

## Rancangan Sistem *E-itinerary* Wisata Di Kawasan Jakarta Utara Dengan Penerapan Algoritma Dijkstra

<sup>1</sup>Destriana Widyaningrum, <sup>2</sup>Chyquitha Danuputri

<sup>1</sup>Fakultas Teknologi dan Desain, Universitas Bunda Mulia (UBM), Jl. Lodan Raya No.2 Jakarta Utara 14430 Telp (021) 6909090

<sup>2</sup> Fakultas Teknologi dan Desain, Universitas Bunda Mulia (UBM), Jl. Lodan Raya No.2 Jakarta Utara 14430 Telp (021) 6909090  
e-mail: des3ana@gmail.com

Received: 22 Februari 2022

Accepted: 08 Februari 2023

### Abstract

*Tourism is one of the most important sectors to encourage the growth of a country. This is related to the economic assistance of the surrounding community with the presence of tourist attractions. North Jakarta is one area that has quite a lot of tourist attractions, besides its location near the ocean, there are many historical buildings from the Dutch heritage that attract tourists to visit. The distance from one tourist location to another is not too far, but the breadth of tourist areas in the area is not enough to be visited in just one day. For tourists out of town and abroad will determine the effective time in determining which tours will be visited first. To support the tourism sector, a system will be designed that can provide recommendations for travel plans by sorting the shortest distances between tourist sites, to optimize visit times at several tourist attractions in North Jakarta. The design of this system was developed on a website platform, with the name e-itinerary. In this research, solving the problem of finding the shortest track from one place to another applies the djikstra algorithm. The results of the research, the user will enter several tourist locations to be visited on the e-itinerary, then by applying the djikstra algorithm the system will look for the path that has the minimum length, from vertex a to vertex z in a weighted graph, the value of the weight is a positive number. The weights used are obtained in the form of travel time on each road segment, the data source from google maps that has been synchronized to the e-itinerary system by embedding the API key. Location data will be presented on the system platform of recommendations for the sequence of tourist sites from the beginning to the end visited.*

**Keywords:** *tour, distance, e-itinerary, djikstra algorithm, API key*

### Abstrak

Pariwisata adalah salah satu sektor yang terpenting untuk mendorong pertumbuhan suatu negara. Hal ini berkaitan dengan terbantunya ekonomi masyarakat sekitar dengan adanya tempat wisata. Jakarta Utara salah satu daerah yang memiliki cukup banyak tempat wisata, selain lokasinya yang berada di dekat lautan, banyak bangun sejarah peninggalan penjajahan Belanda yang menarik wisatawan untuk berkunjung. Jarak antara lokasi wisata satu dengan lainnya pun tidak terlalu jauh, namun luasnya daerah wisata di kawasan tersebut tidak cukup di kunjungi hanya dalam satu hari. Bagi wisatawan luar kota maupun luar negeri akan kesulitan menentukan waktu yang efektif dalam menentukan wisata mana yang akan dikunjungi terlebih dahulu. Dalam mendukung sektor pariwisata, akan dirancang sistem yang dapat memberikan rekomendasi rencana perjalanan wisata dengan mengurutkan jarak paling terdekat antar lokasi wisata, untuk mengoptimalkan waktu kunjungan di beberapa tempat wisata di Jakarta Utara. Rancangan sistem ini dikembangkan dalam platform website, dengan nama *e-itinerary*. Pada penelitian ini, penyelesaian masalah pencarian lintasan terpendek dari suatu tempat ke tempat lain menerapkan algoritma djikstra. Hasil dari penelitian, pengguna akan memasukkan beberapa lokasi wisata yang akan dikunjungi pada *e-itinerary*, kemudian dengan penerapan algoritma djikstra sistem akan mencari lintasan yang memiliki panjang paling minimum,

dari vertex a ke vertex z dalam graf berbobot, nilai bobot tersebut adalah bilangan positif. Bobot yang digunakan diperoleh dalam bentuk waktu tempuh pada masing-masing ruas jalan, sumber data dari google maps yang sudah di sinkronasi ke sistem *e-itinerary* dengan menyematkan API key. Data titik lokasi akan disajikan pada tampilan platform sistem berupa hasil rekomendasi urutan lokasi wisata yang paling awal sampai akhir dikunjungi.

**Kata Kunci:** wisata, jarak, *e-itinerary*, algoritma djikstra, API key

## PENDAHULUAN

Pariwisata Indonesia saat ini menjadi perhatian khusus pemerintah, ditengah krisis perdagangan dunia yang berdampak pada turunnya ekspor Indonesia. Sektor pariwisata memberikan peluang sebagai andalan perolehan devisa, dalam mendukung pertumbuhan suatu negara. Hal ini terkait terbantunya ekonomi masyarakat sekitar daerah wisata. Dimana sektor industri kreatif, kerajinan, kuliner maupun perdagangan lainnya dapat berkembang dengan banyaknya kunjungan wisatawan. Menurut I Gusti Bagus, pariwisata adalah kegiatan yang bertujuan menyelenggarakan jasa, menyediakan, mengusahakan objek dan daya tarik wisata, usaha sarana pariwisata dan usaha lain yang terkait di bidang tersebut.

Lokasi wisata yang paling banyak dikunjungi di ibukota DKI Jakarta adalah kawasan Jakarta Utara, karena letaknya yang berada dekat pesisir lautan dan banyak bangunan sejarah disana. Hal ini diperkuat dengan data dari badan pusat statistik pada Tabel 1, terkait jumlah kunjungan obyek wisata di Jakarta Utara meningkat dan dikunjungi lebih dari 10 juta wisatawan per tahunnya. Lokasi wisata yang paling terkenal adalah taman impian jaya ancol, pelabuhan sunda kelapa, dan terdapat pula bangunan bersejarah peninggalan Belanda dengan arsitektur bangunannya masih asli menjadi salah satu daya tarik untuk dikunjungi, terutama wisatawan asing. Wisatawan yang datang ke daerah wisata baru, tentunya hanya mengetahui nama tempat wisata yang akan dikunjungi, mereka tidak tahu rute menuju lokasi, jarak antar lokasi wisata maupun dari tempat mereka menginap. Seringkali, tidak semua lokasi wisata yang sudah direncanakan dikunjungi, karena lamanya tinggal tidak sesuai dengan lama perjalanan dari lokasi wisata satu ke tempat lainnya. Jarak untuk menuju daerah wisata menjadi pertimbangan tersendiri dari sekian banyak jalan yang tersedia dan dapat dilintasi. Wisatawan membutuhkan rute terpendek untuk mencapai tempat wisata yang dituju guna mempersingkat waktu perjalanan. Sehingga wisatawan dapat menghemat waktu mengunjungi beberapa lokasi wisata sekaligus dalam 1 hari.

Tabel 1. Jumlah Kunjungan Wisata ke Obyek Wisata Di Jakarta Utara 2016-2018[1]

Obyek Wisata	Jumlah Kunjungan Wisatawan ke Obyek Wisata di Jakarta Utara		
	2016	2017	2018
Pelabuhan Sunda Kelapa	43 398,00	58 284,00	45 653,00
Taman Impian Jaya Ancol	17 850 284,00	19 261 212,00	17 575 914,00

Solusi pencarian rute terpendek, dapat diselesaikan dengan pemodelan graf menggunakan algoritma djikstra. Algoritma djikstra menggunakan prinsip *greedy*, yaitu

mencari jalur terpendek dari satu titik (*vertex*) ke titik lainnya yang terhubung. Pemilihan algoritma djikstra dapat menyelesaikan permasalahan rute terpendek dengan satu sumber asal untuk suatu graf dengan nilai sisi, non negatif, menghasilkan pohon jalur terpendek. Vertek-vertik diambil dari beberapa tempat strategis yang dapat dijadikan acuan dan mudah untuk diingat oleh para wisatawan [2].

Algoritma djikstra ini diterapkan pada sebuah aplikasi *location based service* dengan *platform* website yang memanfaatkan Google Map sebagai peta navigasinya. Pada penelitian ini, aplikasi akan mencari rute terpendek ke beberapa tempat wisata di daerah yang telah ditentukan dan nama tempat wisata yang telah dipilih pengguna, dengan menggunakan algoritma djikstra. Sehingga diharapkan para wisatawan mendapatkan agenda perjalanan wisata yang mudah dan cepat untuk mencapai tempat wisata yang ingin dituju melalui informasi rute dan saran lokasi wisata mana yang harus lebih dulu dikunjungi.

### METODE PENELITIAN

Metode perancangan penelitian ini menggunakan metode *waterfall*, terdapat 5 langkah dalam melakukan perancangan rekayasa perangkat lunak. Adapun metode pengembangan sistem, sebagai berikut :

#### 1. *Requirement*

Tahap pertama adalah analisis kebutuhan dengan mengumpulkan data terkait pengembangan sistem. Studi literatur mengenai algoritma djikstra yang akan digunakan untuk mencari rute terdekat. Mencari informasi bagaimana mengakses google APIs. Persiapan pengembangan website menggunakan HTML, PHP dan MySql.

#### 2. *Design*

Rencana pada tahap ini akan menghasilkan suatu peta untuk memandu pengerjaan program dan membuat syarat dan ketentuan untuk sebuah aplikasi sebelum dibuat. Proses ini akan berfokus pada kerangka seperti interface layout, algoritma, dan struktur data.

#### 3. *Implementation*

Perancangan sistem dengan implementasi algoritma djikstra, dengan menerjemahkan berupa kode program pada platform website dan terkoneksi langsung dengan google *maps* yang sudah disematkan menggunakan google APIs.

#### 4. *Verification*

Setelah sistem siap, dilakukan testing secara *white box* untuk menguji algoritma dan fungsi-fungsi pada sistem berjalan sesuai logika dan menguji secara *black box* oleh pengguna, apakah *fitur* pada aplikasi sudah sesuai dengan konsep.

#### 5. *Maintenance*

Pada tahap akhir dilakukan pengoperasian di lingkungan terhadap perangkat lunak oleh pengguna, apakah sudah sesuai dengan tujuan sistem, kemudian melakukan pemeliharaan, seperti penyesuaian.

### A. Algoritma Dijkstra

Algoritma ini diberi nama sesuai nama penemunya, Edsger Wybe Dijkstra. Algoritma *Dijkstra* mencari lintasan terpendek dalam sejumlah langkah. Algoritma ini menggunakan prinsip *Greedy* yang menyatakan bahwa pada setiap langkah kita memilih sisi yang berbobot minimum dan memasukkannya ke dalam himpunan solusi. Input algoritma ini adalah sebuah *graph* berarah yang berbobot (*weighted directed graph*)  $G$  dan sebuah sumber *vertex*  $s$  dalam  $G$  dan  $V$  adalah himpunan semua *vertices* dalam *graph*  $G$ . Berikut langkah-langkah dalam menentukan path pada algoritma djikstra [3]:

1. Tentukan titik mana yang akan menjadi node awal, lalu beri bobot jarak pada node pertama ke node terdekat satu per satu.
2. Beri nilai bobot (jarak) untuk setiap titik ke titik lainnya, lalu set nilai 0 pada node awal dan nilai tak hingga terhadap node lain (belum terisi) 2.
3. Set semua node yang belum dilalui dan set node awal sebagai "Node keberangkatan"
4. Dari node keberangkatan, pertimbangkan node tetangga yang belum dilalui dan hitung jaraknya dari titik keberangkatan. Jika jarak ini lebih kecil dari jarak sebelumnya (yang telah terekam sebelumnya) hapus data lama, simpan ulang data jarak dengan jarak yang baru
5. Saat kita selesai mempertimbangkan setiap jarak terhadap node tetangga, tandai node yang telah dilalui sebagai "Node dilewati". Node yang dilewati tidak akan pernah di cek kembali, jarak yang disimpan adalah jarak terakhir dan yang paling minimal bobotnya.
6. Set "Node belum dilewati" dengan jarak terkecil (dari node keberangkatan) sebagai "Node Keberangkatan" selanjutnya dan ulangi langkah 5.

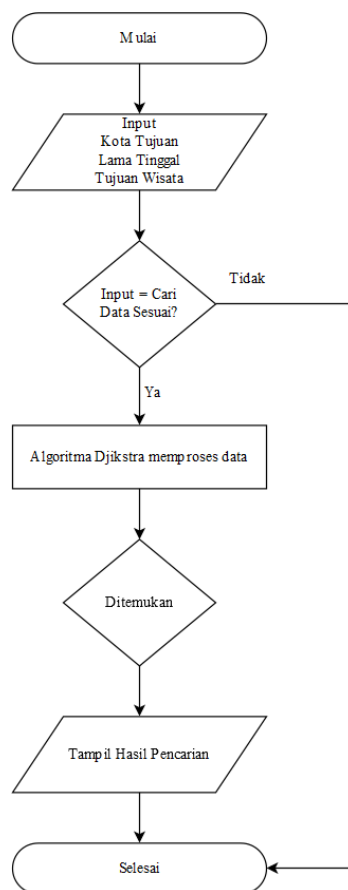
### B. Google APIs

Google Maps adalah layanan gratis yang diberikan oleh Google dan sangat populer. Google Maps adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, Google Maps merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur Google Maps dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan Google Maps Api. Google Maps API adalah suatu library yang berbentuk JavaScript [4]. Berikut adalah cara mendapatkan *key* google APIs [5] :

1. Masuklah ke *Google Developers Console*.
2. Pilih *select a project, or create a new one*.
3. Pada menu sisi sebelah kiri, select *Credentials*.
4. Jika proyek Anda belum memiliki kunci API *Server*, buatlah sekarang dengan memilih *Add credentials > API key > Server key*.
5. Untuk mencegah pencurian kuota, batasi kunci Anda agar permintaan hanya diperbolehkan dari alamat IP sumber milik server Anda.

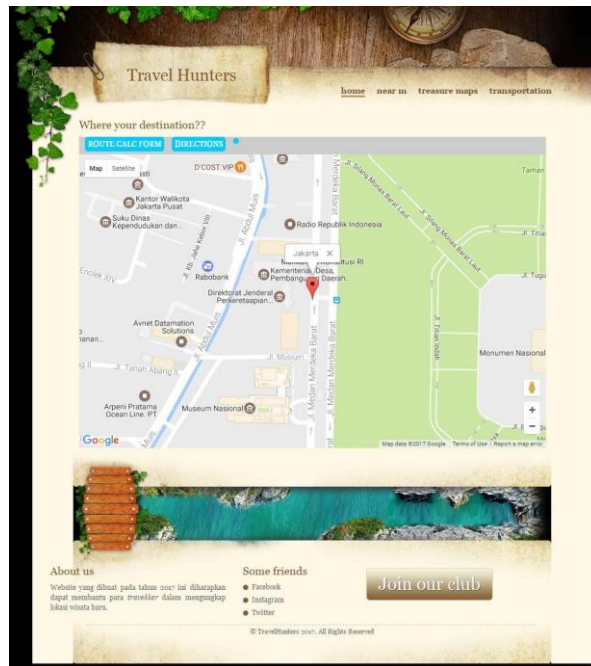
### HASIL DAN PEMBAHASAN

Pengembangan menggunakan metode algoritma djikstra sebagai *logic* digunakan untuk menentukan rute terdekat antar lokasi wisata. Proses ini dilakukan setelah pengguna memasukkan data yang diminta, seperti tujuan wisata atau lokasi saat ini, struktur alur sistem pada Gambar 1, merupakan sistem yang akan berjalan pada *platform* website, dimana logika algoritma djikstra bekerja jika adanya daya yang di input.



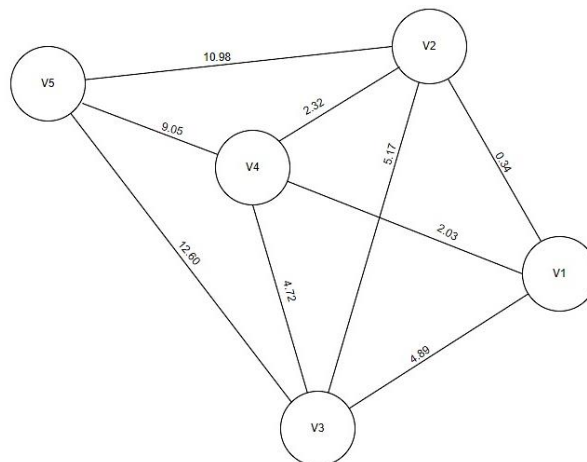
Gambar 1. *Flowcart Sistem*

Pada halaman utama disajikan pada Gambar 2, pengguna dapat menentukan lokasi wisata yang akan dikunjungi dan mendapatkan informasi rute perjalanan menuju lokasi wisata. Koneksi google Api dalam perancangan ini, menampilkan map dan memudahkan pengguna melihat langsung titik-titik lokasi wisata yang dipilih. Setelah pengguna melakukan *input* lokasi, akan dilakukan proses *optimized waypoint* google Api untuk menentukan *distance matrix* yang hasilnya menyajikan jarak antar titik-titik lokasi wisata yang dipilih dan menginformasikan rute perjalanan menuju lokasi. Jarak antar lokasi yang sudah di ketahui dan telah membentuk graf matrik akan dibandingkan jarak antar titik dengan proses algoritma djikstra.



Gambar 2. Tampilan halaman utama *input data*

Pengguna memasukan titik lokasi saat ini dan memilih titik lokasi yang akan dikunjungi, yang didefinisikan sebagai vertex (V). Pada penelitian ini, pengguna pada titik awal berada di Hotel Ibis Style (V1) kemudian dimasukkan 4 titik lokasi tujuan wisata di Jakarta Utara, yaitu Taman Impian Jaya Ancol (V2) , Kampung Nelayan Cilincing (V3), Museum Fatahillah (V4), Wisata Alam Mangrove (V5). Tiap titik akan diberi nilai bobot sesuai dengan jarak antar titik ke titik lainnya, digambarkan pada Gambar 3 berupa graf antara jarak dari V1 ke V5.



Gambar 3. Graf Penelitian Algoritma Dijkstra



Menerapkan langkah-langkah pada algoritma djikstra dengan mendefinisikan sebagai logika kode program, dapat dihitung jika jarak lebih kecil dari jarak sebelumnya (yang telah terekam sebelumnya), akan dihapus data lama, simpan ulang data jarak dengan jarak yang baru. Pada Tabel 2 merupakan hasil dari mengikuti langkah perhitungan logika algoritma djikstra[6].

Tabel 2. Perhitungan Algoritma Dijkstra dari V1 ke V5

Iteration	Unvisited (Q)	Visited (S)	Current	V1	V2	V3	V4	V5
	Initialization {V1,V2,V3,V4,V5}	{-}						
1	{V2,V3,V4,V5}	{V1}	V1		0.34	4.89	2.03	$\infty$
2	{V3,V4,V5}	{V1,V2}	V2			5.51	2.66	11.32
3	{V4,V5}	{V1,V2,V3}	V3				10.23	18.11
4	{V5}	{V1,V2,V3,V4}	V4					19.28

Dengan demikian pada penelitian ini dapat terlihat bahwa jarak terpendek dari V1 ke V5 adalah 19.28 dengan jalur V1 – V2 – V3 – V4 – V5. Hasil rekomendasi pada halaman website disajikan pada Gambar 3, diletakkan dibawah data titik yang diinput oleh pengguna. Jalur rekomendasi ini, diasumsikan jika pengguna menggunakan kendaraan mobil dan waktu perjalanan normal tanpa hambatan (kemacetan).



Gambar 3. Hasil Rekomendasi Pada Website

### **SIMPULAN DAN SARAN**

Pencarian rute terpendek merupakan salah satu persoalan dalam teori graf. Persoalan mencari lintasan terpendek dari setiap graf berbobot, merupakan masalah optimasi. Pada algoritma djikstra, graf berbobot dengan syarat bobot dari masing-masing sisi haruslah bernilai positif ( $\geq 0$ ). Dalam mencari lintasan terpendek algoritma djikstra memberikan hasil yang efisien dan tidak membutuhkan waktu banyak. Namun dalam prosesnya algoritma djikstra hanya membaca rute jalan besar atau yang dapat dilalui mobil.

Pegembangan sistem pada penelitian selanjutnya, dapat di implementasikan menggunakan aplikasi *mobile* dan disertakan dengan teknologi GPS sebagai pemandu rute perjalanan.

### **DAFTAR PUSTAKA**

- [1] Badan Pusat Statistik Jakarta Utara, "Jumlah Kunjungan Wisatawan ke Obyek Wisata di Jakarta Utara 2016-2018," 2022.  
<https://jakutkota.bps.go.id/indicator/16/123/1/jumlah-kunjungan-wisatawan-ke-obyek-wisata-di-jakarta-utara.html>.
- [2] A. B. Setiawan and Y. Nurhidayati, "SISTEM JALUR E-LAUDRY MENGGUNAKAN," pp. 271–276, 2017.
- [3] N. Azizah and D. Mahendra, "Geolocation dengan Metode Dijkstra untuk Menentukan Jalur Terpendek Lokasi Peribadatan," vol. 02, pp. 96–103, 2017.
- [4] A. L. Hananto *et al.*, "Penerapan Algoritma Dijkstra Pada Sistem Monitoring Petugas Lapangan Pemkab Bekasi Berbasis Android," vol. 4, no. 3, 2019.
- [5] D. API, "API."  
<https://developers.google.com/maps/documentation/places/web-service/search?hl=ID>.
- [6] A. Ratnasari, F. Ardiani, and F. N. A, "Penentuan Jarak Terpendek dan Jarak Terpendek Alternatif Menggunakan Algoritma Dijkstra Serta Estimasi Waktu Tempuh," vol. 2013, no. November, pp. 29–34, 2013.