

Implementasi Algoritma Dijkstra Pada Pencarian Rute Terpendek ke Museum di Jakarta

Aldy Cantona¹, Fauziah², Winarsih³

^{1,2,3} Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional

Info Artikel

Riwayat Artikel

Diterima: 20-12-2019

Disetujui: 19-03-2020

Kata Kunci

Algoritma Dijkstra;

Android;

Flutter;

Jalur Terpendek;

Museum

✉ Corresponding Author

Aldy Cantona

Fakultas Teknologi

Komunikasi dan Informatika,

Universitas Nasional

Tel. +62 85770736557

cantona642@gmail.com

ABSTRAK

Persoalan jarak dan awamnya pencarian rute adalah masalah setiap orang ketika mengunjungi tempat-tempat tertentu. Pada penelitian ini, algoritma Dijkstra digunakan untuk mengkalkulasi jarak paling dekat dari satu titik ke museum yang dipilih menjadi tujuan. Pada aplikasi ini dirancang agar beroptimasi mencari rute ke museum dengan bobot jarak paling pendek di kota Jakarta. Dalam proses pencarian mampu diselesaikan menggunakan metode algoritma dijkstra sehingga kita mampu mengetahui tahapan kerja aplikasi dengan solusi perhitungan metode algoritma dijkstra. Mencari jalur terpendek dengan metode algoritma dijkstra dalam bentuk graf dengan bobot dan pemetaan area yang saling terhubung melalui jalur yang sudah ditentukan. Aplikasi ini dirancang agar berjalan pada smartphone dengan platform android yang digunakan mayoritas pada saat ini. Berdasarkan pengujian yang dilakukan, penerapan algoritma Dijkstra untuk mendapatkan rute terpendek dinilai efektif sebab dengan total 20 bobot hanya memerlukan 7 bobot, yang mana hal tersebut menyatakan hanya perlu 35% bobot untuk mencapai Museum Nasional.

PENDAHULUAN

Algoritma Dijkstra adalah salah satu algoritma yang digunakan untuk menyelesaikan masalah jarak terpendek (shortest path problem) pada sebuah graf yang terarah (directed graph). Pada implementasi penelitian ini diterapkan pada museum-museum di Jakarta yang memiliki banyak museum yang bersejarah dengan banyaknya museum yang menyimpan sejarah panjang sampai kemerdekaan Indonesia kini (Retnani, Istiadi, & Roqib, 2015). Karna itulah banyak museum-museum yang di bangun di Indonesia untuk mengenal kembali sejarah. Museum juga sebagai institusi permanen, layanan publik, nirlaba, yang terbuka untuk umum dengan dilakukan usaha mengkoleksi, melekaikan konservasi dan riset, serta berkomunikasi dan memamerkan benda yang benar keberadaannya di museum pada masyarakat umum sebagai keperluan studi ilmiah, pendidikan, antusiasme, serta hobi. Oleh karena itulah museum dapat menjadi tempat bahan studi untuk akademis, dokumentasi, ciri khas masyarakat tertentu, dan acuan gambaran imajinatif untuk masa depan. Sejak tahun 1997 tanggal 18 Mei diperingati sebagai hari museum internasional (international museum day),

sedangkan tanggal 12 Oktober diperingati sebagai hari museum nasional di Indonesia(Kusuma, 2018).

Pada tahun 2000 telah banyak algoritma yang bisa menyelesaikan masalah mencari lintasan terpendek yang pernah ditulis. Salah satu algoritma yang paling sesuai untuk solusi terhadap kasus yang menggunakan graf alternative adalah algoritma dijkstra (nama algoritma ini dipilih sesuai dengan nama penemunya)(Yusuf, Az-zahra, & Apriyanti, 2017). Pada algoritma Dijkstra diterapkan cara untuk mencari lintasan paling pendek pada graf berarah. Tetapi, algoritma ini juga berlaku untuk sebuah graf tak berarah(Nugroho, Nurhayati, & Widiyanto, 2015).

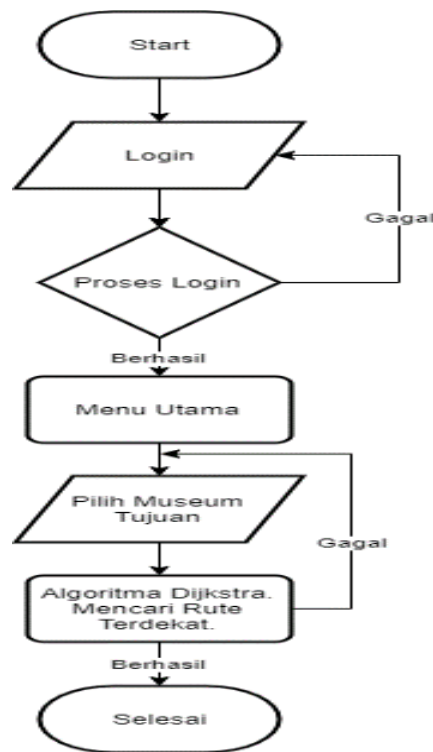
Pada penelitian sebelumnya membahas tentang implementasi metode Dijkstra Dalam Mobile Aplikasi Pencarian SPBU terdekat di Kota Palembang, menggunakan sistem aplikasi yang diintegrasikan pada perhitungan algoritma Dijkstra untuk mencari rute terdekat. Dasar prinsip algoritma Dijkstra adalah dengan mencari dua lintasan dari setiap node dengan bobol nilai paling kecil. Konsep yang saya gunakan kali ini adalah untuk pencarian lokasi museum yang paling dekat dengan posisi pengguna menggunakan GPS (Global Positioning System) yang ada pada smartphone berplatform android agar mengetahui posisi awal pada Algoritma Dijkstra. Smartphone android pada saat ini banyak sekali digunakan oleh mayoritas masyarakat masa kini. Penggunaan aplikasi android sangat mudah dijangkau oleh kalangan masyarakat. Dengan adanya aplikasi ini diharapkan mampu memberikan efektifitas waktu dan jarak selama perjalanan saat mencari rute ke museum (Oktoviana, Mohamad, Yasindan Lucky & Sholichin, 2017).

METODE

Pada prinsipnya penggunaan piranti Flutter yang digunakan untuk membuat aplikasi ini dimana Flutter berperan pada pembuatan aplikasi, mengembangkan aplikasim dokumentasi serta pengetesan aplikasi. Pembuatan aplikasi dilakukan dengan melakukan input *coding system* menggunakan bahasa *Dart* (Junanda, Kurniadi, & Huda, 2016). Sehingga setiap sistem akan menjalankan fungsinya masing-masing dengan output yang akan berjalan sesuai pada rancangan aplikasi yang telah dibuat.

A. Flowchart

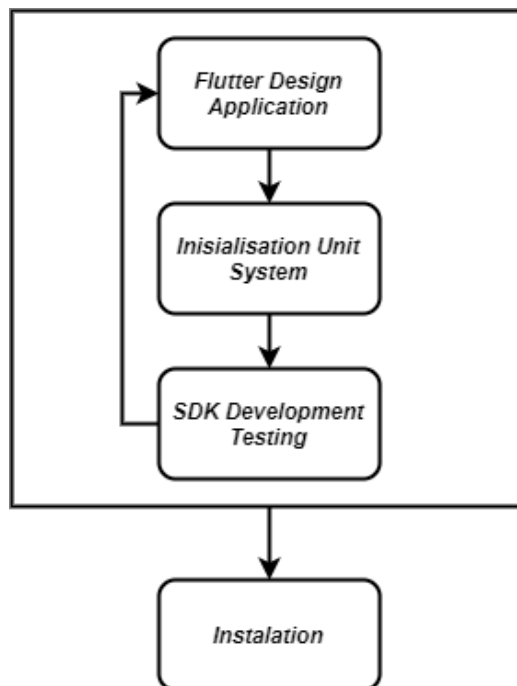
Flowchart berisikan alur proses seluruh tahap dalam mencari rute terdekat menuju museum. Dimulai pada tahap pertama menuju museum yang ingin dikunjungi masuk pada tahap proses pencarian dengan menganalisa rute, hingga mendapatkan rute yang paling efektif (Candra Galih & Andriyat Krisdiawan, 2018). Berikut adalah rancangan aplikasi yang akan dibuat menggunakan alur *flowchart*.



Gambar 1. Flowchart Aplikasi

B. Perancangan Aplikasi

Penelitian ini diawali dengan pengumpulan data dan pemetaan lokasi museum di Kota Jakarta. Data lain yang dibutuhkan adalah data jalan, persimpangan, dan info jalur. Data tersebut akan digunakan untuk meng-gambarkan lokasi dengan jalur terdekat ke museum(Ginting et al., 2018). Perancangan aplikasi ini menggunakan beberapa pengembangan perangkat mobile apps.perangkatini membantu proses pembuatan dimulai dari membuat aplikasi samai melakukan test perangkat lunak sebelum digunakan langsung pada perangkat android(Triansyah & Informatics, 2013). Berikut merupakan diagram perancangan perangkat.



Gambar 2. Diagram Perancangan Perangkat

C. Algoritma Dijkstra

Algoritma Dijkstra adalah salah satu algoritma dalam mencari rute. Pada algoritma Dijkstra juga termasuk dalam algoritma untuk masalah pencarian graf yang mampu menuntaskan masalah mencari lintasan terpendek dengan satu sumber pada sebuah graf yang digunakan untuk menyelesaikan masalah lintasan terpendek dengan satu sumber pada sebuah graf yang tidak memiliki *cost* sisi negatif, dan menghasilkan sebuah alur lintasan terpendek. Algoritma Dijkstra ini sering digunakan untuk melakukan routing (Aprilianingsih, Primananda, & Suharsono, 2017). Algoritma Dijkstra ini sendiri ditujukan untuk menemukan jalur terpendek berdasarkan bobot terkecil dari titik keberangkatan ke titik lainnya. Misalkan sebuah gedung dan monumen dijadikan titik dan jalanan dijadikan garis, maka algoritma Dijkstra akan melakukan perhitungan terhadap semua garis dengan bobot terkecil dari setiap titik (Martin Nugroho Parapat, Deddy Kusianto, 2017).



Gambar 3. Algoritma Dijkstra

Dalam tahapan algoritma Dijkstra ini dapat dilakukan dengan tahapan berikut:

1. Tentukan titik awal dimana titik tersebut akan menjadi node pertama atau node awal, lalu tentukan bobot jarak pada node awal ke node terdekat satu demi satu, Dijkstra akan melakukan developing untuk mencari bobot terkecil dari satu node ke node lainnya dan ke node selanjutnya satu per satu.
2. Tentukan bobot (jarak) dari setiap node ke node lainnya, kemudian set nilai 0 pada node pertama dan nilai tak terhingga pada node lainnya.
3. Setting semua node yang belum dilalui dan setting node awal sebagai "node keberangkatan"
4. Pada node keberangkatan, perhitungkan node lainnya yang paling dekat dengan

keberangkatan yang belum dilalui dan perhitungkan jarak dari awal keberangkatan. Jika jaraknya lebih kecil dari jarak sebelumnya (yang telah diperhitungkan sebelumnya) hapus data yang lama simpan ulang data jarak sebelumnya dengan data jarak yang lebih pendek.

5. Setelah kita selesai mengitung dan memper-timbangkan setiap jarak pada node lainnya yang berdekatan, tandai node yang sudah dilalui sebagai “Node dilewati”. Node yang sudah dilewati tidak akan pernah dicek kembali, jarak yang akan disimpan adalah jarak dengan bobot paling kecil bobotnya,
6. Setting “Node belum dilewati”, dengan bobot jarak paling kecil (dari node keberangkatan) sebagai “Node Keberangkatan” untuk keberangkatan selanjutnya dan ulangi langkah.

HASIL DAN PEMBAHASAN

Pada bagian ini berisi analisa, hasil serta pembahasan dari topik penelitian, yang bisa di buat terlebih dahulu metodologi penelitian. Bagian ini juga merepresentasikan penjelasan yang berupa penjelasan, gambar, tabel dan lainnya.

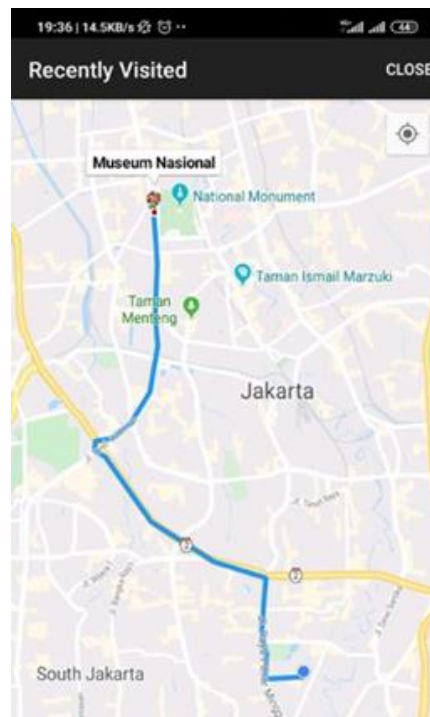
A. Analisa Kebutuhan Sistem

Pada perancangan aplikasi rute museum, memerlukan perangkat pendukung berupa perangkat keras (*hardware*) maupun perangkat lunak (*software*), yang terdiri dari:

1. Kebutuhan hardware: Laptop AMD A6-5200 APU (4) @ 2.000GHz 64 bit, RAM 6GB.
2. Kebutuhan software: OS Ubuntu 18.04 LTS, Android Studio 3.5.2, 64 bit, Draw.io dan Framework flutter v1.9.1+hotfix.2.

B. Tampilan Aplikasi Rute Museum

Penelitian ini dilakukan dengan merancang UI pada aplikasi android menggunakan software Android Studio 3.5.2, 64bit serta Framework flutter v1.9.1+hotfix.2. Perhatikan Gambar 4.



Gambar 4. Tampilan Rute

Gambar 4 merupakan rute menuju ke museum nasional yang akan tampil ketika memilih museum yang akan dituju kearah museum.

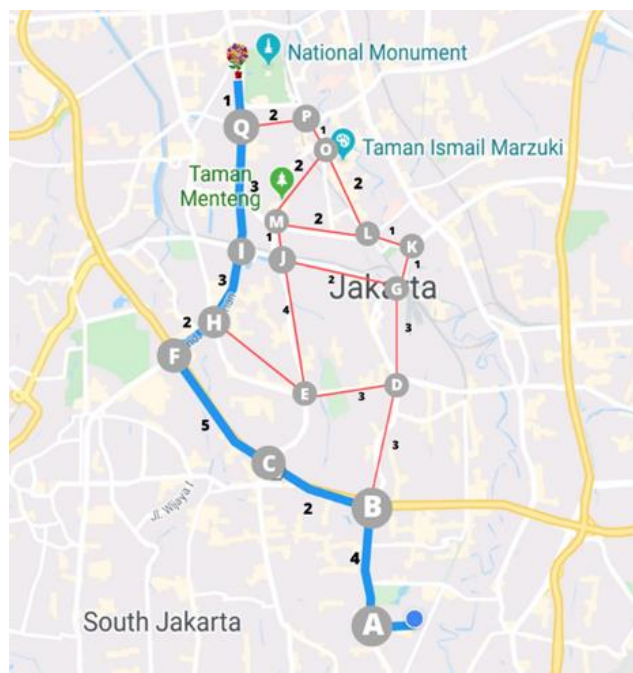
C. Pengujian

Pengujian dilakukan pada Museum Nasional yang ada di Jakarta pusat pada beberapa device yang berbeda.

Tabel 1. Hasil Pengujian Perangkat

Versi	Android	Keterangan
4.0.1	Ice Cream Sandwich	Tidak Berhasil
4.2.2	Jelly Bean	Berhasil
5.0.0	Lollipop	Berhasil
6.1.2	Marshmallow	Berhasil
7.0.1	Nougat	Berhasil
8.0.0	Oreo	Berhasil
9.0.0	Pie	Berhasil

Pada Tabel 1 adalah hasil pengujian versi android dari versi 4.0.1 sampai versi android 9.0.0. Terdapat keterangan “Tidak Berhasil” pada versi android 4.0.1 karena versi minimum yang bisa dijalankan oleh Framework Flutter adalah 4.2.2. Pengujian selanjutnya dilakukan dengan ilustrasi node atau titik-titik serta bobot yang digunakan pada algoritma dijkstra dalam pengambilan rute terdekat dari suatu rumah di daerah jakarta selatan ke arah Museum Nasional.



Gambar 5. Penggambaran Node Algoritma Dijkstra

Pada Gambar 5 merupakan ilustrasi berupa node atau titik serta bobot pada tiap node. Hasil pengujian node pada suatu rumah di daerah jakarta selatan ke Museum Nasional dapat dilihat pada tabel pengujian berikut.

Tabel 2. Hasil Pengujian Algoritma Dijkstra

Langkah	Pilih	Terkecil	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	q
1	a	0	0a	4a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a
2	b	4b	0a	4a	2b	3b	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a
3	c	6c	0a	4a	6b	3b	∞a	5c	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a
4	f	11f	0a	4a	6b	3b	∞a	11c	∞a	2f	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a
5	h	2h	0a	4a	6b	3b	3h	11c	∞a	13f	3h	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a
6	i	3i	0a	4a	6b	3b	3h	11c	∞a	13f	16h	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	4i
7	q (stop)	4q	0a	4a	6b	3b	3h	11c	∞a	13f	16h	∞a	∞a	∞a	∞a	∞a	∞a	∞a	∞a	20i

Pada tabel pengujian tersebut dapat disimpulkan rute terpendek yang dihitung dengan algoritma dijkstra adalah rute A-B-C-F-H-I-Q. Dengan total bobot 20 dimana bobot ini adalah bobot jarak yang paling pendek dibandingkan dengan jarak-jarak dari node atau titik lainnya.

D. Perbandingan

Untuk membuktikan hasil dari algoritma dijkstra telah dilakukan perbandingan dengan algoritma greedy dalam pencarian rute.

Tabel 3. Hasil Pengujian Algoritma Greedy

Langkah	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	Museum Nasional	Titik Yang akan dilalui
A	-	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A
AB	-	-	2	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	B
BC	-	-	-	-	-	5	-	-	-	-	-	-	-	-	-	-	-	-	F
CF	-	-	-	-	-	-	-	2	-	-	-	-	-	-	-	-	-	-	H
FH	-	-	-	-	4	-	-	-	3	-	-	-	-	-	-	-	-	-	I
HI	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3	-	Q
IQ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	-	1	Museum
Selesai	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Selesai

Setelah melakukan perbandingan dengan algoritma greedy mendapatkan alur yang sama yaitu A-B-C-F-H-I-Q untuk mencapai museum. Hasil ini didapatkan karna percabangan dan alur yang dilalui sama. Sehingga menghasilkan rute yang sama. Meskipun secara teori algoritma dijkstra memberikan rute yang lebih pendek dibandingkan algoritma greedy dimana algoritma itu sendiri memilih rute yang optimum dan bukan rute terpendek, oleh karna itu algoritma dijkstra merupakan algoritma terpendek dalam menentukan rute terpendek.

SIMPULAN DAN SARAN

Berdasarkan pada pembahasan yang telah dilakukan pada bab sebelumnya dapat disimpulkan bahwa algoritma yang digunakan merupakan solusi dalam mencari rute terpendek, dengan metode algoritma Dijkstra yang diterapkan dalam smartphone pengguna dapat mempersingkat efektifitas waktu untuk mencari museum di Jakarta. Rute yang dihasilkan pada algoritma ini sangat efektif bila dilalui dengan mengendarakan mobil dengan mengesampingkan kemacetan dan kondisi ganjil genap di Jakarta. Berdasarkan pengujian yang dilakukan, penerapan algoritma Dijkstra untuk mendapatkan rute terpendek dinilai efektif sebab dengan total 20 bobot hanya memerlukan 7 bobot, yang mana hal tersebut menyatakan hanya perlu 35% bobot untuk mencapai Museum Nasional.

DAFTAR PUSTAKA

[1] Aprilianingsih, E. P., Primananda, R., & Suharsono, A. (2017). Analisis Fail Path Pada Arsitektur Software Defined Network Menggunakan Dijkstra Algorithm. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 1(3), 174–183. Retrieved from <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/59>

[2] Candra Galih, E., & Andriyat Krisdiawan, R. (2018). Implementasi Algoritma Dijkstra Pada Aplikasi Wisata Kuningan Berbasis Android. *Jurnal Nuansa Informatika*, 53(95), 45–52. <https://doi.org/10.1017/CBO9781107415324.004>

[3] Ginting, J. V., Barus, E. S., Informatika, M., Informasi, S., Utara, S., & Dijkstra, A. (2018). *Aplikasi Penentuan Rute Rumah Sakit Terdekat Menggunakan Algoritma Dijkstra*. 2(2), 1–8.

- [4] Junanda, B., Kurniadi, D., & Huda, Y. (2016). Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra pada Sistem Informasi Geografis Pemetaan Stasiun Pengisian Bahan Bakar Umum. *Jurnal Vokasional Teknik Elektronika & Informatika*, 4(1), 1–8.
- [5] Kusuma, H. B. (2018). Wayfinding Sign pada Ruang Pameran Tetap di Museum Nasional Indonesia – Jakarta. *Mudra Jurnal Seni Budaya*, 33(2), 242. <https://doi.org/10.31091/mudra.v33i2.331>
- [6] Martin Nugroho 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.
- [7] 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. <https://doi.org/10.14710/jtsiskom.3.2.2015.311-319>
- [8] 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.
- [9] Retnani, W. E. Y., Istiadi, D., & Roqib, A. (2015). Pencarian SPBU Terdekat dan Penentuan Jarak Terpendek Menggunakan Algoritma DIJKSTRA (Studi Kasus di Kabupaten Jember). *Jurnal Nasional Teknik Elektro*, 4(1), 89. <https://doi.org/10.25077/jnte.v4n1.132.2015>
- [10] Triansyah, A., & Informatics, T. (2013). *Implementasi Algoritma Dijkstra Dalam Aplikasi Untuk Menentukan Lintasan Terpendek Jalan Darat Antar Kota Di Sumatera Bagian Selatan*. 5(2), 611–621.
- [11] Yusuf, M. S., Az-zahra, H. M., & Apriyanti, D. H. (2017). *Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi)*. 1(12), 1779–1787.