

ALGORITMA *DJIKTRA* UNTUK MENENTUKAN JALUR TERPENDEK PADA DISTRIBUSI AIR MINERAL

Muhammad Sadli

Jurusan Teknik Elektro, Universitas Malikussaleh Lhokseumawe, Aceh Utara
muhammad.sadli@yahoo.com

ABSTRACT

Implementation issues determine the shortest route to the water distribution using the algorithm Dijkstra. Distribution of mineral water is emphasized in the search by using the shortest path, which will look for some alternative solutions to the completion of a more effective and efficient. problems during this time is crossed by the shortest path to get to the destination, or whether it can keep costs to a minimum and also whether it can reduce the travel time to be traversed. Results of algorithm implementation Dijkstra here to show where the optimum route. Paths are achieved by using the shortest path 21 cities input is: [the city of Lhokseumawe, Kab. North Aceh district. East Aceh, Langsa, Kab. Aceh Tamiang, Kab. Gayo Lues, Kab. Southeast Aceh, City Subussalam, Kab. Singkil, Kab. South Aceh District. Southwest Aceh district. Central Aceh district. Nagan Raya, Kab. West Aceh District. Aceh Jaya, Banda Aceh, Kota Sabang, Kab. Aceh Besar, Kab. Pidie district. Pidie Jaya district. Bireun. By using the search algorithm generates Dijkstra route: [Kota Lhokseumawe, Kab. North Aceh district. East Aceh, Langsa, Kab. Aceh Tamiang, Kab. Gayo Lues, Kab. Southeast Aceh, City Subussalam, Kab. Singkil, Kab. South Aceh District. Southwest Aceh district. Central Aceh district. Nagan Raya, Kab. West Aceh District. Aceh Jaya, Banda Aceh, Kota Sabang, Kab. Aceh Besar, Kab. Pidie district. Pidie Jaya district. Bireun] with a total distance of 2255 kilometers and when input other cases the number of input the city as many as 21 cities also will produce different distances. Dijkstra algorithm is an algorithm to determine the shortest path.

Keywords : Line, Rute, Dijkstra's Algorithm

INTISARI

Implementasi persoalan menentukan rute terpendek pendistribusian air mineral pada menggunakan Algoritma Dijkstra. Pendistribusian air mineral ditekankan pada pencarian dengan menggunakan jalur terpendek, dimana akan dicari beberapa alternatif solusi penyelesaian yang lebih efektif dan efisien. permasalahan selama ini dilintasi adalah jalur yang paling singkat untuk sampai ke kota tujuan, atau apakah dapat menekan biaya seminimal mungkin dan juga apakah dapat mengurangi waktu tempuh perjalanan yang akan dilalui. Hasil implementasi algoritma Dijkstra disini menampilkan rute mana yang optimal. Jalur yang dicapai dengan jalur terpendek dengan menggunakan 21 kota input adalah: [Kota Lhokseumawe, Kab. Aceh Utara, Kab. Aceh Timur, Kota Langsa, Kab. Aceh Tamiang, Kab. Gayo Lues, Kab. Aceh Tenggara, Kota Subussalam, Kab. Aceh Singkil, Kab. Aceh Selatan, Kab. Aceh Barat Daya, Kab. Aceh Tengah, Kab. Nagan Raya, Kab. Aceh Barat, Kab. Aceh Jaya, Kota Banda Aceh, Kota Sabang, Kab. Aceh Besar, Kab. Pidie, Kab. Pidie Jaya, Kab. Bireun. Dengan pencarian menggunakan Algoritma Dijkstra menghasilkan rute: [Kota Lhokseumawe, Kab. Aceh Utara, Kab. Aceh Timur, Kota Langsa, Kab. Aceh Tamiang, Kab. Gayo Lues, Kab. Aceh Tenggara, Kota Subussalam, Kab. Aceh Singkil, Kab. Aceh Selatan, Kab. Aceh Barat Daya, Kab. Aceh Tengah, Kab. Nagan Raya, Kab. Aceh Barat, Kab. Aceh Jaya, Kota Banda Aceh, Kota Sabang, Kab. Aceh Besar, Kab. Pidie, Kab. Pidie Jaya, Kab. Bireun] dengan total jarak 2255 Km dan apabila menginputkan kasus lain dengan jumlah inputan kota sebanyak 21 kota juga maka akan menghasilkan jarak yang berbeda pula. Algoritma djiktra adalah algoritma untuk menentukan jalur terpendek.

Kata kunci : Jalur, Rute, Algoritma Dijkstra

I. PENDAHULUAN

Pemilihan jalur terpendek dalam sebuah perjalanan dari suatu tempat ketempat lain, sering kali perjalanan yang ditempuh tanpa pertimbangan terlebih dahulu. Sehingga kita sering melupakan beberapa faktor seperti :

biaya, waktu dan jarak yang kita tempuh selama melakukan perjalanan tersebut. Maka untuk itu kita perlu menentukan jalur terpendek dari kota asal ke kota tujuan.

Pencarian jalur terpendek dapat terbagi menjadi dua metode, metode *konvensional* dan metode *heuristik*. Metode *konvensional*

cenderung lebih mudah dipahami daripada metode *heuristik*, yaitu hanya membandingkan jarak masing-masing *node* dan kemudian mencari jarak yang terpendek. Tetapi, bila dibandingkan hasilnya, hasil yang diperoleh dari metode *heuristik* lebih *variatif*, hasil yang didapatkan lebih akurat, tingkat kesalahan yang dihasilkan pada perhitungan lebih kecil, dan waktu perhitungan yang diperlukan lebih singkat.

Dari permasalahan ini bagaimana menentukan rute yang tepat sehingga pendistribusian tersebut dapat sampai ke tempat tujuan dalam waktu yang singkat dan efisien. Waktu yang singkat dan efisien disini adalah difokuskan pada jarak yang terpendek. Dengan adanya jarak yang terpendek dan mengesampingkan kondisi jalan maka dapat disimpulkan ongkos dan waktu pengiriman juga akan minimum.

Untuk mencari lintasan terpendek banyak pilihan algoritma yang digunakan untuk mencari lintasan terpendek diantaranya dengan algoritma Djikstra, Misalkan pendistribusian ingin mendistribusikan air mineral ke seluruh Aceh dengan kota tujuan pertama, kedua, dan seterusnya sebagai berikut: Kota Lhokseumawe, Kab. Aceh Utara, Kab. Aceh Timur, Kota Langsa, Kab. Aceh Tamiang, Kab. Gayo Lues, Kab. Aceh Tenggara, Kota Subulussalam, Kab. Aceh Singkil, Kab. Aceh Selatan, Kab. Aceh Barat Daya, Kab. Aceh Tengah, Kab. Nagan Raya, Kab. Aceh Barat, Kab. Aceh Jaya, Kota Banda Aceh, Kota Sabang, Kab. Aceh Besar, Kab. Aceh Pidie, Kab. Pidie Jaya, dan Kab. Bireun dan waktu pendistribusian dari kota pertama sampai ke kota terakhir sudah ditentukan oleh perusahaan maka dengan algoritma Djikstra diharapkan dapat memberikan hasil yang optimum yaitu jalur dengan jarak yang terpendek.

II. LANDASAN TEORI

A. Analisa Sistem

Pada sistem lama distribusi air mineral mengalami keterlambatan sampai ketujuan, sehingga menjadi sulit untuk didapatkan di masing-masing tempat pendistribusian. Hal ini dikarenakan pemilihan rute terpendek hanya

berdasarkan jalan yang sering dilalui, sehingga waktu tempuh yang dilalui sering terabaikan. Untuk itu diperlukan sistem terbaru agar pendistribusian air mineral tidak mengalami keterlambatan. Pada sistem lama pemilihan rute terpendek hanya berdasarkan jalan yang sering dilalui, sehingga waktu tempuh yang dilalui sering terabaikan dan membuat distribusi air mineral sering terlambat sampai tujuan. pendistribusian air mineral pada perusahaan mineral dengan menggunakan Algoritma Djiktra kemudian lebih ditekankan pada jarak terpendek. Pengujian sistem dimaksudkan untuk mengetahui sejauh mana tingkat keberhasilan atau keakuratan sistem dalam pencarian rute terpendek. Sistem ini dibangun dengan menggunakan perangkat lunak menggunakan php dan MySQL. Input dalam sistem ini adalah jumlah kota dan nama kota tujuan, sedangkan output sistem adalah data tertulis nama kota yang dilalui dan jarak.

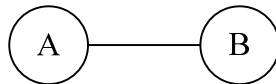
B. Jarak

Menurut Tom Henderson [1], jarak adalah angka yang menunjukkan seberapa jauh benda berubah posisi melalui suatu lintasan tertentu. Dalam fisika atau dalam kehidupan sehari-hari, jarak dapat berupa estimasi jarak fisik dari dua posisi berdasarkan kriteria tertentu, misalnya jarak tempuh Jakarta-Bandung. Jarak berbeda dengan koordinat posisi, jarak tidak mungkin bernilai negatif. Jarak merupakan besaran skalar sedangkan perpindahan merupakan besaran vektor. Jarak yang ditempuh oleh kendaraan biasanya ditunjukkan dalam kilometer, orang atau objek, haruslah dibedakan dengan jarak antara titik satu dengan yang lainnya.

C. Lintasan

Lintasan adalah jalan yang dilalui suatu materi/benda yang bergerak, lintasan atau trayek adalah tempat posisi titik-titik oleh suatu benda yang bergerak. Lintasan gerak benda bisa berupa garis lurus, lingkaran atau parabola, lintasan dapat lebih mudah dipahami dalam bentuk graf [2]

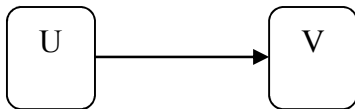
Perhatikan Gambar 1 yang dimaksud dengan lintasan adalah penghubung/jalur dari kota A ke kota B.



Gambar 1. Lintasan

D. Teori Graf

Graf adalah suatu struktur data yang berbentuk *network*/jaringan yaitu hubungan/relasi antar elemennya adalah *many to many*. Definisi *subgraf* adalah suatu *graf* yang merupakan suatu bagian (*subset*) dari *graf* itu sendiri [3]. Perhatikan gambar di bawah berikut:



Gambar 2. Graf

dimana :

U = simpul asal (*source node*)

V = simpul tujuan (*destination node*)

Ciri khas suatu *graf* terdapat 2 syarat, yaitu:

1. Minimal ada 2 elemen yang disebut *node* atau *vertices* atau *point*, yang dalam bahasa Indonesia disebut simpul atau titik
2. Antara 2 *node* dihubungkan langsung dengan sebuah *edge* (sisi atau jalur) yang unik yaitu berupa pasangan dari *pair* [u,v] atau ditulis dengan notasi $e=[u,v]$.

Misalnya, suatu *graf* $e = [u,v]$, maka u dan v dapat disebut *endpoint* dari e dan hubungan antara u dan v disebut berdekatan atau bertetangga. Suatu *graf* dikatakan lengkap jika paling sedikit terdapat suatu hubungan (sisi) diantara 2 *nodes* di dalam *graf*.

E. DFD (Data Flow Diagram)

Data flow diagram atau Diagram Arus Data adalah sebuah teknik grafis yang menggambarkan desain informasi yang diaplikasikan pada saat data bergerak dari *input* menjadi *output*. DFD merupakan alat analisis terstruktur yang baik, karena dapat menggambarkan arus pada suatu sistem secara terstruktur dan jelas [4].

F. Database (DBMS/RDBMS)

Mengolah *database* dengan Borland Delphi 7, Basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer

secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut [5].

Database digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam komputer. Untuk mengelola *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*). DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* untuk membuat, memelihara, mengontrol, dan mengakses *database* secara praktis dan efisien. Dengan DBMS, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada. Sedangkan RDBMS atau *Relationship Database Management System* merupakan salah satu jenis DBMS yang mendukung adanya *relationship* atau Hubungan antar tabel. Di samping RDBMS, terdapat jenis DBMS lain, misalnya *Hierarchy DBMS*, *Object Oriented DBMS*, dsb. Salah satu *software* atau perangkat lunak DBMS yang sering digunakan dalam aplikasi program yaitu *Microsoft Access*.

G. Distribusi

Pengertian sistem distribusi adalah pengaturan penyaluran barang dan jasa dari produsen ke konsumen. Sistem distribusi dapat dibedakan menjadi persoalan transportasi membahas masalah pendistribusian suatu komoditas atau produk dari sejumlah sumber (*supply*) kepada sejumlah tujuan (*destination, demand*) dengan tujuan meminimumkan ongkos pengangkutan [6]. Ciri-ciri khusus persoalan transportasi adalah :

1. Terdapat sejumlah sumber dan sejumlah tujuan tertentu.
2. Kuantitas komoditas atau produk yang didistribusikan dari setiap sumber dan yang diminta oleh setiap tujuan, besarnya tertentu.
3. Komoditas yang dikirim atau yang diangkut dari suatu sumber ke suatu tujuan, besarnya sesuai dengan permintaan dan atau kapasitas sumber. Ongkos pengangkutan komoditas dari suatu sumber ke suatu tujuan, besarnya tertentu.

Distribusi sangat dibutuhkan oleh konsumen untuk memperoleh barang-barang yang dihasilkan oleh produsen, apalagi bila

produksinya jauh. Anda dapat melihat barang yang tidak dihasilkan di daerah Anda tapi sekarang ada di tempat tinggal Anda.

H. Algoritma Dijkstra

Pandang *graph* berarah dengan bobot non negatip $G=(V,E)$, 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. Selain mencari *shortest path* dari vertek tunggal v ke setiap vertek yang lain, dapat juga untuk mencari *shortest path* diantara semua pasangan titik. Secara formal, masalah *shortest path* setiap pasang adalah untuk mencari *shortest path* di antara semua pasang vertek $v_i, v_j \in V$ sedemikian sehingga $i \neq j$. Untuk suatu *graph* dengan n vertek, keluaranya adalah matriks berukuran $n \times n$ dari $D=d(I,j)$ sedemikian sehingga d_{ij} adalah biaya *shortest path* dari vertek v_i ke vertek v_j .

Bobot garis dapat merepresentasikan waktu, biaya, pinalti, kerugian, atau beberapa kuantitas lain secara akumulatif.

Algoritma Dijkstra untuk mencari *single-source shortest path* dari suatu vertek tunggal s , dilakukan secara *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 $\theta(n^2)$. Sedang 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 . Algoritma ini memerlukan kompleksitas $\theta(n^3)$.

Segmen program berikut menunjukkan Algoritma Dijkstra Sekuensial untuk *Shortest Paths Single Source* [Brassard,1996]. Pada prosedur ini untuk setiap vertek $u \in (V-V_T)$, meletakkan $l[u]$, sebagai biaya minimal untuk menjangkau vertek u dari vertek s dimana vertek-vertik berada di V_T .

```

1. Procedure DIJKSTRA-SINGLE-SOURCE-SP(V,E,w,s)
2. Begin
3.    $V_T = \{s\};$ 
4.   For all  $v \in (V-V_T)$  do
5.     If  $(s,v)$  exists set  $l[v]=w(s,v);$ 
6.     Else set  $l[v]=\infty;$ 
7.   While  $V_T \neq V$  do
8.     Begin
9.       Find a vertex  $u$  such that  $l[v]$ 
        $= \min \{ l[v] \mid v \in (V-V_T) \};$ 
10.       $V_T = V_T \cup \{u\};$ 
11.      For all  $v \in (V-V_T)$  do
12.         $l[v] = \min \{ l[v], l[u]$ 
           $+ w(u,v) \};$ 
13.      Endwhile

```

I. Website

Website adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah website biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext* [8].

Domain adalah nama unik yang dimiliki oleh sebuah institusi sehingga bisa diakses melalui internet, misalnya *google.com*, *yahoo.com*, dan lain-lain. Istilah lain yang sering ditemui sehubungan dengan *website* adalah *homepage*. *Homepage* adalah halaman awal sebuah domain.

J. PHP

PHP adalah singkatan dari *PHP: Hypertext Preprocessor*. PHP merupakan bahasa pemrograman untuk membuat web yang bersifat *server-side scripting*. PHP memungkinkan *programmer* membuat halaman web yang bersifat dinamis. PHP dapat dijalankan pada berbagai macam sistem operasi, misalnya Windows, LINUX, dan Mac OS. Selain Apache, PHP juga mendukung beberapa *web server* lain, misalnya Microsoft IIS, Caudium, PWS, dan lain-lain. PHP memanfaatkan database untuk menghasilkan halaman web yang dinamis. Sistem manajemen *database* yang sering digunakan bersama PHP adalah MySQL.

Namun, PHP juga mendukung sistem manajemen database lain seperti Oracle, Microsoft Access, Interbase, dBase, PostgreSQL, dan lain-lain [9]. Hingga kini, PHP sudah berkembang hingga versi 5. PHP 5 mendukung penuh *Object-Oriented Programming* (OOP), integrasi XML, mendukung semua ekstensi terbaru MySQL, pengembangan web services dengan SOAP dan REST, serta ratusan peningkatan lainnya dibandingkan dengan versi sebelumnya. PHP juga bersifat *open source* sehingga setiap orang dapat menggunakannya secara gratis.

K. MySQL

MySQL adalah sebuah sistem manajemen database yang bersifat *open source*. MySQL dibuat dan dikembangkan oleh MySQL AB yang berada di Swedia. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya. Anda dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada di dalam database. MySQL merupakan sistem manajemen database yang bersifat relasional. Artinya data-data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga memanipulasi data akan menjadi lebih cepat.

MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar. MySQL juga dapat menjalankan perintah-perintah *Structured Query Language* (SQL) untuk mengelola database relasional yang ada di dalamnya [9].

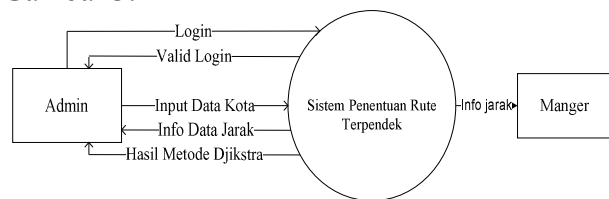
L. Implementasi

Implementasi sistem pendistribusian air mineral dibuat sesuai pada rancangan sebelumnya. Lingkungan implementasi sistem dalam pengimplementasian sistem ini dibutuhkan spesifikasi peralatan sebagai berikut :

1. Perangkat Keras (*Hardware*)
 - a. Intel ® Core i3 CPU 350 @2.26 GHz
 - b. Harddisk 500 GB
2. Perangkat Lunak (*Software*)
 - a. Microsoft Windows XP
 - b. XAMPP for Windows 1.7.3

M. Rancangan Data Flow Diagram (DFD)

Data Flow Diagram merupakan suatu gambaran grafis dari suatu sistem yang menggunakan bentuk-bentuk simbol untuk menggambarkan bagaimana data mengalir melalui suatu proses yang berkaitan. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau *output* dari sistem [10]. Diagram konteks sistem pendukung keputusan penentuan kelayakan lokasi perumahan, dapat dilihat pada Gambar 3.



Gambar 3. Diagram Konteks

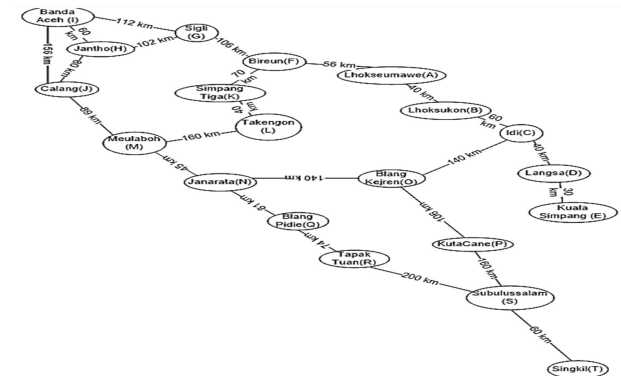
IV. HASIL DAN PEMBAHASAN

Berdasarkan penelitian dan survey lapangan yang dilakukan di Perusahaan air mineral Lhokseumawe, didapatkan hasil bahwa Perusahaan mendistribusikan ke berbagai kota dan kabupaten di seluruh Aceh yang terdiri dari kabupaten Aceh Utara, Aceh Timur, Aceh Tamiang, Aceh Tenggara, Aceh Selatan, Aceh Singkil, Aceh Barat Daya, Aceh Barat, Aceh Jaya, Aceh Besar, Aceh Tengah, Lhokseumawe, Langsa, Subulussalam, Simeulu, Nagan Raya, Gayo Lues, Bener Meriah, Bireun, Pidie Jaya, Banda Aceh, Sabang. Sekarang ini Perusahaan air mineral menggunakan jalur transportasi umum untuk mendistribusikan air mineralnya. Sehingga dalam proses pengangkutan air mineral, truk-truk tidak harus bolak balik dari satu kota ke kota yang lain. Dengan demikian berdasarkan hasil survey lapangan, Perusahaan air tersebut harus memilih rute yang tepat dalam mendistribusikan Air Mineral, supaya pendistribusiannya tepat dan tidak banyak menghabiskan waktu dan biaya. Misalkan Perusahaan air ingin mendistribusikan air mineralnya ke beberapa wilayah, sedangkan wilayah yang akan dikirim berbeda setiap harinya. Bisa saja wilayah yang melakukan permintaan bersamaan pada saat itu akan tidak berbarengan lagi di lain waktu karena

tergantung oleh pemakaian konsumen di daerah tersebut.

A. Pencarian Jalur Terpendek Menggunakan Algoritma Dijkstra

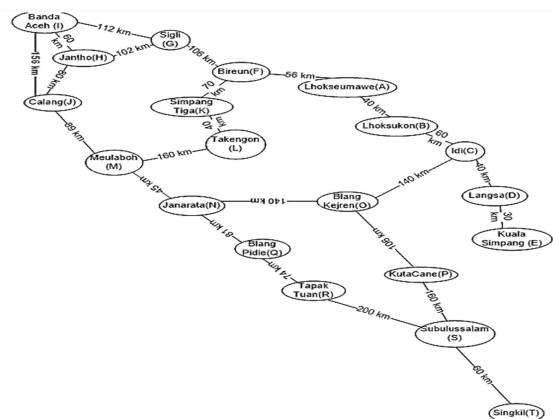
Menentukan jalur terpendek menggunakan gambar dibawah ini sebagai acuan pertama dari kota banda aceh menuju lhokseumawe.



Gambar 4. Peta Distribusi

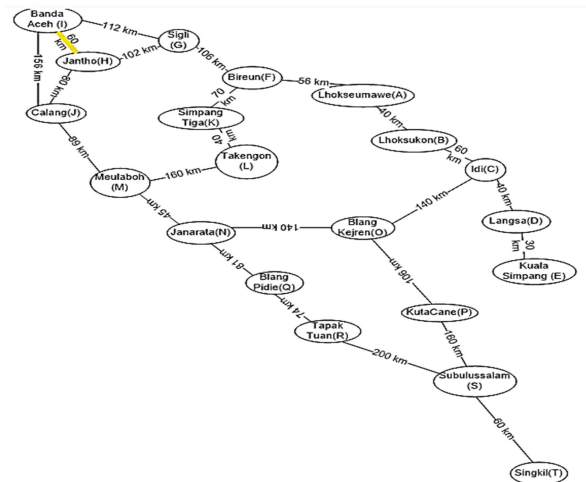
Di bawah ini penjelasan langkah per langkah pencarian jalur terpendek secara rinci dimulai dari *node* awal sampai *node* tujuan dengan nilai jarak terkecil.

1. *Node* awal I, *Node* tujuan A. Setiap *edge* yang terhubung antar *node* telah diberi nilai.



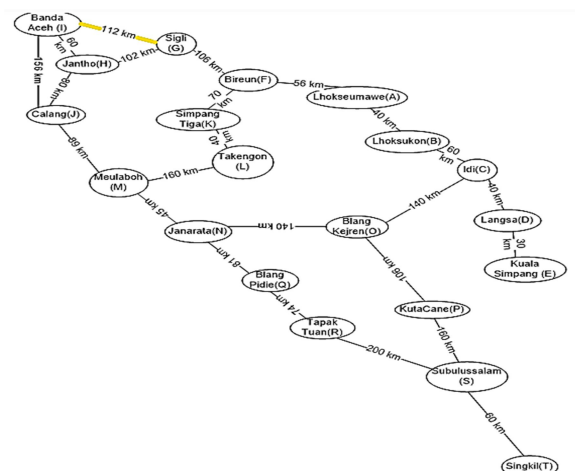
Gambar 5. Algoritma Dijkstra Langkah 1

2. Dijkstra melakukan kalkulasi terhadap *node* tetangga yang terhubung langsung dengan *node* keberangkatan (*node* I), dan hasil yang didapat adalah *node* H karena bobot nilai *node* H paling kecil dibandingkan nilai pada *node* lain.



Gambar 6. Algoritma Dijkstra Langkah 2

3. *Node* H diset menjadi *node* keberangkatan dan ditandai sebagai *node* yang telah terjamah. Dijkstra melakukan kalkulasi kembali terhadap *node-node* tetangga yang terhubung langsung dengan *node* yang telah terjamah. Dan kalkulasi dijkstra menunjukan bahwa *node* G yang menjadi *node* keberangkatan selanjutnya karena bobotnya yang paling kecil dari hasil kalkulasi terakhir.



Gambar 7. Algoritma Dijkstra Langkah 3

4. *Node* G diatur menjadi *node* keberangkatan dan ditandai sebagai *node* yang telah terjamah. Dijkstra melakukan kalkulasi terhadap *node* tetangga yang terhubung langsung dengan *node* keberangkatan (*node* G), dan hasil yang didapat adalah *node* F karena bobot nilai *node* F paling kecil dibandingkan nilai pada *node* lain.

B. Implementasi Pendistribusian Dijkstra

1. Form Data Kota

Form ini berisi id dan nama-nama kota yang akan digunakan pada sistem, adapun tampilannya adalah sebagai berikut :

Gambar 8. Form Data Kota

2. Form Data Jarak

Form ini menyimpan data jarak kota asal dengan kota tetangga yang nantinya akan digunakan untuk perhitungan *dijkstra*, adapun tampilannya adalah sebagai berikut:

Gambar 9. Form Data Jarak

3. Form Dijkstra

Form ini berisi perhitungan dan informasi rute terpendek menggunakan algoritma *Dijkstra*. adapun tampilannya adalah sebagai berikut :

Gambar 10. Form Dijkstra

V. KESIMPULAN

Berdasarkan penelitian ini dapat disimpulkan bahwa:

1. Hasil Pencarian jalur terpendek memberikan hasil jarak maupun jalur yang akan dilintasi dan menampilkan jarak hasil yang akan ditempuh
2. *Algoritma Dijkstra* bisa digunakan untuk melakukan pencarian rute terpendek pendistribusian air mineral yang mencakup wilayah kerja Aceh hasil Pencarian jalur terpendek memberikan hasil yang sama baik jarak maupun jalur yang akan dilintasi

REFERENSI

- [1]. Henderson, T., 2004. "Distance and Displacement". MIT Press and McGraw-Hill.
- [2]. Glosarium. 2014. Lintasan. From <http://glosarium.org/arti/?k=lintasan>, Diakses tanggal 10 September 2014
- [3]. Diestel, Reinhard., 2000. *Graph Theory*. New York: Springer-Verlag.
- [4]. Fatta, Al Hanif, 2009. "Rekayasa Sistem Pengenalan Wajah". Yogyakarta:Penerbit Andi
- [5]. Alam, M., Agus J., 2003. *Mengolah database dengan borland delphi 7*. Jakarta:PT.Elex Media Komputindo.

- [6]. Saputro E.T., Aprilia P., 2012. *“Perencanaan Jadwal dan Rute Distribusi Rokok untuk Menekan Total Biaya Transportasi”*. Universitas Muhammadiyah Malang.
- [7]. Giswiki.org.(Anonim). *Algoritmus Von Dijkstra* (Online),(http://www.giswiki.org/wiki/Algoritmus_von_Dijkstra) diakses tanggal 10 April 2014.
- [8]. Yuhefizer, 2009. *Cara Mudah Membangun Website Interaktif Menggunakan Content Management System Joomla Edisi Revisi*. Jakarta: PT Elex Media Komputindo.
- [9]. Ramadhan Arief.,2006. *Student Guide Series- Pemrograman Web Database dengan PHP dan MySQL*. Jakarta: PT Elex Media Komputindo.
- [10]. Pressman, R.S., 2001, *Software Engineering, A Practitioner’s Approach*, 5th Edition, McGraw-Hill, Inc. New York.