 JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**SURABAYA**

**USULAN TUGAS AKHIR**

1. **IDENTITAS PENGUSUL**

Nama : **Rizki Pratiwi**

NRP : **5107 100 056**

Dosen Wali : **Ir. Muchammad Husni M.Kom.**

1. **JUDUL TUGAS AKHIR**

***”Aplikasi Mobile Pencarian Rute Transportasi Umum dengan Algoritma Best-Path Planning pada Platform Android”***

1. **PENDAHULUAN**

## 3.1 Latar Belakang Masalah

Di zaman yang teknologinya semakin berkembang membuat suatu aplikasi semakin berkembang dan terus berkembang. Apalagi di zaman seperti sekarang aplikasi mobile sangat menarik perhatian masyarakat luas. Hal ini membuat orang untuk terus mencari dan mendapatkan aplikasi terbaru.

Berawal dari susahnya mengetahui jalur transportasi umum apa saja yang bisa digunakan untuk menempuh suatu tujuan di kota khususnya kota Surabaya, membuat penulis tergerak untuk mengembangkan aplikasi pencarian rute transportasi umum ini. Aplikasi ini nantinya akan sangat memudahkan orang-orang yang baru tinggal di Surabaya.

Aplikasi ini nantinya akan berjalan di platform Android. Dengan menggunakan GPS, peta kota Surabaya, dan data jalur transportasi umum yang ada di Surabaya maka akan menciptakan aplikasi yang akan berguna sebagai *guide*. Hanya dengan menjalankan aplikasi ini di handphone ber-*platform* Android maka nantinya pengguna akan mendapati tahapan-tahapan dalam pencarian rute transportasi umum. Hanya dengan memasukkan nama tempat tujuan, pengguna akan bisa langsung mengetahui transportasi umum apa saja yang melewati jalan dimana pengguna berada (karena langsung terdeteksi dari gps) yang dapat digunakan untuk sampai ke tempat tujuan.

Dengan meng-implementasi-kan algoritma Best-Path Planning, maka aplikasi ini nantinya akan diwujudkan sehingga diharapkan dapat memudahkan pengguna smart phone dalam mencari jalur transportasi umum di Surabaya.

## 3.2 Perumusan Masalah

Permasalahan yang diangkat dalam pembuatan Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana menentukan hub-hub dari seluruh rute transportasi umum yang ada di Surabaya.
2. Bagaimana menentukan rute dari daerah asal ke tujuan.
3. Bagaimana menentukan rute dari daerah asal ke tujuan jika tidak ada rute yang sekali jalan.
4. Bagaimana mengaplikasikan tiap fase dari algoritma Best-Path Planning.

## 3.3 Batasan Masalah

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, diantaranya sebagai berikut:

Aplikasi ini berjalan di platform Android.

Aplikasi ini merupakan implementasi dari algoritma *Best-Path Planning for Public Transportation Systems* untuk pencarian rute transportasi umum di Surabaya.

Rute transportasi umum yang digunakan berupa angkutan umum (lyn) dan bus dalam kota Surabaya.

Lokasi-lokasi yang dimungkinkan untuk digunakan sebagai input, adalah lokasi-lokasi yang ada pada [*Google Maps*™](http://www.google.co.id/url?sa=t&source=web&cd=8&ved=0CE0QFjAH&url=http%3A%2F%2Fwww.mapds.com.au%2FProducts%2FGoogleMapsGoogleEarth%2Ftabid%2F179%2FDefault.aspx&rct=j&q=google%20maps%20tm&ei=Yk-eTbHyKsHsrQe3oKj7Ag&usg=AFQjCNHoGY7HSJVksapqO45iByh19mtAFA&sig2=GDVrLL7-qNFeYM0mGd-Wjw&cad=rja).

Hasil pencarian rute dibatasi sampai dengan dua kali transfer.

## 3.4 Tujuan Tugas Akhir

Tujuan pembuatan Tugas Akhir ini adalah mengimplementasikan algoritma *Best-Path Planning* untuk melakukan pencarian rute transportasi umum di Surabaya pada handphone dengan platform Android.

1. **TINJAUAN PUSTAKA**

Ada cukup banyak paper yang membahas mengenai pencarian rute dengan menggunakan algoritma yang bermacam-macam. Pada Tugas Akhir ini, akan digunakan salah satu dari algoritma tersebut yang disebut sebagai Best-Path Planning.

Algoritma pencarian pada umumnya mengasumsikan bahwa kita dapat bergerak secara bebas sehingga pembentukan partial-path dilakukan dari lokasi terakhir ke tetangga-tetangganya untuk kemudian membentuk partial-path lainnya. Partial-path terbaik kemudian dipilih untuk diteruskan ke partial-path selanjutnya. Tanpa strategi yang bagus, hal ini dapat menyebabkan algoritma pencarian menjadi tidak efisien jika lokasi yang dapat diambil dari sebuah lokasi lainnya dibatasi. Skenario tersebut dapat terjadi pada sistem transportasi publik dimana trayek telah ditentukan sehingga seorang penumpang tidak dapat menyuruh sang sopir mengganti rutenya untuk kepentingan penumpang tersebut. Untuk itu, algoritma Best-Path Planning ini menggunakan matriks untuk menggambarkan batasan tersebut sehingga dapat membuat path planning dengan lebih efisien.

Sebenarnya ada beberapa algoritma yang bisa digunakan dalam pencarian rute. Di paper ini saja ada tiga algoritma yang berkembang sampai akhirnya mendapati algoritma yang benar-benar sangat efisien dalam penentuan rute.

Paper ini menggunakan beberapa istilah untuk menjelaskan beberapa aspek yang ada pada rute yang terbatas. Pertama, dengan mengasumsikan bahwa terdapat sebuah set lokasi yang dilalui oleh sebuah set rute. Untuk setiap lokasi tersebut diberikan sebuah nomor identifikasi unik dan menentukan set dari nomor-nomor tersebut sebagai S. Hal yang sama juga lakukan dengan memberikan nomor identifikasi unik terhadap setiap rute yang melalui sebuah daerah dan menentukan set dari nomor-nomor tersebut sebagai R. Paper ini juga memberikan nomor urut K(r,s) untuk setiap lokasi s yang dilalui oleh sebuah rute r. Nilai K pada terminal keberangkatan untuk tiap-tiap rute adalah 1. Jika sebuah rute r tidak melalui lokasi s, maka K(r,s) = 0. Oleh karena itu, K(r,s1) > K(r,s2) > 0 menyatakan bahwa rute r melayani dari s2 ke s1. Berdasarkan fungsi K tersebut, ditetapkan beberapa fungsi yang sangat berguna yaitu :

1. Untuk sebuah lokasi s, set dari rute yang melayani lokasi s ditetapkan sebagai berikut.

SR(s) = { r|K(r,s) > 0, r∈R }

1. Suatu set dari pemberhentian umum yang dilalui oleh dua buah rute r1 dan r2 ditetapkan sebagai berikut.

CS(r1,r2) = { s|K(r1,s) > 0 dan K(r2,s) > 0, s∈S }

1. Untuk dua buah lokasi s dan t, fungsi berikut mengembalikan set dari rute langsung dimana kita dapat bepergian dari s ke t tanpa perlu melakukan transfer transportasi umum.

DS(s,t) = { r|K(r,t) > K(r,s), r∈R }

Fungsi-fungsi diatas akan digunakan pada algoritma yang digunakan pada Tugas Akhir ini.

Sebenarnya ada tiga algoritma yang bisa digunakan untuk pencarian rute. Namun dari ketiga algoritma hanya ada satu algoritma yang dianggap oleh penulis paper sebagai algoritma yang benar-benar efisien untuk pencarian rute. Algoritma yang ada dalam paper ini antara lain :

* + - 1. **Algoritma 1 (HPlanning)** – anggap O sebagai lokasi asal (origin) dan D sebagai lokasi tujuan (destination)
  1. Kasus sepele: Jika O = D, munculkan pesan error dan tidak menghasilkan rute apapun.
  2. Kasus sederhana: jika DS(O,D) tidak kosong, mengembalikan nilai rute pada set.
  3. Kasus sekali tranfer: jika ∃i Є SR(O), ∃j Є SR(D), dan ∃s Є CS(i,j) sehingga DS(O,s) dan DS(s,D) tidak kosong, maka terdapat solusi sekali transfer.
  4. Kasus khusus: mencari rute dengan menggunakan hub.
     + 1. **Algortima 2 (CPlanning)** - anggap O sebagai lokasi asal (origin) dan D sebagai lokasi tujuan (destination). Asumsikan bahwa i ∈ SR(O) dan j ∈ SR(D). Algoritma ini menggunakan matriks konektivitas dimana telah dirancang sebuah ordinal number untuk setiap rute, dan memberi cell Ci,j pada matriks konektivitas C nilai 1 jika seseorang dapat pergi dari lokasi i ke lokasi j. Secara default, Ci,i diberi nilai 0 untuk setiap i. Matriks konektivitas berisi batas bawah dari jumlah transfer yang dibutuhkan untuk melakukan perjalanan antara dua buah lokasi.

Algoritma :

* 1. Kasus sepele: Jika O = D, munculkan pesan error dan tidak menghasilkan rute apapun.
  2. Kasus sederhana: jika DS(O,D) tidak kosong, mengembalikan nilai rute pada set.
  3. Sekali transfer. Jika Ci,j > 0, periksa jika ∃s ∈ CS(i, j) sehingga DS(O, s) dan DS(s, D) tidak kosong. Jika ya, maka terdapat solusi sekali transfer.
  4. Dua kali transfer. Jika  > 0, periksa jika ∃r ∈ CR1(i, j) sehingga ∃s ∈ CS(i, r), ∃t ∈ CS(r, j), DS(O, s) ≠ ∅, DS(s, t) ≠ ∅, dan DS(t, D) ≠ ∅. Jika ya, maka terdapat sebuah solusi dua kali transfer.

Meskipun Algoritma 2 telah menuju kepada realisasi sistem path-planning secara otomatis, tidaklah mudah untuk mengaplikasikan algoritmanya pada best-path planning dimana properti dari suatu path mengandung hal lain selain jumlah transfer. Contoh yang paling umum adalah menghitung best-path yang mana biaya perjalanannya meliputi waktu perjalanan stokastik.

Jika diperhatikan, Algoritma 2 menjadi bias untuk path yang memerlukan transfer lebih banyak. Walaupun path yang memerlukan transfer lebih sedikit cenderung merupakan path yang lebih cepat, tetap saja ada pengecualian. Untuk menjamin optimalitas dari solusi, kita tidak boleh mengabaikan path yang memerlukan multiple transfer. Di sisi lain, mungkin dibutuhkan matriks konektivitas dengan dimensi sangat tinggi untuk menentukan properti potensial dari rencana perjalanan yang mungkin tidak terlalu berguna. Meskipun hal ini dapat dicapai, akan tetapi secara komputasi akan sangat mahal untuk melakukannya.

* + - 1. **Algoritma 3 (TPlanning)** - anggap o dan d sebagai nomor yang ditetapkan untuk lokasi asal (origin) dan lokasi tujuan (destination). Algoritma ini menggunakan matriks transisi. Nomor urut ditetapkan untuk tiap lokasi. Kemudian, menetapkan cell Qi,j dari matriks transisi Q bernilai jumlah rute langsung (direct route) yang seseorang dapat ambil untuk melakukan perjalanan dari i ke j. Untuk Qi,i ditentukan bernilai 0 sebagai default-nya. Qi,j sangat superior jika dibandingkan dengan Tx,y yang dijelaskan pada bagian sebelumnya. Tx,y adalah batas bawah jumlah transfer minimal yang dibutuhkan untuk berpergian dari X ke Y. Lain halnya, Qx,y secara meyakinkan menunjukkan dengan tepat jumlah transfer minimal yang dibutuhkan untuk perjalanan yang sama.

Algoritma :

* 1. Kasus sepele: Jika o = d, munculkan pesan error dan tidak menghasilkan rute apapun.
  2. Kasus langsung: jika Qo,d = 1, mengembalikan nilai pada DS(o, d).
  3. Kasus sekali transfer: jika Qo,d = 2, maka terdapat sebuah lokasi m sehingga Qo,m = 1 dan Qm,d = 1. Gabungkan rute pada DS(o, m) dan rute pada DS(m, d) untuk mendapatkan solusi sekali transfer.
  4. Kasus dua kali transfer: jika Qo,d = 3, maka terdapat lokasi lain m1 dan m2 sehingga Qo,m1 = 1, Qm1,m2 = 1, dan Qm2,d = 1. Gabungkan sebuah rute dari setiap DS(o, m1), DS(m1, m2), DS(m2,d) untuk mendapatkan solusi dua kali transfer.

Algoritma ini lengkap dan optimal dalam hal selalu mendapatkan solusi rute dengan transfer paling sedikit. Algoritma TPlanning ini lebih baik dari CPlanning karena TPlanning membutuhkan pemeriksaan yang lebih sedikit pada saat dijalankan. Oleh karena itu, algoritma inilah yang nantinya akan digunakan pada Tugas Akhir kami.

Diagram alur algoritma TPlanning dapat digambarkan sebagai berikut :

Ya

Pesan Error

o = d?

Tidak

Qo,d = 1?

Ya

Rute = DS(o,d)

Tidak

Qo,d = 2?

Ya

Rute = DS(o, m) + DS(m, d)

Tidak

Ya

Qo,d = 3?

Rute = DS(o, m1) + DS(m1, m2) + DS(m2,d)

Tidak

Pesan bahwa rute lebih dari dua kali transfer / tidak ada rute

1. **METODE PENELITIAN**

Metode penelitian yang digunakan dalam Tugas Akhir ini bisa dipaparkan berikut :

1. **Studi Literatur**

Pada tahap ini dilakukan pencarian dan pemahaman mengenai hal-hal yang berhubungan dengan permasalahan yang ada antara lain :

Pemahaman algotitma Best-Path Planning, android, GPS (Global Positioning System), dan penerapan algoritma Best-Path Planning dalam pencarian rute transportasi umum. Literatur yang digunakan meliputi paper, dokumentasi Tugas Akhir yang berhubungan dengan pencarian rute transportasi umum, GPS, dan dokumentasi dari internet.

1. **Perancangan Perangkat Lunak dan Desain Sistem**

Pada tahap ini dilakukan perancangan sistem dengan menggunakan studi literatur dan mempelajari konsep algoritma, teknologi dan software yg sudah ada. Dari proses tersebut dilakukan desain aplikasi, rumusan rancangan dan pemecahan masalah dari algoritma yang dipakai.

1. **Pengimplementasian**

Pada tahap pengimplementasian ini sudah dilakukan pembuatan aplikasi yang mengimplementasikan dari rancangan yang sudah dibuat sebelumnya dengan berbekal pedoman pada tahap-tahap sebelumnya yaitu rancangan aplikasi, bahasa pemrograman yang digunakan dan bagaimana penerapannya pada teknologi yang akan digunakan untuk membuat aplikasi.

1. **Uji Coba dan Evaluasi**

Pada tahap ini akan dilakukan uji coba terhadap aplikasi yang sudah dibuat dan akan di evaluasi dengan berdasarkan tahapan-tahapan yang sudah ada. Jika di dalam tahap uji coba masih terdapat kekurangan ataupun kesalahan, makan akan diperbaiki demi keberhasilan dan kelayakan aplikasi sesuai dengan tujuan pembuatan Tugas Akhir ini.

1. **Penyusunan Buku Tugas Akhir**

Pada tahap ini dilakukan untuk pembuatan laporan dari semua dasar teori dan metode yang digunakan serta hasil – hasil yang diperoleh selama pengerjaan Tugas Akhir. Laporan Tugas Akhir ini akan dibagi menjadi beberapa bab sebagai berikut:

1. Bab I, Pendahuluan, berisi latar belakang, permasalahan, tujuan, batasan permasalahan, metodologi, dan sistematika penulisan.
2. Bab II, Landasan Teori, akan dibahas dasar ilmu yang mendukung pembahasan Tugas Akhir ini.
3. Bab III, Desain Aplikasi.
4. Bab IV, Implementasi dari aplikasi yang telah dibuat, akan dilakukan pembuatan aplikasi yang sesuai dengan permasalahan dan batasannya yang telah dijabarkan pada bab pertama.
5. Bab V, Uji coba dan analisa hasil, akan dilakukan uji coba berdasarkan parameter-parameter yang ditetapkan, dan kemudian dilakukan analisa terhadap hasil uji coba tersebut.
6. Bab VI, Penutup, berisi kesimpulan yang dapat diambil dari Tugas Akhir ini beserta saran untuk pengembangan selanjutnya.
7. **RINGKASAN TUGAS AKHIR**

Dalam Tugas Akhir ini akan dibuat sebuah aplikasi pencarian rute transportasi umum kota Surabaya. Aplikasi ini dirancang untuk berjalan pada *smart phone* dengan *platform* Android. Dengan aplikasi ini, pengguna akan diberikan informasi berupa rute-rute transportasi umum yang tersedia dari lokasinya sekarang ke tempat tujuan. Aplikasi ini dapat memberikan informasi rute transportasi umum sampai dengan dua kali transfer. Selain dari tempatnya berada sekarang, user juga dapat memasukkan inputan dari lokasi awal yang diinginkan menuju lokasi tujuan. Algoritma yang digunakan dalam aplikasi ini ialah *Best-Path Planning* sehinng diharapkan proses pencarian rutenya menjadi lebih cepat jika dibandingkan dengan algoritma-algoritma lainnya.

1. **JADWAL KEGIATAN**

Tugas Akhir ini diharapkan bisa dikerjakan menurut jadwal sebagai berikut:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan (2011) | | | | | | | | | | | | | | | |
| Mar ‘11 | | | | Apr’11 | | | | Mei’11 | | | | Jun’11 | | | |
| 1. | Penyusunan Proposal Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**
2. Chao-Lin Liu, 2002, *Best-Path Planning for Public Transportation Systems,* In Proceedings of the Fifth International IEEE Conference on Intelligent Transportation System.
3. Pratomo, Baskoro Adi, 2008, *Perancangan dan Pembuatan Perangkat Lunak Informasi Angkutan Umum dengan Memakai Cell ID untuk Menentukan Posisi*.

**LEMBAR PENGESAHAN**

Surabaya, 05 April 2011

Menyetujui,

# Dosen pembimbing I Dosen pembimbing II

Ary Mazharudin Shiddiqi S.Kom, M.Kom Baskoro Adi Pratomo, S.Kom, M.Kom

NIP. 19810620 2005011 003 NIP. 510000003

# 