

# SISTEM INFORMASI GEOGRAFIS PENCARIAN LOKASI AGEN BUS DAN TRAVEL TERDEKAT DI KOTA SEMARANG BERBASIS MOBILE DENGAN METODE DIJKSTRA

M. Rizki Kurniawan.<sup>1)</sup>, Oky Dwi Nurhayati <sup>2)</sup>, Kurniawan Teguh Martono<sup>3)</sup>

Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro  
Jalan Prof. Sudharto, Tembalang, Semarang, Indonesia

E – mail : [risknoscale@gmail.com](mailto:risknoscale@gmail.com)

*Abstract – Nowadays, the mobility of people who travel to a place through road transportation is increasing, either private vehicle or public transportation services. By the high intensity of use the road transportation, especially the road transportation service in Semarang, in this cases are bus and travel, it makes the transportation service users often confused in determining transportation agency to be used, especially if user is in hurry. Therefore, transportation user need a mobile application that can provide bus or travel transportation agency locator services which nearby user location and provide destination route which is expected. The Bus and Travel agency – Based Applications Mobile Locator Using Dijkstra Method is the ultimate solution in the search of bus and travel agency in Semarang. The purpose of development this application is developing a search guidance system of bus and travel agency location which nearby user location that can provide an efficient route and save the time.*

*The Bus and Travel agency – Based Applications Mobile Locator is implemented using ionic framework that runs on Android. Sqlite as the media storage of this, allows user to perform agency data management. In Addition, it is supported by the relevance of agency data which is obtained from Department of Transportation, Communication, and Information Central of Java and transportation magazine site.*

*The use of Google Maps API supports search location using Dijkstra Method in search of bus and travel agency, that expected to facilitate the users find travel or bus agency as expected, and its information that's already available.*

**Keywords:** *Android, Sqlite, Google Maps API, Ionic Framework, Dijkstra*

## I. PENDAHULUAN

### A. LATAR BELAKANG

Era teknologi informasi saat ini mengalami perkembangan yang pesat. Faktor ketersediaan informasi dan kemudahan pengguna dalam mengakses informasi dimana dan kapanpun inilah yang menjadikan pemanfaatan teknologi informasi diterapkan hampir dalam semua bidang. Salah satu teknologi yang dapat dimanfaatkan untuk mengatasi hal ini adalah penggunaan teknologi nirkabel. Tidak terkecuali dalam menyajikan sebuah peta atau rute, dengan memanfaatkan teknologi nirkabel, dapat dikembangkan sistem informasi geografis berbasis *mobile*.

Saat ini banyak orang – orang yang berpergian baik menggunakan kendaraan pribadi maupun memanfaatkan layanan transportasi umum. Dengan tingginya mobiltas orang – orang yang berpergian ini, khususnya pengguna layanan transportasi umum di wilayah kota semarang, dalam hal ini transportasi bus dan travel, tentunya pengguna membutuhkan sebuah perangkat lunak *mobile* yang dapat memberikan layanan pencarian agen transportasi bus atau travel yang menyediakan rute perjalanan yang diinginkan. Pembuatan perangkat lunak ini diharapkan akan menjadi solusi yang dapat digunakan oleh pengguna layanan ini sebagai penunjuk jalan untuk meningkatkan efisiensi pencarian rute agen terdekat dari lokasi perangkat pengguna di wilayah kota semarang memanfaatkan metode pencarian *Dijkstra*.

### B. RUMUSAN MASALAH

Bagaimana menciptakan sebuah sistem informasi geografis berbasis *mobile* yang *user friendly* dan dapat membantu pengguna dalam melakukan pencarian rute tujuan agen travel atau bus terdekat di wilayah kota semarang dari lokasi perangkat pengguna?

### C. BATASAN MASALAH

Dalam penulisan tugas akhir ini, penulis membatasi pembahasan masalah sebagai berikut :

1. Sistem operasi perangkat pengguna menggunakan Android (4.0 atau lebih) yang mempunyai fitur GPS.
2. Sistem diimplementasikan menggunakan Ionic framework.
3. Algoritma yang digunakan adalah Algoritma Dijkstra.
4. Menggunakan Google Maps JavaScript API v3 sebagai library peta yang digunakan dalam pengembangan.
5. Sistem hanya memuat agen transportasi bus dan travel untuk wilayah kota semarang
6. Agen transportasi yang dijadikan objek penelitian adalah agen transportasi bus AKAP dan agen Travel yang melayani pemberangkatan dari lokasi agen berada
7. Rute keberangkatan agen berasal dari wilayah Semarang

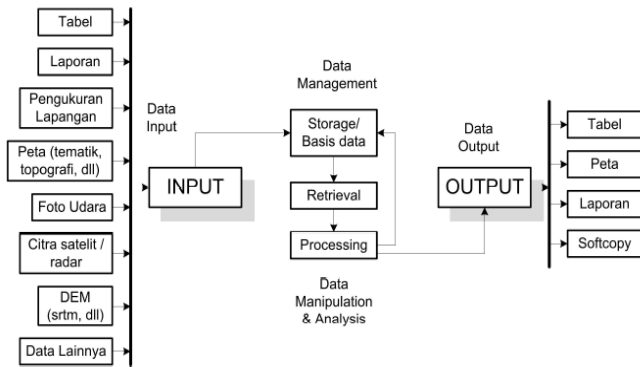
### D. TUJUAN MASALAH

1. Membantu pengguna dalam pencarian rute agen travel atau bus terdekat sehingga diharapkan dengan perangkat lunak ini dapat memberikan rute yang efisien dan menghemat waktu
2. Menciptakan dan mengimplementasikan sebuah perangkat lunak sistem informasi geografis berbasis *mobile*

## II. TINJAUAN PUSTAKA

### A. SISTEM INFORMASI GEOGRAFIS

SIG sebagai sistem komputer yang digunakan untuk memanipulasi data geografi. SIG mempunyai kemampuan untuk menghubungkan berbagai data pada suatu titik tertentu di bumi, menggabungkannya, menganalisa, dan akhirnya memetakan hasilnya. Data yang diolah pada SIG adalah data spasial yaitu sebuah data yang berorientasi geografis dan merupakan lokasi yang memiliki sistem koordinat tertentu, sebagai dasar referensinya. Sehingga aplikasi SIG dapat menjawab beberapa pertanyaan seperti lokasi, kondisi, tren, pola dan pemodelan. Kemampuan inilah yang membedakan SIG dengan sistem informasi lainnya. SIG dapat diuraikan menjadi beberapa subsistem, seperti yang terlihat pada Gambar 1 berikut.



Gambar 1 Ilustrasi Uraian Sub – sistem SIG

### B. GPS (Global Positioning System)

*Global Positioning System* (GPS) adalah suatu sistem radio navigasi penentuan posisi menggunakan satelit. GPS dapat memberikan posisi suatu objek di muka bumi dengan akurat dan cepat (koordinat tiga dimensi x,y,z) dan memberikan informasi waktu serta kecepatan bergerak secara kontinyu di seluruh dunia.

### C. PERANGKAT BERGERAK

Menurut Gartner (2003), perangkat bergerak secara umum dapat diklasifikasikan dalam 2 kategori, yaitu :

- [1]. Telepon selular dengan peningkatan kemampuan komputasi, termasuk menampilkan grafik dan kemampuan berinteraksi dengan grafik (seperti : *smartphone* dan *communicator*).
- [2]. Komputer portable, yang dapat diintegrasikan dengan kemampuan komunikasi audio – video (seperti : PDA, dan lain – lain).

Pada Umumnya perangkat bergerak yang digunakan untuk aplikasi klien kartografi memiliki sistem operasi, yang memberi pengguna sebuah antarmuka dan control sinkronisasi perangkat. Terdapat beberapa sistem operasi yang populer di pasaran, seperti : Symbian, Palm OS, Linux, Android, dan Windows Mobile.

### D. ANDROID

Android merupakan sistem operasi telepon seluler yang tumbuh di tengah sistem operasi lainnya yang berkembang dewasa ini. Sistem Operasi lainnya seperti Windows Mobile, iOS-iPhone, Symbian, dan masih banyak lagi juga menawarkan kekayaan isi dan keoptimalan berjalan di atas perangkat keras yang ada. Akan tetapi, sistem operasi yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh karena itu adanya keterbatasan dari

aplikasi pihak ketiga untuk mendapatkan data asli ponsel, berkomunikasi antar proses serta keterbatasan distribusi aplikasi pihak ketiga untuk *platform* mereka.

### E. ALGORITMA DIJKSTRA

Algoritma ini didasarkan pada representasi *adjacency matrix* untuk sebuah *graph*. Algoritma ini tidak hanya menemukan jalur terpendek dari satu verteks tertentu ke verteks lain, tetapi juga jalur terpendek dari verteks tertentu ke semua verteks lain. Algoritma ini sering digunakan pada *routing*. Algoritma dijkstra mencari lintasan terpendek dalam sejumlah langkah. Algoritma ini menggunakan strategi *greedy* sebagai berikut :

- Untuk setiap simpul sumber(*source*) dalam graf, algoritma ini akan mencari jalur dengan *cost* minimum
- antara simpul tersebut dengan simpul lainnya. Algoritma juga dapat digunakan untuk mencari total biaya(*cost*) dari lintasan terpendek yang dibentuk dari sebuah simpul ke sebuah simpul tujuan. Sebagai contoh, bila simpul pada graf merepresentasikan kota dan bobot sisi merepresentasikan jarak antara 2 kota yang mengapitnya, maka algoritma *Dijkstra* dapat digunakan untuk mencari rute terpendek antara sebuah kota dengan kota lainnya.

### F. BASIS DATA

Basis Data dapat didefinisikan dalam berbagai sudut pandang seperti berikut :

1. Himpunan kelompok data yang saling berhubungan yang diorganisasi sedemikian rupa sehingga kelak dapat dimanfaatkan dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi kebutuhan.

Komponen dari basis data adalah entitas dan atribut. Berikut adalah daftar entitas yang akan dirancang dalam basis data penelitian ini, seperti yang ditunjukkan pada Tabel 1.

TABEL 1 Daftar Entitas yang terlibat

No.	Nama Entitas	Keterangan
1.	Agen	Entitas yang berisi mengenai informasi agen travel dan bus yang terdapat di wilayah semarang.
2.	Jenis Agen	Entitas yang berisi mengenai jenis agen yang akan dimuat dan disimpan dalam sistem
3.	Profil	Entitas yang berisi mengenai profil agen travel dan bus.
4.	Rute Agen	Entitas yang memuat informasi tentang rute tujuan agen travel dan bus di wilayah semarang.
5.	Jenis Rute	Entitas yang memuat semua rute tujuan yang bisa dilayani oleh sistem
6.	Jalur	Entitas yang berisi mengenai mode jalur yang bisa dilewati ketika melakukan pencarian agen tujuan, contohnya tol atau jalan raya.
7.	Jelajah	Entitas yang berisi mengenai mode jelajah yang bisa dilakukan ketika melakukan pencarian agen tujuan.
8.	Opsi Map	Entitas yang berisi informasi mengenai informasi yang rinci ketika pengguna melakukan pencarian
9.	Parameter Pencarian	Entitas yang berisi mengenai jenis parameter pencarian yang disediakan dalam sistem.
10.	Spesifikasi Pencarian	Entitas yang berisi mengenai spesifikasi pencarian berdasarkan parameter yang dilakukan pengguna.

11.	Lokasi Perangkat	Entitas yang berisi mengenai informasi lokasi perangkat pengguna ketika melakukan pencarian
-----	------------------	---

Tabel 2 berikut ini adalah atribut dari entitas agen.

TABEL 2 Atribut dari entitas agen

Nama	Tipe	Keterangan
id_agen	TEXT	ID Agen, <i>Primary key</i>
id_jenis	INTEGER	ID Jenis agen, <i>Foreign Key</i>
nama_agen	TEXT	Nama agen
lat_lokasi	INTEGER	Latitude agen
long_lokasi	INTEGER	Longitude agen

Tabel 3 berikut ini adalah atribut dari entitas jenis agen.

TABEL 3 Atribut dari entitas jenis agen

Nama	Tipe	Keterangan
id_jenis	INTEGER	ID jenis, <i>Primary key</i>
Jenis_agen	TEXT	Jenis agen

Tabel 4 berikut ini adalah atribut dari entitas profil.

TABEL 4 Atribut dari entitas profil

Nama	Tipe	Keterangan
id_profil	TEXT	ID profil, <i>Primary key</i>
id_agen	TEXT	ID agen, <i>Foreign key</i>
alamat_agen	TEXT	Alamat agen
phone_agen_satu	TEXT	Telepon pertama agen
phone_agen_dua	TEXT	Telepon kedua agen

Tabel 5 berikut ini adalah atribut dari entitas rute dari agen bus dan travel.

TABEL 5 Atribut dari entitas rute

Nama	Tipe	Keterangan
id_rute	INTEGER	ID rute, <i>Primary key</i>
id_agen	TEXT	ID agen, <i>Foreign key</i>
asal_rute	TEXT	Rute asal
Id_tujuan_rute	INTEGER	ID rute tujuan, <i>Foreign Key</i>
flag	INTEGER	Penanda tujuan rute

Table 6 berikut ini adalah atribut dari entitas jenis rute .

TABEL 6 Atribut dari entitas jenis rute

Nama	Tipe	Keterangan
id_tujuan_rute	INTEGER	ID jalur, <i>Primary key</i>
Tujuan_rute	TEXT	Rute tujuan agen

Tabel 7 berikut ini adalah atribut dari entitas jalur.

TABEL 7 Atribut dari entitas jalur

Nama	Tipe	Keterangan
id_jalur	INTEGER	ID jalur, <i>Primary key</i>
mode_jalur	TEXT	Mode jalur

Tabel 8 berikut ini adalah atribut dari entitas jelajah .

TABEL 8 Atribut dari entitas jelajah

Nama	Tipe	Keterangan
id_jelajah	INTEGER	ID jelajah, <i>Primary key</i>
mode_jelajah	TEXT	Mode jelajah

Tabel 9 berikut ini adalah atribut entitas opsi map.

TABEL 9 Atribut dari entitas tabel opsi map

Nama	Tipe	Keterangan
id_opsi_map	INTEGER	ID jalur, <i>Primary key</i>
id_spesifikasi	INTEGER	ID spesifikasi, <i>Foreign Key</i>
id_lokasi_perangkat	INTEGER	ID lokasi perangkat pengguna, <i>Foreign Key</i>
id_kelajahan	INTEGER	ID kelajahan, <i>Foreign Key</i>
id_jalur	INTEGER	ID jalur, <i>Foreign Key</i>

Tabel 10 berikut ini adalah atribut entitas parameter pencarian.

TABEL 10 Atribut dari entitas tabel parameter pencarian

Nama	Tipe	Keterangan
id_param	INTEGER	ID parameter, <i>Primary key</i>
Jenis parameter	TEXT	Jenis parameter

Tabel 11 berikut ini adalah atribut entitas spesifikasi pencarian.

TABEL 11 Atribut dari entitas tabel spesifikasi pencarian

Nama	Tipe	Keterangan
id_spesifikasi	INTEGER	ID spesifikasi, <i>Primary key</i>
id_param	INTEGER	ID parameter, <i>Foreign Key</i>
jenis_spesifikasi	TEXT	Spesifikasi pencarian berdasarkan parameter yang dipilih

Tabel 12 berikut ini adalah atribut entitas lokasi perangkat.

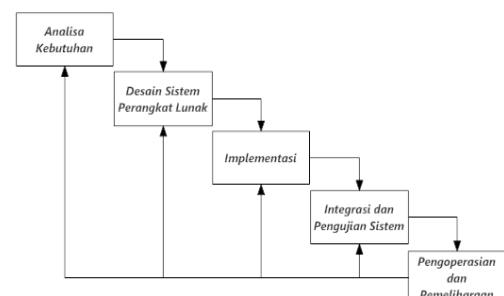
TABEL 12 Atribut dari entitas tabel lokasi perangkat

Nama	Tipe	Keterangan
id_lokasi_perangkat	INTEGER	ID lokasi perangkat, <i>Primary key</i>
lat_pencarian	INTEGER	Latitude pencarian
long_pencarian	INTEGER	Longitude pencarian

### III. ANALISA DAN PERANCANGAN SISTEM

#### A. TAHAPAN PENGEMBANGAN METODOLOGI

Terdapat beberapa tahapan yang dilakukan dalam menyusun sebuah perangkat lunak. Metode *Waterfall* adalah suatu proses pengembangan perangkat lunak berurutan, dimana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase – fase perencanaan, permodelan, implementasi (konstruksi), dan pengujian. Berikut adalah gambar pengembangan perangkat lunak atau linear. Pada Gambar 2 berikut merupakan tahapan metode *Waterfall*.



Gambar 2 Metode *Waterfall*

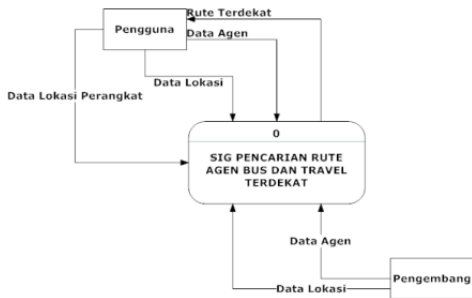
**B. DESAIN PERANGKAT LUNAK**

Permodelan perangkat lunak digunakan untuk memberikan tahapan yang jelas dalam pembuatan perangkat lunak. Berikut ini adalah permodelan yang digunakan dalam perangkat lunak, yaitu:

**1. DIAGRAM ALIR DATA**

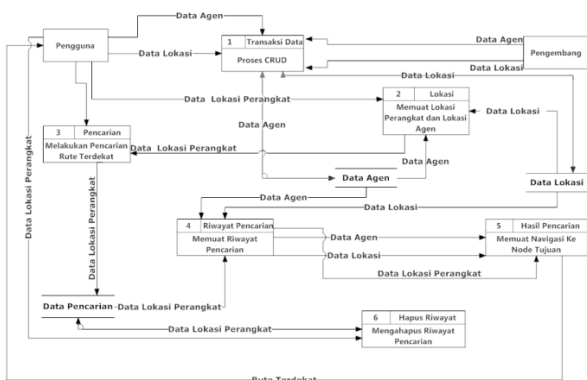
Diagram alir data merupakan perangkat – perangkat analisis dan perancangan yang terstruktur sehingga memungkinkan analis sistem memahami sistem dan subsistem secara visual sebagai suatu rangkaian aliran data yang saling berkaitan.

Diagram konteks merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, serta menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data-aliran data utama menuju dan dari sistem.



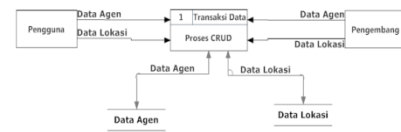
Gambar 3 Diagram Konteks

Gambar 3 menunjukkan diagram konteks dari SIG pencarian lokasi agen terdekat. Diagram tersebut menjelaskan bahwa sistem memiliki menggunakan 2 buah entitas, yaitu pengguna dan pengembang. Ketika pengguna melakukan instalasi aplikasi untuk pertama kalinya, sistem akan menginisiasi semua data agen dan data lokasi agen di wilayah Kota Semarang yang telah dimasukkan oleh pengembang. Data agen sendiri memuat informasi agen, seperti nama agen, alamat agen, rute agen, dan nomor telepon agen yang dapat dihubungi. Sedangkan Data lokasi memuat informasi letak agen bus dan travel. Data agen dan data lokasi yang telah dimasukkan tadi dapat dikelola oleh pengguna. Pengguna juga dapat melakukan proses pencarian lokasi agen terdekat dengan memberikan data lokasi perangkat pengguna ke sistem. Untuk detail dari diagram konteks dibuatlah diagram alir data level nol dan satu. Pada diagram alir data level nol, aplikasi ini memiliki 6 buah proses seperti yang ditunjukkan pada Gambar 4.



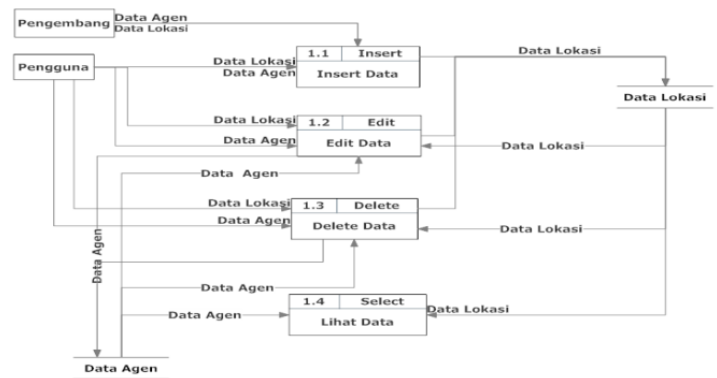
Gambar 4 Diagram Alir Data Level Nol

Kegiatan yang dilakukan pada proses 1 adalah proses CRUD. Proses ini merupakan proses dimana pengguna dan pengembang dapat melakukan pengelolaan data agen dan data lokasi. Data – data ini akan disimpan dalam media penyimpanan data agen dan data lokasi. Proses ini dapat dilihat dengan jelas pada Gambar 5.



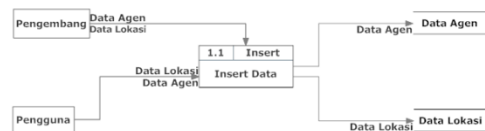
Gambar 5 Proses CRUD

Untuk menjelaskan lebih rinci dari proses ini, maka dibuatlah diagram alir data level satu. Gambar 6 berikut menjelaskan proses kelola data yang dapat dilakukan oleh pengguna dan pengembang.



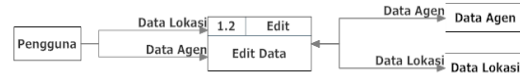
Gambar 6 Diagram Alir Data Level Satu Proses CRUD

Kegiatan yang dilakukan pada proses 1.1 adalah proses insert data. Proses ini merupakan proses dimana pengembang dan pengguna dapat memasukkan data agen dan data lokasi setiap agen ke dalam sistem, yang kemudian disimpan dalam media penyimpanan data agen dan data lokasi. Proses ini dapat dilihat lebih jelas pada Gambar 7.



Gambar 7 Proses Insert Data

Kegiatan yang dilakukan pada proses 1.2 adalah proses edit data. Proses ini merupakan proses dimana pengguna data mengubah data agen dan data lokasi setiap agen. Proses ini dapat dilihat lebih jelas pada Gambar 8.



Gambar 8 Proses Edit Data

Kegiatan yang dilakukan pada proses 1.3 merupakan proses delete data. Proses ini merupakan proses dimana pengguna dapat menghapus data agen dan data lokasi. Proses ini dapat dilihat lebih jelas pada Gambar 9.



Gambar 9 Proses Delete Data





TABEL 13 Tabel Definisi Modul Perangkat Lunak

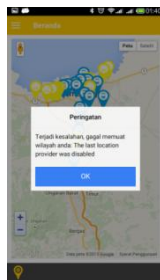
No	Nama Modul	Keterangan
1.	Map	Modul ini berisi fungsi – fungsi yang berkaitan dengan data yang akan dimuat di peta
2.	Agen	Modul ini berisi fungsi – fungsi yang berkaitan dengan kelola data agen.
3.	Riwayat	Modul ini berisi fungsi – fungsi yang berkaitan dengan riwayat pencarian pengguna.
4.	Tentang aplikasi	Modul ini berkaitan dengan informasi tentang aplikasi, dan fitur pengosongan riwayat.

Pada aplikasi pencarian lokasi terdekat agen travel dan bus dengan metode *Dijkstra* ini, pengambilan data lokasi menggunakan fitur GPS yang disediakan perangkat pengguna. Pengguna dapat mengaktifkan fitur GPS ini pada menu pengaturan atau *setting* pada perangkat bergerak, dalam hal ini perangkat bergerak dengan sistem operasi *Android*. Implementasi pengaturan GPS ini dapat dilihat pada Gambar 15.



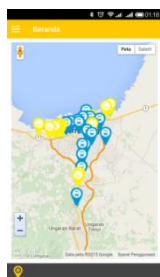
Gambar 15 Implementasi GPS

Penggunaan fitur GPS ini digunakan untuk keakuratan dalam pengambilan data lokasi perangkat pengguna. Pada Gambar 15 diatas, terlihat pengguna menyalakan fitur Satelit GPS dan fitur Lokasi Wi-Fi dan Jaringan Seluler. Apabila kedua fitur ini tidak digunakan maka aplikasi tidak dapat melakukan pencarian rute agen travel atau bus terdekat, seperti yang ditunjukkan pada Gambar 16.



Gambar 16 Peringatan Apabila Fitur Akses Lokasi Tidak Dinyalakan

Antarmuka halaman beranda merupakan halaman dimana pengguna pertama kali masuk ke dalam sistem Antarmuka beranda ini dapat dilihat pada Gambar 17.



Gambar 17 Antarmuka Beranda

Pada antarmuka beranda diatas, antarmuka menunjukkan lokasi agen yang telah dimasukkan dalam basisdata. Untuk agen bus ditunjukkan dengan penanda berwarna kuning, dan agen travel ditunjukkan dengan penanda berwarna biru. Untuk mendukung tujuan penelitian ini, aplikasi ini menyediakan fitur – fitur pencarian. Untuk memunculkan menu pencarian tersebut, pengguna dapat men – *tap* ikon penanda lokasi di pojok kiri bawah. Berikut ini adalah fitur – fitur yang terdapat pada menu beranda, yaitu:

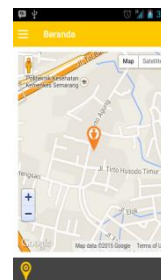
1. Lokasi Saya

Fitur lokasi saya mempunyai fungsi untuk menentukan posisi pengguna berada. Untuk menggunakan fitur ini pengguna dapat men – *tap* ikon penanda lokasi di pojok kiri bawah, dan memilih fitur lokasi saya, seperti yang ditunjukkan pada Gambar 18.



Gambar 18 Menu Pencarian Rute Terdekat

Pada fitur lokasi saya, pengguna dapat mengetahui lokasinya pada saat itu juga di peta yang terdapat pada halaman beranda Implementasi dari fitur lokasi saya dapat dilihat pada Gambar 19.



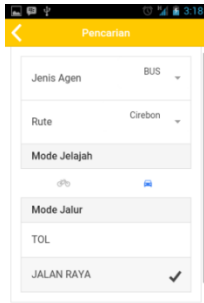
Gambar 19 Fitur Lokasi Saya Pada Antarmuka Beranda

2. Fitur Pencarian Lokasi Agen Terdekat

Selanjutnya, untuk melakukan pencarian lokasi agen terdekat, pengguna dapat memilih salah satu dari fitur pencarian sesuai kebutuhan. Berikut ini adalah pencarian lokasi agen berdasarkan parameter pencarian yang disediakan oleh aplikasi, yaitu :

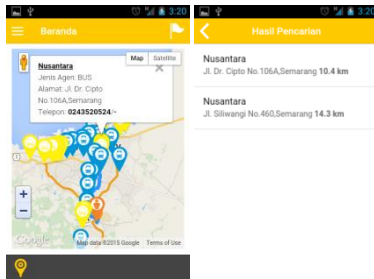
a. Pencarian Lokasi Agen Terdekat Berdasarkan Jenis Agen

Pada fitur pencarian berdasarkan jenis agen, pengguna akan dihadapkan pada form pencarian berdasarkan jenis agen, dimana pengguna diminta memilih salah satu jenis agen dan rute tujuan yang diinginkan untuk mendapatkan lokasi agen terdekat. Berikut ini adalah contoh dari fitur pencarian lokasi agen terdekat berdasarkan jenis agen yang dapat dilihat pada Gambar 20.



Gambar 20 Implementasi Fitur Pencarian Berdasarkan Jenis Agen

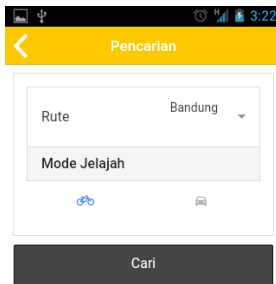
Pada Gambar 20 diatas pengguna melakukan pencarian dengan parameter jenis agen bus. Implementasi dari hasil pencarian berdasarkan parameter jenis agen ini dapat dilihat pada Gambar 21.



Gambar 21 Hasil Pencarian Lokasi Agen Terdekat Berdasarkan Jenis Agen

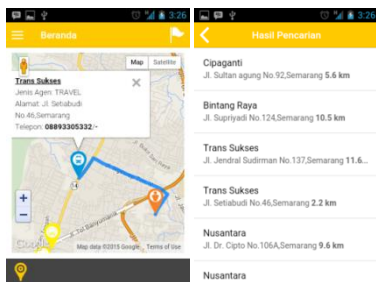
b. Pencarian Lokasi Agen Terdekat Berdasarkan Rute Agen

Untuk mendapatkan lokasi agen terdekat dari lokasi pengguna berdasarkan rute tujuan, pengguna hanya perlu memasukkan rute tujuan yang dituju. Implementasi dari fitur pencarian lokasi agen terdekat berdasarkan rute agen ini dapat dilihat pada Gambar 24.



Gambar 24 Implementasi Fitur Pencarian Berdasarkan Rute Agen

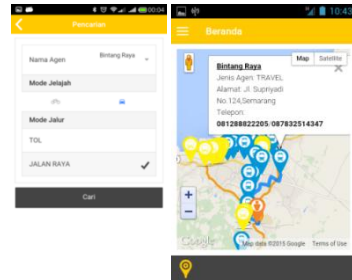
Hasil dari pencarian lokasi agen dengan parameter berdsasarkan rute perjalanan bandung ini dapat dilihat pada Gambar 25.



Gambar 25 Hasil Pencarian Lokasi Agen Terdekat Berdasarkan Rute Agen

c. Pencarian Lokasi Agen Berdasarkan Nama Agen

Pencarian lokasi agen berdasarkan nama agen dilakukan dengan memasukkan nama agen tujuan, dan aplikasi akan memuat semua lokasi agen yang serupa kemudian sistem akan mencari rute lokasi agen terdekat. Implementasi dari pencarian lokasi agen berdasarkan nama agen ini dapat dilihat pada Gambar 26.



Gambar 26 Implementasi dan Hasil Fitur Pencarian Berdasarkan Nama Agen

B. PENGUJIAN

Pada tahap ini dilakukan pengujian sistem dengan menggunakan metode *white – box*. Pengujian *white – box* adalah metode pengujian perangkat lunak yang menekankan pada fungsionalitas dari sebuah perangkat lunak dengan mengetahui bagaimana struktur di dalam perangkat lunak tersebut. Berikut tahap pengujian perangkat lunak.

1. PENGUJIAN PADA MODUL MAP

Fungsi – fungsi yang diujikan pada modul map, dapat dilihat pada Tabel 14.

TABEL 14 Tabel Pengujian Pada Modul Map

Nama Fungsi	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
Initialize()	Masuk ke dalam halaman beranda dimana <i>Google Map API</i> akan dimuat	<i>Google Map API</i> berhasil dimuat dalam aplikasi	Berhasil
centerOnMe()	Masuk ke dalam halaman beranda dimana <i>Google Map API</i> akan dimuat	Memuat wilayah lokasi pengguna berada dengan nilai <i>zoom:16</i>	Berhasil
mapCreated()	Memasukkan data – data agen berdasarkan jenisnya	Memuat penanda agen di map	Berhasil
allAgenOnMap()	Masuk pada halaman beranda	Fungsi ini dapat dipanggil untuk memunculkan penanda tiap agen bus dan travel	Berhasil
showActionSheet()	Men – tap tombol ikon fitur pencarian di sebelah kiri pojok bawah dari halaman beranda	Menampilkan <i>actionsheet</i> fitur pencarian	Berhasil
showModalCari()	Melakukan pengujian terhadap salah satu menu pencarian	Berhasil menampilkan <i>modal</i> berisi form pencarian lokasi agen	Berhasil
setJelajah() dan	Melakukan pengujian	Nilai dari masing –	Berhasil

Nama Fungsi	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
isJelajah()	terhadap mode jelajah yang terdapat pada form pencarian	masing mode jelajah berhasil dimuat	
saveEmpty()	Melakukan pengujian dengan men – tap tombol di masing – masing form pencarian	Inisiasi form pencarian	Berhasil
removeModalCari()	Men – tap tombol ikon <i>back</i> pada form pencarian	Berhasil keluar dari form pencarian	Berhasil
loadRuteByNama()	Men – tap menu pencarian lokasi agen berdasarkan nama agen terkait	Berhasil menampilkan form pencarian lokasi agen berdasarkan nama agen terkait	Berhasil
loadRuteByJenis()	Men – tap menu pencarian lokasi agen terdekat berdasarkan jenis agen	Berhasil menampilkan form pencarian lokasi agen terdekat berdasarkan jenis agen	Berhasil
loadRuteByRute()	Men – tap menu pencarian lokasi agen terdekat berdasarkan rute agen	Berhasil menampilkan form pencarian lokasi agen terdekat berdasarkan rute tujuan agen	Berhasil
hitungJarak()	Melakukan pencarian lokasi agen terdekat berdasarkan jenis agen dan rute tujuan agen	Berhasil memuat semua jarak antara lokasi pengguna ke lokasi agen – agen yang berkaitan	Berhasil
navigasiDijkstra()	Melakukan pencarian lokasi agen terdekat berdasarkan jenis agen dan rute tujuan agen	Berhasil menampilkan navigasi ke lokasi agen tujuan berdasarkan informasi yang dikirimkan dari fungsi <i>hitungjarak()</i>	Berhasil
hitungJarak2()	Melakukan pencarian lokasi agen berdasarkan nama agen terkait	Berhasil memuat semua jarak antara lokasi pengguna ke lokasi agen – agen yang berkaitan	Berhasil
insertCari()	Melakukan pencarian lokasi agen berdasarkan semua parameter yang ada	Menyimpan hasil pencarian dalam basisdata	Berhasil

## 2. PENGUJIAN PADA MODUL AGEN

Fungsi – fungsi yang diujikan pada modul agen, dapat dilihat pada Tabel 15.

TABEL 15 Tabel Pengujian Pada Modul Map

Nama Fungsi	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
toggleLeft()	Men – tap navigasi bar yang berada pada menu data agen	Menu bar berhasil dimuat	Berhasil
refreshAgen()	Memasukan data agen dan mengubah data agen yang telah tersimpan	Data agen berhasil ditambah dan data agen yang menagalami perubahan berhasil diubah	Berhasil
showTambahDialog()	Men – tap tombol tambah data agen dan ubah data agen	<i>Modal</i> Form data agen berhasil ditampilkan	Berhasil
exitDialog()	Men – tap tombol <i>back</i> pada <i>modal</i> form data agen	Keluar dari <i>modal</i> form data agen	Berhasil
saveEmpty()	Men – tap tombol simpan dan ubah yang ada pada <i>modal</i> form data agen	Menginisiasi form tambah data	Berhasil
getAlamat()	Melakukan penambahan dan pengubahan data agen	Konversi alamat menjadi koordinat <i>latitude</i> dan <i>longitude</i>	Berhasil
doInput()	Melakukan penambahan data agen	Menyimpan penambahan data agen	Berhasil
doEdit()	Melakukan pengubahan data agen	Menyimpan pengubahan data agen	Berhasil
removeData()	Men – tap tombol hapus data agen	Menghapus data agen	Berhasil
editData()	Men – tap tombol edit data agen	Memuat data agen terkait ke dalam form	Berhasil

## 3. PENGUJIAN PADA MODUL RIWAYAT

Fungsi – fungsi yang diujikan pada modul riwayat, dapat dilihat pada Tabel 16.

TABEL 16 Tabel Pengujian Pada Modul Map

Nama Fungsi	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
mapCreated()	Men – tap tombol lihat hasil pada data pencarian tertentu	Memuat <i>Google Map</i> API di <i>modal</i> hasil pencarian	Berhasil
showMap()	Men – tap tombol lihat hasil pada data pencarian tertentu	Menampilkan <i>modal</i> hasil pencarian dan memuat hasil pencarian	Berhasil
exitMap()	Men – tap tombol <i>back</i> pada <i>modal</i> hasil pencarian	Keluar dari <i>modal</i> hasil pencarian	Berhasil
hitungJarak()	Men – tap tombol lihat hasil pada data pencarian	Memuat semua jarak antara lokasi pengguna ke lokasi agen –	Berhasil



Nama Fungsi	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
	tertentu	agen yang berkaitan	
navigasiDijkstra()	Men – tap tombol lihat hasil pada data pencarian tertentu	Berhasil menampilkan navigasi ke lokasi agen tujuan	Berhasil
hitungJarak2()	Melakukan pencarian lokasi agen berdasarkan nama agen terkait	Berhasil memuat semua jarak antara lokasi pengguna ke lokasi agen – agen yang berkaitan	Berhasil
refreshRiwayat()	Melakukan pencarian lokasi agen dengan berbagai parameter pencarian	Memuat data baru hasil pencarian yang dilakukan pengguna	Berhasil
hapusRiwayat()	Men – tap tombol hapus pada data pencarian tertentu	Menghapus data terkait dari riwayat pencarian	Berhasil

#### 4. PENGUJIAN PADA MODUL TENTANG APLIKASI

Fungsi – fungsi yang diujikan pada modul tentang aplikasi, dapat dilihat pada Tabel 17.

TABEL 17 Tabel Pengujian Pada Modul Map

Nama Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil Pengujian
pushSettingChange()	Menggeser <i>toggle</i> on/off di menu pengaturan	Menghapus semua data pencarian	Berhasil

#### C. PENGOPERASIAN DAN PEMELIHARAAN

Pada tahap pengoperasian, aplikasi yang telah siap digunakan akan disitrusikan melalui *Google Play Store*, yang merupakan layanan distribusi konten untuk pengguna perangkat ber – *platform* Android. Serta untuk mengantisipasi perubahan yang disesuaikan dengan permintaan kebutuhan pengguna, baik berkaitan dengan fungsionalitas perangkat lunak ataupun perangkat keras, maka dapat dilakukan pemeliharaan oleh pengembang terhadap aplikasi SIG ini.

### V. KESIMPULAN

#### A. KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian yang telah dilakukan tentang Aplikasi Pencarian Rute Agen Travel dan Bus Terdekat dengan Metode Dijkstra di Wilayah Kota Semarang ini adalah sebagai berikut:

1. Dengan aplikasi pencarian rute agen travel dan bus ini, pengguna dapat dengan mudah menemukan agen berdasarkan parameter yang telah disediakan
2. Dengan menggunakan *Google Maps* API maka aplikasi dapat memberikan layanan lokasi dengan baik
3. Aplikasi yang dikembangkan dengan menggunakan media penyimpanan lokal *Sqlite*, memudahkan pengguna dalam mengelola data – data agen.
4. Penggunaan *Ionic Framework* dan sistem operasi Android dengan minimum versi 4.0, membuat

antarmuka aplikasi menjadi lebih menarik dan mudah digunakan oleh pengguna.

5. Metode pencarian yang digunakan, dalam hal ini *Dijkstra* telah menentukan rute agen terdekat dari lokasi perangkat pengguna dengan membandingkan jarak antara perangkat pengguna ke *node – node* terkait.

#### B. SARAN

Saran yang dapat diambil dari Tugas Akhir ini adalah sebagai berikut:

1. Penambahan fitur pada aplikasi sangat disarankan, semisal penambahan fitur pemesanan tiket langsung bus dan travel, jam keberangkatan armada yang telah agen sediakan, dan tentunya tidak hanya di wilayah semarang saja .
2. Oleh karena digunakannya *ionic framework* , diharapkan aplikasi pencarian rute agen travel dan bus terdekat ini dapat berjalan di *platform* lain, seperti iOS, windows phone, dan Blackberry OS.

### DAFTAR PUSTAKA

- [1] Riyanto, 2010, *Sistem Informasi Geografis Berbasis Mobile*, Yogyakarta : GAVA MEDIA
- [2] Rachman, S. Nofan Maulana.2012. *Sistem Informasi Geografi Pariwisata Kota Yogyakarta Berbasis Android 2.2*.Naskah Publikasi Sekolah Tinggi Ilmu Manajemen Informatika dan Komputer
- [3] Prahasta, Eddy. 2009. *Sistem Informasi Geografis : Konsep-konsep Dasar (Perspektif Geodesi & Geomatika)*. Penerbit Informatika, Bandung
- [4] Siswanto,2011,*Algoritma dan Struktur Data Non Linear dengan Java*, Yogyakarta : GRAHA ILMU
- [5] *Ionic Framework*. Online at <http://ionicframework.com/docs/guide/preface.html>, diakses pada 17 Desember 2014, 15.00 WIB.
- [6] *Google Maps JavaScript Api V3*. Online at <http://developers.google.com/maps/documentation/javascript/> diakses pada Tanggal 6 Maret 2014, 03.25 WIB.
- [7] Kusriani, 2007, *Strategi Perancangan dan Pengelolaan Basis Data*, Yogyakarta: Andi Offset
- [8] Connolly, T., Begg, C. 2010. *Database Systems: a practical approach to design, implementation, and management. 5th Edition*. America: Pearson Education.
- [9] *SQLite*. Online at <http://www.sqlite.org/about.html> , diakses pada 6 maret 2014, 04.00 WIB.
- [10] Pressman, Roger S.2001. *Software Engineering : A Practitioner's Approach, Fifth Edition*. Singapore:The McGraw-Hill Companies, Inc.
- [11] Majalah Transportasi. Online at <http://majalahtransportasi.blogspot.com>, diakses pada 20 Februari 2015, 20.00 WIB.