JURNAL

SISTEM REKOMENDASI RUTE TERPENDEK MENGGUNAKAN METODE DJIKSTRA

Oleh:

YULY NURHIDAYATI

12.1.03.02.0288

Dibimbing oleh :

1. **Rini Indriati, S.Kom.,M.Kom.**
2. **Ahmad Bagus Setiawan, S.T.,M.M.,M.Kom.**

PROGRAM STUDI

FAKULTAS TEKNIK INFORMATIKA

UNIVERSITAS NUSANTARA PGRI KEDIRI

TAHUN 2017

Skripsi Oleh :

YULY NURHIDAYATI
NPM : 12.1.03.02.0288

Judul:

SISTEM REKOMENDASI RUTE TERPENDEK MENGGUNAKAN METODE DJIKSTRA

Telah dipertahankan di depan Panitia Ujian/Sidang Skripsi
Program Studi Teknik Informatika FT UN PGRI Kediri
Pada tanggal : 18 Januari 2017

Dan Dinyatakan telah memenuhi Persyaratan

Panitia Penguji :

1. Ketua : Rini Indriati, S.Kom., M.Kom
2. Penguji I : Ardi Sanjaya, M.Kom
3. Penguji II : Ahmad Bagus Setiawan, S.T., M.M., M.Kom

Mengetahui,
Dekan FT

Dr. Suryo Widodo, M.Pd.
NIP.19640202 199103 1 002

iii

SURAT PERNYATAAN
ARTIKEL SKRIPSI TAHUN 2017

Yang bertanda tangan di bawah ini:

Nama Lengkap	: Yuly Nurhidayati
NPM	: 12.1.03.02.0288
Telepun/HP	: 085735971354
Alamat Surel (Email)	: yulynur2013@gmail.com
Judul Artikel	: Sistem Rekomendasi Rute Terpendek Menggunakan Metode Djikstra
Fakultas – Program Studi	: Teknik-Teknik Informatika
Nama Perguruan Tinggi	: UNIVERSITAS NUSANTARA PGRI KEDIRI
Alamat Perguruan Tinggi	: JL. K.H Ahmad Dahlan No. 76, Mojoroto, Kediri, Jawa Timur 64112

Dengan ini menyatakan bahwa :

a. artikel yang saya tulid merupakan karya saya pribadi (bersama tim penulis) dan bebas plagiarisme;
b. artikel telah diteliti dan disetujui untuk diterbitkan oleh Dosen Pembimbing I dan II.

Demikian surat pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari ditemukan ketidaksesuaian data dengan pernyataan ini dan atau ada tuntutan dari pihak lain, saya bersedia bertanggungjawab dan diproses sesuai dengan ketentuan yang berlaku.

Mengetahui		Kediri, 24 Januari 2017
Pembimbing I	Pembimbing II	Penulis,
Rini Indriati, S.Kom.,M.Kom. NIDN. 0725057003	Ahmad Bagus Setiawan, S.Kom.,M.M.,M.Kom. NIDN. 0703018704	Yuly Nurhidayati 12.1.03.02.0288

SISTEM REKOMENDASI RUTE TERPENDEK MENGGUNAKAN METODE DJIKSTRA

Yuly Nurhidayati
12.1.03.02.0288
Fakultas Teknik – Teknik Informatika
Yulynur2013@gmail.com
Rini Indriati, S.Kom.,M.Kom. dan Ahmad Bagus Setiawan, S.Kom.,M.M.,M.Kom.
UNIVERSITAS NUSANTARA PGRI KEDIRI

ABSTRAK

Yuly Nurhidayati: Sistem Rekomendasi Rute Terpendek Menggunakan Metode Djikstra, Skripsi, Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri, 2016

Usaha *laundry* adalah usaha yang bergerak dibidang jasa cuci dan setrika. Berkembangnya bisnis *laundry* kiloan menjadikan persaingan di sektor ini menjadi semakin ketat. Untuk menjaga agar usaha ini tidak sepi oleh pelanggan penyedia jasa *laundry* harus memiliki ciri khas untuk menarik pelanggan salah satunya menyediakan jasa antar jemput pakaian. Oleh sebab itu, diperlukan sebuah aplikasi sistem pengambilan keputusan yang dapat membantu merekomendasikan rute terpendek. Tujuan untuk merekomendasikan rute terpendek menggunakan metode *djikstra* ada untuk mempermudah sistem pengambilan keputusan.

Algoritma *Dijkstra* sebagai metode pencarian rute terpendek. Algoritma *Dijkstra* merupakan algoritma penelusuran yang menyelesaikan permasalahan rute terpendek dengan satu sumber asal untuk suatu *node* dengan nilai sisi non negatif, menghasilkan pohon jalur terpendek.

Sistem ini dapat memberikan informasi mengenai urutan *customer* yang terdekat sampai yang terjauh yang harus dikunjungi menggunakan algoritma *Dijkstra*, dan memberikan informasi rute jalan yang bisa dilewati menggunakan data dari *OpenStreetMaps*. Sistem juga dapat memberikan nilai jarak tempuh dan waktu tempuh. Untuk pengembangan lebih lanjut sistem *E-Laundry* diharapkan dapat digunakan pada multi *platfom.*

KATA KUNCI : rute, jarak terpendek, *openstreetmaps*, metode *djikstra.*

I. PENDAHULUAN

A. LATAR BELAKANG

Usaha laundry adalah usaha yang bergerak dibidang jasa cuci dan setrika. Keberadaan jasa cuci mencuci dan setrika sudah menjadi bagian dari kebutuhan hidup manusia. Berkembangnya bisnis laundry kiloan menjadikan persaingan di sektor ini menjadi semakin ketat. Untuk menjaga agar usaha ini tidak sepi oleh pelanggan setiap penyedia jasa laundry memiliki ciri khas dan cara promosi masing-masing, seperti menyediakan jasa antar jemput cucian (Anonimous, 2013).

Untuk menyediakan layanan antar jemput cucian juga tidak mudah dikarenakan kurir yang belum tentu mengetahui rute yang dituju sehingga dapat menyebabkan melebarnya rute yang dia lalui. Oleh sebab itu, terjadilah pemborosan waktu dan bahan bakar. Untuk menyeleseikan masalah ini maka dibutuhkan aplikasi yang dapat memberikan rekomendasi kepada kurir sehingga dapat melewati rute terpendek menuju ke pelanggan.

Salah satu cara mencari rute terpendek ialah menggunakan algoritma Dijkstra sebagai metode pencarian rute terpendek. Alasan memilih algoritma Dijkstra karena algoritma tersebut merupakan algoritma penelusuran graf yang menyelesaikan permasalahan rute terpendek dengan satu sumber asal untuk suatu graf dengan nilai sisi non negatif, menghasilkan pohon jalur terpendek (Rutter, 2009). Berdasarkan latar belakang maka dilakukan penelitian dengan judul : Sistem Recomendasi Rute Terpendek Menggunakan Algoritma Djikstra.

B. RUMUSAN MASALAH

Rumusan masalah pada penelitian ini adalah merancang dan membuat sistem recomendasi rute terpendek yang efisien guna memperoleh jalur terdekat dengan menggunakan algoritma djikstra.

C. BATASAN MASALAH

Batasan masalah untuk penelitian ini adalah sebagai berikut :

1. Sistem yang digunakan adalah berbasis android dan website.
2. DBMS yang digunakan adalah MySQL dan SQLite.
3. Lokasi penelitian berada di Kota Kediri dan Kab. Nganjuk.
4. Penentuan rute untuk mencari jarak terdekat menuju lokasi tujuan menggunakan algoritma Dijkstra.
5. Sistem client dijalankan pada platform Android.

6. Tidak membahas aspek-aspek keamanan komunikasi data dari sisi client maupun server.
7. Rute jalan dari lokasi awal ke lokasi tujuan diperoleh dari Open Street Maps.
8. Kecepatan kendaraan konstan, dengan tidak memperhitungkan kemacetan, rel kereta api, lalu lintas dan penutupan jalan.
9. Rute jalan yang paling umum dilalui dan dikenali oleh Open Street Maps.

D. TUJUAN PENELITIAN

Adapun tujuan dari pembuatan sistem ini adalah menghasilkan sistem penentu rute terdekat menuju pesanan pelanggan, serta memberikan saran perjalanan yang efektif dan efisien.

II. METODE

A. TEORI

1. Teori Graf

Suatu Graf $G=(V,E)$ didefinisikan sebagai pasangan himpunan sisi dan simpul dengan $V(G)$ = Himpunan simpul $\{v1, v2, \ldots, vn\}$ dan $E(G)$ = Himpunan sisi $\{e1, e2, \ldots, en\}$. Setiap sisi berhubungan dengan satu atau dua simpul. Dua buah simpul dikatakan berhubungan atau bertetangga (*adjacent*) jika ada sisi yang menghubungkan keduanya. Berdasarkan orientasi yang ada pada sisinya, graf dapat dikelompokkan menjadi dua yaitu: Graf berarah (*direct graf*) yaitu graf yang setiap sisinya diberikan arah sehingga untuk dua simpul *vi* dan *vj*, maka (*vi,vj vj,vi*) dan graf tak berarah (*undirect graf*) yaitu graf yang sisinya tidak mengandung arah sehingga untuk dua simpul *vi* dan *vj*, maka (*vi,vj*) (*vj,vi*). Selain itu juga dikenal graf berbobot yaitu graf yang sisinya memiliki bobot atau (Ahuja *et al*, 1993).

2. Algoritma Djisktra

Kata algoritma berasal dari nama Abu Ja'far Muhammad Ibnu Musa Al-Khawarizmi Al-Khawarizmi dibaca orang Barat menjadi *Algorism*. Kata ini kemudian berubah menjadi *algorithm* karena terpengaruh kata arithmetic, dan di Indonesia kata ini menjadi algoritma. Algoritma adalah langkah-langkah logis yang diperlukan dalam menyelesaikan suatu masalah. (Inspirasi Al-quran Dalam Algoritma Alami, Fatchurrohman dkk, 2006)

Dari pengertian tersebut dapat pula dikatakan bahwa tujuan dari penggunaan algoritma adalah untuk mendapatkan petunjuk dalam menyelesaikan suatu permasalahan. Pada dasarnya, algoritma dijkstra merupakan salah satu bentuk

algoritma *greedy*. Algoritma ini termasuk algoritma pencarian graf yang digunakan untuk menyelesaikan masalah lintasan terpendek dengan satu sumber pada sebuah graf yang tidak memiliki cost sisi negative, dan menghasilkan sebuah pohon lintasan terpendek. Algoritma ini sering digunakan pada *routing*. Algoritma dijkstra mencari lintasan terpendek dalam sejumlah langkah. Algoritma ini menggunakan strategi *greedy* sebagai berikut:

a. Untuk setiap simpul sumber (*source*) dalam graf, algoritma ini akan mencari jalur dengan cost minimum antara simpul tersebut dengan simpul lainnya.
b. Algoritma ini juga dapat digunakan untuk mencari total biaya (*cost*) dari lintasan terpendek yang dibentuk dari sebuah simpul ke sebuah simpul tujuan.

Dijkstra menemukan jalur dengan biaya terendah (yaitu rute terpendek) antara simpul tersebut dengan setiap simpul lainnya.Algoritma ini juga dapat digunakan untuk menemukan jalur terpendek dari simpul asal ke simpul tujuan dengan cara menghentikan algoritma ketika jalur terpendek ke simpul tujuan telah ditentukan (Rutter, 2009).

Berikut ini adalah algoritma Dijkstra yang dijelaskan dalam bentuk notasi *pseudo code* sebagai berikut (Munir, 2005):

Procedur Dijkstra (input m: matriks, a : simpul awal)

(Mencari lintasan terpendek dari simpul awal a ke semua simpul lainnya

Masukan: matriks ketetanggaan (m) dari graf berbobot G dan simpul awal a

Keluaran: lintasan terpendek dari a ke semua simpul lainnya

}

Deklarasi

S1, S2, ..., Sn : integer{tabel integer}

d1, d2, ..., dn : integer{tabel integer}

i, j, k : integer

Algoritma

{ Langkah 0 (inisialisasi: }

for i □ 1 to n do

Si□ 0

di□ mai

endfor

{ Langkah 1: }

Sa □ 1 (karena simpul a adalah simpul asal lintasan terpendek, jadi

simpul a sudah pasti terpilih dalam lintasan terpendek)

da □ ∞ (tidak ada lintasan terpendek dari simpul a ke a)

{ Langkah 2, 3, ..., n – 1:}

for k □ 2 to n – 1 do

j □ simpul dengan Sj = 0 dan dj minimal

Sj □ 1 { simpul j sudah terpilih kedalam lintasan terpendek }

{ perbarui tabel d }

for semua simpul i dengan Si = 0 do

if dj + mji < di then

di □ dj + mji

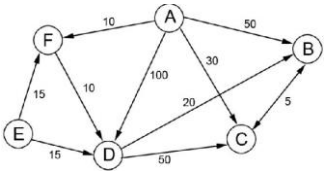
endif

endfor

endfor

Cara kerja algoritma Dijkstra memakai stategi *greedy*. Dimana strategi *greedy* pada algoritma Dijkstra menyatakan bahwa pada setiap langkah, ambil sisi yang berbobot minimum yang menghubungkan sebuah simpul yang sudah terpilih dengan sebuah simpul lain yang belum terpilih. Lintasan dari simpul asal ke simpul yang baru haruslah merupakan lintasan yang terpendek diantara semua lintasannya ke simpul-simpul yang belum dipilih (Dewi, 2010).

Lihat sebuah graf berarah pada Gambar 2.1, diberikan graf berbobot G=(V,E) dan akan ditelusuri graf dari titik A ke semua simpul lainnya yang dapat diakses. Dimana Titik A akan dipertimbangkan sebagai simpul asal. Sebelumnya, perhatikan bahwa graf yang akan digunakan adalah graf berbobot. Graf diberi bobot karena setiap sisinya harus bernilai non-negatif numerik. Pada masing-masing simpul mempunyai anak panah yang menunjukkan arah perjalanan yang memungkinkan simpul untuk dilalui. Adapun bobot untuk mencapai simpul B (50), C (30), D (100), dan F (10). Dimana bobot pada setiap sisi dapat menyatakan jarak, ongkos, waktu, dan sebagainya (Rutter, 2009).

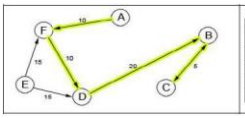


Gambar 1.1 Graf Berarah (Rutter, 2009)

Dengan menerapkan algoritma Dijkstra diperoleh lintasan terpendek dengan jarak terpendek. Adapun hasil dari penelusuran graf berbobot dari simpul IV - 37

asal ke simpul akhir adalah AFDBC dengan Etersisadi tak terhingga, karena ujung

dari anak panahnya hanya berasaldari simpulE dan jauh dari salah satu arahsehingga mustahil untukmencapai E. Sebagaimana terlihat pada gambar dibawah ini.



From

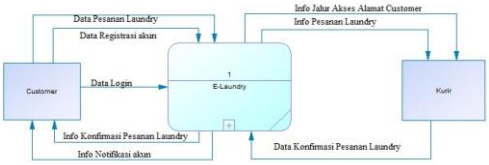
A to	B	C	D	E	F
1. A	50/A	30/A	100/A	∞	10/A
2. F	50/A	30/A	10/F	∞	10/A
3. D	20/D	50/D	10/F	∞	10/A
4. B	20/D	50/D	10/F	∞	10/A
5. C	5/C	50/D	10/F	∞	10/A

Gambar 1.2 Hasil Penerapan Algoritma Dijkstra (Rutter, 2009)

B. PERANCANGAN SISTEM

1. Konteks Diagram

Konteks diagram merupakan gambaran umum dari sebuah aplikasi yang mengambarkan keseluruhan *input*, proses, *output* yang terdapat dalam sebuah aplikasi.

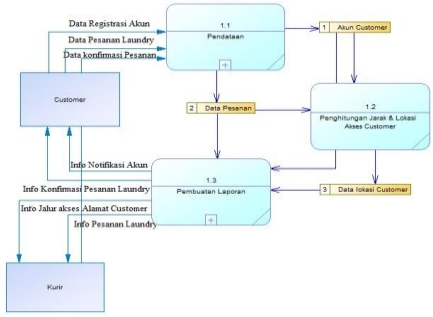


Gambar 2.1 Context Diagram

2. *Data Flow Diagram* (DFD)

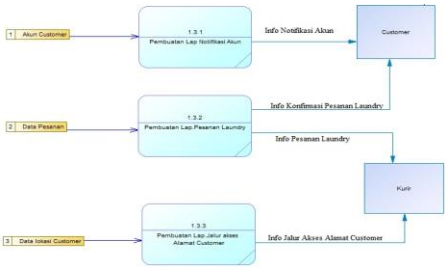
DFD adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang dimana penggunaannya sangat membantu dalam memahami sistem keseluruhan secara logika, terstruktur dan jelas.

2.1. DFD Level 0



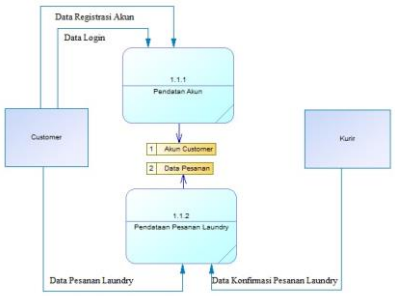
Gambar 2.2 DFD Level 0

2.2. DFD level 1 pembuatan laporan



Gambar 3.6 DFD Level 1 Pembuatan Laporan

2.3. DFD level 1 pendaftaran



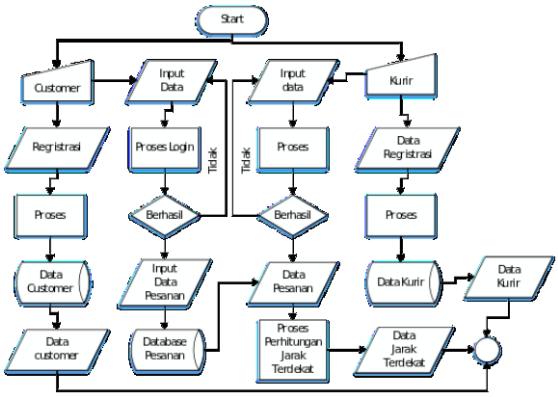
Gambar 3.7 DFD Level 1 Pendataan

3. FLOWCHART

Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.

Flowchart merupakan cara penyajian dari suatu algoritma.

Perancangan perangkat lunak sistem rekomendasi rute terpendek aplikasi *E-laundry* dapat di lihat pada gambar flowchart sebagai berikut:



Gambar 3.3 Flowchat User

4. CMD (Conceptual Data Modeling)

Gambar

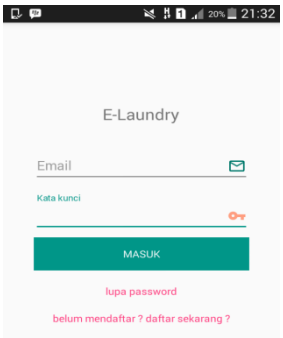
III. HASIL DAN KESIMPULAN

3.1 Implementasi Program

Tahap selanjutnya setelah perancangan adalah tahap implementasi program. Pada tahap implementasi ini, aplikasi dibuat menggunakan bahasa PHP, Java dan basis data MySQL dan SQLite.

3.1.1 Tampilan *Login User*

Pada form ini pengguna memasukkan data email dan password yang sudah didaftarkan untuk masuk ke dalam sistem.



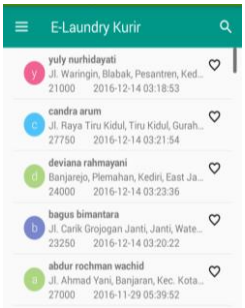
Gambar 3.1 Login

3.1.2 Tampilan Menu Pemesanan

Gambar 3.2 Menu Pemesanan

3.1.3 Tampilan List Konsumen

Pada menu ini akan ditampilkan data konsumen yang memesan jasa *e-laundry*.



Gambar 3.3 Tampilan List Konsumen

3.1.4 Tampilan Rute terpendek

Pada menu ini fitur yang ditampilkan adalah rute terpendek yang akan dilewati oleh kurir dan jalur manakah yang akan ditempuh.

Gambar 3.4 Tampilan Rute Terpendek

3.2 Simpulan

1. Sistem *E-laundry* dapat menampilkan fitur rute terpendek, jarak tempuh dan waktu tempuh.
2. Metode Djikstra dapat diterapkan dengan baik sehingga dapat menampilkan fitur rute terpendek.
3. *OpenStreetMaps* yang digunakan bisa memfasilitasi fitur untuk memperoleh informasi jalan satu arah dan jalan yang tidak bisa dilewati.

3.3 Saran

Berikut ini beberapa saran yang penulis sampaikan untuk pengembangan sistem *E-laundry* selanjutnya, adalah sebagai berikut:

1. Untuk pengembangan skripsi selanjutnya dapat membahas *server*, sehingga admin bisa memantau kinerja *user* melalui sistem.
2. Sistem *E-laundry* untuk selanjutnya dapat digunakan pada multi *platform*.
3. Untuk pengembangan selanjutnya waktu tempuh diperoleh dengan mempertimbangan kondisi jalan dan kemacetan.

4. Sistem *E-laundry* menggunakan algoritma yang efisien untuk permasalahan *Traveling Salesman Problem* (TSP)untuk memperoleh hasil yang optimal.

IV. DAFTAR PUSTAKA

Ahuja, R.K., T.L. Magnanti , J.B. Orlin. 1993. *Network Flow: Theory, Algorithms and Applications.* Prentice Hall, New Jersey.

Anonimous. 2013. *Antar Jemput Laundry Kiloan*. "*Dari Jasa Antar Jemput Secuter*", (Online), (http://antarjemputsecuter.wordpress.com, diakses 19 April 2014)

Android Studio. 2010. "*Mengenal Android Studio*", (Online), tersedia: https://developer.android.com/studio/intro/index.html?hl=id, diunduh 19 April 2016)

Dewi, L.J.E., "*Pencarian Rute Terpendek Tempat Wisata di Bali dengan menggunakan Algoritma Dijkstra*", SNATI, 2010

Fahronzi, Luthfi, 2013. "*Aplikasi Location Basedservice (Lbs) Untuk Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra Dengan Studi Kasus : Pt. Coca Cola Amatil Indonesia Sales Office Pekanbaru''.* Skripsi. Dipublikasikan. Pekanbaru: Fakultas Teknik, Universitas Islam Negeri Sultan Syarif Kasim Riau

Faizah, Ifatul, 2010. "*Rancang Bangun Perangkat Lunak Penentuan Rute Perjalanan Wisata Di Malang Menggunakan Algoritma Dijkstra".* Skripsi. Dipublikasikan. Malang : FT UIN Malang

Fitria, Apri Triansyah. *Oktober 2013,"* Implementasi Algoritma Dijkstra Dalam Aplikasi Untuk Menentukan Lintasan Terpendek Jalan Darat Antar Kota Di Sumatera Bagian Selatan**".** *Jurnal Sistem Informasi (JSI), VOL. 5, NO. 2, Oktober 2013.*

Indonesia, Wikipedia. (2015). "*Kediri, Kediri-Wikipedia Indonesia, Ensiklopedia bebas berbahasa Indonesia*.",(Online), tersedia: https://id.m.wikipedia.org/wiki/kediri,_Kediri, diakses 06 Maret 2016 pukul 11:07 WIB

Indonesia, Wikipedia. (2015). " *SQLite Wikipedia Indonesia, Ensiklopedia bebas berbahasa Indonesia* ", (Online), tersedia: https://id.wikipedia.org/wiki/SQLite, diakses 06 Maret 2016 pukul 11:07 WIB

Indonesia, Wikipedia. (2015). "*OpenStreetMaps Wikipedia Indonesia, Ensiklopedia bebas berbahasa Indonesia*." ,(Online), tersedia: https://id.wikipedia.org/wiki/*OpenStreetMap*, diakses 06 Maret 2016 pukul 11:07 WIB

Munir, Rinaldi,2005. *Matematika Diskrit*. Bandung: Informatika Bandung.

Munir, R. 2008. *Matematika Diskrit*. Penerbit Informatika. Bandung

Rutter, S.J., "*Dijkstra's Algorithm Final Project*", EDUC, 2009.

Sarwoko, E. A. 2003. Perancangan Arsitektur Pemaralelan untuk mencari Shortest Path dengan Algoritma *Djikstra*. *Jurnal Matematika dan Komputer*. **6:** 137-143.