



Artikel Penelitian

Penerapan Algoritma *Floyd Warshall* dalam Aplikasi Penentuan Rute Terpendek Mencari Lokasi BTS (*Base Tower Station*) pada PT.GCI Palembang

Vera Apriliani Nawagusti^a

^aJurusan Teknik Elektro, Program Studi Teknik Telekomunikasi, Politeknik Negeri Sriwijaya, Jl. Sriwijaya Negara Bukit Besar, Kota Palembang, Sumatera Selatan 30139, Indonesia

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 00 Februari 00

Revisi Akhir: 00 Maret 00

Diterbitkan Online: 00 April 00

KATA KUNCI

Pencarian Rute Terpendek,
Algoritma *Floyd Warshall*,
Graf,
PT.GCI Palembang

KORESPONDENSI

Telepon: 082184926168

E-mail: veraapriliani97@gmail.com

A B S T R A C T

For PT.GCI Palembang optimization of the shortest route is needed in the process of finding the location of BTS (*Base Tower Station*) because with the optimization of the shortest route determination can assist the technician in optimizing the distance to the location of BTS needs to be done maintenance and repair so as to increase the efficiency of time and the required costs. The shortest route problem is a problem in finding the path between two vertices on a weighted graph that has a combined value of the number of weights on the side of the graph passed by the minimum number. In this research using *Floyd Warshall* algorithm to determine the shortest route. Where the *Floyd Warshall* algorithm can compare all track opportunities in the graph for each side of all the nodes it passes to the least amount. Based on the results of calculations from this research obtained the shortest route that is from point 1 – 4- 3 – 2 that is equal to 10,9 km

1. PENDAHULUAN

PT.GCI Palembang adalah sebuah perusahaan yang menyediakan produk dan layanan berupa *Radio Network Planning and Optimization*, *UMTS/CDMA/GSM CW testing Propagation Model Tuning*, *In-building System Design and Implementation Managed Service Operations & Maintenance (O&M)*, dan *BTS & MW & Core Engineering Service* [1]. PT.GCI Palembang juga bekerjasama dengan perusahaan lain dalam mengerjakan beberapa *project* operator telepon seluler. Salah satunya *project* perawatan BTS (*Base Tower Station*) milik operator telepon seluler seperti 3 yang tersebar di seluruh wilayah kota Palembang. Agar kualitas sinyal dari jaringan BTS tetap terjaga, maka PT.GCI Palembang harus tetap melakukan pemeriksaan dan perawatan terhadap BTS tersebut. Oleh karena itu dibutuhkan sebuah aplikasi penentuan rute terpendek antar BTS di kota Palembang yang dapat berjalan pada *platform* android, agar dapat membantu teknisi dalam mengoptimalkan jarak tempuh menuju lokasi BTS apabila terjadi suatu kerusakan dan perawatan pada BTS tersebut.

Location Based Service (LBS) merupakan suatu layanan yang dapat digunakan untuk mengetahui posisi berdasarkan titik geografis dari lokasi pengguna dan lokasi lain yang diinginkan menggunakan suatu perangkat *mobile* [2]. Teknologi berupa posisi ini selanjutnya dikembangkan menjadi layanan penentuan rute terpendek. Teori graf dapat memecahkan permasalahan rute terpendek. Menurut teori graf, permasalahan rute terpendek dapat diartikan sebagai suatu permasalahan dalam menemukan lintasan antara dua buah simpul pada graf berbobot pada sisi graf yang dilintasi dengan jumlah yang paling minimum [3].

Terdapat beberapa metode algoritma yang dapat diterapkan untuk menyelesaikan permasalahan penentuan rute terpendek seperti *Dijkstra*, *Bellman-Ford*, *Floyd Warshall*, dan lain sebagainya. Algoritma *Dijkstra* lebih cepat dalam menemukan rute terpendek, namun algoritma *Dijkstra* hanya memikirkan solusi terbaik yang akan diambil pada setiap langkah tanpa memikirkan konsekuensinya ke depan serta pada algoritma *Dijkstra* tidak dapat mengatasi graf berbobot negatif [4][5]. Sedangkan pada algoritma *Bellman-Ford* dapat menjamin ditemukannya solusi optimum global dengan menggunakan perhitungan pohon ruang status namun pada algoritma *Bellman-Ford* membutuhkan waktu

yang lebih lama dibandingkan dengan algoritma *Dijkstra* [6]. Pada algoritma *Floyd Warshall* yang menerapkan program dinamis lebih menjamin keberhasilan dalam penemuan solusi lintasan terpendek, karena algoritma ini dapat membandingkan semua kemungkinan lintasan pada *graf* untuk setiap sisi dari semua simpul yang dilewati [7][8].

Berdasarkan hal tersebut, pada penelitian ini solusi yang digunakan dalam kasus penyelesaian penentuan rute terpendek dalam mencari lokasi BTS (*Base Tower Station*) pada PT.GCI Palembang diselesaikan dengan menerapkan algoritma *Floyd Warshall*. Aplikasi penentuan rute terpendek mencari lokasi BTS di PT.GCI Palembang dengan menerapkan algoritma *Floyd Warshall* ini bertujuan untuk membantu teknisi dalam mengoptimalkan jarak tempuh menuju lokasi BTS apabila terjadi suatu kerusakan dan perawatan pada BTS tersebut.

Berdasarkan uraian diatas, maka dirumuskan beberapa permasalahan, yaitu :

1. Bagaimana proses tampilan pada aplikasi penentuan rute terpendek mencari lokasi BTS di kota Palembang ?
2. Bagaimana melakukan perhitungan menggunakan teori graf dengan menerapkan algoritma *Floyd Warshall* dalam menentukan rute terpendek mencari lokasi BTS di kota Palembang ?

Untuk mendapatkan hasil yang optimal dalam penelitian ini, maka diberikan batasan-batasan penelitian sebagai berikut :

1. BTS yang dipetakan pada aplikasi ini terletak di wilayah kota Palembang.
2. Aplikasi ini berjalan pada *platform* android.
3. Aplikasi ini menerapkan algoritma *Floyd Warshall* dalam pencarian rute terpendek.
4. Sumber data lokasi-lokasi BTS didapatkan dari PT.GCI Palembang.

Adapun tujuan yang ingin dicapai pada penelitian ini adalah untuk merancang suatu aplikasi penentuan rute terpendek yang menerapkan algoritma *Floyd Warshall* dalam mencari lokasi BTS pada PT.GCI Palembang, sehingga dapat mengoptimalkan jarak tempuh teknisi dalam menuju lokasi BTS yang menjadi tujuan.

2. TINJAUAN PUSTAKA

2.1. Location Based Service (LBS)

Qusay H.Mahmoud menyatakan bahwa *Location Based Service* (LBS) merupakan layanan yang berguna untuk mengetahui posisi dari pengguna, dimana informasi tersebut digunakan untuk menyediakan jasa dan aplikasi personal [5]. Terdapat 2 pendekatan dasar yang digunakan untuk pengaplikasian LBS, yaitu [5] :

1. memproses data lokasi di server dan mengirimkan hasilnya ke alat.
2. Memperoleh data lokasi dari alat tersebut menurut aplikasi yang menggunakannya secara langsung

Beberapa pendekatan yang digunakan aplikasi dalam mendapatkan posisi dari pengguna, antara lain :

1. Memanfaatkan jaringan telepon seluler, dimana tingkat ketepatan aplikasi dalam mendeteksi posisi berdasarkan

ukuran dari sel dan memungkinkan tingkat akurasi yang lebih rendah.

2. Memanfaatkan satelit, dengan mengaplikasikan GPS dalam mendapatkan informasi posisi yang akurat.

2.2. Global Positioning System (GPS)

Global Positioning System atau yang dikenal dengan GPS, merupakan suatu alat atau sistem yang berfungsi untuk menginformasikan posisi pengguna di permukaan bumi yang berbasis satelit. Informasi dikirim dari satelit berupa sinyal radio dengan data digital [4]. Posisi unit GPS ditetapkan berdasarkan titik koordinat *latitude* dan *longitude*.

2.3. Graf

Graf dapat digunakan dalam menyelesaikan permasalahan rute terpendek. Menurut teori graf, permasalahan rute terpendek didefinisikan sebagai suatu persoalan dalam menemukan lintasan antara dua buah simpul pada graf berbobot yang memiliki jumlah yang paling minimum dari jumlah bobot pada sisi graf yang dilintasi [3]. Suatu graf G terdiri dari dua himpunan graf, yaitu himpunan V dan himpunan E , dimana V merupakan himpunan tak kosong dari simpul-simpul, sedangkan E merupakan sisi yang menghubungkan sepasang simpul [9].

Berikut ini proses sebuah graf dibentuk untuk menggambarkan lokasi-lokasi dalam menyelesaikan persoalan rute terpendek [10]:

1. Gabungkan tiap titik dari setiap rute menjadi sebuah *connected graph* (graf terhubung).
2. Berikan arah lintasan pada rute sebagai aliran (*flow*) sehingga terbentuk suatu *directed graph* dari *connected graph* yang ada.
3. Informasi dari jarak tempuh yang didapatkan diubah menjadi bobot jarak. Implementasikan bobot jarak yang ada menjadi sebuah aliran beban *directed graph* sehingga menjadi sebuah *weighted graph* (graf berbobot).

Representasi Graf

Berikut ini representasi dari graf antara lain [9] :

1. *Adjacency Matrix*
Sebuah matriks digunakan untuk menunjukkan *adjacency set* dari setiap verteks dalam baris dan kolomnya. Dimana baris pada matriks menunjukkan nomor verteks *adjacency* berasal sedangkan kolom pada matriks menunjukkan nomor verteks kemana arah *adjacency*. Elemen matriks $[x,y]$ bernilai 1 apabila terdapat sisi dari x ke y , namun bernilai 0 apabila terdapat lintasan yang berawal dan berakhir pada simpul yang sama.
2. *Adjacency List*
Pada *adjacency list* representasi matriksnya berupa matriks sparse, dimana sebagian besarnya berisikan bilangan nol. Untuk efisiensi ruang, tiap baris pada matriks digantikan list yang hanya terdapat verteks-verteks dalam *adjacency set* V_x dari setiap verteks x .

2.4. Algoritma Floyd Warshall

Algoritma *Floyd Warshall* merupakan pemrograman dinamis yang melakukan penyelesaian masalah dengan melihat solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait. Prinsip optimalitas ialah prinsip yang diterapkan oleh

pemrograman dinamis, yaitu jika solusi total optimal, maka bagian solusi sampai suatu tahap (misalnya tahap ke-i) juga optimal [8].

Algoritma *Floyd Warshall* memiliki input graf berarah dan berbobot. Serta output dari algoritma *Floyd Warshall* adalah dengan menghitung bobot terkecil dari semua lintasan yang menghubungkan sebuah pasangan titik, dan melakukannya sekaligus untuk semua pasangan titik [9].

Berikut ini digunakan Formulasi Rekrusif untuk algoritma *Floyd Warshall* antara lain [9] :

1. Vertex-vertex antara dalam *short path*
2. Jika $V = \{1,2,3,...,N\}$, untuk $k=0,.....n$. Maka $d_{ij}^{(k)} =$
 - w_{ij} jika $k = 0$
 - $\min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$, untuk $k > 0$
3. Solusi dari $d_{ij}^{(n)}$ merupakan matriks *shortest path* dari vertex i ke vertex j .

Gambar 1 menampilkan *Pseudocode* dari algoritma *Floyd Warshall*.

```
//Algoritma Floyd Warshall
for (k=1; k<=n; k++)
{
    for (i=1; i<=n; i++)
    {
        for (j=1; j<=n; j++)
        {
            if ( ( X[i,j] > (X[i,i] + X[k,j] ) ) )
            {
                X[i,j] = (X[i,i] + X[k,j])
            }
        }
    }
}
```

Gambar 1. *Pseudocode* Algoritma *Floyd Warshall* [8]

2.5. Google Maps API

Google Maps API adalah sebuah layanan (*service*) yang diberikan oleh *Google* kepada para pengguna untuk memanfaatkan *Google Maps* dalam mengembangkan aplikasi [11]. *Google Maps* API menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis *services* yang dimiliki, serta mengijinkan kepada pengguna untuk membangun aplikasi *enterprise* di dalam websitenya [11].

3. METODOLOGI

Perancangan aplikasi pada penelitian ini terbagi menjadi dua bagian yaitu perancangan tampilan (.xml) dan perancangan perintah program (.java). Perancangan tersebut diawali dengan perancangan tampilan aplikasi. Didalam penelitian ini, desain

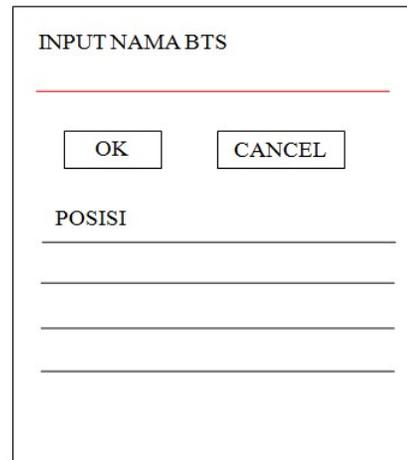
tampilan yang dirancang akan diuji coba apakah sudah sesuai dengan yang diharapkan. Setelah pengujian tersebut, dilakukan perancangan perintah program pada aplikasi. Seperti halnya pada perancangan tampilan, pengujian juga akan dilakukan pada perintah program aplikasi.

3.1. Perancangan Desain Tampilan Aplikasi

Perancangan tampilan ialah perancangan untuk tampilan aplikasi yang dilakukan pada *layout* yang terdapat pada aplikasi android studio. Pembuatan desain aplikasi dapat dilakukan pada file XML atau *Graphical Layout* di dalam *software*. Pada *layout* terdapat pilihan *widget* seperti *button*, *text*, *images*, dan lain-lain yang dapat diatur sesuai dengan desain yang kita inginkan. Pada file XML, *widget* dapat diatur menggunakan *coding* sedangkan pada *graphical layout*, *widget* dapat diatur secara manual.

3.1.1 Rancangan Tampilan Menu Utama

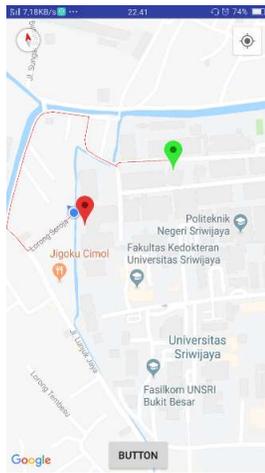
Pada perancangan ini, terdapat menu input nama BTS, dimana pada menu ini akan diinput nama-nama BTS yang mengalami kerusakan. Pada menu input nama BTS, tampilannya akan diatur menggunakan *widget* berupa *button*. *Button* tersebut berupa *button OK* dan *cancel*. *Button OK*, akan ditampilkan beberapa *list* nama BTS yang mengalami kerusakan, sedangkan *button cancel*, digunakan untuk membatalkan *list* nama BTS yang akan diinput. Rancangan tampilan utama dapat dilihat pada Gambar 2.



Gambar 2. Rancangan Tampilan Menu Input Nama BTS

3.1.2 Rancangan Tampilan Floyd Warshall

Pada perancangan ini akan menampilkan rute terpendek dari hasil perhitungan menggunakan algoritma *Floyd Warshall*.



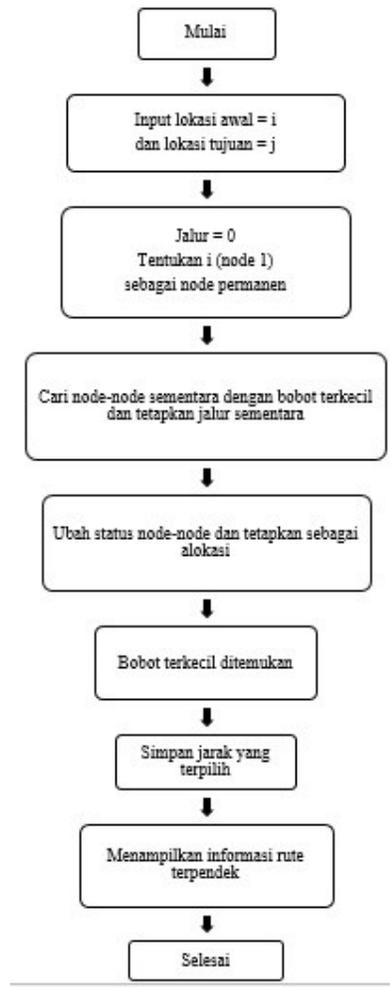
Gambar 3. Rancangan Tampilan Menu *Floyd Warshall*

3.2. Perancangan Perintah Aplikasi

Pada perancangan ini dilakukan proses penentuan rute terpendek menggunakan algoritma *Floyd Warshall*. Berikut ini *flowchart* dari perancangan perintah program menu algoritma *Floyd Warshall* yang dapat terlihat pada Gambar 4.

4. HASIL DAN PEMBAHASAN

Pada tahapan ini merupakan tahapan dimana didapatkan hasil dari perancangan *layout* yang telah diimplementasikan kedalam program yang dibuat dengan menggunakan android studio seperti yang terlihat pada Gambar 5 dan 6, sehingga menjadi sebuah aplikasi penentuan rute terpendek dalam mencari lokasi BTS (*Base Tower Station*) di kota Palembang.

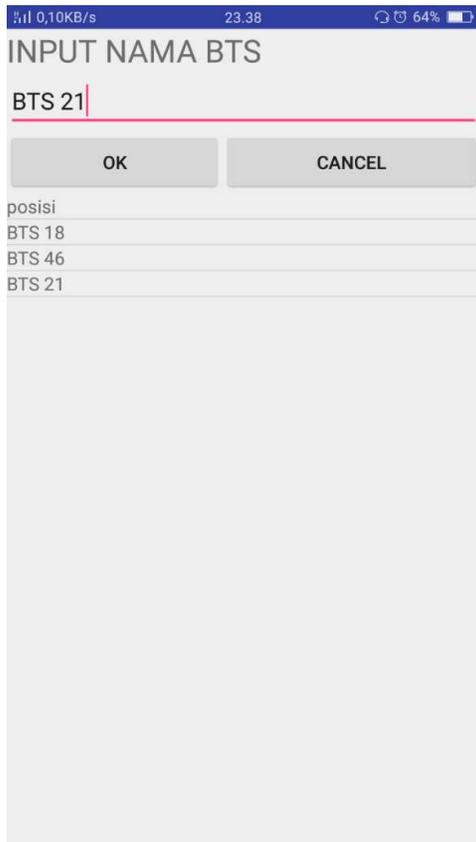


Gambar 4. *Flowchart* Perancangan Perintah Program Algoritma *Floyd Warshall*

4.1. Tampilan Aplikasi

Pada penelitian ini didapatkan 2 tampilan dari aplikasi penentuan rute terpendek mencari lokasi BTS (*Base Tower Station*) di kota Palembang, yang

terdiri dari tampilan menu input nama BTS, seperti pada Gambar 5 dan tampilan menu *Floyd Warshall* pada Gambar 6.



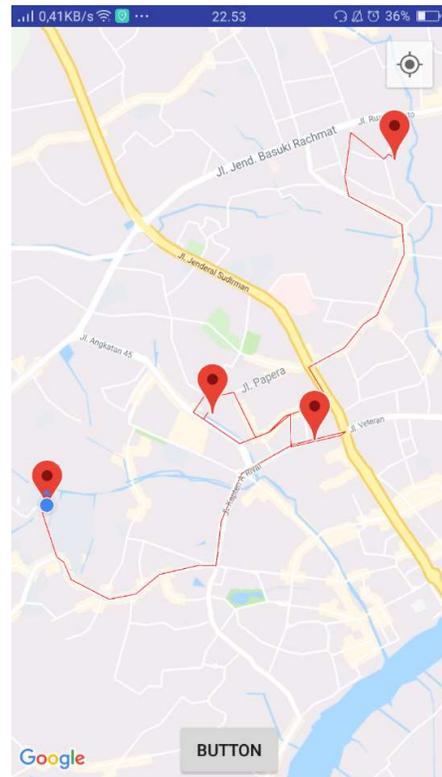
Gambar 5. Tampilan Menu Input Nama BTS

Gambar 5 menunjukkan tampilan menu input nama BTS, dimana nama-nama BTS yang diinput merupakan letak pin lokasi BTS yang perlu dilakukan perawatan maupun perbaikan seperti pada kasus ini BTS yang perlu dilakukan perbaikan maupun perawatan yaitu BTS 18, BTS 46, dan BTS 21. Sedangkan posisi merupakan lokasi teknisi saat ini yang terbaca secara otomatis oleh GPS pada *smartphone* teknisi.

Pada Gambar 6 merupakan tampilan menu *Floyd Warshall*, dimana pada menu ini merupakan hasil rute terpendek yang dihasilkan dengan menerapkan perintah program algoritma *Floyd Warshall* pada android studio.

4.2. Pengujian Aplikasi

Pada penelitian ini dilakukan pengujian terhadap penentuan rute terpendek menuju lokasi tujuan. Pengujian ini dilakukan dengan cara perhitungan secara manual untuk melihat apakah proses penentuan rute terpendek pada aplikasi yang dibuat sudah sesuai teori graf dengan menerapkan algoritma *Floyd Warshall* atau tidak.



Gambar 6. Tampilan Menu *Floyd Warshall*

Berikut pengujian rute terpendek pada penelitian ini : misalkan teknisi akan melakukan perawatan pada BTS 18, BTS 21, dan BTS 46, dimana titik 1 diasumsikan sebagai posisi awal teknisi yang dapat kita lihat pada gambar 6 terdapat titik biru, dan titik 2 sebagai BTS 18 yang terletak di Jl. Orde Baru atau jika dilihat pada gambar 6 lokasi pin terletak di sekitar Jl. Jend. Basuki Rachmat, titik 3 sebagai BTS 21 yang terletak di Jl. Angkatan 45, dan titik 4 sebagai BTS 46 yang terletak di Jl. Kapten A. Rivai. Dengan menerapkan metode algoritma *Floyd Warshall* dalam mencari rute terpendek dari titik 1 menuju ke titik 2, 3, dan 4, didapatkan bahwa :

Tabel dibawah ini menunjukkan data jarak lokasi BTS 18, BTS 21 dan BTS 46.

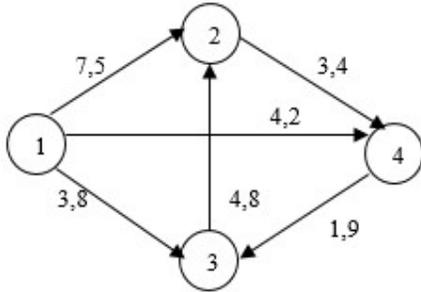
Tabel 1. Lokasi BTS (*Base Tower Station*)

Nama BTS	Alamat BTS	Latitude	Longitude
BTS 18	Jl. Orde Baru/ Rawasari	-2.95573	104.75804
BTS 21	Jl. Angkatan 45	-2.97552778	104.7438611
BTS 46	Jl. Kapten A. Rivai	-2.9776	104.7518

Tabel 2. Jarak Antar Lokasi BTS (*Base Tower Station*)

Lokasi Awal	Lokasi Tujuan	Jarak (km)
1	2	7,5 km
1	3	3,8 km
1	4	4,2 km
2	4	3,4 km
3	2	4,8 km
4	3	1,9 km

Berikut ini ditampilkan ilustrasi dari suatu graf berbobot yang merepresentasikan jarak antar titik lokasi BTS yang ditunjukkan pada Gambar 7 :



Gambar 7. Graf berbobot jarak antar titik

Berikut ini tahapan perhitungan algoritma *Floyd Warshall* dalam menentukan rute terpendek yang digambarkan melalui matriks. Dimana baris pada matriks menunjukkan lokasi awal dan kolom pada matriks menunjukkan lokasi tujuan.

Dari Gambar 6, didapat :

- K = 0, 1, 2, 3, 4
- i = 1, 2, 3, 4
- j = 1, 2, 3, 4

Matriks hubung graf, K = 0

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	∞	3,4
3	∞	4,8	0	∞
4	∞	∞	1,9	0

Gambar 8. Matriks hubung K = 0

Matriks hubung graf, K = 1

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	∞	3,4
3	∞	4,8	0	∞
4	∞	∞	1,9	0

Gambar 9. Matriks hubung K = 1

Rumus : $X[i,j] \leq X[i,k] + X[k,j]$ (1)

Penyelesaian :

- $X^0[2,3] < X^0[2,1] + X^0[1,3]$

- $\infty = \infty + 3,8$
- $X^0[2,4] < X^0[2,1] + X^0[1,4]$
 $3,4 < \infty + 4,2$
- $X^0[3,2] < X^0[3,1] + X^0[1,2]$
 $4,8 < \infty + 7,5$
- $X^0[3,4] < X^0[3,1] + X^0[1,4]$
 $\infty = \infty + 4,2$
- $X^0[4,2] < X^0[4,1] + X^0[1,2]$
 $\infty = \infty + 7,5$
- $X^0[4,3] < X^0[4,1] + X^0[1,3]$
 $1,9 < \infty + 3,8$

Matriks hubung graf, K = 2

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	∞	3,4
3	∞	4,8	0	8,2
4	∞	∞	1,9	0

Gambar 10. Matriks hubung K = 2

Penyelesaian :

- $X^1[1,3] < X^1[1,2] + X^1[2,3]$
 $3,8 < 7,5 + \infty$
- $X^1[1,4] < X^1[1,2] + X^1[2,4]$
 $4,2 < 7,5 + 3,4$
- $X^1[3,1] < X^1[3,2] + X^1[2,1]$
 $\infty < 4,8 + \infty$
- $X^1[3,4] < X^1[3,2] + X^1[2,4]$
 $8,2 = 4,8 + 3,4$
- $X^1[4,1] < X^1[4,2] + X^1[2,1]$
 $\infty = \infty + \infty$
- $X^1[4,3] < X^1[4,2] + X^1[2,3]$
 $1,9 < \infty + \infty$

Matriks hubung graf, K = 3

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	∞	3,4
3	∞	4,8	0	8,2
4	∞	6,7	1,9	0

Gambar 11. Matriks hubung K = 3

Penyelesaian :

- $X^2[1,2] \quad X^2[1,3] + X^2[3,2]$
 $7,5 < 3,8 + 4,8$
- $X^2[1,4] \quad X^2[1,3] + X^2[3,4]$
 $4,2 < 3,8 + 8,2$
- $X^2[2,1] \quad X^2[2,3] + X^2[3,1]$
 $\infty = \infty + \infty$
- $X^2[2,4] \quad X^2[2,3] + X^2[3,4]$
 $3,4 < \infty + 8,2$
- $X^2[4,1] \quad X^2[4,3] + X^2[3,1]$
 $\infty = 1,9 + \infty$
- $X^2[4,2] \quad X^1[2,3] + X^2[3,2]$
 $6,7 < 1,9 + 4,8$

Matriks hubung graf, K = 4

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	5,3	3,4
3	∞	4,8	0	8,2
4	∞	6,7	1,9	0

Gambar 12. Matriks hubung K = 4

Penyelesaian :

- $X^3[1,2] \quad X^3[1,4] + X^3[4,2]$
 $7,5 < 4,2 + 6,7$
- $X^3[1,3] \quad X^3[1,4] + X^3[4,3]$
 $3,8 < 4,2 + 1,9$
- $X^3[2,1] \quad X^3[2,4] + X^3[4,1]$
 $\infty = 3,4 + \infty$
- $X^3[2,3] \quad X^3[2,4] + X^3[4,3]$
 $5,3 = 3,4 + 1,9$
- $X^3[3,1] \quad X^3[3,4] + X^3[4,1]$
 $\infty = 8,2 + \infty$
- $X^3[3,2] \quad X^3[3,4] + X^3[4,2]$
 $4,8 < 8,2 + 6,7$

Hasil akhir lintasan terpendek *Floyd Warshall*

	1	2	3	4
1	0	7,5	3,8	4,2
2	∞	0	5,3	3,4
3	∞	4,8	0	8,2
4	∞	6,7	1,9	0

Gambar 13. Matriks akhir lintasan terpendek *Floyd Warshall*

Dimana :

- $1 - 2 - 3 - 4 = 7,5 + 5,3 + 8,2 = 21$ km
- $1 - 2 - 4 - 3 = 7,5 + 3,4 + 1,9 = 12,8$ km
- $1 - 3 - 2 - 4 = 3,8 + 4,8 + 3,4 = 12$ km
- $1 - 3 - 4 - 2 = 3,8 + 8,2 + 6,7 = 18,7$ km
- $1 - 4 - 2 - 3 = 4,2 + 6,7 + 5,3 = 16,2$ km
- $1 - 4 - 3 - 2 = 4,2 + 1,9 + 4,8 = 10,9$ km

Jadi, berdasarkan perhitungan secara manual menggunakan algoritma *Floyd Warshall* didapatkan rute terpendeknya adalah 1-4-3-2 yaitu sebesar 10,9 km. Itu artinya rute terpendek yang dipilih dari posisi teknisi saat ini adalah posisi awal – BTS 46 – BTS 21 – BTS 18 yaitu sejauh 10,9 km. Hasil rute terpendek yang dihasilkan menggunakan perhitungan manual menghasilkan rute yang sama dengan rute terpendek yang dihasilkan aplikasi seperti pada gambar 6 yaitu dari posisi awal – BTS 46 (Jl. Kapten A. Rivai) – BTS 21 (Jl. Angkatan 45) - BTS 18 (Jl. Orde Baru/ wilayah di sekitar Jl. Jend. Basuki Rachmad).

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari penelitian ini didapatkan kesimpulan sebagai berikut :

1. Tampilan desain pada aplikasi sudah sesuai dengan perancangan tampilan, baik pada tampilan menu input nama BTS maupun pada menu *Floyd Warshall*.
2. Proses penentuan rute terpendek pada aplikasi telah berhasil, karena menghasilkan indikator yang sama antara hasil penentuan rute terpendek yang dihasilkan aplikasi dengan hasil perhitungan secara manual yang dihasilkan teori *Floyd Warshall*. Dimana rute terpendek yang dihasilkan algoritma *Floyd Warshall* untuk menuju BTS 18, BTS 21 dan BTS 46 adalah 1-4-3-2 yaitu sebesar 10,9 km.

5.2. Saran

1. Aplikasi ini membutuhkan kualitas jaringan internet telepon seluler yang baik agar dapat mengakses rute terpendek.
2. Penggunaan aplikasi ini lebih optimal apabila digunakan diluar gedung/bangunan, agar GPS lebih akurat dalam mendeteksi posisi saat ini.

DAFTAR PUSTAKA

- [1] PT. GCI Indonesia. "Products & Services" Internet : <http://www.gcidesign.co.id/index.php/products/>. [Aug 12, 2018]
- [2] B. Yulianto *et al.*, "Location based services: positioning techniques," *Provider*, vol. 12, no. 1, pp. 4982–4987, 2010.
- [3] D. T. Salaki, "Penentuan Lintasan Terpendek dari FMIPA ke Rektorat dan Fakultas lain di UNSRAT Manado menggunakan Algoritma Djikstra," *Ilm. Sains*, vol. 11, no. Djikstra, pp. 73–76, 2011.
- [4] J. T. Informatika, F. Teknik, and U. H. Oleo, "Aplikasi pencarian rute terpendek apotek di kota kendari menggunakan algoritma," vol. 3, no. 1, pp. 9–16, 2017.
- [5] R. M and R. F. Uzzy, "Penentuan Jalur Terpendek Menuju Cafe Di Kota Malang Menggunakan Metode Bellman-Ford dengan Location Based Service Berbasis Android," *J. Ilm. Teknol. dan Inf. ASIA*, vol. 8, no. 2, pp. 49–64, 2014.
- [6] M. S. Handaka, "Perbandingan Algoritma Dijkstra (Greedy), Bellman-Ford (BFS-DFS), dan Floyd-Warshall (Dynamic Programming) dalam Pengaplikasian Lintasan Terpendek pada Link-State Routing Protocol," 2011.
- [7] M. K. Ni Ketut Dewi Ari Jayanti, "Letak Parkir," *Semin. Nas. Inform.*, pp. 75–81, 2014.
- [8] R. Aprian and D. Novandi, "Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam Penentuan Lintasan Terpendek (Single Pair Shortest Path)," pp. 1–5, 2007.
- [9] R. Saputra, "Sistem Informasi Geografis Pencarian Rute Optimum Obyek Wisata Kota Yogyakarta Dengan Algoritma Floyd-Warshall," *J. Mat.*, vol. 14, pp. 19–24, 2011.
- [10] K. Surendro, "Pemanfaatan *Enterprise Architecture Planning* Untuk Perencanaan Strategis Sistem informasi," *J. Inform.*, vol. 8, no. 1, pp. 1–9, 2007.
- [11] F. Mahdia and F. Noviyanto, "Pemanfaatan Google Maps API untuk Pembangunan Sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile Web (Studi Kasus : Badan Penanggulangan Bencana Daerah Kota Yogyakarta)," *J. Sarj. Tek. Inform.*, vol. 1, no. 1, pp. 162–171, 2013.

NOMENKLATUR

Adapun nomenklatur dari semua persamaan matematika dalam penelitian ini antara lain :

- t arti dari titik/node awal pada lintasan graf
j arti dari titik/node akhir pada lintasan graf

BIODATA PENULIS**Vera Apriliani Nawagusti**

Vera, dilahirkan di Baturaja, 7 April 1997, mahasiswa aktif Teknik Elektro, Program Studi Teknik Telekomunikasi, Politeknik Negeri Sriwijaya.