

**SISTEM APLIKASI PENENTUAN RUTE TERPENDEK PADA
JARINGAN MULTI MODA TRANSPORTASI UMUM
MENGUNAKAN ALGORITMA DIJKSTRA**

**Tesis
untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-2
Program Studi Magister Sistem Informasi**



Sofyan Arifianto

J4F008027

**PROGRAM PASCASARJANA
UNIVERSITAS DIPONEGORO
SEMARANG
2012**

SISTEM APLIKASI PENENTUAN RUTE TERPENDEK PADA JARINGAN MULTI MODA TRANSPORTASI UMUM MENGUNAKAN ALGORITMA DIJKSTRA

ABSTRAK

Biaya transportasi merupakan komponen yang sangat signifikan, rata – rata 15 %– 20%, bahkan di Jakarta dapat mencapai 25%-30% dari pengeluaran biaya rumah tangga. Di Kota Jakarta sudah disediakan beberapa jenis transportasi umum dan penumpang bebas memilih jenis transportasi mana yang digunakan untuk menuju ke tempat yang disediakan. Jenis transportasi di kota besar yang rutanya bisa direpresentasikan sebagai *graph* antara lain, bus Transjakarta dan KRL *commuter line*.

Penentuan rute terpendek pada jaringan transportasi umum dapat ditentukan dengan algoritma *Dijkstra*. Jalur transportasi di anggap sebagai *graph* berarah dan berbobot, titik merepresentasikan sebuah tempat pemberhentian kendaraan, dan sisi sebagai jalur kendaraan. *User* menginput tempat awal dan tempat tujuan, rute terpendek dari titik awal ke semua titik yang ada ditentukan dengan algoritma *Dijkstra*, sehingga *output* berupa rute terpendek dari tempat awal menuju tempat tujuan dapat ditemukan.

Perhitungan rute terpendek yang dipilih dari beberapa alternatif rute secara manual menghasilkan nilai yang sama dengan menggunakan Aplikasi Penentuan Rute Terpendek Menggunakan Algoritma Dijkstra sehingga aplikasi tersebut layak untuk digunakan.

Kata Kunci : Algoritma Dijkstra, Rute Terpendek, Transportasi Umum

APPLICATION SYSTEM IN DETERMINING THE SHORTEST ROUTE MULTI MODE NETWORK ON PUBLIC TRANSPORTATION USING DIJKSTRA ALGORITHM

ABSTRACT

Transportation costs are a significant component, average 15%-20%, even in Jakarta it can reach 25% -30% of household expenses. In Jakarta city, it has been provided several types of public transportation and the passengers are free to choose which kind of transportation is used to reach into the provided place. The transportation types in big cities that the route can be presented as graph are transjakarta bus and KRL commuter line.

Shortest route determination on public transportation can be determined by Dijkstra algorithm. Transportation lane is considered as a directed and weighted graph. The points represent a vehicle stop. User input the starting and destination place, the shortest route from the starting point to all other points are determined by Dijkstra algorithm, so that the output of the shortest route from the beginning to the destination can be found.

The shortest route calculation that is chosen from several alternatives route manually result the same value by using the application of shortest route determination using Dijkstra algorithm so that the application is feasible to use.

Key words : Dijkstra algorithm, the shortest route, public transportation

BAB I

PENDAHULUAN

1.1. Latar Belakang

Permasalahan transportasi pada kota besar yang mempunyai jaringan transportasi yang rumit umumnya sama. orang sering mengetahui rute perjalanan ke tempat yang biasa di kunjunginya, akan tetapi jika tempat tujuan tersebut belum pernah dikunjungi rata - rata mereka sering kesulitan untuk menentukan rute untuk mencapai tempat tersebut (Chiu et al, 2005). selama ini orang akan bertanya kepada orang lain yang mengetahui betul jaringan transportasi di kota tersebut.

Biaya transportasi merupakan komponen yang sangat signifikan, rata – rata 15 %– 20%, bahkan di Jakarta dapat mencapai 25%-30% dari pengeluaran biaya rumah tangga. Dengan adanya sistem penentuan rute terpendek diharapkan dapat mengefektifkan biaya perjalanan.

Di Kota Jakarta sudah disediakan beberapa jenis transportasi umum dan penumpang bebas memilih jenis transportasi mana yang digunakan untuk menuju ke tempat yang disediakan. Jenis transportasi di kota besar yang rutenya bisa direpresentasikan sebagai *graph* antara lain, bus Transjakarta dan KRL *commuter line*. Dua moda transportasi tersebut memiliki ciri khas di banding alat transportasi lainnya, yaitu memiliki tempat pemberhentian yang sudah ditentukan. Sehingga rute dapat digambarkan menyerupai sebuah *graph*. Selain itu penumpang sering mengalami kesulitan menentukan rute terpendek, dan sulit menentukan lokasi tempat untuk berganti dengan moda transportasi lain.

Permasalahan di atas dapat diselesaikan dengan membuat suatu sistem yang dapat menjawab semua pertanyaan penumpang yang dapat menentukan jalur transportasi umum yang dikehendaki. Sistem ini sekaligus dapat menentukan rute terpendek sehingga penumpang tidak perlu menempuh jarak yang jauh untuk mencapai tujuannya.

Penentuan jalur terpendek dipilih menggunakan algoritma *Dijkstra*. Algoritma ini dapat menentukan jalur terpendek dari *graph* berbobot yang

bobotnya bernilai lebih besar dari nol (positif), dari titik awal ke semua titik yang dikehendaki, sehingga nantinya akan ditemukan jalur terpendek dari titik awal dan titik tujuan yang diinputkan. Untuk mempermudah penggunaan sistem ini, maka calon penumpang hanya menginputkan tentang Informasi titik awal, dan titik tujuan. Setelah itu akan ada *output* tentang informasi kepada penumpang tersebut untuk naik kendaraan umum, dan kalau perlu transit atau pindah jalur juga akan diinformasikan tempat transitnya, dan tempat turun penumpang sesuai tujuannya.

Hal tersebut yang mendasari pentingnya penelitian tentang penentuan rute terpendek menggunakan Algoritma *Dijkstra*, untuk membantu masyarakat dalam penggunaan transportasi umum.

1.2. Perumusan Masalah

Berdasarkan latar belakang sebelumnya peneliti merumuskan permasalahan penelitian yaitu

1. Penentuan rute terpendek jalur transportasi umum dari input tempat asal sampai tujuan.
2. Menghitung jarak tempuh dari tempat asal sampai tujuan.
3. Membuat *prototype* untuk menentukan jalur terpendek bila menggunakan transportasi umum.

1.3. Batasan Masalah

Permasalahan dalam penelitian ini dibatasi sebagai berikut:

1. Untuk memudahkan penelitian akan dipilih suatu jaringan transportasi umum yang sudah ada yaitu jaringan kereta api KRL commuter line dan bus Transjakarta.
2. Dalam penentuan rute terpendek menggunakan *algoritma Dijkstra* dengan titik awal adalah tempat asal penumpang, dan titik akhir adalah tujuan akhir penumpang.

1.4 Keaslian Penelitian

Sistem informasi bagi penumpang telah dilakukan di Kota Hongkong dengan menggunakan jaringan transportasi umum yang cukup rumit. Sistem tersebut dapat menggabungkan informasi dari berbagai jenis angkutan umum yang ada di kota tersebut, antara lain bus, mini bus, kereta api dan jenis transportasi lainnya. Dalam sistem tersebut Algoritma *Dijkstra* digunakan sebagai dasar untuk menemukan lintasan terpendek, selain itu mereka juga memodifikasi Algoritma tersebut agar sesuai dengan kebutuhan yang ada (Chiu et al, 2005).

Penelitian di Singapura yang juga meneliti tentang pemecahan permasalahan transportasi di Singapura, mengemukakan tentang perhitungan jam keberangkatan dan kedatangan setiap model transportasi diperhitungkan secara detil, sehingga penumpang mengetahui berapa lama waktu perjalanan yang harus ditempuh. Sistem ini dibangun setelah sistem di Singapura tertata rapi dan jadwal kedatangan serta keberangkatan telah ditetapkan sesuai dengan jadwal (Meng et al, 1999).

Di Indonesia juga sudah dilakukan penelitian tentang permasalahan transportasi di kota Jakarta yaitu menentukan rute tercepat pada jaringan Transjakarta menggunakan algoritma *Dijkstra* (Florens dkk, 2009). Pada penelitian ini lebih dititikberatkan pada waktu tempuh perjalanan pada jaringan Transjakarta.

Penelitian yang peneliti lakukan yaitu menentukan informasi bagi penumpang yang menempuh perjalanan dengan menggunakan transportasi umum. Penumpang diinformasikan rute mana yang paling pendek, dan dimana tempat pergantian jenis transportasi umum bila diperlukan. Dengan algoritma *Dijkstra* maka pertama kali tempat awal penumpang diibaratkan sebagai titik awal atau posisi awal. Kemudian dari titik awal akan dicari lintasan ke semua titik. Setelah itu titik awal akan berubah menjadi titik berikutnya. Sehingga iterasi tersebut akan berakhir bila titik awal sama dengan tujuan akhir penumpang. Dan hasil akhirnya adalah rute terpendek dari titik awal sampai dengan titik tujuan akhir. Adapun perbedaan lain yaitu mengenai penelitian yang sudah dilakukan pada beberapa negara lain dapat dilihat pada tabel 1.1.

Tabel 1.1 : Penelitian yang sudah dilakukan

	Penelitian Foo Hee Meng dkk, 2000	Penelitian Dickson K.W. Chiu dkk, 2005	Penelitian Rufina Florens dkk, 2009
Judul Penelitian	A Multi-Criteria, Multi-Modal Passenger Route Advisory Sistem	A Multi-Modal Agent Based Mobile Route Advisory Sistemfor Public Transport Network	Analisis Dan Perancangan Sistem Pencarian jalur tercepat Untuk Transportasi Bus Transjakarta Menggunakan Algoritma Dijkstra
Tempat studi kasus	Singapura	Hongkong	Jakarta
Jenis transportasi umum	Transportasi umum berbasis rel dan jalan raya	Dari berbagai jenis transportasi baik yang berbasis jalan raya, laut, maupun berbasis rel (Kereta Api, Trem, ferry, dsb)	Jaringan Transportasi Transjakarta
Algoritma yang di pakai	Algoritma Dijkstra	Algoritma Dijkstra	Algoritma Dijkstra

1.5 Tujuan Penelitian

Penelitian ini bertujuan untuk membuat program aplikasi untuk penentuan rute terpendek pada jalur transportasi umum menggunakan algoritma *Dijkstra*.

1.6 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat yaitu:

1. Dapat menentukan rute terpendek bagi penumpang angkutan umum bus Transjakarta dan KRL *commuter line*.
2. Efektifitas waktu perjalanan.
3. Meminimalkan biaya perjalanan.

BAB II TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Permasalahan yang terjadi di kota besar yang mempunyai berbagai macam jenis transportasi umum dari yang berbasis jalan raya maupun berbasis rel yang sangat menyulitkan penumpang yang akan menempuh suatu perjalanan. Tidak semua orang dapat mengingat rute yang harus dilewati dan jenis alat transportasi umum apa yang harus di gunakan (Chiu et al, 2005). Oleh karena itu dilakukan penelitian yang menciptakan suatu program aplikasi yang dapat memberi informasi penumpang tentang keluhan-keluhan tersebut.

Penelitian di Singapura yang juga meneliti tentang pemecahan permasalahan transportasi di Singapura, mengemukakan tentang perhitungan jam keberangkatan dan kedatangan setiap model transportasi diperhitungkan secara detail, sehingga penumpang mengetahui berapa lama waktu perjalanan yang harus ditempuh. Sistem ini dibangun setelah sistem di Singapura tertata rapi dan jadwal kedatangan serta keberangkatan telah ditetapkan sesuai dengan jadwal (Meng et al, 1999). Sedangkan di Indonesia masalah jadwal tidak di perhitungkan karena tidak ada jadwal yang pasti. Sehingga dalam penulisan ini memperhitungkan kondisi normal, yakni dianggap jalan lancar dan tanpa hambatan.

Algoritma *Dijkstra* untuk menentukan rute terpendek. Algoritma *Dijkstra* digunakan pada *graph* berarah dan berbobot. Jika bobot *graph* > 0 maka digunakan *Dijkstra* dengan level satu, dan bila bobot *graph* ada yang negatif akan digunakan level dua (Cantone dan Faro, 2004). Dalam penelitian ini akan dipakai algoritma *Dijkstra* yang memakai bobot >0 , karena bobot *graph* merepresentasikan jarak antar titik sehingga bobotnya selalu positif.

Penelitian yang akan dikembangkan berjudul “Sistem Aplikasi Penentuan Rute Terpendek Pada Jaringan Multi Moda Transportasi Umum Menggunakan Algoritma *Dijkstra* ”. Pada penelitian ini, Algoritma *Dijkstra* digunakan untuk menentukan lintasan terpendek dari jalur transportasi umum. Jalur transportasi umum akan diibaratkan sebagai *graph* berbobot, titik merepresentasikan sebuah

tempat pemberhentian, sisi sebagai rel kereta api atau jalur bus Transjakarta, dan bobot sebagai jarak antar tempat pemberhentian. *Input* adalah tempat awal penumpang dan tujuan akhir penumpang. Dengan algoritma *Dijkstra* maka pertama kali tempat awal penumpang diibaratkan sebagai titik awal atau posisi awal. Kemudian dari titik awal akan dicari lintasan ke semua titik. Setelah itu titik berikutnya akan menjadi titik acuan. Sehingga iterasi tersebut akan berakhir bila titik acuan sama dengan tujuan akhir penumpang. Dan hasil akhirnya adalah rute terpendek dari titik awal sampai dengan titik tujuan akhir.

Tempat studi kasus yang akan dilakukan adalah jaringan kereta api KRL *commuter line* dan bus Transjakarta. Sehingga dalam aplikasinya titik merepresentasikan sebuah stasiun pemberhentian atau halte bus Transjakarta dan sisi sebagai rel kereta api atau jalur bus Transjakarta.

2.2 Landasan Teori

2.2.1 Algoritma *Dijkstra*

Algoritma ini diberi nama sesuai nama penemunya, Edsger Wybe Dijkstra. Algoritma *Dijkstra* mencari lintasan terpendek dalam sejumlah langkah. Algoritma ini menggunakan prinsip *Greedy* yang menyatakan bahwa pada setiap langkah kita memilih sisi yang berbobot minimum dan memasukkannya ke dalam himpunan solusi

Input algoritma ini adalah sebuah *graph* berarah yang berbobot (*weighted directed graph*) G dan sebuah sumber *vertex* s dalam G dan V adalah himpunan semua *vertices* dalam *graph* G (Rosen, 1999):

Properti algoritma *Dijkstra*:

1. Matriks ketetanggaan $M[m_{ij}]$

$$m_{ij} = \text{bobot sisi } (i, j)$$

$$m_{ii} = 0$$

$$m_{ij} = \infty, \text{ jika tidak ada sisi dari simpul } i \text{ ke simpul } j$$

2. Larik $S = [s_i]$ yang dalam hal ini,

$$s_i = 1, \text{ jika simpul } i \text{ termasuk ke dalam lintasan terpendek}$$

$$s_i = 0, \text{ jika simpul } i \text{ tidak termasuk ke dalam lintasan terpendek}$$

3. Larik/tabel $D = [d_i]$ yang dalam hal ini,

d_i = panjang lintasan dari simpul awal s ke simpul i

Algoritma Lintasan Terpendek *Dijkstra* (Mencari lintasan terpendek dari simpul a ke semua simpul lain)

Langkah 0 (inisialisasi):

- inisialisasi $s_i = 0$ dan $d_i = m_{ai}$ untuk $i = 1, 2, \dots, n$

Langkah 1:

- isi s_a dengan 1 (karena simpul a adalah simpul asal lintasan terpendek, jadi sudah pasti terpilih)

- isi d_a dengan ∞ (tidak ada lintasan terpendek dari simpul a ke a)

Langkah 2, 3, ..., n :

- cari j sedemikian sehingga $s_j = 0$ dan $d_j = \min\{d_1, d_2, \dots, d_n\}$

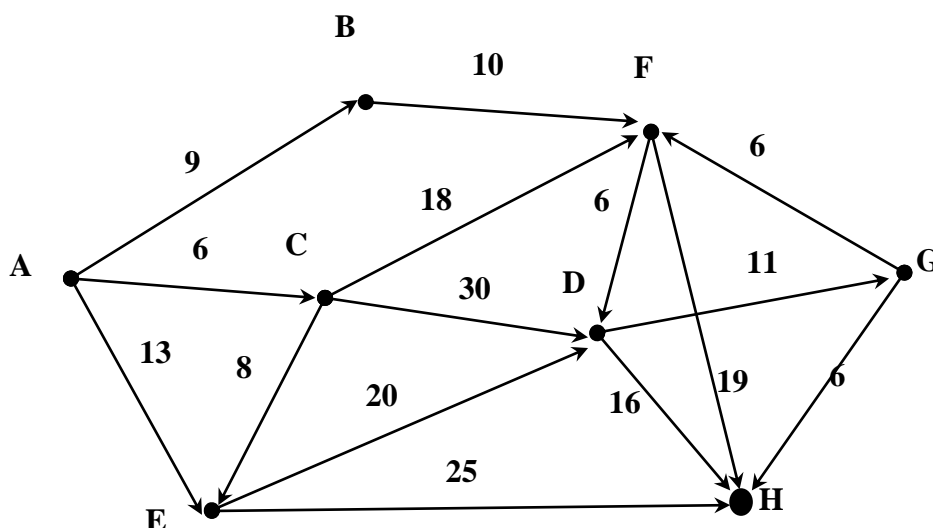
- isi s_j dengan 1

perbarui d_i , untuk $i = 1, 2, 3, \dots, n$ dengan:

d_i (baru) = $\min\{d_i$ (lama), $d_j + m_{ji}\}$.

Contoh

Misal diberikan *graph* berbobot seperti gambar 2.1 di bawah



Gambar 2.1 : Contoh *graph* berbobot

Misalnya diberikan *graph* berbobot dan berarah seperti gambar di atas. Akan dicari lintasan terpendek dari simpul A ke semua simpul lain. Maka untuk menyelesaikan permasalahan tersebut dapat menggunakan tabel seperti pada tabel 2.1.

Jadi, lintasan terpendek dari:

A ke C adalah A, C dengan panjang = 6

A ke B adalah A, B dengan jarak = 9

A ke E adalah A, B, E dengan jarak = 13

A ke F adalah A, B, F dengan jarak = 19

A ke D adalah A, B, F, D dengan jarak = 25

A ke G adalah A, B, F, D, G dengan jarak = 36

A ke H adalah A, B, E, H dengan jarak = 38.

Tabel 2.1 : Penyelesaian menggunakan algoritma *Dijkstra*

Iterasi ke	Simpul yang dipilih	Lintasan	S								D							
			A	B	C	D	E	F	G	H	A	B	C	D	E	F	G	H
inisialisasi	-	-	0	0	0	0	0	0	0	0	∞	∞	∞	∞	∞	∞	∞	∞
1	A	A	1	0	0	0	0	0	0	0	∞	9	6	∞	13	∞	∞	∞
2	C	A,C	1	0	1	0	0	0	0	0	∞	9	6	36	13	∞	∞	∞
3	B	A,B	1	1	1	0	0	0	0	0	∞	9	6	36	13	19	∞	∞
4	E	A,B,E	1	1	1	0	1	0	0	0	∞	9	6	33	13	19	∞	38
5	F	A,B,F	1	1	1	0	1	1	0	0	∞	9	6	25	13	19	∞	38
6	D	A,B,F,D	1	1	1	1	1	1	0	0	∞	9	6	25	13	19	36	38
7	G	A,B,F,D,G	1	1	1	1	1	1	1	0	∞	9	6	25	13	19	36	38
8	H	A,B,E,H	1	1	1	1	1	1	1	1	∞	9	6	25	13	19	36	38

2.2.2 Sistem Transportasi Umum dan Permasalahannya

Transportasi adalah usaha memindahkan, menggerakkan, mengangkut, atau mengalihkan suatu obyek dari suatu tempat ke tempat lain agar lebih bermanfaat atau berguna untuk tujuan tertentu. Pengertian sistem transportasi secara operasional yaitu masalah masalah yang berhubungan dengan perpindahan barang dan atau orang (dengan memakai energi) yang bermanfaat bagi manusia (Kadir, 2006). Sedangkan pengertian transportasi kota adalah transportasi yang dilakukan di dalam suatu kota.

Transportasi umum mempunyai permasalahan yang sama dalam hal ini mengenai kebingungan penumpang menentukan rutenya. Dalam sebuah kota besar dengan jaringan transportasi yang rumit, penumpang sering tidak tahu bagaimana cara untuk mencapai tempat tujuan mereka kecuali tempat tersebut sering dikunjungi. Selain itu, setiap penumpang mungkin ingin merencanakan rute tercepat atau yang paling ekonomis untuk tujuan mereka (Chiu et al, 2005)

Penelitian ini mengambil beberapa permasalahan penumpang yang akan menempuh perjalanan dari satu tempat ke tempat lain di kota Jakarta. Misal penumpang akan bepergian dari stasiun Bekasi ke Halte Blok M maka penumpang tersebut akan mempunyai berbagai macam alternatif transportasi umum yang akan di pilihnya.

Penumpang dapat memilih jenis transportasi apa yang dipilihnya dan jalur mana yang akan dilewati. Bila penumpang tersebut tidak mengetahui jaringan kereta api KRL *commuter line* dan rute dari bus Transjakarta tentunya sulit menentukan jurusan kereta api dan bus Transjakarta yang akan dinaikinya untuk mencapai tempat tujuan. Tentunya sangat sulit menghafal kereta dan bus Transjakarta yang melayani tersebut. Hal ini sering terjadi pada penumpang yang dalam menempuh perjalanan harus berganti dengan jenis transportasi lainnya.

BAB III

METODE PENELITIAN

3.1 Bahan Penelitian

Obyek yang dijadikan bahan penelitian adalah jalur transportasi umum berbasis rel yang ada di jabodetabek dan jalur Transjakarta di DKI Jakarta. Jalur kereta api KRL *commuter line* dan jalur bus Transjakarta mempunyai beberapa titik persinggungan sehingga memudahkan penumpang untuk berganti jenis transportasi dari KRL ke bus Transjakarta dan juga sebaliknya.

3.2 Alat Penelitian

Alat penelitian yang digunakan dalam proses penelitian ini menggunakan *notebook* dengan *procesor intel I5*, RAM 4 Gbyte, Hard disk 750 Gbyte, VGA 2 Gbyte, dengan menggunakan *sistem operasi windows 7 profesional*. Kemudian perangkat lunak yang digunakan antara lain:

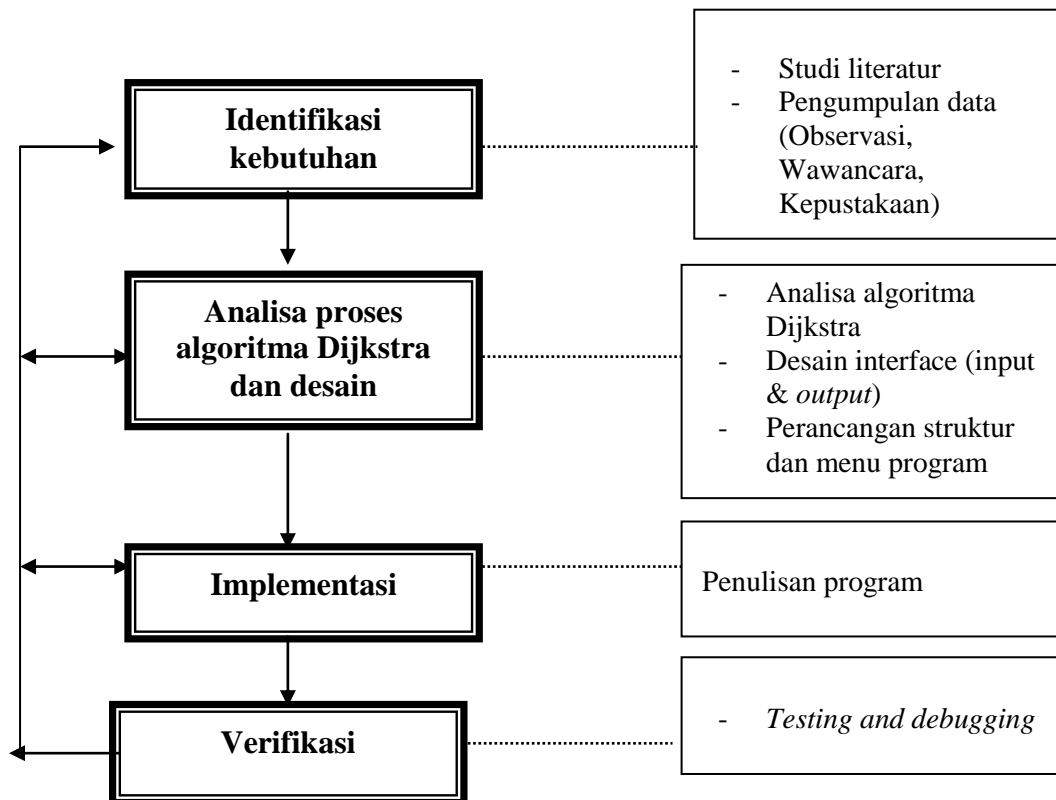
- 1) Paket Database Management System (DBMS) yang digunakan adalah MySQL.
- 2) Penulisan kode program menggunakan bahasa Pemrograman C#, dengan software penyusunnya visual studio 2010.

3.3 Jalan Penelitian

Berikut ini adalah penjelasan dari masing-masing tahapan dalam penelitian. Dalam pengerjaan penelitian, mengikuti tahapan sebagai berikut:

1. Identifikasi kebutuhan.
2. Analisa proses *algoritma Dijkstra* dan desain.
3. Implementasi.
4. Verifikasi.

Gambar 3.1 merupakan tahapan dari program aplikasi yang dibangun.



Gambar 3.1 : Tahapan penelitian

3.4 Identifikasi kebutuhan

Tahap ini mengidentifikasi beberapa literatur yang diperlukan untuk menggali teori yang telah berkembang dalam bidang ilmu yang berkepentingan, mencari metode-metode, teknik penelitian, baik dalam mengumpulkan data atau menganalisa data untuk mengetahui sampai kemana ilmu yang berhubungan dengan penelitian telah berkembang dan sebagai dasar untuk mengaplikasikannya kedalam penelitian ini.

Peneliti menelusuri literatur dari sumber referensi jurnal nasional/internasional, buku, dan sumber internet. Bagian ini terus dilakukan beriringan dengan bagian lain sampai bagian akhir penelitian. Hal ini dilakukan agar bila dalam tahap selanjutnya peneliti membaca atau mempelajari sumber referensi lain yang mendukung penelitian, sumber referensi tersebut dapat di masukkan sebagai referensi untuk memperkuat bahwa penelitian ini layak di laksanakan. Hal lain yang dilakukan adalah mengumpulkan data yang diperlukan. Dalam hal ini peneliti mengumpulkan data – data tentang jaringan kereta api KRL *commuter line* dan halte bus Transjakarta, diantaranya nama stasiun, nama halte, dan jarak antar stasiun atau atau jarak antar halte bus Transjakarta. Selain itu juga peneliti mengumpulkan data jurusan kereta api kelas KRL *commuter line*, dan bus Transjakarta beserta rute yang dilewatinya. Dalam

mengambil data ada beberapa cara yaitu, wawancara langsung dengan pejabat terkait di PT KAI untuk data jarak antar stasiun, dan PT Transjakarta untuk data jarak antar halte.

3.4.1 Data stasiun yang melayani KRL *commuter line*

Data berupa nama stasiun yang melayani perjalanan KRL *commuter line* bersumber dari peta resmi KRL *commuter line* yang dikeluarkan PT KAI pada 5 Desember 2011 lalu. Dalam peta tersebut terdapat 69 stasiun yang melayani KRL *commuter line*. Yang terdiri dari ;

1. 18 Stasiun yang menghubungkan Jakarta Kota – Bekasi.
2. 3 stasiun yang menghubungkan Kampung Bandan – Tanjung priok.
3. 6 Stasiun yang menghubungkan Kampung Bandan – Jatinegara.
4. 1 stasiun yang menghubungkan Kampung Bandan – Duri.
5. 9 stasiun yang menghubungkan Duri – Tangerang.
6. 13 stasiun yang menghubungkan Tanah abang – Parung Panjang.
7. 3 stasiun yang menghubungkan Tanah abang – Manggarai.
8. 16 stasiun yang menghubungkan Manggarai – Bogor.

Selain nama stasiun data yang lain dari PT KAI adalah daftar jarak antar stasiun, yang nantinya digambarkan sebagai bobot dari *graph* yang dihitung dengan algoritma *Dijkstra*. Data jarak stasiun bersumber dari buku “jarak antar stasiun” yang di keluarkan oleh PT KAI. Untuk lebih jelas dapat dilihat pada gambar peta KRL *commuter line* pada gambar 3.2.



Gambar 3.2: Peta Rute KRL *Commuter line*

3.4.2 Data Halte yang melayani bus Transjakarta

Data tentang halte bus Transjakarta bersumber dari buku “Profil Transjakarta 2012” yang dikeluarkan oleh PT Transjakarta. Saat ini terdapat 11 koridor yang sudah beroperasi, koridor tersebut antara lain ;

1. Koridor 1 melayani rute Blok M – Kota, dengan panjang rute 12,9 Km, terdapat 20 halte.
2. Koridor 2 melayani rute Pulo Gadung – Harmoni, dengan panjang rute 14 Km, terdapat 23 halte.
3. Koridor 3 melayani rute Kalideres – Harmoni, dengan panjang rute 19 Km, terdapat 13 halte.
4. Koridor 4 melayani rute Pulo Gadung – Dukuh Atas, dengan panjang rute 11,85Km, terdapat 17 halte.

5. Koridor 5 melayani rute Ancol – Kampung Melayu, dengan panjang rute 13,5 Km, terdapat 17 halte.
6. Koridor 6 melayani rute Ragunan – Dukuh Atas, dengan panjang rute 13,3 Km, terdapat 18 halte.
7. Koridor 7 melayani rute Kp. Rambutan – Kp Melayu, dengan panjang rute 12,8 Km, terdapat 13 halte.
8. Koridor 8 melayani rute Lebak Bulus - Harmoni, dengan panjang rute 26 Km, terdapat 21 halte.
9. Koridor 9 melayani rute Pinang Ranti – Pluit, dengan panjang rute 28,8 Km terdapat 24 halte.
10. Koridor 10 melayani rute Tanjung Priok – Cililitan, dengan panjang rute 19,4 Km terdapat 18 halte.
11. Koridor 11 melayani rute Pulo Gebang – Kp Melayu, dengan panjang rute 11,76 Km terdapat 14 halte.

3.5 Analisa proses algoritma *Dijkstra* dan desain

Sesuai dengan permasalahan dalam rumusan masalah maka gambar pada peta di asumsikan sebagai *graph* berbobot dan berarah maka diperlukan sebuah program aplikasi yang dapat menyimpan informasi dari *graph* tersebut.

Program aplikasi bekerja setelah admin memasukan data tentang jaringan kereta api KRL *commuter line* dan jalur bus Transjakarta, serta beberapa rute yang dilayani KRL dan bus Transjakarta. Setelah itu program aplikasi dapat bekerja sesuai permintaan pengguna dalam hal ini calon penumpang. *input* dari pengguna diterima oleh program aplikasi yaitu berupa tempat awal dan tempat tujuan penumpang, kemudian menentukan rute terpendek dengan menggunakan algoritma *Dijkstra* dengan mengambil data yang sesuai dari database yang telah diinputkan admin. Setelah ditemukan rute terpendek, informasi tersebut langsung dikirim ke pengguna.

Sesuai dengan Algoritma yang dipakai langkah–langkah yang dilakukan adalah :

1. Tempat awal di asumsikan sebagai titik acuan, dan tempat tujuan di asumsikan titik akhir .
2. Menghitung jarak dari titik awal ke seluruh titik pada *graph*.
3. Kemudian perhitungan berhenti bila semua titik telah dihitung.
4. Sehingga *output* yang diharapkan didapat yaitu jarak terpendek dari titik awal menuju titik akhir.

Setelah analisa kemudian dilanjutkan mengimplementasikan analisa tersebut ke sebuah program aplikasi sehingga di dapat *output* akhir yaitu terbentuknya sebuah program aplikasi untuk menentukan jarak terpendek pada jaringan transportasi umum.

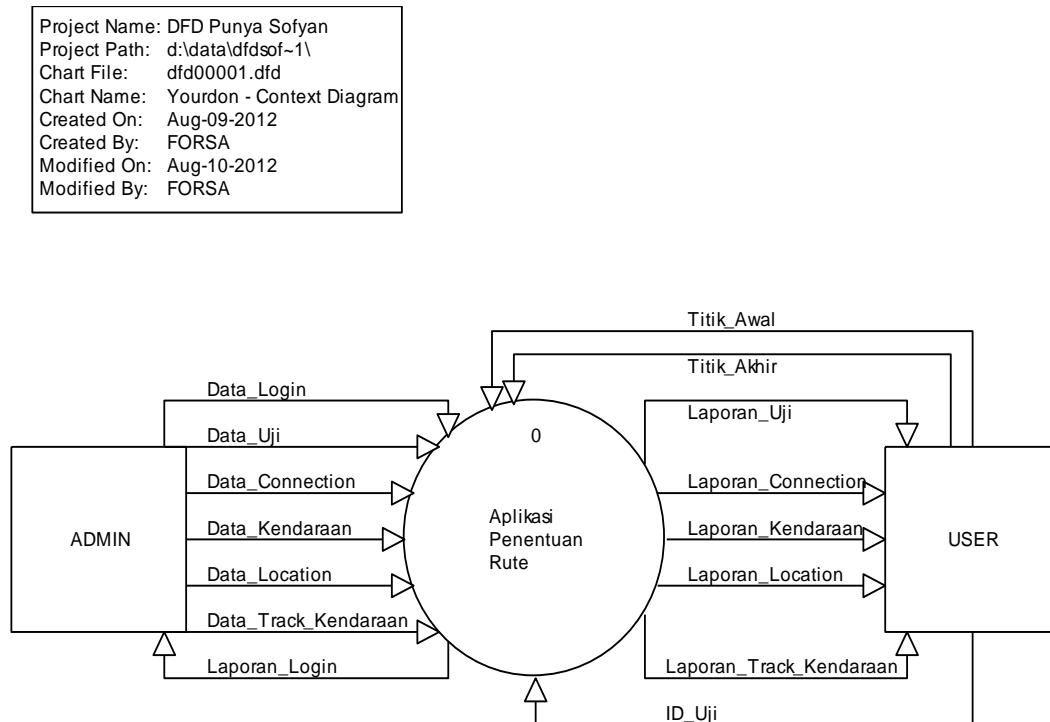
3.5.1 Analisis Sistem

Dibagian analisis sistem ini menjelaskan tentang Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), dan relasi antar tabel.

a. Data Flow Diagram (DFD) Sistem

A. Konteks diagram

Pada bagian ini akan dijelaskan gambaran sistem dalam bentuk data flow diagram. data flow diagram digunakan untuk menggambarkan tentang bagaimana hubungan antara entitas sistem dan aktivitas apa saja yang dapat dilakukan oleh masing-masing entitas tersebut. Untuk lebih jelasnya diagram konteks sistemnya dapat dilihat pada Gambar 3.3.

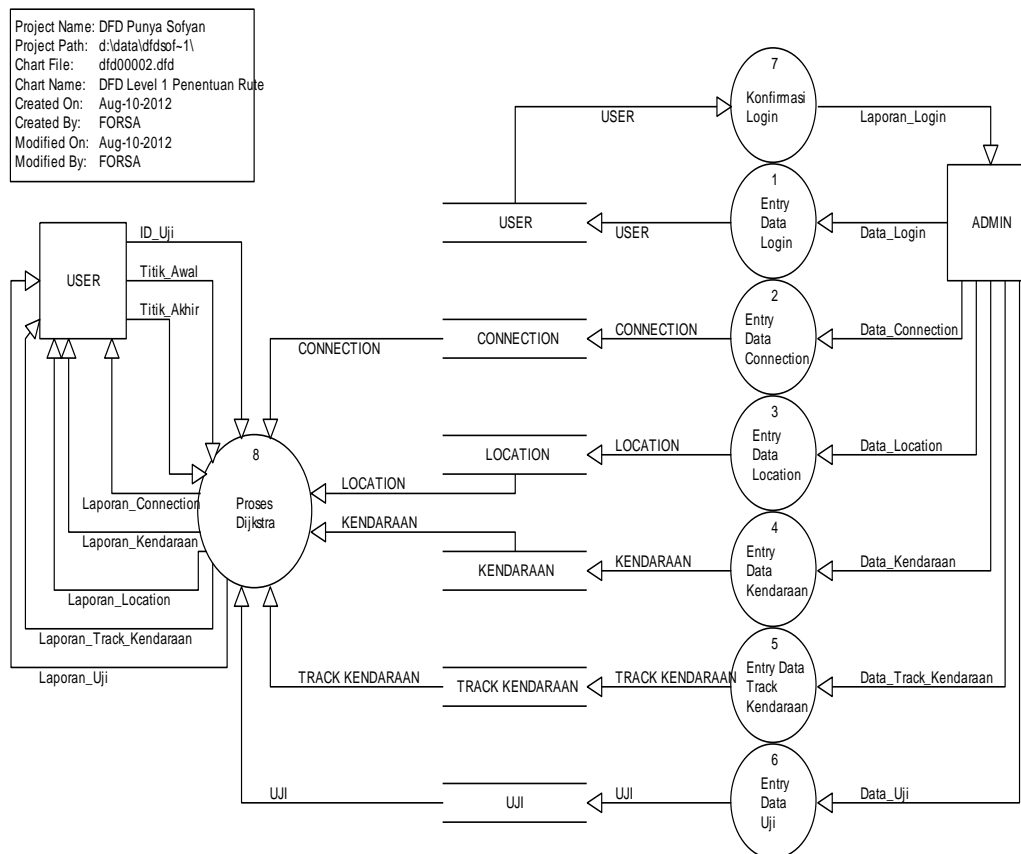


Gambar 3.3. Diagram konteks program aplikasi penentuan rute terpendek

Administrator menginput data pada peta berupa data project, kendaraan, location, connection, track kendaraan. Sekaligus input untuk memeriksa program yaitu lokasi awal dan lokasi tujuan. Setelah data pada peta diinputkan semua, program dapat bekerja sesuai dengan lokasi awal dan lokasi tujuan yang diinputkan user. Sehingga akan dihasilkan *output* berupa rute terpendek.

B. Data Flow Diagram Level 1

Selanjutnya dari diagram konteks diturunkan menjadi lebih rinci dalam bentuk Data Flow Diagram level 1 sebagai berikut:



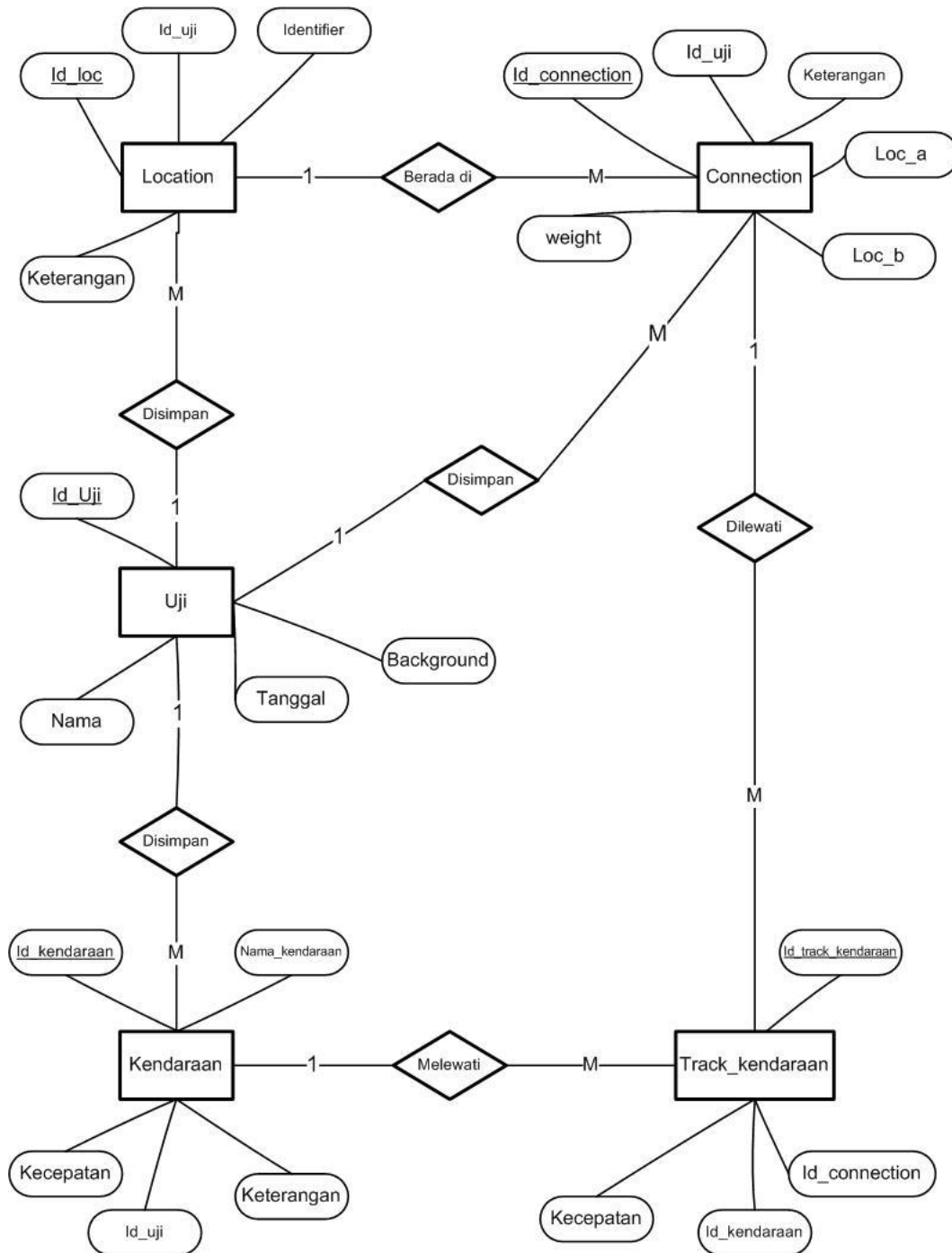
Gambar 3.4 DFD level 1 Program aplikasi penentuan rute terpendek

Proses pengisian data dilakukan oleh admin dan user, admin menginputkan data dari peta, yaitu data kendaraan, location, connection, track kendaraan dan data uji. Sedangkan user menginputkan titik awal dan titik akhir yang nantinya di proses sebagai masukan di proses *Dijkstra*. Semua data akan disimpan terlebih dahulu di masing – masing tabel.

Proses Dijkstra akan mengambil data dari masing –masing tabel untuk dihitung menjadi rute terpendek yang akan diinformasikan sebagai output kepada user..

b. Entity Relationship Diagram (ERD)

ERD digunakan untuk menggambarkan hubungan (*relationship*) antar entitas (*entity*). ERD selengkapnya dalam sistem ini dapat dilihat pada Gambar 3.5



Gambar 3.5 ERD program aplikasi penentuan rute terpendek

3.5.2 Perancangan Sistem

Pada subbab ini akan dijelaskan tentang rancangan tabel, rancangan antar muka sistem, dan rancangan menu.

A. Rancangan Tabel

Rancangan tabel-tabel yang ada dalam sistem ini adalah sebagai berikut

Tabel 3.1 Tabel location

No	Field	Type	Null	Default	Extra
1	<i>Id_loc</i>	int(11)	No	NULL	Auto increment
2	Id_uji	int(11))	No		
3	identifier	varchar(5)	No		
4	keterangan	varchar(150)			

Tabel 3.2 Tabel connection

No	Field	Type	Null	Default	Extra
1	<i>Id_connection</i>	int(11)	No	NULL	Auto increment
2	Id_uji	int(11))	No		
3	loc_a	varchar(4)	No		
4	loc_b	varchar(4)	No		
5	weight	int(11)	No		
6	keterangan	varchar(150)	No		

Tabel 3.3 Tabel kendaraan

No	Field	Type	Null	Default	Extra
1	<i>Id_kendaraan</i>	int(11)	No	NULL	Auto increment
2	Id_uji	int(11))	No		
3	nama	varchar(30)	No		
4	Kecepatan	float	No		
5	keterangan	varchar(150)	No		

Tabel 3.4 Tabel trek kendaraan

No	Field	Type	Null	Default	Extra
1	<i>Id_track_kendaraan</i>	int(11)	No	NULL	Auto increment
2	id_connection	int(11)	No		
3	id_kendaraan	int(11)	No		
4	keterangan	varchar(150)	No		

Tabel 3.5 Tabel uji

No	Field	Type	Null	Default	Extra
1	<i>Id_uji</i>	int(11)	No	NULL	Auto increment
2	nama	varchar(30)	No		
3	tanggal	Timestamp	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
4	background	varchar(60)	Yes		

B. Rancangan Antarmuka Sistem

1) Rancangan Antarmuka untuk Administrator

a. Rancangan form utama

Rancangan form utama dalam sistem ini dapat dilihat seperti gambar 3.6.

Penentuan Jalur Terpendek dengan Algoritma Dijkstra

File Manage Help Bobot Text Bobot Add Location Add Location Button Save Button Save Combo New Uji Button Load

Titik Awal Combo Titik Awal Titik Combo Titik Awal

Data Peta yang direpresentasikan sebagai Graph

Login User Pass Go

Gambar 3.6 Rancangan form Utama

Pada form ini admin dapat menginputkan location, connection beserta bobotnya. Setelah diinputkan semua dapat pula di simpan dan di updae sesuai dengan yang diinginkan.

b. Rancangan form menej titik

Rancangan form pengaturan titik dalam sistem ini dapat dilihat seperti gambar 3.7.

The image shows a web form titled "Menej Titik". At the top left, the title "Menej Titik" is displayed. Below the title is a table with two columns: "ID" and "Keterangan". The table is currently empty. To the right of the table, there are three buttons: "Edit", "Delete", and "Refresh". The text "Tabel Titik" is centered below the table.

Gambar 3.7 Rancangan form menej titik

Form ini berfungsi untuk mengatur location atau titik. apabila titik tersebut ingin di edit atau di hapus.

c. Rancangan form pengaturan connection

Rancangan form pengaturan connection dalam sistem ini dapat dilihat seperti gambar 3.8.

Menej Track

ID	Dari	Ke	Jarak	Keterangan
Tabel Track / Ruas				

Edit

Delete

Refresh

Gambar 3.8 Rancangan form menej track

Form ini berfungsi untuk mengatur connection atau sisi apabila sisi tersebut ingin di edit atau di hapus.

d. Rancangan form pengaturan kendaraan

Rancangan form pengaturan Kendaraan dalam sistem ini dapat dilihat seperti gambar 3.9.

Menej Kendaraan

ID	Nama	Keterangan
Tabel Track		

Edit

Delete

Refresh

Tambah baru

Nama

Keterangan

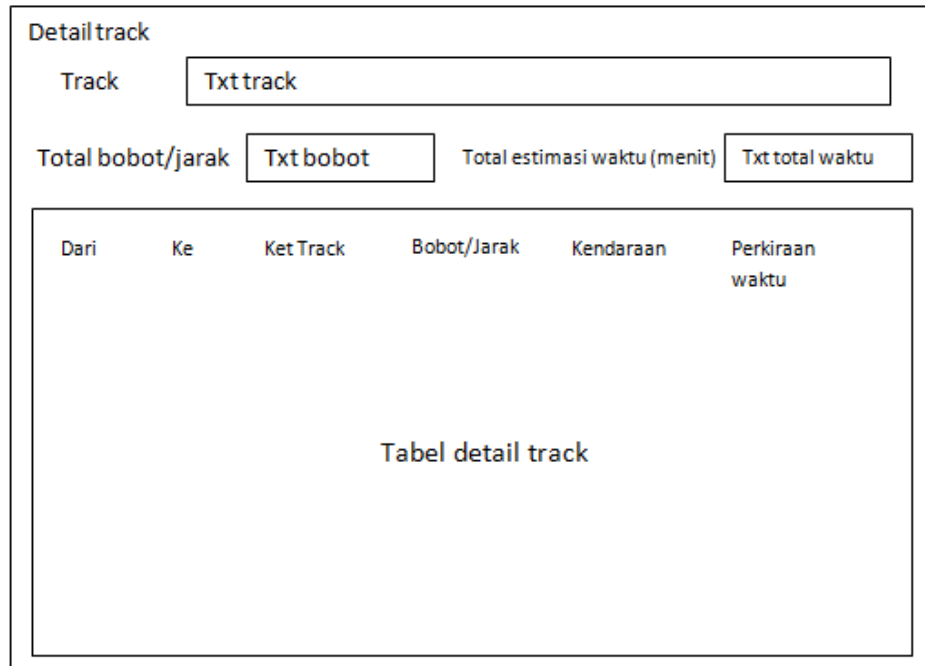
Simpan

Gambar 3.9 Rancangan Form menej Kendaraan

Form ini berfungsi untuk menambah atau mengedit kendaraan dan rutenya.

2) Rancangan form detail track untuk User

Rancangan form untuk user dalam sistem ini dapat dilihat seperti gambar 3.10.



The image shows a form titled "Detail track" with the following elements:

- A label "Track" followed by a text input field labeled "Txt track".
- A label "Total bobot/jarak" followed by a text input field labeled "Txt bobot".
- A label "Total estimasi waktu (menit)" followed by a text input field labeled "Txt total waktu".
- A table with the following headers: "Dari", "Ke", "KetTrack", "Bobot/Jarak", "Kendaraan", and "Perkiraan waktu".
- The table body is empty, with the text "Tabel detail track" centered below the header row.

Gambar 3.10 Rancangan form detail track untuk user

Form ini digunakan user untuk menginputkan titik awal dan titik akhir sesuai yang diinginkan.