

## Optimasi Web GIS Pendoron Darah Tetap Di Kabupaten Cilacap Dengan Algoritma Dijkstra

Emi Nadiyah\*<sup>1</sup>, Ninik Agustin<sup>2</sup>, Mizan Ahmad<sup>3</sup>

*Informatika, Universitas Nahdlatul Ulama Al Ghazali Cilacap  
Jl. Kemerdekaan Barat No 17 RT.01 Rw.05 Kesugihan Kidul, Kab Cilacap, Jawa Tengah*

<sup>1</sup>eminadiyah@gmail.com

<sup>2</sup>ninik.agustin@unugha.id

<sup>3</sup>mizanahmad36@gmail.com

Dikirim pada 17-10-2024, Direvisi pada 28-10-2024, Diterima pada 10-11-2024

### Abstract

Dalam dunia medis, stok darah sangat penting dan sangat berguna dalam pelayanan kesehatan. Kekurangan stok darah dapat membahayakan nyawa pasien dan menunda tindakan medis yang mendesak. Saat ini, pencari pendonor darah sering kali harus mencari sendiri calon donor, yang sering kali memakan waktu dan tidak efisien. Dengan kondisi tersebut, diperlukan sebuah sistem yang mampu membantu menyelesaikan masalah kekurangan stok darah secara lebih efektif dan efisien. Sistem ini dirancang dengan menggunakan bahasa pemrograman PHP dengan *framework* Laravel versi 10. Algoritma Dijkstra bekerja untuk menentukan rute terpendek menuju lokasi pendonor darah. Penelitian ini bertujuan menciptakan platform pencarian pendonor darah yang dapat memudahkan masyarakat dalam mencari donor berdasarkan golongan darah yang dibutuhkan. Selain itu, sistem ini juga memungkinkan pengguna untuk memantau stok darah terkini yang ada di Palang Merah Indonesia (PMI) di wilayah Kabupaten Cilacap. Pengembangan sistem ini dilakukan dengan menggunakan metode SDLC *prototype*. Proses dimulai dengan melakukan analisis kebutuhan, dilanjutkan dengan perancangan dan pengembangan sistem. Sistem ini mampu memetakan lokasi donor darah terdekat, berdasarkan golongan darah yang dicari. Algoritma Dijkstra digunakan untuk menghitung rute terpendek dari lokasi pencari menuju pendonor. Hasil perhitungan algoritma ini memastikan bahwa pengguna dapat menemukan rute tercepat dan termudah untuk mencapai lokasi donor.

**Keywords:** Algoritma Dijkstra, Web GIS, Rute terdekat

*This is an open access article under the [CC BY-SA](#) license.*



---

### Corresponding Author:

Emi Nadiyah

Informatika, Universitas Nahdlatul Ulama Al Ghazali Cilacap, Jalan Kemerdekaan Barat No. 17, Gligir, Kesugihan Kidul, Kec. Kesugihan Kabupaten Cilacap, Jawa Tengah, 53274, Indonesia. Email: [eminadiyah@gmail.com](mailto:eminadiyah@gmail.com)

---

## I. PENDAHULUAN

Palang Merah Indonesia (PMI) adalah lembaga yang berperan dalam menyediakan darah bagi yang memerlukan. PMI bekerja sama dengan rumah sakit yang terdaftar untuk memudahkan proses pemesanan darah bagi pasien, salah satunya PMI yang ada di Kabupaten Cilacap. Berdasarkan data statistik Kabupaten Cilacap, menerima donatur darah sebanyak 18.557 dan digunakan untuk tindakan medis sebanyak 17.487. Namun, masalah muncul ketika persediaan darah habis, sehingga pasien harus mencari pendonor darah sendiri untuk memenuhi kebutuhannya [1]. Proses donor darah melibatkan pengambilan darah dari individu secara sukarela untuk disimpan di bank darah yang kemudian digunakan untuk transfusi darah. Darah yang didonorkan memiliki nilai sangat penting bagi pasien yang mengalami berbagai kondisi medis, termasuk kecelakaan, transpalasi organ, kanker, anemia, thalassemia, dan leukemia. Apabila kebutuhan

darah terpenuhi, hal ini dapat menghindari penundaan operasi atau mengurangi resiko kegagalan operasi bagi pasien [1].

Pada bulan September tahun 2022, data statistik Kabupaten Cilacap menunjukan jumlah donatur lebih sedikit dari jumlah penerima donator, artinya masih belum mencukupi kebutuhan darah. Proses penyampaian informasi terkait donor darah terhadap masyarakat masih kurang, sehingga harus selalu aktif mendatangi PMI Kabupaten Cilacap. Hal ini tentu tidak efektif dan efisien. Perkembangan teknologi informasi mendorong inovasi diberbagai bidang termasuk dalam bidang layanan informasi geografis. Teknologi informasi geografis dapat digunakan untuk membantu menyelesaikan permasalahan tersebut. Sistem informasi geografis berbeda dengan sistem biasa, karena dengan teknologi GIS mampu mengambil, menganalisa, mengecek, mengintegrasikan, mengolah atau memproses data dan menampilkanya secara spasial serta merepresentasikan bentuk atau kondisi bumi [2]. Sistem ini dirancang berbasis *website* agar dapat diakses dalam berbagai perangkat dan *platform* dengan menggunakan *browser web* standar, sehingga pengguna tidak perlu mengunduh aplikasi khusus pada perangkat mereka. Hal ini membuat situs *web* lebih mudah diakses dan lebih cepat diadopsi oleh pengguna [3].

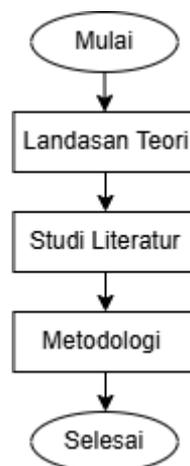
Pada tahun 1950, masalah penentuan jalur terpendek telah menjadi topik yang sering dibahas dan diteliti. Penerapan pencarian jalur terpendek ini telah diaplikasikan di berbagai bidang untuk mengoptimalkan kinerja transmisi otomatis. Algoritma ini bekerja dengan membandingkan bobot terkecil dari titik awal hingga titik tujuan untuk menemukan jalur yang paling efisien dan efektif [5]. Pada penelitian yang sudah ada, Algoritma Dijkstra digunakan untuk menentukan rute terdekat, Seperti penelitian yang dilakukan oleh Saputrama, untuk menentukan rute terdekat dari kampus A ke B di UIN Raden Fatah [4]. Di penelitian ini, penulis menggunakan Algoritma Dijkstra untuk menyelesaikan permasalahan terkait rute terdekat dalam pencarian pendonor darah. Dengan menggunakan Algoritma Dijkstra, maka bisa lebih menghemat waktu, karena ada beberapa opsi terkait pendonor, dan yang dipilih adalah pendonor terdekat.

Algoritma Dijkstra memproses dengan lebih cepat daripada algoritma lain seperti A\*(A Star) [6]. Algoritma A\* menggunakan perhitungan *heuristic Euclidean*, dimana mengansumsikan garis lurus. Dengan demikian akan terjadi kesalahan atau selisih perhitungan karena tidak semua jalur lurus. Seperti penelitian yang dilakukan oleh Yusra Fernando, terdapat selisih 21 meter dari jarak sesungguhnya [7]. Dalam penelitian lain menyebutkan bahwa, Algoritma Dijkstra memiliki kompleksitas waktu lebih kecil untuk menghitung waktu program daripada Algoritma Bellman-Ford [8].

Algoritma Dijkstra digunakan untuk menentukan rute terdekat menuju lokasi pendonor. Lokasi pendonor darah tersebar di beberapa titik, untuk menghemat waktu, maka Algoritma Dijkstra bekerja dengan menentukan rute terdekatnya. Kemudian, untuk pengembangan sistem ini dibangun menggunakan *Web GIS* untuk mempermudah pencari pendonor darah ketika membutuhkan darah. Dengan adanya sistem ini, pencari pendonor darah bisa melakukan pencarian dengan menggunakan perangkat seluler atau pc pengguna. Sistem ini nantinya bisa menampilkan terkait rute menuju lokasi pendonor darah. Rute terpendek tersebut dihasilkan dari perhitungan menggunakan Algoritma Dijkstra.

## II. METODOLOGI PENELITIAN

Penelitian ini memiliki alur seperti gambar berikut:



Gambar 1. Diagram Alir Penelitian

### 1. Landasan Teori

#### a. Web GIS

*Geographic Information System (GIS)* merupakan informasi mengenai karakteristik tempat-tempat di permukaan bumi, karakteristik permukaan bumi, dan lokasi objek. *Web GIS* dapat mengambil, menganalisa, dan mengintegrasikan, mengolah, atau memproses dan menampilkannya secara spasial serta menampilkan bentuk atau kondisi bumi.

#### b. Algoritma Dijkstra

Algoritma Dijkstra merupakan algoritma yang bisa digunakan dalam pencarian rute terdekat. Perbandingan kompleksitas waktu Algoritma Dijkstra lebih cepat dalam memproses data dibandingkan Algoritma Bellman-Ford, sehingga Algoritma Dijkstra digunakan dalam penelitian ini dibandingkan Algoritma lain. Cara kerjanya dengan membandingkan bobot terkecil dari *node* awal hingga *node* akhir. Berikut cara kerja Algoritma Dijkstra:

- Inisialisasi titik awal ke tujuan.
- Beri label sementara jarak pada tiap titik.
- Menentukan nilai terkecil dari setiap titik label sementara.
- Memberi tanda label permanen dari titik terkecil.
- Menghapus label sementara dari daftar.
- Melakukan perbandingan jarak terdekat melalui label permanen.
- Jika titik terdekat ditemukan lebih dekat maka label yang dibandingkan diberi label permanen, jika tidak maka diabaikan.
- Lintasan terpendek ditemukan.

#### c. PMI

Palang Merah Indonesia (PMI) merupakan organisasi nasional di Indonesia yang berkitmen dalam pergerakan sosial kemanusiaan, dimana mempertahankan tujuh prinsip-prinsip dasar Internasional Palang merah dan Bulan Sabit Merah.

#### d. PHP

Kerangka PHP membantu pengembang dalam membangun aplikasi *web* menjadi lebih cepat dan mudah dengan menyediakan kerangka dasar. PHP dilengkapi fitur API, *libraries*, *extentions*, dan membantu pengembang menjadi lebih produktif dalam mengurangi kode berulang dalam proyek.

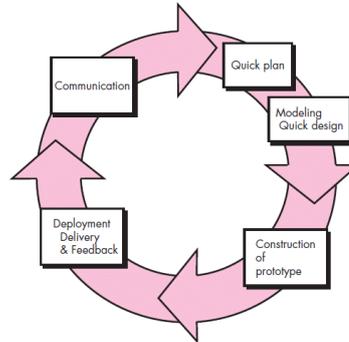
#### e. Google Maps API

Merupakan layanan peta virtual yang disediakan oleh Google. Google API memungkinkan aplikasi untuk melihat, menyimpan, dan memperbaharui peta dalam bentuk data API

dengan menggunakan model fitur dalam peta.

## 2. Metodologi Pengembangan Sistem

Metode yang digunakan untuk pengembangan sistem pada penelitian ini adalah SLDC *prototype*. Metode *prototype* dalam pengembangannya pengguna sudah diikuti sertakan dengan tujuan lebih akrab dan mengetahui system yang tekah dibuat, sedangkan metode *waterfall* focus menyelesaikan sistem dahulu, baru menunjukkan kepada pengguna. Berikut alur pengembangan sistem:



Gambar 2. Metode *Prototype*

### a. *Communication*

Langkah dimana pengembang mengadakan pertemuan dengan pihak terkait untuk mengidentifikasi kebutuhan perangkat lunak yang sudah diketahui saat ini, serta untuk menjelaskan area-area yang memerlukan definisi lebih lanjut pada iterasi berikutnya.

### b. *Quick Plan*

Proses pembuatan prototype dilakukan dengan cepat, diikuti pemodalan dalam bentuk rancangan cepat.

### c. *Modelling Quick Design*

Perancangan dimodelkan menggunakan model berorientasi objek dengan memanfaatkan tools uml, seperti use case untuk mendefinisikan fungsi sistem, class diagram untuk menampilkan clas-clas sistem, *activity diagram* untuk menggambarkan alur proses sistem.

### d. *Construction of prototype*

Rancangan cepat menjadi dasar untuk memulai pembangunan prototype, dengan focus pada representasi aspek-aspek perangkat lunak yang terlihat oleh pengguna akhir, seperti desain antarmuka pengguna atau format tampilan.

### e. *Development Delivery and Feedback*

Pengembangan prototype diberikan kepada *client* untuk mengevaluasi hasil yang telah dibuat dan memberikan masukan yang digunakan untuk memperbaiki spesifikasi kebutuhan. Iterasi terjadi ketika pengembang melakukan penyempurnaan terhadap prototype tersebut. Development atau pengembangan sistem pencarian pendonor dengan Algoritma Dijkstra melalui beberapa tahapan sebagai berikut:

#### 1) Perancangan Desain

Pada tahap ini, pengembangan melakukan desain pada figma karena akan mempermudah pengembang dalam melakukan revisi atau perbaikan

#### 2) Perancangan Database

Entitas utama dalam diagram ini mencakup admin, profil, *user*, stok darah, pendonor, pencari, *node*, *graf*. Setiap entitas memiliki *atribut* spesifik, misalnya entitas profi memiliki *atribut* seperti *user\_id*, nama, tempatlahir, tanggallahir, jeniskelamin, alamat, desa, kecamatan, kabupaten, provinsi, kodepos, golongan\_darah, rhesus, pekerjaan, *update*, *create*

#### 3) Perancangan Backend

Perancangan *backend* untuk sistem pencari pendonor darah GIS menggunakan Algoritma Dijkstra dibangun dengan menggunakan PHP *Framework* Laravel versi 10 dan menggunakan *mysql* sebagai *database*. Struktur proyek terdiri dari *file* utama *dashboard.php* sebagai direktori *routes* utama, direktori *controller* yang mengandung logika bisnis, dan direktori *models* untuk definisi skema dengan *mysql*. Fitur utama dari sistem ini mengambil semua pengguna yang tersimpan di *mysql* dan membuat pengguna baru melalui *url* yang ditentukan, selain itu konfigurasi disimpan dalam *file .env* untuk mengatur *port serve* dan *url koneksi mysql*.

#### 4) Perancangan *Frontend*

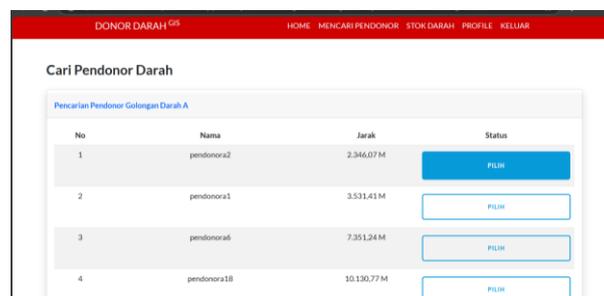
Perancangan *frontend* untuk sistem ini dibangun dengan menggunakan bahasa pemrograman PHP *framework* laravel versi 10. Menggunakan *blade templating engine* untuk integrasi logika PHP secara efisien dan aman. Dengan *blade*, dapat membuat komponen dinamis dan terstruktur yang mengelola tampilan aplikasi. Kemudian, untuk CSS *Framework* menggunakan *bootstrap* untuk *styling* yang responsif dan konsisten.

#### 5) *Deployment*

Untuk melakukan *deployment website* agar dapat diakses secara publik maka perlu di *hosting*. *Upgrade* atau *Update* kebutuhan untuk *web serve* untuk manajemen proses. Unggah *file source code* ke VPS menggunakan *AaPanel File Manager*.

### III. HASIL DAN PEMBAHASAN

Hasil pengembangan system untuk pencarian pendonor darah dengan Algoritma Dijkstra adalah menghasilkan *website* GIS menggunakan Algoritma Dijkstra untuk pencarian rute terpendek pada pendonor darah. Sistem ini dibuat menggunakan bahasa pemrograman PHP dengan *framework* Laravel dan Visual Studio Code sebagai IDE untuk menulis programnya dengan menggunakan metode *SLDC prototype*. Dengan menggunakan 5 sampel titik pendonor darah A, 9 sampel titik pendonor darah B, 3 sampel titik pendonor darah AB, dan 3 sampel titik untuk pendonor darah O.



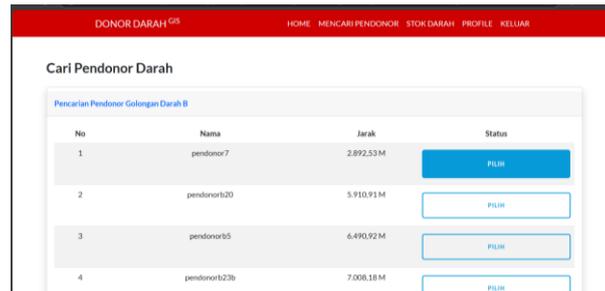
No	Nama	Jarak	Status
1	pendonora2	2.346,07 M	<input type="button" value="PILIH"/>
2	pendonora1	3.531,41 M	<input type="button" value="PILIH"/>
3	pendonora6	7.351,24 M	<input type="button" value="PILIH"/>
4	pendonora18	10.130,77 M	<input type="button" value="PILIH"/>

Gambar 3. Pencarian Gol Darah A



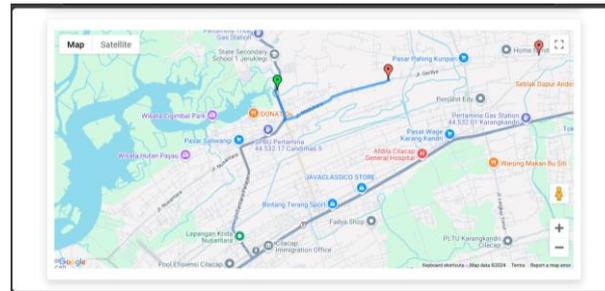
Gambar 4. Hasil Rute Pencarian Gol Darah A

Gambar di atas adalah hasil pencarian golongan darah A terdekat dari lokasi pencarian. Dengan total 5 pendonor untuk golongan darah A, secara *default* sistem menampilkan hasil pencarian golongan yang terdekat dengan rutenya. Selain itu, terdapat fitur untuk memilih pendonor lain.



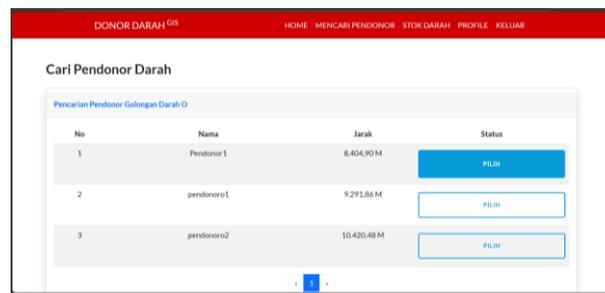
No	Nama	Jarak	Status
1	pendonor7	2,892,53 M	<input type="button" value="PILIH"/>
2	pendonorb20	5,910,91 M	<input type="button" value="PILIH"/>
3	pendonorb5	6,490,92 M	<input type="button" value="PILIH"/>
4	pendonorb23b	7,008,18 M	<input type="button" value="PILIH"/>

Gambar 5. Pencarian Golongan Darah B



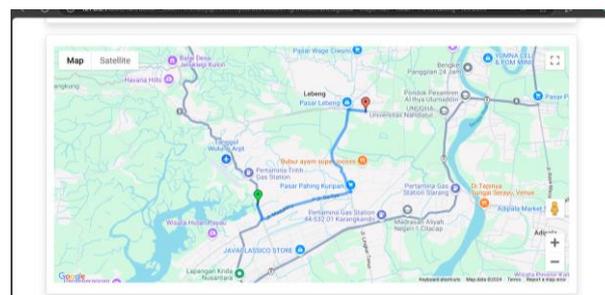
Gambar 6. Hasil Rute Pencarian Golongan Darah B

Gambar di atas adalah hasil pencarian golongan darah B terdekat dari lokasi pencarian. Dengan total 9 pendonor untuk golongan darah A, secara *default* sistem menampilkan hasil pencarian golongan yang terdekat dengan rutanya.



No	Nama	Jarak	Status
1	Pendoror 1	8,404,90 M	<input type="button" value="PILIH"/>
2	pendonor01	9,291,86 M	<input type="button" value="PILIH"/>
3	pendonor02	10,420,48 M	<input type="button" value="PILIH"/>

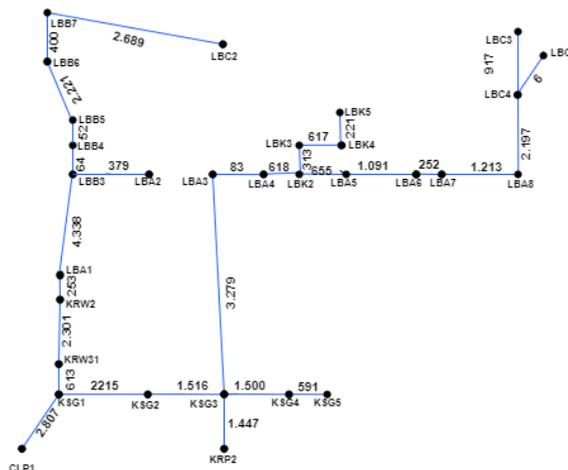
Gambar 7. Pencarian Pendoror O



Gambar 8. Hasil Rute Pencarian Golongan Darah O

Gambar di atas adalah hasil pencarian golongan darah O terdekat dari lokasi pencarian. Dengan total 3 pendonor untuk golongan darah A, secara *default* sistem menampilkan hasil pencarian golongan yang terdekat dengan rutanya.





Gambar 12. Graf Perhitungan Sistem

#### IV. KESIMPULAN

Berdasarkan penelitian yang sudah dilakukan, penelitian menghasilkan:

1. Pengembangan website pencarian pendonor darah tetap menggunakan Algoritma Dijkstra menghasilkan sebuah *platform* yang dapat digunakan oleh pengguna ketika ingin mencari pendonor darah yang lebih efektif dan efisien daripada pencarian secara manual. Platform ini dikembangkan dengan PHP framework laravel versi 10. Fitur utama yang dimiliki yaitu pencarian pendonor bagi pengguna dengan memakai layanan Google Maps.
2. Fitur pencarian menggunakan Algoritma Dijkstra dikembangkan dengan membuat *node-node* terlebih dahulu. Setelah membuat *node*, langkah selanjutnya adalah menghubungkan *node-node* yang sudah dibuat dan menghasilkan *vertex*. Kemudian membuat titik pendonor mulai dari golongan A, B, AB, dan O. Alur kerja sistem untuk menentukan rute terdekat dengan menfilter terlebih dahulu golongan darah yang akan dicari, setelah itu Algoritma Dijkstra bekerja untuk menentukan rute terdekat menuju lokasi pendonor.

#### DAFTAR PUSTAKA

- [1] A. L. Turnip, "Sistem Informasi Geografis Pendonor Darah Tetap Di Bandar Lampung Menggunakan Algoritma Dijkstra," *JECISIT*, vol. 1, no. 1, pp. 70–77, 2021.
- [2] R. Kurniawan, S. Fachrurrazi, and M. Ula, "Sistem Informasi Geografis Pencarian Rute Terpendek Menuju Rumah Sakit Dengan Menggunakan Metode Algoritma Dijkstra," *J. Sist. Inf.*, vol. 1, no. 1, pp. 598–599, 2020.
- [3] Ve. Y. R. H. Ari Purwanto, Verdi Yasin, "Perancangan Aplikasi Teknologi Informasi Helpdesk Berbasis Web Pada Instalasi Rekam Medik Dan Admisi RSCM Jakarta," *J. Widya*, vol. 2, no. 2, pp. 129–145, 2021.
- [4] H. R Saputrama, "Penerapan Algoritma Dijkstra untuk Menentukan Rute Terpendek dari Kampus A UIN Raden Fatah ke Tempat Bersejarah di Palembang," *E-Jurnal Mat.*, vol. 10, no. 3, pp. 173–178, Sep. 2021.
- [5] H. Pratiwi, "Application Of The Dijkstra Algorithm To Determine The Shortest Route From City Center Surabaya To Historical Places," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 4, no. 1, pp. 213–223, Jan. 2022.
- [6] R. Umar, "Analisis Perbandingan Algoritma Dijkstra, A-Star, Dan Floyd Warhsal Dalam Pencarian Rute Terdekat Pada Objek Wisata Kabupaten Dompu," *JTIK*, vol. 8, no. 2, pp. 227–234, 2021.
- [7] Y. Fernando, M. A. Mustaqov, and D. A. Megawaty, "Penerapan Algoritma a-Star Pada Aplikasi Pencarian Lokasi Fotografi Di Bandar Lampung Berbasis Android," *J. Teknoinfo*, vol. 14, no. 1, pp. 27–34.
- [8] A. Serdano, M. Zarlis, and D. Hartama, "Perbandingan Algoritma Dijkstra dan Bellman-Ford Dalam Pencarian Jarak Terpendek Pada SPBU," *Semin. Nas. Sains Teknol. Inf. SENSASI 2019*, vol. 1, no. 1, pp. 259–264, 2019.