

PERANCANGAN APLIKASI COMMUTER LINE ASSISTANT MENGUNAKAN ALGORITMA DIJKSTRA BERBASIS ANDROID

Ferry Irawan, Lukman Hakim
irawan_ferry@rocketmail.com, ferry.irawan@digdaya.co.id, lhakim@bundamulia.ac.id
Teknik Informatika Universitas Bunda Mulia

Abstract

The vast technological developments have contributed to the introduction of information technology in the scope of land transportations including electric train(KRL) which is provided by PT. KAI Commuter Jabodetabek.

Dijkstra Algorithm can be used to develop an android mobile application to give passengers easier experience in using KRL Commuter Line Jabodetabek and gain more efficiency in searching schedules and realtime positions of KRL.

Kata Kunci: *Android, Java, Dijkstra*

Pendahuluan

KAI (KAI, 2013)[4] menjelaskan PT KAI *Commuter* Jabodetabek (PT KCJ) mulai merealisasikan penerapan teknologi informatika dalam pelayanannya melalui kartu *COMMET (Commuter Line E-Ticketing)* dan tarif progresif KRL. PT KCJ juga turut meluncurkan sebuah Aplikasi Posisi KRL *Realtime* yang dapat diakses melalui website resmi KRL *Commuter Line* Jabodetabek.

Dalam penerapannya, sistem-sistem tersebut juga memiliki beberapa kekurangan sehingga sistem ini perlu dikembangkan lagi. Penerapan tarif progresif masih membingungkan penumpang, Aplikasi Web Posisi *Realtime* KRL dan Jadwal KRL yang disediakan PT KCJ juga tidak praktis dalam penggunaannya. Melihat kekurangan-kekurangan tersebut, baik dari kartu *COMMET* maupun Aplikasi Posisi *Realtime* KRL, sistem ini perlu dikembangkan demi mendapatkan efisiensi dalam penggunaan sistem-sistem tersebut. Salah satu solusinya yaitu melalui pengembangan aplikasi mobile berbasis android yang tingkat

penggunaannya tergolong tinggi di kalangan penumpang KRL.

METODE

Cormen (Cormen, 2009)[1] mengatakan sebuah masalah pencarian jarak terdekat dalam sebuah grafik dimana diandaikan tidak ada jarak antar simpul yang negatif dapat diselesaikan menggunakan sebuah algoritma *greedy* yang disebut algoritma Dijkstra.

Dijkstra (Dijkstra, 1959)[2], menjelaskan algoritma Dijkstra menentukan jarak terdekat dari suatu sumber s melalui perhitungan beban (jarak) masing-masing simpul saat ditemukan percabangan. Setiap jarak dari simpul terpendek akan tambahkan nilainya ke dalam sebuah vertex yang hasil akhirnya merupakan jarak terpendek dari sumber s ke tujuan.

Pada awalnya, algoritma ini hanya menentukan jarak terpendek saja, namun dengan sedikit modifikasi, algoritma Dijkstra juga dapat digunakan untuk menentukan jalur mana yang dilewati dari sumber ke tujuan untuk mencapai jarak terpendek tersebut.

Dalam penggunaannya, algoritma Dijkstra mencari jarak terdekat dari simpul awal ke seluruh simpul lainnya, tidak hanya ke simpul tujuan saja, maka dari itu algoritma Dijkstra termasuk dalam metode pencarian *Greedy*.

Sebagaimana telah disebutkan oleh Cormen & Thomas, algoritma Dijkstra bekerja dalam sebuah grafik yang terdiri dari simpul-simpul yang berhubungan. Rute *loopleft Commuter Line* Jabodetabek merupakan sebuah lintasan transportasi yang menghubungkan antar beberapa stasiun. Stasiun dapat diibaratkan sebagai simpul-simpul dan lintasan antar stasiun tersebut membuat sebuah rute yang dapat diibaratkan sebagai sebuah grafik. Disebutkan juga pencarian rute terdekat menggunakan algoritma Dijkstra bekerja dalam sebuah grafik yang diandaikan tidak ada jarak negatif antar simpul, dalam pembahasan di dalam ruang lingkup KRL, sebuah KRL hanya menempuh satu arah dalam satu kali perjalanan. Satu arah ini menunjukkan bahwa KRL selalu menempuh jarak positif dari stasiun awal hingga stasiun akhir hingga algoritma Dijkstra sesuai untuk diterapkan dalam pencarian rute terdekat dalam lintas *loopleft KRL Commuter Line* Jabodetabek.

PEMBAHASAN

Algoritma Dijkstra dapat diterapkan untuk melakukan perhitungan tarif progresif KRL yang dihitung berdasarkan jarak terdekat dari stasiun awal hingga stasiun tujuan yang ditempuh oleh penumpang.

Selain itu juga dapat juga diterapkan untuk menentukan jalur terdekat yang ditempuh oleh KRL. Jalur terdekat ini kemudian dapat digunakan untuk menentukan rute transit yang dapat ditempuh oleh KRL. Selain rute transit, jalur terdekat tersebut juga dapat menentukan perkiraan waktu tiba KRL di stasiun tujuan dengan memodifikasi penerapan algoritma

Dijkstra dengan menambahkan sebuah variabel waktu yang dihitung berdasarkan stasiun-stasiun yang ditempuh melalui jalur terdekat.

Pada rute *loopleft Commuter Line* Jabodetabek, pencarian jarak terdekat menggunakan algoritma Dijkstra dapat dilakukan berdasarkan cara berikut:

Stasiun-stasiun aktif yang disinggahi KRL dideklarasikan sebagai sebuah simpul yang saling berhubungan dengan stasiun berikutnya. Di antara simpul-simpul stasiun tersebut, diberikan jarak yang sama yaitu masing-masing 1 satuan jarak stasiun, dan tiap-tiap jarak antar simpul tersebut diberikan nilai waktu tempuh yang berbeda dalam satuan menit sesuai dengan lama tempuh KRL dari stasiun satu ke stasiun lainnya.

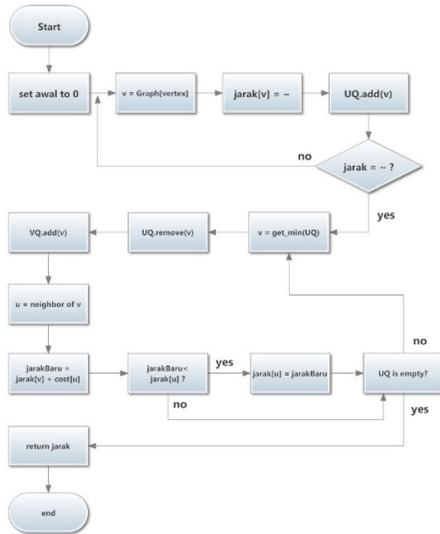
Dari simpul-simpul stasiun tersebut, dibuat grafik yang disesuaikan dengan arah perjalanan KRL. Dalam hal ini, dibuat 2 buah grafik yang menggambarkan arah maju dan arah balik dari sebuah perjalanan KRL.

Dengan terbentuknya grafik perjalanan KRL tersebut, dapat dijalankan algoritma Dijkstra untuk menentukan jarak terdekat dari simpul stasiun awal yang ditentukan ke seluruh simpul lainnya secara *greedy*.

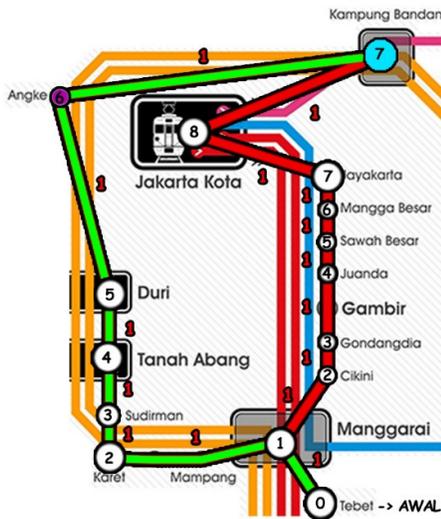
Dari hasil tersebut dapat dipilih salah satu stasiun sebagai stasiun akhir dan dapat diperoleh rute terdekat dari stasiun awal menuju stasiun tersebut. Selain itu, dengan disertakannya variabel waktu tempuh antar simpul stasiun, dapat ditentukan juga perkiraan waktu tempuhnya. Langkah-langkah algoritma Dijkstra ditunjukkan dalam flowchart pada gambar 1.

Contoh penerapan algoritma Dijkstra dalam rute *loopleft KRL Commuter Line Jabodetabek* dapat ditunjukkan dalam gambar 2.

Hasil yang ditunjukkan dalam gambar tersebut diperoleh melalui *runtime* algoritma Dijkstra yang ditunjukkan dalam tabel 1.



Gambar 1. Flowchart Algoritma Dijkstra



Gambar 2. Contoh pencarian rute terdekat TEB-KPB menggunakan algoritma Dijkstra

Tabel 1. Tabel runtime algoritma Dijkstra TEB-KPB

PSEUDOCODE	KETERANGAN	JARAK FINAL														
		TEB	MRI	CKI	GDD	JUA	SW	MGB	JYK	JAK	KRT	SUD	THB	DU	AK	KPB
Line 1-4	jarak[v] = -	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---
Line 6	awal = TEB	0	---	---	---	---	---	---	---	---	---	---	---	---	---	---
Line 8-19	v = TEB	0	1	---	---	---	---	---	---	---	---	---	---	---	---	---
Line 8-19	v = MRI	0	1	2	---	---	---	---	---	---	---	---	---	---	---	---
Line 8-19	v = CKI	0	1	2	3	---	---	---	---	---	---	---	---	---	---	---
Line 8-19	v = KRT	0	1	2	3	4	---	---	---	---	---	---	---	---	---	---
Line 8-19	v = GDD	0	1	2	3	4	5	---	---	---	---	---	---	---	---	---
Line 8-19	v = SUD	0	1	2	3	4	5	6	---	---	---	---	---	---	---	---
Line 8-19	v = JUA	0	1	2	3	4	5	6	7	---	---	---	---	---	---	---
Line 8-19	v = THB	0	1	2	3	4	5	6	7	8	---	---	---	---	---	---
Line 8-19	v = SW	0	1	2	3	4	5	6	7	8	9	---	---	---	---	---
Line 8-19	v = DU	0	1	2	3	4	5	6	7	8	9	10	---	---	---	---
Line 8-19	v = MGB	0	1	2	3	4	5	6	7	8	9	10	11	---	---	---
Line 8-19	v = AK	0	1	2	3	4	5	6	7	8	9	10	11	12	---	---
Line 8-19	v = JYK	0	1	2	3	4	5	6	7	8	9	10	11	12	13	---
Line 8-19	v = KPB	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Line 8-19	v = JAK	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Line 21	UQ = empty	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Ket:
 = simpul unvisited
 = simpul tetangga dari simpul v, masih bersifat unvisited
 = simpul sudah dikenali, namun masih bersifat unvisited
 = simpul bersifat visited
 = simpul dengan jarak minimum, dikeluarkan dari unvisited
 = jarak bertambah besar

Relasi	Posisi	Jadwal
DP-JNG	Ber. Cawang	05:14
	2 Stasiun	05:17
B00-JAK	Ber. Pasar Minggu	05:21
	5 Stasiun	05:25
B00-JNG	Ber. Lenteng Agung	05:29
	7 Stasiun	05:32
B00-JNG	Ber. Depok	05:43
	12 Stasiun	05:47
B00-JAK	Ber. Bojong Gede	05:54
	14 Stasiun	05:59
B00-JNG	Ber. Cilebut	06:01
	15 Stasiun	06:04

Gambar 3. Window Posisi Realtime KRL

Pada program disediakan informasi posisi realtime KRL yang ditunjukkan pada gambar 3. Pada informasi ini disediakan informasi mengenai relasi komuter pada jadwal tertentu berada pada posisi dimana.

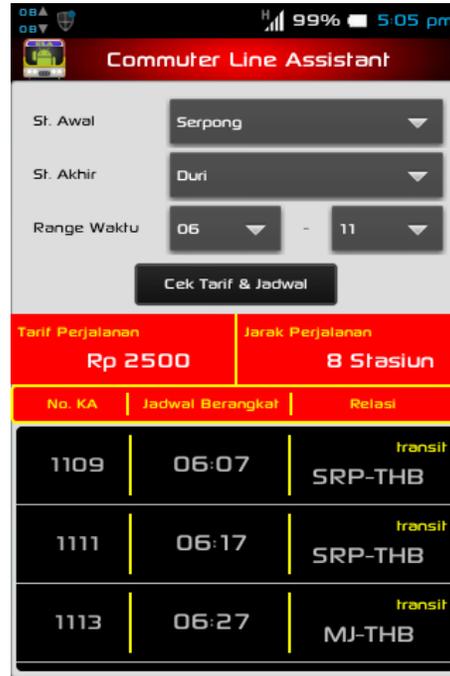
HASIL

Dengan menerapkan pembahasan tersebut menggunakan bahasa pemrograman *java* berbasis *Android*, dihasilkan aplikasi *Commuter Line Assistant* dengan fitur sebagai berikut:

1. Cek jadwal KRL
2. Perhitungan tarif progresif KRL
3. Informasi rute transit
4. Cek posisi KRL *realtime*
5. Perhitungan delay perkiraan jadwal tiba KRL
6. Komentar KRL dengan integrasi dengan *Facebook*

Hasil tampilan aplikasi dapat dilihat pada gambar 3, gambar 4, gambar 5 dan gambar 6.

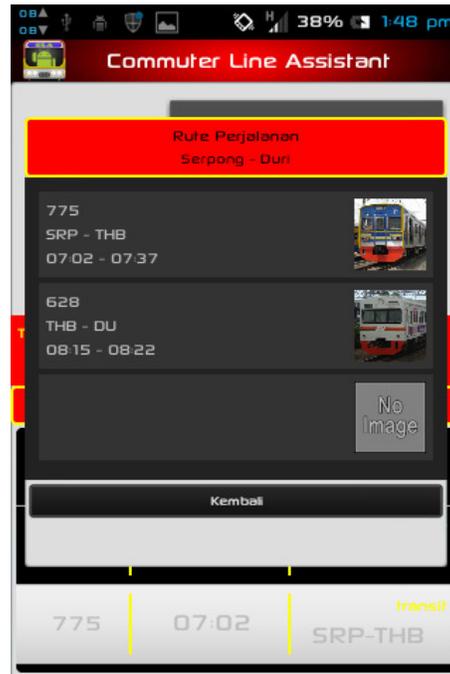
Gambar 4 adalah tampilan informasi *Commuter Line*, berisi informasi Asal dan tujuan, gambar 5 adalah informasi Jadwal dan Tarif KRL, berisi informasi nomor KA, Jadwal berangkat dan relasi.



Gambar 5. *Window* Jadwal&Tarif KRL



Gambar 4. *Window* Informasi KRL



Gambar 6. *Window* Informasi Rute Transit KRL

Gambar 6 berisi informasi rute transit KRL, dari suatu rute. Berisi informasi stasiun asal dan stasiun tujuan serta dimana KRL akan transit.

SIMPULAN

1. Aplikasi *Commuter Line Assistant* dapat memudahkan pengguna layanan jasa KRL *Commuter Line* Jabodetabek dalam mencari jadwal, mengetahui posisi KRL, memeriksa rute transit, dan mengecek tarif yang dikenakan sesuai dengan tarif progresif KRL *Commuter Line* Jabodetabek.
2. Algoritma Dijkstra dapat diterapkan dalam pencarian rute terdekat di ruang lingkup KRL *Commuter Line* lintas Jabodetabek dan dapat diterapkan dalam bahasa pemrograman Java berbasis *Android*.

SARAN

1. Pembenahan layout yang lebih *user-friendly*
2. Perbaikan bug-bug dalam aplikasi

3. Penambahan fitur-fitur baru
4. Dapat selalu dikembangkan mengikuti sistem yang saat ini berjalan.

DAFTAR PUSTAKA

- [1] Cormen, Thomas H. (2009). *Introduction to Algorithms. 3rd Edition*. New York: McGraw-Hill.
- [2] Dijkstra, E. W. (1959). *A note on two problems in connexion with graphs. Numerische Mathematik 1*. pp 269-271.
- [3] Google Inc. (2013). *Android. The World's Most Popular Mobile Platform*. <http://developer.android.com/about/index.html> , (diakses 24 Juli 2013).
- [4] PT KAI & PT KCJ. (2013). *1 Februari 2012 Ktb/Kls Diganti Menjadi Kartu Commet*. <http://www.krl.co.id/BERITA-TERKINI/1-februari-2012-ktbkls-diganti-menjadi-kartu-commet.html>, (diakses 18 Juli 2013).