

**APLIKASI REKOMENDASI JALUR BUS TRANS JOGJA  
MENGUNAKAN GOOGLE MAPS API DAN  
GOOGLE PLACES API**



Disusun Oleh:

N a m a : Fawwaz Kurniawan

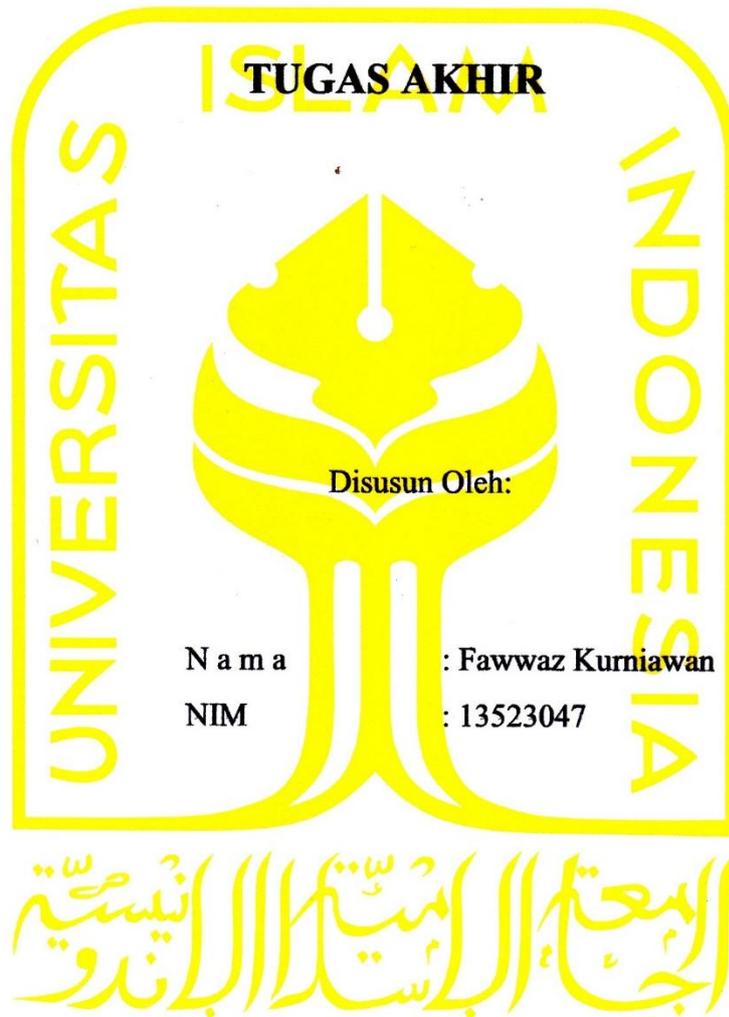
NIM : 13523047

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA**

**2018**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**APLIKASI REKOMENDASI JALUR BUS TRANS JOGJA  
MENGUNAKAN GOOGLE MAPS API DAN  
GOOGLE PLACES API**



Yogyakarta, 8 Januari 2018

Pembimbing,

( Dr. Raden Teduh Dirgahayu, S.T., M.Sc. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

**APLIKASI REKOMENDASI JALUR BUS TRANS JOGJA  
MENGUNAKAN GOOGLE MAPS API DAN  
GOOGLE PLACES API**

**TUGAS AKHIR**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia Yogyakarta, 8 Januari 2018

Tim Penguji

Dr. Raden Teduh Dirgahayu, S.T., M.Sc.

**Anggota 1**

Almed Hamzah, S.T., M.Eng.

**Anggota 2**

Hanson Prihantoro Putro, S.T., M.T.

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia



Hendrik, S.T., M.Eng.)

## HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Fawwaz Kurniawan

NIM : 13523047

Tugas akhir dengan judul:

### **APLIKASI REKOMENDASI JALUR BUS TRANS JOGJA MENGUNAKAN GOOGLE MAPS API DAN GOOGLE PLACES API**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 8 Januari 2018



( Fawwaz Kurniawan )

## HALAMAN PERSEMBAHAN

### **ALLAH SWT**

*Alhadulillah, puji syukur kepada Allah SWT karena berkat nikmat, rahmat, serta karunia yang telah diberikan-Nya, penulis dapat menyelesaikan tugas akhir ini dengan baik.*

### **Bapak Suranto dan Ibu Rusmiyati**

*Terimakasih tak terhingga penulis ucapkan kepada kedua orang tua penulis, Bapak Suranto juga Ibu Rusmiyati yang selalu tanpa henti memberikan segala dukungan baik dalam bentuk kasih sayang, do'a, motivasi, moril, materil, maupun dalam bentuk lainnya. Melalui capaian ini adalah salah satu wujud terima kasih penulis kepada Bapak dan Ibu semoga mampu membuat bahagia sekaligus bangga kepada anaknya.*

### **Keluarga, Sahabat - Sahabat, S3V Squad, Rekan – Rekan, Responden, Seluruh Pembaca**

*Kepada seluruh keluarga penulis terkhusus keluarga besar Trah Hardjo Soewito, sahabat-sahabat, S3V Squad, rekan – rekan penulis, para responden yang tidak bisa penulis sebutkan satu persatu yang juga selalu memberikan dukungan dalam segala bentuk apapun, serta kepada pembaca, semoga apa yang telah penulis tulis ini mampu memberikan manfaat dalam wujud apapun bagi kita semua.*

Terima kasih

## HALAMAN MOTO

*“ Sesungguhnya sesudah kesulitan itu ada kemudahan. Maka apabila kamu telah selesai (dari suatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain. Dan hanya kepada Tuhanmulah hendaknya kamu berharap. ”*

(Q.S. Asy Syarh: 6-8)

*“ Yakinlah bahwa akan ada sesuatu yang indah dari sebuah kerja keras, terus berusaha serahkan hasil akhir pada Allah SWT. ”*

(Anonymous)

*“ Urip iku urup ”*

*“Hidup itu nyala/ menerangi”*

*“(Hidup itu hendaknya memberi manfaat kepada orang lain maupun lingkungannya.)”*

( Pepatah Jawa )

## KATA PENGANTAR

### *Assalaamu'alaikum warahmatullaahi wabarakatuh*

Alhamdulillah, Puji syukur penulis panjatkan kepada Allah SWT. atas segala nikmat, rahmat, serta karunia yang telah diberikan-Nya sehingga penulis dapat menyelesaikan laporan tugas akhir dengan baik. Shalawat juga tercurah kepada Nabi Agung Rasulullah Muhammad SAW yang telah membimbing kaum muslimin menuju jalan kebenaran dan semoga kelak kita semua akan mendapat syafaat dari beliau, Aamiin.

Tugas akhir ini dilaksanakan sebagai salah satu syarat yang harus dipenuhi untuk memperoleh gelar sarjana di Jurusan Teknik Informatika Universitas Islam Indonesia. Dalam proses pengerjaan tugas akhir ini penulis memperoleh banyak pengetahuan, ilmu, dengan tentu menemui berbagai kesulitan namun dengan bantuan berbagai pihak akhirnya tugas akhir ini dapat terselesaikan dengan baik. Maka dari itu penulis hendak mengucapkan terima kasih kepada:

1. Bapak Nandang Sutrisno, SH., LLM., M.Hum.,Ph.D. selaku Rektor Universitas Islam Indonesia.
2. Bapak Dr. Drs. Imam Djati Widodo, M.Eng.Sc. selaku dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Bapak Hendrik, S.T., M.Eng., selaku Ketua Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
4. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku dosen pembimbing yang telah meluangkan waktu serta memberikan segala bentuk bantuan dalam membimbing penulis dalam mengerjakan tugas akhir.
5. Bapak Suranto dan Ibu Rusmiyati selaku orang tua penulis yang senantiasa memberikan segala dukungan baik dalam bentuk do'a, motivasi, moril, materil, maupun dalam bentuk lainnya.
6. Segenap Keluarga Penulis, kakak penulis Agung Prasetyo Jati, adik penulis Abdul Aziz, Keluarga besar Trah Hardjo Soewito yang turut memberikan dukungan motivasi kepada penulis.
7. Rahmad Arifan Hr, Nur Muhammad, Sugeng Riyanto, Nabil Muhammad Firdaus yang telah memberikan bantuan berupa ide, saran, dan masukan kepada penulis.
8. Segenap keluarga besar teman-teman di Fakultas Teknologi Industri terutama dari Jurusan Teknik Informatika Universitas Islam Indonesia yang telah memberikan bantuan dan dukungannya.

9. Seluruh responden yang telah membantu penulis dengan memberikan penilaian, saran dan masukan mengenai tugas akhir penulis.
10. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang turut membantu penulis dalam pengerjaan tugas akhir.

Penulis menyadari bahwa tugas akhir yang dikerjakan ini masih belum sempurna, karena keterbatasan kemampuan dan pengalaman. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi kesempurnaan tugas akhir ini. Akhir kata, penulis berharap agar penelitian ini dapat bermanfaat bagi semua pihak.

***Wassalaamu 'alaikum warahmatullaahi wabarakatuh***

Yogyakarta, 8 Januari 2018



( Fawwaz Kurniawan )

## SARI

Trans Jogja adalah salah satu pelaksana transportasi umum berbasis *BRT (Bus Rapid Transit)* yang mulai dioperasikan pada bulan Maret tahun 2008 dan hadir sebagai salah satu opsi untuk mengakomodasi transportasi umum di wilayah sekitar Yogyakarta. Hingga awal Juni 2017 total terdapat 17 jalur bus serta lebih dari 200 halte khusus Trans Jogja yang melewati daerah strategis di wilayah Yogyakarta. Bagi pengguna Trans Jogja yang belum mengetahui informasi tentang Trans Jogja apalagi belum hafal dan mengenal wilayah Yogyakarta dan ditambah banyaknya jalur serta halte yang beroperasi tentu akan memunculkan permasalahan dalam menghafal dan mencari informasi seputar Trans Jogja, maka dari itu penulis ingin membantu pengguna Trans Jogja dengan membuat aplikasi panduan praktis Trans Jogja berbasis Android karena sistem operasi ini mempunyai pengguna terbanyak di dunia.

Perancangan fungsionalitas aplikasi ini terdiri dari *Use Case Diagram*, *Activity Diagram*, dan basisdata. Aplikasi rekomendasi jalur bus Trans Jogja dikembangkan untuk perangkat Android versi 4.4 (Kitkat) ke atas dengan memanfaatkan *Location Based Service (LBS)* dan menggunakan Google Maps API serta Google Places API sebagai pendukung layanan tersebut. Graf diterapkan guna memetakan sistem transportasi Trans Jogja serta dalam pencarian jalur bus salah satunya menggunakan algoritma Floyd-Warshall. Adapun dengan menggunakan algoritma Floyd-Warshall, setelah proses pencarian selesai, maka akan ditemukan jarak terpendek dari seluruh pasangan titik yang mungkin beserta jalur yang dilewati.

Aplikasi rekomendasi jalur Bus Trans Jogja ini berhasil dibangun untuk membantu pengguna mendapatkan panduan mengenai Trans Jogja. Setelah melalui berbagai ujicoba serta meminta tanggapan dari responden, hasilnya membuktikan bahwa aplikasi panduan jalur bus Trans Jogja layak dijadikan salah satu panduan penggunaan Trans Jogja.

Kata kunci: Trans Jogja, Location Based Service (LBS), Graf, Floyd-Warshall, Google Maps API, Google Places API, Android.

## GLOSARIUM

<i>Activity Diagram</i>	Diagram yang digunakan untuk memberikan gambaran mengenai alur aktivitas tertentu pada sebuah sistem
<i>Application Programming Interface</i>	Kepanjangan dari API yaitu kumpulan fungsi - fungsi, kode pemrograman yang dibuat seseorang/ kelompok/ penyedia layanan yang bisa diintegrasikan kedalam sistem aplikasi yang dikembangkan.
<i>Bus Rapid Transit (BRT)</i>	Sistem transportasi bus cepat yang memiliki keunggulan dalam sisi infrastruktur, pelayanan, dan kualitas dibandingkan dengan sistem angkutan bus biasa.
<i>Communication Network</i>	Jaringan komunikasi, bagian dari komponen LBS
<i>Data and Content Provider</i>	Penyedia data, bagian dari komponen LBS
<i>Dynamic Programming</i>	Pemrograman dinamis. salah satu jenis pemrograman dengan karakteristik solusi yang ditemukan berasal dari solusi yang ada pada setiap tahap.
<i>Edge</i>	Bagian graf yang juga disebut sisi/ garis antar dua titik
<i>Greedy</i>	Salah satu jenis pemrograman dengan karakteristik untuk mencapai solusi optimum global dapat dilakukan dengan pendekatan yakni mengambil suatu nilai yang pada saat itu adalah merupakan nilai optimum lokal
<i>Location Based Service (LBS)</i>	Layanan informasi yang memanfaatkan jaringan internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada perangkat.
<i>Mobile</i>	Identik dengan sebuah perangkat yang bersifat bergerak
<i>Node</i>	Bagian graf yang juga disebut titik
<i>Positioning Component</i>	Komponen penentu posisi/ lokasi, bagian dari LBS
<i>Service and Application Provider</i>	Penyedia layanan aplikasi, bagian dari LBS
<i>Vertex</i>	Bagian graf yang juga disebut titik
<i>Web Service</i>	Aplikasi yang dapat berkomunikasi dengan aplikasi - aplikasi yang lain melalui jaringan dengan menggunakan serangkaian aturan-aturan (protokol) yang terstandarisasi.

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR .....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR .....	vii
SARI .....	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR .....	xiv
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
1.6 Metode Penelitian .....	3
1.6.1 Metode Pengumpulan Data .....	3
1.6.2 Analisis Kebutuhan .....	4
1.6.3 Perancangan Aplikasi .....	4
1.6.4 Implementasi Rancangan .....	4
1.6.5 Pengujian Aplikasi .....	4
1.7 Sistematika Penulisan .....	5
<b>BAB II LANDASAN TEORI.....</b>	<b>6</b>
2.1 Trans Jogja.....	6
2.2 <i>Location Based Service (LBS)</i> .....	8
2.3 Graf .....	10
2.4 Algoritma Floyd-Warshall .....	11
2.5 Google Maps API dan Google Places API .....	12
2.6 Android .....	13
2.7 Tinjauan Aplikasi Sejenis .....	13
<b>BAB III METODE PENELITIAN .....</b>	<b>17</b>
3.1 Pengumpulan Data .....	17
3.1.1 Studi Pustaka .....	17
3.1.2 Observasi .....	17
3.1.3 Wawancara .....	17
3.2 Analisis Kebutuhan.....	18
3.2.1 Identifikasi Pengguna .....	18
3.2.2 Analisis Kebutuhan Proses .....	18
3.3 Perancangan Sistem .....	20
3.3.1 Perancangan Fungsionalitas .....	20
3.3.2 Perancangan Perilaku Sistem .....	21
3.3.3 Perancangan Basis Data .....	27
3.3.4 Perancangan Antarmuka.....	28
3.4 Rencana Pengujian.....	35

BAB IV PEMBAHASAN.....	38
4.1 Implementasi Graf dan Pembahasan Algoritma Floyd-Warshall .....	38
4.1.1 Implementasi Graf .....	38
4.1.2 Pembahasan Algoritma Floyd-Warshall .....	39
4.2 Implementasi Google Maps API dan Google Places API .....	49
4.2.1 Implementasi Google Maps API .....	49
4.2.2 Implementasi Google Places API.....	49
4.3 Batasan Implementasi .....	50
4.4 Implementasi Aplikasi .....	51
4.4.1 Implementasi Menu Pencarian Panduan Jalur bus .....	51
4.4.2 Implementasi pada Menu Rute Bus.....	57
4.4.3 Implementasi pada Menu Halte Bus .....	59
4.4.4 Implementasi pada Menu Informasi Lain Seputar Trans Jogja.....	59
4.4.5 Implementasi Antarmuka Lihat Lokasi Pengguna Saat Ini.....	60
4.5 Pengujian Aplikasi .....	61
4.5.1 Pengujian Fungsionalitas Aplikasi .....	61
4.5.2 Pengujian Kompatibilitas Aplikasi.....	63
4.5.3 Pengujian Aplikasi pada Pengguna ( <i>User Acceptance Test</i> ).....	65
BAB V KESIMPULAN DAN SARAN .....	68
5.1 Kesimpulan .....	68
5.2 Saran .....	68
DAFTAR PUSTAKA .....	70
LAMPIRAN.....	72

## DAFTAR TABEL

Tabel 3.1 Struktur tabel halte.....	27
Tabel 3.2 Struktur tabel bus.....	27
Tabel 3.3 Struktur tabel jenis_halte.....	27
Tabel 3.4 Struktur tabel antar_halte.....	28
Tabel 3.5 Rencana pengujian fungsionalitas.....	35
Tabel 4.1 Matriks jarak pada kondisi awal.....	41
Tabel 4.2 Matriks jalur pada kondisi awal.....	41
Tabel 4.3 Matriks jarak saat iterasi pertama selesai.....	43
Tabel 4.4 Matriks jalur saat iterasi pertama selesai.....	43
Tabel 4.5 Matriks jarak saat iterasi kedua selesai.....	43
Tabel 4.6 Matriks jalur saat iterasi kedua selesai.....	44
Tabel 4.7 Matriks jarak saat iterasi ketiga selesai.....	46
Tabel 4.8 Matriks jalur saat iterasi ketiga selesai.....	46
Tabel 4.9 Matriks jarak saat iterasi keempat selesai.....	46
Tabel 4.10 Matriks jalur saat iterasi keempat selesai.....	46
Tabel 4.11 Matriks jarak saat iterasi kelima selesai.....	47
Tabel 4.12 Matriks jalur saat iterasi kelima selesai.....	47
Tabel 4.13 Perbandingan matriks jarak sebelum dan sesudah dihitung.....	47
Tabel 4.14 Perbandingan matriks jalur sebelum dan sesudah dihitung.....	48
Tabel 4.15 Hasil pengujian fungsionalitas.....	61
Tabel 4.16 Hasil perhitungan versi Android yang digunakan pada smartphone responden....	66
Tabel 4.17 Hasil perhitungan frekuensi jawaban dan nilai tiap tiap pertanyaan.....	66

## DAFTAR GAMBAR

Gambar 2.1 Peta jalur bus Trans Jogja (Dishub DIY, 2017).....	6
Gambar 2.2 Lima komponen pada <i>location based service</i> (Steiniger, 2006).....	8
Gambar 2.3 Komponen LBS dan alur informasi .....	10
Gambar 2.4 Jenis-jenis graf .....	10
Gambar 2.5 Pseudocode algoritma Floyd-Warshall .....	12
Gambar 2.6 Beberapa tampilan aplikasi Trans Jogja (Google Play, 2012).....	14
Gambar 2.7 Beberapa tampilan aplikasi Trans Jogja (guide) (Google Play, 2016) .....	14
Gambar 2.8 Beberapa tampilan aplikasi iTransJogja (Google Play, 2013).....	15
Gambar 2.9 Beberapa tampilan aplikasi 0274 (Google Play, 2016).....	16
Gambar 3.1 Use case diagram aplikasi rekomendasi jalur bus Trans Jogja .....	20
Gambar 3.2 Activity diagram lihat halte terdekat.....	21
Gambar 3.3 Pencarian halte terdekat dari lokasi asal dan juga lokasi tujuan .....	22
Gambar 3.4 Pencarian jalur dari halte A ke halte B .....	23
Gambar 3.5 Activity diagram lihat informasi rute bus .....	24
Gambar 3.6 Activity diagram lihat informasi halte bus beserta lokasinya .....	25
Gambar 3.7 Activity diagram lihat informasi lain seputar Trans Jogja.....	25
Gambar 3.8 Activity diagram lihat lokasi pengguna saat ini.....	26
Gambar 3.9 Relasi basis data aplikasi rekomendasi bus Trans Jogja .....	28
Gambar 3.10 Rancangan antarmuka penentuan lokasi awal dan lokasi tujuan .....	29
Gambar 3.11 Rancangan antarmuka halte terdekat dengan lokasi awal dan tujuan .....	30
Gambar 3.12 Rancangan antarmuka daftar rekomendasi pilihan halte .....	30
Gambar 3.13 Rancangan antarmuka daftar rekomendasi pilihan bus.....	31
Gambar 3.14 Rancangan antarmuka detail panduan jalur bus dari halte A ke halte B.....	32
Gambar 3.15 Rancangan antarmuka daftar rute bus .....	32
Gambar 3.16 Rancangan antarmuka detail rute bus .....	33
Gambar 3.17 Rancangan antarmuka daftar halte bus .....	33
Gambar 3.18 Rancangan antarmuka informasi lain seputar trans jogja .....	34
Gambar 3.19 Rancangan antarmuka lihat lokasi pengguna saat ini .....	35
Gambar 3.20 Kuesioner yang akan digunakan untuk pengujian ke pengguna .....	37
Gambar 4.1 Contoh penerapan graf dengan matriks .....	39
Gambar 4.2 Pseudocode algoritma Floyd-Warshall .....	40
Gambar 4.3 Pseudocode algoritma Floyd-Warshall dan baris tambahan guna simpan jalur ..	40

Gambar 4.4 Graf sederhana dari graf Trans Jogja .....	41
Gambar 4.5 Pseudocode penelusuran jalur algoritma Floyd-Warshall .....	48
Gambar 4.6 Antar muka halaman utama aplikasi .....	51
Gambar 4.7 Antarmuka saat menentukan lokasi asal atau lokasi tujuan.....	52
Gambar 4.8 Antarmuka setelah lokasi asal dan tujuan ditentukan .....	53
Gambar 4.9 Antarmuka daftar halte terdekat dari lokasi asal dan lokasi tujuan .....	53
Gambar 4.10 Antarmuka ketika nama salah satu halte ditekan .....	54
Gambar 4.11 Antarmuka daftar rekomendasi pilihan halte bus .....	55
Gambar 4.12 Antarmuka detail rekomendasi pilihan bus.....	56
Gambar 4.13 Antarmuka detail perjalanan .....	57
Gambar 4.14 Antarmuka daftar jalur bus Trans Jogja.....	57
Gambar 4.15 Antarmuka detail rute perjalanan bus jalur 1A .....	58
Gambar 4.16 Beberapa antarmuka pada menu informasi halte bus.....	59
Gambar 4.17 Beberapa antarmuka pada menu informasi lain seputar Trans Jogja.....	60
Gambar 4.18 Beberapa antarmuka yang dapat lokasi pengguna saat ini.....	60
Gambar 4.19 Kuesioner untuk pengujian kepada pengguna .....	65

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Daerah Istimewa Yogyakarta dikenal sebagai salah satu destinasi wisata. DIY juga dikenal sebagai kota pendidikan dengan banyaknya pelajar dari luar Yogyakarta untuk menuntut ilmu, mulai dari SMA hingga perguruan tinggi. Langkah dalam meningkatkan dan menanggapi predikat tersebut adalah dengan diikuti pengembangan infrastruktur pendukung salah satunya adalah bidang transportasi umum. Trans Jogja merupakan salah satu wujud pengembangan pada bidang transportasi umum oleh pemerintah daerah DIY yang telah menjangkau beberapa wilayah strategis di Kabupaten Sleman, Kabupaten Bantul, serta Kota Yogyakarta.

Trans Jogja adalah salah satu pelaksana transportasi umum berbasis *BRT (Bus Rapid Transit)* yang mulai dioperasikan pada bulan Maret, tahun 2008. Trans Jogja hadir sebagai salah satu opsi untuk mengakomodasi transportasi umum di wilayah sekitar Yogyakarta. Hingga awal Juni 2017 total terdapat 17 jalur bus Trans Jogja yaitu 1A sampai 6A, 1B sampai 6B, 7, 8, 9, 10, dan 11. Jalur - jalur tersebut melewati daerah strategis di wilayah Yogyakarta seperti terminal-terminal bus, bandara Adi Sucipto, beberapa objek wisata, beberapa lingkungan akademis (sekolah serta universitas), pasar-pasar tradisional, instansi pemerintah, hingga beberapa rumah sakit. Jumlah halte bus Trans Jogja sampai saat ini terdapat lebih dari 200 halte yang tersebar di beberapa wilayah Kabupaten Sleman, Kabupaten Bantul, serta Kota Yogyakarta. Penumpang Trans Jogja hanya dikenakan biaya termahal Rp3.500,00 untuk naik bus Trans Jogja. Dengan ongkos yang bisa dikatakan cukup terjangkau, penumpang bisa keliling wilayah Yogyakarta yang dilewati oleh jalur bus Trans Jogja, termasuk jika penumpang hendak berpindah dari sebuah jalur ke jalur yang lainnya di sebuah halte sesuai peraturan yang berlaku.

Bagi pengguna Trans Jogja yang belum mengetahui informasi tentang Trans Jogja, apalagi belum hafal dan belum mengenal wilayah Yogyakarta, tentu akan memunculkan masalah tersendiri. Banyaknya jumlah halte dan jalur yang tersedia ditambah dengan adanya tambahan jalur baru yang beroperasi belum lama ini, tentu membuat pengguna Trans Jogja baik itu calon penumpang yang baru akan menggunakan trans Jogja, maupun bagi penumpang yang sudah sering menggunakan Trans Jogja, tentu akan kesulitan menghafal jalur serta halte.

Seiring perkembangan zaman penggunaan layanan yang memanfaatkan informasi lokasi semakin banyak digunakan. Google Maps merupakan salah satu implementasi dari layanan yang berbasis lokasi. Google Maps merupakan aplikasi yang paling banyak digunakan mengungguli beberapa aplikasi sejenisnya seperti Apple Maps, Mapquest, Yahoo maps, Waze, dan Bing Maps (Search Engine Land, 2016). Google Maps API dan Google Places API merupakan produk dari Google yang dikembangkan dengan konsep LBS. Penggunaan API tersebut dapat dengan mudah kita jumpai pada beberapa aplikasi populer yang ada saat ini seperti aplikasi transportasi online berbasis lokasi seperti GO-JEK.

Berdasarkan masalah yang telah dikemukakan, penulis ingin membantu pengguna Trans Jogja dengan membuat panduan praktis Trans Jogja berupa sebuah aplikasi berbasis Android dan memanfaatkan Google Maps API dan Google Places API terkait konsep LBS. Penulis memilih mengembangkan aplikasi rekomendasi Trans Jogja berbasis Android karena banyaknya pengguna platform smartphone ini di dunia yaitu sebesar 85,0% (International Data Corporation, 2017). Panduan tersebut dapat memberikan rekomendasi – rekomendasi jalur yang bisa dipilih penumpang ketika hendak menggunakan Trans Jogja dari suatu tempat asal menuju ke tempat tujuan berdasarkan halte terdekat bus Trans Jogja dari lokasi yang ditentukan. Selain itu panduan Trans Jogja ini juga memuat informasi seputar Trans Jogja seperti jalur-jalur bus yang tersedia, informasi halte-halte yang ada disertai posisinya pada GPS.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dikemukakan, penulis merumuskan masalah sebagai berikut:

Bagaimana membangun aplikasi rekomendasi jalur Trans Jogja untuk membantu pengguna Trans Jogja dengan memberikan rekomendasi – rekomendasi panduan jalur bus dari lokasi asal ke lokasi tujuan beserta informasi jalur dan halte bus?

## **1.3 Batasan Masalah**

Berikut ini merupakan batasan pekerjaan dalam mengembangkan aplikasi:

- a. Aplikasi dirancang untuk platform Android mulai dari versi 4.4 ke atas.
- b. Data meliputi jalur bus serta halte bus yang diolah adalah data seputar Trans Jogja yang diperoleh penulis berdasarkan peta Trans Jogja dari Dishub DIY hingga Juni 2017.
- c. Lokasi halte pada Google Maps tidak mutlak tepat sama jika dibandingkan dengan posisi halte sesungguhnya namun sudah cukup menggambarkan lokasi halte tersebut.

- d. Perhitungan jarak halte terdekat dengan lokasi yaitu jarak ditarik garis lurus dari 2 buah titik koordinat longitude dan latitude sebuah lokasi sehingga tidak mutlak tepat sama jika dibandingkan jarak sesungguhnya tetapi sudah cukup memberikan gambaran kondisinya.
- e. Rekomendasi panduan jalur bus yang terbentuk adalah rute terpendek berdasarkan total jarak tempuh menurut algoritma Floyd-Warshall (bisa memunculkan jalur bus dengan jumlah transit bus yang berbeda – beda mengikuti jalur yang terbentuk oleh algoritma Floyd-Warshall). Ditambah pencarian manual untuk pencarian jalur tanpa transit/ berganti jalur serta pencarian manual untuk maksimal satu kali transit/ berganti jalur bus.
- f. Jarak terpendek yang terbentuk adalah berdasarkan perhitungan jarak radius terpendek yang ditempuh sehingga bukan berdasarkan pada waktu tempuh terpendek.
- g. Jarak antar halte didapatkan memanfaatkan perhitungan jarak antar dua titik dengan memanfaatkan fitur perhitungan jarak Google Maps

#### **1.4 Tujuan Penelitian**

Tujuan penelitian yang dilakukan penulis adalah membuat panduan jalur bus Trans Jogja dengan mengembangkan aplikasi *mobile* untuk membantu penumpang Trans Jogja. Panduan tersebut dapat digunakan untuk membantu pengguna ketika hendak menggunakan Trans Jogja. Panduan tersebut memberikan rekomendasi jalur bus yang bisa dipilih dan informasi – informasi lain seputar Trans Jogja dan dikembangkan untuk aplikasi berbasis Android dibantu dengan Google Maps API dan Google Places API .

#### **1.5 Manfaat Penelitian**

Adapun manfaat dari penelitian ini adalah membantu masyarakat umum dalam menggunakan bus Trans Jogja mulai dari mencari panduan jalur bus, mencari informasi seputar Trans Jogja, mencari informasi rute dan halte bus, hingga aturan – aturan yang berlaku dalam penggunaan Trans Jogja.

#### **1.6 Metode Penelitian**

Metode penelitian yang digunakan penulis dalam tugas akhir ini adalah sebagai berikut:

##### **1.6.1 Metode Pengumpulan Data**

Metode pengumpulan yang digunakan penulis berikut ini:

a. Studi Pustaka

Studi Pustaka digunakan untuk mengumpulkan data serta informasi – informasi yang dibutuhkan dengan bersumber dari buku, jurnal, materi perkuliahan, juga informasi lain yang dipublikasikan dan dapat diakses melalui internet.

b. Observasi

Metode observasi digunakan untuk mengetahui kondisi secara langsung serta guna memperoleh data dan informasi mengenai jalur - jalur yang dilewati oleh bus trayek tertentu dan ordinat geografis dari halte – halte yang ada sebenarnya.

c. Wawancara

Metode wawancara digunakan untuk menambah serta menggali informasi lain guna melengkapi informasi yang telah didapat sekaligus juga memperkuat dan mengonfirmasi mengenai data yang telah diperoleh tersebut.

### **1.6.2 Analisis Kebutuhan**

Pada tahap analisis ini dimaksudkan untuk mencari tau apa saja yang dibutuhkan bagi calon penumpang Trans Jogja dengan dibantu data dan informasi yang telah diperoleh sebelumnya.

### **1.6.3 Perancangan Aplikasi**

Tahap perancangan aplikasi merupakan proses yang mendefinisikan arsitektur, komponen, antar muka, dan karakteristik suatu sistem. Tahap ini juga digunakan untuk memodelkan aplikasi yang akan dibuat sesuai dengan kebutuhan-kebutuhan yang telah dibahas pada tahap analisis kebutuhan.

### **1.6.4 Implementasi Rancangan**

Tahap ini merupakan implementasi berdasarkan rancangan yang telah dibuat pada tahap sebelumnya yaitu proses pengembangan aplikasi rekomendasi jalur bus Trans Jogja sebagai wujud solusi berdasarkan permasalahan yang telah dikemukakan.

### **1.6.5 Pengujian Aplikasi**

Tahap terakhir pada pengembangan aplikasi rekomendasi ini adalah pengujian aplikasi. Tahap ini bermaksud untuk memastikan bahwa aplikasi yang telah dikembangkan mampu beroperasi dengan baik serta sudah sesuai dengan kebutuhan pengguna.

## 1.7 Sistematika Penulisan

Sistematika penulisan dalam laporan ini dibagi menjadi beberapa bab. Adapun sistematika penulisan pada penelitian ini adalah sebagai berikut:

a. Bab I Pendahuluan

Pada bab ini memuat tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

b. Bab II Landasan Teori

Pada bab ini membahas mengenai dasar – dasar teori yang melandasi pengembangan aplikasi rekomendasi jalur bus Trans Jogja dan penulisan tugas akhir.

c. Bab III Metodologi

Pada bab ini memuat tahap – tahap dalam mengembangkan aplikasi mulai dari pengumpulan data, analisis kebutuhan, perancangan sistem, serta rencana pengujian.

d. Bab IV Hasil dan Pembahasan

Pada bab ini membahas mengenai implementasi graf dan penerapan algoritma Floyd-Warshall pada graf Trans Jogja yang disederhanakan, juga memuat hasil pengembangan aplikasi yang telah selesai dikembangkan dan juga terdapat hasil dari pengujian aplikasi.

e. Bab V Kesimpulan dan Saran

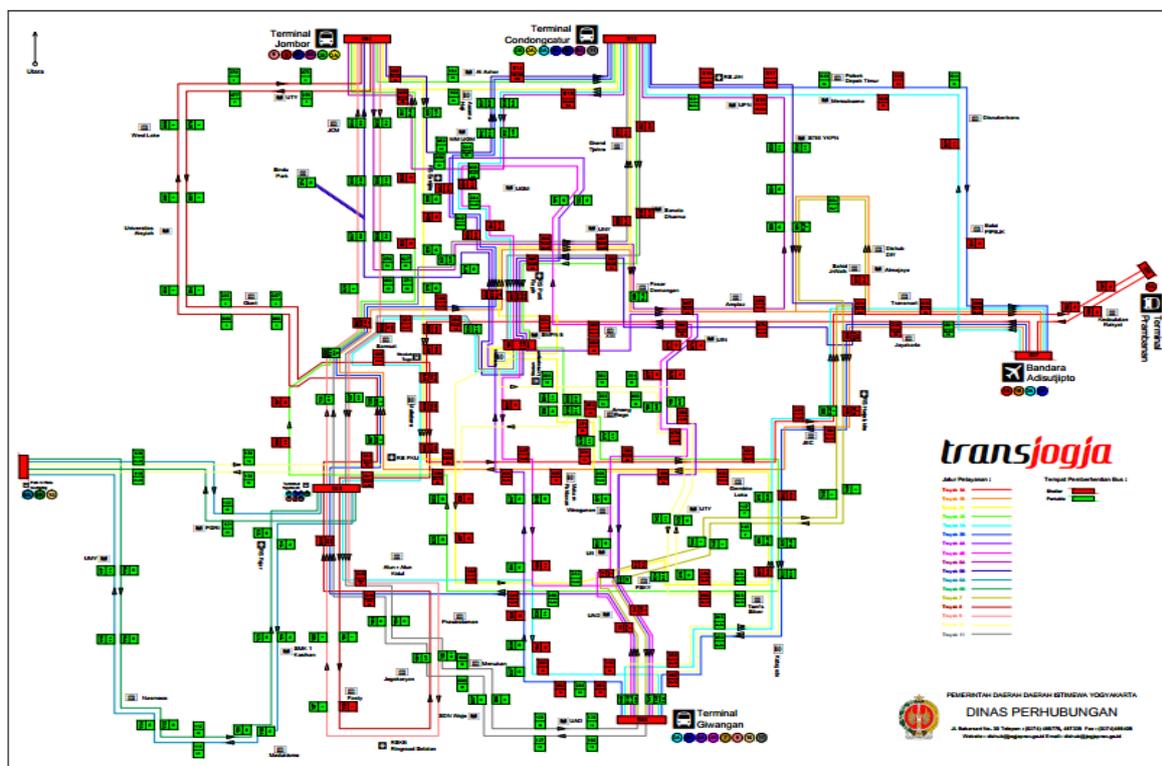
Pada bab ini berisi kesimpulan berdasarkan seluruh proses pengembangan aplikasi serta masukan berupa saran yang bisa menjadi pertimbangan bagi pengembangan kedepannya.

## BAB II LANDASAN TEORI

### 2.1 Trans Jogja

Trans Jogja adalah salah satu pelaksana transportasi umum berbasis *BRT (Bus Rapid Transit)* yang mulai dioperasikan di Yogyakarta pada bulan Maret, tahun 2008. PT Anindya Mitra Internasional (AMI) adalah sebuah BUMD milik Pemda Yogyakarta yang telah ditunjuk oleh Dinas Perhubungan Pemda Yogyakarta untuk menjadi operator pengelola Trans Jogja mulai dari tahun 2016 lalu namun dalam pelaksanaannya hingga kini masih dibantu dengan pengelola sebelumnya yaitu PT Jogja Tugu Trans (JTT) dengan alasan sedang dalam masa transisi kepengelolaan (PT AMI, 2016).

Hingga kini beberapa wilayah di Kabupaten Sleman, Kota Yogyakarta, serta Kabupaten Bantul telah dijangkau oleh bus Trans Jogja dengan jumlah trayek jalur bus yang tersedia sebanyak tujuh belas rute trayek bus (Dishub DIY, 2017). Ketujuh belas rute yang dimaksud antara lain adalah rute jalur 1A hingga 6A, 1B hingga 6B, 7, 8, 9, 10, dan 11 seperti terlihat pada Gambar 2.1.



Gambar 2.1 Peta jalur bus Trans Jogja (Dishub DIY, 2017)

Jalur - jalur tersebut telah menjangkau berbagai lokasi – lokasi strategis di wilayah Yogyakarta mulai dari terminal – terminal bus, stasiun kereta api, bandara Adi Sucipto, pasar-pasar tradisional, lingkungan akademis (SD, SMP, SMA, dan Perguruan Tinggi), rumah sakit, instansi pemerintah, beberapa destinasi tempat wisata, serta tempat strategis lainnya. Sementara itu, halte atau shelter yang merupakan tempat pemberhentian dan pengangkutan penumpang bus Trans Jogja telah terdapat lebih dari dua ratus halte baik itu halte yang terdapat penjaga maupun tanpa penjaga dan masih akan terus bertambah seiring berjalannya waktu.

Berdasarkan informasi yang terdapat pada website resmi Dishub DIY tarif tiket perjalanan penumpang bus Trans Jogja terbagi menjadi tiga jenis. Yang pertama bagi masyarakat umum bukan pemegang kartu langganan dikenakan biaya Rp. 3500 per tiket tiap satu penumpangnya. Kedua adalah bagi masyarakat umum yang memiliki kartu langganan khusus Trans Jogja atau bisa juga memiliki kartu elektrik yang dikeluarkan oleh pihak bank contohnya e-Money dari bank Mandiri, Flazz dari bank BCA juga bank BPD DIY dikenakan biaya yang lebih murah daripada masyarakat umum non pemegang kartu langganan yaitu sebesar Rp. 2700 per tiket bagi setiap pelanggan masyarakat umum pemegang kartu tersebut. Jenis tiket yang terakhir adalah bagi pelajar pemegang kartu langganan dikenakan biaya yang jauh lebih murah yaitu Rp. 1800 per tiket tiap pelanggan.

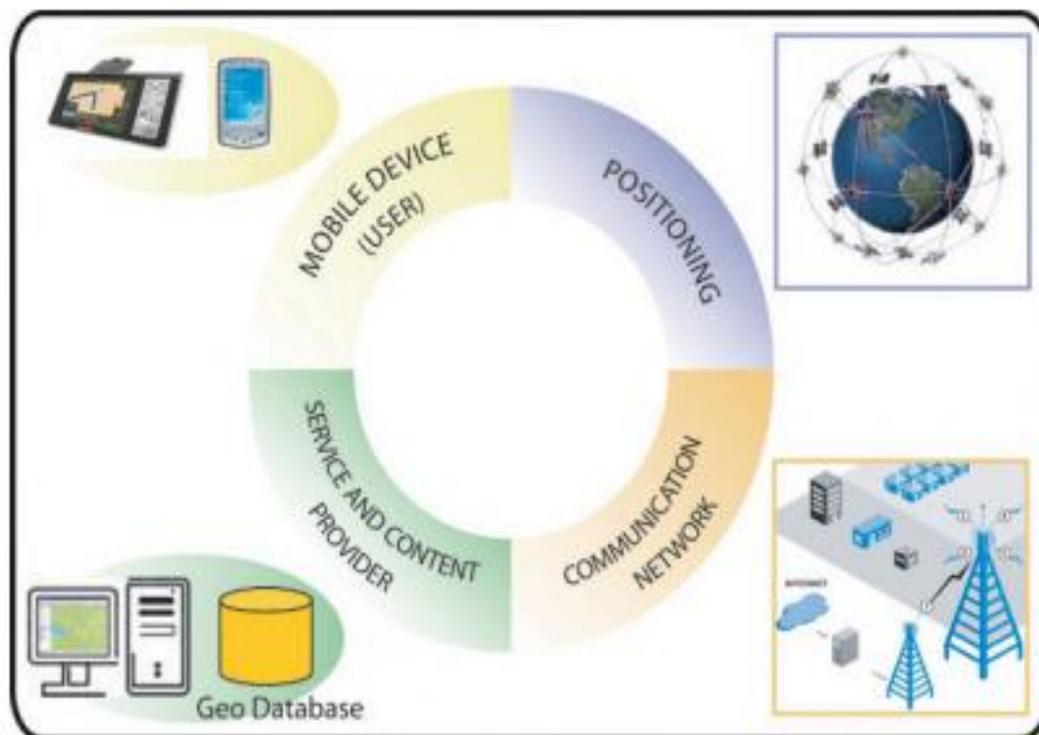
Untuk naik dan turunnya penumpang bus hanya bisa dilakukan pada halte atau shelter khusus bus Trans Jogja baik halte yang terdapat karyawan penjaganya maupun halte portable (tanpa karyawan penjaga). Penumpang bus Trans Jogja hanya cukup membayar satu kali saja untuk naik bus menuju ke tempat tujuannya termasuk jika hendak transit untuk berganti jalur bus, namun ada beberapa kondisi yang mengharuskan penumpang harus membayar tiket perjalanan lagi. Kondisi umum yang mengharuskan penumpang harus membayar ulang tiket adalah bagi penumpang yang bukan pemegang kartu langganan ketika penumpang tersebut berhenti dan hendak naik atau berganti bus jalur lain di sebuah halte/ shelter yang bersifat portable (tidak ada karyawan penjaga) maka penumpang tersebut diwajibkan untuk membayar ulang tiket saat naik dalam bus. Sementara itu, bagi penumpang pemegang kartu langganan tidak dikenakan biaya saat berganti bus jalur lain pada halte portable asalkan tidak lebih dari 60 menit sejak penumpang tersebut pertama kali membayar tiket (tap pada alat khusus) namun jika telah lebih dari waktu yang telah ditentukan maka penumpang pemegang kartu langganan akan kembali dikenakan biaya. Kondisi lain yang mengharuskan penumpang membayar ulang tiket adalah ketika ia telah turun dan keluar dari halte/ shelter yang terdapat karyawan penjaganya tanpa seizin penjaga padahal ia hendak naik kembali untuk berganti bus.

Untuk pembelian kartu pelanggan Trans Jogja yang bukan diterbitkan oleh dari bank, dapat dilayani pada halte khusus yang bertanda POS/ Card Center antara lain pada halte Terminal Jombor, halte Tentara Pelajar 2 (Jl. Tentara Pelajar, depan SMP N 14 Yogyakarta), halte Sudirman 1 (depan RS Bethesda), Halte Jl. Solo (depan Royal Ambarukmo), Halte Bandara Adi Sucipto, dan Halte Senopati 2 (depan Taman Pintar).

## 2.2 Location Based Service (LBS)

*Location Based Service (LBS)* adalah layanan informasi yang dapat diakses menggunakan perangkat *mobile* melalui jaringan internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada perangkat *mobile* (Virrantaus, et al, 2001). Adanya layanan ini maka hanya dengan memberikan informasi lokasi pengguna saja memungkinkan pengembangan berbagai macam aplikasi yang memanfaatkan informasi lokasi untuk kemudian diproses lebih lanjut menjadi hal yang lebih bermakna lagi. Contoh pengembangan aplikasi menggunakan layanan berbasis lokasi yang paling umum dijumpai adalah aplikasi ojek online seperti GO-JEK dan sejenisnya.

Dalam konsep layanan berbasis lokasi terdapat lima komponen penting sebagai pendukungnya (Steiniger, et al, 2006). Kelima komponen tersebut dapat dilihat seperti pada Gambar 2.2.



Gambar 2.2 Lima komponen pada *location based service* (Steiniger, 2006)

a. *Mobile Device*

*Mobile device* seperti *smartphone*, *mobilephone*, *PDA*, atau bahkan *PC/ Notebook* sekalipun merupakan unsur penting dari *location based service* karena semua informasi yang diminta maupun pengiriman informasi yang akan diproses bermula melalui perangkat ini oleh pengguna. Begitu pula dengan hasil informasi yang akan didapat oleh pengguna.

b. *Position Component*

Komponen selanjutnya adalah komponen penentuan posisi oleh pengguna. Komponen ini juga cukup vital karena informasi mengenai posisi merupakan informasi dasar yang akan diproses. Penentuan posisi oleh pengguna dapat dibantu menggunakan jaringan komunikasi atau dengan *Global Position System (GPS)*.

c. *Comuunication Network*

Komponen selanjutnya adalah jaringan komunikasi yang memiliki peran sebagai media dalam mengirimkan permintaan sebuah informasi atau bisa juga dibarengi dengan pengiriman informasi posisi ke sebuah gateway untuk meneruskannya ke internet kemudian *service and content provider* dan begitu pula sebaliknya.

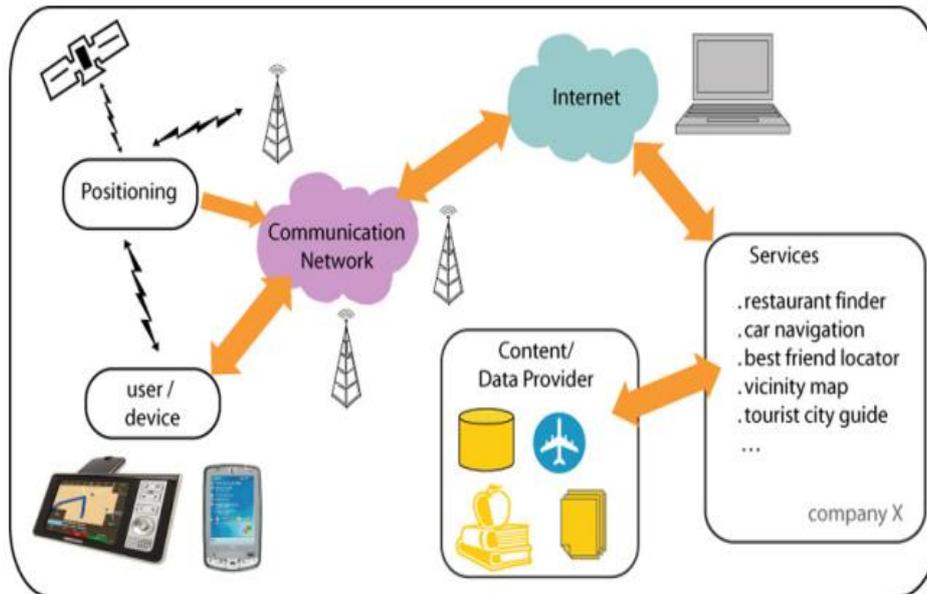
d. *Service and Application Provider*

*Service and Application Provider* bertugas untuk memproses permintaan yang dikirim oleh client sebelumnya misalnya melakukan proses untuk perhitungan antar lokasi, menunjukkan rute menuju ke sebuah lokasi, mencari informasi lebih spesifik mengenai lokasi – lokasi dan lain sebagainya.

e. *Data and Content Provider*

Komponen *data and content provider* sebenarnya adalah satu rangkaian dengan *service and application provider* karena pada dasarnya komponen ini bertugas menyediakan data yang dibutuhkan oleh *service and application provider* dalam memproses permintaan client meskipun terkadang data tersebut juga disimpan langsung oleh *service and application provider* itu sendiri.

Untuk memberikan gambaran mengenai peran dari komponen – komponen *location based service* di atas serta alur informasi yang dikirim maka dapat terlihat ilustrasinya seperti pada Gambar 2.3.

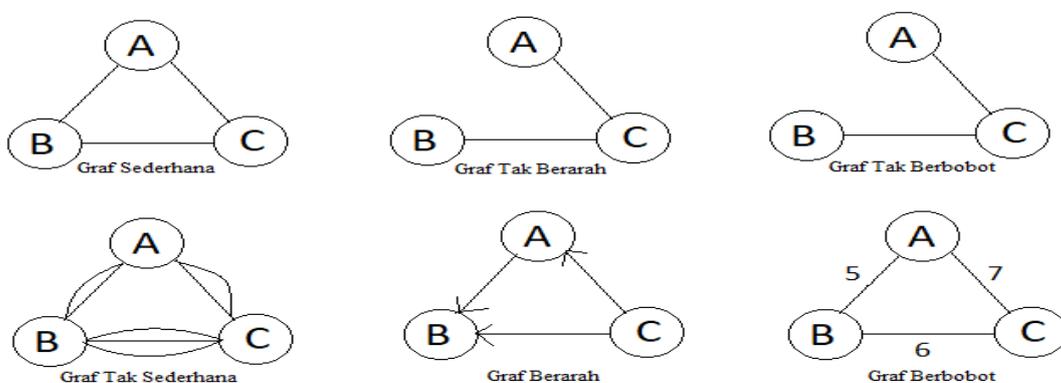


Gambar 2.3 Komponen LBS dan alur informasi

### 2.3 Graf

Sebuah graf  $G$  adalah pasangan dari  $G = (V, E)$  yang terdiri atas himpunan terbatas dari  $V$  dimana elemen-elemen dari  $V$  disebut dengan *vertex* serta himpunan  $E$  yang anggotanya merupakan sebuah pasangan *vertex* dari unsur  $V$ . sebuah elemen  $e = \{a, b\}$  disebut *edge* dengan *vertex*  $a$  dan  $b$  (Jungnickel, 2008).

Sederhananya Graf merupakan kumpulan titik-titik atau sering disebut dengan *vertex* atau *node* dan dari kumpulan titik-titik tersebut terdapat minimal sepasang titik yang berhubungan. Hubungan antar dua titik tersebut disebut dengan *edge* atau sisi. Terdapat jenis – jenis graf yang dapat dibedakan berdasarkan karakteristik - karakteristik tertentu mulai dari ada tidaknya gelang atau sisi ganda, arah yang dimiliki edge, ada tidaknya bobot pada sisi dan lain sebagainya yang dapat dilihat pada Gambar 2.4.



Gambar 2.4 Jenis-jenis graf

Penerapan graf dalam kehidupan sehari-hari dapat dengan mudah dijumpai misalnya topologi sebuah jaringan komputer, jaringan pipa air minum PAM, jaringan listrik, jalur bus Trans Jogja dan lain sebagainya. Jika konsep graf diterapkan pada jalur bus Trans Jogja maka halte – halte bus Trans Jogja yang tersebar di beberapa wilayah Yogyakarta akan bertidak sebagai titik. Sedangkan jalur bus dari halte A ke halte B misalnya adalah merupakan sisi yang memiliki bobot baik itu berupa nama jalur atau bisa juga berupa jarak antar dua halte tersebut.

Jadi dengan menggunakan graf maka dapat dilakukan penelusuran jalur serta mencari jarak terpendek dari sebuah titik asal menuju ke titik tujuan dengan menghitung total jarak yang ditempuh berdasarkan jalur yang dilalui. Dalam mencari jalur terpendek tersebut telah terdapat beberapa algoritma – algoritma yang bisa digunakan antara lain yaitu algoritma Dijkstra, algoritma Bellman-Ford, algoritma Floyd-Warshall, algoritma A\* Search yang tentunya mempunyai karakteristik masing-masing dalam mencari jalur terpendek.

#### **2.4 Algoritma Floyd-Warshall**

Algoritma Floyd-Warshall merupakan satu diantara algoritma pencarian jalur terpendek dari sebuah graf. Algoritma ini merupakan penerapan *dynamic programming* (Handaka, 2010). *Dynamic programming* memecah permasalahan dengan menguraikan solusi sehingga nantinya solusi dari permasalahan tersebut dapat dipandang sebagai serangkaian keputusan yang saling berkaitan dengan karakteristik bahwa solusi yang dihasilkan pada setiap tahap berasal dari solusi tahap sebelumnya (Munir, 2005).

Algoritma Floyd-Warshall dipilih karena keunggulan karakteristiknya dibandingkan algoritma pencarian jalur terpendek lainnya. Keunggulan algoritma Floyd-Warshall adalah proses yang lebih cepat dalam menentukan jarak terpendek dengan pencarian yang dilakukan untuk mencari jarak terpendek dari semua pasangan titik yang mungkin (Sukrisno, 2010). Jika dibandingkan algoritma lain misalnya Dijkstra, hasil penelusuran jalur terpendek dari algoritma Floyd-Warshall lebih terjamin optimum. Hal tersebut dikarenakan algoritma Dijkstra tergolong jenis *greedy* yang hanya menentukan jarak terpendek dengan mengambil nilai optimum yang bersifat lokal tidak menyeluruh sehingga bisa jadi nilai akhir yang ditemukan belum tentu merupakan nilai optimum atau jarak yang terpendek (Handaka, 2010).

Dalam pencarian jalur menggunakan algoritma Floyd-Warshall tujuan akhir yang dicari adalah menemukan jalur terpendek dari semua titik menuju semua titik atau mencari jalur terpendek semua pasangan titik yang mungkin selama masih ada jalur yang bisa menghubungkan antar titik tersebut. Sesuai dengan prinsip *dynamic programming* ini maka

algoritma Floyd-Warshall akan mencari setiap jalur terpendek dari suatu titik ke titik lain tahap demi tahap hingga sampai semua tahap selesai maka akan diketahui jalur terpendek semua pasangan titik yang mungkin.

Penerapan algoritma ini menggunakan matriks berukuran  $n \times n$ . Nilai  $n$  adalah jumlah titik yang terdapat pada graf. Nilai dari matriks baris  $i$ , kolom  $j$  pada awalnya adalah merepresentasikan besarnya jarak atau bobot yang ada dari titik  $i$  ke titik  $j$  namun setelah dilakukan perhitungan algoritma Floyd-Warshall hingga tahap akhir maka Nilai dari matriks baris  $i$ , kolom  $j$  akan berubah menjadi nilai jarak terpendek dari titik  $i$  ke titik  $j$ . Gambar 2.5 adalah pseudocode dasar dari algoritma Floyd-Warshall.

```

for k=1 to n do
  for i=1 to n do
    for j=1 to n do
      if d[i][j] > d[i][k] + d[k][j] then
        d[i][j] = d[i][k] + d[k][j]

```

Gambar 2.5 Pseudocode algoritma Floyd-Warshall

Algoritma Floyd-Warshall akan melakukan iterasi sebanyak  $n$  kali dengan bantuan iterasi  $k$ , sesuai banyaknya vertex yang ada. Pada setiap iterasi akan selalu dibandingkan untuk mencari nilai terendah dari perbandingan jarak dari titik  $i$  ke  $j$  dibandingkan dengan nilai jarak titik  $i$  ke  $k$  ditambah nilai jarak titik  $k$  ke  $j$ . Apabila nilai jarak  $i$  ke  $j$  lebih besar nilai jarak dari titik  $i$  ke  $k$  ditambah jarak titik  $k$  ke  $j$ , maka nilai jarak  $i$  ke  $j$  pada iterasi ini akan berubah menjadi nilai jumlah jarak dari  $i$  ke  $k$  ditambah  $k$  ke  $j$  begitu seterusnya hingga semua iterasi selesai dan dari hasil akhir iterasi akan didapatkan matriks yang berisi jarak terpendek dari satu titik ke titik lainnya untuk semua pasangan titik yang mungkin.

## 2.5 Google Maps API dan Google Places API

*Service and application provider* serta *data and content provider* merupakan komponen *location based service* yang berperan sebagai pemroses permintaan client. Contoh dari komponen tersebut adalah *Application Programming Interface (API)*. API berisi sekumpulan sintaks, fungsi-fungsi, dari kode pemrograman yang dibuat oleh seseorang atau sekelompok orang atau penyedia layanan yang dapat digunakan dan diintegrasikan dengan aplikasi lain.

Google Maps API dan Google Place API adalah produk dari Google yang merupakan salah satu contoh penyedia layanan termasuk pada golongan *Service and application provider* serta *data and content provider*. Google Maps API merupakan API yang secara umum lebih

berfokus pada menampilkan peta beserta pengolahan informasi contohnya penambahan ikon, pin, serta garis, juga perhitungan jarak, waktu tempuh yang terdapat pada ke suatu tempat. Hingga saat ini Google Maps API telah dapat diintegrasikan kedalam berbagai platform aplikasi seperti Android, iOS, Web, dan *Web Service* (Google Developers, 2017).

Google Places API merupakan API yang lebih berfokus tentang tempat seperti informasi tempat bisnis. Fitur yang terdapat pada API ini antara lain place picker dan place autocomplete. Place picker digunakan untuk mengambil koordinat suatu tempat serta place autocomplete yang membantu mempermudah seseorang dalam mencari nama suatu tempat dengan memberikan prediksi nama ketika mengetik beberapa huruf. Dengan memanfaatkan API ini didapatkan informasi detail dari 100 juta lokasi suatu tempat dengan basisdata yang dimiliki oleh Google Maps dan Google+ (Google Developers, 2017).

## 2.6 Android

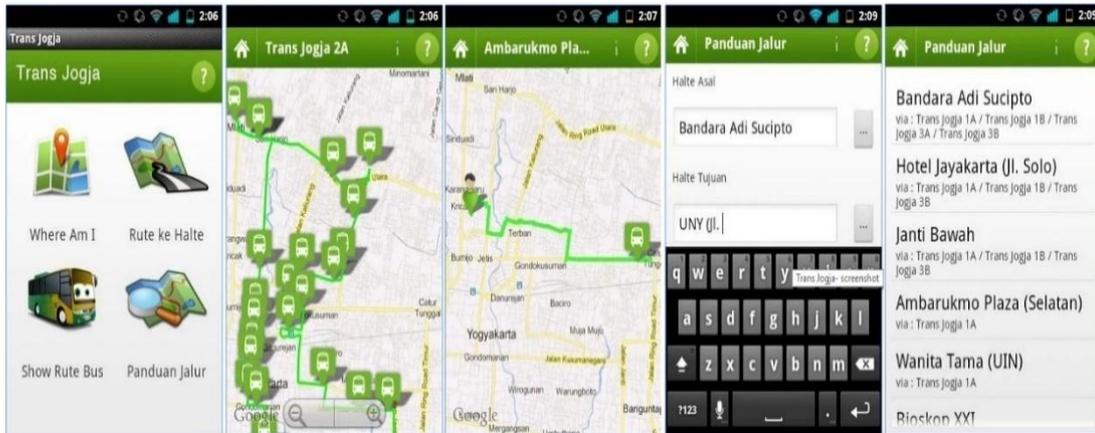
Android merupakan sistem operasi terpopuler di dunia yang berjalan pada perangkat seperti ponsel, jam tangan pintar, mobil pintar, hingga TV pintar (Android, 2017). Sistem operasi ini merupakan milik Google Inc. setelah mengakuisisi perusahaan Android Inc. pada tahun 2005. Android Inc merupakan perusahaan software yang didirikan oleh Andy Rubin yang sebelumnya juga merupakan founder dari perusahaan kecil perangkat lunak untuk perangkat mobile Danger. Perangkat pertama yang menggunakan sistem operasi Android adalah T-Mobile G1 yaitu produk dari HTC yang muncul pada tahun 2008 dan tahun-tahun setelahnya mulai dioperasikan pada bermacam-macam vendor sehingga menjadikan sistem operasi Android ini menjadi sistem operasi yang paling banyak digunakan (International Data Corporation, 2017).

Penamaan versi Android yang unik dengan *codename* berbagai nama-nama makanan ringan ini dimulai dari Android versi 1.5 dengan *codename* Cupcake yang merupakan update dari versi 1.1 sebelumnya dan versi-versi Android setelahnya diberi codenameurut pada huruf awal mulai dari Donut yaitu Android versi 1.6 hingga yang paling terbaru adalah Android Oreo yang merupakan Android versi 8.0.

## 2.7 Tinjauan Aplikasi Sejenis

Pada penelusuran yang dilakukan, penulis menemukan setidaknya ada empat aplikasi Android mengenai Trans Jogja yang telah tersedia di Google Play Store. Aplikasi – aplikasi tersebut antara lain adalah Trans Jogja, Trans Jogja (guide), iTransJogja, 0274.

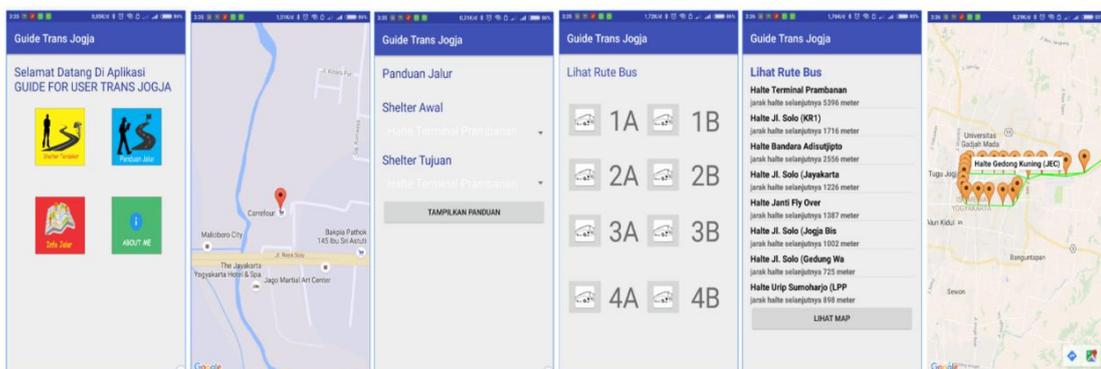
## a. Trans Jogja



Gambar 2.6 Beberapa tampilan aplikasi Trans Jogja (Google Play, 2012)

Aplikasi Trans Jogja seperti pada Gambar 2.6 ditawarkan oleh symphonyBox telah diunduh sebanyak 10000 lebih di Google Play Store dan terakhir kali diperbarui pada tanggal 6 September 2012 (Google Play, 2012). Aplikasi ini memiliki empat menu utama. Menu yang pertama adalah “Where am I” yang akan menunjukkan posisi pengguna dan akan menampilkan empat halte terdekat dari posisi tersebut. Menu selanjutnya adalah Rute ke Halte yang akan memberikan panduan menuju ke rute tujuan yang telah dipilih sebelumnya. Menu ketiga adalah Detail Rute yang akan menampilkan rute bus dari jalur 1A hingga 3A dan 1B hingga 3B. Menu terakhir adalah panduan jalur yang akan memberikan panduan jalur berdasarkan halte asal menuju halte tujuan yang harus ditentukan sendiri baik itu diketik maupun dipilih manual.

## b. Trans Jogja (guide)



Gambar 2.7 Beberapa tampilan aplikasi Trans Jogja (guide) (Google Play, 2016)

Gambar 2.7 merupakan beberapa antarmuka pada aplikasi Trans Jogja (guide). Aplikasi ini ditawarkan oleh ArStudio terakhir kali diupdate pada tanggal 10 Desember 2016

(Google Play, 2016). Aplikasi ini telah diunduh oleh pengguna Android melalui Google Playstore lebih dari 1000. Fitur pada aplikasi ini hampir mirip dengan aplikasi Trans Jogja yang telah dibahas sebelumnya antara lain memiliki fitur Shelter Terdekat yang akan menampilkan semua lokasi shelter atau halte bus Trans Jogja, lalu fitur Info Jalur yang akan menampilkan rute jalur bus dari jalur 1A hingga 4A dan 1B hingga 4B, kemudian fitur lainnya yaitu Panduan Jalur yang akan memberikan panduan jalur berdasarkan halte asal menuju halte tujuan yang harus ditentukan dengan cara memilih manual halte busnya.

c. iTransJogja

Aplikasi ini ditawarkan oleh CompFest ini memiliki tiga fitur utama seperti Nearby Shelter, Bus Route Information, Search Destination (Google Play, 2013). Fitur pertama akan menampilkan lokasi shelter atau halte bus Trans Jogja. Fitur kedua akan menampilkan informasi rute bus jalur jalur 1A hingga 3A dan 1B hingga 3B. Untuk fitur “Search destination” ini hampir mirip dengan fitur panduan bus seperti pada dua aplikasi sebelumnya namun perbedaannya adalah alur panduan yang ditampilkan adalah mulai dari pengguna mencari sebuah lokasi dari daftar lokasi populer di Yogyakarta yang juga telah ada daftarnya pada menu tersebut dan dibagi menjadi empat kategori yaitu Culinary, Tourism, Education, Public Facility. Setelah memilih lokasi tadi maka akan ditunjukkan shelter mana yang harus dituju dan jalur bus yang hendaknya diambil. Hingga penulis mengunduh aplikasi ini sebelumnya sudah ada lebih dari 1000 orang mengunduhnya sementara aplikasi ini terakhir kali diperbarui pada tanggal 27 Agustus 2013. Gambar 2.8 adalah beberapa tampilan pada aplikasi iTransJogja.

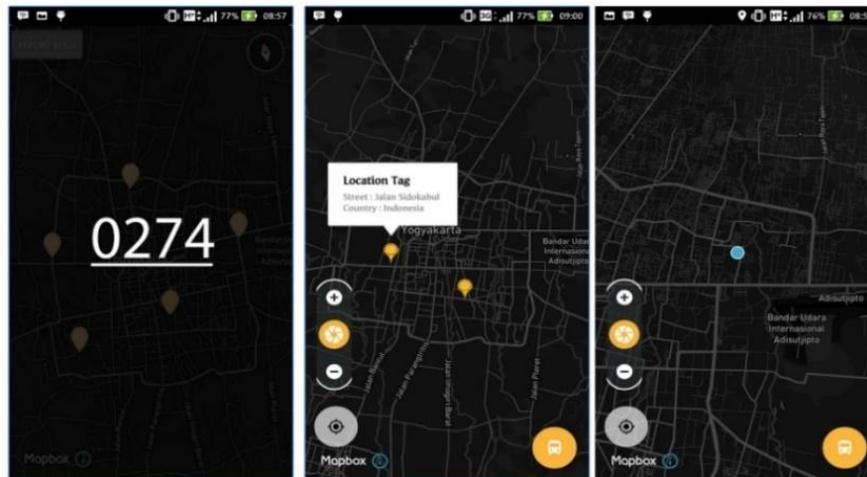


Gambar 2.8 Beberapa tampilan aplikasi iTransJogja (Google Play, 2013)

d. 0274

Aplikasi sejenis lainnya yang penulis temui adalah 0274 (Google Play, 2016). Aplikasi yang ditawarkan oleh MoCi Labs UAJY ini telah diunduh oleh 50 lebih pengguna sejak

terakhir kali diperbarui pada 12 April 2016. Berbeda pada ketiga aplikasi sebelumnya aplikasi ini hanya memiliki fitur untuk menampilkan jalur bus Trans Jogja mulai dari jalur 1A hingga 3A dan 1B hingga 3B. Beberapa tampilan aplikasi 0274 dapat dilihat pada Gambar 2.9.



Gambar 2.9 Beberapa tampilan aplikasi 0274 (Google Play, 2016)

Berikut ini adalah fitur-fitur yang secara umum terdapat pada keempat aplikasi di atas:

- 1) Memiliki fitur yang sama yaitu menampilkan informasi rute bus, menampilkan informasi halte bus, dan panduan jalur bus kecuali pada aplikasi 0274 yang hanya menampilkan jalur bus Trans Jogja saja serta tidak semuanya merupakan jalur terpendek.
- 2) Dari keempat aplikasi di atas jalur bus yang tersedia hanya terbatas hingga jalur 3B kecuali aplikasi Trans Jogja (guide) yang ditambah jalur 4A dan 4B.
- 3) Pada beberapa aplikasi di atas untuk mendapatkan panduan jalur bus harus memilih halte bus terlebih dahulu bukan dari suatu tempat.
- 4) Tidak ada fitur yang memuat informasi berkaitan dengan Trans Jogja.

Sedangkan untuk aplikasi yang akan dikembangkan memiliki keunggulan:

- 1) Pencarian jalur berdasarkan suatu tempat atau lokasi tertentu kemudian akan dicari halte terdekat dari lokasi tersebut dan terdapat jalur terpendek dari tiap – tiap rekomendasi.
- 2) Jalur bus yang tersedia sejumlah 17 jalur bus (1A – 6A, 1B – 6B, 7, 8, 9, 10, 11).
- 3) Memberikan pilihan - pilihan rekomendasi sebagai opsi pilihan jalur bus yang bisa digunakan dari lokasi asal menuju lokasi tujuan.
- 4) Terdapat fitur yang memuat informasi berkaitan dengan Trans Jogja mulai seperti sekilas tentang Trans Jogja serta aturan – aturan yang berlaku pada penggunaan Trans Jogja.

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Pengumpulan Data**

Dalam mendukung pengembangan aplikasi rekomendasi jalur bus Trans Jogja ini diperlukan dukungan data seperti data jalur bus, koordinat lokasi halte, serta informasi lainnya seputar Trans Jogja. Untuk mengumpulkan data tersebut dilakukan dengan tiga metode yaitu studi pustaka, metode observasi, dan wawancara.

##### **3.1.1 Studi Pustaka**

Untuk mendapatkan informasi – informasi yang akan digunakan dalam membantu pengembangan aplikasi dilakukan dengan metode ini. Informasi yang didapat bisa melalui berbagai sumber seperti buku, jurnal, materi perkuliahan maupun informasi lain yang dapat diakses melalui internet.

##### **3.1.2 Observasi**

Metode ini bertujuan untuk mendapatkan data seputar bus Trans Jogja. Observasi dilakukan dalam kurun waktu kurang lebih satu bulan dengan terjun langsung ke jalanan serta juga melalui observasi secara virtual melalui Google Street View pada Google Maps. Dari hasil observasi inilah diperoleh data seputar Trans Jogja seperti peta jalur bus Trans Jogja, jalur – jalur bus yang tersedia beserta rute – rute yang dilewati, letak geografis halte – halte bus Trans Jogja yang ada, jarak antar halte beserta bus yang melewatinya. Data inilah yang akan diolah pada aplikasi untuk menghasilkan informasi kepada pengguna aplikasi.

##### **3.1.3 Wawancara**

Tujuan wawancara adalah menggali informasi lain guna menambah dan memperkuat data yang telah diperoleh sebelumnya. Wawancara ini dilakukan beberapa kali pada saat kegiatan observasi di lapangan dengan narasumber beberapa pegawai yang berjaga di beberapa halte bus Trans Jogja, beberapa penumpang bus, dan salah seorang sopir bus. Dari hasil wawancara ini diperoleh informasi – informasi tambahan serta data tambahan sehingga mampu untuk melengkapi data yang telah diperoleh sebelumnya.

## 3.2 Analisis Kebutuhan

### 3.2.1 Identifikasi Pengguna

Dalam penggunaan aplikasi rekomendasi jalur bus Trans Jogja hanya terdapat satu jenis pengguna dan tidak terdapat pengguna lain seperti admin maupun yang lainnya. Pengguna yang dimaksud ialah siapa saja yang hendak menggunakan aplikasi ini untuk mencari panduan jalur bus maupun informasi lain seputar Trans Jogja baik itu seorang calon penumpang bus Trans Jogja maupun masyarakat umum

### 3.2.2 Analisis Kebutuhan Proses

Berikut proses yang terdapat pada aplikasi beserta kebutuhan masukan dan keluarannya:

- a. Proses lihat halte terdekat dari lokasi asal dan juga lokasi tujuan.

Deskripsi	Proses dalam melihat halte-halte mana saja yang terdekat dari lokasi asal dan juga lokasi tujuan yang telah ditentukan pengguna.
Input	Lokasi yang telah ditentukan pengguna baik nama lokasi ataupun titik tertentu pada peta yang nantinya berupa titik koordinat longitude dan latitude.
Output	Menampilkan nama-nama halte yang terdekat dengan lokasi asal dan lokasi tujuan yang ditentukan pengguna.

- b. Proses lihat rekomendasi jalur yang bisa ditempuh dari lokasi asal menuju lokasi tujuan.

Deskripsi	Proses dalam melihat rekomendasi jalur – jalur bus yang bisa digunakan dari lokasi asal menuju lokasi tujuannya.
Input	Lokasi asal dan juga lokasi tujuan yang ditentukan pengguna.
Output	Menampilkan rekomendasi jalur – jalur bus yang bisa digunakan dari suatu lokasi asal menuju lokasi tujuannya.

- c. Proses lihat jalur dari halte asal ke halte tujuan.

Deskripsi	Proses dalam mencari jalur bus yang bisa digunakan dari sebuah halte asal menuju halte tujuannya.
Input	Halte asal dan juga halte tujuan yang dipilih pengguna.
Output	Menampilkan informasi mengenai panduan jalur bus dari halte asal menuju halte tujuan berdasarkan halte pilihan.

## d. Proses lihat informasi tentang rute bus Trans Jogja

Deskripsi	Proses dalam melihat informasi mengenai rute yang dilewati oleh trayek bus tertentu. Trayek bus yang ada yaitu trayek 1A hingga 11.
Input	Trayek bus yang dipilih (antara jalur 1A hingga 11).
Output	Menampilkan informasi mengenai jalur bus yang dipilih berupa rute yang dilewati oleh jalur bus tertentu.

## e. Proses lihat informasi tentang halte bus Trans Jogja.

Deskripsi	Proses dalam melihat informasi mengenai halte bus yang ada. Informasi berupa nama halte dan lokasinya.
Input	Lokasi halte bus yang dipilih pengguna baik nama halte ataupun ikon halte bus tertentu pada peta yang nantinya berupa titik koordinat longitude dan latitude dari halte bus tersebut
Output	Menampilkan informasi mengenai halte yang dipilih.

## f. Proses lihat informasi lain seputar Trans Jogja.

Deskripsi	Proses dalam melihat informasi – informasi lain seputar Trans Jogja. Informasi dapat berupa sekilas info mengenai Trans Jogja, juga tata cara serta ketentuan dalam menggunakan bus Trans Jogja.
Input	Pengguna memilih menu informasi lain seputar Trans Jogja.
Output	Menampilkan informasi – informasi lain mengenai Trans Jogja. Informasi yang ditampilkan mengenai info singkat tentang Trans Jogja, tiket, tatacara serta aturan mengenai Trans Jogja.

## g. Proses lihat posisi pengguna saat ini.

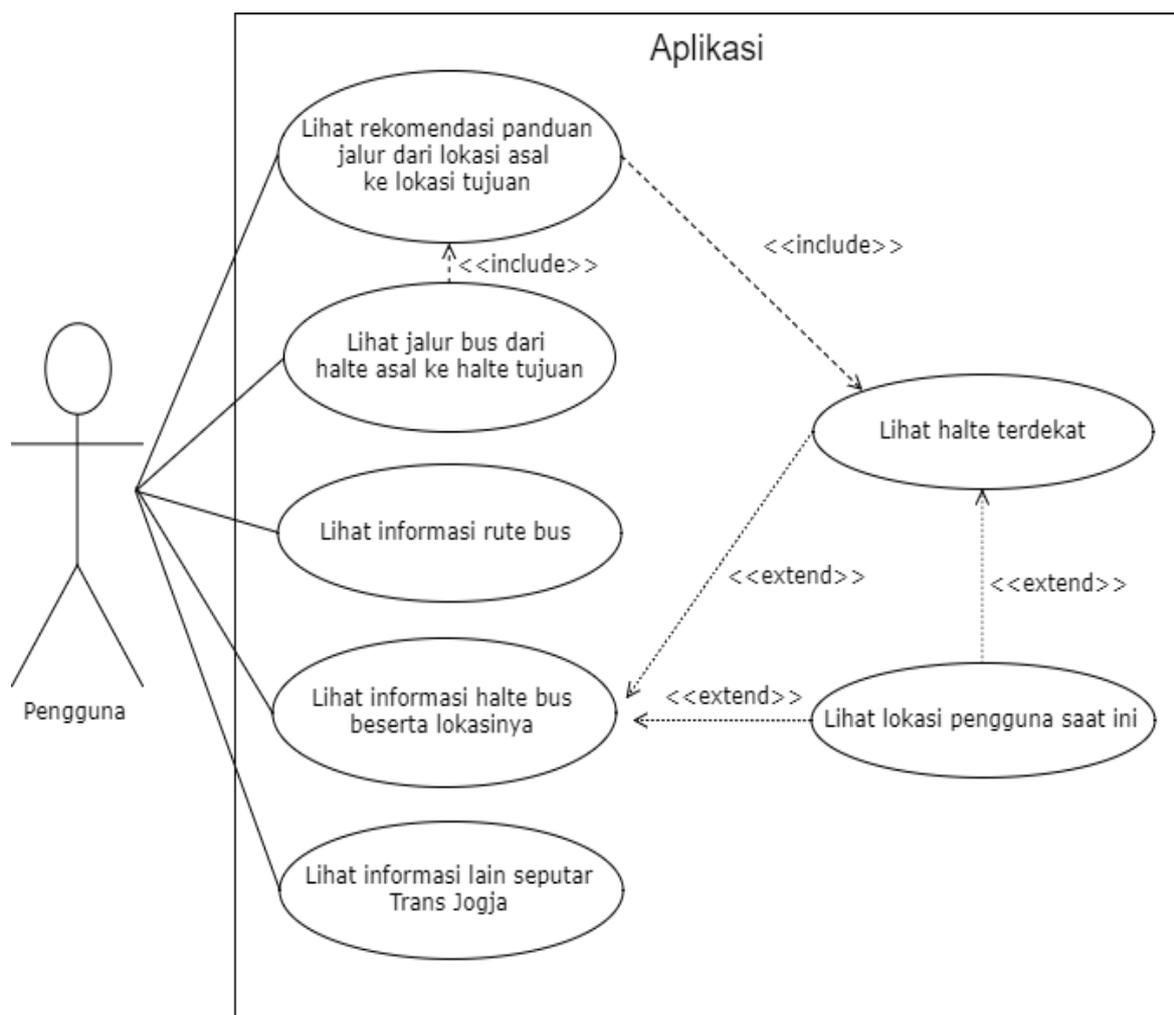
Deskripsi	Proses dalam melihat posisi pengguna saat ini yang ditampilkan dan ditunjukkan dengan tanda tertentu pada peta (Google Maps).
Output	Menampilkan letak lokasi pengguna saat ini dalam peta (Google Maps) yang didapatkan dari informasi GPS / lokasi yang ada pada smartphone pengguna.

### 3.3 Perancangan Sistem

Setelah dilakukan analisis kebutuhan maka dapat diketahui kebutuhan – kebutuhan sistem baik kebutuhan fungsionalitas, masukan, keluaran, dan proses. Berdasarkan hasil analisis kebutuhan tersebut pada tahap selanjutnya adalah tahap perancangan yang terdiri dari perancangan fungsionalitas sistem, perancangan perilaku sistem, perancangan basis data, dan perancangan antarmuka.

#### 3.3.1 Perancangan Fungsionalitas

Pada aplikasi rekomendasi jalur bus Trans Jogja ini hanya terdapat satu jenis pengguna yaitu pengguna biasa baik itu seorang calon penumpang bus Trans Jogja maupun masyarakat umum dan untuk memodelkan fungsionalitas pengguna dari aplikasi rekomendasi jalur bus Trans Jogja maka digunakan usecase diagram seperti pada Gambar 3.1.



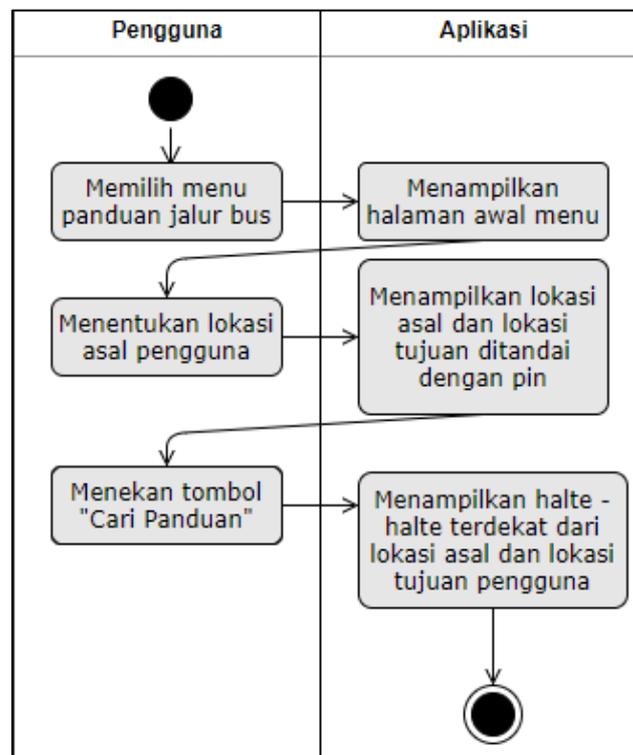
Gambar 3.1 Use case diagram aplikasi rekomendasi jalur bus Trans Jogja

### 3.3.2 Perancangan Perilaku Sistem

Setelah mengetahui kebutuhan fungsionalitas dari pengguna selanjutnya untuk memodelkan perilaku sistem sebagai alur tindakan yang berlaku maka digunakanlah *activity diagram*. Berikut ini adalah activity diagram dari tiap – tiap usecase yang ada:

a. Lihat halte terdekat

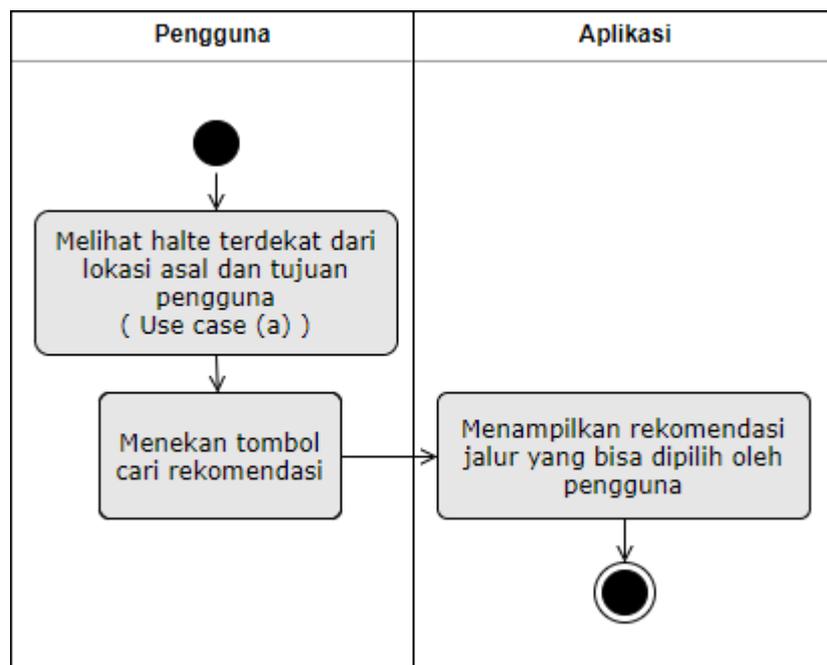
Proses lihat halte terdekat merupakan salah satu tahapan yang harus dilalui pengguna ketika ingin mencari panduan jalur bus Trans Jogja dari lokasi asal dan lokasi pengguna. Namun, sebelum melihat halte terdekat dari lokasi asal dan lokasi tujuan, pengguna haruslah terlebih dahulu menentukan lokasi asal dan juga lokasi pengguna. Tahap pertama ialah memilih menu cari panduan jalur bus. Setelah itu, aplikasi akan menampilkan halaman awal menu cari panduan yang memuat peta dan terdapat dua buah kolom yaitu lokasi asal dan lokasi tujuan. Setelah lokasi asal dan lokasi tujuan ditentukan aplikasi akan menampilkan kedua lokasi tersebut pada peta dengan penanda berupa pin. Setelah mengetahui lokasi asal dan tujuan selanjutnya pengguna menekan tombol cari panduan dan aplikasi akan berpindah halaman yang menampilkan halte – halte yang terdekat dengan lokasi asal dan tujuan tadi. Untuk memberikan gambaran mengenai aktivitas lihat halte terdekat dapat dilihat pada Gambar 3.2.



Gambar 3.2 Activity diagram lihat halte terdekat

- b. Lihat rekomendasi panduan jalur dari lokasi asal ke lokasi tujuan

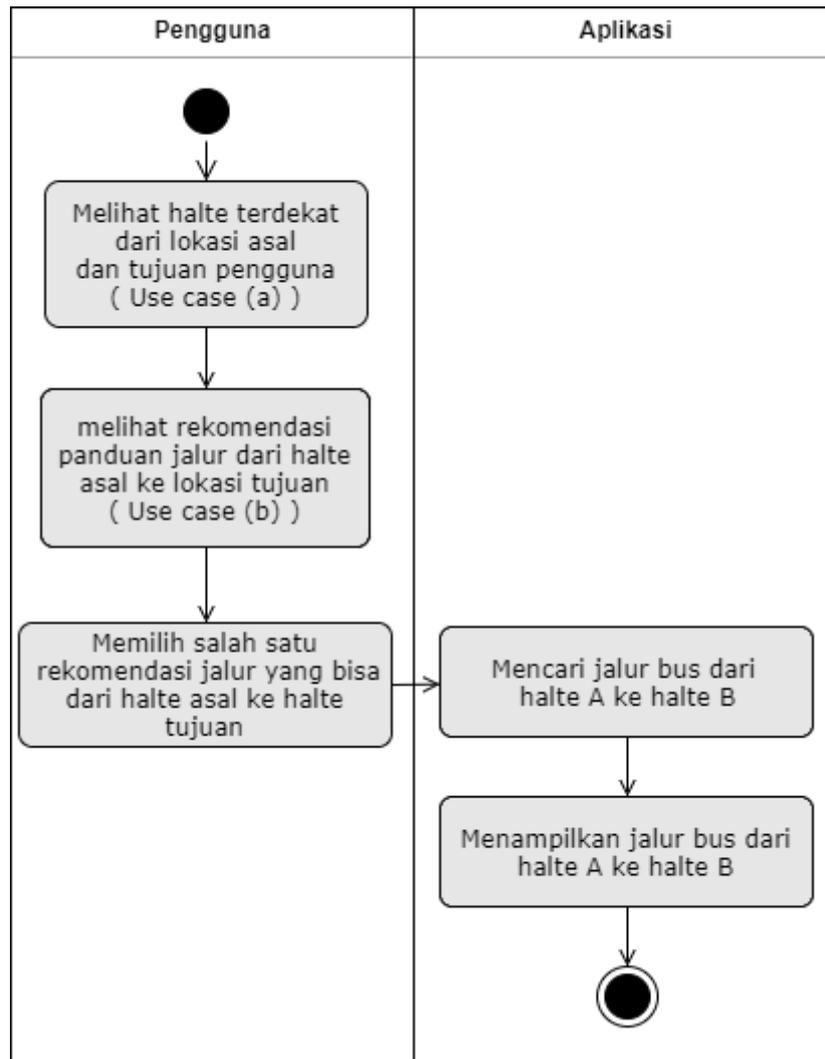
Pencarian halte terdekat dapat dilakukan setelah pengguna menentukan lokasi asal dan lokasi tujuan serta melihat halte terdekat seperti yang dijelaskan pada use case (a). Setelah lokasi asal dan tujuan sudah ditentukan maka aplikasi akan menampilkan masing - masing empat halte terdekat dari lokasi tadi. Dari keempat halte terdekat dari masing - masing lokasi tadi maka aplikasi akan menampilkan daftar rekomendasi jalur bus yang dapat dipilih pengguna. Alur aktivitas ini dapat dilihat pada Gambar 3.3.



Gambar 3.3 Pencarian halte terdekat dari lokasi asal dan juga lokasi tujuan

- c. Lihat jalur bus dari halte asal ke halte tujuan

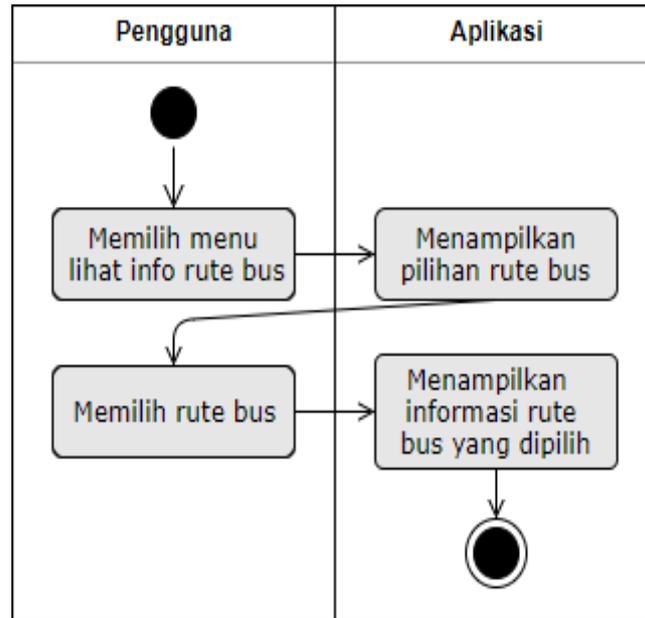
Setelah pengguna menentukan lokasi, melihat halte terdekat kemudian melihat beberapa rekomendasi jalur yang dapat dipilih. Aktivitas terakhir yang ada pada menu cari panduan jalur bus adalah pencarian jalur bus yang dilewati dari halte asal dan halte tujuan. Jalur bus yang ditampilkan merupakan jalur bus dari halte asal dan halte tujuan berdasarkan rekomendasi yang telah dipilih pada langkah sebelumnya. Setelah pengguna memilih salah satu rekomendasi tadi maka aplikasi akan mencari kemudian menampilkan jalur yang akan dilewati dari halte asal menuju halte tujuan. Aktivitas ini digambarkan pada usecase diagram pada Gambar 3.4.



Gambar 3.4 Pencarian jalur dari halte A ke halte B

d. Lihat informasi rute bus

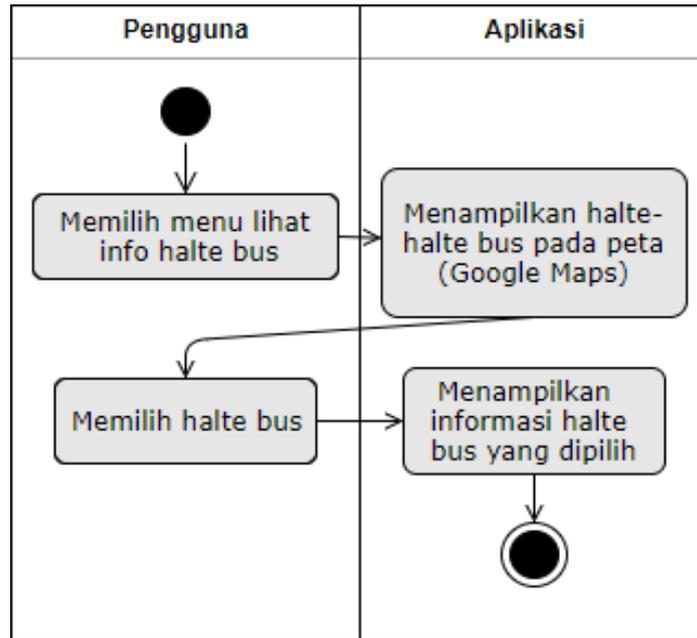
Aktivitas lain yang terdapat pada aplikasi ini adalah lihat info mengenai informasi jalur bus Trans Jogja yang tersedia mulai dari jalur 1A sampai jalur 11. Pengguna dapat melihat informasi mengenai rute bus dengan terlebih dahulu memilih menu lihat rute bus. Setelah memilih menu tersebut aplikasi akan menampilkan beberapa daftar rute bus yaitu 1A hingga 11. Selanjutnya pengguna harus memilih salah satu rute bus tersebut dan aplikasi akan menampilkan informasi detail rute bus tersebut. Aktivitas lihat informasi rute bus ini dapat dilihat pada Gambar 3.5.



Gambar 3.5 Activity diagram lihat informasi rute bus

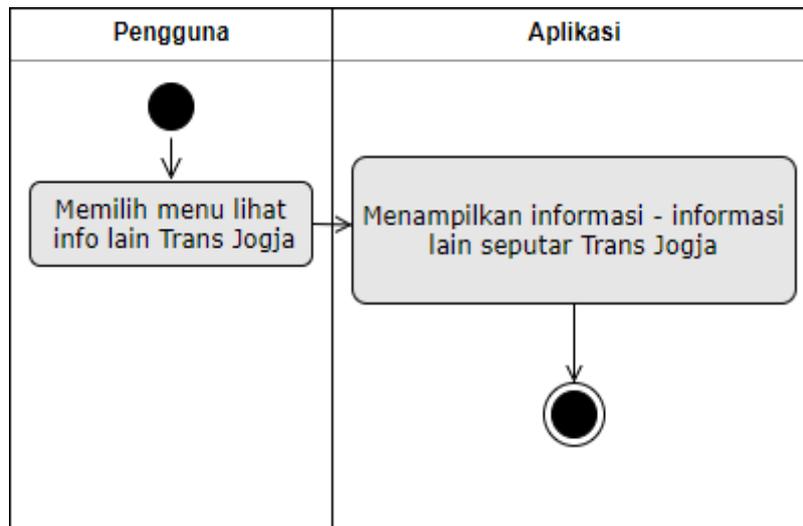
e. Lihat informasi halte bus beserta lokasinya

Selain informasi seputar bus aplikasi juga akan menyediakan informasi mengenai halte bus Trans Jogja. Alur dari aktivitas lihat informasi halte bus mirip dengan aktivitas lihat informasi rute bus. Pengguna dapat melihat informasi mengenai halte bus dengan terlebih dahulu memilih menu lihat halte bus. Setelah memilih menu tersebut aplikasi akan menampilkan halte – halte bus pada peta (Google Maps) dengan ditandai menggunakan pin/ marker. Selanjutnya pengguna harus memilih salah satu halte bus pada peta tersebut. Pemilihan dapat dengan menunjuk pin halte tersebut atau dengan mengetikkan nama dari sebuah halte. Setelah halte dipilih kemudian aplikasi akan menampilkan informasi mengenai halte bus tersebut. Informasi berupa letak halte dan bus apa saja yang berhenti pada halte tersebut. Alur aktivitas melihat informasi halte bus digambarkan pada activity diagram yang dapat dilihat pada Gambar 3.6.



Gambar 3.6 Activity diagram lihat informasi halte bus beserta lokasinya

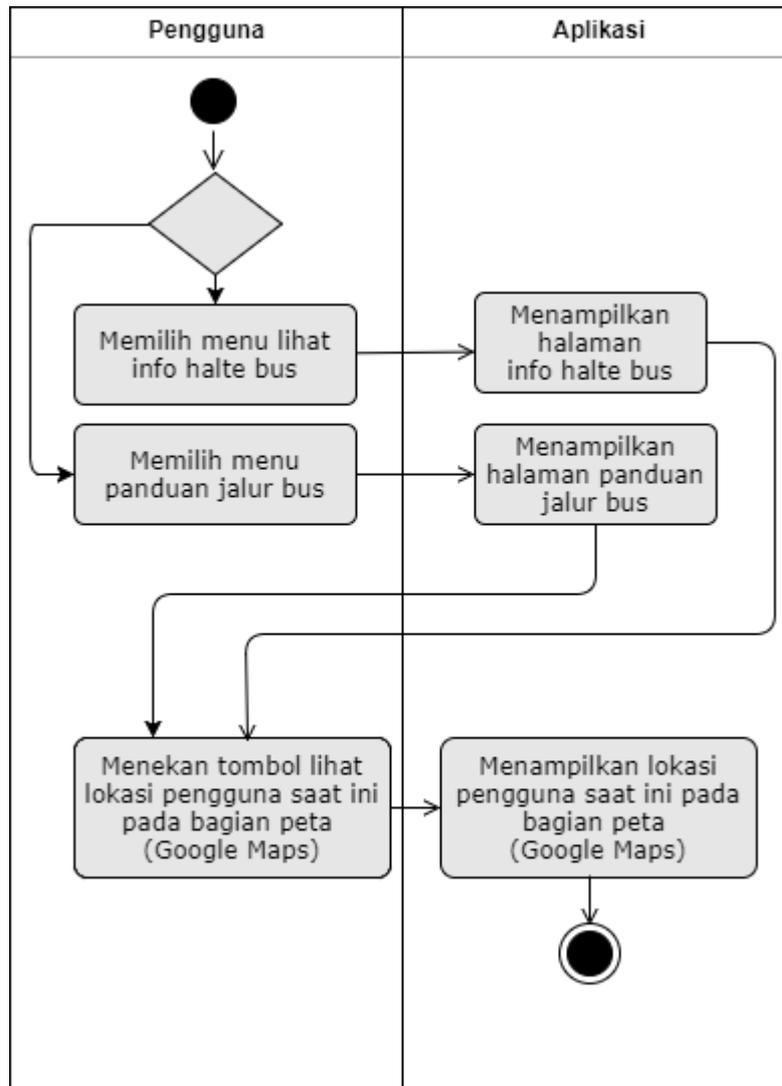
f. Lihat informasi lain seputar Trans Jogja



Gambar 3.7 Activity diagram lihat informasi lain seputar Trans Jogja

Gambar 3.7 menjelaskan aktivitas melihat informasi – informasi lain seputar Trans Jogja. Informasi tersebut antara lain yaitu sekilas mengenai Trans Jogja juga aturan dan tatacara naik bus Trans Jogja. Alur dari aktivitas lihat informasinya diawali dengan terlebih dahulu memilih menu lihat info lain seputar bus Trans Jogja oleh pengguna. Setelah memilih menu tersebut aplikasi akan menampilkan informasi – informasi lain seputar Trans Jogja seperti yang telah disebutkan sebelumnya.

- g. Lihat lokasi pengguna saat ini.



Gambar 3.8 Activity diagram lihat lokasi pengguna saat ini

Gambar 3.8 menjelaskan aktivitas melihat posisi pengguna saat ini pada peta (Google Maps). Informasi pengguna saat ini bisa dilihat pada beberapa menu aplikasi yaitu pada menu panduan jalur bus Trans Jogja dan pada menu informasi halte bus. Alur dari aktivitas ini diawali dengan terlebih dahulu memilih salah satu dari menu jalur bus Trans Jogja atau pada menu informasi halte bus. Setelah memilih salah satu menu tersebut aplikasi akan menampilkan peta (Google Maps) pada bagian tertentu. Untuk mengetahui lokasi pengguna saat ini cukup menekan tombol lihat lokasi kemudian aplikasi akan menampilkan lokasi pengguna saat ini.

### 3.3.3 Perancangan Basis Data

Basis data yang terdapat pada aplikasi ini berjumlah empat tabel yaitu tabel halte, tabel jenis halte, tabel bus, dan tabel antar\_halte.

#### a. Tabel Halte

Data seputar halte bus Trans Jogja disimpan pada tabel halte. Data tersebut meliputi nama halte, alamat halte, lokasi halte yang berupa titik koordinat (longitude dan latitude), serta jenis halte. Struktur tabel Halte dapat dilihat pada Tabel 3.1

Tabel 3.1 Struktur tabel halte

Nama Field	Tipe	Keterangan
id_halte	integer	<i>Primary key</i> , harus unik dan tidak boleh kosong
nama_halte	varchar	Menyimpan data nama halte
alamat	varchar	Menyimpan data alamat halte (nama jalan, tempat)
longitude	double	Menyimpan data titik koordinat longitude halte bus
latitude	double	Menyimpan data titik koordinat latitude halte bus
bus_tersedia	varchar	Menyimpan data bus yang tersedia pada halte bus
jenis_halte	integer	Menyimpan data jenis halte, juga sebagai <i>foreign key</i> menghubungkan dengan tabel Jenis Halte

#### b. Tabel Bus

Data seputar halte bus Trans Jogja disimpan pada tabel halte. Data tersebut meliputi nama bus beserta detail rute perjalanan yang dilewati oleh sebuah bus. Struktur tabel Bus dapat dilihat pada Tabel 3.2

Tabel 3.2 Struktur tabel bus

Nama Field	Tipe	Keterangan
id_bus	integer	<i>Primary key</i> , harus unik dan tidak boleh kosong
nama_bus	varchar	Menyimpan data nama bus
detail_rute	varchar	Menyimpan data tentang detail rute perjalanan bus

#### c. Tabel Jenis\_Halte

Tabel Jenis Halte menyimpan data keterangan jenis halte yang mempunyai relasi dengan tabel Halte. Struktur tabel Jenis Halte dapat dilihat pada Tabel 3.3

Tabel 3.3 Struktur tabel jenis\_halte

Nama Field	Tipe	Keterangan
id_jenis	integer	<i>Primary key</i> , harus unik dan tidak boleh kosong
keterangan	varchar	Menyimpan data keterangan dari jenis halte

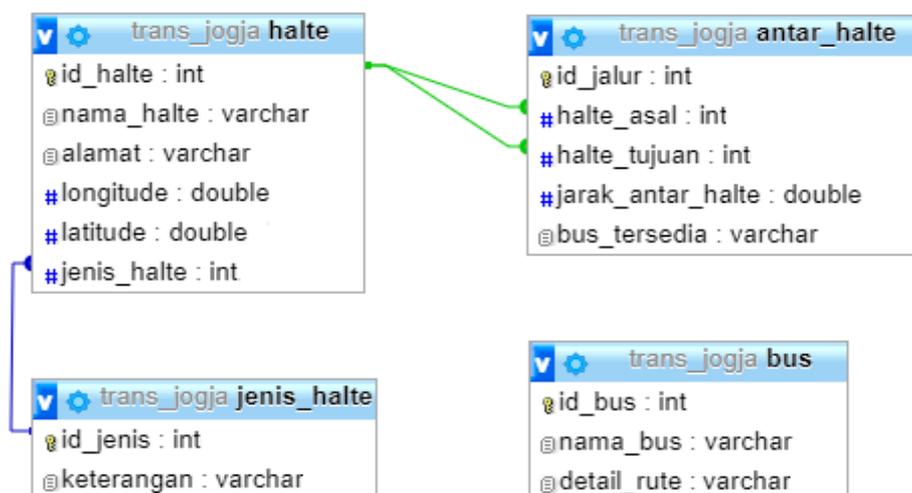
d. Tabel Antar\_Halte

Tabel Antar Halte menyimpan data hubungan dari dua buah halte. Data yang tersimpan berupa jarak antar dua halte, serta bus yang melewati antar dua halte tersebut. Struktur tabel Antar Halte dapat dilihat pada Tabel 3.4

Tabel 3.4 Struktur tabel antar\_halte

Nama Field	Tipe	Keterangan
id_antar_halte	integer	<i>Primary key</i> , harus unik dan tidak boleh kosong
halte_asal	integer	Menyimpan data halte asal
halte_tujuan	integer	Menyimpan data halte tujuan
jarak_antar_halte	double	Menyimpan data jarak antar kedua halte (jarak dari halte asal hingga halte tujuan dalam meter)
bus_tersedia	varchar	Menyimpan data bus – bus mana saja yang melewati jalan dari halte asal hingga halte tujuan)

Berikut ini adalah relasi yang terdapat antara keempat tabel di atas.



Gambar 3.9 Relasi basis data aplikasi rekomendasi bus Trans Jogja

### 3.3.4 Perancangan Antarmuka

Tahap perancangan selanjutnya adalah perancangan antarmuka. Perancangan antarmuka digunakan untuk memodelkan tampilan aplikasi yang akan dikembangkan. Perancangan ini berupa gambar sederhana yang memberikan ilustrasi mengenai tampilan yang terdiri dari ikon,

tombol, isi halaman dari tiap – tiap tampilan pada fitur – fitur aplikasi yang akan dikembangkan.

a. Perancangan Antarmuka Lihat Halte Terdekat

Sebelum lihat halte terdekat, pengguna harus lebih dahulu menentukan lokasi asal dan lokasi tujuan. Gambar 3.10 merupakan antarmuka awal ketika pengguna memilih menu cari panduan jalur bus dari lokasi asal ke lokasi tujuan sekaligus juga merupakan halaman utama dari aplikasi. Terdapat dua kolom untuk menentukan lokasi asal dan tujuan pengguna dapat berupa nama lokasi tertentu atau juga suatu titik pada peta (Google Maps). Selain itu, terdapat pula dua buah tombol. Tombol pertama adalah tombol cari yang berfungsi untuk melanjutkan ke tahap selanjutnya dalam pencarian panduan jalur yaitu melihat halte terdekat dari lokasi yang telah ditentukan tadi. Tombol kedua adalah tombol lihat lokasi saat ini yang berfungsi untuk menunjukkan lokasi atau posisi pengguna saat ini pada peta (Google Maps).



Gambar 3.10 Rancangan antarmuka penentuan lokasi awal dan lokasi tujuan

Setelah mengetahui lokasi asal dan lokasi tujuan, tahap selanjutnya dalam mendapatkan panduan jalur bus adalah melihat daftar halte – halte mana saja yang terletak dekat dengan lokasi asal dan lokasi tujuan tadi. Pada antarmuka ini terdapat dua tab yaitu asal dan tujuan. Setiap tab tadi akan menampilkan daftar halte – halte terdekat dari lokasi. Jumlah halte yang ditampilkan masing – masing ada empat halte terdekat dengan lokasi asal, dan empat halte terdekat dengan lokasi tujuan. Terdapat tombol “Cari Rekomendasi” untuk melanjutkan proses pencarian panduan jalur bus. Rancangan antarmuka ini dapat dilihat pada Gambar 3.11.

**Panduan TJ**

**Halte Terdekat Dari**

Asal	Tujuan
_____	<input type="checkbox"/>

**CARI REKOMENDASI**

Gambar 3.11 Rancangan antarmuka halte terdekat dengan lokasi awal dan tujuan

- b. Antarmuka Lihat Rekomendasi Panduan Jalur dari Lokasi Asal ke Lokasi Tujuan
- Setelah melihat daftar halte terdekat dan menekan tombol lanjut maka aplikasi akan menampilkan antarmuka rekomendasi pilihan halte – halte. Terdapat pula informasi singkat mengenai jarak terpendek yang harus ditempuh, jarak menuju halte terdekat dari lokasi asal, dan jarak halte terdekat dari lokasi tujuan. Selain itu, juga ada informasi singkat mengenai ketersediaan pilihan bus. Rekomendasi ini berasal dari pasangan – pasangan halte terdekat dari lokasi asal serta lokasi tujuan yang telah ditentukan sebelumnya. Untuk melihat lebih detail mengenai pilihan bus – bus mana saja yang bisa dipilih, pengguna perlu menekan “Lihat Detail Rekomendasi”. Gambar 3.12 adalah rancangan antar muka dari daftar rekomendasi pilihan halte.

**Panduan TJ**

**Rekomendasi**

Halte A

Halte B

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

**Lihat Detail Rekomendasi**

Gambar 3.12 Rancangan antarmuka daftar rekomendasi pilihan halte

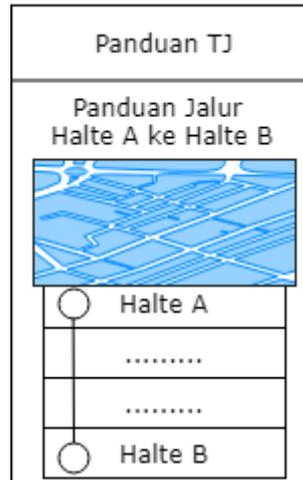
Rekomendasi jalur bus kemudian dibuat lebih khusus lagi menjadi rekomendasi pilihan bus yang dapat digunakan pengguna dari halte asal dan halte tujuan yang telah dipilih pada tahap sebelumnya. Pada daftar rekomendasi pilihan bus ini nantinya bisa jadi dikelompokkan menjadi tiga atau dua buah sub menu yaitu pilihan jalur bus yang tidak perlu transit, kemudian pilihan jalur bus dengan hanya satu kali transit serta pilihan jalur terpendek. Adanya pengelompokan tersebut tergantung pada pilihan pasangan halte asal dan halte tujuan yang dipilih. Sehingga pada beberapa pilihan bus mungkin hanya terdapat dua pengelompokan saja atau lengkap menjadi tiga pengelompokan. Namun jika tidak terdapat jalur tanpa transit dan sekali transit, maka akan langsung diarahkan ke halaman detail panduan jalur. Gambar 3.13 adalah rancangan antar muka daftar rekomendasi pilihan bus.

Panduan TJ		
Pilihan bus dari halte A ke halte B		
No Transit	Sekali Transit	Terpendek
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
Lihat Detail Jalur		

Gambar 3.13 Rancangan antarmuka daftar rekomendasi pilihan bus

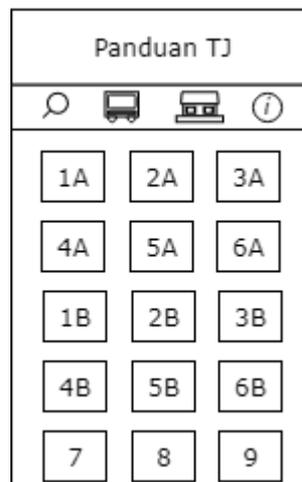
c. Antarmuka Lihat Jalur Bus dari Halte Asal ke Halte Tujuan

Setelah pengguna memilih salah satu jalur yang direkomendasikan tadi, maka antarmuka akan berpindah bagian detail panduan jalur bus. Pada antar muka ini terdapat daftar halte – halte mana saja yang akan dilewati oleh bus dari halte asal pilihan menuju halte tujuan beserta informasi bus apa saja yang harus dipilih dan juga terdapat informasi mengenai jarak tempuhnya. Pada bagian akhir akan ditampilkan ringkasan mengenai perjalanan yang dilakukan. Rancangan antarmuka untuk melihat detail panduan jalur bus dari halte asal ke halte tujuan dapat dilihat pada Gambar 3.14.



Gambar 3.14 Rancangan antarmuka detail panduan jalur bus dari halte A ke halte B

d. Perancangan Antarmuka Lihat Informasi Rute Bus

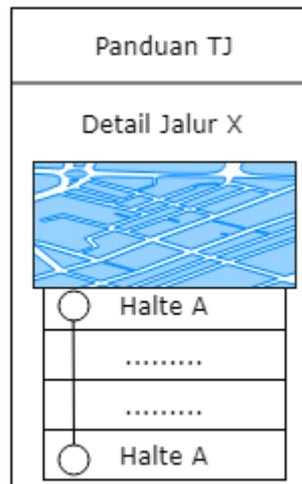


Gambar 3.15 Rancangan antarmuka daftar rute bus

Beralih ke menu yang lainnya, selanjutnya terdapat menu daftar rute bus untuk melihat informasi mengenai infor rute bus. Bagian awal dari antarmuka menu daftar rute bus tampak pada Gambar 3.15 yang terdapat beberapa ikon – ikon dari rute bus yang ada yaitu 1A hingga 11. Pengguna harus menekan salah satu ikon – ikon tadi jika hendak melihat informasi lebih lanjut mengenai rute bus yang telah dipilih tersebut.

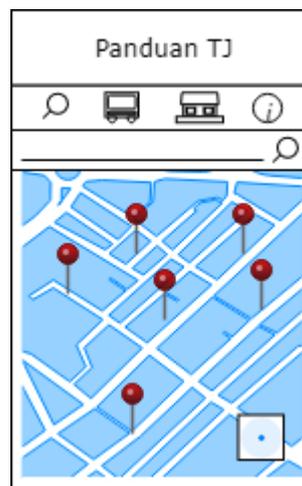
Setelah memilih salah satu rute bus antarmuka selanjutnya adalah informasi detail dari rute pilihan tersebut. Pada bagian utama antarmuka terdapat urutan daftar – daftar halte pemberhentian bus beserta jalan – jalan yang dilalui oleh bus pilihan pengguna mulai dari awal perjalanan bus hingga akhir pemberhentian bus. Seperti pada bagian lanjutan dari antarmuka utama tiap – tiap menu, pengguna hanya perlu menekan tombol kembali untuk

kembali ke antarmuka menu utama daftar rute bus. Rancangan antarmuka detail rute bus terdapat pada Gambar 3.16.



Gambar 3.16 Rancangan antarmuka detail rute bus

e. Perancangan Antarmuka Lihat Informasi Halte Bus



Gambar 3.17 Rancangan antarmuka daftar halte bus

Menu ketiga yang terdapat pada aplikasi rekomendasi jalur bus adalah lihat informasi seputar halte bus Trans Jogja. Pada halaman awal menu aplikasi akan menampilkan daftar halte yang ada pada sebuah peta seperti pada rancangan antarmuka Gambar 3.17. Daftar halte – halte tersebut akan ditampilkan pada peta (Google Maps) berupa pin – pin pada lokasi halte. Ketika salah satu pin tersebut disentuh atau pengguna menuliskan nama halte pada kotak pencarian halte, maka akan menampilkan informasi mengenai nama halte dan informasi bus – bus rute mana saja yang tersedia pada halte tersebut. Pada bagian pojok

bawah sebelah kanan terdapat tombol lokasi pengguna saat ini yang berfungsi untuk menunjukkan posisi pengguna saat ini pada peta (Google Maps).

f. Antarmuka Menu Informasi Lain Seputar Trans Jogja

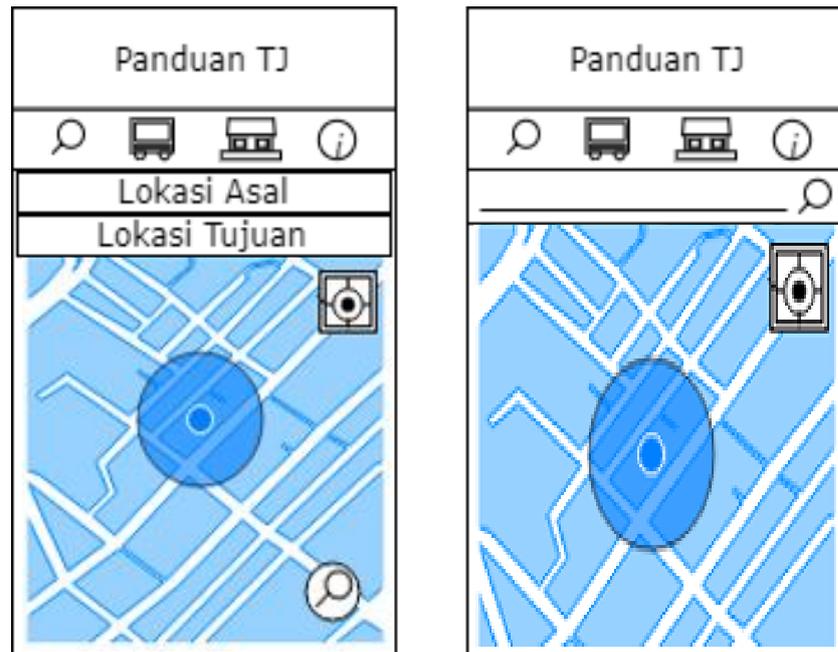
Menu terakhir yang terdapat pada aplikasi rekomendasi bus Trans Jogja adalah lihat informasi lain seputar Trans Jogja. Rancangan antarmuka untuk menu ini dapat dilihat pada Gambar 3.18. Pada antarmuka menu ini memuat informasi lain mengenai Trans Jogja berupa sekilas mengenai Trans Jogja. Informasi lain adalah mengenai tatacara dan aturan menggunakan bus Trans Jogja.



Gambar 3.18 Rancangan antarmuka informasi lain seputar trans jogja

g. Antarmuka Lihat Lokasi Pengguna Saat Ini

Proses dalam melihat lokasi pengguna saat ini dapat dilihat pada menu pencarian panduan jalur bus dan juga menu lihat informasi halte bus. Secara otomatis jika kondisi smartphone terhubung ke internet maka peta akan menampilkan lokasi pengguna saat ini, dan ketika posisi peta bergeser dan lokasi pengguna saat ini tidak terlihat maka pengguna hanya perlu menekan tombol untuk kembali melihat lokasi pengguna saat ini. Gambar 3.19 adalah rancangan antarmuka untuk lihat posisi pengguna saat ini.



Gambar 3.19 Rancangan antarmuka lihat lokasi pengguna saat ini

### 3.4 Rencana Pengujian

Guna memastikan bahwa aplikasi yang akan dikembangkan mampu beroperasi dengan baik serta sudah sesuai dengan kebutuhan pengguna maka dilakukan pengujian aplikasi. Rencana pengujian aplikasi dibagi menjadi tiga bagian yaitu pengujian fungsionalitas, pengujian kompatibilitas aplikasi, serta pengujian aplikasi pada pengguna (User Acceptance Test).

Pengujian fungsionalitas dilakukan untuk mengetahui apakah kebutuhan aplikasi telah terpenuhi dan berjalan baik atau tidak. Pengujian akan dilakukan oleh penulis melalui perangkat milik penulis dengan skenario pengujian fungsionalitas yang telah ditentukan. Pada Tabel 3.5 adalah skenario yang digunakan untuk pengujian fungsionalitas.

Tabel 3.5 Rencana pengujian fungsionalitas

No.	Fungsionalitas	Skenario	Aktivitas
1.	Lihat halte terdekat	Menentukan lokasi asal dan tujuan	Memilih menu panduan, menekan kolom pencarian, mencari kemudian menentukan lokasi asal serta lokasi tujuan. Menentukan lokasi asal/ tujuan jauh dari pusat Yogyakarta. Mengosongkan lokasi kemudian menekan tombol cari

		Lihat halte terdekat	Klik ikon cari (setelah menentukan lokasi asal dan tujuan), tekan nama halte, tekan ikon halte.
2.	Lihat Rekomendasi panduan dari lokasi asal ke lokasi tujuan	Lihat rekomendasi pilihan halte	Klik “lihat rekomendasi” (setelah lihat halte terdekat)
		Lihat rekomendasi pilihan jalur	klik “lihat detail rekomendasi bus” (setelah lihat rekomendasi pilihan halte)
3.	Lihat Jalur bus dari halte asal ke halte tujuan	Lihat detail jalur bus	Klik pilihan jalur bus (Setelah memilih jalur bus pada rekomendasi pilihan jalur)
4.	Lihat Informasi jalur bus	Lihat Informasi jalur bus	Memilih menu info jalur bus, memilih jalur bus
5.	Lihat Informasi halte bus	Lihat Informasi halte bus	Memilih menu info halte bus, mencari halte bus, menekan ikon halte bus
6.	Lihat Informasi lain seputar Trans Jogja	Lihat Informasi lain seputar Trans Jogja	Memilih menu info lain seputar Trans Jogja, memilih sub info
7.	Lihat lokasi pengguna saat ini	Lihat lokasi pengguna saat ini	Menekan tombol lihat lokasi pengguna saat ini

Pengujian berikutnya adalah pengujian kompatibilitas aplikasi untuk mengetahui seberapa luas aplikasi mampu berjalan pada perangkat dengan spesifikasi yang berbeda-beda termasuk versi android pada perangkat tersebut. Pada pengujian ini dilakukan pengujian pada perangkat android, yaitu: Asus Zenfone 4s, Xiaomi Redmi Note 2, Xiaomi Redmi Note 3, Xiaomi Redmi Note 4, serta Asus ZenPad C 7.0 (Z170CG).

Pengujian yang terakhir adalah pengujian aplikasi pada pengguna (*User Acceptance Test*) dengan cara meminta pengguna menjalankan aplikasi untuk mencoba segala menu yang ada serta memberikan penilaian terhadap aplikasi yang telah dikembangkan melalui pertanyaan skala pada kuesioner. Dalam hal menjalankan aplikasi untuk pengujian pada pengguna responden akan diberikan tawaran apakah akan melakukan pengujian aplikasi menggunakan

perangkat milik penulis atau memasang aplikasi pada perangkatnya pribadi. Gambar 3.20 berikut ini adalah pertanyaan kuesioner untuk pengujian aplikasi pada pengguna.

### **KUESIONER PENGUJIAN APLIKASI PANDUAN JALUR BUS TRANS JOGJA BERBASIS ANDROID**

Nama : \_\_\_\_\_ Pekerjaan: \_\_\_\_\_

- Pernah menggunakan bus Trans Jogaja : Pernah/ Belum (Lingkari salah satu)
  - Saya mencoba aplikasi dengan menginstallnya pada smartphone saya:
    - a. Ya\*
    - b. Tidak, Tidak, Saya mencoba langsung pada smartphone pengembang
- (lingkari salah satu)**

**\* Jika mencoba aplikasi dengan menginstallnya pada smartphone pribadi**

- Merk – Seri Smartphone (contoh: Samsung – Galaxy s8) : \_\_\_\_\_
- Versi Android pada Smartphone : \_\_\_\_\_

No	Pertanyaan	Jawaban berdasar skala				
		STS	TS	N	S	SS
1.	Aplikasi berjalan dengan baik, lancar					
2.	Aplikasi mudah untuk dipahami					
3.	Tata letak (Menu, Tombol, icon, dsb.) dan tampilan antarmuka aplikasi mudah digunakan.					
4.	Aplikasi dapat memberikan rekomendasi dan panduan jalur bus Trans Jogja yang digunakan, dari suatu lokasi asal menuju lokasi tujuan.					
5.	Aplikasi dapat memberikan informasi lain yang masih terkait dengan Trans Jogja (Informasi sekilas tentang Trans Jogja, informasi rute bus, informasi tentang halte bus) <u>dengan baik</u>					
6.	Informasi yang disajikan pada aplikasi mudah untuk saya pahami					
7.	Pengetahuan saya mengenai Trans Jogja bertambah setelah menggunakan Aplikasi ini					
8.	Aplikasi ini layak untuk dijadikan salah satu panduan praktis penggunaan bus Trans Jogja					

Gambar 3.20 Kuesioner yang akan digunakan untuk pengujian ke pengguna

## BAB IV PEMBAHASAN

### 4.1 Implementasi Graf dan Pembahasan Algoritma Floyd-Warshall

Aplikasi Rekomendasi Jalur Bus Trans Jogja yang dibangun dengan mengimplementasikan konsep graf untuk menggambarkan sistem transportasi bus Trans Jogja. Kemudian dalam langkah mencari jalur bus yang dibentuk salah satunya adalah dengan menggunakan algoritma Floyd-Warshall.

#### 4.1.1 Implementasi Graf

Konsep graf digunakan untuk menerjemahkan sistem transportasi bus Trans Jogja ke dalam bentuk yang lebih dimengerti oleh aplikasi. Sesuai dengan ciri dan karakteristiknya bahwa graf adalah sekumpulan titik dan dari beberapa titik terdapat pasangan titik yang saling dihubungkan oleh suatu garis. Gambar 2.1 menunjukkan sistem transportasi bus Trans Jogja yang nantinya akan diterjemahkan ke dalam bentuk graf.

Pada Gambar 2.1 terlihat terdapat titik atau kotak berwarna hijau dan merah yang merupakan tanda atau simbol untuk halte bus Trans Jogja dimana kotak merah untuk halte berwujud shelter dan hijau adalah halte portable kecil. Kemudian diantara dua halte terdapat garis-garis dengan arah tertentu yang menghubungkan keduanya. Garis tersebut merupakan tanda bahwa terdapat jalur bus yang melewati halte asal menuju halte tujuan dan tiap – tiap warna garis menunjukkan jalur bus yang beroperasi (misal: warna kuning untuk bus jalur 2A).

Ketika sistem transportasi bus Trans Jogja ini diterjemahkan menjadi sebuah graf Trans Jogja maka yang bertidak sebagai *vertex* atau titik adalah halte bus Trans Jogja. Sedangkan jalur bus yang melalui halte asal menuju halte tujuan atau garis yang menghubungkan kedua halte adalah *edge*. Kemudian jika dilihat dari jenis graf yang terbentuk, maka sistem transportasi bus Trans Jogja merupakan jenis graf berarah juga berbobot karena pada *edge* graf ini terdapat nilai tertentu. Bobot graf yang dimaksud yaitu nilai jarak antar halte dan juga jalur bus apa saja yang melintas antar halte tersebut.

Pada program/ aplikasi rekomendasi jalur bus Trans Jogja konsep graf diimplementasikan dengan menggunakan matriks yaitu array dua dimensi berukuran  $n \times n$  seperti ilustrasi pada Gambar 4.1.

Matriks Jalur						Matriks Bus Tersedia					
	1	2	3	....	n		1	2	3	....	n
1	....	....	....	....	....	1	....	....	....	....	....
2	....	....	1500	....	....	2	....	....	3A, 4B	....	....
....	....	....	....	....	....	....	....	....	....	....	....
n	....	....	....	....	....	n	....	....	....	....	....

Gambar 4.1 Contoh penerapan graf dengan matriks

Nilai dari  $n$  merupakan banyaknya node yang ada atau jika pada sistem transportasi bus Trans Jogja nilai  $n$  adalah jumlah halte yang ada. Nilai yang disimpan pada indeks  $\text{matriks}[\text{baris}][\text{kolom}]$  adalah bobot yang dimiliki oleh edge dari graf Trans Jogja. Seperti yang telah disebutkan sebelumnya, bobot yang dimaksud bisa berupa jarak antar halte dan juga bisa berupa daftar bus yang melewati halte A menuju Halte B sehingga terdapat dua buah matriks yaitu matriks jarak $[\dots][\dots]$  untuk menyimpan nilai jarak antar halte dan juga matriks busTersedia $[\dots][\dots]$  untuk menyimpan daftar bus tersedia. Misalkan pada matriks jarak $[2][3] = 1500$ , maka berarti bahwa jarak dari halte 2 menuju halte 3 adalah sebesar 1500 meter. Hal itu juga berlaku pada matriks busTersedia misalkan busTersedia $[2][3] = "3A, 4B"$  maka berarti bahwa dari halte 2 menuju halte 3 dapat ditempuh dengan menggunakan bus jalur 3A, atau 4B.

Dari dua buah matriks tadi sudah cukup untuk menerjemahkan graf agar dimengerti oleh aplikasi. Kemudian aplikasi pun bisa melakukan pencarian atau penelusuran jalur bus Trans Jogja. Pada aplikasi rekomendasi jalur bus Trans Jogja penelusuran jalur dari halte asal menuju halte tujuan salah satunya dibantu dengan algoritma Floyd-Warshall. Digunakannya algoritma ini karena karakteristik algoritma Floyd-Warshall mampu melakukan pencarian jalur bus terpendek dari semua pasangan titik yang mungkin berbeda dengan algoritma lainnya misalnya algoritma Dijkstra yang hanya mencari jalur terpendek dari satu pasang saja.

#### 4.1.2 Pembahasan Algoritma Floyd-Warshall

Untuk melakukan penelusuran jalur dari halte asal menuju halte tujuan digunakan algoritma Floyd-Warshall. Berbeda dengan algoritma penelusuran jalur lainnya seperti Dijkstra misalnya yang hanya mencari jalur terpendek dari satu titik menuju satu titik, algoritma Floyd-Warshall memiliki kelebihan yaitu penelusuran yang dilakukan akan menghasilkan jarak terpendek dari semua pasangan titik yang mungkin sekaligus menyimpan jalur yang dilewati dari semua pasangan titik tersebut. Gambar 4.2 adalah pseudocode dasar dari algoritma Floyd-Warshall.

```

for k=1 to n do
  for i=1 to n do
    for j=1 to n do
      if d[i][j] > d[i][k] + d[k][j] then
        d[i][j] = d[i][k] + d[k][j]

```

Gambar 4.2 Pseudocode algoritma Floyd-Warshall

Jika dikehendaki untuk sekaligus menyimpan jalur yang dibentuk ada sedikit tambahan dari pseudocode di atas yaitu dengan menambahkan satu baris perintah pada akhir baris (dicetak miring), sehingga menjadi seperti Gambar 4.3.

```

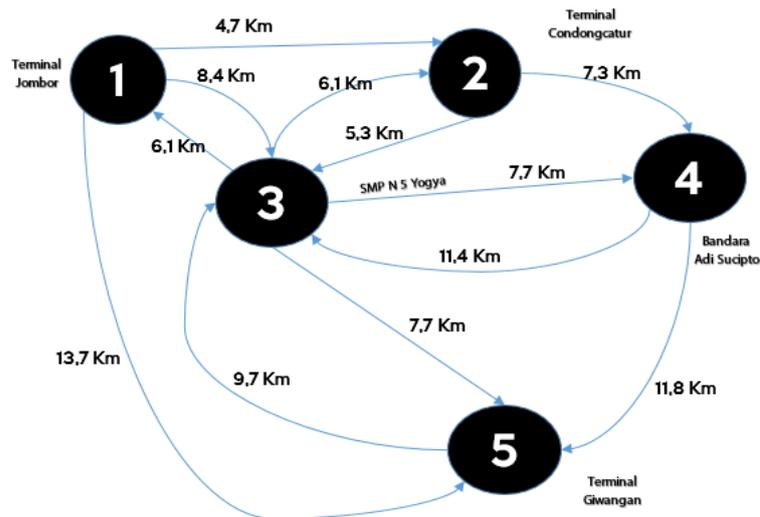
for k=1 to n do
  for i=1 to n do
    for j=1 to n do
      if d[i][j] > d[i][k] + d[k][j] then
        d[i][j] = d[i][k] + d[k][j]
        p[i][j] = p[i][k]

```

Gambar 4.3 Pseudocode algoritma Floyd-Warshall dan baris tambahan guna simpan jalur

Algoritma Floyd-Warshall akan melakukan iterasi sebanyak  $n$  kali atau sebanyak jumlah titik dari graf (nilai  $k$ ). Pada setiap iterasi tersebut akan dibandingkan jarak dari suatu titik asal menuju titik tujuan (jarak titik  $i$  ke  $j$ ) dengan penjumlahan jarak dari titik asal ke titik lain ditambah titik lain menuju titik tujuan (jarak  $i$  ke  $k$  ditambah jarak  $k$  ke  $j$ ). Hal tersebut dimaksudkan untuk mencari jarak terpendek dari titik asal ke titik tujuan (dari  $i$  ke  $j$ ) serta menyimpan jalur yang harus dilalui dari titik asal ke titik tujuan ( $p[i][j]$ ). Jika ternyata jarak dari penjumlahan ( $i$  ke  $k$  ditambah  $k$  ke  $j$ ) lebih kecil, maka jarak dari titik asal ke tujuan akan berubah menjadi nilai penjumlahan tersebut. Hasil akhir pada setiap iterasi tersebut akan mempengaruhi hasil akhir dari iterasi – iterasi selanjutnya.

Untuk memberikan penjelasan mengenai implementasi algoritma Floyd-Warshall pada graf Trans Jogja maka akan diterapkan pada graf Trans Jogja yang disederhanakan dengan mengurangi banyak titik serta banyak sisinya. Gambar 4.4 adalah graf Trans Jogja yang telah disederhanakan berdasarkan titik atau halte yang merepresentasikan graf Trans Jogja



Gambar 4.4 Graf sederhana dari graf Trans Jogja

Graf di atas merupakan bentuk yang disederhanakan, jadi sebenarnya dalam bentuk graf Trans Jogja lengkap masih terdapat banyak node/ tertex dan juga edge lainnya. Misalkan dari titik 1 menuju titik 3 sebenarnya masih banyak jalur lain serta titik lain yang bisa ditempuh. Pada tahap implementasi graf dan algoritma adalah mengisi nilai dari tiap matriks jarak sesuai dengan data yang ada pada graf. Nilai dari matriks jarak[1][2] berisi jarak dari halte 1 menuju halte 2 yaitu sebesar 4,7 Km. Sedangkan untuk pasangan titik yang tidak ada edgenya diisi dengan nilai tak terhingga (inf) sehingga matriks jarak akan berisi seperti Tabel 4.1.

Tabel 4.1 Matriks jarak pada kondisi awal

	1	2	3	4	5
1	inf	4,7	8,4	inf	13,7
2	inf	inf	5,3	7,3	inf
3	6,1	6,1	inf	7,7	7,7
4	inf	inf	11,4	inf	11,8
5	inf	inf	9,7	inf	inf

Tabel 4.2 Matriks jalur pada kondisi awal

	1	2	3	4	5
1	1	2	3	4	5
2	1	2	3	4	5
3	1	2	3	4	5
4	1	2	3	4	5
5	1	2	3	4	5

Selanjutnya adalah melakukan iterasi sebanyak 5 kali sesuai dengan jumlah titik yang ada pada graf. Setiap iterasi digunakan variable  $i, j, k$  sebagai bantuan penunjuk indeks matriks kemudian dilakukan perbandingan antar jarak ( $d[i][j] > d[i][k] + d[k][j]$ ). Berikut ini adalah pembahasan dari setiap iterasi dimulai dari iterasi 1 hingga selesai yaitu iterasi 5.

a. Iterasi 1 (Nilai  $k = 1$ )

1. Ketika nilai  $i = 1$

-  $j = 1$

Cek apakah  $d[1][1] > d[1][1] + d[1][1]$  ?

Apakah  $\infty > \infty + \infty$  ? Tidak, sehingga tidak ada perubahan

-  $j = 2$

Cek apakah  $d[1][2] > d[1][1] + d[1][2]$  ?

Apakah  $4,7 > \infty + 4,7$  ? Tidak, sehingga tidak ada perubahan

-  $j = 3$

Cek apakah  $d[1][3] > d[1][1] + d[1][3]$  ?

Apakah  $8,4 > \infty + 8,4$  ? Tidak, sehingga tidak ada perubahan

-  $j = 4$

Cek apakah  $d[1][4] > d[1][1] + d[1][4]$  ?

Apakah  $\infty > \infty + \infty$  ? Tidak, sehingga tidak ada perubahan

-  $j = 5$

Cek apakah  $d[1][5] > d[1][1] + d[1][5]$  ?

Apakah  $13,7 > \infty + 13,7$  ? Tidak, sehingga tidak ada perubahan

Untuk mempersingkat penjelasan, maka pada iterasi selanjutnya hanya akan dibahas ketika terjadi perubahan nilai saja kemudian hasil akhir dari tiap iterasinya.

2. Ketika nilai  $i = 3$  dan nilai  $j = 3$

Cek apakah  $d[3][3] > d[3][1] + d[1][3]$  ?

Apakah  $\infty > 6,1 + 8,4$  ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[3][3]$  menjadi nilai  $d[3][1] + d[1][3]$  ( $6,1 + 8,4 = 14,5$ ). Serta pada matriks jalur nilai dari matriks  $p[3][3]$  berubah menjadi  $p[3][1] = 1$ .

Hasil akhir dari iterasi pertama adalah seperti pada Tabel 4.3 dan Tabel 4.4:

Tabel 4.3 Matriks jarak saat iterasi pertama selesai

	1	2	3	4	5
1	inf	4,7	8,4	inf	13,7
2	inf	inf	5,3	7,3	inf
3	6,1	6,1	<u>14,5</u>	7,7	7,7
4	inf	inf	11,4	inf	11,8
5	inf	inf	9,7	inf	inf

Tabel 4.4 Matriks jalur saat iterasi pertama selesai

	1	2	3	4	5
1	1	2	3	4	5
2	1	2	3	4	5
3	1	2	<u>1</u>	4	5
4	1	2	3	4	5
5	1	2	3	4	5

## b. Iterasi 2 (Nilai k = 2)

## 1. Ketika nilai i = 1 dan nilai j = 4

Cek apakah  $d[1][4] > d[1][2] + d[2][4]$ ?

Apakah  $\text{inf} > 4,7 + 7,3$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[1][4]$  menjadi nilai  $d[1][2] + d[2][4]$  ( $4,7 + 7,3 = 12,0$ ). Serta pada matriks jalur nilai dari matriks  $p[1][4]$  berubah menjadi  $p[1][2] = 2$

## 2. Ketika nilai i = 3 dan nilai j = 3

Cek apakah  $d[3][3] > d[3][2] + d[2][3]$ ?

Apakah  $14,50 > 6,1 + 5,3$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[3][3]$  menjadi nilai  $d[3][2] + d[2][3]$  ( $6,1 + 5,3 = 11,4$ ). Serta pada matriks jalur nilai dari matriks  $p[3][3]$  berubah menjadi  $p[3][2] = 2$ .

Hasil akhir dari iterasi kedua adalah seperti pada Tabel 4.5 dan Tabel 4.6:

Tabel 4.5 Matriks jarak saat iterasi kedua selesai

	1	2	3	4	5
1	inf	4,7	8,4	<u>12,0</u>	13,7
2	inf	inf	5,3	7,3	inf
3	6,1	6,1	<u>11,4</u>	7,7	7,7
4	inf	inf	11,4	inf	11,8
5	inf	inf	9,7	inf	inf

Tabel 4.6 Matriks jalur saat iterasi kedua selesai

	1	2	3	4	5
1	1	2	3	<u>2</u>	5
2	1	2	3	4	5
3	1	2	<u>2</u>	4	5
4	1	2	3	4	5
5	1	2	3	4	5

## c. Iterasi 3 (Nilai k = 3)

1. Ketika nilai  $i = 1$  dan  $j = 1$ 

Cek apakah  $d[1][1] > d[1][3] + d[3][1]$  ?

Apakah  $\inf > 8,4 + 6,1$  ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[1][1]$  menjadi nilai  $d[1][3] + d[3][1]$  ( $8,4 + 6,1 = 14,5$ ). Serta pada matriks jalur nilai dari matriks  $p[1][1]$  berubah menjadi  $p[1][1] = 3$

2. Ketika nilai  $i = 2$  dan  $j = 1$ 

Cek apakah  $d[2][1] > d[2][3] + d[3][1]$ ?

Apakah  $\inf > 5,3 + 6,1$  ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[2][1]$  menjadi nilai  $d[2][3] + d[3][1]$  ( $5,3 + 6,1 = 11,4$ ). Serta pada matriks jalur nilai dari matriks  $p[2][1]$  berubah menjadi  $p[2][3] = 3$

3. Ketika nilai  $i = 2$  dan  $j = 2$ 

Cek apakah  $d[2][2] > d[2][3] + d[3][2]$ ?

Apakah  $\inf > 5,3 + 6,1$  ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[2][2]$  menjadi nilai  $d[2][3] + d[3][2]$  ( $5,3 + 6,1 = 11,4$ ). Serta pada matriks jalur nilai dari matriks  $p[2][1]$  berubah menjadi  $p[2][3] = 3$

4. Ketika nilai  $i = 2$  dan  $j = 5$ 

Cek apakah  $d[2][5] > d[2][3] + d[3][5]$ ?

Apakah  $\inf > 5,3 + 7,7$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[2][5]$  menjadi nilai  $d[2][3] + d[3][5]$  ( $5,3 + 7,7 = 13,0$ ). Serta pada matriks jalur nilai dari matriks  $p[2][5]$  berubah menjadi  $p[2][3] = 3$

5. Ketika nilai  $i = 4$  dan  $j = 1$ 

Cek apakah  $d[4][1] > d[4][3] + d[3][1]$ ?

Apakah  $\inf > 11,4 + 6,1$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[4][1]$  menjadi nilai  $d[4][3] + d[3][1]$  ( $11,4 + 6,1 = 17,5$ ). Serta pada matriks jalur nilai dari matriks  $p[4][1]$  berubah menjadi  $p[4][3] = 3$

6. Ketika nilai  $i = 4$  dan  $j = 2$

Cek apakah  $d[4][2] > d[4][3] + d[3][2]$ ?

Apakah  $\inf > 11,4 + 6,1$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[4][2]$  menjadi nilai  $d[4][3] + d[3][2]$  ( $11,4 + 6,1 = 17,5$ ). Serta pada matriks jalur nilai dari matriks  $p[4][2]$  berubah menjadi  $p[4][3] = 3$

7. Ketika nilai  $i = 4$  dan  $j = 4$

Cek apakah  $d[4][4] > d[4][3] + d[3][4]$ ?

Apakah  $\inf > 11,4 + 7,7$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[4][4]$  menjadi nilai  $d[4][3] + d[3][4]$  ( $11,4 + 7,7 = 19,1$ ). Serta pada matriks jalur nilai dari matriks  $p[4][4]$  berubah menjadi  $p[4][3] = 3$

8. Ketika nilai  $i = 5$  dan  $j = 1$

Cek apakah  $d[5][1] > d[5][3] + d[3][1]$ ?

Apakah  $\inf > 9,7 + 6,1$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[5][1]$  menjadi nilai  $d[5][3] + d[3][1]$  ( $9,7 + 6,1 = 15,8$ ). Serta pada matriks jalur nilai dari matriks  $p[5][1]$  berubah menjadi  $p[5][3] = 3$

9. Ketika nilai  $i = 5$  dan  $j = 2$

Cek apakah  $d[5][2] > d[5][3] + d[3][2]$ ?

Apakah  $\inf > 9,7 + 6,1$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[5][2]$  menjadi nilai  $d[5][3] + d[3][2]$  ( $9,7 + 6,1 = 15,8$ ). Serta pada matriks jalur nilai dari matriks  $p[5][2]$  berubah menjadi  $p[5][3] = 3$

10. Ketika nilai  $i = 5$  dan  $j = 4$

Cek apakah  $d[5][4] > d[5][3] + d[3][4]$ ?

Apakah  $\inf > 9,7 + 7,7$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[5][4]$  menjadi nilai  $d[5][3] + d[3][4]$  ( $9,7 + 7,7 = 17,4$ ). Serta pada matriks jalur nilai dari matriks  $p[5][4]$  berubah menjadi  $p[5][3] = 3$

11. Ketika nilai  $i = 5$  dan  $j = 5$

Cek apakah  $d[5][5] > d[5][3] + d[3][4]$ ?

Apakah  $\inf > 9,7 + 7,7$ ? Ya, sehingga terjadi perubahan yaitu nilai pada matriks  $d[5][5]$  menjadi nilai  $d[5][3] + d[3][4]$  ( $9,7 + 7,7 = 17,4$ ). Serta pada matriks jalur nilai dari matriks  $p[5][5]$  berubah menjadi  $p[5][3] = 3$ .

Hasil akhir dari iterasi ketiga adalah seperti pada Tabel 4.7 dan Tabel 4.8:

Tabel 4.7 Matriks jarak saat iterasi ketiga selesai

	1	2	3	4	5
1	<u>14,5</u>	4,7	8,4	12,0	13,7
2	<u>11,4</u>	<u>11,4</u>	5,3	7,3	<u>13,0</u>
3	6,1	6,1	11,4	7,7	7,7
4	<u>17,5</u>	<u>17,5</u>	11,4	<u>19,1</u>	11,8
5	<u>15,8</u>	<u>15,8</u>	9,7	<u>17,4</u>	<u>17,4</u>

Tabel 4.8 Matriks jalur saat iterasi ketiga selesai

	1	2	3	4	5
1	<u>3</u>	2	3	2	5
2	<u>3</u>	<u>3</u>	3	4	<u>3</u>
3	1	2	2	4	5
4	<u>3</u>	<u>3</u>	3	<u>3</u>	5
5	<u>3</u>	<u>3</u>	3	<u>3</u>	<u>3</u>

d. Iterasi 4 (Nilai  $k = 4$ )

Pada iterasi keempat ini tidak terjadi perubahan nilai, sehingga akhir dari iterasi keempat adalah seperti Tabel 4.9 dan Tabel 4.10:

Tabel 4.9 Matriks jarak saat iterasi keempat selesai

	1	2	3	4	5
1	<u>14,5</u>	4,7	8,4	12,0	13,7
2	<u>11,4</u>	<u>11,4</u>	5,3	7,3	<u>13,0</u>
3	6,1	6,1	11,4	7,7	7,7
4	<u>17,5</u>	<u>17,5</u>	11,4	<u>19,1</u>	11,8
5	<u>15,8</u>	<u>15,8</u>	9,7	<u>17,4</u>	<u>17,4</u>

Tabel 4.10 Matriks jalur saat iterasi keempat selesai

	1	2	3	4	5
1	<u>3</u>	2	3	2	5
2	<u>3</u>	<u>3</u>	3	4	<u>3</u>
3	1	2	2	4	5
4	<u>3</u>	<u>3</u>	3	<u>3</u>	5
5	<u>3</u>	<u>3</u>	3	<u>3</u>	<u>3</u>

## e. Iterasi 5 (Nilai k = 5)

Pada iterasi kelima ini tidak terjadi perubahan nilai, sehingga akhir dari iterasi kelima adalah seperti Tabel 4.11 dan Tabel 4.12:

Tabel 4.11 Matriks jarak saat iterasi kelima selesai

	1	2	3	4	5
1	<u>14,5</u>	4,7	8,4	12,0	13,7
2	<u>11,4</u>	<u>11,4</u>	5,3	7,3	<u>13,0</u>
3	6,1	6,1	11,4	7,7	7,7
4	<u>17,5</u>	<u>17,5</u>	11,4	<u>19,1</u>	11,8
5	<u>15,8</u>	<u>15,8</u>	9,7	<u>17,4</u>	<u>17,4</u>

Tabel 4.12 Matriks jalur saat iterasi kelima selesai

	1	2	3	4	5
1	<u>3</u>	2	3	2	5
2	<u>3</u>	<u>3</u>	3	4	<u>3</u>
3	1	2	2	4	5
4	<u>3</u>	<u>3</u>	3	<u>3</u>	5
5	<u>3</u>	<u>3</u>	3	<u>3</u>	<u>3</u>

Setelah semua iterasi selesai maka pada Tabel 4.13 dan Tabel 4.14 adalah perbandingan antara matriks jarak dan matriks jalur pada kondisi awal sebelum dan sesudah dilakukan perhitungan algoritma Floyd-Warshall dengan setelah dilakukan perhitungan.

Tabel 4.13 Perbandingan matriks jarak sebelum dan sesudah dihitung

Matriks Jalur Tahap Awal						Matriks Jarak pada akhir iterasi					
	1	2	3	4	5		1	2	3	4	5
1	inf	4,7	8,4	inf	13,7	1	<u>14,5</u>	4,7	8,4	12,0	13,7
2	inf	inf	5,3	7,3	inf	2	<u>11,4</u>	<u>11,4</u>	5,3	7,3	<u>13,0</u>
3	6,1	6,1	inf	7,7	7,7	3	6,1	6,1	11,4	7,7	7,7
4	inf	inf	11,4	inf	11,8	4	<u>17,5</u>	<u>17,5</u>	11,4	<u>19,1</u>	11,8
5	inf	inf	9,7	inf	inf	5	<u>15,8</u>	<u>15,8</u>	9,7	<u>17,4</u>	<u>17,4</u>

Tabel 4.14 Perbandingan matriks jalur sebelum dan sesudah dihitung

Matriks Jalur Tahap Awal						Matriks Jarak pada akhir iterasi					
	1	2	3	4	5		1	2	3	4	5
1	1	2	3	4	5	1	<u>3</u>	2	3	<u>2</u>	5
2	1	2	3	4	5	2	<u>3</u>	<u>3</u>	3	4	<u>3</u>
3	1	2	3	4	5	3	1	2	<u>2</u>	4	5
4	1	2	3	4	5	4	<u>3</u>	<u>3</u>	3	<u>3</u>	5
5	1	2	3	4	5	5	<u>3</u>	<u>3</u>	3	<u>3</u>	<u>3</u>

Setelah perhitungan algoritma Floyd-Warshall selesainya maka dapat diketahui jarak terpendek dan jalur yang dilewati dari sluruh pasangan titik asal menuju titik tujuannya selagi dimungkinkan adanya jalur menuju ke titik tersebut. Sebagai contoh dalam pencarian jalur terpendek dari titik 1 menuju titik 4.

Pada saat awal sebelum dilakukan perhitungan algoritma Floyd-Warshall jika dilihat pada matriks jarak[1][4] maka akan bernilai infinity karena memang tidak ada jalur langsung yang terbentuk. Namun ketika setelah selesai dilakukan perhitungan maka matriks jarak[1][4] akan bernilai 12,0. Nilai 12,0 ini merupakan jarak terpendek yang ditempuh dari titik 1 menuju titik 4 dengan jalur 1 => 2 => 4 dibandingkan dengan opsi lainnya. Jalur 1 => 2 => 4 yang terbentuk tersebut dapat diketahui dengan melakukan penelusuran pada matriks jalur dengan pseudocode pada Gambar 4.5.

```

for i=1 to n
  for j=1 to n
    if i != j then
      u = i
      v = j
      while u != v
        u = jalur[u][v]
        if i = asal && j = tujuan then
          jalur.add(u)

```

Gambar 4.5 Pseudocode penelusuran jalur algoritma Floyd-Warshall

Alur penelusuran di atas akan lebih mudah dipahami dengan melihat tabel matriks jalur final yang telah terbentuk. Penelusuran dilakukan urut per baris dan per kolom pada matriks

jalur. Pertama-tama akan dicari terlebih dahulu per baris dan per kolom, pada matriks bagian mana saja yang berubah nilainya dibandingkan dengan nilai awal sebelum dilakukan perhitungan dengan algoritma Floyd-Warshall dan tidak dilakukan pengecekan matriks pada baris dan kolom yang sama (matriks baris 1 kolom 1, matriks baris 2 kolom 2, dst. tidak perlu dicek meskipun terjadi perubahan nilai).

Ketika ditemukan adanya perubahan nilai akan dicek apakah nilai yang berubah tersebut berada pada baris dan kolom yang menunjukkan titik asal dan tujuan (baris merupakan titik asal, kolom merupakan titik tujuan). Jika perubahan tersebut bukan pada baris dan kolom yang menunjukkan titik asal dan titik tujuan, maka lanjutkan mencari nilai matriks jalur yang berubah, namun jika nilai yang berubah tersebut berada pada baris dan kolom yang menunjuk pada titik asal dan tujuan maka simpanlah nilai tersebut lalu lihat nilai yang berubah tersebut kemudian lakukan pengecekan nilai pada matriks baris nilai yang berubah tadi menuju ke titik tujuan jika nilainya sudah sama dengan titik tujuan maka simpan nilainya dan jalur terpendek dari titik asal menuju titik tujuan sudah ditemukan. Dalam kasus pencarian jalur dari titik 1 menuju 4 terjadi perubahan nilai pada matriks jalur di baris pertama dan kolom keempat. Perubahan itu sesuai dengan ketentuan yang menunjukkan titik asal dan tujuan, maka simpan nilai tersebut yaitu 2. Lalu lakukan pengecekan nilai pada matriks jalur baris 2 ke kolom 4. Karena nilai dari baris 2 kolom 4 bernilai 4 yang artinya titik tujuan, maka simpan nilai 4 dan jalur dari titik 1 menuju titik 4 telah terbentuk yaitu  $1 \Rightarrow 2 \Rightarrow 4$ .

Kelemahan penerapan algoritma Floyd-Warshall pada graf Trans Jogja adalah jalur terpendek yang ditempuh bisa memungkinkan perlunya transit/ berganti jalur bus yang beraneka ragam mulai dari tanpa transit hingga lima kali transit.

## **4.2 Implementasi Google Maps API dan Google Places API**

### **4.2.1 Implementasi Google Maps API**

Penggunaan Google Maps API pada aplikasi yang dikembangkan dapat dilihat pada bagian – bagian aplikasi yang menampilkan sebuah peta. Bagian API yang digunakan adalah marker untuk menandai suatu lokasi dengan simbol tertentu. Selain marker bagian API yang digunakan adalah polyline untuk membentuk garis lurus antar dua buah titik koordinat.

### **4.2.2 Implementasi Google Places API**

Penggunaan Google Places API pada aplikasi yang dikembangkan adalah pada bagian penentuan lokasi. Lokasi yang ditentukan dapat berupa nama suatu tempat yang terkenal, juga bisa berupa sebuah titik pada peta. Modul pada API yang digunakan pada pengembangan

aplikasi ini adalah Place Picker untuk mendapatkan koordinat suatu tempat atau suatu titik pada peta Google Maps. Selain itu modul lain yang digunakan Place Autocomplete yang berfungsi untuk membantu memudahkan pengguna ketika mencari suatu alamat maupun suatu tempat pada kotak pencarian dengan memberikan prediksi nama suatu tempat.

### 4.3 Batasan Implementasi

a. Data yang diolah

Data yang diolah seperti rute perjalanan jalur bus, halte bus merupakan data yang diperoleh penulis berdasarkan data dan informasi dari peta Trans Jogja yang diterbitkan oleh Dishub DIY hingga Juni 2017

b. Lokasi halte bus

Lokasi halte pada peta Google Maps tidaklah mutlak tepat sama jika dibandingkan dengan lokasi halte sesungguhnya. Namun lokasi halte pada peta Google Maps tersebut sudah cukup untuk memberikan gambaran mengenai lokasi halte pada lokasi sebenarnya.

c. Perhitungan jarak halte terdekat dengan lokasi

Perhitungan jarak dihitung dari jarak ditarik garis lurus dari dua buah titik koordinat longitude dan latitude dari sebuah lokasi sehingga hasil perhitungan tidaklah tepat sepenuhnya karena bergantung dengan penempatan suatu titik akan tetapi jarak ini bisa digunakan sebagai gambaran mengenai jarak dari dua buah lokasi

d. Rekomendasi jalur yang diberikan

Rekomendasi jalur bus yang diberikan bisa dikelompokkan hingga tiga jenis tergantung dari halte asal dan halte tujuan yang dipilih. Ketiga pengelompokan tersebut ialah jalur bus tanpa transit, jalur bus sekali transit, dan yang pasti terdapat jalur terpendek. Rekomendasi jalur bus terpendek ditentukan dengan penerapan algoritma Floyd – Warshall sehingga dengan algoritma ini bisa memunculkan rekomendasi dengan jumlah transit/ berganti jalur bus yang berbeda – beda. Rekomendasi lain yang dicari melalui pengecekan apakah memungkinkan untuk menggunakan bus dengan tanpa transit dengan mencocokkan apakah terdapat jalur bus yang sama pada halte asal dan halte tujuan dan pengecekan apakah memungkinkan terdapat sekali transit/ berganti jalur bus.

e. Jarak antar halte

Perhitungan jarak antar halte didapatkan memanfaatkan perhitungan jarak antar dua titik dengan memanfaatkan fitur perhitungan jarak Google Maps sehingga jarak dalam keadaan sesungguhnya bisa jadi berbeda dengan yang ada pada aplikasi.

#### 4.4 Implementasi Aplikasi

Pada aplikasi yang telah dikembangkan jumlah halte bus yang diolah hanya terdapat sebanyak 212 halte bus saja dari 265 halte yang terdapat pada peta bus Trans Jogja milik Dishub DIY. Jumlah tersebut tentunya berbeda dengan keadaan nyata pada saat ini karena memang jumlah halte akan terus bertambah. Sedangkan jalur bus yang diolah yaitu sebanyak 17 jalur yang ada yaitu 1A – 6A, 1B – 6B, 7, 8, 9, 10, dan 11 yang berlaku hingga Juni 2017 sehingga memungkinkan adanya perbedaan dengan rute perjalanan yang terbaru berlaku untuk saat ini maupun kedepan.

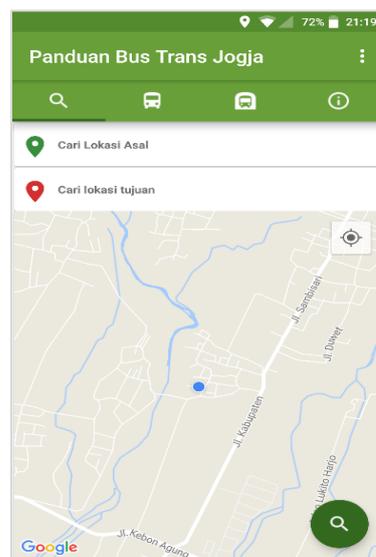
Selanjutnya pada pembahasan implementasi aplikasi ini dibahas mengenai implementasi antarmuka dengan dikelompokkan tiap menu dan tiap antarmuka yang terdapat pada menu tersebut. Terdapat sebuah skenario kasus nyata yang digunakan untuk memperlihatkan sekaligus memberikan gambaran mengenai aplikasi ketika dijalankan misalkan pada saat pencarian panduan jalur bus.

##### 4.4.1 Implementasi Menu Pencarian Panduan Jalur bus

Untuk memberikan gambaran tentang jalannya aplikasi ketika pengguna ingin mendapatkan panduan jalur bus yang bisa digunakan, maka digunakan skenario kasus ketika pengguna hendak menggunakan Trans Jogja untuk pergi dari Bandara Adi Sucipto menuju ke Hotel Inna Garuda. Berikut ini adalah tahapan – tahapan yang dilalui.

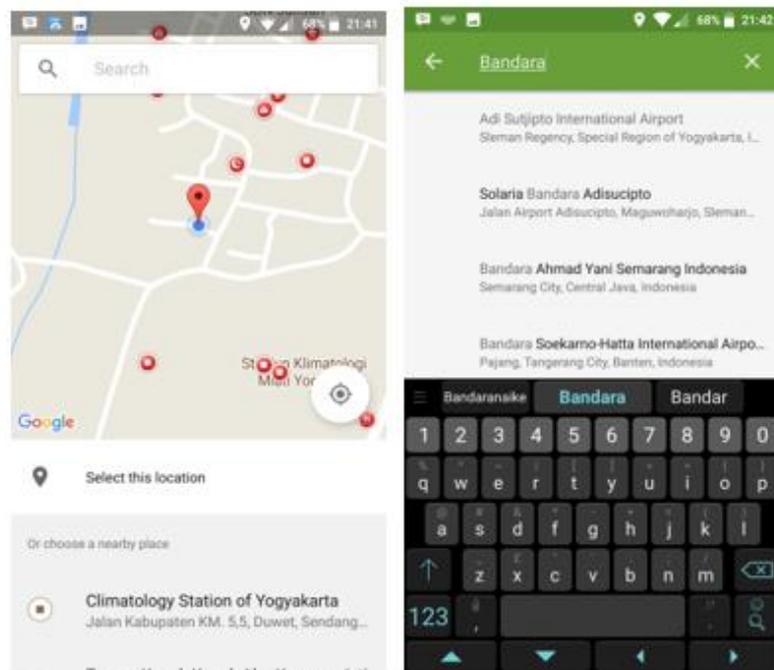
a. Implementasi antarmuka penentuan lokasi asal dan lokasi tujuan

Pada awal ketika pengguna menjalankan aplikasi maka akan tampil halaman utama seperti Gambar 4.6.

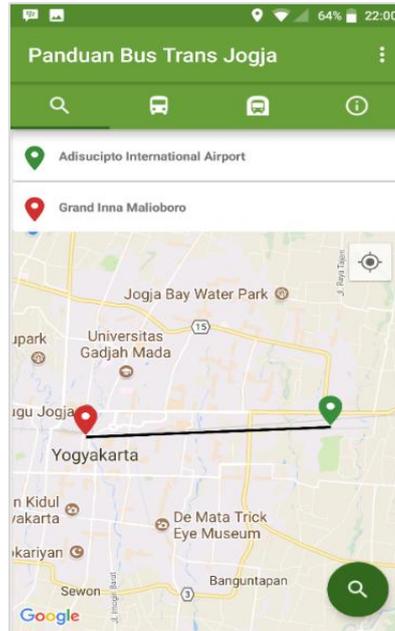


Gambar 4.6 Antar muka halaman utama aplikasi

Kemudian pengguna menentukan lokasi asal yaitu Bandara Adi Sucipto dan lokasi tujuan yaitu Hotel Inna Garuda dengan menekan bagian “Cari lokasi asal” dan “Cari lokasi tujuan”. Setelah menekan pada bagian tersebut secara langsung aplikasi akan menampilkan peta dan pada bagian bawah akan berisi daftar – daftar tempat terdekat dengan lokasi pengguna saat ini. Pengguna tersebut bisa memilih salah satu tempat yang terdapat pada daftar tersebut atau menentukan pin pada lokasi tertentu atau juga bisa mencari sebuah lokasi pada kolom pencarian seperti pada Gambar 4.7. Setelah lokasi asal dan lokasi tujuan selesai ditentukan maka aplikasi menampilkan kedua lokasi tersebut ditandai dengan pin warna hijau untuk lokasi asal (Bandara Adi Sucipto) dan pin warna merah untuk lokasi tujuan (Hotel Inna Garuda) kemudian dihubungkan sebuah garis seperti pada Gambar 4.8. Pada kolom pencarian akan berubah menjadi nama lokasi atau alamat suatu lokasi. Kemudian pengguna harus menekan ikon “cari” untuk melanjutkan ke tahap selanjutnya dalam mencari panduan.

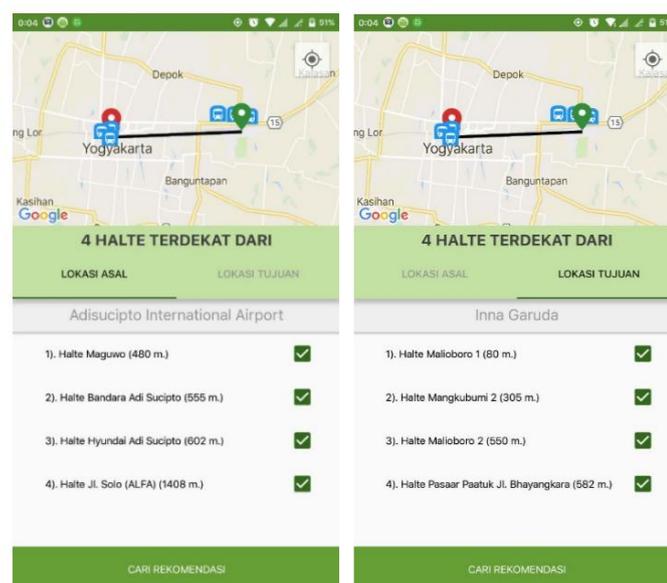


Gambar 4.7 Antarmuka saat menentukan lokasi asal atau lokasi tujuan



Gambar 4.8 Antarmuka setelah lokasi asal dan tujuan ditentukan

- b. Implementasi antarmuka halte terdekat dengan lokasi asal dan tujuan
- Setelah pengguna menekan tombol search, maka aplikasi akan mencari dari masing – masing empat halte terdekat yang diukur jarak lurus dari titik lokasi asal (Bandara Adi Sucipto) dan lokasi tujuan (Hotel Inna Garuda) kemudian juga menampilkan daftar empat halte terdekat dari lokasi asal dan tujuan tersebut beserta letak halte – halte pada peta Google Maps seperti pada Gambar 4.9.



Gambar 4.9 Antarmuka daftar halte terdekat dari lokasi asal dan lokasi tujuan

Ketika pengguna ingin melihat letak suatu halte misalkan halte bandara Adi Sucipto, maka hanya perlu menekan nama halte tersebut pada daftar kemudian tampilan peta akan berfokus pada halte yang dipilih serta menampilkan informasi singkat mengenai halte tersebut seperti pada Gambar 4.10. Untuk melanjutkan proses pencarian panduan jalur bus. Pengguna perlu menekan tombol “CARI REKOMENDASI”.



Gambar 4.10 Antarmuka ketika nama salah satu halte ditekan

c. Implementasi antarmuka rekomendasi pilihan halte

Tahap selanjutnya dalam mencari panduan bus Trans Jogja adalah melihat daftar rekomendasi pilihan halte. Rekomendasi pilihan halte merupakan pasangan – pasangan dari halte – halte terdekat dari lokasi asal dan lokasi tujuan. Pada setiap rekomendasi pilihan halte terdapat informasi mengenai jarak halte asal dengan lokasi asal, jarak terpendek antar kedua halte, jarak halte tujuan menuju lokasi tujuan, informasi jalur bus untuk jalur terpendeknya serta jika dimungkinkan ada informasi singkat ketersediaan pilihan jalur bus dengan tanpa transit juga sekali transit. Tampilan rekomendasi pilihan halte berdasarkan lokasi asal (Bandara Adi Sucipto) menuju lokasi tujuan (Hotel Inna Garuda) akan terlihat seperti pada Gambar 4.11.

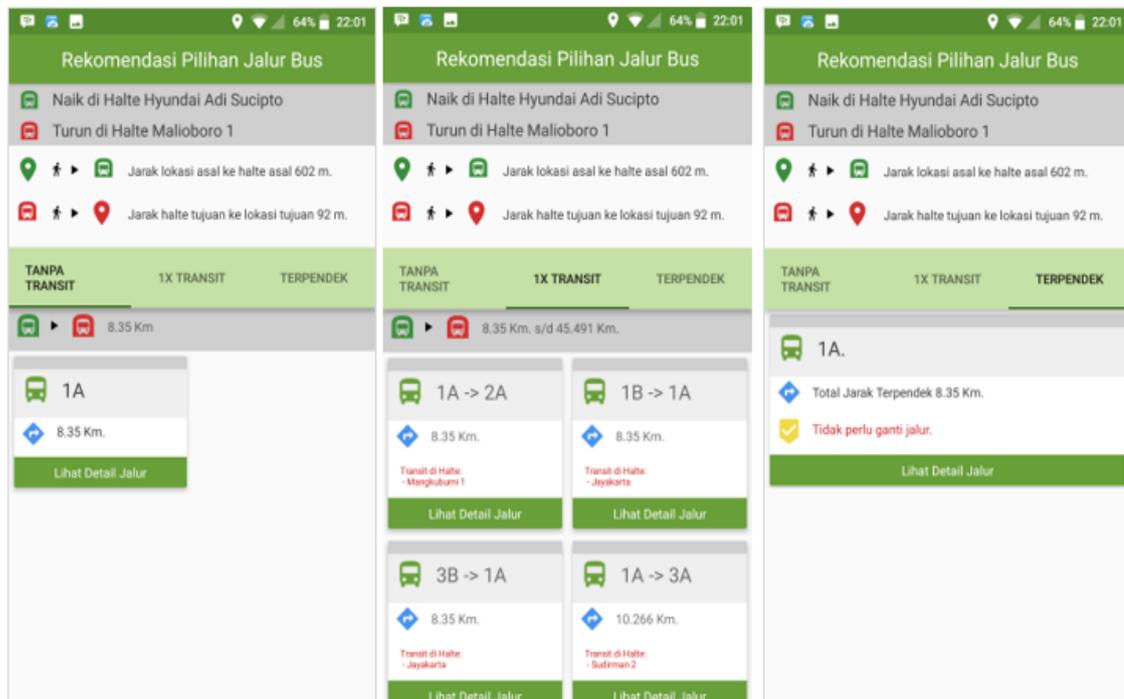


Gambar 4.11 Antarmuka daftar rekomendasi pilihan halte bus

Langkah selanjutnya adalah melihat detail rekomendasi untuk melihat pilihan jalur bus yang bisa digunakan dari pilihan halte asal dan tujuan bus dengan menekan tombol “Lihat detail rekomendasi” pada bagian bawah tiap-tiap rekomendasi.

d. Implementasi antarmuka rekomendasi pilihan jalur bus

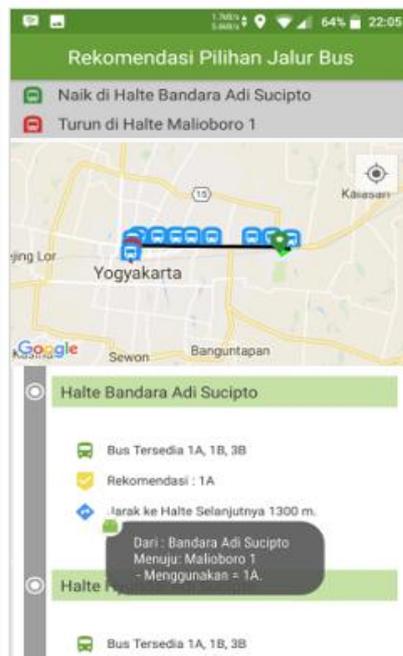
Setelah memilih pasangan halte asal dan tujuan tahap selanjutnya pengguna dapat melihat rekomendasi pilihan rute bus yang bisa digunakan. Apabila dimungkinkan adanya pilihan rute bus tanpa transit, sekali transit, maka tampilan rekomendasi pilihan rute bus akan terbagi menjadi 3 bagian. Untuk kasus lokasi asal Bandara Adi Sucipto menuju lokasi tujuan Hotel Inna Garuda, kemudian memilih rekomendasi pilihan halte yang dari Halte Bandara Adi Sucipto menuju Halte Malioboro 1, maka tampilan untuk rekomendasi pilihan rute busnya tampak seperti pada Gambar 4.12.



Gambar 4.12 Antarmuka detail rekomendasi pilihan bus

Pada tahap terakhir pengguna hanya perlu memilih salah satu pilihan rute busnya untuk melihat lebih detail mengenai jalur tersebut, halte – halte mana yang dilalui, berapa jarak antar halte dan total jarak tempuhnya beserta ringkasan perjalanannya.

- e. Implementasi antarmuka detail panduan jalur bus dari halte asal menuju halte tujuan
- Tahap terakhir dalam mencari panduan adalah melihat detail jalur bus yang memuat perjalanan bus. Terdapat informasi halte – halte mana saja yang dilewati, bus apa saja yang bisa melewati antar kedua halte, bus yang direkomendasikan untuk dipilih, jarak menuju halte berikutnya, dan jika harus transit pada halte tertentu maka akan ada keterangan pada salah satu halte bahwa harus ransit berganti jalur bus lain. Dan di bagian terakhir terdapat ringkasan singkat perjalanan. Untuk perjalanan dari lokasi asal Bandara Adi Sucipto menuju lokasi tujuan Hotel Inna Garuda dengan naik pada halte Bandara Adi Sucipto dan turun pada halte Malioboro 1 dengan naik bus tanpa transit sekaligus jalur terpendek yaitu 1A, maka tampilan detail panduan bus akan seperti pada Gambar 4.13 dengan ikon bus berlatar belakang biru merupakan halte – halte yang dilalui, ikon pin hijau merupakan lokasi asal, ikon pin merah merupakan lokasi tujuan.

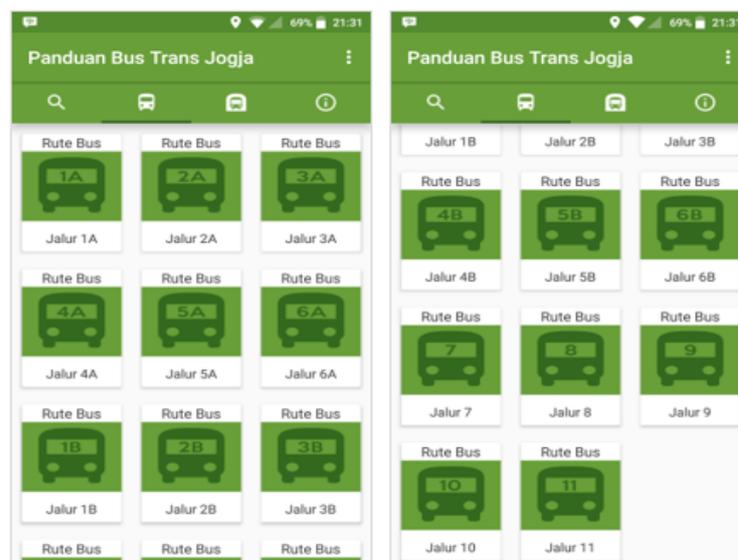


Gambar 4.13 Antarmuka detail perjalanan

#### 4.4.2 Implementasi pada Menu Rute Bus

Menu lain pada aplikasi adalah Rute Bus yang akan memberikan informasi mengenai rute bus Trans Jogja yaitu mulai dari rute bus jalur 1A hingga jalur 11. Pada saat awal pengguna memilih menu ini maka akan ditampilkan daftar jalur bus kemudian pengguna memilih salah satu jalur bus untuk ditampilkan rute bus jalur tersebut.

a. Implementasi antarmuka daftar rute bus

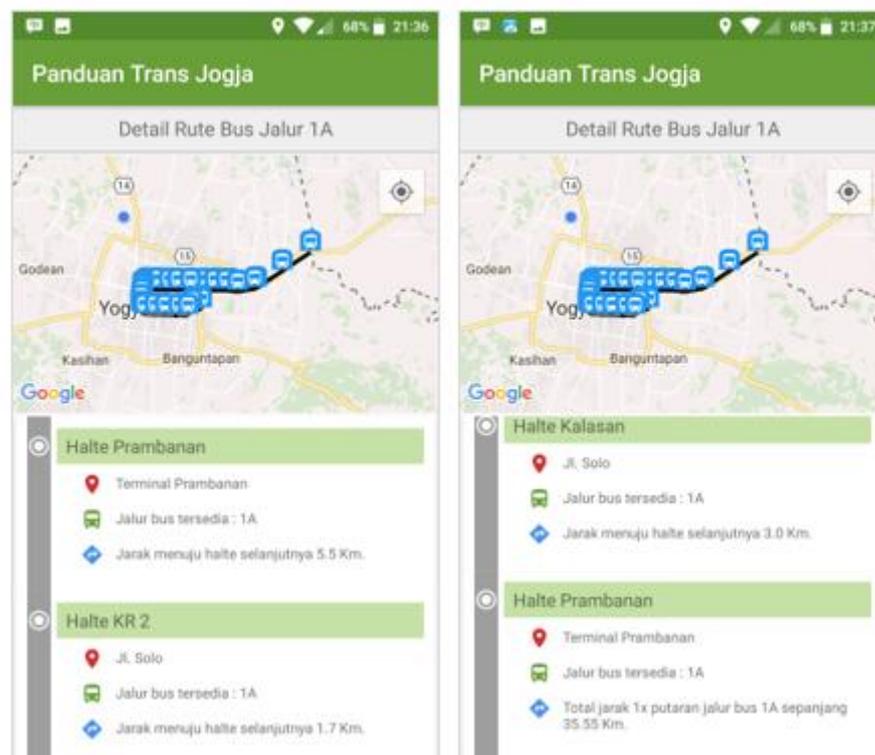


Gambar 4.14 Antarmuka daftar jalur bus Trans Jogja

Gambar 4.14 adalah tampilan antarmuka daftar jalur bus Trans Jogja yaitu mulai jalur 1A hingga jalur 11 yang ditampilkan secara grid 3 kolom per barisnya. Selanjutnya pengguna hanya perlu memilih salah satu jalur bus tersebut untuk melihat detail rute bus jalur yang dipilih.

b. Implementasi detail rute bus

Setelah memilih salah satu jalur bus, maka aplikasi akan menampilkan detail rute perjalanan bus jalur tersebut. Misalkan pengguna memilih jalur 1A, maka detail rute memuat urutan perjalanan bus mulai dari halte awal bus berangkat yaitu halte Prambanan hingga kembali menuju halte asal tersebut untuk satu kali putaran seperti pada Gambar 4.15.

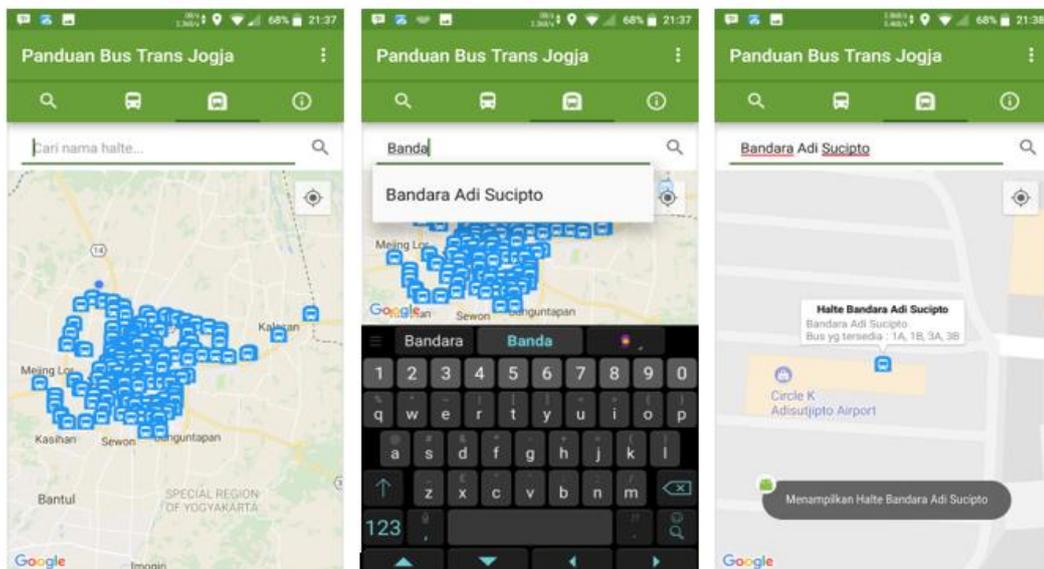


Gambar 4.15 Antarmuka detail rute perjalanan bus jalur 1A

Pada bagian atas terdapat peta beserta ikon halte – halte bus yang dilalui oleh bus jalur jalur 1A. Sedangkan pada bagian urutan rute perjalanan bus terdapat daftar halte yang dilalui beserta informasi alamat halte dan bus yang juga tersedia pada halte tersebut. Ketika pengguna menekan salah satu nama halte maka pada bagian peta akan berpindah fokus untuk menunjukkan letak halte tersebut.

#### 4.4.3 Implementasi pada Menu Halte Bus

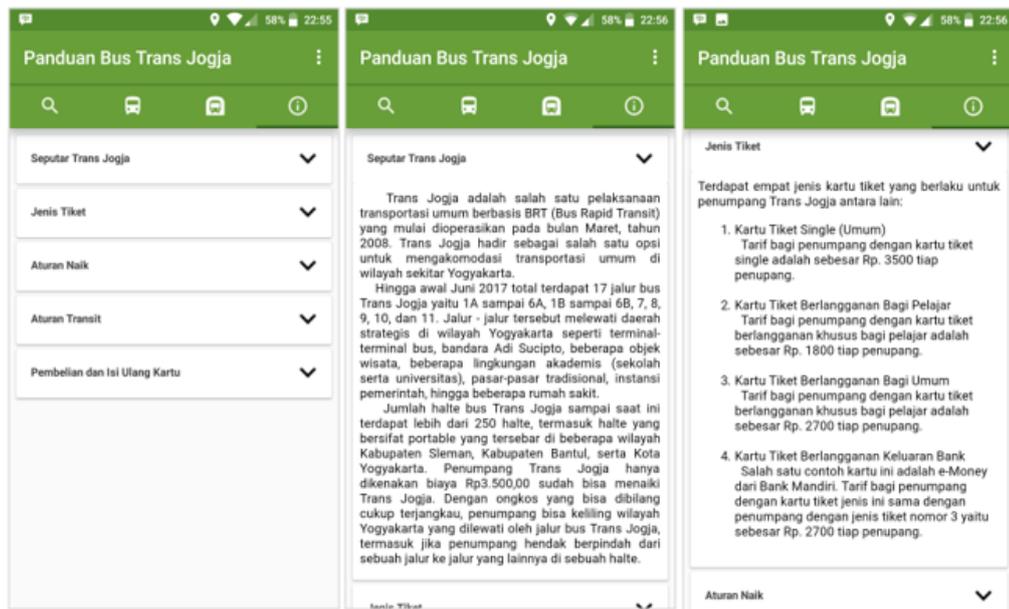
Menu ketiga yang terdapat pada aplikasi adalah menu halte bus seperti pada Gambar 4.16 yang memuat informasi mengenai halte – halte bus Trans Jogja. Pada bagian awal menu akan menampilkan daftar halte – halte bus Trans Jogja yang ditampilkan pada peta. Pengguna dapat mencari halte bus tertentu dengan menuliskan nama halte pada kolom pencarian. Apabila halte yang dicari berhasil ditemukan maka tampilan peta akan berpindah fokus menuju halte yang dicari tersebut dengan menampilkan informasi mengenai halte tersebut yaitu nama halte, alamat halte, serta bus yang tersedia pada halte tersebut. Jika pengguna ingin melihat informasi sebuah halte pengguna hanya cukup menekan salah satu ikon halte bus untuk menampilkan informasi halte tersebut.



Gambar 4.16 Beberapa antarmuka pada menu informasi halte bus

#### 4.4.4 Implementasi pada Menu Informasi Lain Seputar Trans Jogja

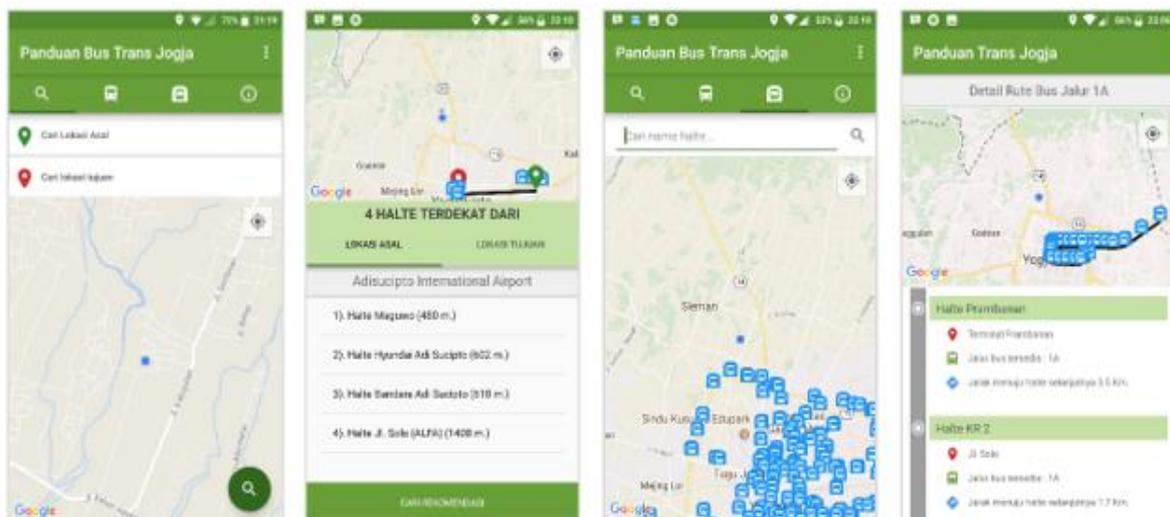
Menu terakhir yang terdapat pada aplikasi adalah menu informasi lain seputar Trans Jogja. Pada menu ini akan menampilkan informasi seputar Trans Jogja yang dikelompokkan menjadi lima bagian yaitu informasi sekitas Trans Jogja, informasi jenis tiket, informasi aturan naik, informasi aturan transit, dan informasi pembelian dan isi ulang kartu. Gambar 4.17 merupakan tampilan antarmuka menu informasi lain seputar Trans Jogja.



Gambar 4.17 Beberapa antarmuka pada menu informasi lain seputar Trans Jogja

#### 4.4.5 Implementasi Antarmuka Lihat Lokasi Pengguna Saat Ini

Untuk mengetahui lokasi pengguna saat ini dapat memanfaatkan beberapa antarmuka yang memuat peta seperti pada menu panduan bus, menu detail rute perjalanan bus, dan menu informasi halte bus yaitu dengan icon titik bulat berwarna biru seperti pada Gambar 4.18. Pengguna hanya perlu menekan icon untuk memperlihatkan letak lokasi pengguna saat ini pada peta.



Gambar 4.18 Beberapa antarmuka yang dapat lokasi pengguna saat ini

## 4.5 Pengujian Aplikasi

Bagian pengujian aplikasi dibagi menjadi tiga bagian yaitu pengujian fungsionalitas, pengujian kompatibilitas aplikasi, serta pengujian aplikasi kepada pengguna (UAT). Pengujian fungsionalitas dilakukan untuk mengetahui apakah kebutuhan aplikasi telah terpenuhi dan berjalan baik atau tidak. Pengujian kompatibilitas dilakukan untuk mengetahui apakah aplikasi dapat berjalan atau tidak pada beberapa perangkat Android dengan spesifikasi yang berbeda – beda. Pengujian aplikasi pada pengguna dimaksudkan untuk mengetahui tanggapan pengguna mengenai aplikasi yang telah dikembangkan.

### 4.5.1 Pengujian Fungsionalitas Aplikasi

Pengujian fungsionalitas dilakukan oleh penulis melalui perangkat penulis pribadi yaitu Xiaomi Redmi Note 3. Tabel 4.15 berikut mencatat hasil dari pengujian fungsionalitas yang telah dilakukan.

Tabel 4.15 Hasil pengujian fungsionalitas

No.	Fungsionalitas	Skenario	Aktivitas	Hasil	Ket.
1.	Lihat halte terdekat	Menentukan lokasi asal dan tujuan	Memilih menu panduan, menekan kolom pencarian, mencari kemudian menentukan lokasi asal serta lokasi tujuan. Menentukan lokasi asal/ tujuan jauh dari pusat Yogyakarta. Mengosongkan lokasi kemudian menekan tombol cari	Aplikasi menampilkan kolom pencarian. Setelah lokasi asal dan tujuan ditentukan aplikasi menampilkan lokasi asal dan tujuan pada peta dengan pin dan dihubungkan dengan sebuah garis. Jika lokasi asal terlalu jauh dari pusat Yogyakarta aplikasi menampilkan peringatan bahwa lokasi yang ditentukan terlalu jauh dari pusat Yogyakarta. Muncul peringatan lokasi belum ditentukan	Berhasil
		Lihat halte terdekat	Klik ikon cari (setelah menentukan lokasi	Aplikasi menampilkan daftar halte terdekat dengan lokasi asal dan tujuan.	Berhasil

			asal dan tujuan), tekan nama halte, tekan ikon halte.	Ketika nama halte ditekan fokus peta berpindah pada halte tersebut dan menampilkan informasi mengenai halte.	
2.	Lihat Rekomendasi panduan dari lokasi asal ke lokasi tujuan	Lihat rekomendasi pilihan halte	Klik “lihat rekomendasi” (setelah lihat halte terdekat)	Aplikasi menampilkan daftar rekomendasi pilihan halte yang dapat dipilih dan menampilkan informasi mengenai jarak menuju halte asal dan tujuan, jarak tempuh terpendek, serta informasi ketersediaan jalur bus	Berhasil
		Lihat rekomendasi pilihan jalur	klik “lihat detail rekomendasi bus” (setelah lihat rekomendasi pilihan halte)	Aplikasi menampilkan daftar pilihan jalur bus yang bisa digunakan untuk menempuh perjalanan dari halte asal menuju halte tujuan yang dipilih.	Berhasil
3.	Lihat detail jalur bus	Lihat detail jalur bus	Klik pilihan jalur bus (Setelah memilih jalur bus pada rekomendasi pilihan jalur)	Aplikasi menampilkan detail perjalanan bus dari halte asal dan tujuan serta jalur bus yang telah dipilih pada tahap sebelumnya disertai informasi halte yang dilalui dan ringkasan perjalanan	Berhasil
4.	Lihat Informasi jalur bus	Lihat Informasi jalur bus	Memilih menu info jalur bus, memilih jalur bus	Aplikasi menampilkan daftar jalur bus dari jalur 1A hingga 11, saat salah satu jalur dipilih akan menampilkan rute perjalanan bus 1x putaran dan menampilkan halte – halte yang dilalui pada peta.	Berhasil

5.	Lihat Informasi halte bus	Lihat Informasi halte bus	Memilih menu info halte bus, mencari halte bus, menekan ikon halte bus	aplikasi menampilkan daftar halte bus pada peta dengan ikon halte bus. Ketika dalam pencarian halte bus halte bus ditemukan fokus peta berubah pada halte yang dicari dan menampilkan informasi halte bus. Ketika salah satu ikon bus ditekan aplikasi menampilkan informasi halte bus tersebut.	Berhasil
6.	Lihat Informasi lain seputar Trans Jogja	Lihat Informasi lain seputar Trans Jogja	Memilih menu info lain seputar Trans Jogja, memilih sub info	Aplikasi menampilkan informasi lain seputar Trans Jogja yang dikelompokkan menjadi informasi sekilas Trans Jogja, informasi jenis tiket, informasi aturan naik, informasi aturan transit, dan informasi pembelian dan isi ulang kartu. Ketika dipilih salah satu sub info, aplikasi menampilkan informasi dari sub info tersebut	Berhasil
7.	Lihat lokasi pengguna saat ini	Lihat lokasi pengguna saat ini	Menekan tombol lihat lokasi pengguna saat ini	Fokus peta berpindah ke titik biru yang merupakan tanda lokasi pengguna saat ini	Berhasil

Berdasarkan hasil pengujian fungsionalitas di atas dapat disimpulkan bahwa aplikasi telah berjalan dengan baik serta telah sesuai dengan kebutuhan seperti yang telah dirancang sebelumnya.

#### 4.5.2 Pengujian Kompatibilitas Aplikasi

Pengujian kompatibilitas aplikasi dilakukan pada beberapa perangkat dengan spesifikasi yang berbeda – beda. Berikut ini adalah daftar perangkat yang digunakan untuk pengujian beserta keterangan dan spesifikasinya:

a. Xiaomi Redmi Note 2

Xiaomi Redmi Note 2 yang digunakan untuk pengujian memiliki spesifikasi layar 5.5 inch, prosessor octa core 2.0 GHz, RAM 2 GB, penyimpanan internal 16 GB, dan sistem operasi Android versi 5.0 (Lollipop). Aplikasi dapat berjalan pada perangkat ini dengan baik.

b. Xiaomi Redmi Note 3

Xiaomi Redmi Note 3 yang digunakan untuk pengujian memiliki spesifikasi layar 5.5 inch, prosessor Hexa core 1.8 GHz, RAM 3 GB, penyimpanan internal 32 GB, dan pernah menggunakan sistem operasi Android versi 6.0.1 (Marshmallow) dan 8.0.0 (Oreo). Aplikasi dapat berjalan pada perangkat ini dengan baik.

c. Xiaomi Redmi Note 4

Xiaomi Redmi Note 4 yang digunakan untuk pengujian memiliki spesifikasi layar 5.5 inch, prosessor octa core 2.0 GHz, RAM 3 GB, penyimpanan internal 32 GB, dan sistem operasi Android versi 7.1.2 (Nougat). Aplikasi dapat berjalan pada perangkat ini dengan baik

d. Asus Zenfone 4s

Asus Zenfone 4s yang digunakan untuk pengujian memiliki spesifikasi layar 4.5 inch, prosessor dual core 1.2 GHz, RAM 1 GB, penyimpanan internal 8 GB, dan sistem operasi Android versi 4.4.4 (Kitkat). Aplikasi juga dapat berjalan pada perangkat ini. Namun pada beberapa bagian terdapat sedikit perbedaan tampilan dengan perangkat yang digunakan sebelumnya.

e. Asus ZenPad C 7.0 3G (Z170CG)

Asus ZenPad C 7.0 3G (Z170CG) yang digunakan untuk pengujian memiliki spesifikasi layar 7.0 inch, prosessor Intel AtomTM X3-C3230 Quad-Core 64-bit, RAM 1 GB, penyimpanan internal 4 GB, dan sistem operasi Android versi 5.0.2 (Lollipop). Aplikasi juga dapat berjalan pada perangkat ini. Namun pada beberapa bagian terdapat sedikit perbedaan tampilan dengan perangkat yang digunakan sebelumnya.

Berdasarkan hasil pengujian pada beberapa perangkat dengan berbagai spesifikasi yang berbeda-beda dapat disimpulkan secara umum aplikasi memiliki kompatibilitas luas karena mampu berjalan pada berbagai perangkat smartphome dengan spesifikasi yang berbeda-beda.

### 4.5.3 Pengujian Aplikasi pada Pengguna (*User Acceptance Test*)

Pengujian aplikasi kepada pengguna diajukan kepada 30 responden yaitu pengguna atau calon pengguna bus Trans Jogja (belum pernah menggunakan bus Trans Jogja) dengan cara meminta responden mengoperasikan aplikasi serta mencoba menu-menu yang terdapat pada aplikasi dengan menggunakan pada ponsel milik penulis atau jika berkenan memasang terlebih dahulu aplikasi pada ponsel milik responden. Setelah itu, responden diberikan kuesioner berisi delapan buah pertanyaan pilihan ganda (STS = Sangat Tidak Setuju, TS = Tidak Setuju, N = Normal, S = Setuju, SS = Sangat Setuju) mengenai aplikasi rekomendasi jalur bus Trans Jogja. Gambar 4.19 Merupakan kuesioner yang digunakan untuk melakukan pengujian.

Nama : \_\_\_\_\_ Pekerjaan: \_\_\_\_\_

- **Pernah menggunakan bus Trans Jogja :** Pernah/ Belum (Lingkari salah satu)
- **Saya mencoba aplikasi dengan menginstallnya pada smartphone saya:**
  - a. Ya\*
  - b. Tidak, Tidak, Saya mencoba langsung padasmartphone pengembang

**(lingkari salah satu)**

**\* Jika mencoba aplikasi dengan menginstallnya pada smartphone pribadi**

- Merk – Seri Smartphone (contoh: Samsung – Galaxy s8) : \_\_\_\_\_
- Versi Android pada Smartphone : \_\_\_\_\_

No	Pertanyaan	Jawaban berdasar skala				
		STS	TS	N	S	SS
1.	Aplikasi berjalan dengan baik, lancar					
2.	Aplikasi mudah untuk dipahami					
3.	Tata letak (Menu, Tombol, icon, dsb.) dan tampilan antarmuka aplikasi mudah digunakan.					
4.	Aplikasi dapat memberikan rekomendasi dan panduan jalur bus Trans Jogja yang digunakan, dari suatu lokasi asal menuju lokasi tujuan.					
5.	Aplikasi dapat memberikan informasi lain yang masih terkait dengan Trans Jogja (Informasi sekilas tentang Trans Jogja, informasi rute bus, informasi tentang halte bus) <u>dengan baik</u>					
6.	Informasi yang disajikan pada aplikasi mudah untuk saya pahami					
7.	Pengalaman saya mengenai Trans Jogja bertambah setelah menggunakan Aplikasi ini					
8.	Aplikasi ini layak untuk dijadikan salah satu panduan praktis penggunaan bus Trans Jogja					

Gambar 4.19 Kuesioner untuk pengujian kepada pengguna

Perhitungan nilai tiap pertanyaan dihitung dengan cara berikut:

$$\text{Nilai tiap pertanyaan} = \frac{\text{Jumlah total poin per soal}}{\text{Jumlah responden}}$$

Jumlah total poin per soal dihitung dengan cara menjumlahkan total poin per skala jawaban ((total poin STS) + (total poin TS) + (total poin N) + (total poin S) + (total poin SS)).

Total poin STS dihitung dengan cara mengalikan total jawaban STS dikalikan 1, total poin TS dihitung cara mengalikan total jawaban TS dikalikan 2, total poin N dihitung cara mengalikan total jawaban N dikalikan 3, total poin S dihitung cara mengalikan total jawaban S dikalikan 4, dan total poin SS dihitung cara mengalikan total jawaban SS dikalikan 5. Setelah itu total poin per soal dibagi dengan jumlah responden yaitu 30 untuk mendapatkan nilai dari tiap soal.

Dari 30 responden, sebanyak 20 responden mencoba aplikasi dengan memasang aplikasi di smartphone milik responden terlebih dahulu, sedangkan jumlah responden yang mencoba aplikasi pada smartphone milik penulis sebanyak 10 responden. Dari 30 responden, sebanyak 24 responden pernah menggunakan Trans Jogja dan sisanya yaitu sebanyak 6 responden belum pernah menggunakan Trans Jogja. Berdasarkan 20 responden yang memasang aplikasi pada smartphone miliknya Tabel 4.16 adalah hasil perhitungan untuk jenis versi Android yang digunakan oleh 20 responden tersebut, sedangkan Pada Tabel 4.17 merupakan hasil rekapitulasi frekuensi jawaban 30 responden serta perhitungan nilai dari tiap - tiap pertanyaan kuesioner bagian utama.

Tabel 4.16 Hasil perhitungan versi Android yang digunakan pada smartphone responden

	Versi Android			
	KitKat	Lollipop	Marshmallow	Nougat
Jumlah	4	7	6	3
Presentase	20%	35%	30%	15%

Tabel 4.17 Hasil perhitungan frekuensi jawaban dan nilai tiap pertanyaan

Pertanyaan Nomor	Jawaban					Nilai
	STS	TS	N	S	SS	
1	0	0	4	16	10	4,200
2	0	0	4	20	6	4,067
3	0	0	8	19	3	3,833
4	0	0	1	9	20	4,633
5	0	0	2	15	13	4,367
6	0	0	3	23	4	4,033
7	0	0	2	15	13	4,367
8	0	0	2	4	24	4,733
Total	0	0	26	121	93	4,279

Dari hasil pengujian aplikasi pada pengguna terhadap 30 responden melalui kuesioner yang telah dilakukan dapat disimpulkan bahwa secara umum responden setuju Aplikasi Rekomendasi Jalur Bus Trans Jogja ini dapat dijadikan sebagai panduan penggunaan Trans Jogja dengan nilai rata – rata penilaian 4,279 dari delapan pertanyaan pada kuesioner. Responden setuju bahwa aplikasi tersebut layak untuk dijadikan sebagai salah satu panduan bus Trans Jogja terbukti 80% responden sangat setuju dan 13,3% setuju. Secara umum responden setuju aplikasi mampu memberikan rekomendasi pilihan jalur bus yang dapat dipilih oleh pengguna dengan 53% responden setuju dan 33,3% sangat setuju. Sebanyak 66,7% responden setuju dan 20% sangat setuju bahwa aplikasi juga dapat memberikan informasi lain seputar Trans Jogja (informasi bus Trans, informasi halte bus Tans Jogja, informasi sekilas Trans Jogja) dan setuju bahwa informasi tersebut ditampilkan dengan tata letak yang cukup mudah digunakan yaitu 63,3% responden setuju dan 10% sangat setuju sehingga pengetahuan responden mengenai Trans Jogja menjadi bertambah terlihat bahwa 50% responden setuju dan 43,3% sangat setuju.

Berdasarkan dari rangkaian hasil pengujian menunjukkan bahwa aplikasi yang dikembangkan dapat membantu pengguna yaitu masyarakat umum dalam menggunakan bus Trans Jogja. Aplikasi mampu memberikan rekomendasi – rekomendasi panduan jalur bus yang dapat dipilih oleh pengguna ketika hendak pergi dari suatu lokasi asal menuju ke lokasi tujuan. Hal tersebut terbukti sesuai dengan hasil pengujian fungsionalitas maupun dari pengujian aplikasi pada pengguna yang menyatakan bahwa 9 orang atau sekitar hampir 30 % orang setuju dan sekitar 66,7% atau 20 orang sangat setuju dengan pernyataan tersebut. Rekomendasi – rekomendasi yang diberikan dapat dipahami oleh pengguna dan mampu menambah pengetahuan tentang Trans Jogja.

Aplikasi dapat berjalan dengan baik dan telah sesuai dengan kebutuhan berdasarkan dari hasil pengujian fungsionalitas yang menunjukkan bahwa dari semua skenario pengujian dapat memberikan hasil yang sesuai seperti yang diharapkan. Selain itu aplikasi yang dikembangkan juga mempunyai kompatibilitas luas terbukti dari hasil pengujian kompatibilitas yang menunjukkan bahwa aplikasi dapat berjalan pada beberapa perangkat dengan spesifikasi yang berbeda – beda dan mampu berjalan dengan baik pula pada berbagai versi android seperti yang juga telah terpasang dan dijalankan pada 20 smarphone responden yang juga turut memasang dan melakukan pengujian aplikasi.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Setelah dilakukan penelitian, perancangan, implementasi, hingga pengujian pada tugas akhir ini, maka telah berhasil dibangun aplikasi rekomendasi Trans Jogja. Aplikasi dibangun dengan memanfaatkan Google Maps API dan Google Places API dan menerapkan konsep graf dalam salah satu pencarian jalur bus serta dikembangkan untuk perangkat berbasis Android. Aplikasi dapat membantu pengguna Trans Jogja mendapatkan rekomendasi panduan jalur bus yang bisa dipilih dari lokasi asal ke lokasi tujuan beserta informasi jalur dan halte bus.

Selain digunakan untuk membantu mencari rekomendasi panduan jalur bus, pengguna juga bisa memanfaatkan aplikasi ini untuk mendapatkan informasi – informasi lain seperti melihat halte terdekat dari lokasi asal dan lokasi tujuan yang telah ditentukan, mencari informasi lain mengenai jalur bus Trans Jogja (Jalur 1A hingga jalur 11) beserta rute perjalanannya, mencari informasi halte Trans Jogja seperti alamat halte dan bus yang tersedia pada halte tersebut, mencari tahu informasi sekilas tentang Trans Jogja, informasi jenis tiket, informasi aturan naik Trans Jogja, informasi aturan transit, serta informasi mengenai pembelian dan isi ulang kartu Trans Jogja sehingga aplikasi ini layak dijadikan sebagai salah satu panduan penggunaan Trans Jogja. Aplikasi ini memiliki kompatibilitas yang luas karena dapat berjalan dengan baik pada berbagai perangkat Android yang memiliki spesifikasi yang berbeda – beda.

Berdasarkan hasil kuesioner yang telah dilakukan kepada 30 responden sebanyak 80% responden sangat setuju bahwa Aplikasi Rekomendasi jalur bus Trans Jogja ini layak untuk dijadikan sebagai salah satu panduan Trans Jogja.

#### **5.2 Saran**

Aplikasi rekomendasi jalur bus Trans Jogja yang telah dibangun ini masih terdapat beberapa kekurangan serta masih terbuka untuk bisa dikembangkan lagi. Melalui hasil pengujian, diskusi dengan dosen dosen pembimbing, umpan balik dari beberapa responden, berikut ini adalah saran dan masukan untuk pengembangan aplikasi kedepannya, yaitu:

- a. Mencari jalur – jalur lain yang bisa ditempuh kemudian dibandingkan dengan jalur – jalur yang sudah ditemukan sebagai tambahan rekomendasi jalur karena masih dimungkinkan

adanya pilihan jalur bus lain yang bisa lebih optimal yaitu lebih sedikit transit dengan selisih jarak yang tidak terlalu jauh dengan jarak terpendek.

- b. Menambah fitur pengaturan – pengaturan tertentu dalam mencari panduan misalkan hanya memilih beberapa halte tertentu, rekomendasi diurutkan berdasarkan jarak terpendek yang ditempuh atau diurutkan dari jarak terpendek jalan kaki dari lokasi asal maupun tujuan ke halte, membatasi hanya menampilkan rekomendasi dengan jumlah tertentu. Dengan pengaturan – pengaturan tersebut akan menambah tingkat kesesuaian dengan kemauan pengguna dalam mendapatkan rekomendasi pilihan jalur.
- c. Memperbaiki tata letak antarmuka di bagian rute perjalanan bus pada peta dengan memperlihatkan rute yang dilalui sebenarnya agar lebih memperlihatkan gambaran rute perjalanan yang akan dilalui.
- d. Mengembangkan aplikasi yang mendukung multibahasa (misalkan tersedia untuk Bahasa Indonesia, English) agar bisa lebih dipahami oleh pengguna yang tidak terbiasa dan paham dengan Bahasa Indonesia.
- e. Terhubung dengan media sosial untuk berbagi mengenai Trans Jogja kepada teman – teman di media sosial.
- f. Memberikan tanda khusus misalkan warna berbeda untuk rekomendasi terbaik untuk memudahkan pengguna dalam menemukan rekomendasi yang dianggap terbaik.
- g. Menambahkan perkiraan waktu tempuh pada rekomendasi atau tiap perjalanan yang dapat digunakan pula sebagai bahan pertimbangan dalam memilih rekomendasi.

## DAFTAR PUSTAKA

- Android (2017). *Android*. Diakses pada 5 September 2017, dari <https://www.android.com/>
- Dishub DIY (2017). *Trans Jogja | DISHUB DI Yogyakarta*. Diakses pada 3 September 2017, dari <http://dishub.jogjaprov.go.id/produk/trans-jogja>
- Google Developers A (2017). *Google Maps API*. Diakses pada 5 September 2017, dari <https://developers.google.com/maps>
- Google Developers B (2017). *Google Places API*. Diakses pada 5 September 2017, dari <https://developers.google.com/places>
- Google Play A (2012). *Trans Jogja*. Diakses pada 29 Januari 2018, dari <https://play.google.com/store/apps/details?id=id.web.yudanta.tejo>
- Google Play B (2017). *Trans Jogja (guide)*. Diakses pada 29 Januari 2018, dari <https://play.google.com/store/apps/details?id=com.arman.guidetransjogja>
- Google Play C (2013). *iTransJogja*. Diakses pada 29 Januari 2018, dari <https://play.google.com/store/apps/details?id=com.iTransJogja>
- Google Play D (2016). *0274*. Diakses pada 29 Januari 2018, dari [https://play.google.com/store/apps/details?id=plbtw.plbtw\\_0274](https://play.google.com/store/apps/details?id=plbtw.plbtw_0274)
- Handaka, M. S. (2010). *Perbandingan Algoritma Dijkstra (Greedy), Bellman-Ford (BFS-DFS), dan Floyd-Warshall (Dynamic Programming) dalam Pengaplikasian Lintasan Terpendek pada Link-State Routing Protocol*. Bandung: Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung ITB.
- International Data Corporation (2017). *Smartphone OS Market Share Q3 2017*. Diakses pada 7 September 2017, dari <http://www.idc.com/promo/smartphone-market-share/os>
- Jungnickel, D. (2008). *Graph, Network, and Algorithms*. Berlin:Springer.
- Munir, R. (2005). *Diktat Kuliah IF2251 Strategi Algoritmik*. Program Studi Informatika Sekolah Teknik Elektro dan Informatika ITB.
- PT AMI (2016). *Website Resmi PT. Anindya Mitra Internasional*. Diakses pada 3 September 2017, dari <http://www.anindya.co.id/index.php/unit/detail/18>
- Search Engine Land (2016). *New Survey Says Google Maps Favored by Nearly 70 Percent of iPhone Users*. Diakses pada 29 Januari 2018, dari <https://searchengineland.com/new-survey-says-google-maps-favored-nearly-70-percent-iphone-users-251955>
- Steiniger, S., Neun, M., Edwardes, A., & Lenz, B. (2008). Foundations of LBS. *CartouCHE-Cartography for Swiss Higher Education. Obtido em, 20, 2010.*

- Sukrisno, A.T. (2010). *Perancangan Prototype Dinamyc Exit Sign dengan Mengembangkan Metode Floyd-Wharshall Algorithm pada Perencanaan Proses Evakuasi Gedung Bertingkat*. Surabaya: ITS
- Virrantaus, K., Markkula, J., Garmash, A., Terziyan, V., Veijalainen, J., Katanosov, A., & Tirri, H. (2001). Developing GIS-supported location-based services. In *Web information systems engineering, 2001. Proceedings of the Second International Conference on* (Vol. 2, halaman. 66-75). IEEE.

## LAMPIRAN

## KUESIONER PENGUJIAN APLIKASI PANDUAN JALUR BUS TRANS JOGJA BERBASIS ANDROID

Nama : \_\_\_\_\_ Pekerjaan: \_\_\_\_\_

- Pernah menggunakan bus Trans Jogja : Pernah/ Belum (Lingkari salah satu)

- Saya mencoba aplikasi dengan menginstallnya pada smartphone saya:

a. Ya\*

b. Tidak, Tidak, Saya mencoba langsung pada smartphone pengembang  
(lingkari salah satu)

\* Jika mencoba aplikasi dengan menginstallnya pada smartphone pribadi

• Merk – Seri Smartphone (contoh: Samsung – Galaxy s8): \_\_\_\_\_

• Versi Android pada Smartphone : \_\_\_\_\_

No	Pertanyaan	Jawaban berdasar skala				
		STS	TS	N	S	SS
1.	Aplikasi berjalan dengan baik, lancar					
2.	Aplikasi mudah untuk dipahami					
3.	Tata letak (Menu, Tombol, icon, dsb.) dan tampilan antarmuka aplikasi mudah digunakan.					
4.	Aplikasi dapat memberikan rekomendasi dan panduan jalur bus Trans Jogja yang digunakan, dari suatu lokasi asal menuju lokasi tujuan.					
5.	Aplikasi dapat memberikan informasi lain yang masih terkait dengan Trans Jogja (Informasi sekilas tentang Trans Jogja, informasi rute bus, informasi tentang halte bus) <u>dengan baik</u>					
6.	Informasi yang disajikan pada aplikasi mudah untuk saya pahami					
7.	Pengetahuan saya mengenai Trans Jogja bertambah setelah menggunakan Aplikasi ini					
8.	Aplikasi ini layak untuk dijadikan salah satu panduan praktis penggunaan bus Trans Jogja					

STS = Sangat Tidak Setuju

TS = Tidak Setuju

N = Normal

S = Setuju

SS = Sangat Setuju