhan perangkat keras (*hardware*) perangkat lunak (*software*).
4. **Rancangan**
Sistem informasi pada travel berbasis android untuk pengguna dan *driver* sedangkan admin menggunakan dibangun dengan menggunakan bahasa pemrograman web PHP serta database MySQL dan desain sistem dibuat menggunakan UML (Use Case Diagram)[12].
5. **Pembuatan Program Dengan Algoritma Dijkstra**
Tahap ini dilakukan adalah pembuatan program sesuai dengan perancangan yang telah dibuat sebelumnya, Pembuatan Program menggunakan Aplikasi Android Studio dan Bahasa PHP serta penerapan dari algoritma Dijkstra.
6. **Pengujian**
Pada tahapan ini pengujian program dilakukan dengan menggunakan BlacBox Testing dengan harapan bahwa perancangan yang sudah dibuat dapat berjalan dengan sesuai kehendak[13].

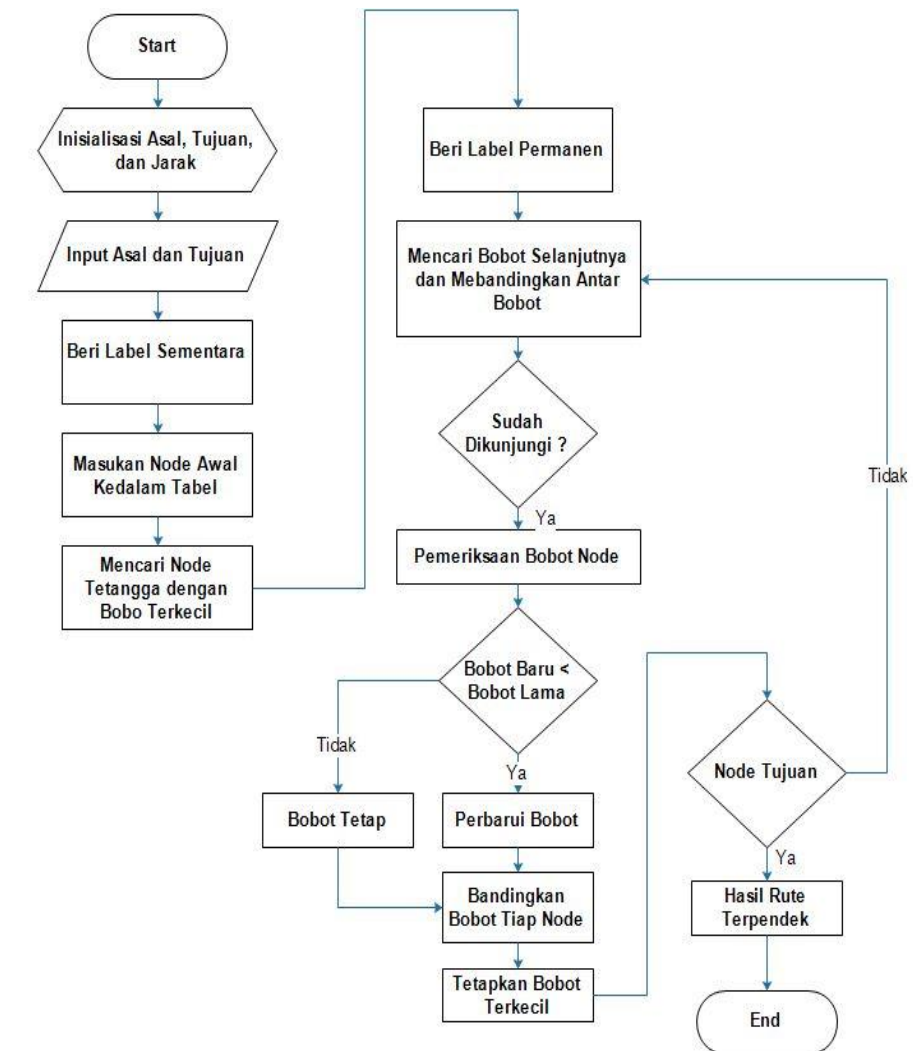
2.2 Algoritma Dijkstra

Algoritma Dijkstra merupakan sebuah algoritma digunakan untuk menentukan rute terpendek pada graph berarah atau graph tak berarah yang memiliki bobot tanpa mengenumerasi secara eksplisit semua rute alternatif yang



mungkin menjadi solusi rute optimal[14]. Algoritma Dijkstra membutuhkan parameter berupa tempat asal dan tempat tujuan. Parameter tersebut berisi rute berbentuk sisi(vertex) atau vertices dalam bentuk jamak untuk diperbandingkan. Setiap sisi dari rute ini adalah pasang vertices(u,v) yang melambangkan hubungan dari vertex u ke vertex v. Himpunan semua tepi kita sebut sebagai E. Bobot (weights) dari semua sisi dihitung dengan Rumus : $w:E \rightarrow 0, \infty$?

Jadi, $w(u,v)$ adalah jarak tak-negatif dari vertex u ke vertex v. Ongkos (cost) dari sebuah sisi dapat dianggap sebagai jarak antara dua vertices, yaitu jumlah jarak semua sisi dalam rute tersebut. Untuk sepasang vertex s dan t dalam V, algoritma ini menghitung jarak terpendek dari s ke t . Adapun flowchart proses dari algoritma dijkstra pada pembuatan aplikasi travel terlihat pada gambar 2.



Gambar 2. Flowchart Algoritma Dijkstra

Penjelasan untuk flowchart algoritma Dijkstra yang ditunjukkan oleh gambar 3 diatas adalah sebagai berikut:

1. Inisialisasi vertex.
2. Inisialisasi jarak antar vertex
3. Tentukan vertex awal (s) dan vertex tujuan (t).
4. Beri label permanen = 0 ke verteks awal (s) dan label sementara = ∞ ke verteks lainnya.
5. Untuk setiap verteks V yang belum mendapat label permanen, mendapat label sementara = $\min \{ \text{label lama } V, (\text{label lama } V + D) \}$.
6. Cari nilai minimum diantara semua verteks yang masih berlabel sementara.
7. Jadikan verteks minimum yang berlabel sementara menjadi verteks dengan label permanen, jika lebih dari satu verteks pilih sembarang.
8. Ulangi langkah 5 sampai 7 hingga verteks tujuan mendapat label permanen.
9. Simpan hasil perhitungan. Tampilkan hasil perhitungan.

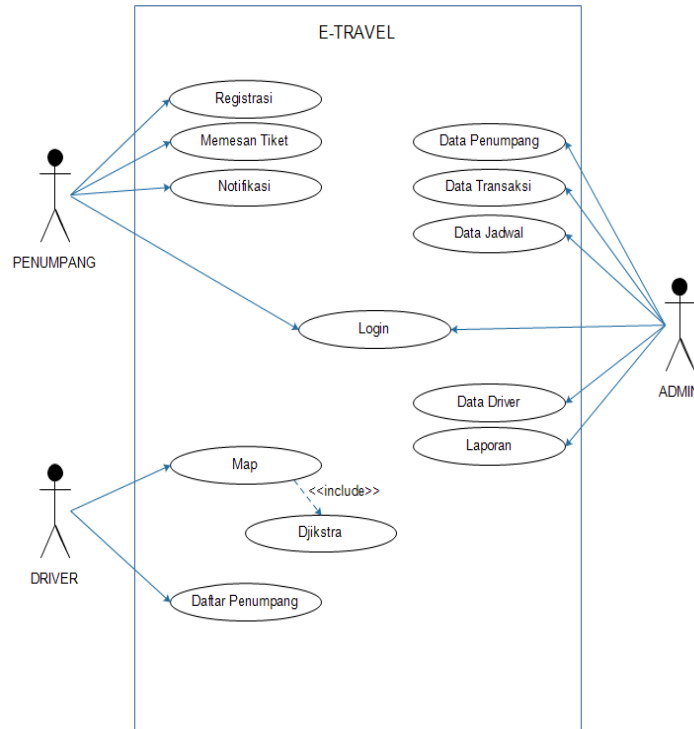
Adapun untuk implementasi dari algoritma dijkstra bekerja dengan membuat jalur ke satu simpul optimal pada setiap langkah. Jadi pada langkah ke n, setidaknya ada n node yang sudah kita tahu jalur terpendek.



3. HASIL DAN PEMBAHASAN

3.1 Rancangan Use Case Diagram

Use case diagram ini hanya digunakan oleh penumpang, admin dan Driver.



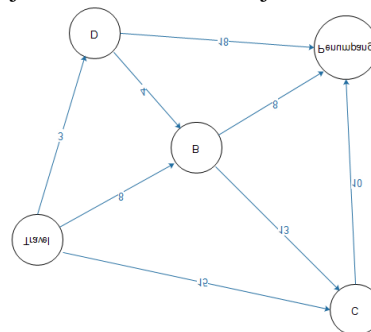
Gambar 3. Use Case Diagram

Penjelasan dari Use Case Diagram diatas adalah :

1. Admin melakukan login pada sistem, tugas-tugas yang dilakukan oleh admin setelah login ialah melakukan olah data penumpang, olah data driver, olah data transaksi, olah data jadwal dan laporan.
2. Driver melakukan login pada sistem, tugas – tugas yang dilakukan oleh admin setelah login ialah melihat data penumpang serta output rute menuju penjemputan penumpang.
3. Penumpang melakukan registrasi dan login pada sistem, tugas – tugas yang dilakukan oleh admin setelah login ialah melakukan transaksi pemesanan tiket dan mendapatkan notifikasi.

3.2 Implementasi Penerapan Algoritma Dijkstra

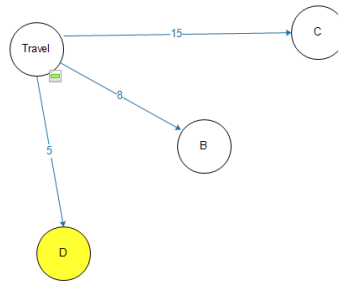
- a. Menerap langkah kerja algoritma dijktra untuk titik awal tujuan.



Gambar 4. Graph Awal

Pada graph diatas diketahui, node awal diberi nama dengan node Travel dan titik akhir node atau tujuannya adalah pada node Penumpang.

- b. Mulai menghitung dengan bobot yang diberikan pada graph, Dimulain dari node awal. Lihat semua node yang terhubung pada node awal yaitu node B, C, D dimana dari bobot tiap node terpilih node D sebagai node awal yang baru karna memiliki bobot terkecil yaitu 5



Gambar 5. Set Node Awal Baru

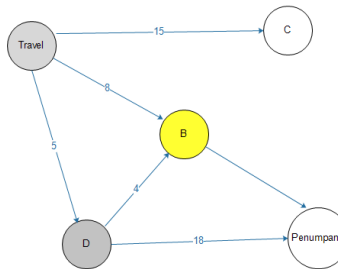
Jalur untuk travel sebagai node awal dapat dilihat pada tabel dibawah ini :

Tabel 1. Menghilangkan nilai bobot dimulai dari node awal

V	A (Travel)	B	C	D	E
A	0	8a	15a	3a	Tak Hingga

Dari tabel diatas dapat dijelaskan, dimulai dari node awal yaitu A yang kemudian dihubungkan ke node yang terhubung ke node A yaitu B, C, dan D dengan nilai bobot yang terlihat pada table diatas

- c. Dari node awal yang baru yaitu D, dilihat lagi semua node yang terhubung ke node D yaitu node Node B, dan E (penumpang). Dengan nilai bobot terkecil adalah node B yaitu 4.



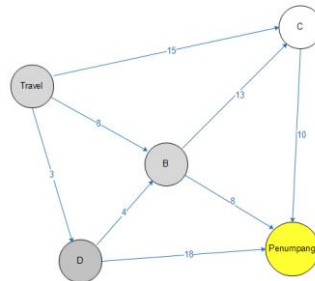
Gambar 6. Set Node Awal Baru Dari Node D

Tabel 2. Memilih nilai bobot terkecil node awal

V	A (Travel)	B	C	D	E
A	0	8a	15a	5a	Tak Hingga
D		4d	0	0	18d

Penjelasan, diketahui pada bobot yang terdapat pada node adalah nilai bobot yang terhubung ke node D yang terkecil yaitu 4. Jadi, untuk node berikut akan dimulai dari node B:

- d. Mengulang Langkah yang sama dengan melihat seluruh node yang terhubung ke node awal yang baru yaitu B adalah Node C, dan E (penumpang). Semua node yang terhubung dipilih node terkecil yaitu 8



Gambar 7. Node sampai Tujuan

Tabel 3. Rute terpendek dari graph node awal

V	A (Travel)	B	C	D	E
A	0	8a	15a	5a	Tak Hingga
D		4d	0	0	18d
B		0	13b	0	8b

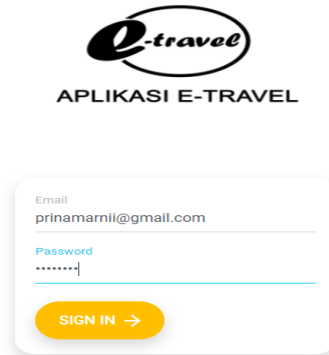
Didapat dari nilai bobo node awal pada node B ke node yang terhubung bahwa node B mendapat bobot tekecil langsung ketujuan yaitu node E (penumpang) dengan nilai bobot 8.



Kesimpulan yang didapat pada pada graph diatas didapat rute terpendek menuju tempat tujuan adalah Rute : A – D – B – E

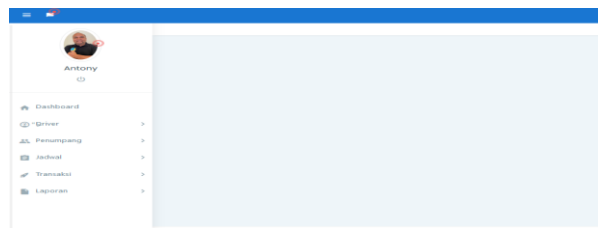
3.3 Tampilan Aplikasi

Gunakan Halaman admin merupakan halaman web yang memungkinkan seseorang untuk dapat mengelola data driver, jadwal, transaksi dan laporan. Berikut ini akan dijelaskan halaman admin pada sistem yang dibangun. Tampilan login merupakan bentuk tampilan yang akan ditampilkan diawal ketika pertama membuka sistem dan hal yang harus dilakukan sebelum menggunakan sistem.



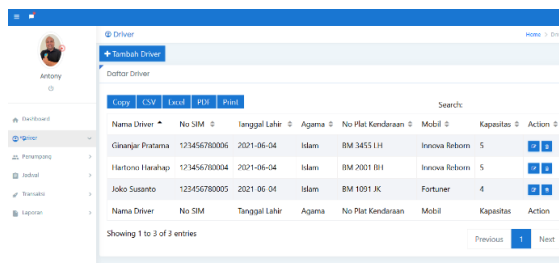
Gambar 8. Login

Halaman utama Merupakan halaman yang ditampilkan setelah pengguna melakukan login kedalam sistem.

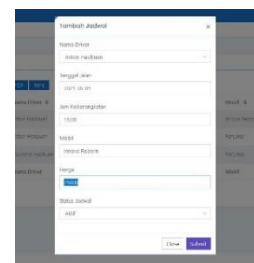


Gambar 9. Halaman Utama

Admin Data Driver Merupakan form yang digunakan untuk melakukan pengelolaan data driver yang dilakukan oleh admin.

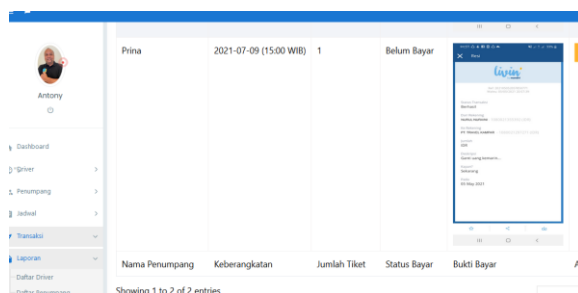


Gambar 10. Daftar jadwal



Gambar 11. Tambah jadwal

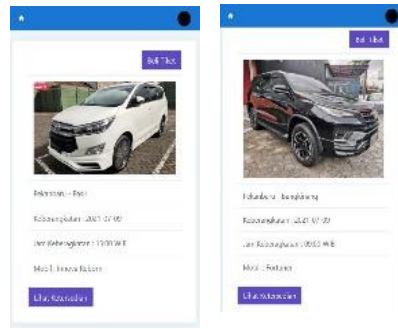
Halaman Admin Data Transaksi Merupakan halaman yang berisikan untuk mengelola data transaksi sebagai dari penumpang yang membeli tiket travel.



Gambar 12. Data Transaksi



Halaman Utama penumpang Merupakan halaman yang menunjukkan menu yang dapat diakses oleh penumpang.



Gambar 13. Halaman utama penumpang

Halaman Penumpang Transaksi Merupakan halaman penumpang untuk melakukan transaksi pembelian tiket travel.



Gambar 14. Halaman transaksi

Halaman Driver Merupakan halaman yang berisikan menu-menu yang dapat diakses oleh driver.



Gambar 15. Daftar list penumpang

Tampilan rute penumpang Merupakan halaman yang menunjukkan hasil perhitungan dari penerapan algoritma dijkstrak.



Gambar 16. Hasil cluster

4. KESIMPULAN

Aplikasi Travel merupakan sebuah aplikasi yang menyediakan sebuah informasi dan pesanan tiket keberangkatan penumpang yang ada dikota riau. Dengan menerapkan algoritma djikstra pada aplikasi ini untuk mencari rute terpendek dengan tujuan lokasi calon penumpang yang ada dikota riau. Pengguna aplikasi ini digunakan dari titik awal lokasi keberangkatan calon penumpang agar dapat digunakan dimanapun sehingga admin dan driver mudah



menentukan proses penjemputan penumpang. Dengan menggunakan algoritma djijkstra menghasilkan data lokasi yang akurat. Pengujian algoritma djijkstra ini berhasil untuk menjemput titik lokasi penumpang yang telah diinput dan diproses menghasilkan data pengujian dengan membandingkan rute Google Maps berbasis Android. Untuk tahap pengujian pencarian rute menggunakan algoritma djijkstra menghasilkan output yang diinginkan.

REFERENCES

- [1] S. Sunardi, A. Yudhana, and A. A. Kadim, "Implementasi Algoritma Dijkstra Untuk Analisis Rute Transportasi Umum Transjogja Berbasis Android," *J. Sist. Inf. Bisnis*, vol. 9, no. 1, p. 32, 2019, doi: 10.21456/vol9iss1pp32-38.
- [2] Chayatul Ichan, "Aplikasi Android Pencarian Mikrolet Kota Surabaya Dengan Menggunakan Algoritma Dijkstra," *Tek. Inform. Fak. Tek.*, vol. 1, no. 1, pp. 6–8, 2018.
- [3] F. P. Juniawan *et al.*, "Shortest Path Determination of the Tourist Destination in Toboali Using Web-Based Dijkstra Algorithm," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, pp. 211–218, 2020, doi: 10.25126/jtiik.202071954.
- [4] A. Cantona, F. Fauziah, and W. Winarsih, "Implementasi Algoritma Dijkstra Pada Pencarian Rute Terpendek ke Museum di Jakarta," *J. Teknol. dan Manaj. Inform.*, vol. 6, no. 1, pp. 27–34, 2020, doi: 10.26905/jtmi.v6i1.3837.
- [5] D. Lesmana, "Algoritma Dijkstra untuk Menentukan Jalur Tercepat pada Pendistribusian Barang Berbasis Mobile Dijkstra 's Algorithm for Determining the Fastest Path for Mobile-Based Goods Distribution," vol. 08, no. 4, pp. 362–368, 2020, doi: 10.26418/justin.v8i4.42250.
- [6] C. Prianto and M. Kusnadi, "Penerapan Algoritma Dijkstra Untuk Menentukan Rute Terbaik Pada Mobile E-Parking Berbasis Sistem Informasi Geografis," *J. Inform. J. Pengemb. IT*, vol. 3, no. 3, pp. 329–335, 2018, doi: 10.30591/jpit.v3i3.941.
- [7] E. Budihartono, "Penerapan Algoritma Dijkstra Untuk Sistem Pendukung Keputusan Bagi Penentuan Jalur Terpendek Pengiriman Paket Barang Pada Travel," *Senit*, pp. 69–78, 2016, [Online]. Available: <https://ejournal.poltektegal.ac.id/index.php/prosiding/article/viewFile/360/344>.
- [8] E. C. Galih and R. A. Krisdiawan, "Implementasi Algoritma Dijkstra Pada Aplikasi Wisata Kuningan Berbasis Android," *Nuansa Inform.*, vol. 12, no. 1, 2018, doi: 10.25134/nuansa.v12i1.1344.
- [9] M. S. Yusuf, H. M. Az-zahra, and D. H. Apriyanti, "Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 12, pp. 1779–1781, 2017.
- [10] M. T. Ismail, S. Andryana, and A. Gunaryati, "Implementasi Algoritma Dijkstra untuk Pencarian Rute Terpendek Menuju Rumah Sakit pada Aplikasi C-Hos (Covid-19 Hospital)," vol. 5, pp. 888–895, 2021, doi: 10.30865/mib.v5i3.3077.
- [11] T. D. PUTRI, W. SUGENG, and E. SAFITRI, "Algoritma Dijkstra untuk Penentuan Jarak Tempuh Terpendek Pengantaran Katering Pabrik," *MIND J.*, vol. 5, no. 2, pp. 108–120, 2021, doi: 10.26760/mindjournal.v5i2.108-120.
- [12] F. Fatmawati and J. Munajat, "Implementasi Model Waterfall Pada Sistem Informasi Persediaan Barang Berbasis Web (Studi Kasus: PT.Pamindo Tiga T)," *J. Media Inform. Budidarma*, vol. 2, no. 2, pp. 1–9, 2018, doi: 10.30865/mib.v2i2.559.
- [13] D. S. Purnia, A. Rifai, and S. Rahmatullah, "Penerapan Metode Waterfall dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android," *Semin. Nas. Sains dan Teknol. 2019*, pp. 1–7, 2019.
- [14] I. P. W. Gautama and K. Hermanto, "Penentuan Rute Terpendek dengan Menggunakan Algoritma Dijkstra pada Jalur Bus Sekolah," *J. Mat.*, vol. 10, no. 2, p. 116, 2020, doi: 10.24843/jmat.2020.v10.i02.p128.
- [15] Y. Heriyanto, "Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car," *J. Intra-Tech*, vol. 2, no. 2, pp. 64–77, 2018.
- [16] A. Firman, H. F. Wowor, X. Najoan, J. Teknik, E. Fakultas, and T. Unsrat, "Sistem Informasi Perpustakaan Online Berbasis Web," *E-Journal Tek. Elektro Dan Komput.*, vol. 5, no. 2, pp. 29–36, 2016.
- [17] Y. Effendi, "Rancangan Aplikasi Game Edukasi Berbasis Mobile Menggunakan App Inventor," *J. Intra-Tech*, vol. 2, no. 1, pp. 39–48, 2018.