

PENCARIAN RUTE TERPENDEK MENUJU LOKASI LEMBAGA SOSIAL MENGGUNAKAN ALGORITMA DIJKSTRA BERBASIS ANDROID PADA APLIKASI PEDULI SEKITAR

Agus Santoso¹⁾, Muhammad Ainur Rony, S.Kom, M.T.I²⁾

¹Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur

^{1,2}Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260

E-mail: agussantoso552@gmail.com¹⁾, ainur.rony@gmail.com²⁾

Abstrak

Aplikasi Peduli Sekitar merupakan aplikasi yang menjembatani antara seorang pengguna dengan suatu lembaga sosial. Ketika pengguna akan memberikan bantuan (barang atau jasa) langsung ke tempat lembaga sosial tersebut, pengguna mengalami kendala pada ketidaktahuannya akan lokasi lembaga sosial yang terdekat dari posisi pengguna. Selain itu pengguna juga tidak tahu mengenai rute terpendek menuju lokasi suatu lembaga sosial tersebut, hal ini yang mendorong penulis untuk mengembangkan aplikasi Peduli Sekitar berbasis Android dengan membuat pencarian lokasi suatu lembaga sosial untuk memudahkan pengguna dalam mengetahui lokasi dan rute menuju suatu lembaga sosial. Metode yang digunakan yaitu Algoritma Dijkstra. Algoritma Dijkstra digunakan untuk menghasilkan perhitungan jarak yang optimal menuju lokasi lembaga sosial yang dituju. Tujuan penulisan skripsi ialah untuk menghasilkan aplikasi mobile yang dapat mengetahui lokasi suatu lembaga sosial dan dapat menampilkan rute terpendek dari lokasi pengguna menuju lokasi suatu lembaga sosial. Berdasarkan uji coba yang dilakukan maka dapat disimpulkan bahwa aplikasi dapat membantu pengguna dalam mencari dan mengetahui lokasi suatu lembaga sosial serta dapat menampilkan rute terpendek dari lokasi pengguna menuju lokasi suatu lembaga sosial.

Kata Kunci: Android, Algoritma Dijkstra, Rute, Lembaga Sosial, Peduli Sekitar

1. PENDAHULUAN

Teknologi perangkat mobil phone saat ini dapat difungsikan berbagai macam, tidak seperti dahulu yang hanya dapat berfungsi sebagai pengiriman pesan pendek dan komunikasi suara. Dengan semakin berkembangnya zaman dan semakin bertambahnya kebutuhan layanan baru pada mobile phone, para vendor selalu membuat hal yang baru dengan mengembangkan teknologi baru bagi produknya. Teknologi Global Positioning System (GPS) merupakan salah satu teknologi yang dapat dimanfaatkan pada perangkat mobile phone. Salah satu contoh dalam memanfaatkan adanya teknologi GPS ialah seorang pengguna dapat mengetahui lokasi keberadaannya secara realtime.

Aplikasi Peduli Sekitar merupakan aplikasi yang menjembatani antara seorang pengguna (dermawan) dengan suatu lembaga sosial. Seorang pengguna aplikasi Peduli Sekitar yang akan memberikan bantuan (barang atau jasa) secara langsung ke tempat lembaga sosial harus melakukan komunikasi kepada penanggung jawab suatu lembaga sosial. Dalam proses pemberian bantuan tersebut, pengguna mengalami kendala pada ketidaktahuannya akan rute terpendek dari lokasi pengguna menuju suatu lokasi lembaga sosial.

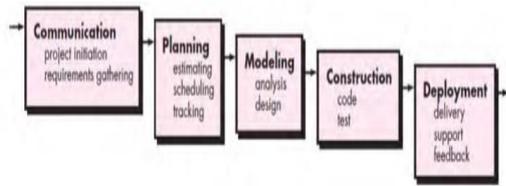
Berawal dari latar belakang diatas, penulis dapat menyimpulkan sebuah rumusan masalah ialah bagaimana cara aplikasi dapat menampilkan lokasi

lembaga sosial dan menampilkan rute terpendek menuju lokasi lembaga sosial kepada pengguna. Hal ini yang mendorong penulis untuk mengembangkan aplikasi Peduli Sekitar berbasis Android dengan membuat pencarian rute terpendek menuju lokasi lembaga sosial. Metode yang digunakan ialah Algoritma Dijkstra. Algoritma Dijkstra digunakan untuk menghasilkan perhitungan jarak yang optimal menuju lokasi lembaga sosial yang dituju. Tujuan membuat pencarian rute terpendek menuju lokasi lembaga sosial yaitu untuk memudahkan pengguna dalam mengetahui lokasi suatu lembaga sosial dan mengetahui rute terpendek menuju lokasi suatu lembaga sosial.

2. LANDASAN TEORI

2.1 Metodologi

Pada pembuatan aplikasi Pencarian Lokasi Suatu Lembaga Sosial digunakan teknik pengembangan aplikasi dengan metode Waterfall. Waterfall Model adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software* [1]. Berikut ini fase-fase dalam Waterfall Model:



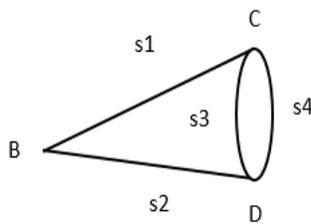
Gambar 1. Waterfall Model

2.2 Teori Graf

Graf adalah kumpulan simpul (*node*) yang dihubungkan satu sama lain melalui sisi/busur (*edges*). Suatu Graf *G* terdiri dua himpunan yaitu himpunan *V* dan himpunan *E*.

- a. Vertex (simpul) – *V* = Himpunan simpul yang terbatas dan tidak kosong.
- b. Edge (sisi/busur) – *E* = Himpunan busur yang menghubungkan sepasang simpul.

Dapat dikatakan graf adalah kumpulan dari simpul-simpul yang dihubungkan oleh sisi-sisi.



Gambar 2. Graf

Vertex (*V*) = {*B*, *C*, *D*}
 Edge (*E*) = {*s1*, *s2*, *s3*, *s4*}

2.3 Algoritma Dijkstra

Algoritma Dijkstra ditemukan oleh seorang ahli yang bernama Edger W. Dijkstra. Algoritma Dijkstra ialah salah satu bentuk algoritma yang terkenal untuk memecahkan sebuah masalah optimasi dan memiliki sifat sederhana. Algoritma Dijkstra merupakan algoritma greedy yang berfungsi untuk mencari jarak terpendek dalam sebuah graph berarah yang memiliki bobot-bobot edge dengan nilai tak negatif. Arti kata greedy secara harfiah yaitu tamak atau rakus, dalam arti yang positif, Algoritma Greedy hanya mencari solusi terbaik yang akan digunakan pada setiap langkah, dengan tanpa memikirkan akibat kedepannya.

Node (simpul/titik) digunakan pada Algoritma Dijkstra, karena pada Algoritma Dijkstra menggunakan graf berarah dalam menentukan rute terpendek. Algoritma Dijkstra memiliki tujuan untuk menemukan rute terpendek berdasarkan pada bobot terkecil dari satu *node* ke *node* lainnya. Misal *node* menggambarkan sebuah gedung dan *edge* menggambarkan sebuah jalan, maka Algoritma Dijkstra melakukan perhitungan terhadap semua kemungkinan bobot terkecil dari setiap *node*.

Prosedur Algoritma Dijkstra sebagai berikut:

Masukan: graf terhubung dan berarah $G = (V,E)$, matriks bobot *C*, titik awal *v0*
 Keluaran: jarak terpendek dari titik *v0* ke titik lain dalam *D*

Prosedur Dijkstra (*G,C,v0*)

Mulai

$S \leftarrow \{v0\}$;

$D[v0] \leftarrow 0$;

Untuk masing-masing *v* dalam $V - \{v0\}$ lakukan

$D[v] \leftarrow C[v0,v]$;

Selama $S \neq V$ lakukan

Mulai

Pilih salah satu titik *w* di $V - S$ dimana $D[w]$ adalah minimum;

$S \leftarrow S \cup \{w\}$;

Untuk masing-masing *v* di $V - S$ lakukan

$D[v] \leftarrow \min(D[v], D[w] + C[w,v])$;

Selesai

[2].

Berikut ini urutan logika pada Algoritma Dijkstra:

- a. Berilah nilai bobot (jarak) untuk setiap titik ke titik lainnya, lalu atur atau set nilai 0 pada *node* awal dan nilai tak hingga terhadap *node* lain (belum terisi),
- b. Atur semua titik “belum terpilih” dan atur titik awal sebagai “titik keberangkatan”,
- c. lalu hapus data yang lama (yang sebelumnya terekam), kemudian simpan ulang data jarak dengan sebuah jarak yang baru,
- d. Ketika telah selesai dalam mempertimbangkan setiap jarak pada titik tetangga, tandai titik yang terpilih sebagai “titik terpilih”. Titik yang telah terpilih tidak akan dicek kembali, jarak yang disimpan ialah jarak terakhir dan yang paling minimum jaraknya,
- e. Atur “titik belum terpilih” dengan jarak paling minimum (dari titik keberangkatan), menjadi “titik keberangkatan” selanjutnya. Kemudian lanjutkan kembali ke langkah 3.

2.4 Google Maps

Google Maps merupakan layanan peta secara online dari Google. Layanan *Google Maps* dapat menampilkan informasi geografis hingga hampir seluruh lokasi di bumi. Layanan *Google Maps* bersifat interaktif, karena peta dapat digeser ataupun digerak-gerakkan sesuai keinginan pengguna, dapat juga mengubah tingkat zoom serta mengubah tampilan peta. Fasilitas yang terdapat pada *Google Maps* diantaranya menjajah peta, mencari lokasi tertentu sekaligus mencari jalur atau rute menuju lokasi tersebut seperti hotel, tempat hiburan, tempat wisata, lokasi bisnis, sekolah dan lain-lain.

2.5 Global Positioning System (GPS)

GPS adalah singkatan dari *Global Positioning System*, yang merupakan sistem navigasi yang menggunakan teknologi satelit yang dapat menerima sinyal dari satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke bumi. Sinyal ini diterima oleh alat penerima (*receiver*) di permukaan, dimana GPS *receiver* ini akan mengumpulkan informasi dari satelit GPS, yaitu:

- a. Waktu
GPS *receiver* menerima informasi waktu dari jam atom yang mempunyai keakurasian sangat tinggi.
- b. Lokasi
GPS memberikan informasi lokasi dalam tiga dimensi yaitu *Latitude*, *Longitude* dan Elevasi.
- c. Kecepatan
Ketika berpindah tempat, GPS dapat menunjukkan informasi kecepatan berpindah tersebut.
- d. Arah perjalanan
GPS dapat menunjukkan arah tujuan.
- e. Simpan lokasi
Tempat-tempat yang sudah pernah atau ingin dikunjungi bisa disimpan oleh GPS *receiver*.
- f. Komulasi data
GPS *receiver* dapat menyimpan informasi *track*, seperti total perjalanan yang sudah pernah dilakukan, kecepatan rata-rata, kecepatan paling tinggi, kecepatan paling rendah, waktu/jam sampai tujuan, dan sebagainya[3].

Posisi lokasi yang ditunjukkan oleh GPS dapat terjadi sebuah kesalahan atau yang dikenal dengan tingkat akurasi. Berikut beberapa hal yang dapat mempengaruhi tingkat akurasi pada suatu GPS:

- a. Kesalahan *Ephemeris*
Terjadi jika satelit tidak dapat mentransmisikan posisinya di orbit dengan tepat.
- b. Keadaan *Ionosphere*
Ionosphere berada pada jarak sekitar 43-50 mil diatas permukaan bumi. Satelit yang melewati *ionosphere* akan menjadi lambat dikarenakan adanya plasma (gas dengan tingkat kepadatan rendah). Walaupun GPS *receiver* berusaha untuk mengkoreksi/ memperbaiki faktor keterlambatan yang terjadi tetap saja aktivitas tertentu dari plasma bisa menyebabkan kesalahan perhitungan.
- c. Keadaan *Troposphere*
Troposphere adalah bagian terendah dari atmosfer sampai dengan ketinggian

sekitar 11 mil dari permukaan tanah. Variasi pada temperatur, tekanan, dan kelembaban bisa menyebabkan perbedaan kecepatan penerimaan gelombang radio.

- d. Kesalahan Waktu
Karena penempatan jam atom pada setiap GPS *receiver* tidak berjalan sebagaimana mestinya. Kesalahan waktu dari GPS *receiver* yang tidak presisi dapat menimbulkan ketidakakurasian.
- e. Kesalahan *Multipath*
Terjadi karena sinyal satelit membentur permukaan keras (seperti bangunan atau tebing) sebelum mencapai GPS *receiver*. Hal tersebut bisa menyebabkan terjadinya *delay* sehingga perhitungan jarak menjadi tidak akurat.
- f. Buruknya Sinyal Satelit
Keadaan langit yang terhalang akan menyebabkan GPS sulit menerima data satelit[3].

Cara kerja GPS secara logika ada lima langkah sebagai berikut:

- a. Memakai perhitungan "*triangulation*" dari satelit.
- b. Untuk perhitungan "*triangulation*", GPS mengukur jarak menggunakan *travel time* sinyal radio.
- c. Untuk mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi.
- d. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
- e. Terakhir harus mengoreksi *delay* sinyal waktu perjalanan di atmosfer sampai diterima *receiver*.

3. ANALISIS DAN PERANCANGAN APLIKASI

3.1 Analisis Masalah

Dalam penggunaan aplikasi Pencarian Rute Terpendek Menuju Lokasi Lembaga Sosial pada Peduli Sekitar, saat pengguna (dermawan) akan memberikan bantuan (barang atau jasa) langsung ke tempat lembaga sosial tersebut, pengguna mengalami kendala pada ketidaktahuan akan lokasi lembaga sosial yang terdekat dari posisi pengguna. Selain itu pengguna juga tidak tahu mengenai rute terpendek menuju lokasi suatu lembaga sosial tersebut.

3.2 Solusi

Untuk mengatasi masalah yang muncul, oleh karena itu penulis mengembangkan aplikasi Peduli Sekitar berbasis Android dengan membuat pencarian rute menuju lokasi suatu lembaga sosial dengan tujuan memudahkan pengguna dalam mengetahui lokasi lembaga sosial yang terdekat dari lokasi keberadaan pengguna serta mengetahui rute

terpendek menuju lokasi suatu lembaga sosial tersebut dengan menggunakan Algoritma Dijkstra.

3.3 Rancangan Basis Data

Dalam pembuatan aplikasi Pencarian Rute Terpendek Menuju Lokasi Lembaga Sosial dibutuhkan basis data untuk menyimpan data yang dibutuhkan dalam menjalankan aplikasi. Pembuatan basis data tersebut dibutuhkan rancangan. Spesifikasi basis data sebagai berikut:

- a. Nama Tabel : graph
- Media : Harddisk
- Isi : Data Graph
- Primary Key : id

Tabel 2. Spesifikasi Tabel Graph

Field	Type	Length
id	Integer Auto Increment	11
simpul_awal	Varchar	10
simpul_tujuan	Varchar	10
jalur	Text	400
bobot	Double	8,8

- b. Nama Tabel : jalur_mobil
- Media : Harddisk
- Isi : Data Jalur Mobil
- Primary Key : id

Tabel 3. Spesifikasi Tabel Jalur Mobil

Field	Type	Length
id	Integer Auto Increment	11
No_trayek	Varchar	10
simpul	Text	50

- c. Nama Tabel : jalur_mobil
- Media : Harddisk
- Isi : Data Yayasan
- Primary Key : id

Tabel 4. Spesifikasi Tabel Yayasan

Field	Type	Length
id	Integer Auto Increment	11
tujuan	Varchar	50
koordinat	Text	20

3.4 Rancangan Layar Aplikasi

Sebuah tatap muka atau *interface* diperlukan untuk memudahkan pengguna dalam menggunakan aplikasi. Berikut ini adalah rancangan layar untuk Pencarian Rute Terpendek Menuju Lokasi Lembaga Sosial pada Aplikasi Peduli Sekitar.

Gambar 3 berikut ini merupakan rancangan layar login aplikasi yang digunakan untuk masuk ke aplikasi dimana pengguna harus memasukan *email* dan *password* terlebih dahulu.



Gambar 3. Rancangan Layar Login

Kemudian Gambar 4 merupakan rancangan layar menu utama yaitu halaman utama aplikasi terdapat menu diantaranya *home*, *cari kebutuhan*, *tentang*, *ubah data diri*, *ubah password* dan *logout*. Menu *home* digunakan untuk menampilkan peta dan lokasi lembaga sosial dengan muncul icon peduli sekitar. Selain itu pada menu *home* juga terdapat fungsi mencari lokasi yang digunakan untuk mencari lembaga sosial berdasarkan lokasi yang diinginkan. Menu *Cari Kebutuhan* digunakan untuk mencari lembaga sosial berdasarkan kebutuhan yang diinginkan. Menu *tentang* merupakan halaman yang berisi logo peduli sekitar, versi aplikasi, dan alamat website peduli sekitar. Menu *Ubah Data Diri* digunakan untuk mengubah data pengguna akun aplikasi diantaranya nama lengkap, nama panggilan, email, no telepon, tempat dan tanggal lahir. Menu *Ubah Password* digunakan pengguna untuk mengubah password akun aplikasi. Dan Menu *Logout* digunakan pengguna untuk keluar dari aplikasi.



Gambar 4. Rancangan Layar Menu Utama

Kemudian Gambar 5 merupakan rancangan layar lihat lembaga sosial. Pada halaman ini menampilkan data lembaga sosial. Data lembaga sosial yang terdapat pada menu lihat lembaga sosial diantaranya nama lembaga sosial(yayasan), nama penanggung jawab, nomor telepon, nomor *handphone*, *email*, *website*, alamat lokasi lembaga sosial, alamat sosial media seperti *facebook*, *twitter* dan *instagram* dan data lainnya.



Gambar 5. Rancangan Layar Lihat Lembaga Sosial

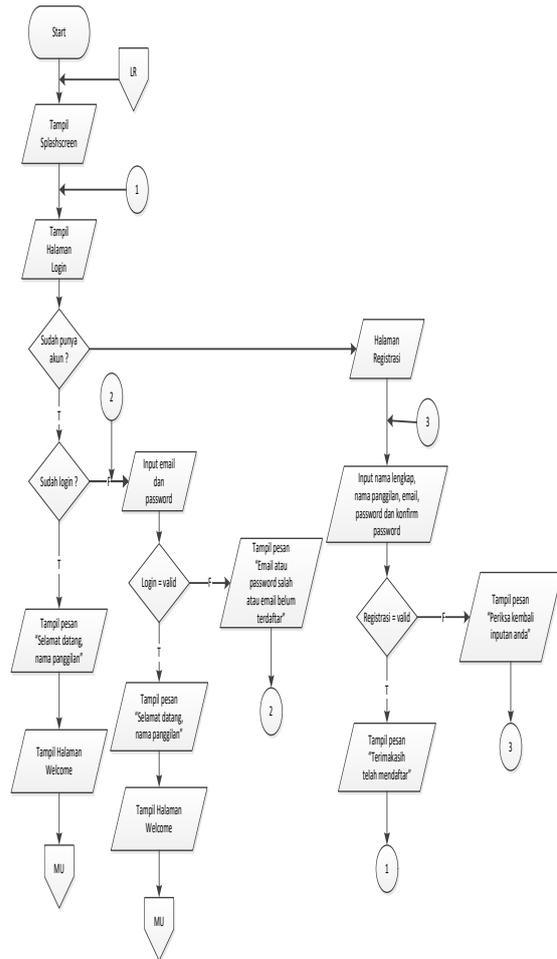
Kemudian Gambar 6 merupakan rancangan layar temukan rute terpendek. Pada halaman ini digunakan untuk menampilkan rute terpendek dari lokasi keberadaan pengguna menuju lokasi suatu lembaga sosial.



Gambar 6. Rancangan Layar Temukan Rute Terpendek

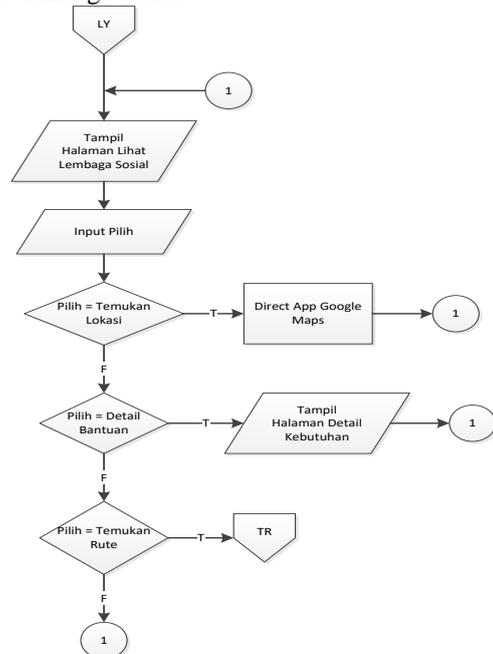
3.5 Flowchart

Pada Gambar 7 menggambarkan proses yang terjadi pada halaman login dan register. Jika pengguna belum mempunyai akun aplikasi atau pengguna merupakan pengguna baru maka pengguna harus melakukan proses registrasi terlebih dahulu. Setelah proses registrasi berhasil, maka akan tampil pesan berhasil mendaftar. Dan sebaliknya jika proses registrasi tidak berhasil, maka akan tampil pesan *error*. Selanjutnya seorang pengguna harus *login* terlebih dahulu dengan memasukkan *email* dan *password* untuk dapat masuk halaman utama aplikasi. Jika proses *login* berhasil, maka selanjutnya tampil pesan selamat datang dan masuk ke menu utama aplikasi. Dan sebaliknya jika proses *login* tidak berhasil, maka akan tampil pesan *error*.



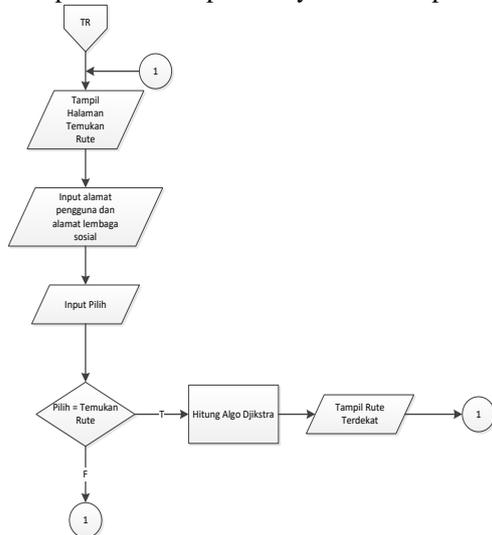
Gambar 7. Flowchart Halaman Login dan Register

Gambar 8 menggambarkan proses pada halaman lihat lembaga sosial.



Gambar 8. Flowchart Halaman Lihat Lembaga Sosial

Kemudian Gambar 9 menggambarkan proses yang terjadi pada halaman temukan rute terpendek. Pada halaman ini pengguna terlebih dahulu diharuskan memilih *list* lembaga sosial tujuan, kemudian pengguna meng-tap peta untuk menunjuk lokasi tepat keberadaan pengguna. Setelah itu aplikasi melakukan perhitungan menggunakan algoritma Dijkstra, kemudian menampilkan rute terpendeknya ke dalam peta.

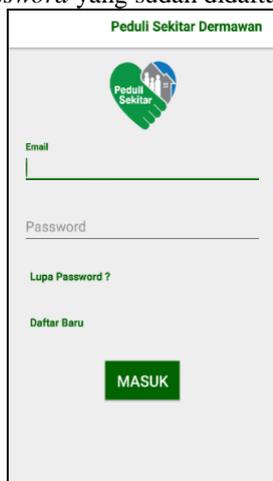


Gambar 9. Flowchart Halaman Temukan Rute Terpendek

4. HASIL DAN PEMBAHASAN

4.1 Tampilan Layar

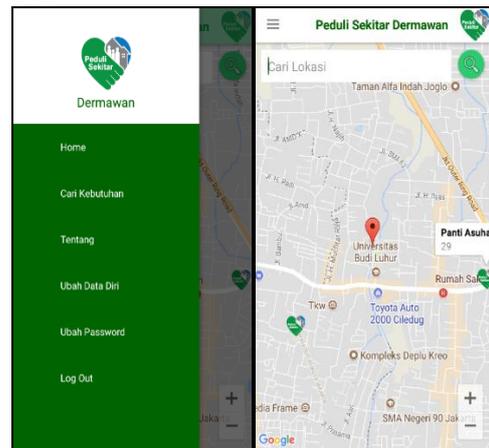
Gambar 10 merupakan tampilan layar login pada aplikasi. Pengguna harus melakukan proses login sebelum masuk ke aplikasi dengan memasukkan *email* dan *password* yang sudah didaftarkan.



Gambar 10. Tampilan Layar Login

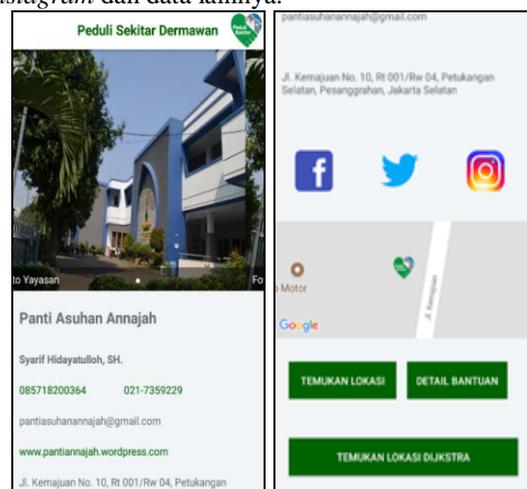
Gambar 11 merupakan tampilan layar menu utama yang menampilkan berbagai menu yang ada pada aplikasi diantaranya menu *home*, cari kebutuhan, ubah data diri, ubah *password* dan *logout*. Menu *home* digunakan untuk menampilkan peta dan lokasi lembaga sosial dengan muncul icon peduli

sekitar. Selain itu pada menu *home* juga terdapat fungsi mencari lokasi yang digunakan untuk mencari lembaga sosial berdasarkan lokasi yang diinginkan. Menu Cari Kebutuhan digunakan untuk mencari lembaga sosial berdasarkan kebutuhan yang diinginkan. Menu tentang merupakan halaman yang berisi logo peduli sekitar, versi aplikasi, dan alamat website peduli sekitar. Menu Ubah Data Diri digunakan untuk mengubah data pengguna akun aplikasi diantaranya nama lengkap, nama panggilan, *email*, no telepon, tempat dan tanggal lahir. Menu Ubah Password digunakan pengguna untuk mengubah password akun aplikasi. Dan Menu Logout digunakan pengguna untuk keluar dari aplikasi.



Gambar 11. Tampilan Layar Menu Utama

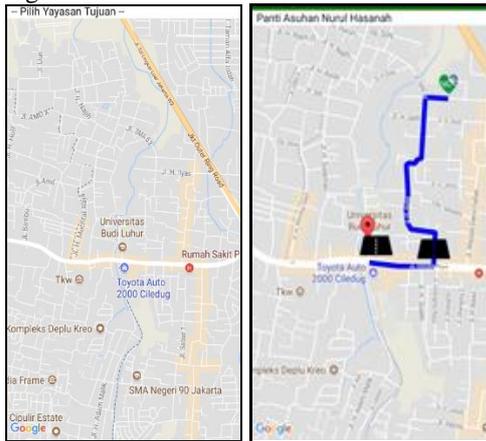
Gambar 12 merupakan tampilan layar lihat lembaga sosial merupakan tampilan yang berisi data lembaga sosial yang bersangkutan diantaranya, nama lembaga sosial, penanggung jawab, no telepon, no *handphone*, *email*, alamat lokasi lembaga sosial dan alamat sosial media seperti *facebook*, *twitter* dan *instagram* dan data lainnya.



Gambar 12. Tampilan Layar Lihat Lembaga Sosial

Gambar 13 merupakan tampilan layar temukan rute terpendek untuk menampilkan rute terpendek

dari lokasi keberadaan pengguna menuju lokasi suatu lembaga sosial.



Gambar 13. Tampilan Layar Temukan Rute Terpendek

4.2 Pengujian Program

Penulis melakukan pengujian terhadap perhitungan Algoritma Dijkstra pada peta, apakah Algoritma Dijkstra dapat menyelesaikan dan menampilkannya dalam peta. Pada pengujian ini penulis melakukan uji tes sebagai berikut:

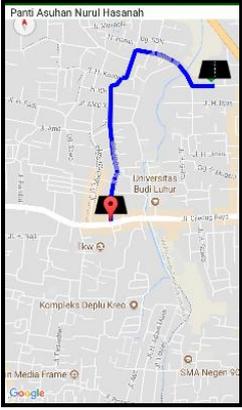
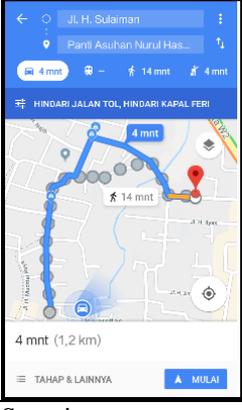
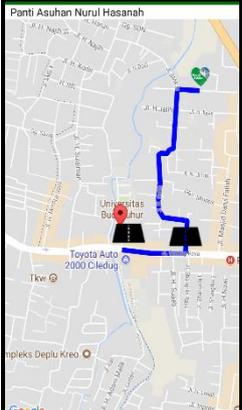
Tabel 5. Tabel Uji Coba

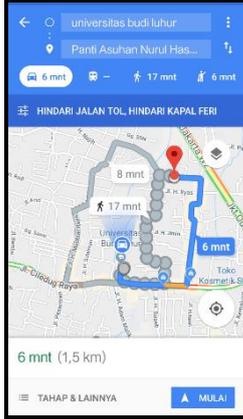
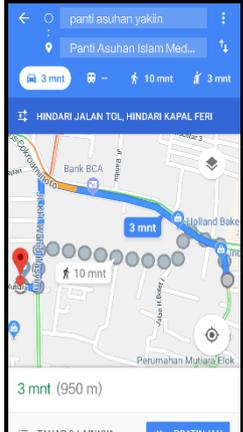
a.	Koordinat Awal	-6.2340904, 106.743381 (Jalan Haji Sulaiman)
	Koordinat Tujuan	-6.230083027450176, 106.7504945397377 (Panti Asuhan Nurul Hasanah)
	Relasi Vertex dan Jarak (meter)	RV1= 17-18-19-3 (839,9+185,3+504,1 = 1.529,3m) RV2= 17-0-16-3 (275+291,9+1006= 1.572,9m) RV3= 17-0-16-2-3 (275+291,9+148,9+950 = 1.665,8 m)
	Gambar	
b.	Koordinat Awal	-6.2355478, 106.7451385 (Universitas Budi Luhur)
	Koordinat	-6.230083027450176,

	Tujuan	106.7504945397377 (Panti Asuhan Nurul Hasanah)
	Relasi Vertex dan Jarak (meter)	RV1= 0-17-18-19-3 (275+839,9+185,3+504,1= 1.804,3 m) RV2= 0-16-3 (291,9+1006 = 1.297,9 m) RV3= 0-16-2-3 (291,9+148,9+950= 1.390,8 m)
	Gambar	
c.	Koordinat Awal	-6.237207514949379, 106.7435073852539 (Panti Asuhan Yakiin)
	Koordinat Tujuan	-6.237215513945321, 106.73851042985916 (Panti Asuhan Media Kasih)
	Relasi Vertex dan Jarak (meter)	RV1=5-4-20-21-7 (110,9+199,6+373,4+91,7 = 775,6 m) RV2= 5-4-6-7 (110,9+563,9+302,3 = 977,1 m)
	Gambar	

Tabel 6. Perbandingan dengan hasil Google Maps

a.	Koordinat	Awal : Jalan Haji Sulaiman
----	-----------	----------------------------

	Awal dan Tujuan	Tujuan : Panti Asuhan Nurul Hasanah
	Gambar Hasil Algoritma Dijkstra	
	Gambar Hasil Google Maps	
	Sesuai atau Tidak sesuai	Sesuai
b.	Koordinat Awal dan Tujuan	Awal : Universitas Budi Luhur Tujuan : Panti Asuhan Nurul Hasanah
	Gambar Hasil Algoritma Dijkstra	

	Gambar Hasil Google Maps	
	Sesuai atau Tidak sesuai	Tidak sesuai
c.	Koordinat Awal dan Tujuan	Awal : Panti Asuhan Yakiin Tujuan : Panti Asuhan Media Kasih
	Gambar Hasil Algoritma Dijkstra	
	Gambar Hasil Google Maps	
	Sesuai atau Tidak sesuai	Tidak sesuai

4.3 Evaluasi Program

Yang dimaksud evaluasi program yaitu untuk menganalisa hasil yang telah dicapai oleh program yang dikembangkan. Program ini tentu saja mempunyai kelebihan dan kekurangan. Adapun kelebihan dan kekurangan program yang dikembangkan sebagai berikut:

a. Kelebihan Program

Ada beberapa hal yang dapat dijadikan catatan, diantaranya adalah:

- 1) Aplikasi ini dapat menampilkan lokasi pengguna dan lokasi lembaga sosial di peta secara *realtime*.
- 2) Aplikasi ini dapat menampilkan rute atau jalur lokasi pengguna menuju lokasi lembaga sosial tujuan.
- 3) Aplikasi dapat membantu pengguna ketika ingin mendapatkan informasi rute atau jalur menuju lembaga sosial terdekat dalam memberikan bantuan terhadap lembaga sosial yang bersangkutan.

b. Kekurangan Program

Kekurangan pada aplikasi ini adalah Algoritma Dijkstra pada aplikasi ini hanya menghitung pada area atau tempat yang sudah terdefinisi, sehingga jika terdapat area baru yang belum terdefinisi seperti *node* dan *graf*-nya, maka algoritma tidak dapat berjalan sebagaimana mestinya.

5. KESIMPULAN

Dari keseluruhan bab di atas, dalam merancang dan membangun aplikasi Pencarian Rute Terpendek Menuju Lokasi Lembaga Sosial pada Peduli Sekitar menggunakan Algoritma Dijkstra terdapat kesimpulan yang diambil yaitu aplikasi ini dapat menampilkan rute terpendek menuju lokasi lembaga sosial sehingga membantu pengguna (dermawan) dalam menemukan rute terpendek di peta dari keberadaan pengguna menuju lokasi lembaga sosial tujuan.

Dalam aplikasi ini, Algoritma Dijkstra hanya menghitung pada area atau tempat yang sudah terdefinisi, sehingga jika terdapat area baru yang belum terdefinisi seperti *node* dan *graph*-nya, maka algoritma tidak dapat berjalan sebagaimana mestinya.

6. DAFTAR PUSTAKA

- ^[1]Pressman, R. (2012). *Rekayasa Perangkat Lunak*. Bandung: Andi Publisher.
- ^[2]Satyananda, D. (2012). *Struktur Data*. Malang: Universitas Negeri Malang.
- ^[3]Wisnu. (2012). *GPS Pada Android*. Jakarta: Jasakom