

DECISION SUPPORT SYSTEM UNTUK PENENTUAN RUTE TERPENDEK BUS DAMRI KOTA SURABAYA MENGGUNAKAN ALGORITMA DIJKSTRA

Christian Willy Horizont¹, Ahmad Habib^{1,*}, Geri Kusnanto¹, Ery Sadewa Yudha Wrahatnala¹

¹Teknik Informatika, Fakultas Teknik, Universitas 17 Agustus 1945 Surabaya
E-mail: habib@untag-sby.ac.id*

ABSTRACT

The growth of information technology especially internet and web-based application can make information access of Damri bus route in Surabaya easier that will be developed with PHP, HTML, Javascript, and MySQL there're also tools needed are Laragon, VSCode, Web-Browser, and Navicat. The method used in this application development is Dijkstra algorithm, These algorithm used for calculate closest route from one bus-stop to another. These web-based application provide information of shortest bus routes and also help determining bus which new passengers should take to get to some place.

Keywords: Decision Support System, bus damri, dijkstra, closest route

ABSTRAK

Perkembangan di bidang teknologi informasi khususnya teknologi internet dan aplikasi berbasis web dapat mempermudah dalam mengakses informasi rute Damri di kota Surabaya yang akan dibuat menggunakan PHP, HTML, Javascript dan MySQL adapun alat yang dibutuhkan ialah Laragon, VSCode, Web-Browser, dan Navicat. Metodologi yang digunakan dalam pembuatan aplikasi ini adalah algoritma Dijkstra, Algoritma ini dipakai untuk menghitung jarak terdekat dari suatu halte ke halte lainnya. Aplikasi berbasis web ini akan memberikan informasi berupa rute bus terpendek serta akan membantu menentukan bus yang harus dinaiki calon penumpang untuk mencapai ke suatu tujuan.

Kata Kunci: Sistem pendukung keputusan, bus damri, dijkstra, rute terdekat

1. Pendahuluan

Perkembangan di bidang teknologi informasi khususnya teknologi internet dapat mempermudah dan membantu berbagai bidang pekerjaan yang terkait dengan kemudahan akses dan jarak. Makin mudahnya akses internet pada masa sekarang turut mendorong berbagai moda transportasi untuk meningkatkan layanan untuk calon penumpang[1][2].

Bus kota merupakan salah satu moda transportasi umum darat yang telah ada sejak dulu di Indonesia. Namun keberadaannya kini mulai jarang terlihat di jalanan. Khususnya untuk Damri di kota Surabaya penyediaan informasi rute yang digunakan selama ini masih kurang merata dan belum mengikuti perkembangan teknologi seperti

transportasi lain yakni tidak semua halte bus disediakan informasi rute serta tidak adanya informasi rute di web Damri. Hal ini dinilai kurang efektif dan efisien dalam memudahkan calon penumpang untuk dapat memperoleh informasi rute bus[3]. Ditambah dengan pesatnya kemajuan teknologi saat ini melihat transportasi lain yang sudah berbasis online sehingga bus kota kini menjadi kurang diminati dibanding transportasi lain[4][5].

Penduduk kota Surabaya kini terdata mencapai 3.2 juta jiwa, hingga 10 Oktober 2019 1232 jiwa diantaranya adalah pendatang. Penduduk pendatang ini lebih memilih transportasi umum berbasis online karena masih baru atau belum mengenal wilayah-wilayah di kota Surabaya serta

dinilai lebih cepat untuk menuju ke suatu tempat dikota Surabaya[6][7].

Persoalan penentuan rute terpendek dapat diselesaikan dengan berbagai macam algoritma, salah satunya algoritma *Dijkstra*. Algoritma ini menghitung bobot terkecil tiap-tiap titik sehingga tercapai nilai terkecil dari titik awal ke titik tujuan. Pada model *algoritma dijkstra graph* berarah dengan bobot non negatif $G=(V,E)$, masalah *path* terpendek dengan *single source* adalah untuk mencari *path* terpendek dari suatu vertek $v \in V$ ke semua vertek yang lain di dalam V . Suatu *shortest path* dari u ke v adalah *path* dengan bobot minimal. Untuk mencari *shortest path* dari vertek tunggal v ke setiap vertek yang lain, dapat juga untuk digunakan mencari *shortest path* di antara semua pasangan titik. Formalnya, masalah *shortest path* setiap pasang adalah untuk mencari *shortest path* di antara semua pasang vertek $v_i, v_j \in V$ sedemikian sehingga $i \neq j$. *Graph* dengan n vertek, luarannya adalah matriks berukuran $n \times n$ dari $D=d(i,j)$ sedemikian sehingga $d_{i,j}$ adalah biaya *shortest path* dari vertek v_i ke vertek v_j . Garis untuk merepresentasikan bobot lamanya waktu, beban biaya, besaran pinalti, beban kerugian, atau beberapa kuantitas lain yang dihitung secara akumulatif. Pencarian model *single source shortest path* dari suatu vertek tunggal s , dapat dilakukan dengan cara *increment* mencari *shortest path* dari s ke vertek yang lain di G dan selalu memilih suatu *edge* ke suatu vertek tertutup yang terdekat, dengan kompleksitas $\Theta(n_2)$. Algoritma pencarian *all pairs shortest path* dari suatu vertek ke semua vertek yang lain, untuk semua pasangan dengan mengeksekusi algoritma *single source* pada setiap prosesor, untuk semua vertek v . Model algoritma ini memerlukan kompleksitas $\Theta(n_3)$. Untuk segmen program berikut menunjukkan *algoritma dijkstra sekuensial* untuk *shortest paths single source*. Pada prosedur ini untuk setiap vertek $u \in (V-V_T)$, meletakkan $l[u]$, sebagai biaya minimal untuk menjangkau vertek u dari vertek s di mana vertek-vertik berada di V_T [8][9].

```

Procedure DIJKSTRA SINGLE SOURCE SP(V,E,w,s)
Begin
  VT={s};
  For all v ∈(V-VT) do
    If(s,v) exists set l[v]=w(s,v);
    Else set l[v]=∞;
  While VT≠V do
  Begin
    Find a vertex u such that l[v]= min { l[v] | v ∈(V-VT)
    };
    VT=VT∪{u};
    For all v∈(V-VT) do
      l[v] = min { l[v], l[u] + w(u,v) };
  Endwhile

```

Dari permasalahan tersebut memunculkan ide gagasan untuk membuat suatu aplikasi berbasis web, yang di dalamnya dapat memberikan informasi berupa rute bus terpendek serta dapat menentukan bus yang harus dinaiki penumpang untuk mencapai ke suatu tujuan[10], [11]. Metodologi yang digunakan dalam pembuatan aplikasi ini adalah algoritma *Dijkstra*. Algoritma ini dipakai untuk menghitung jarak terdekat dari suatu halte kehalte lainnya[12], [13]. Bahasa pemrogramannya adalah *PHP*, *HTML*, dan *Javascript*. Untuk tampilan antarmukanya menggunakan *CSS3* dan *bootstrap*. *Database*-nya menggunakan *MySQL*. Code Editor dan alat yang digunakan ialah *Laragon*, *VSCode*, *Web-Browser*, dan *Navicat*[14], [15].

Aplikasi berbasis web yang dinamakan *GodamSUB* ini nantinya akan digunakan sebagai media informasi penentuan rute bus terdekat untuk calon penumpang bus Damri di kota Surabaya. Dengan adanya *GodamSUB* ini diharapkan akan mempermudah calon penumpang atau warga Surabaya dalam mendapatkan informasi rutebus terdekat di wilayah kota Surabaya[13], [16].

2. Metode

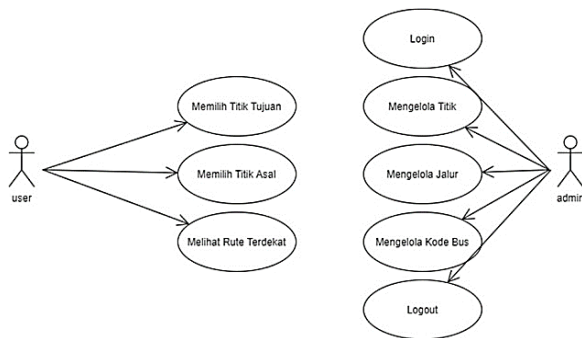
2.1. Pengumpulan Data

Metode pengumpulan data untuk mendapatkan rute beserta kode bus dilakukan dengan wawancara langsung ke Damri di jalan raya kali rungkut Surabaya,

juga dari aplikasi android milik Dishub Kota Surabaya bernama “GOBIS Surabaya Bus”,serta dari badan pusat statistik kota Surabaya di mana titik pemberhentian bus Damri di kota Surabaya berjumlah sebanyak 131 jalur [17][4].

2.2. Perancangan Sistem

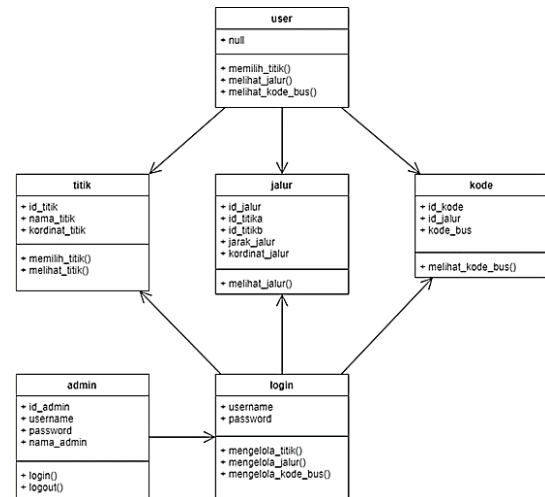
Hasil perancangan sistem menggunakan pendekatan *object oriented design* di mana untuk *use case diagram* dalam penelitian ini dapat dilihat pada Gambar 1 [6], [10], [18].



Gambar 1. Use case diagram

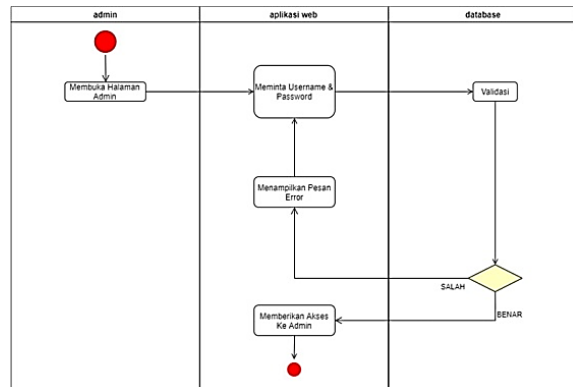
Pada Gambar 1 dijelaskan bahwa admin diharuskan untuk melakukan *login* terlebih dahulu. Jika sudah maka admin bisa melakukan mengelola titik, mengelola jalur, dan mengelola kode bus. *User* bisa melakukan memilih titik asal, memilih titik tujuan, dan melihat rute terdekat[19].

Untuk perancangan *class diagram* dapat dilihat pada Gambar 2, di mana dijelaskan lebih rinci mengenai prosedur yang terjadi di sistem tersebut dan juga adanya atribut dan operasi pada setiap aktor dan proses.



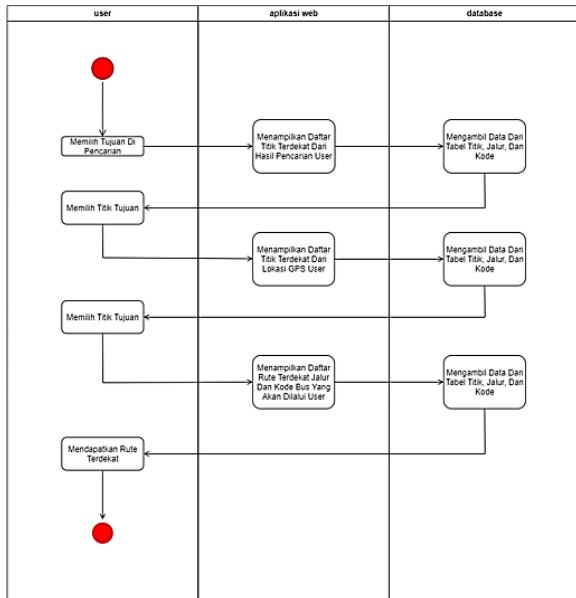
Gambar 2. Class diagram

Untuk perancangan *activity diagram* *login admin* dapat dilihat pada Gambar 3, di mana dijelaskan aktivitas *User* harus *login* sebagai admin untuk dapat mengakses halaman utama manajemen sebagai *administrator*.



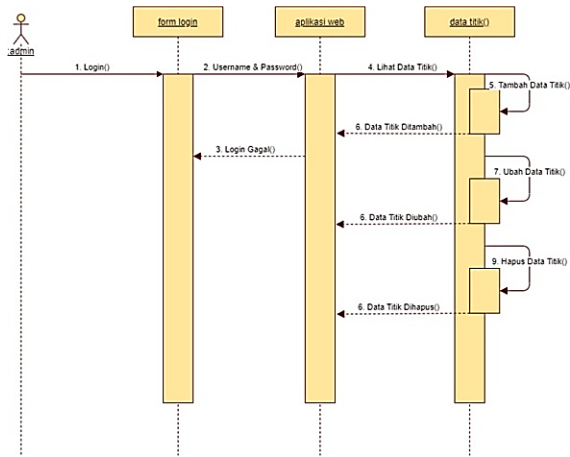
Gambar 3. Activity diagram login admin

Hasil perancangan *activity diagram* untuk pencarian rute terpendek dari data yang dibuat oleh admin data tersebut digunakan untuk pencarian rute terdekat. Pada Gambar 4 dijelaskan berbagai aktivitas dari *user*. Di mana *user* memilih tujuan di pencarian, memilih titik tujuan, memilih titik asal untuk mendapatkan rute bus Damri terdekat.



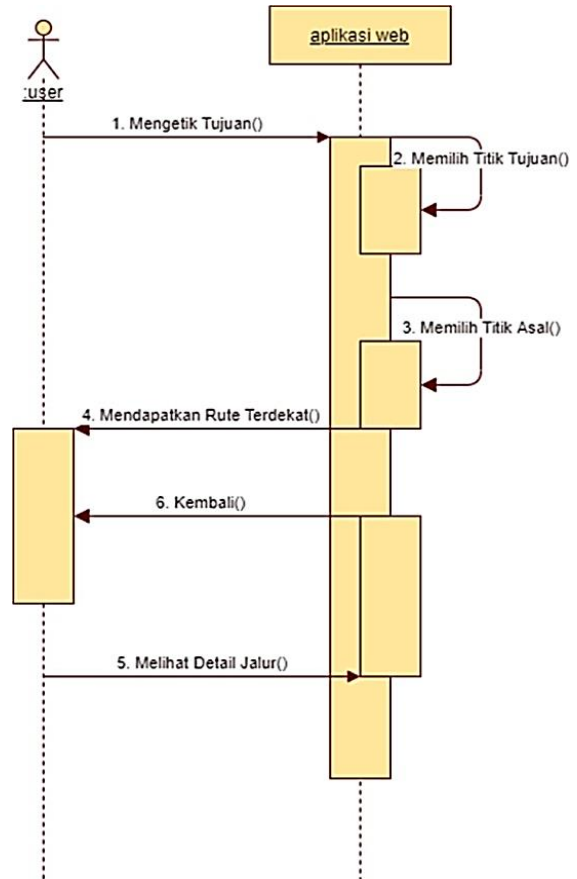
Gambar 4. Activity diagram pencarian rute terdekat

Hasil perancangan *sequence diagram* CRUD data titik rute dapat dilihat pada Gambar 5, di mana menampilkan urutan langkah admin dalam mengelola data titik rute.



Gambar 5. Sequence diagram CRUD data titik

Untuk perancangan *sequence diagram* pencarian rute terdekat dapat dilihat pada Gambar 6, di mana dijelaskan admin dapat mengubah jalur dengan menggeser jalur yang ada di *google maps*.



Gambar 6. Sequence diagram

3. Hasil dan Pembahasan

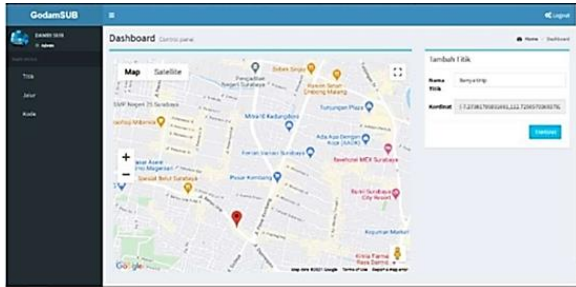
Hasil implementasi pengembangan sistem dalam penelitian ini di mana diawali dengan halaman *login* sebagai admin, seperti pada Gambar 7.



Gambar 7. Halaman login

Pada Gambar 7 *login* untuk akses halaman admin, di mana untuk keamanan maka admin harus terlebih dahulu

memasukkan *Username & password* untuk *login* agar nantinya bisa mengelolah data bus Damri Surabaya seperti tampak pada Gambar 8 [20][7].



Gambar 8. Halaman pengolahan rute

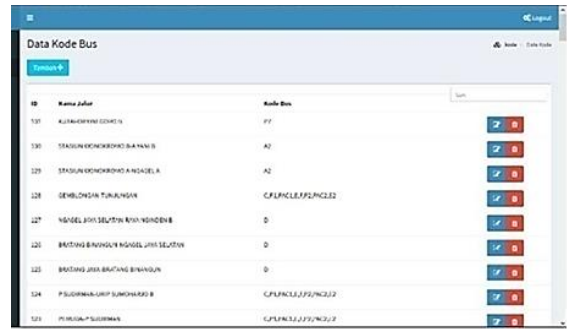
Kemudian pada Gambar 9 menunjukan di mana admin memilih titik asal dan titik tujuan dan menggeser jalur pada *google maps* untuk menentukan jalur antar kedua titik tersebut, lalu untuk menyimpan ke database admin harus meng-*click* tombol tambah.

Di mana pada Gambar 9 diatas admin melakukan peregeseran jalur pada *google maps* untuk menentukan jalur dengan kordinat baru lalu untuk menyimpan ke database admin harus meng-*click* tombol ubah.



Gambar 9. Halaman pergeseran rute

Untuk tambah data kode bus, pada Gambar 10 menunjukan di mana *login* sebagai admin juga dapat melakukan penambahan data tersebut, dengan cara *login* terlebih dahulu sebagai admin lalu masuk ke menu kode kemudian meng-*click* tombol tambah.



Gambar 10. Halaman tambah kode bus

Untuk melakukan penghapusan data jalur, pertama admin harus masuk ke menu kode lalu meng-*click* tombol hapus pada jalur yang akan dihapus untuk menghapus jalur tersebut dari *database*.

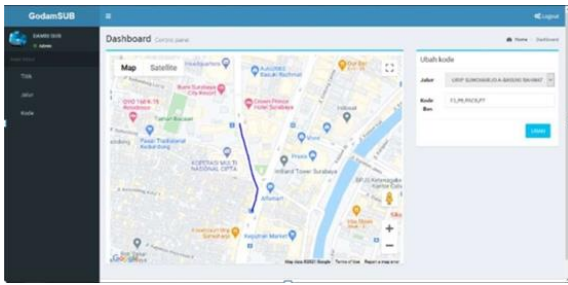
Hasil penambahan kode bus itu fasilitas yang diberikan akses sebagai admin untuk menambahkan jumlah armada baru nantinya dalam perjalanannya pemerintah kota melakukan kebijakan diperlukan penambahan jumlah armada, diamana sistem ini memiliki kemampuan penyesuaian, sedangkan hasil penambahan juga akan nampak pada Gambar 11.

Selain fasilitas penambahan seperti nampak pada Gambar 11, admin juga mampu melakukan manajemen perubahan kode armada bus seperti pada Gambar 12.



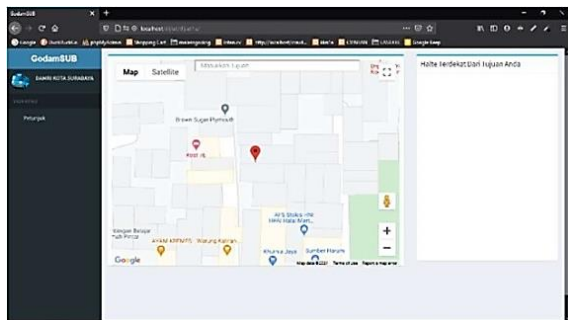
Gambar 11. Tampilan penambahan bus

Dalam pencarian rute terpendek yang merupakan titik terdekat tujuan oleh calon penumpang pengguna jasa bus Damri.



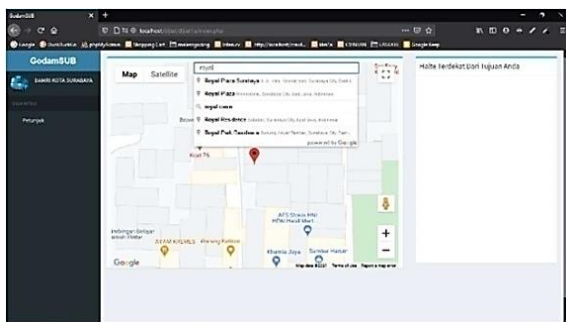
Gambar 12. Perubahan kode bus

User dapat menggunakan fasilitas seperti tampak pada Gambar 13, di mana user membuka halaman awal web melalui berbagai browser kemudian user akan melihat tampilan google maps melalui fasilitas pencarian.



Gambar 13. Halaman pencarian rute terdekat

Pada Gambar 13 fasilitas untuk melakukan pencarian rute bus Damri terdekat, pertama User harus memasukkan titik tujuan yang diinginkan pada kolom pencarian lalu memilih salah satu dari list untuk dijadikan titik tujuan, lebih detailnya bisa dilihat pada Gambar 14.



Gambar 14. Tampilan pilihan jalur

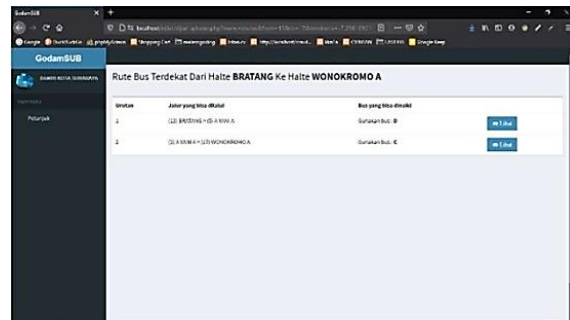
Di mana pada Gambar 15 user harus meng-click tombol pilih di salah satu halte

terdekat dari tujuan yang didapat dari hasil pencarian. Butuh waktu sekitar kurang lebih 1 menit untuk mendapatkan rekomendasi halte terdekat dari tujuan.



Gambar 15. Tampilan daftar halte terdekat

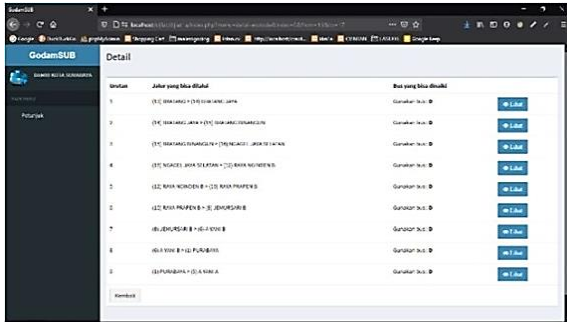
Pada Gambar 15 user harus meng-click tombol pilih di salah satu halte terdekat dari lokasi user. Butuh waktu sekitar kurang lebih 1 menit untuk mendapatkan rekomendasi halte terdekat dari lokasi user.



Gambar 16. Hasil penentuan rute terdekat menggunakan Dijkstra

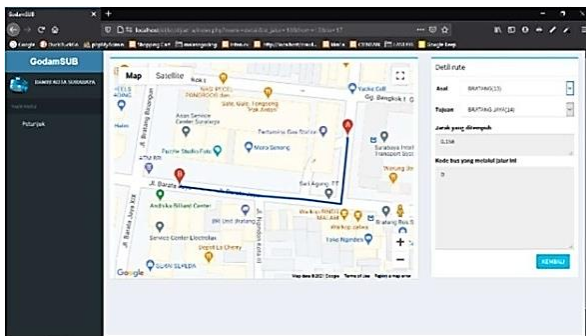
Pada Gambar 16 user mendapatkan rute terdekat berupa daftar urutan jalur dan kode bus yang bisa dinaiki. user dapat melihat detail jalur dengan meng-click tombol lihat di salah satu jalur.

Pada Gambar 17 user dapat melihat detail jalur yang dilalui serta kode bus pada masing-masing jalur.



Gambar 17. Detil rute dari Bratang ke Ahmad Yani.

Tampilan melalui *google maps* dapat dilihat pada Gambar 18, di mana *User* dapat melihat detail jalur secara grafis.



Gambar 18. Detil *google maps* rute dari Bratang ke Ahmad Yani.

4. Simpulan

Hasil penelitian yang sudah dilakukan, penulis menyimpulkan bahwa, bahwa algoritma *Dijkstra* bersifat *greedy*, di mana *google maps* tidak selalu akurat dalam menampilkan marker pada koordinat yang tersimpan di database dan juga *google maps directions service* tidak selalu bisa diandalkan dalam membuat jalur serta *distance matrix* API membutuhkan waktu lama untuk mencari halte terdekat dari lokasi *User* dan tujuan.

Aplikasi web GodamSUB yang dihasilkan dalam penelitian ini dapat membantu mencari kode bus yang dapat dipilih calon penumpang dengan rute terdekat secara detil dari halte pemberangkatan ke halte tujuan.

Adapun saran-saran yang dapat disampaikan untuk pengembang selanjutnya adalah untuk mempercantik

tampilan halaman web dan juga dapat menunjukkan jalur dari lokasi penumpang ke halte terdekat di *google maps* untuk menunjukkan jalur dari lokasi tujuan ke halte terdekat serta penambahan fitur *tracking*.

5. Daftar Pustaka

- [1] Nurdin, Syafrizal, Elfindri, and Syofyardi, "Transportation cost and regional trade perspective: Evidence of Indonesia logistic performance," *Opcion*, vol. 35, no. Special Issue 21, 2019.
- [2] J. Carneiro, P. Alves, G. Marreiros, and P. Novais, "Group decision support systems for current times: Overcoming the challenges of dispersed group decision-making," *Neurocomputing*, vol. 423, 2021.
- [3] Rifqi Hanan Riokka and S. Zainab, "Mapping of Restaurant Distribution in The Wonokromo District Area Using Google Platform," *CI-TECH*, vol. 2, no. 01, 2021.
- [4] A. Setiawan, L. Y. Hendrati, and Y. A. Mirasa, "The Mapping and Analysis of Diphtheria Cases in Surabaya (2017-2018)," *Jurnal Biometrika dan Kependudukan*, vol. 10, no. 1, 2021.
- [5] L. Aggarwal, P. Goswami, and S. Sachdeva, "Multi-criterion Intelligent Decision Support system for COVID-19," *Applied Soft Computing*, vol. 101, 2021.
- [6] M. H. Firmansyah, "Implementation of Occupational Safety and Healthy Risk Management in Islamic Hospital Surabaya A. Yani," *Medical and Health Science Journal*, vol. 5, no. 2, 2021.
- [7] A. A. A. Gassar and S. H. Cha, "Review of geographic information systems-based rooftop solar photovoltaic potential estimation approaches at urban scales," *Applied Energy*, vol. 291, 2021.
- [8] M. Klum, T. Schenck, A. Pielmus, T. Tigges, and R. Orglmeister, "Short distance impedance pneumography," *Current Directions in Biomedical Engineering*, vol. 4, no. 1, 2018.

- [9] D. D. Zhu and J. Q. Sun, "A New Algorithm Based on Dijkstra for Vehicle Path Planning Considering Intersection Attribute," *IEEE Access*, vol. 9, 2021.
- [10] G. Sun and J. Zacharias, "Can bicycle relieve overcrowded metro? Managing short-distance travel in Beijing," *Sustainable Cities and Society*, vol. 35, 2017.
- [11] M. Vallet-Regi and A. J. Salinas, "Role of the short distance order in glass reactivity," *Materials*, vol. 11, no. 3, 2018.
- [12] Y. L. Zheng *et al.*, "Exploring a new adaptive routing based on the dijkstra algorithm in optical networks-on-chip," *Micromachines*, vol. 12, no. 1, 2021.
- [13] M. Akram, A. Habib, and J. C. R. Alcantud, "An optimization study based on Dijkstra algorithm for a network with trapezoidal picture fuzzy numbers," *Neural Computing and Applications*, vol. 33, no. 4, 2021.
- [14] R. B. Velasco, I. Carpanese, R. Interian, O. C. G. Paulo Neto, and C. C. Ribeiro, "A decision support system for fraud detection in public procurement," *International Transactions in Operational Research*, vol. 28, no. 1, 2021.
- [15] Y. Yun, D. Ma, and M. Yang, "Human-computer interaction-based Decision Support System with Applications in Data Mining," *Future Generation Computer Systems*, vol. 114, 2021.
- [16] X. Ren, S. Wu, Z. Xing, R. Xu, W. Cai, and Z. Lei, "Behavioral responses of western flower thrips (*Frankliniella occidentalis*) to visual and olfactory cues at short distances," *Insects*, vol. 11, no. 3, 2020.
- [17] A. Habib and A. Kartika W. H., "Development of an Online Sales Information System for SMEs Using Incremental Methods," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 4, no. 1, pp. 51–62, 2020.
- [18] A. Hajar, I. Nabawi, L. Kartikawati, F. R. Yudana, S. Budi, and N. Prasetiyantara, "Pengolahan Data Spasial-Geolocation Untuk Menghitung Jarak 2 Titik," *Creative Information Technology Journal*, vol. 8, no. 1, 2021.
- [19] A. Habib, M. A. Jani, D. A. Pratama, and E. Ronando, "Development of archives management information system with RFID and SMS gateway," *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 4, pp. 5227–5243, 2020.
- [20] A. Habib, A. Darwanto, E. Ronando, and Slamet, "Pump control system using microcontroller and Short Message Service (SMS) gateway for flood prevention," *Springer Proceedings in Physics*, vol. 193, no. January, pp. 607–621, 2017.