

IMPLEMENTASI ALGORITMA DIJKSTRA DALAM PERANCANGAN APLIKASI PENENTUAN RUTE TERPENDEK PADA OBJEK PARIWISATA DANAU TOBA DAN SEKITARNYA

Mahrizal Masri¹⁾, Andika Pratomo Kiswanto²⁾, Budhi Santri Kusuma³⁾

¹⁾Dosen ITM, ²⁾Alumni ITM, ³⁾Dosen UMA

Abstrak

Algoritma Dijkstra merupakan salah satu algoritma yang dapat menyelesaikan permasalahan pencarian rute terpendek yang merupakan permasalahan dalam teori graf. Implementasi sistem ini akan menghasilkan berupa aplikasi yang berbentuk web, dengan script PHP dan MySQL sebagai pengelola basis datanya. Sistem ini menggunakan Algoritma Dijkstra. Sistem ini sangat berguna untuk pengunjung karena wisatawan bisa mengetahui rute terpendek yang bisa ditempuh dari dan menuju suatu tempat wisata yang diinginkan di daerah objek pariwisata Danau Toba dan Sekitarnya dengan menggunakan sistem ini. Dengan adanya penelitian rute terpendek dengan menggunakan Algoritma Dijkstra ini para wisatawan diharapkan dapat menambah informasi mengenai rute tercepat dan akhirnya para wisatawan memilih rute yang terbaik yang dapat mengefisienkan semua biaya dan mengetahui semua informasi objek wisata yang ada di Danau Toba.

Kata-Kata Kunci : Implementasi, Graf, Algoritma Dijkstra, Pencarian Rute Terpendek, Pariwisata

I. PENDAHULUAN

Sumatera Utara merupakan salah satu provinsi di Indonesia yang dikenal dengan banyaknya objek wisata yang tersebar di banyak kota dan kabupatennya. Salah satunya adalah Danau Toba yang sudah terkenal keindahan alamnya sampai ke Negara lain.

Karena banyaknya tempat wisata yang ada di sekitaran danau Toba maka wisatawan ingin mengetahui informasi yang jelas dimana lokasi tempat wisata yang ada serta jalur yang harus ditempuh untuk menuju lokasi tempat wisata yang dimaksud.

Pencarian jalur terpendek merupakan suatu permasalahan untuk menemukan sebuah jalur antara dua node dengan jumlah bobot minimal. Pada kasus pencarian jalur tercepat antara dua lokasi pada peta untuk melakukan perjalanan antara dua lokasi tersebut. Salah satu metode yang digunakan untuk menyelesaikan permasalahan pencarian jalur terpendek dengan menggunakan algoritma Dijkstra. Algoritma Dijkstra akan mencari jalur terpendek mulai dari node awal sampai node tujuan. Algoritma ini akan membandingkan bobot terkecil dari node awal sampai ke node tujuan untuk menemukan jalur paling efisien.

Algoritma Dijkstra adalah algoritma untuk pencarian jalur (*path*) yang termurah dari suatu *vertex* awal ke *vertex* akhir. Algoritma ini berbasis pada teknik *greedy*, dimana digunakan sebuah himpunan *vertex* S yang mula-mula diisi dengan *vertex* awal, pada setiap langkah *vertex* v berikutnya yang memiliki bobot terkecil atau biaya termurah, ditambahkan sebuah array D biasa digunakan untuk merekam

II. TINJAUAN PUSTAKA

2.1 Pengertian Pariwisata

Pariwisata adalah suatu aktivitas yang kompleks yang dapat dipandang sebagai suatu sistem yang besar, yang mempunyai berbagai komponen seperti ekonomi, ekologi, politik, sosial, budaya dan seterusnya. Melihat pariwisata sebagai sebuah sistem, berarti analisis mengenai berbagai aspek kepariwisataan tidak bisa dilepaskan dari subsistem yang lain, seperti politik, sosial ekonomi, budaya dan seterusnya, dalam hubungan saling ketergantungan dan saling terkait (*interconnect tedness*). Sebagai sebuah sistem, antar komponen dalam sistem tersebut terjadi hubungan interdependensi, yang berarti bahwa perubahan pada salah satu subsistem akan menyebabkan juga terjadinya perubahan pada subsistem yang lainnya, sampai akhirnya kembali ditemukan harmoni yang baru. Pariwisata adalah sistem dari berbagai elemen yang tersusun seperti sarang laba-laba : “ *like a spider’s web- touch one part of it and reverberations will be felt throughout*” (Fennel, 1999), seperti yang terdapat dalam jurnal *Sudana. 2013 Strategi Pengembangan Dewa Wisata Ekologis di Desa Belimbing, Kecamatan Puputan Kabupaten Tabanan, 13 (1), 12.*

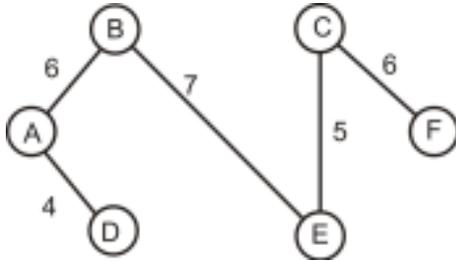
2.2 Algoritma Dijkstra

Algoritma Dijkstra (minimum dari path dari *graph* berbobot) adalah algoritma untuk pencarian jalur (*path*) yang termurah dari suatu *vertex* awal ke *vertex* akhir. Algoritma ini berbasis pada teknik *greedy*, dimana digunakan sebuah himpunan *vertex* S yang mula-mula diisi dengan *vertex* awal pada setiap langkah *vertex* v berikutnya yang memiliki bobot terkecil ditambahkan. Sebuah array D biasa digunakan untuk merekam jalur terpendek dari *vertex* awal ke *vertex* akhir (Suarga 2012:236)

Ada beberapa kasus pencarian lintasan terpendek yang diselesaikan menggunakan algoritma Djikstra, yaitu:

1. Pencarian lintasan terpendek antara dua buah simpul tertentu (*a pair shortest path*).
2. Pencarian lintasan terpendek dari simpul tertentu ke semua simpul yang lain (*single source shortest path*).

Pencarian lintasan terpendek antara dua buah simpul yang melalui beberapa simpul tertentu (*intermediate shortest path*)



Gambar 1. Hasil Graph Berbobot

Sumber: Dr.Suarga (*Algoritma dan Pemrograman*, 2012:238)

2.3 Pengenalan UML (*unfied Modeling Language*)

Menurut Adi Nugroho (2009:7) dalam konteks UML,tahapan konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh pengguna. Selanjutnya *use case diagram* tidak hanya sangat penting pada tahap analisis, tetapi juga sangat penting untuk perancangan (*design*), untuk mencari (mencoba menemukan) kelas-kelas yang terlibat dalam aplikasi, dan untuk melakukan pengujian (*testing*).

2.4 Pengenalan Google Maps

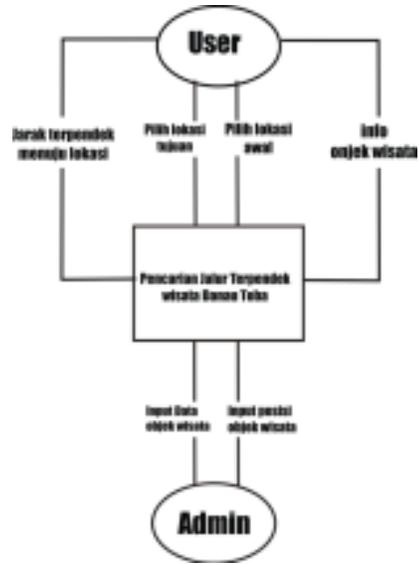
Google Map Service adalah sebuah jasa peta global *virtual* gratis dan *online* yang disediakan oleh perusahaan *Google*. *Google Maps* yang dapat ditemukan di alamat <http://maps.google.com>.*Google Maps* menawarkan peta yang dapat diseret dan gambar satelit untuk seluruh dunia.*Google Maps* juga menawarkan pencarian suatu tempat dan rute perjalanan.

Google Maps API merupakan sebuah layanan (*service*) yang diberikan oleh *Google* kepada para pengguna untuk memanfaatkan *Google Map* dalam mengembangkan aplikasi. *Google Maps API* menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis *services* yang dimiliki, serta mengizinkan kepada pengguna untuk membangun aplikasi *enterprise* di dalam websitenya.

III. METODE PENELITIAN

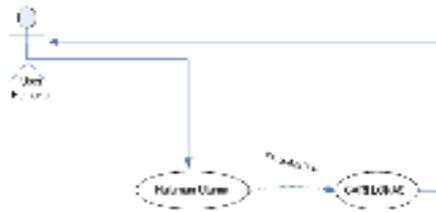
3.1. Data Flow Diagram (DFD)

DFD adalah gambaran grafis dari suatu sistem yang menggunakan sejumlah bentuk simbol yang menggambarkan bagaimana data mengalir dari suatu proses yang saling berkaitan.



Gambar 2. Diagram Konteks

3.2 Use Case Diagram user



Gambar 3. Use Case Diagram

Use Case Diagram User akan menguraikan proses user untuk melakukan proses pencarian jalur terpendek objek wisata di danau toba, Dimana proses pencarian jalur terpendek terdapat pada menu pencarian jalur.

Pada proses *Use Case Diagram User* user dapat melakukan aksi sebagai berikut :

1. *User* dapat melakukan pencarian jalur dengan memasukkan posisi awal dan tujuan yang telah di tentukan oleh sistem.
2. *User* dapat melihat data data objek wisata yang telah di masukkan oleh admin.

3.3 Desain Database

Database adalah representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Tujuan dari perancangan *database server* pada aplikasi ini adalah untuk mengatasi ketebatasan memori pada perangkat *mobile*. Disamping itu *database* yang diletakkan pada *web server* ditujukan untuk pengembangan/penambahan kata oleh admin. *Database* yang dirancang pada *web server* menggunakan *MYSQL*.

Berikut ini nama-nama tabel yang digunakan beserta *field-field* yang terdapat pada masing-masing table. Pada setiap tabel dibawah hanya admin yang

dapat melakukan *update*(menambah, mengedit dan menghapus *record*).

1. Tabel informasi

Tabel lokasi adalah tabel yang digunakan untuk menyimpan data informasi objek wisata di sekitaran danau toba. Tabel ini mempunyai struktur seperti terlihat pada Tabel 1 berikut :

Tabel 1. Tabel informasi

Nama Field	Tipe	Panjang	Keterangan
Id_info	Int	10	Id info
Id_jalan	Varchar	10	Id jalan
Jenis	Varchar	20	Jenis objek
Lat	Varchar	50	latititude lokasi
Lng	Varchar	50	longitude lokasi

2. Tabel Jalan

Tabel lokasi adalah tabel yang digunakan untuk menyimpan data informasi lokasi latititude dan lokasi longitude yang menjadi titik objek wisata di sekitaran danau toba. Tabel ini mempunyai struktur seperti terlihat pada Tabel 2.

Tabel 2. Tabel Jalan

Nama Field	Tipe	Panjang	Keterangan
Id_jalan	Varchar	10	Id jalan
Nama_jalan	Varchar	10	Nama jalan
Keterangan	text		Keterangan
Jenis	Varchar	20	Jenis jalan

3.4 Perancangan Sistem Antarmuka (*Interface*)

Pada tahap ini dilakukan perancangan sistem antar muka untuk berinteraksi antara user dengan sistem. Tahapan ini sangat penting karena desain antarmuka yang baik akan membuat pengguna merasakan kenyamanan dalam menggunakan sebuah aplikasi komputer. Berikut struktur menu-menu yang akan dirancang pada aplikasi pencarian rute terpendek ini.

3.4.1 Perancangan Data Objek Wisata



Gambar 4. Perancangan Data Objek Wisata

Rancangan gambar di atas ini menampilkan pengolahan posisi objek di peta. Rancangan halaman ini sendiri dikhususkan untuk seorang admin. Admin sendiri harus terlebih dahulu login apabila ingin masuk ke halaman ini. Setelah itu admin menginput latititude lokasi dan longitude lokasi pada form yang telah di sediakan oleh sistem.

3.4.2 Perancangan Pencarian Rute Terpendek



Gambar 5. Perancangan Pencarian Rute Terpendek

Rancangan gambar di atas ini menampilkan pengolahan pencarian jalur terpendek. Pada halaman ini user diwajibkan terlebih dahulu memasukkan titik awal pada kotak start setelah itu submit, barulah sistem melakukan proses pencarian jalur terpendek dengan algoritma djikstra, dan sistem menampilkan keterangan jalur yang dilewati dan jarak tempuh.

Pada halaman ini terdapat kotak waypoint yang berfungsi sebagai daerah dimana pencarian rute terpendek itu.

3.4.3 Data Objek Wisata Di Danau Toba

Table 3. Data wisata di Danau Toba

No	Nama Objek Wisata	X (latitute)	Y (longitute)
1	Taman Simalem Resort	2.887915	98.507382
2	Pertunjukan sigale-gale	2.651844	98.860807
3	Air terjun Sipiso-piso	2.916373	98.519498
4	Menara pandang tele	2.552086	98.639748
5	Pantai indah situngkir	2.697794	98.690051
6	Museum Huta Bolon Simanindo	2.752233	98.744477
7	Air Terjun Efrata	2.537343	98.644697
8	Rumah pengasingan Bung karno	2.665530	98.926415
9	Tuk-tuk	2.675020	98.856908
10	Tugu perjuangan tigras	2.810267	98.779568
11	Kebun Teh Sidamanik	2.856243	98.907570
12	Air terjun bah biak	2.830865	98.911019
13	Air Terjun Gibeon	2.590653	98.997140
14	Medan	3.539264	98.718062
15	Lubuk pakam	3.554429	98.879625
16	Sei rampah	3.491258	99.125894
17	Tebing tinggi	3.353946	99.166893
18	Pematang siantar	2.953368	99.058028
19	berastagi	3.197924	98.512499
	Pandu Lakedise	2.665087	98.935575
20			
21	Pelabuhan Simanindo	2.754545	98.745374
22	Pelabuhan Tigras	2.798035	98.789056
23	Pelabuhan ajibata	2.654108	98.933817
24	Rumah makan citra padang	2.664073	98.931193
25	Toko souvenir	2.664556	98.929189
27	Toko Mega Tindaon	2.666200	98.928060
28	Kedan mak bora	2.666100	98.928331

IV. HASIL DAN PEMBAHASAN

Dalam bab ini akan dibahas mengenai hasil uji coba sistem yang telah dirancang dan dibuat. Uji coba dilakukan untuk mengetahui apakah sistem dapat berjalan sebagaimana mestinya dengan lingkungan uji coba yang telah ditentukan serta dilakukan sesuai dengan perancangan.

Dalam menjalankan aplikasi web pencarian rute terpendek menuju sebuah Objek wisata danau toba dengan algoritma dijkstra, sebelumnya harus mempersiapkan kebutuhan dari program yang akan diimplementasikan baik dari segi perangkat keras maupun perangkat lunak komputer. Pada pengujian ini, hasilnya berupa sistem informasi pencarian rute terpendek menuju sebuah Objek wisata danau toba.

Untuk menjalankan aplikasi pencarian rute terpendek menuju Objek wisata danau toba dengan algoritma Dijkstra,

4.2. Halaman Home

Halaman ini akan tampil pertama sekali sewaktu user masuk ke halaman website dengan alamat `http://localhost/wisata_toba/` di *web browser* seperti *Google Chrome*, sebelum *admin* memproses data, *admin* diharuskan *login* terlebih ke menu login dengan memasukkan *username* dan *password*, fungsinya adalah untuk mencegah pengguna yang tidak berhak masuk ke halaman input data rute terpendek,



Gambar 6. Halaman Home

Di dalam halaman utama terdapat beberapa sub menu, diantaranya yaitu:

1. Data Objek Wisata
Menu ini berisi tentang informasi objek ,antara lain Wisata, Rumah ibadah, Hotel, Rumah makan, Pelabuhan, Kota dan toko souvenir,serta bagi admin dapat menginput data objek pada menu ini.
2. Posisi Objek Wisata
Menu ini berisi data *latitude* dan *Longitude* posisi objek di peta.
3. Pencarian Jalur
Menu ini menampilkan peta dan proses pencarian jalur terpendek.
4. Login
Menu untuk menampilkan halaman login admin.

4.3 Halaman Menu Login

Pada halaman *login* sistem akan menampilkan *User name* dan *Password*,setelah admin memasukkan *user name* dan *Password* dan mengklik tombol *login*, jika *login* berhasil makas sistem akan membuka halaman menu *home*,



Gambar 7. Halaman Menu Login

Di dalam halaman logina terdapat beberapa sub menu, diantaranya yaitu:

1. *User name*
Di kotak *user name* admin mengisi data yaitu dengan kata "admin".
2. *Password*
Di kotak *password* admin mengisi data yaitu dengan kata "admin".
3. *Login*
Tombol ini jika di klik akan memproses data yang dimasukkan, Jika data yang dimasukkan benar, Maka akan tampil Halaman admin.

4.4 Halaman Menu Pencarian Rute

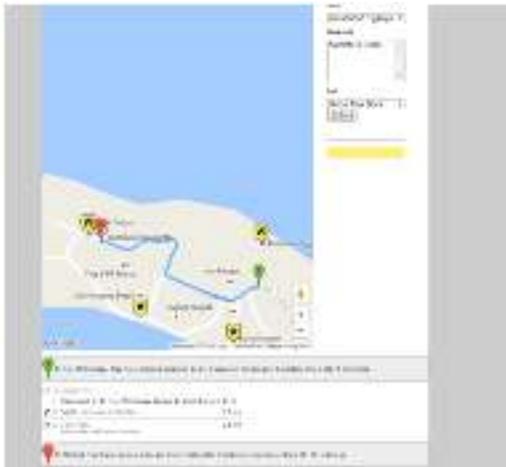
Halaman Form Pencarian disediakan tabel untuk user memilih posisi start dan tujuan objek wisata yang diinginkan lalu user mengklik tombol *submit* untuk mencari jalur terpendek.



Gambar 8. Halaman Menu Pencarian Jalur

4.5 Tampilan Hasil Pencarian Jalur

Pada pengujian ini lokasi awal di Pelabuhan tigaraja dan titik tujuan adalah rumah makan mak bora dengan jalur yang dilewati Jln.Kol TPR Sinaga dan Jln. Marihat, Hasil jarak tempuh sekitar 0,5 km dan dengan jarak waktu 2 menit dengan mengendarai mobil, hasil pencarian jalur terpendek dapat di lihat pada Gambar 9.



Gambar 9. Tampilan Hasil Pencarian Jalur

4.6 Pemeliharaan Sistem (*Maintenance*)

Pemeliharaan sistem informasi adalah suatu upaya untuk memperbaiki, menjaga, menanggulangi, mengembangkan sistem yang ada. Pemeliharaan ini di perlukan untuk meningkatkan efisiensi dan efektivitas kinerja sistem yang kita ada agar dalam penggunaannya dapat optimal. Berikut ini beberapa pengertian lain tentang pemeliharaan sistem dari beberapa sumber:

- Merupakan siklus terakhir dari SDLC.
- Pemeriksaan periodik, audit dan permintaan pengguna akan menjadi *source*.

Untuk melakukan perawatan sistem diseluruh masa hidup sistem Pemeliharaan sistem merupakan cara terbaik untuk menjaga efisiensi sistem yang sudah ada. Berikut merupakan beberapa alasan mengapa kita perlu memelihara sistem yang ada:

- a. agar dapat meningkatkan sistem / kinerja sistem.
- b. Menyesuaikan dengan perkembangan, agar sistem yang ada tidak tertinggal.

Jenis – jenis pemeliharaan sistem meliputi :

- a. Pemeliharaan Korektif
Adalah pemeliharaan yang mengkoreksi kesalahan – kesalahan yang ditemukan pada sistem, pada saat sistem di jalankan / berjalan.
- b. Pemeliharaan Adaptif
Yaitu pemeliharaan yang bertujuan untuk menyesuaikan perubahan yang terjadi.
- c. Pemeliharaan Perfektif
Pemeliharaan ini bertujuan untuk meningkatkan cara kerja suatu sistem.
- d. Pemeliharaan Preventif
Pemeliharaan ini bertujuan untuk menangani masalah – masalah yang ada.

V. KESIMPULAN DAN SARAN

1.1 Kesimpulan

Berdasarkan pembahasan dan evaluasi dari bab-bab terdahulu, maka dapat diambil kesimpulan sebagai berikut:

1. Pencarian jarak terpendek menggunakan Algoritma Dijkstra dapat di implementasikan kedalam sistem informasi geografis untuk menunjukkan jarak terpendek yang akan dilalui menuju lokasi tujuan.
2. Hasil dari pencarian rute terpendek dilengkapi jarak tempuh dan waktu tempuh dari titik awal dan titik tujuan.
3. Routing pencarian belum dapat membedakan atribut untuk jalan satu arah, kontur dan peraturan lalu lintas lainnya dalam keadaan nyata.

DAFTAR PUSTAKA

- [1] Gusmao Antonio, Pramono Hadi Sholeh, Sunaryo, 2013, *Sistem Informasi Geografis Pariwisata Berbasis Web dan Pencarian Jalur Terpendek Dengan Algoritma Dijkstra*, EECCID Vol. 7, No. 2
- [2] Havaluddin, 2011, *Memahami Penggunaan UML (Unified Modelling Language)*, Jurnal Informatika Mulawarman, Vol. 6, No. 1
- [3] Kusri (2007) *Strategi Perancangan dan Pengelolaan Basis Data*, Yogyakarta : CV. Andi Offset
- [4] Kementerian Agraria dan Tata Ruang/ Badan Pertahanan Nasional.
<http://www.bpn.go.id/Publikasi/Peraturan-Perundangan/Undang-Undang/undang-undang-nomor-9-tahun-1990-909>. . (diakses 09 November 2016 pukul 07.13 wib).
- [5] Lamhot Sitorus, 2015, *Algoritma dan Pemrograman*, Yogyakarta : CV. Andi Offset
- [6] Mahdia Faya dan Nofiyanto Fiftin, 2013, *Pemanfaatan Google Maps API Untuk pengembangan sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile Web*, Jurnal Sarjana Teknik Informatika Vol. 1, No. 1
- [7] Nugroho Adi, 2009, *Rekayasa Perangkat Lunak Menggunakan UML dan Java*, Yogyakarta : CV. Andi Offset
- [8] Suarga, 2012, *Algoritma dan Pemrograman*, Yogyakarta : CV. Andi Offset
- [9] Sudanaputu, 2013, *Strategi Pengembangan Desa Wisata Ekologis di Desa Belimbing, Kecamatan Pupuan Kabupaten Tabanan, Analisis Pariwisata*, Vol. 13, No. 1.