

**APLIKASI PENCARIAN MINIMARKET MENGGUNAKAN  
METODE HAVERSINE FORMULA UNTUK  
MENENTUKAN JARAK TERDEKAT**

**SKRIPSI**

**Oleh:**

**ALFAN KURNIAWAN**

**NIM. 08650044**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2014**

**APLIKASI PENCARIAN MINIMARKET MENGGUNAKAN  
METODE HAVERSINE FORMULA UNTUK  
MENENTUKAN JARAK TERDEKAT**

**SKRIPSI**

**Diajukan Kepada :**

**Dekan Fakultas Sains dan Teknologi**

**Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang  
untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:**

**ALFAN KURNIAWAN**

**NIM. 08650044**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2014**

**HALAMAN PERSETUJUAN**

**APLIKASI PENCARIAN MINIMARKET MENGGUNAKAN  
METODE HEVERSSINE FORMULA UNTUK  
MENENTUKAN JARAK TERDEKAT**

**SKRIPSI**

Oleh:

**ALFAN KURNIAWAN**

**NIM. 08650044**

Telah Diperiksa dan Disetujui oleh:

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Ririen Kusumawati, M.Kom**  
**NIP. 19720309 200501 2 002**

**Ach. Nashicuddin, M.A**  
**NIP. 19730705 200003 1 002**

**Tanggal 14 Maret 2014**

**Mengetahui,**  
**Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdiان**  
**NIP. 19740424 200901 1 008**

## HALAMAN PENGESAHAN

### APLIKASI Pencarian Minimarket Menggunakan Metode Heversine Formula Untuk Menentukan Jarak Terdekat

#### SKRIPSI

Oleh :

**ALFAN KURNIAWAN**  
**NIM. 08650044**

Telah Dipertahankan Di Depan Dewan Penguji Skripsi  
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal, 14 Maret 2014

Susunan Dewan Penguji:	Tanda Tangan
1. Penguji Utama : A'la Syauqi, M.Kom ( ) NIP. 19771201 200801 1 007	
2. Ketua : Fatchurrochman, M.Kom ( ) NIP. 19700731 200501 1 002	
3. Sekretaris : Ririen Kusumawati, M.Kom ( ) NIP. 19720309 200501 2 002	
4. Anggota : Ach. Nashicuddin, M.A ( ) NIP. 19730705 200003 1 002	

Mengetahui dan Mengesahkan  
Ketua Jurusan Teknik Informatika

**Dr. Cahyo Crysdiان**  
**NIP. 19740424 200901 1 008**

## **SURAT PERNYATAAN ORISINALITAS PENELITIAN**

Saya yang bertanda tangan di bawah ini :

Nama : Alfian Kurniawan

NIM : 08650044

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Penelitian : APLIKASI PENCARIAN MINIMARKET MENGGUNAKAN  
METODE HAVERSINE FORMULA UNTUK  
MENENTUKAN JARAK TERDEKAT

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur jiplakan, maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 14 Maret 2014  
Penulis

Alfian Kurniawan  
NIM. 08650044

## MOTTO

يَرْفَعُ اللَّهُ الَّذِينَ ءَامَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ ﴿١١﴾

“Niscaya Allah akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat. dan Allah Maha mengetahui apa yang kamu kerjakan”. (QS. Al-Mujaadilah:11)

“Sesungguhnya ilmu pengetahuan menempatkan orang nya kepada kedudukan terhormat dan mulia (tinggi) . Ilmu pengetahuan adalah keindahan bagi ahlinya di dunia dan di akhirat “ (H.R Ar- Rabii’).

“ Barang siapa merintis jalan mencari ilmu maka Alloh akan memudahkan baginya jalan ke surga “ (H.R Muslim).

يُؤْتِي الْحِكْمَةَ مَنْ يَشَاءُ ۚ وَمَنْ يُؤْتَ الْحِكْمَةَ فَقَدْ أُوتِيَ خَيْرًا كَثِيرًا ۗ وَمَا

يَذَّكَّرُ إِلَّا أُولُو الْأَلْبَابِ ﴿٢٦٩﴾

“Allah menganugerahkan Al Hikmah (ilmu yang berguna) kepada siapa yang dikehendaki-Nya. dan Barangsiapa yang dianugerahi hikmah, ia benar-benar telah dianugerahi karunia yang banyak. dan hanya orang-orang yang berakallah yang dapat mengambil pelajaran (dari firman Allah)”. (QS. Al-Baqarah : 269)

## PERSEMBAHAN



*Yaa Allah.....*

*Terima kasih atas nikmat dan rahmat-Mu yang agung ini.*

*Hari ini hamba bahagia. Sebuah perjalanan panjang dan gelap telah Kau berikan secercah cahaya terang. Meskipun hari esok penuh teka-teki dan tanda tanya yang aku sendiri belum tahu pasti jawabanya. Di tengah malam aku bersujud, kupinta kepada-mu di saat aku kehilangan arah, kumohon petunjuk-mu. Aku sering tersandung, terjatuh, terluka dan terkadang harus kutelan antara keringat dan air mata. Namun aku tak pernah takut, aku takkan pernah menyerah karena aku tak mau kalah, Aku akan terus melangkah berusaha dan berdo'a tanpa mengenal putus asa.*

*Ku persembahkan karya tulis ini untuk*

*Ayahanda Imam Syuhadi dan Ibunda Siti Musyarofah tercinta yang senantiasa bersujud dan bermunajat kepada Allah SWT untuk kebaikan dan kesuksesan putra tercintanya, serta senantiasa mendukung, memotivasi dan memberiku inspirasi untuk terus berjuang. Karya tulis ini ku persembahkan sebagai jawaban atas kepercayaan yang telah berikan oleh kedua orang tuaku serta perwujudan*

*bhaktiku kepada kedua orang tuaku*

*Spesial Thank's untuk*

*teman-temanku tercinta serta teman-teman seperjuangan jurusan Teknik Informatika angkatan 2008 yang selalu memberikan suntikan motivasi dan semangat dalam menjalani hidup untuk tidak pantang menyerah*

## KATA PENGANTAR

*Assalamu'alaikum Wr. Wb.*

Puji syukur alhamdulillah penulis panjatkan kehadirat Allah SWT atas berkat, rahmat, taufik dan hidayah-Nya, penyusunan skripsi yang berjudul “Aplikasi Pencarian Minimarket Menggunakan Metode Haversine Formula Untuk Menentukan Jarak Terdekat” dapat diselesaikan dengan baik.

Penulis menyadari bahwa dalam proses penulisan skripsi ini banyak mengalami kendala, namun berkat bantuan, bimbingan, kerjasama dari berbagai pihak dan berkah dari Allah SWT sehingga kendala-kendala yang dihadapi tersebut dapat diatasi. Untuk itu penulis menyampaikan ucapan terima kasih dan penghargaan kepada Ibu Ririen Kusumawati, M.Kom selaku pembimbing I dan Bapak Ach. Nashicuddin, M.A selaku pembimbing II dengan sabar, tekun, tulus dan ikhlas meluangkan waktu, tenaga dan pikiran memberikan bimbingan, motivasi, arahan, dan saran-saran yang sangat berharga kepada penulis selama menyusun skripsi.

Selanjutnya ucapan terima kasih penulis sampaikan pula kepada:

1. Prof. Dr. H. Mudjia Rahardjo, M.Si selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr. Drh. Bayyinatul Muchtaromah, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdiان selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

4. Fatchurrohman, M.Kom, selaku Dosen Wali, yang membimbing perencanaan studi selama menuntut ilmu di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
5. Imam Syuhadi dan Siti Musyarofah selaku Ayahanda dan Ibunda tercinta yang dengan sepenuh hati memberikan dukungan moral maupun spiritual sehingga penulisan tugas akhir ini dapat terselesaikan.
6. Dina Fitriana, S.E yang memberikan motivasi dan semangat.
7. Teman-teman Jurusan Teknik Informatika angkatan 2008 dan teman-teman kuliah UIN Maulana Malik Ibrahim Malang maupun kampus lainnya.
8. Teman-teman kost Joyo Raharjo saya ucapkan terima kasih karena telah memberikan dukungan
9. Dan semua pihak yang telah membantu dalam menyelesaikan skripsi ini.

Penulis menyadari bahwa skripsi ini masih terdapat banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun agar skripsi ini dapat lebih baik lagi. Akhir kata penulis berharap kerangka acuan skripsi ini dapat memberikan wawasan dan pengetahuan kepada para pembaca pada umumnya dan pada penulis pada khususnya. Amin Ya Rabbal Alamin.

*Wassalamu'alaikum Wr. Wb.*

Malang, 14 Maret 2014

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	i
<b>HALAMAN PENGAJUAN</b> .....	ii
<b>HALAMAN PERSETUJUAN</b> .....	iii
<b>HALAMAN PENGESAHAN</b> .....	iv
<b>SURAT PERNYATAAN ORISINALITAS PENELITIAN</b> .....	v
<b>MOTTO</b> .....	vi
<b>HALAMAN PERSEMBAHAN</b> .....	vii
<b>KATA PENGANTAR</b> .....	viii
<b>DAFTAR ISI</b> .....	x
<b>DAFTAR TABEL</b> .....	xiv
<b>DAFTAR GAMBAR</b> .....	xv
<b>DAFTAR LAMPIRAN</b> .....	xvii
<b>ABSTRAK</b> .....	xviii
<b>ABSTRACT</b> .....	xix
<b>BAB I PENDAHULUAN</b> .....	1
<b>1.1 Latar Belakang</b> .....	1
<b>1.2 Rumusan Masalah</b> .....	7
<b>1.3 Batasan Masalah</b> .....	7
<b>1.4 Tujuan Penelitian</b> .....	8
<b>1.5 Manfaat Penelitian</b> .....	8
<b>1.6 Sistematika Pembahasan</b> .....	8
<b>BAB II TINJAUAN PUSTAKA</b> .....	10
<b>2.1 Ayat Al-Qur'an Terkait Anjuran Allah Untuk Mencari Kemudahan</b> .....	10
<b>2.2 Minimarket</b> .....	15
<b>2.3 Haversine Formula</b> .....	16
<b>2.4 JSON</b> .....	19
<b>2.5 MySQL</b> .....	23
<b>2.6 Google Maps</b> .....	23
<b>2.7 Layanan Berbasis Lokasi (<i>Location Based Service</i>)</b> .....	27
<b>2.7.1 Piranti Mobile</b> .....	28

2.7.2 Jaringan Komunikasi.....	28
2.7.3 Komponen <i>Positioning</i> (Penunjuk Posisi) .....	28
2.7.4 Penyedia Layanan Aplikasi .....	29
2.7.5 Penyedia Data dan Konten .....	29
2.8 <i>Global Positioning System</i> (GPS) .....	30
2.9 <i>Android</i> .....	32
<b>BAB III ANALISIS DAN PERANCANGAN SISTEM</b> .....	38
3.1 Peta .....	38
3.2 Cara Kerja Sistem GPS .....	39
3.3 Desain Sistem .....	40
3.3.1 Arsitektur Sistem .....	40
3.3.2 Rancangan Sistem .....	42
3.4 UML ( <i>Unified Modelling Language</i> ) .....	45
3.4.1 <i>Use Case</i> .....	45
3.5 <i>Activity Diagram</i> .....	49
3.5.1 <i>Activity Diagram</i> Proses Pencarian Lokasi Pengguna .....	49
3.5.2 <i>Activity Diagram Client Request Map</i> .....	49
3.5.3 <i>Activity Diagram Koneksi HTTP</i> .....	50
3.5.4 <i>Activity Diagram Request Data Minimarket</i> .....	51
3.5.5 <i>Activity Diagram Perhitungan Haversine Formula</i> .....	52
3.5.6 <i>Activity Diagram Menejemen Data Server</i> .....	53
3.6 <i>Sequence Diagram</i> .....	54
3.6.1 <i>Sequence Diagram Client Request Lokasi</i> .....	54
3.6.2 <i>Sequence Diagram Client Request Map</i> .....	55
3.6.3 <i>Sequence Diagram Koneksi HTTP</i> .....	55
3.6.4 <i>Sequence Diagram Request Data Minimarket</i> .....	56
3.6.5 <i>Sequence Diagram Perhitungan Haversine Formula</i> .....	56
3.6.6 <i>Sequence Diagram Server Application Subsystem</i> .....	57
3.7 <i>Class Diagram</i> .....	57
3.8 <i>Flowchart System</i> .....	62
3.4 Perancangan <i>Desain Database</i> .....	64
3.4.1 <i>Tabel Database Minimarket</i> .....	64
3.9 <i>Desain Interface</i> .....	65

3.10.1	<i>Desain Form Awal</i>	65
3.10.2	<i>Desain Form Tampilan Utama</i>	66
3.10.3	<i>Desain Form Menu Lokasiku</i>	66
3.10.4	<i>Desain Form Tipe Peta</i>	67
3.10.5	<i>Desain Form Seting Radius</i>	68
3.10.6	<i>Desain Form Menu Refresh Map</i>	68
3.10.7	<i>Desain Form Menu Tentang</i>	69
3.10.8	Tombol <i>Zoom In</i> dan <i>Zoom Out</i>	70
3.10.9	<i>Desain Panel Informasi Radius</i>	70
3.10	Uji Kelayakan Aplikasi	71
3.11	Uji Kelayakan Aplikasi Berdasarkan Kuisisioner	72
<b>BAB IV IMPLEMENTASI HASIL DAN PEMBAHASAN</b>		74
4.1	Sumber Data	74
4.2	Ruang Lingkup Implementasi	74
4.2.1	Ruang Lingkup Perangkat Keras	75
4.2.2	Ruang Lingkup Perangkat Lunak	75
4.3	Implementasi Sistem	76
4.3.1	Tahap <i>Request</i> Lokasi	77
4.3.2	Tahap Menampilkan <i>Map</i>	78
4.3.3	Tahap <i>Request Data Minimarket</i>	78
4.3.4	Tahap Kalkulasi <i>Haversine Formula</i>	79
4.3.5	Tahap Membuat <i>Marker/Direction</i> di <i>Map</i>	80
4.3.6	Tahap Menampilkan Jarak Terdekat	80
4.3.7	<i>Source Code</i> Koneksi Dengan <i>Database</i>	81
4.4	Implementasi Antarmuka	82
4.4.1	<i>Splash Screen</i>	82
4.4.2	Form Tampilan Utama	83
4.4.3	Form Tampilan Peta	84
4.4.4	Menu Lokasiku	85
4.4.5	Menu Tipe Peta	86
4.4.6	Menu Seting Radius	87
4.4.7	Form Menu Tentang	88
4.4.8	Menu <i>Refresh Map</i>	89

<b>4.4.9</b> Tombol <i>Zoom In</i> dan <i>Zoom Out</i> .....	89
<b>4.4</b> Implementasi Rumus <i>Haversine Formula</i> .....	90
<b>4.5</b> Evaluasi dan Hasil Analisis Pengujian .....	92
<b>4.5.1</b> Pengujian Menemukan & Menampilkan Lokasi User .....	92
<b>4.5.2</b> Pengujian Menampilkan Icon <i>Minimarket</i> Berdasarkan Tipe Peta .....	93
<b>4.5.3</b> Pengujian Menampilkan Icon <i>Minimarket</i> Berdasarkan Radius .....	94
<b>4.5.4</b> Pengujian Menampilkan <i>Minimarket</i> Dengan Jarak Terdekat Berdasarkan Satu Lokasi <i>User</i> .....	95
<b>4.5.5</b> Pengujian Menampilkan <i>Minimarket</i> Terdekat Berdasarkan Lokasi User Yang Variatif Dan Berapa Waktu Yang Dibutuhkan .....	97
<b>4.5.6</b> Pengujian Perangkat Lunak .....	108
<b>4.5.7</b> Hasil Kuisisioner Uji Kelayakan Aplikasi .....	110
<b>4.6</b> Hubungan Aplikasi Pencarian <i>Minimarket</i> Dengan Perspektif Islam .....	117
<b>BAB V PENUTUP</b> .....	119
<b>5.1</b> Kesimpulan .....	119
<b>5.2</b> Saran .....	120
<b>DAFTAR PUSTAKA</b> .....	121
<b>LAMPIRAN</b> .....	123

## DAFTAR TABEL

<b>Tabel 3.1</b> Penanganan <i>Service Client Server</i> .....	41
<b>Tabel 3.2</b> Perancangan <i>Database</i> Aplikasi Pencarian <i>Minimarket</i> Terdekat ..	65
<b>Tabel 4.1</b> Hasil Pengujian Aplikasi Berdasarkan Satu Lokasi .....	96
<b>Tabel 4.2</b> Hasil Pengujian Menampilkan <i>Minimarket</i> Terdekat Berdasarkan Lokasi <i>User</i> Yang Variatif Dan Berapa Waktu Yang Dibutuhkan .....	107
<b>Tabel 4.3</b> Pengujian Aplikasi Pada Berbagai <i>Device</i> .....	109
<b>Tabel 4.4</b> Rekapitulasi Hasil Kuisisioner .....	110
<b>Tabel 4.5</b> Hasil Perhitungan Kuesioner Dengan Skala Likert.....	116

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Segitiga Bola Diselesaikan Dengan Hukum <i>Haversine</i> <i>Formula</i> .....	18
<b>Gambar 2.2</b> <i>Object</i> Dalam JSON .....	21
<b>Gambar 2.3</b> <i>Array</i> Dalam JSON .....	21
<b>Gambar 2.4</b> <i>Value</i> Atau Nilai Dalam Format JSON .....	21
<b>Gambar 2.5</b> Tipe Data <i>String</i> Pada Format JSON.....	22
<b>Gambar 2.6</b> <i>Number</i> Atau Angka Dalam Format JSON .....	22
<b>Gambar 2.7</b> Komponen Pendukung Utama Teknologi LBS .....	30
<b>Gambar 2.8</b> Diagram Komponen Utama <i>Android</i> .....	34
<b>Gambar 3.1</b> Tampilan Peta Malang.....	38
<b>Gambar 3.2</b> Arsitektur Sistem .....	40
<b>Gambar 3.3</b> Rancangan Sistem .....	42
<b>Gambar 3.4</b> <i>Use Case</i> Aplikasi Pencarian <i>Minimarket</i> .....	46
<b>Gambar 3.5</b> <i>Activity Diagram</i> <i>Client Request</i> Lokasi .....	49
<b>Gambar 3.6</b> <i>Activity Diagram</i> <i>Client Request</i> <i>Map</i> .....	50
<b>Gambar 3.7</b> <i>Activity Diagram</i> Koneksi <i>HTTP</i> .....	51
<b>Gambar 3.8</b> <i>Activity Diagram</i> <i>Request</i> Data <i>Minimarket</i> .....	52
<b>Gambar 3.9</b> <i>Activity Diagram</i> Proses Perhitungan <i>Haversine</i> <i>Formula</i> .....	53
<b>Gambar 3.10</b> <i>Activity Diagram</i> Menejemen Data <i>Server</i> .....	54
<b>Gambar 3.11</b> <i>Sequence Diagram</i> <i>Client Request</i> Lokasi .....	54
<b>Gambar 3.12</b> <i>Sequence Diagram</i> <i>Client Request</i> <i>Map</i> .....	55
<b>Gambar 3.13</b> <i>Sequence Diagram</i> Koneksi <i>HTTP</i> .....	55
<b>Gambar 3.14</b> <i>Sequence Diagram</i> <i>Request</i> Data <i>Minimarket</i> .....	56
<b>Gambar 3.15</b> <i>Sequence Diagram</i> Proses Perhitungan <i>Haversine</i> <i>Formula</i> ....	56
<b>Gambar 3.16</b> <i>Sequence Diagram</i> <i>Server</i> <i>Application</i> <i>Subsystem</i> .....	57
<b>Gambar 3.17</b> <i>Class Diagram</i> Aplikasi Pencarian <i>Minimarket</i> .....	59
<b>Gambar 3.18</b> <i>Flowchart</i> <i>Application</i> <i>Subsystem</i> .....	62
<b>Gambar 3.19</b> <i>Form</i> Tampilan Awal .....	65
<b>Gambar 3.20</b> <i>Form</i> Tampilan Utama .....	66
<b>Gambar 3.21</b> <i>Form</i> Tampilan <i>Menu</i> Lokasiku .....	67
<b>Gambar 3.22</b> <i>Form</i> Tipe Peta .....	67
<b>Gambar 3.23</b> <i>Form</i> <i>Seting</i> <i>Radius</i> .....	68

<b>Gambar 3.24</b> <i>Form Menu Refresh Map</i> .....	69
<b>Gambar 3.25</b> <i>Form Tentang</i> .....	69
<b>Gambar 3.26</b> <i>Tombol Zoom In Dan Zoom Out</i> .....	70
<b>Gambar 3.27</b> <i>Tampilan Panel Informasi Radius</i> .....	71
<b>Gambar 4.1</b> <i>Samsung Galaxy SIII Mini</i> .....	76
<b>Gambar 4.2</b> <i>Splash Screen Aplikasi</i> .....	83
<b>Gambar 4.3</b> <i>Tampilan Utama Aplikasi</i> .....	84
<b>Gambar 4.4</b> <i>Form Tampilan Peta</i> .....	85
<b>Gambar 4.5</b> <i>Menu Lokasiku</i> .....	85
<b>Gambar 4.6</b> <i>Menu Tipe Peta</i> .....	86
<b>Gambar 4.7</b> <i>Form Menu Tipe Pilihan Peta</i> .....	86
<b>Gambar 4.8</b> <i>Menu Seting Radius</i> .....	87
<b>Gambar 4.9</b> <i>Menu Pilihan Seting Radius</i> .....	87
<b>Gambar 4.10</b> <i>Menu Tentang</i> .....	88
<b>Gambar 4.11</b> <i>Form Menu Tentang</i> .....	88
<b>Gambar 4.12</b> <i>Menu Refresh Map</i> .....	89
<b>Gambar 4.13</b> <i>Tombol Zoom In Dan Zoom Out</i> .....	90
<b>Gambar 4.14</b> <i>Menampilkan Lokasi User</i> .....	93
<b>Gambar 4.15</b> <i>Menampilkan Icon Minimarket Berdasarkan Tipe Peta</i> .....	94
<b>Gambar 4.16</b> <i>Menampilkan Minimarket Berdasarkan Radius Yang Berbeda</i> .....	95
<b>Gambar 4.17</b> <i>Menampilkan Minimarket Terdekat Berdasarkan Satu Lokasi User</i> .....	96
<b>Gambar 4.18</b> <i>Screenshot Hasil Pengujian 1</i> .....	98
<b>Gambar 4.19</b> <i>Screenshot Hasil Pengujian 2</i> .....	99
<b>Gambar 4.20</b> <i>Screenshot Hasil Pengujian 3</i> .....	100
<b>Gambar 4.21</b> <i>Screenshot Hasil Pengujian 4</i> .....	101
<b>Gambar 4.22</b> <i>Screenshot Hasil Pengujian 5</i> .....	102
<b>Gambar 4.23</b> <i>Screenshot Hasil Pengujian 6</i> .....	103
<b>Gambar 4.24</b> <i>Screenshot Hasil Pengujian 7</i> .....	104
<b>Gambar 4.25</b> <i>Screenshot Hasil Pengujian 8</i> .....	105
<b>Gambar 4.26</b> <i>Screenshot Hasil Pengujian 9</i> .....	106
<b>Gambar 4.27</b> <i>Screenshot Hasil Pengujian 10</i> .....	107

## DAFTAR LAMPIRAN

Kuesioner Uji Kelayakan Produk .....	123
Hasil Kuesioner Uji Kelayakan Produk .....	124



## ABSTRAK

Alfan Kurniawan. 2014. **Aplikasi Pencarian Minimarket Menggunakan Haversine Formula Untuk Menentukan Jarak Terdekat.**

Pembimbing : (I) Ririen Kusumawati, M.Kom (II) Ach. Nashicuddin, M.A

---

**Kata Kunci** : Aplikasi, Pencarian, Minimarket, *Haversine Formula*, Jarak Terdekat, Malang.

Saat ini mobilitas masyarakat dalam kehidupan sehari-hari semakin meningkat. Dalam satu hari orang dapat berpindah-pindah dari satu kota ke kota lain untuk keperluan tertentu dan memakan waktu yang tidak sebentar bahkan perlu menginap. Untuk itu perlu menyediakan kebutuhan sehari-hari dengan menuju *Minimarket* terdekat. Namun tiap kali orang kesulitan untuk mencari *Minimarket* karena tidak semua orang mengetahui jalan dan kondisi kota tersebut. Hal itu mengakibatkan tidak efisiennya waktu, tenaga dan biaya. Menanggapi adanya permasalahan diatas, perlu adanya aplikasi pencarian *minimarket* terdekat untuk memudahkan pengguna mencari *minimarket*.

Tujuan dari penelitian ini adalah membuat aplikasi yang dapat mencari *minimarket* dengan jarak terdekat menggunakan *Haversine Formula* serta teknologi *Google Maps* untuk mendapatkan hasil yang optimal. *Haversine Formula* merupakan persamaan dalam navigasi dengan memberikan jarak lingkaran besar (radius) antara dua titik pada permukaan bola (bumi) berdasarkan bujur dan lintang (*longitude* dan *latitude*).

Dari hasil penelitian ini, tercipta aplikasi pencarian *minimarket* untuk menentukan jarak terdekat dengan lokasi pengguna. Dalam pencarian *minimarket*, data diambil dari *database server*. *Device* yang digunakan harus memiliki perangkat GPS dan kondisi lalu lintas kota Malang yang sebenarnya diabaikan. Berdasarkan dari pengujian kelayakan aplikasi, 79.36% dari 50 responden menyatakan aplikasi ini layak digunakan dan di publikasikan ke masyarakat.

## ABSTRACT

Alfan Kurniawan. 2014. **Minimarket Search Application Using Methods Haversine Formula To Determine The Shortest Distance.**

Promotor : (I) Ririen Kusumawati, M.Kom (II) Ach. Nashicuddin, M.A

---

Today the mobility of people, everyday is increasing. In one day people can move from one city to another and takes a long time even, need to stay. It is necessary to provide the daily needs to the nearest minimarket. But each time people difficulty to find a minimarket because not everyone knows the condition of the road that city. It resulted in the inefficient time, effort and cost. In response to the above problems, need for minimarket search application to facilitate users search for nearby minimarket.

The purpose of this research is to create an application that can search the minimarket with the shortest distance using Haversine Formula and Google Maps technology to obtain optimal results. Haversine Formula is an equation in navigation by providing great circle distance (radius) between two points on the surface of the sphere (the earth) based on longitude and latitude.

Finally, From these research, the researcher creates an application minimarket search to determine the shortest distance to the location of the user. In search of the minimarket, the data taken from the database server. Device that is used must have a GPS tools and the actual traffic conditions are ignored. Based on testing the feasibility of the application, 79.36% of the 50 respondents stated this application is feasible and published to the public.

**Keywords:** *Application, Searcher, Minimarket, Haversine Formula, Shortest Distance, Malang.*

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Saat ini mobilitas masyarakat dalam kehidupan sehari-hari semakin meningkat. Dalam satu hari orang dapat berpindah-pindah dari satu kota ke kota lain untuk keperluan kerja, bisnis, wisata dan sebagainya. Terkadang untuk melakukan perjalanan antar satu kota ke kota lain memakan waktu yang tidak sebentar dan perlu menginap di kota tujuan. Untuk itu perlu menyediakan kebutuhan sehari-hari seperti makanan, minuman, sabun mandi, pasta gigi, obat-obatan dan sebagainya. Langkah mudah untuk memenuhi keperluan tersebut adalah menuju *minimarket* terdekat, karena saat ini telah banyak *minimarket* yang didirikan bahkan di kota kecil sekalipun. Namun tiap kali orang kesulitan untuk mencari *minimarket* terdekat yang akan di tuju karena tidak semua orang mengetahui jalan dan kondisi kota tersebut.

*Minimarket* sebagai salah satu tempat untuk membeli kebutuhan sehari-hari, karena menjual berbagai macam barang kebutuhan sehari-hari dan makanan. Pencarian *minimarket* perlu diaplikasikan pemetaannya secara akurat, karena hasil pencarian yang akurat sangat penting untuk menentukan jarak terdekat.

Melihat dari permasalahan di atas maka, skripsi ini bertujuan memberikan solusi yaitu dengan membangun APLIKASI PENCARIAN MINIMARKET MENGGUNAKAN METODE HAVERSINE FORMULA UNTUK

MENENTUKAN JARAK TERDEKAT. Sebagaimana dijelaskan dalam Al-Qur'an Surat Al-Insyrah ayat 5 dan 6:

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا ﴿٥﴾

“Karena sesungguhnya sesudah kesulitan itu ada kemudahan.” (QS. Al-Insyrah:5)

إِنَّ مَعَ الْعُسْرِ يُسْرًا ﴿٦﴾

“Sesungguhnya sesudah kesulitan itu ada kemudahan.” (QS. Al-Insyrah:6)

Tafsir menurut Departemen Agama pada surat Al-Insyrah ayat 5 di atas menerangkan bahwa Allah mengungkapkan bahwa sesungguhnya di dalam setiap kesempitan di situ terdapat kelapangan, dan di dalam setiap kekurangan sarana untuk mencapai suatu keinginan di situ pula terdapat jalan keluar jika seseorang dalam menuntut sesuatu tetap berpegang pada kesabaran dan tawakal kepada Tuhannya. Kemudian tafsir surat Al-Insyrah ayat 6 di atas menerangkan ayat ini adalah ulangan dari ayat sebelumnya dengan tujuan untuk menguatkan arti yang terkandung dalam ayat yang terdahulu, yakni bila kesulitan itu dihadapi dengan tekad yang sungguh-sungguh dan berusaha dengan sekuat tenaga dan pikiran untuk melepaskan diri dari padanya, tekun dan sabar serta tidak mengeluh atas kelambatan datangnya kemudahan, pasti kemudahan itu akan tiba.

Tafsir menurut Ustadz Muhammad Abduh Tuasikal menyebutkan bahwa ayat ini diawali dengan huruh fa (fa-inna ma'al 'usri yusran) untuk menunjukkan adanya kaitan antara kedua keadaan tersebut, yaitu antara timbulnya kesulitan dan datangnya kemudahan. Digunakannya kata Al sebelum kata Usri dalam kalimat

(fa-inna ma'al 'usri yusran) memberikan makna umum, yaitu segala macam kesulitan. Misalnya kesulitan berupa kemiskinan, kelemahan, pengkhianatan, dan apapun kesulitan yang biasa dijumpai dalam kehidupan.

Jenis kesulitan apapun pasti dapat diatasi, sepanjang orang yang menghadapi kesulitan tersebut memiliki jiwa yang kuat untuk mencari solusinya, Ayat ini mengajarkan bahwa setiap menghadapi berbagai kesulitan, kita harus yakin bahwa akan ada penyelesaiannya, akan ada jalan keluarnya. Pengulangan ini dimaksudkan agar setiap orang benar-benar yakin bahwa saat menghadapi kesulitan, sesungguhnya pada waktu yang bersamaan setiap manusia pasti akan bisa menemukan solusinya asalkan setiap manusia memiliki jiwa yang kuat, berpikir keras, ikhtiar yang sungguh-sungguh dan maksimal, serta berdoa kepada Allah SWT.

Haram hukumnya berputus asa saat menghadapi kesulitan sebab putus asa adalah karakternya orang-orang yang tidak bertuhan dan sesat. "Tiada yang putus asa dari rahmat Allah kecuali orang-orang yang sesat". (Q.S. Al-Hijr 15:56). Jadikanlah kesulitan sebagai media untuk mendewasakan karakter dan sikap.

Dalam tafsir Ibnu Katsir Juz 30 menjelaskan bahwa Allah Ta'ala memberitahukan, bersama kesulitan itu ada kemudahan. Kemudian Dia mempertegas berita tersebut. Ibnu Jarir meriwayatkan dari al-Hasan, dia berkata: Nabi Muhammad SAW pernah keluar rumah pada suatu hari dalam keadaan senang dan gembira, dan beliau juga dalam keadaan tertawa seraya bersabda yang artinya "Satu kesulitan itu tidak akan pernah mengalahkan dua kemudahan, satu

kesulitan itu tidak akan pernah mengalahkan dua kemudahan, karena bersama kesulitan itu pasti terdapat kemudahan, sesungguhnya bersama kesulitan itu terdapat kemudahan”.

Tafsir menurut Fi Zhilalil-Qur’an menjelaskan bahwa sesungguhnya kesulitan itu tidak lepas dari kemudahan yang menyertai dan mengiringinya. Hal ini sudah menyertaimu secara praktis. Maka, ketika terasa berat beban tugasmu, Kami lapangkan dadamu, sehingga terasa ringan beban yang memberatkan punggungmu. Kemudahan akan selalu mengiringi kesulitan, menghilangkan beban dan rasa beratnya.

Persoalan ini sangat serius hingga diulangi lagi penyebutan kalimatnya, “Fa inna ma’al ‘usri yusran, inna ma’al ‘usri yusran” ‘Karena sesungguhnya sesudah kesulitan itu ada kemudahan.Sesungguhnya sesudah kesulitan itu ada kemudahan’.

Kalimat ini mengisyaratkan bahwa Rasulullah SAW berada dalam kesulitan, kesempitan, dan penderitaan, yang memerlukan perhatian seperti ini. Beliau membutuhkan penyebutan ini,hadirnya symbol-simbol pertolongan, pemaparan tempat-tempat perlindungan, dan penegasan dengan segala bentuk penegasan.

Tafsir Al-Mishbah tulisan Prof.DR. Quraish Shihab menjelaskan bahwa pada ayat ke 5 dan 6 surat Al-Insyirah, terdapat kata ‘Usra yang berarti kesukaran, kesulitan. Dan kata yusra yang berarti kemudahan. Kata ini di ucapkan 2 kali. Dalam kaedah tafsir jika ada pengulangan kata memakai ‘ AL ‘ maka maknanya

sama, namun jika pengulangan kata tidak menggunakan ‘AL’ maka maknanya berbeda. Artinya boleh jadi Allah ingin mengatakan bahwa setiap ada satu kesulitan boleh jadi terdapat dua kemudahan yang bisa manusia capai, hal itu bermaksud bahwa manusia sebenarnya tidak perlu khawatir apabila kesulitan datang, karena justru ada kemudahan yang terkandung dalam kesulitan tersebut.

Ada Ulama yang berkata bahwa setelah kesusahan akan datang kemudahan. Pendapat dari Quraish Shihab adalah bahwa setiap ada kesulitan yang di alami manusia di dalamnya pasti ada peluang untuk menjadi kemudahan. Seperti hujan yang turun akan menjadi musibah bagi seorang tukang cuci karena pakaiannya tidak kering, namun bagi petani akan menjadi berkah yang luar biasa karena ladangnya akan tersirami air dari langit. Bagaimana manusia memandang sesuatu masalah menjadi suatu peluang adalah hal yang terpenting, dalam ayat ini Allah swt mengulangnya sampai dua kali, artinya Allah menegaskan bahwa sesuatu yang manusia anggap sebagai masalah justru sering menjadi berkah bagi manusia dan sebaliknya sesuatu yang sering di anggap berkah bagi manusia justru sering menjadi petaka menurut Allah.

Manusia sebenarnya tidak memiliki hak sama sekali untuk memprotes Allah swt atas segala kehendaknya, Allah memiliki pengetahuan yang manusia tidak pernah sama sekali memikirkannya, baik itu berupa kesempatan atau kelapangan semua adalah hak mutlak Allah untuk memberikannya pada manusia, tinggal bagaimana manusia memandang kesempatan dan kelapangan itu menjadi

berkah, jika di beri kesempatan manusia meski bersabar, jika di beri kelapangan manusia meski bersyukur.

Dari ayat tersebut menerangkan secara jelas bahwa Allah akan memberikan kemudahan setelah kesulitan jika manusia mau berusaha dan berjuang di jalan Allah dengan sabar dan tawakkal. Karena hanya kepada Allah lah kita berharap.

Apabila dijelaskan maka sesungguhnya Allah telah memberikan segala sesuatu yang dibutuhkan manusia untuk memenuhi kebutuhannya. Karena manusia di beri akal dan ilmu pengetahuan yang bermanfaat dan dapat dikembangkan untuk memudahkan segala urusan manusia di dunia. Berkaitan dengan hal di atas, dalam proses pembuatan aplikasi pencarian *Minimarket* menggunakan metode *Haversine Formula* untuk menentukan jarak terdekat, semuanya merupakan pemberian dari Allah untuk kita melalui karunia akal dan ilmu pengetahuan yang bermanfaat untuk membantu memudahkan pekerjaan manusia.

Objek penelitian *Minimarket* yang digunakan dalam skripsi ini adalah *Minimarket* yang umum berada kota Malang. Misalnya, Indomaret, Alfamart, Alfamidi, dan sebagainya. Pada aplikasi ini untuk menampilkan peta di ambil dari server *Google Maps* dan untuk perhitungan radius dan jarak terdekat menggunakan metode *Haversine Formula*. *Haversine Formula* merupakan persamaan penting yang digunakan dalam navigasi. *Haversine Formula* memberikan jarak lingkaran

besar (radius) antara dua titik pada permukaan bola (Bumi) berdasarkan bujur dan lintang (latitude & longitude).

## 1.2 Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah dalam skripsi ini adalah bagaimana membangun aplikasi pencarian *Minimarket* menggunakan metode *Haversine Formula* untuk menentukan jarak terdekat?

## 1.3 Batasan Masalah

Agar penelitian ini tidak menyimpang, maka perlu adanya pembatasan masalah. Pembatasan masalah yang dimaksudkan untuk membatasi ruang lingkup permasalahan yang akan di bahas, bukan untuk mengurangi sifat ilmiah suatu pembahasan. Batasan masalah skripsi ini adalah sebagai berikut:

- a. Penelitian ini difokuskan pada pembuatan aplikasi berbasis *mobile device* yang berjalan pada system operasi *android*
- b. Aplikasi yang di buat berbasis *Client & Server* serta *Web Service* untuk memudahkan update map tanpa harus menginstal ulang aplikasi
- c. Objek penelitian *Minimarket* difokuskan di Kota Malang
- d. Pada penelitian ini kondisi lalu lintas sebenarnya diabaikan
- e. Peta yang digunakan pada aplikasi ini berasal dari server Google Maps.
- f. *Minimarket* yang ditampilkan mengabaikan kebiasaan user yang fanatik *minimarket* tertentu.

#### 1.4 Tujuan

Tujuan dari skripsi ini adalah membangun aplikasi pencarian Minimarket menggunakan metode *Haversine Formula* dengan kemampuan dapat menentukan jarak terdekat dengan posisi user.

#### 1.5 Manfaat

Manfaat dari penelitian ini yaitu aplikasi dapat menemukan *Minimarket* dengan jarak terdekat sesuai posisi user untuk meminimalisir tenaga, waktu dan biaya.

#### 1.6 Sistematika Pembahasan

Sistematika pembahasan skripsi ini dikelompokkan menjadi lima Bab sebagai berikut:

##### **BAB I : PENDAHULUAN**

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian serta sistematika pembahasan.

##### **BAB II : DASAR TEORI**

Bab ini membahas tentang teori-teori yang mendukung dan berhubungan dengan judul penulisan skripsi.

##### **BAB III : ANALISIS DAN PERANCANGAN SISTEM**

Bab ini menjelaskan mengenai analisis dan perancangan sistem aplikasi berbasis Android.

#### **BAB IV : IMPLEMENTASI HASIL DAN PEMBAHASAN**

Bab ini membahas tentang laporan skripsi berupa tahapan implementasi dan uji coba dari perancangan sistem serta analisis hasil yaitu implementasi tabel dan pembuatan program aplikasi.

#### **BAB V : PENUTUP**

Bab ini berisi kesimpulan dan saran tentang hasil perancangan dan implementasi program.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Ayat Al-Qur'an Terkait Anjuran Allah Untuk Mencari Kemudahan

Sebagaimana Firman Allah dalam Al-Qur'an surat Al-Muzzammil ayat 20 dijelaskan bahwa:

﴿ إِنَّ رَبَّكَ يَعْلَمُ أَنَّكَ تَقُومُ أَدْنَىٰ مِنْ ثُلُثِي اللَّيْلِ وَنِصْفَهُ ۖ وَثُلُثَهُ ۖ وَطَائِفَةٌ مِّنَ الَّذِينَ مَعَكَ ۗ  
 وَاللَّهُ يُقَدِّرُ اللَّيْلَ وَالنَّهَارَ ۚ عَلِمَ أَن لَّنْ نَّحْصُوهُ فَتَابَ عَلَيْكُمْ ۖ فَاقْرَءُوا مَا تَيَسَّرَ مِنَ  
 الْقُرْآنِ ۚ عَلِمَ أَن سَيَكُونُ مِنكُم مَّرْضَىٰ ۙ وَآخَرُونَ يَضْرِبُونَ فِي الْأَرْضِ يَبْتَغُونَ مِن  
 فَضْلِ اللَّهِ ۙ وَآخَرُونَ يُقْتَلُونَ فِي سَبِيلِ اللَّهِ ۖ فَاقْرَءُوا مَا تَيَسَّرَ مِنْهُ ۚ وَأَقِيمُوا الصَّلَاةَ  
 وَآتُوا الزَّكَاةَ وَأَقْرِضُوا اللَّهَ قَرْضًا حَسَنًا ۚ وَمَا تُقَدِّمُوا لِأَنفُسِكُمْ مِن خَيْرٍ تَجِدُوهُ عِنْدَ  
 اللَّهِ هُوَ خَيْرًا وَأَعْظَمَ أَجْرًا ۚ وَاسْتَغْفِرُوا اللَّهَ ۚ إِنَّ اللَّهَ غَفُورٌ رَّحِيمٌ ۝۲۰

“*Sesungguhnya Tuhanmu mengetahui bahwasanya kamu berdiri (sembahyang) kurang dari dua pertiga malam, atau seperdua malam atau sepertiganya dan (demikian pula) segolongan dari orang-orang yang bersama kamu. dan Allah menetapkan ukuran malam dan siang. Allah mengetahui bahwa kamu sekali-kali tidak dapat menentukan batas-batas waktu-waktu itu, Maka Dia memberi keringanan kepadamu, karena itu bacalah apa yang mudah (bagimu) dari Al Quran. Dia mengetahui bahwa akan ada di antara kamu orang-orang yang sakit dan orang-orang yang berjalan di muka bumi mencari sebagian karunia Allah; dan orang-*

*orang yang lain lagi berperang di jalan Allah, Maka bacalah apa yang mudah (bagimu) dari Al Quran dan dirikanlah sembahyang, tunaikanlah zakat dan berikanlah pinjaman kepada Allah pinjaman yang baik. dan kebaikan apa saja yang kamu perbuat untuk dirimu niscaya kamu memperoleh (balasan)nya di sisi Allah sebagai Balasan yang paling baik dan yang paling besar pahalanya. dan mohonlah ampunan kepada Allah; Sesungguhnya Allah Maha Pengampun lagi Maha Penyayang”(QS. Al-Muzzammil:20).*

Tafsir Al-Qur'an Surat Al-Muzzammil ayat 20 menurut Departemen Agama yaitu bahwa pada ayat pertama surat ini Allah memerintahkan salat malam, maka ayat penutup ini menunjukkan betapa pengasihnya Allah kepada hamba-Nya. Dia memberikan keringan/kemudahan pada hamba-Nya dengan tidak mewajibkan salat tahjud setiap malam, bila dinyatakan berat mengerjakannya. Allah menegaskan bahwa Dia mengetahui sebagian kaum muslimin bersama Nabi mengerjakan salat malam itu sepanjang  $\frac{2}{3}$  malam, atau  $\frac{1}{2}$  nya atau  $\frac{1}{3}$  nya. Waktu itu masih merupakan perintah wajib yang tentu saja kadang-kadang terasa berat sekalipun salat tahajud itu khusus difardukan kepada Rasulullah SAW, dan disunahkan bagi umatnya.

Banyak diantara para sahabat tidak mengetahui dengan pasti ukuran  $\frac{1}{2}$  atau  $\frac{1}{3}$  malam itu. Sehingga oleh karena takut lupa/lupa dari waktu salat malam yang diperintahkan itu, ada diantara mereka berjaga-jaga sepanjang malam. Hal itu amat melelahkan tubuh mereka, sebab mereka bangun sampai fajar. Tentu saja bangun dan berjaga-jaga demikian melemahkan fisik. Untuk meringankan itu, Allah menurunkan ayat terakhir ini.

Dari ayat 20 ini dapat pula di ambil pelajaran lain, bahwa mengerjakan perintah fardu itu tidak boleh melebihi dari batas ukuran yang ditentukan agar tidak memberatkan diri sendiri. Oleh karena itu Allah memerintahkan bagi yang biasa salat malam apabila terasa agak memberatkan boleh dikurangi lama waktunya. Sehingga dikerjakan dalam keadaan tidak terpaksa. Begitulah Allah memudahkan sesuatu yang berat menjadi ringan, agar seseorang selalu mengerjakan yang mudah.

Tafsir menurut Jalalain menjelaskan bahwa Pada saat zaman Nabi, mereka (sahabat) melakukan salat malam untuk mengikuti jejak Nabi SAW. Sehingga disebutkan bahwa ada diantara mereka orang-orang yang tidak menyadari berapa rakaat salat malam yang telah mereka kerjakan, dan waktu malam tinggal sebentar lagi. Sesungguhnya Nabi SAW. Selalu melakukan salat sunah sepanjang malam, karena demi melaksanakan perintah Allah secara hati-hati. Para sahabat mengikuti jejaknya selama satu tahun atau lebih dari satu tahun, sehingga disebutkan bahwa telapak-telapak kaki mereka bengkak-bengkak karena terlalu banyak salat.

Berkaitan dengan hal di atas dikarenakan ada tiga golongan yaitu orang-orang yang sakit, orang yang melakukan perjalanan, orang yang dalam rangka mencari rizki melalui berniaga dan lain lainnya. Ketiga golongan orang-orang tersebut amat berat bagi mereka untuk melaksanakan salat malam. Akhirnya Allah memberikan keringanan/kemudahan kepada mereka bahwa kalian dapat melakukan salat malam sesuai batas kemampuan masing-masing dan membaca apa yang mudah dari Al-Qur'an.

Tafsir menurut Ibnu Katsir menerangkan bahwa berdasarkan hadis yang pernah dirawikan oleh Ubbaddah bin Shamit bahwa Nabi pernah bersabda: “Tidaklah ada sembahyang, bagi orang yang tidak membaca Fatihatil Kitab,” Tetapi ulama-ulama dalam Mahzab Hanafi ada yang berpendapat bahwa meskipun bukan Fatihah yang di baca, asal saja ayat Al-Qur’an walau satu ayat, sembahyangnya sah juga. Seperti yang telah diperintahkan Allah “Maka bacalah mana yang mudah (bagimu) dari Al-Qur’an”.

Ar-Razi menukilkan dalam tafsirnya perkataan Muqatil; “Ada sahabat Rasulullah yang sembahyang seluruh malam, karena takut kalau-kalau kurang sempurna mengerjakan sembahyang wajib. Tuhan telah tahu bahwa akan ada diantara kamu yang sakit.” Tentu saja orang yang sakit tidak diberati dengan perintah. Dan lagi kalau ada orang yang sembahyang saja terus-terusan satu malam, niscaya dia akan kurang tidur. Kurang tidur pun bisa menimbulkan sakit. Oleh karena itu dalam ayat 20 ini Allah memberikan keringan/kemudahan dalam menjalankan salat malam dan membaca apa yang mudah dari Al-Qur’an.

Tafsir Menurut Quraish Shihab menjelaskan sesungguhnya Tuhanmu mengetahui bahwa kamu, Muhammad, terkadang bangun malam kurang dari dua pertiganya. Di malam yang lain kamu bangun pada seperdua atau sepertiganya. Para pengikutmu pun melakukan hal yang sama seperti kamu. Tidak ada yang dapat menetapkan ukuran siang dan malam serta memastikan waktunya selain Allah. Dia

Mahatahu bahwa kamu tidak mungkin dapat menghitung secara pasti seluruh bagian siang dan malam itu.

Dari itu, Allah memberikan keringanan/kemudahan kepada kalian. Maka bacalah, dalam salat, ayat-ayat al-Qur'ân yang mudah. Allah Maha tahu bahwa di antara kalian ada yang menderita sakit sehingga sulit untuk melakukan ibadah di waktu malam. Demikian pula Allah mengetahui di antara kalian ada yang selalu bepergian untuk berniaga dan bekerja mencari karunia Allah. Di antara kalian pun ada yang tengah berjihad di jalan Allah untuk menegakkan kebenaran. Maka bacalah ayat al-Qur'ân yang mudah, lakukanlah kewajiban salat, tunaikanlah kewajiban zakat dan berikanlah pinjaman kepada Allah dengan pinjaman yang baik, yaitu dengan cara bersedekah kepada kaum fakir sebagai tambahan atas kewajiban yang telah ditentukan.

Dari ayat di atas menerangkan bahwa Allah memberikan kemudahan kepada umatnya dalam hal beribadah, seperti dalam firmanNya, “*Maka Dia (Allah) memberi keringanan kepadamu, karena itu bacalah apa yang mudah (bagimu) dari Al Quran*”. Jelas bahwa Allah menuntun kita untuk melakukan hal-hal yang memudahkan kita dalam kehidupan sehari-hari, dalam firmanNya “*Dia mengetahui bahwa akan ada di antara kamu orang-orang yang sakit dan orang-orang yang berjalan di muka bumi mencari sebagian karunia Allah; dan orang-orang yang lain lagi berperang di jalan Allah*”. Dijelaskan bahwa orang-orang yang melakukan perjalanan di muka bumi

untuk mencari karunia atau rizki Allah, di beri keringanan dan di beri kemudahan dalam menjalankan salat malam.

Merujuk dari hal di atas, maka dalam skripsi ini *minimarket* sebagai objek penelitian digunakan untuk memberikan solusi dengan membuat aplikasi pencarian *minimarket* terdekat. Dengan tujuan untuk meringankan/memudahkan seseorang dalam melakukan pencarian *minimarket* di Kota Malang. Mengingat tingginya *mobilitas* masyarakat pada saat ini untuk keperluan kerja, bisnis atau wisata yang dapat memakan waktu lebih dari satu hari.

## 2.2 *Minimarket*

Dalam dunia perdagangan saat ini, toko barang kebutuhan sehari-hari dengan ruangan yang tidak terlalu luas (*minimarket*) bukan lagi merupakan istilah asing bagi masyarakat umum, terutama yang tinggal di kota-kota besar. *Minimarket* merupakan perantara pemasar antara produsen dan konsumen akhir di mana aktivitasnya adalah melaksanakan penjualan eceran.

*Minimarket* adalah toko swalayan yang hanya memiliki satu atau dua mesin register. Sementara Supermarket adalah swalayan besar yang juga menjual barang-barang segar dan kebutuhan sehari-hari seperti sayur dan daging dengan jumlah mesin registernya mencapai tiga ke atas. Menurut Hendri Ma'ruf dalam bukunya Pemasaran Ritel pengertian *minimarket* adalah Toko yang mengisi kebutuhan

masyarakat akan warung yang berformat modern yang dekat dengan permukiman penduduk sehingga dapat mengungguli toko atau warung (Hendri Ma'ruf, 2005).

Sebuah *minimarket* sebenarnya semacam “toko kelontong” atau yang menjual segala macam barang dan makanan, namun tidak selengkap dan sebesar sebuah *Supermarket*. Berbeda dengan toko kelontong, *minimarket* menerapkan sistem swalayan, dimana pembeli mengambil sendiri barang yang ia butuhkan dari rak-rak dagangan dan membayarnya di kasir. Sistem ini juga membantu agar pembeli tidak berhutang. Menurut Hendri Ma'ruf yang disebut *minimarket* biasanya luas ruangnya adalah antara 50 m<sup>2</sup> sampai 200 m<sup>2</sup> serta berada pada lokasi yang mudah di jangkau konsumen. *Minimarket* yang ada di Indonesia adalah Alfamart, Indomaret, Ceriamart, Starmart, Circle K, dan lain-lain.

### 2.3 Haversine Formula

Rumus haversine adalah persamaan penting dalam navigasi, haversine formula memberikan jarak lingkaran besar (radius) antara dua titik pada permukaan bola (Bumi) berdasarkan bujur dan lintang. Penggunaan formula ini mengasumsikan pengabaian efek elipsoidal (diasumsikan bumi tidak bulat sempurna melainkan lebih mendekati bentuk telur dengan permukaan yang tidak rata), cukup akurat untuk sebagian besar perhitungan, juga pengabaian ketinggian bukit dan kedalaman lembah di permukaan bumi. Untuk dua titik pada bola (dari radius R) dengan garis lintang dan  $\phi_1$   $\phi_2$ , pemisah lintang  $\Delta\phi = \phi_1 - \phi_2$ , dan bujur

pemisah  $\Delta\lambda$ , di mana sudut dalam radian,  $d$  jarak antara dua titik (sepanjang lingkaran besar dari bola; melihat jarak bola) berhubungan dengan lokasi oleh rumus sebagai berikut:

$$\text{Haversin}\left(\frac{d}{R}\right) = \text{haversin}(\theta_1 - \theta_2) + \cos(\theta_1) \cos(\theta_2) \text{haversin}(\Delta\lambda)$$

Dimana :

$$\text{Haversin}(\theta) = \sin^2(\theta/2) = (1 - \cos(\theta))/2.$$

$\Phi 1$  = garis lintang (*latitude*) dari titik 1.

$d$  = jarak dua titik (antara posisi user dan koordinat yang di tuju).

$\phi 2$  = garis lintang (*latitude*) dari titik 2.

$R$  = jari-jari bola (*radius bumi*).

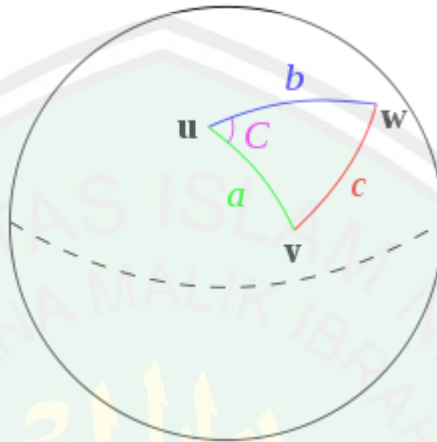
$\Delta\lambda$  = pemisah bujur (*longitude*)

Sudut pada rumus menggunakan satuan radian untuk menggunakan fungsi trigonometri. Rumus haversine digunakan pada penelitian ini untuk melakukan perhitungan jangkauan radius pencarian minimarket yang posisinya terdekat dengan user. Rumus ini juga digunakan untuk substitusi perhitungan jarak antara posisi minimarket dan posisi user.

Algoritma ini nantinya akan digunakan untuk menghitung jarak antara dua titik koordinat GPS. Dalam hal ini adalah titik koordinat posisi user dengan posisi koordinat yang di tuju, sehingga algoritma ini menjadi kunci utama untuk

menentukan jarak antara posisi user dengan *minimarket* terdekat berdasarkan radius.

Lihat gambar 2.1 berikut ini.



**Gambar 2.1** Segitiga bola diselesaikan dengan hukum haversine formula

Jika panjang dari ketiga sisi adalah  $a$  (u ke v),  $b$  (dari u untuk w), dan  $c$  (dari v ke w), dan sudut  $c$  sebaliknya adalah  $C$ , maka hukum haversine sebagai berikut.  $\text{Haversin}(c) = \text{haversin}(a-b) + \sin(a)\sin(b)\text{haversin}(c)$ .

Maka dari rumus di atas didapatkan rumus yang akan diterapkan pada skripsi ini seperti berikut.

$$R = \text{Radian Bumi} = 6371 \text{ km.}$$

$$\Delta \text{lat} = \text{lat2} - \text{lat1}$$

$$\Delta \text{long} = \text{long2} - \text{long1}$$

$$a = \sin^2(\Delta \text{lat}/2) + \cos(\text{lat1}) * \cos(\text{lat2}) * \sin^2(\Delta \text{long}/2)$$

$$c = 2 * \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R * c$$

Dari perhitungan di atas, untuk menentukan jarak paling dekat harus terdapat pembanding. Jadi harus terdapat banyak titik lokasi dan ditemukan jarak, barulah dibandingkan mana jarak terpendek. Untuk membatasi ruang lingkup peta, digunakan radius agar wilayah yang di cover tidak melebar. Untuk jalur jalan yang sebenarnya sesuai kondisi memanfaatkan google maps, lalu akan ditampilkan marker dimana jarak terdekat akan diberi *marker* dengan warna biru. Untuk perhitungan query akan dilakukan di server dan hasilnya objek data JSON karena, jika datanya banyak maka server dapat mengakomodir.

## 2.4 JSON

JSON (JavaScript Object Notation) merupakan format yang ringan untuk memasukan data ke dalam sebuah variabel. Sangat mudah dimengerti dan di implementasikan oleh manusia, dan mudah juga untuk komputer dalam melakukan parsingnya (<http://www.json.org>).

JSON (JavaScript Object Notation) adalah sebuah format *data-interchange* yang ringan. Sangat mudah bagi manusia untuk membaca dan menuliskannya. Semudah mesin memarsing dan men-generate format datanya. JSON merupakan bagian dari bahasa pemrograman JavaScript (Standard ECMA-262 3rd Edition – December 1999). JSON merupakan format teks yang sepenuhnya independen tetapi menggunakan konvensi yang familiar dengan bahasa pemrograman dari keluarga-C, termasuk C, C++, C#, Java, JavaScript, Perl, Python, dan sebagainya. Kelebihan

inilah yang membuat JSON menjadi sebuah bahasa *data-interchange* yang ideal (Adnan Sohail, 2009).

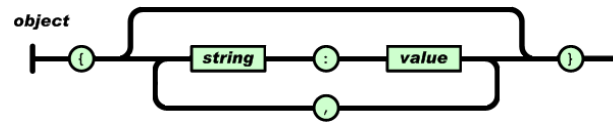
JSON di bangun dalam dua struktur:

- a. Beberapa pasangan dari nama/nilai. Dalam beberapa bahasa perograman biasa disebut dengan istilah object, record, struct, tabel hash, key list atau associative array.
- b. Nilai-nilai yang terusun secara ordered list. Biasa disebut dengan array, vector, list atau daftar dalam bahasa pemrograman.

JSON adalah struktur data yang universal, dalam artian bisa digunakan dalam berbagai bahasa pemrograman. Hampir semua bahasa pemrograman mendukung penuh JSON dalam berbagai format. Hal ini memungkinkan format data yang dapat dipertukarkan menggunakan bahasa pemrograman juga menggunakan dasar dari struktur JSON (Douglas Crockford, 2008).

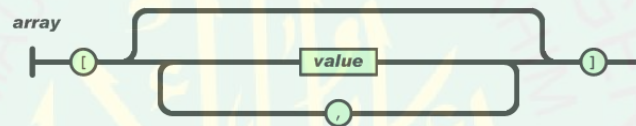
Format data JSON mempunyai aturan sebagai berikut:

Object adalah satu set nama/nilai yang tidak terurut (An object is an unordered set of name/value pairs). Penulisan object di mulai dengan tanda { (left brace) dan diakhiri dengan tanda } (right brace). Setiap nama diikuti oleh tanda : (colon) dan pasangan nama/nilai dipisahkan dengan tanda , (comma). Lihat gambar 2.2 berikut ini.



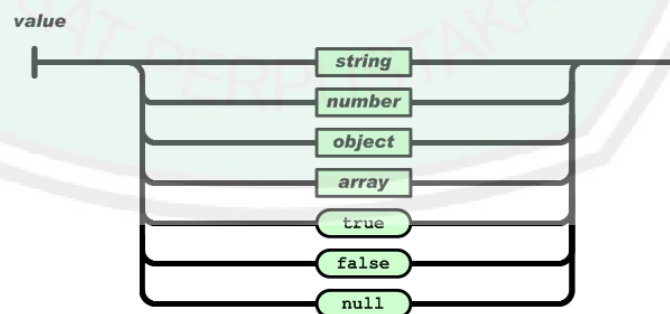
**Gambar 2.2** Object dalam JSON

Array adalah sekumpulan nilai yang teratur (An array is an ordered collection of values). Penulisan sebuah array di mulai dengan tanda [ (left bracket) dan diakhiri dengan tanda ] (right bracket). Nilai dipisahkan menggunakan tanda , (comma). Lihat gambar 2.3 berikut ini.



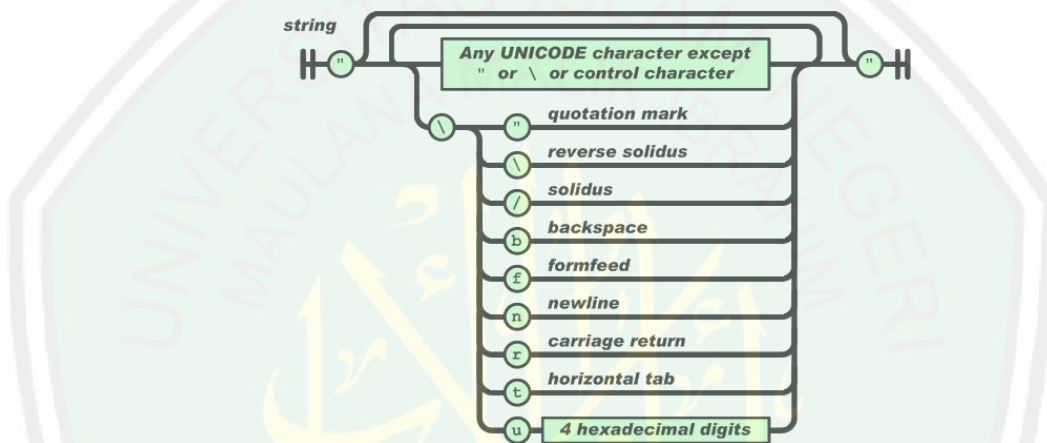
**Gambar 2.3** Array dalam JSON

Nilai bisa berupa string dalam tanda kutip, atau number (angka), TRUE atau FALSE atau NULL, sebuah object atau sebuah array. Struktur ini dapat di tulis menggunakan metode bersarang. Lihat gambar 2.4 berikut ini.



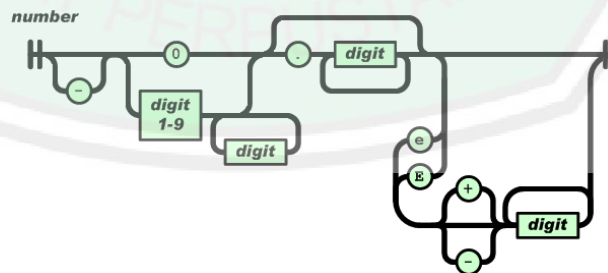
**Gambar 2.4** Value atau nilai dalam format JSON

String adalah rangkaian atau urutan karakter unicode yang berada dalam tanda kutip, bisa juga hanya berisi karakter kosong, menggunakan tanda \ (backslash) untuk escape. Karakter di representasikan sebagai string tunggal. Tipe data string pada JSON sangat mirip dengan definisi pada bahasa C atau Java. Lihat gambar 2.5 berikut ini.



**Gambar 2.5** Tipe data string pada format JSON

Number sangat mirip dengan definisi pada bahasa C atau Java, hanya saja tipe bilangan oktal dan heksadesimal tidak digunakan. Lihat gambar 2.6 berikut ini.



**Gambar 2.6** Number atau angka dalam format JSON

Ruang kosong (Whitespace) dapat diselipkan di antara pasangan atau token apa saja. Kecuali detail encoding, yang secara jelas akan berfungsi untuk menjelaskan bahasa yang digunakan.

## 2.5 MySQL

MySQL merupakan *software database open source* yang paling populer di dunia, dimana saat ini digunakan lebih dari 100 juta pengguna di seluruh dunia. Dengan kehandalan, kecepatan dan kemudahan penggunaannya, MySQL menjadi pilihan utama bagi banyak pengembang *software* dan aplikasi baik di *platform web* maupun *desktop*. Pengguna MySQL tidak hanya sebatas pengguna perseorangan maupun perusahaan kecil, namun perusahaan seperti Yahoo!, Alcatel-Lucent, Google, Nokia, Youtube, Wordpress dan Facebook juga merupakan pengguna MySQL.

MySQL pertama kali dibuat dan dikembangkan di Swedia, yaitu oleh David Axmark, Allan Larsson dan Michael "Monty" Widenius. Mereka mengembangkan MySQL sejak tahun 1980-an. Saat ini versi MySQL yang sudah stabil mencapai versi 5x, dan sedang dikembangkan versi 6x.(Ahmad Solichin, 2010).

## 2.6 Google Maps

Google Maps adalah aplikasi yang mirip dengan Google Earth, namun ditampilkan dengan antarmuka web. Karena menggunakan *thin client browser* maka pengalaman yang akan diperoleh dari Google Maps tidak akan penuh seperti ketika

menggunakan Google Earth. Jadi Google Maps adalah versi ringan dari Google Earth dengan beberapa yang di hapus. Google Maps bisa di akses di <http://maps.google.com> (Ali Zaki, 2010).

Google Maps API adalah layanan gratis yang diberikan oleh Google untuk mendukung perkembangan SIG, sehingga para developer *web* maupun sistem yang lain menggunakan Google Maps sebagai penunjang fasilitas di dalam sistemnya. Agar dapat menggunakan Google Maps, dibutuhkan suatu media penghubung antara Google Maps dengan sistem yang di bangun. Maka, dikembangkan suatu interface guna memfasilitasi hal tersebut, yaitu Google Maps API.

Google Maps API adalah antarmuka pemrograman aplikasi yang menyediakan berbagai fungsi dan sekumpulan objek dalam bahasa *JavaScript* sehingga citra digital atau peta Google Maps dapat ditampilkan pada halaman *website* lain. Melalui Google Maps API inilah para pengguna internet maupun developer tidak perlu bersusah payah membangun suatu citra digital atau peta, sehingga yang difokuskan adalah data-data yang menjadi pendukung saja.

Walaupun tidak berbayar, untuk dapat menggunakan Google Maps pada suatu halaman *website*, pengguna harus melakukan pendaftaran secara online terlebih dahulu untuk mendapatkan API Key. Google Maps dapat menerima input koordinat dalam format derajat lintang dan derajat bujur, tanpa mengenal menit dan detik dengan tanda positif untuk lintang utara dan bujur timur, sedangkan untuk lintang

selatan dan bujur barat digunakan tanda negatif. Di dalam koordinat Google Maps, garis lintang dan bujur di kenal dengan nama *latitude* dan *longitude*. *Latitude* (lat) merupakan garis yang membentang dari selatan ke utara, sedangkan *longitude* (lng) merupakan garis yang membentang dari barat ke timur. Pada garis ekuator, *latitude* bernilai 0, maksudnya adalah, semua yang berada di bagian bawah dari ekuator (selatan) bernilai negative, sedangkan di atasnya (utara) akan bernilai positif. Untuk *longitude*, acuannya adalah *prime meridian*. Sesuai sejarah Greenwich England, semua posisi yang berada di sebelah timur garis ini, akan bernilai positif, sedangkan yang berada di sebelah barat, bernilai negative.

Saat ini versi terakhir Google Map API adalah versi 3. Versi ini, akan tampil lebih cepat dari versi sebelumnya khususnya untuk browser ponsel. Ya, ponsel, kita bisa membangun situs web yang dilengkapi peta untuk iPhone dan ponsel dengan system operasi Android.

Google Maps merupakan sebuah layanan peta dunia visual berbasis web yang disediakan oleh Google. Google maps menawarkan peta yang dapat di geser (*panned*), diperbesar (*zoom in*), diperkecil (*zoom out*), dapat di ganti dalam beberapa mode (*map, satelit, hybrid, dll*), fitur pencarian rute (*routing*), petunjuk arah dari suatu objek peta ke objek yang lain (*direction*), dan juga pencari tempat (*place*) bisnis di Amerika, Kanada, Jepang, Hongkong, Cina, Inggris, dll (hanya pusat kota). Sampai saat ini Google Maps masih berada dalam tahap beta, dan masih terus dikembangkan dengan data yang selalu diperbarui secara berkala (Riyanto, 2010).

Google Maps memanfaatkan teknologi digital imaging seperti, foto Satelit sehingga kita bisa melihat bagaimana landscape planet bumi apabila di lihat dari luar angkasa.

Google Maps API adalah fasilitas dari Google untuk menambahkan peta dalam website. Google Maps API dapat ditambahkan ke website menggunakan JavaScript. API tersebut menyediakan banyak fasilitas dan utilitas untuk memanipulasi peta dan menambahkan konten ke peta melalui berbagai layanan, memungkinkan Kita untuk membuat aplikasi peta yang kuat pada website (Catur, 2010).

Pada Google Maps API terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

1. ROADMAP, ini di pilih, untuk menampilkan peta biasa 2 dimensi
2. SATELLITE, untuk menampilkan foto satelit
3. TERRAIN, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai
4. HYBRID, akan menunjukkan foto satelit yang diatasnya tergambar pula apa yang tampil pada ROADMAP (jalan dan nama kota).

## 2.7 Layanan Berbasis Lokasi (*Location Based Service*)

Definisi *location based service* atau di singkat LBS adalah kemampuan untuk mencari lokasi geografis dari *mobile device* dan menyediakan layanan berdasarkan lokasi yang diperolehnya. Konsep LBS ini menghasilkan layanan informasi yang dapat dicocokkan dengan lokasi keberadaan calon penerima informasi. Hal ini menyebabkan peningkatan nilai informasi dikarenakan penerima dapat mengasosiasikan pengetahuan atau informasi yang di dapat dengan keberadaannya.

LBS dapat diklasifikasikan menjadi 3 jenis, yaitu *local information*, *traffic and tracking information* dan *general services*. *Local information* memungkinkan pengguna untuk mencari layanan di sekitar mereka. *Traffic and tracking information* berfokus kepada pelacakan aset atau manusia. Sedangkan *general services* tidak menyediakan informasi ke pengguna, namun menggunakan data lokasi pengguna. Contoh dari *general services* ini antara lain layanan *emergency* (darurat).

Ada dua definisi yang bisa dijelaskan. Definisi Pertama; LBS adalah layanan informasi yang dapat di akses menggunakan piranti *mobile* melalui jaringan Internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada piranti *mobile*. Definisi Kedua; Layanan IP *nirkabel* yang menggunakan informasi geografis untuk memberikan layanan informasi lokasi kepada pengguna. Beberapa layanan aplikasi yang memberikan petunjuk posisi/lokasi piranti *mobile* berada

(Stefan Steiniger, 2006). Terdapat empat komponen pendukung utama dalam teknologi Layanan Berbasis Lokasi, antara lain (Stefan Steiniger, 2006):

### **2.7.1 Piranti Mobile**

Piranti Mobile adalah salah satu komponen penting dalam LBS. Piranti ini berfungsi sebagai alat bantu (*tool*) bagi pengguna untuk meminta informasi. Hasil dari informasi yang di minta dapat berupa teks, suara, gambardan lain sebagainya. Piranti *mobile* yang dapat digunakan bisa berupa PDA, *smartphone*, *laptop*. Selain itu, piranti *mobile* dapat juga berfungsi sebagai alat navigasi di kendaraan seperti halnya alat navigasi berbasis GPS.

### **2.7.2 Jaringan Komunikasi**

Komponen kedua adalah jaringan komunikasi. Komponen ini berfungsi sebagai jalur penghubung yang dapat mengirimkan data-data yang di kirim oleh pengguna dari piranti *mobile*-nya untuk kemudian dikirimkan ke penyedia layanan dan kemudian hasil permintaan tersebut dikirimkan kembali oleh penyedia layanan kepada pengguna.

### **2.7.3 Komponen Positioning (Penunjuk Posisi)**

Setiap layanan yang diberikan oleh penyedia layanan biasanya akan berdasarkan pada posisi pengguna yang meminta layanan tersebut. Oleh karena itu diperlukan komponen yang berfungsi sebagai pengolah/pemroses yang akan menentukan posisi pengguna layanan saat itu. Posisi pengguna tersebut bisa didapatkan melalui jaringan komunikasi *mobile* atau juga menggunakan *Global Positioning System* (GPS).

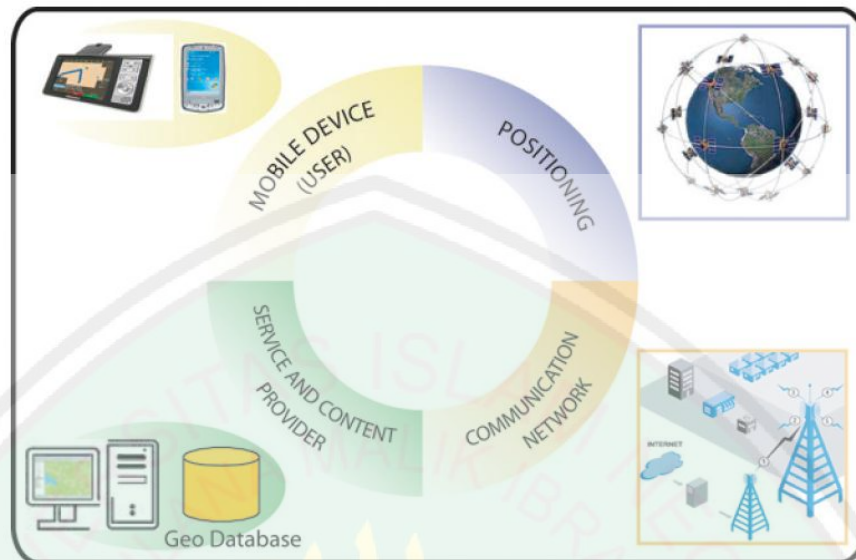
#### **2.7.4 Penyedia Layanan dan Aplikasi**

Penyedia layanan merupakan komponen LBS yang memberikan berbagai macam layanan yang bisa digunakan oleh pengguna. Sebagai contoh ketika pengguna meminta layanan agar bisa tahu posisinya saat itu, maka aplikasi dan penyedia layanan langsung memproses permintaan tersebut, mulai dari menghitung dan menentukan posisi pengguna, menemukan rute jalan, mencari data di Yellow Pages sesuai dengan permintaan, dan masih banyak lagi yang lainnya.

#### **2.7.5 Penyedia Data dan Konten**

Penyedia layanan tidak selalu menyimpan seluruh data dan informasi yang diolahnya. Karena bisa jadi berbagai macam data dan informasi yang diolah tersebut berasal dari pengembang/pihak ketiga yang memang memiliki otoritas untuk menyimpannya. Sebagai contoh basis data geografis dan lokasi bisa saja berasal dari badan-badan milik pemerintah atau juga data-data perusahaan/bisnis/industri bisa saja berasal dari Yellow Pages, maupun perusahaan penyedia data lainnya.

Secara lengkap keempat komponen pendukung LBS tersebut dapat di lihat pada gambar 2.7 seperti berikut.



**Gambar 2.7** *Komponen pendukung utama teknologi LBS (Stefan Steiniger, 2006)*

## 2.8 Global Positioning System (GPS)

GPS (Global Positioning System) adalah sistem satelit navigasi dan penentuan posisi yang dimiliki dan dikelola oleh Amerika Serikat. Sistem ini didesain untuk memberikan posisi dan kecepatan tiga dimensi serta informasi mengenai waktu. GPS terdiri dari 3 segmen yaitu segmen angkasa, kontrol/pengendali, dan pengguna. Segmen angkasa terdiri dari 24 satelit yang beroperasi dalam 6 orbit pada ketinggian 20.200 km dengan periode 12 jam (satelit akan kembali ke titik yang sama dalam 12 jam). Segmen Kontrol/Pengendali terdapat pusat pengendali utama yang terdapat di Colorado Springs, dan 5 stasiun pemantau lainnya dan 3 antena yang tersebar di bumi ini. Pada sisi pengguna dibutuhkan penerima GPS yang biasanya terdiri dari penerima, prosesor, dan antenna (Abidin Hasanudin Z, 2000).

GPS atau Global Positioning System dalam pengertian sederhana adalah salah satu sistem yang akan membantu kita untuk mengetahui posisi kita berada saat ini. GPS bekerja dengan mentransmisikan sinyal dari satelit ke perangkat *GPS* (handphone atau Blackberry yang dilengkapi teknologi GPS). Untuk memperoleh detail posisi yang seakurat mungkin, GPS sebaiknya digunakan di ruang terbuka. Penggunaan GPS di dalam ruangan, hutan ataupun di tempat yang banyak gedung-gedung tinggi, akan membuat GPS bekerja kurang akurat.

GPS (Global Positioning System) adalah sebuah sistem navigasi berbasis radio yang menyediakan informasi koordinat posisi, kecepatan, dan waktu kepada pengguna di seluruh dunia. Jasa penggunaan satelit GPS tidak dikenakan biaya. Pengguna hanya membutuhkan GPS receiver untuk dapat mengetahui koordinat lokasi. Keakuratan koordinat lokasi tergantung pada tipe GPS receiver (g-excess.com, 2009).

Teknologi GPS pertama kali digunakan oleh United States Departement of Defense (DOD) untuk kebutuhan militer. Sistem GPS mulai digunakan sejak tahun 1980, namun pemakaian secara umum oleh publik baru sekira tahun 1990-an.

Keistimewaan GPS adalah mampu bekerja dalam berbagai kondisi cuaca, siang atau malam. Keakuratan sebuah perangkat GPS bisa mencapai 15 meter, bahkan model terbaru yang dilengkapi teknologi Wide Area Augmentation System (WAAS) keakuratannya sampai 3 meter (Ahmed El-Rabbany, 2002).

## 2.9 Android

Penemu Android adalah Andy Rubin yang lahir pada tanggal 22 Juni 1946 di New Bedford, Amerika Serikat. Andy Rubin bersama-sama dengan Rich Miner, Nick Sears, dan Chris White mendirikan Android.inc. Android adalah sebuah sistem operasi mobile yang berbasiskan pada versi modifikasi dari Linux. Pertama kali sistem operasi ini dikembangkan oleh perusahaan Android.Inc. Nama perusahaan inilah yang pada akhirnya digunakan sebagai nama proyek sistem operasi mobile tersebut, yaitu sistem operasi Android.

Pada tahun 2005, sebagai dari strategi untuk memasuki pasar mobile, Google membeli Android dan mengambil alih proses pengembangannya sekaligus team developer Android. Goole menginginkan Android untuk menjadi sistem operasi Open Source dan gratis, kebanyakan code Android di rilis di bawah lisensi Open Source Apache yang berarti setiap orang bebas untuk menggunakan dan mengunduh source code Android secara penuh.

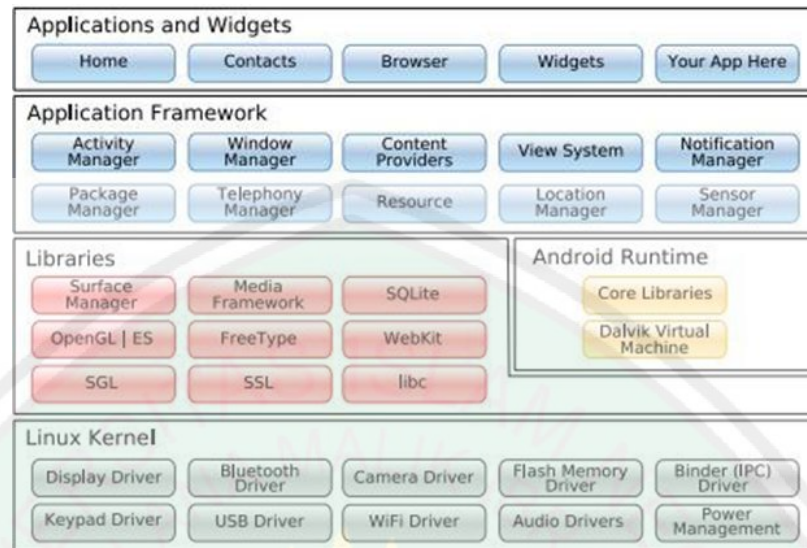
Terlebih lagi para vendor bebas untuk mengubah sekaligus membuat penyesuaian untuk Android. Di samping itu, perusahaan dapat secara bebas untuk membuat perbedaan dari produk vendor lainnya. Model pengembangan yang sederhana membuat Android sangat atraktif dan hal tersebutlah membuat para vendor tertarik untuk mencoba sistem operasi Android (Wahana Komputer, 2013).

Hingga tahun 2012, Android telah berkembang dengan pesat. Dalam kurun 3 tahun Android telah di produksi dalam berbagai versi, dan versi terakhir yang di produksi disebut sebagai Android versi 4.1 atau Android Jelly Bean (cyberkomputer.com, 2012 ).

Perkembangan Android dari tahun 2009 - 2012

- Android Versi 1.1 (Rilis 9 Maret 2009).
- Android Versi 1.5 (Cupcake – Rilis Pertengahan Mei 2009)
- Android versi 1.6 (Donut – Rilis September 2009)
- Android Versi 2.0/2.1 (Éclair – Rilis 3 Desember 2009)
- Android Versi 2.2 (Froyo: Frozen Yoghurt – Rilis 20 Mei 2010)
- Android Versi 2.3 (Gingerbread – Rilis 6 Desember 2010)
- Android Versi 3.0/3.1 (Honeycomb – Rilis Mei 2011)
- Android Versi 4.0 (ICS : Ice Cream Sandwich – Rilis 19 Oktober 2011)
- Android versi 4.1 (Jelly Bean – Rilis 27 Juni 2012)
- Android versi 4.2 (A New Flavor of Jelly Bean – Rilis 13 November 2012)

Secara garis besar Arsitektur Android dapat digambarkan seperti gambar 2.8 berikut ini.



**Gambar 2.8** Diagram komponen utama android. (Syafaat, 2010)

Pada gambar 2.8 diperlihatkan sistem operasi Android memiliki 4 lapisan (*layer*) yang merupakan komponen sistem Android. Google mengibaratkan Android sebagai sebuah tumpukan *software*. Setiap lapisan dari tumpukan ini menghimpun beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Berikut ini susunan dari lapisan-lapisan tersebut jika di lihat dari lapisan dasar hingga lapisan teratas.

a. *Applications dan Widgets*

*Applications dan Widgets* adalah *layer* di mana pengguna berhubungan dengan aplikasi. Di mana biasanya pengguna men-*download* aplikasi kemudian menginstalasi dan menjalankan aplikasi tersebut. Di *layer* ini terdapat aplikasi inti termasuk klien email, program SMS, kalender, peta, browser, kontak, dan

lain-lain. Semua aplikasi di tulis dengan menggunakan bahasa pemrograman Java.

b. *Applications Frameworks*

Android adalah “*Open Development Platform*” yaitu Android menawarkan kepada pengembang atau memberikan kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alarm, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju *API framework* seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi di rancang supaya pengguna dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*).

*Applications Frameworks* adalah *layer* di mana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android. Karena pada *layer* inilah aplikasi dapat di rancang dan di buat, seperti *content-providers* yang berupa SMS dan panggilan telepon.

Komponen-komponen yang termasuk di dalam *Applications Frameworks* adalah sebagai berikut :

- 1) *Views*
- 2) *Content Provider*

3) *Resource Manager*

4) *Notification Manager*

5) *Activity Manager*

c. *Libraries*

*Libraries* adalah *layer* di mana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya berjalan di atas kernel, *layer* ini meliputi berbagai library C/C++ inti seperti Libc dan SSL, serta :

- 1) *Libraries* media untuk pemutaran media audio dan video.
- 2) *Libraries* untuk manajemen tampilan.
- 3) *Libraries Graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D.
- 4) *Libraries SQLite* untuk dukungan *database*.
- 5) *Libraries SSL* dan *Webkit* terintegrasi dengan *web browser* dan *security*.
- 6) *Libraries LiveWebcore* mencakup modern *web browser* dengan *engine embedded web view*.
- 7) *Libraries 3D* yang mencakup implementasi OpenGL ES 1.0 API's.

d. *Android Run Time*

*Layer* yang membuat aplikasi Android dapat dijalankan di mana dalam prosesnya menggunakan implementasi Linux. *Dalvik Virtual Machine (DVM)*

merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam Android *Run Time* di bagi menjadi dua bagian yaitu :

- 1) *Core Libraries*, aplikasi Android di bangun dalam bahasa Java, sementara DVM sebagai virtual mesinnya buka *Virtual Machine* Java, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa Java/C yang ditangani oleh *Core Libraries*.
- 2) *Dalvik Virtual Machine*, Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, di mana merupakan pengembangan yang mampu membuat linux *kernel* untuk melakukan *threading* dan manajemen tingkat rendah.

e. *Linux Kernel*

*Linux Kernel* adalah *layer* di mana inti dari *operating system* dari Android itu berada. Berisi *file-file system* yang mengatur sistem *processing*, *memory*, *resource*, *drivers*, dan sistem-sistem operasi Android. *Linux kernel* yang digunakan android adalah *linux kernel release 2.6* (Nazruddin Safaat H., 2012).

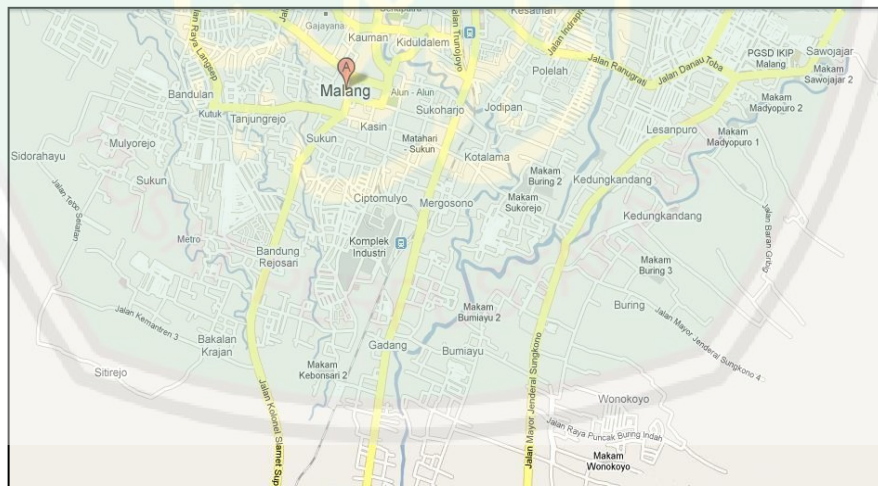
## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

Dalam bab ini akan dijelaskan mengenai arsitektur dari aplikasi *pencarian Minimarket menggunakan Haversine Formula untuk menentukan jarak terdekat* dan juga mengenai cara kerja tiap bagian dari aplikasi tersebut.

#### 3.1 Peta

Penggunaan peta pada aplikasi yang akan dibuat dalam penelitian ini adalah dari server Google Maps dan *server localhost* untuk pengambilan data keterangan Minimarket. Peta pada aplikasi yang akan buat ini di ambil dari web site [www.maps.google.com](http://www.maps.google.com) peta malang lihat gambar 3.1 berikut ini.



**Gambar 3.1** Tampilan peta Malang

Parameter *mapTypeId* menentukan jenis peta yang akan ditampilkan.

Pilihannya ada dua:

- a. ROADMAP 2D, untuk menampilkan peta biasa dua dimensi.
- b. SATELLITE, untuk menampilkan foto satelit.

### 3.2 Cara Kerja Sistem GPS

Cara kerja dari sistem ini meliputi :

- a. User memberikan koordinat lokasi dengan membaca melalui *devices* GPS *Mobile Phone*, kemudian memilih inputan yang berupa : setting type map dan setting radius.
- b. Aplikasi *client* terhubung dengan *server* dan memberikan informasi yang diperlukan.
- c. Pada sisi *server*, informasi akan di proses dengan data yang telah tersimpan dalam database, untuk memperoleh tempat yang diinginkan *user*, *server* mengirim kembali hasil proses ke *client*.
- d. Aplikasi *client* akan menerima hasil informasi dari *server*, dan akan di olah agar dapat di baca oleh *end user*.
- e. Koordinat *user* di ambil dari *devices* GPS *Mobile Phone*, karena belum ada akses yang disediakan *provider* untuk memperoleh lokasi bagi *user* yang akurat.
- f. Database lokasi yang tersedia yaitu data – data Minimarket di kota malang.
- g. Informasi yang diperoleh user dapat berupa *text* dan berupa *image* peta malang dari fasilitas yang di inginkan oleh *user*.
- h. Jenis *mobile phone* yang digunakan adalah *platform* berbasis Android.

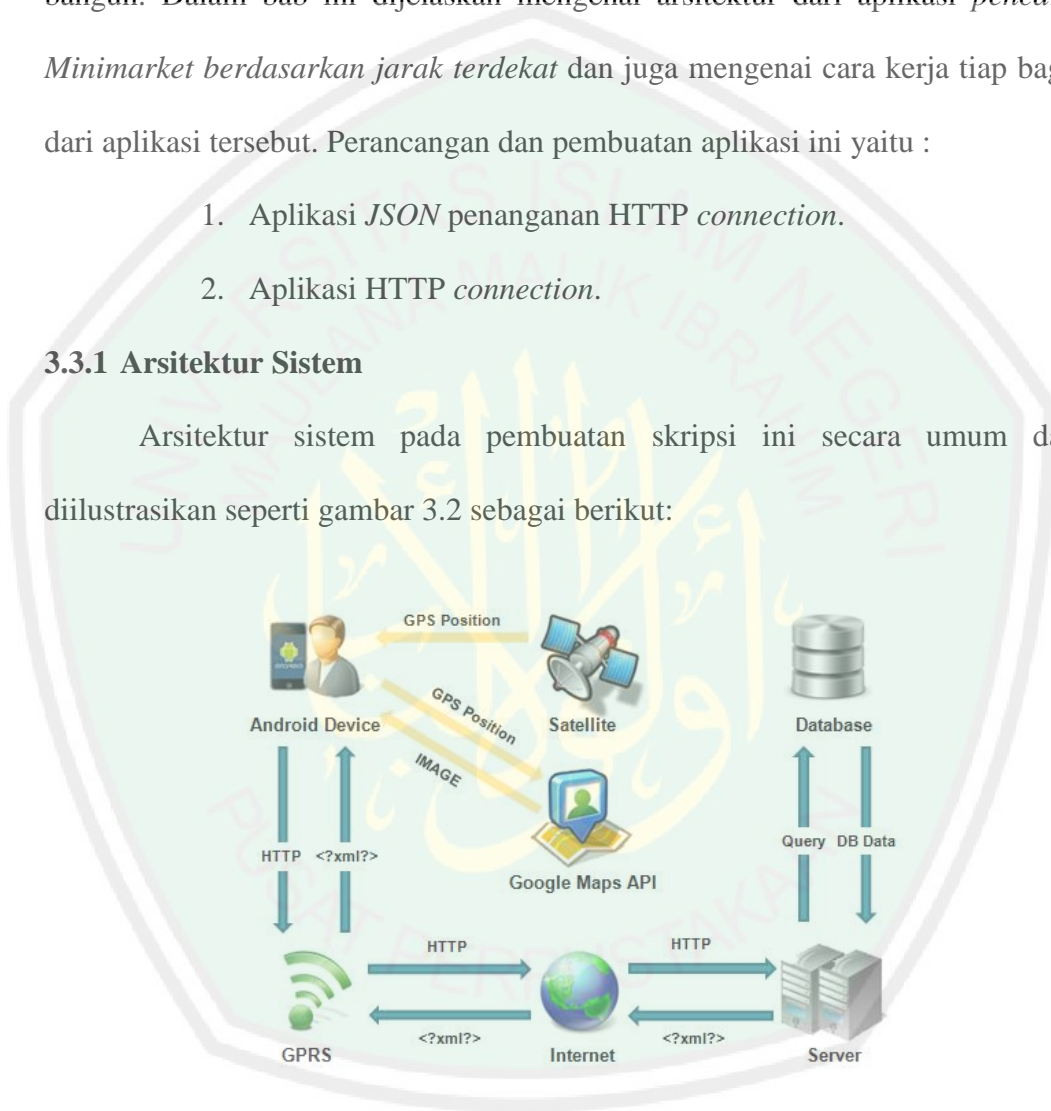
### 3.3 Desain Sistem

Desain sistem akan menjabarkan sebuah rancangan sistem yang akan dibangun. Dalam bab ini dijelaskan mengenai arsitektur dari aplikasi *pencarian Minimarket berdasarkan jarak terdekat* dan juga mengenai cara kerja tiap bagian dari aplikasi tersebut. Perancangan dan pembuatan aplikasi ini yaitu :

1. Aplikasi *JSON* penanganan *HTTP connection*.
2. Aplikasi *HTTP connection*.

#### 3.3.1 Arsitektur Sistem

Arsitektur sistem pada pembuatan skripsi ini secara umum dapat diilustrasikan seperti gambar 3.2 sebagai berikut:



**Gambar 3.2** Arsitektur sistem

*Server* menerima input dari *client* berupa koordinat *longitude* dan *latitude* *user* berada. Untuk memperoleh koordinat posisi, *user* dapat menggunakan *device* tambahan yaitu *GPS receiver*. Dalam pembuatan skripsi ini menggunakan *GPS receiver*, yang digunakan untuk *handle* *GPS* sehingga

dalam memperoleh koordinat posisi, *user* hanya perlu melihat informasi yang ditampilkan pada layar *mobile phone*. Client dapat *request* ke *server* melalui cara koneksi dan aplikasi yang digunakan *client*. Tabel 3.1 Tabel penanganan *Service Client Server* menggunakan JSON.

**Tabel 3.1** Penanganan *Service Client Server*

Aplikasi <i>Client</i>	<i>Server</i>
JSON Object connection	JSON Array handle

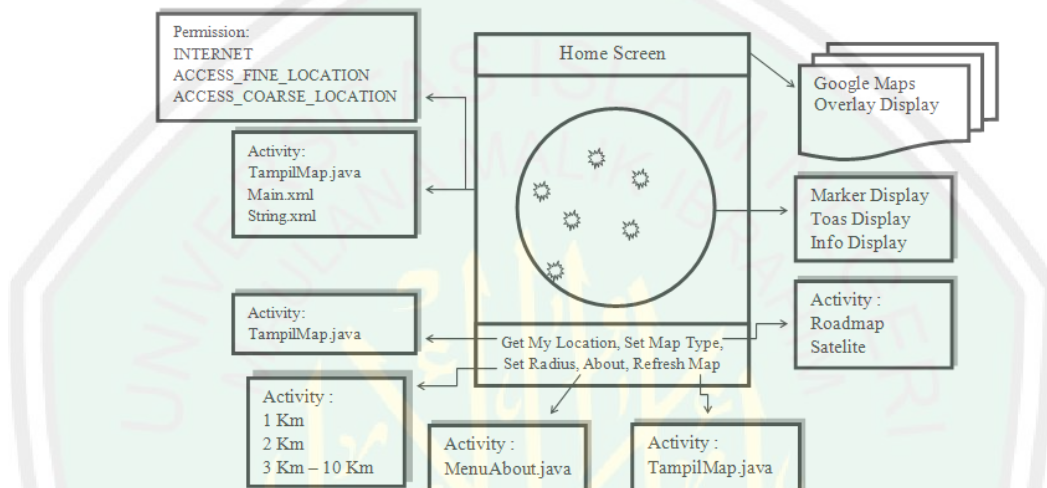
Deskripsi sistem dan cara kerjanya yang telah di buat pada skripsi ini secara umum adalah sebagai berikut :

Sistem terdiri dari *client* dan *server*.

- a. Aplikasi *client* berjalan pada *mobile phone* bersistem operasi Android, dan mengirim koordinat posisi GPS ke server untuk memperoleh data yang diperlukan.
- b. Data yang di terima *client* akan di olah dengan *Formula Haversine* untuk menentukan jarak dan radius sesuai dengan kebutuhan *user* sebelum di tampilkan pada *screen*.
- c. Bagian *server* berperan sebagai penerima *request* dari *client*, dan mengirimkan informasi atau data yang di minta kembali ke *client*. Untuk menampilkan *image* peta pada *client*. *Client* menggunakan peta yang ditangani aplikasi dari *server Google Maps* dan *server lokal*.

### 3.3.2 Rancangan Sistem

Rancangan Sistem Aplikasi yang akan di buat adalah seperti rancangan gambar 3.3 di bawah ini adapun modul - modul java yang akan di buat meliputi *class*, *activity*, *permission* dan *marker display*.



Gambar 3.3 Rancangan sistem

#### A. Permission

*Permission* adalah sebuah perintah untuk memanggil *service* pada android untuk digunakan pada aplikasi *permission* ini terdapat pada file *AndroidManifest.xml* diantaranya *permission* - *permission* yang digunakan sebagai berikut :

##### a. *INTERNET*

Adalah *permission* yang digunakan untuk memungkinkan aplikasi untuk membuka socket jaringan.

##### b. *ACCESS\_FINE\_LOCATION*

Adalah *permission* yang digunakan untuk memungkinkan aplikasi untuk mengakses lokasi menggunakan sinyal GPS satelit.

c. *ACCESS\_COARSE\_LOCATION*

Adalah *permission* yang digunakan untuk Memungkinkan aplikasi untuk mengakses lokasi menggunakan *Cell-ID* dan *WiFi*.

**B. Activity**

*Activity* adalah sebuah kelas yang digunakan untuk membuat sebuah *layout* yang akan ditampilkan dalam layar *handphone* di mana *layout* ini berisi *interface – interface* seperti tombol, *listview*, *combobox* yang bertujuan agar pengguna *user* dapat berinteraksi dengan aplikasi. Setiap *Activity* mewakili satu layar. Dalam aplikasi yang akan di buat memiliki beberapa *Activity* diantaranya:

a. *TampilMaps.java*

Merupakan *Activity* yang berfungsi untuk menampilkan gambar peta dari *Google Maps* pada layar *handphone*.

b. *Main.xml*

Adalah digunakan untuk menggambar *layout* dan *user interface* di *Android*, dengan menggunakan format bahasa *xml* (*Extensible Markup Language*).

c. *String.xml*

Adalah sebagai pembantu dari *main.xml* agar *main.xml* terlihat lebih rapi dan tidak terlalu panjang dan memudahkan kita dalam membacanya maka bisa di buat *string object* pada *string.xml* tinggal memanggil nama parameternya saja.

d. Get my location

Dalam menu untuk menentukan di mana lokasi client dengan bantuan Gps, lalu setelah lokasi user ditemukan baru akan muncul marker Minimarket namun, belum muncul jarak terdekat dengan posisi client.

e. Set map type

Pada set map type, user dapat mengubah atau mengganti tipe map sesuai kebutuhan. Ada dua tipe map yang tersedia yaitu Roadmap dan Satelite.

f. Set Radius

Di menu ini user mengatur radius untuk menentukan berapa radius kilometer yang marker Minimarket yang ingin di munculkan di map. Di sini terdapat pilihan radius mulai dari 1 Km sampai 10 Km.

g. Refresh Map

Menu ini digunakan untuk me-*refresh* peta setelah dilakukan pengaturan. *Refresh Map* bertujuan untuk menampilkan icon *minimarket* dan melakukan kalkulasi yang menampilkan jarak terdekat.

h. Direction

Aksi digunakan untuk melakukan *direction* dan menampilkan marker jalan dan icon *minimarket* yang di tuju.

i. About

Menu ini berisi tentang pembuat aplikasi dan tahun berapa aplikasi ini di buat, versi berapa aplikasi tersebut di buat serta jika ada link ke website pembuat aplikasi.

j. Display

Pada display aplikasi terdapat *Toas* merupakan sebuah *pop-up* yang akan muncul beberapa saat untuk menampilkan informasi jarak serta jenis *Minimarket*.

### 3.4 UML (*Unified Modelling Language*)

Desain sistem dengan menggunakan UML (*Unified Modelling Language*), untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan, baik secara manual maupun komputerisasi.

#### 3.4.1 Use Case

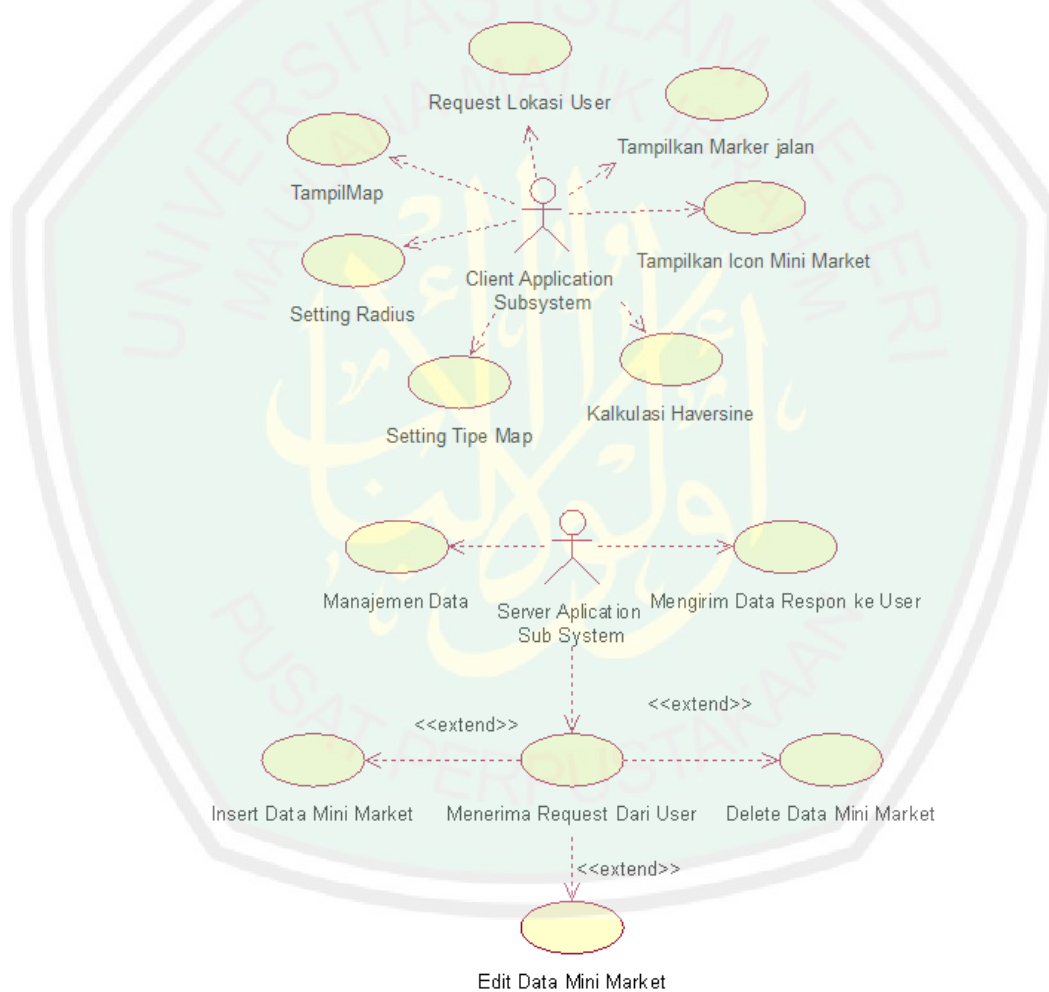
*Use case* model merupakan pemodelan struktural yang mencerminkan fungsionalitas sistem dan merupakan kebutuhan fungsional yang digambarkan dari sudut pandang *user* pada sebuah sistem.

Pada aplikasi pencarian lokasi *Minimarket* berdasarkan jarak terdekat pada *mobile phone*, terdapat dua *actor* yang mempunyai peran dan fungsinya masing – masing.

- a. *Client* adalah seseorang yang melakukan aktifitas dengan mengaktifkan *system* GPS, dalam menjalankan aplikasi pencarian lokasi *Minimarket* berdasarkan jarak terdekat pada *mobile phone*.

- b. *Server* adalah suatu *system* yang dapat menyimpan, menampilkan dan memberikan *respons* serta hasil untuk setiap aksi dari *client* dalam penggunaan aplikasi melalui *maps*.

Berikut ini gambaran *Use Case* secara umum pada aplikasi pencarian *Minimarket* terdekat. Lihat gambar 3.4 di bawah ini.



**Gambar 3.4** *Use case* aplikasi pencarian *minimarket*

Penjelasan *Use Case* pada gambar 3.4 di atas sebagai berikut:

***Server Application Subsystem:***

**A. Management Data**

**1. *Insert Data Minimarket***

*Case administrator Insert Data Minimarket* menjelaskan bahwa administrator dapat melakukan aktifitas manajemen data yang berupa : *insert* data Minimarket.

**2. *Update Data Minimarket***

*Case administrator Update Data Minimarket* menjelaskan bahwa administrator dapat melakukan aktifitas manajemen data yang berupa : *update* data Minimarket.

**3. *Delete Data Minimarket***

*Case administrator Delete Data Minimarket* menjelaskan bahwa administrator dapat melakukan aktifitas manajemen data yang berupa : *delete* data Minimarket.

**B. Menerima *Request User***

Menggambarkan suatu aktifitas penerimaan *respons* terhadap permintaan *User* ketika seorang user membutuhkan satu atau lebih informasi mengenai lokasi Minimarket.

**C. Mengirim Data *Respon* ke *User***

Menggambarkan proses pengiriman data yang dibutuhkan *User* setelah *server* menerima *request user*.

***Client Application Subsystem:***

1. ***Request lokasi user***

*Use Case* ini menggambarkan suatu aktifitas dalam proses permintaan koordinat atau posisi *user* kepada sistem satelit GPS.

2. ***Tampil Map***

*Use Case* ini menggambarkan suatu aktifitas dalam proses menampilkan gambar peta atau map dari *server Google Maps*.

3. ***Tampilkan Icon Minimarket***

*Use Case* ini menggambarkan suatu aktifitas dalam proses menampilkan *icon* semua *minimarket* yang ada di *server* sesuai radius *default*.

4. ***Setting Tipe Map***

*Use Case* ini menggambarkan suatu aktifitas dalam proses mengatur tipe *map* yaitu *satellite* atau *roadmap 2D*.

5. ***Setting Radius***

*Use Case* ini menggambarkan suatu aktifitas dalam proses mengatur radius dari 1 Km hingga 10 Km.

6. ***Kalkulasi Haversine***

*Use Case* ini menggambarkan suatu aktifitas dalam melakukan proses perhitungan dengan menggunakan *Haversine Formula*.

7. ***Tampilkan Marker Jalan***

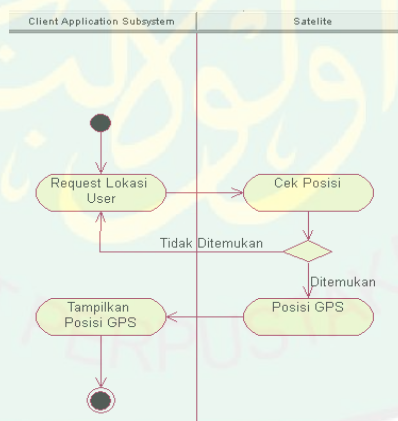
*Use Case* ini menggambarkan suatu aktifitas dalam melakukan proses menampilkan *marker* jalan *minimarket* yang akan di lalui *user*.

### 3.5 Activity Diagram

*Activity diagram* merupakan bentuk *flow diagram* yang memodelkan alur kerja sebuah proses dan urutan aktivitas dalam sebuah proses. *Diagram* ini mirip dengan *flowchart* karena dapat memodelkan sebuah alur kerja dari satu aktifitas ke aktifitas lainnya. *Activity diagram* juga berguna untuk menggambarkan perilaku parallel atau bagaimana perilaku dalam berbagai *use case* berinteraksi.

#### 3.5.1 Activity Diagram Client Request Lokasi

*Activity Diagram* di bawah ini menjelaskan proses *request* posisi koordinat *longitude* dan *latitude* dari GPS untuk mengetahui posisi pengguna. Lihat gambar 3.5 berikut ini.



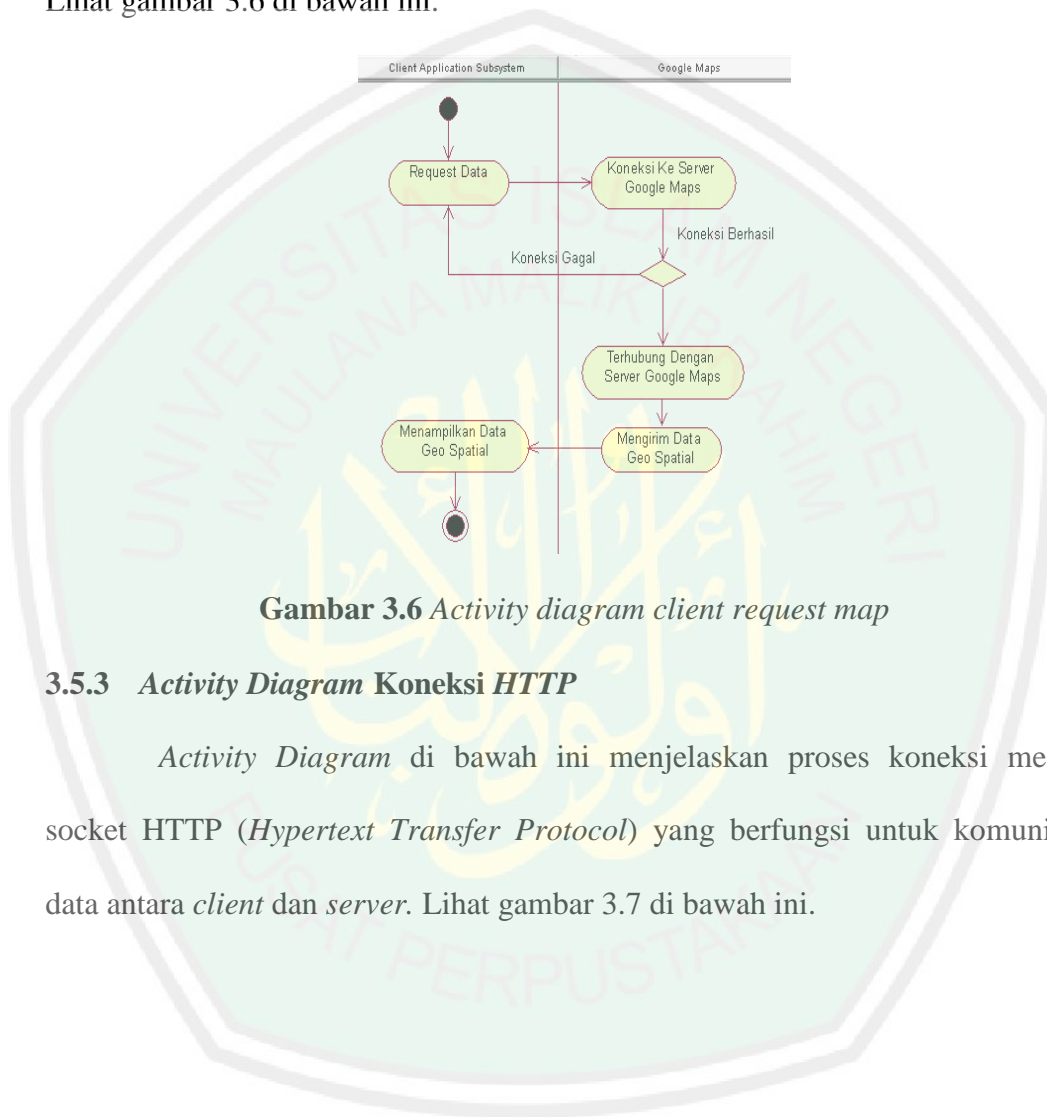
**Gambar 3.5** Activity diagram client request lokasi

#### 3.5.2 Activity Diagram Client Request Map

*Activity Diagram* di bawah ini menjelaskan proses menampilkan peta dari *server Google Maps* ke aplikasi klien, yaitu diawali dengan permintaan tampilan peta kemudian dilanjutkan koneksi ke *server Google Maps* jika terhubung klien akan mengirimkan permintaan peta ke *server Google Maps* sesuai

dengan posisi GPS yang di terima dari satelit kemudian google maps mengirimkan kembalian berupa data *Geo Spatial* untuk ditampilkan berupa peta.

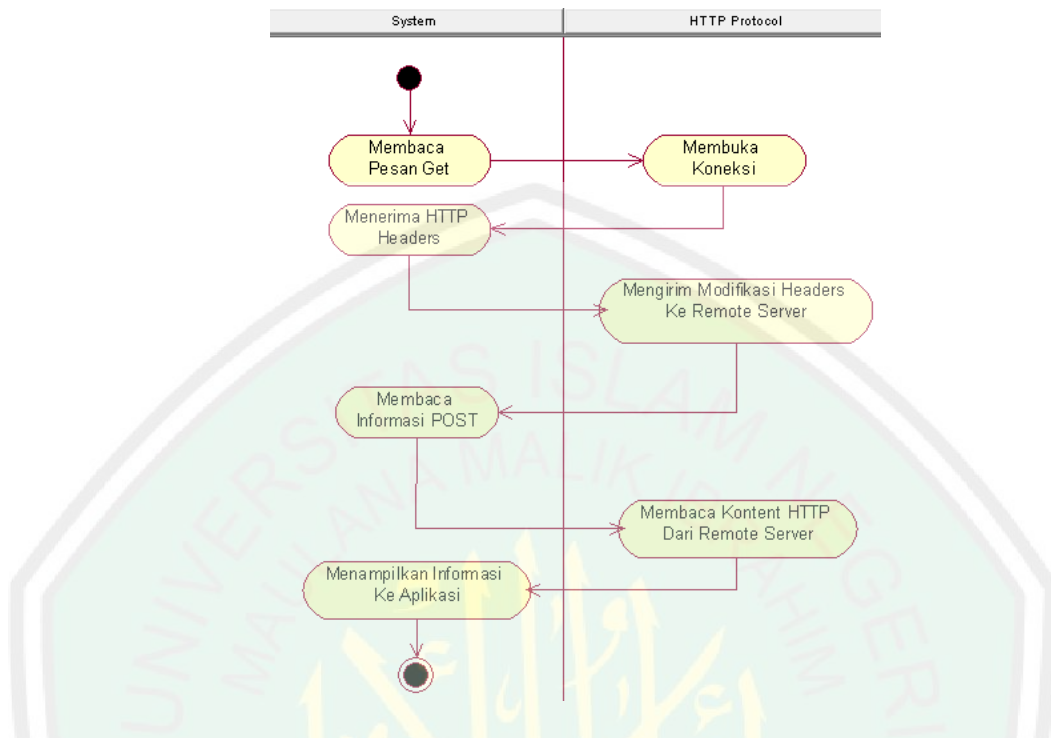
Lihat gambar 3.6 di bawah ini.



**Gambar 3.6** Activity diagram client request map

### 3.5.3 Activity Diagram Koneksi HTTP

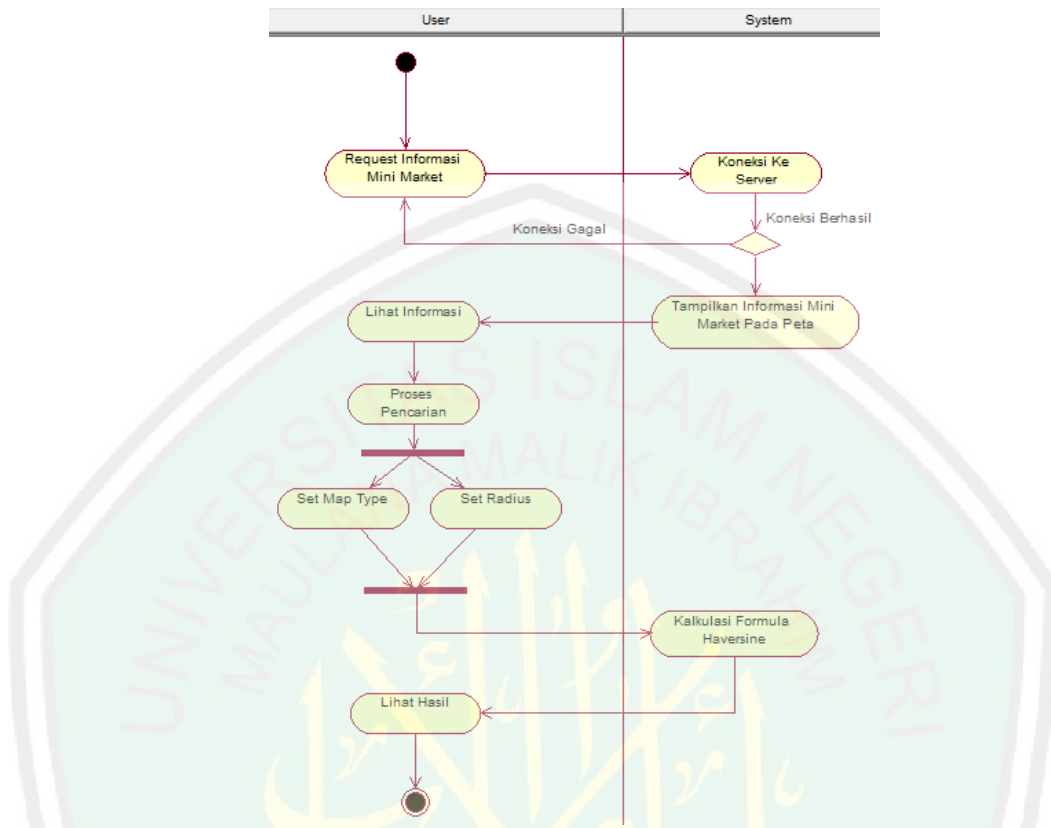
Activity Diagram di bawah ini menjelaskan proses koneksi melalui socket HTTP (*Hypertext Transfer Protocol*) yang berfungsi untuk komunikasi data antara *client* dan *server*. Lihat gambar 3.7 di bawah ini.



**Gambar 3.7** Activity diagram koneksi HTTP

#### 3.5.4 Activity Diagram Request Data Minimarket

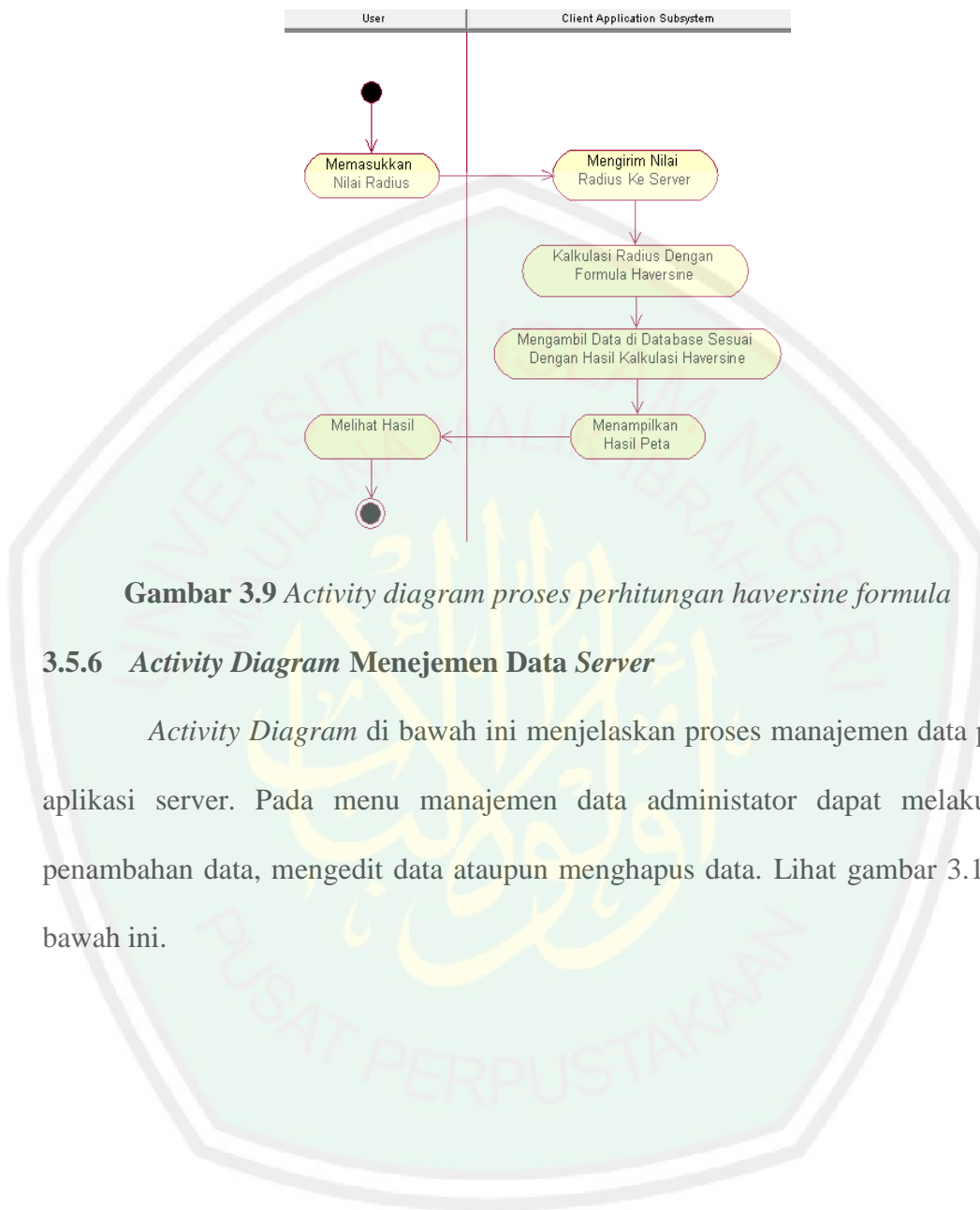
Activity Diagram di bawah ini menjelaskan proses *request* data Minimarket pertama user akan meminta informasi *minimarket* kemudian system akan membuka koneksi ke server melalui koneksi *http* jika koneksi berhasil maka akan diteruskan mengambil data dari *database* untuk ditampilkan ke peta. Selanjutnya *user* dapat melihat informasi yang ditampilkan dan memilih pilihan menu untuk menyaring informasi lebih detail lagi. Yaitu berdasarkan set tipe peta dan set radius sesuai dengan yang di *input* oleh *user*. Selanjutnya akan di kalkulasi oleh *haversine formula* dan ditampilkan hasilnya. Lihat gambar 3.8 di bawah ini.



**Gambar 3.8** Activity diagram proses request data minimarket

### 3.5.5 Activity Diagram Proses Perhitungan Haversine Formula

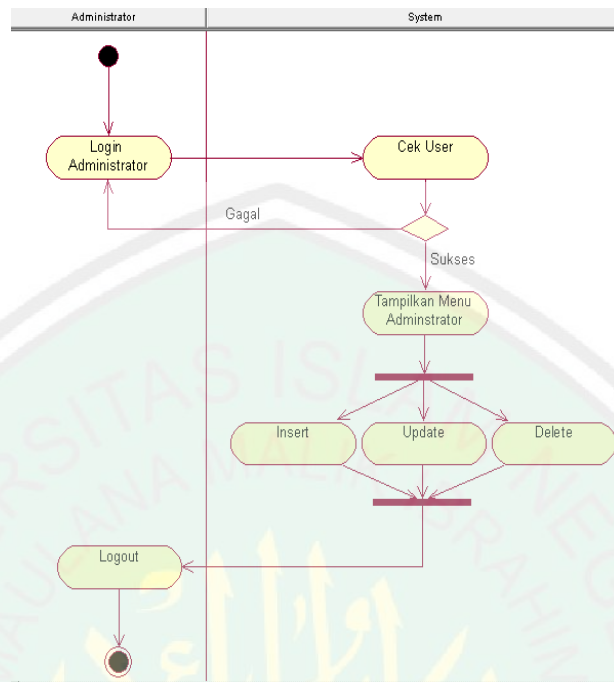
Activity Diagram di bawah ini adalah proses perhitungan jarak radius dengan menggunakan rumus Haversine proses pertama yaitu di mulai *user* memasukan nilai radius dalam satuan kilometer kemudian diteruskan dengan mengirim nilai radius ke server lalu query rumus Haversine di client kemudian dengan query rumus Haversine didapatkan hasil data dari *database* untuk ditampilkan ke peta. Lihar gambar 3.9 berikut ini.



**Gambar 3.9** Activity diagram proses perhitungan haversine formula

### 3.5.6 Activity Diagram Manajemen Data Server

Activity Diagram di bawah ini menjelaskan proses manajemen data pada aplikasi server. Pada menu manajemen data administrator dapat melakukan penambahan data, mengedit data ataupun menghapus data. Lihat gambar 3.10 di bawah ini.

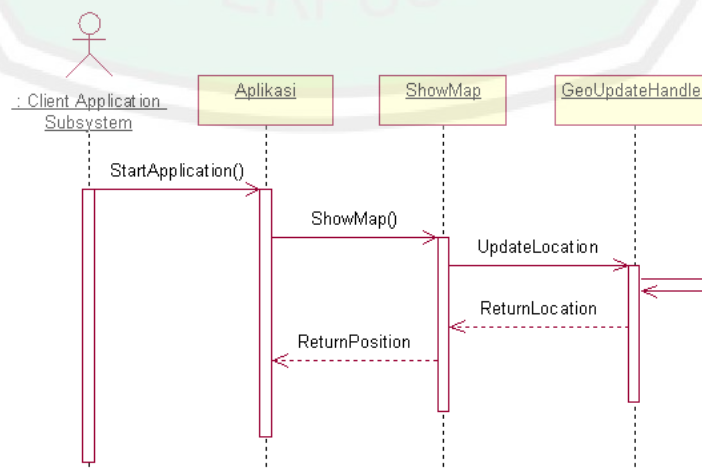


Gambar 3.10 Activity diagram manajemen data server

### 3.6 Sequence Diagram

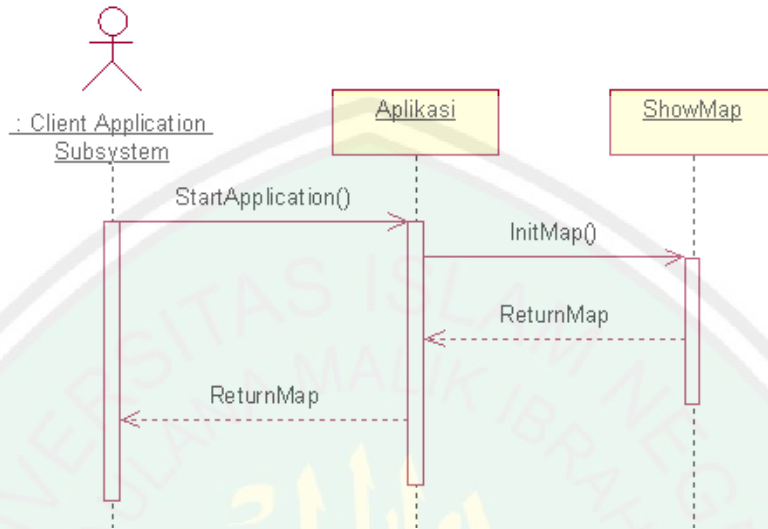
Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Dalam sequence diagram diperlihatkan tahap demi tahap apa saja yang seharusnya terjadi untuk menghasilkan sesuatu di dalam Use case.

#### 3.6.1 Sequence Diagram Client Request Lokasi



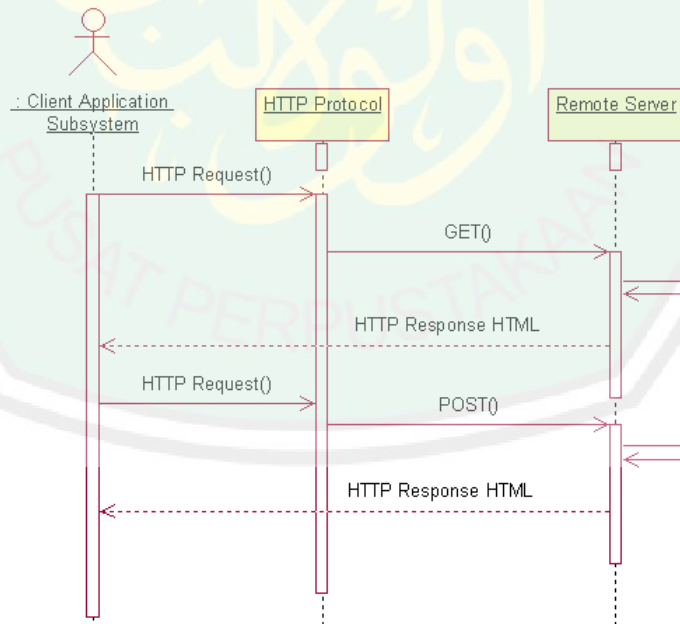
Gambar 3.11 Sequence diagram client request lokasi

### 3.6.2 Sequence Diagram Client Request Map



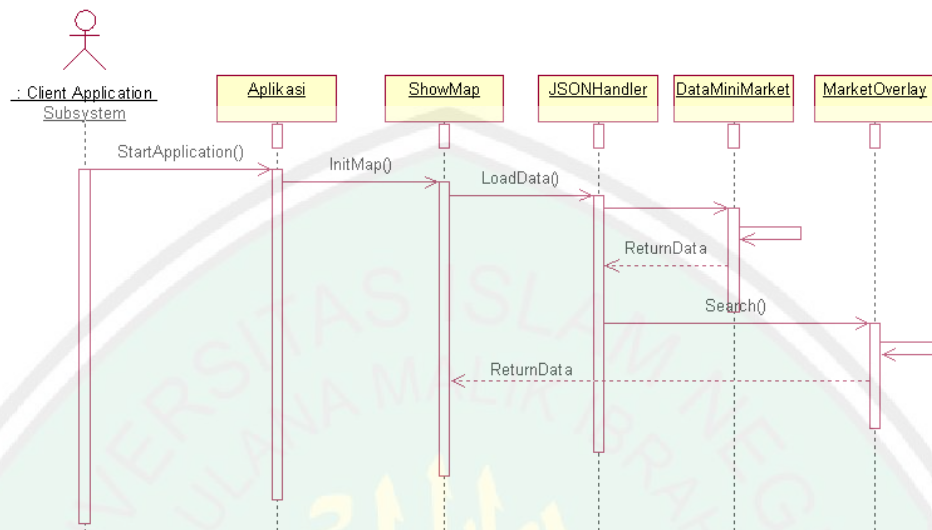
Gambar 3.12 Sequence diagram client request map

### 3.6.3 Sequence Diagram Koneksi HTTP



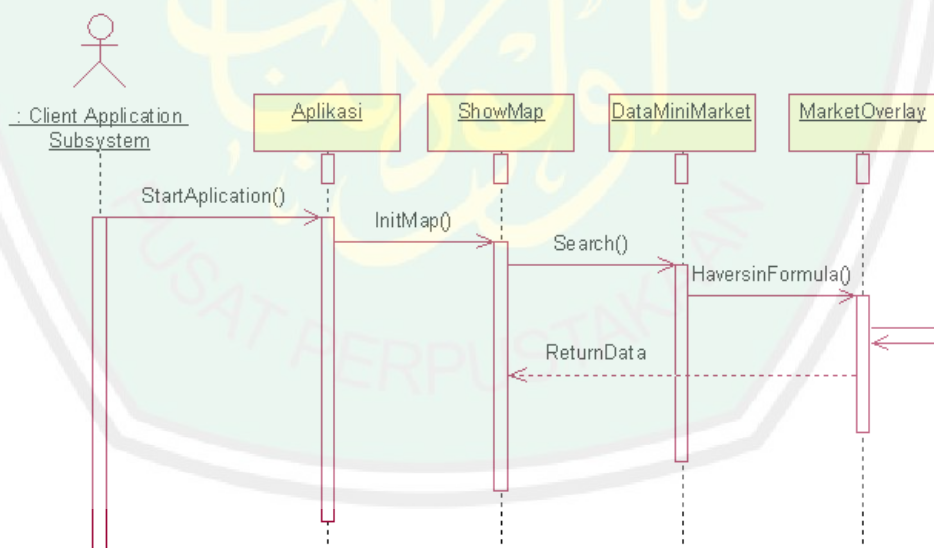
Gambar 3.13 Sequence diagram koneksi HTTP

### 3.6.4 Sequence Diagram Request Data Minimarket



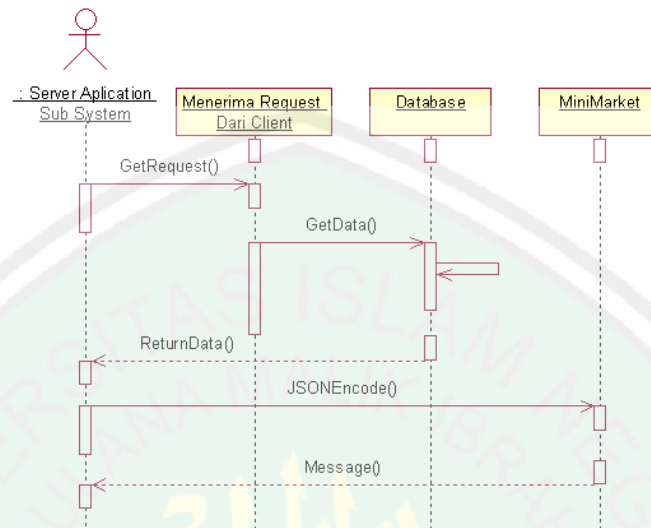
Gambar 3.14 Sequence diagram request informasi minimarket

### 3.6.5 Sequence Diagram Perhitungan Haversine Formula



Gambar 3.15 Sequence diagram perhitungan haversine formula

### 3.6.6 Sequence Diagram Server Application Subsystem

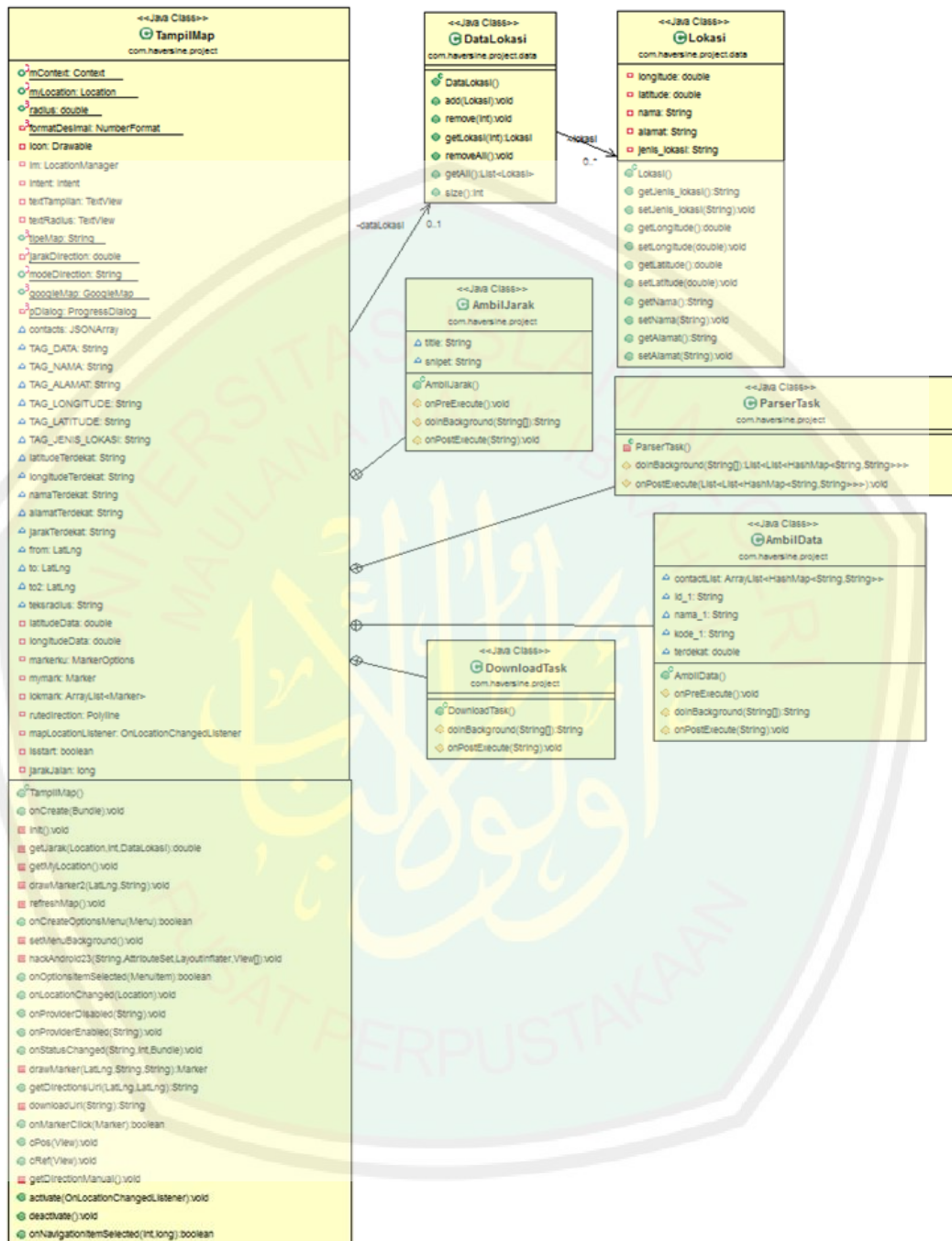


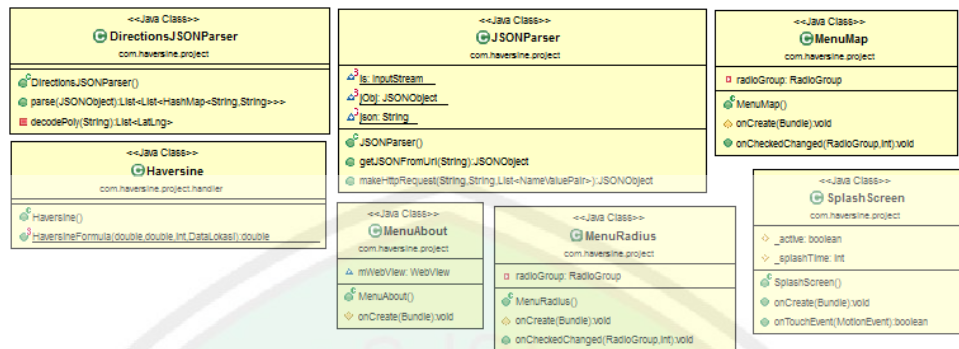
**Gambar 3.16** Sequence diagram server application subsystem

### 3.7 Class Diagram

*Class Diagram* adalah gambaran struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, perwarisan, asosiasi, dan lain – lain.

*Class Diagram* menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan, pada gambar 3.17 di bawah ini ditunjukkan gambaran tentang *Class Diagram* aplikasi yang akan di buat.





**Gambar 3.17** Class diagram aplikasi pencarian minimarket

Pada gambar 3.17 di atas terdapat dua puluh *class* yang digunakan, antara lain:

a. ***Class DirectionJSONParser***

*Class ini berfungsi untuk memarsing data direction dari google ke aplikasi sebelum ditampilkan.*

b. ***Class JSONParser***

*Class ini digunakan untuk mengubah String JSON menjadi objek Array antara client dan server.*

c. ***Class MenuMap***

*Class ini digunakan untuk menampilkan menu-menu yang ada di aplikasi seperti menu lokasiku, menu tipe peta, menu seting radius, menu tentang dan menu refresh map.*

d. ***Class Haversine***

*Class ini digunakan untuk melakukan kalkulasi/perhitungan menggunakan Haversine Formula untuk menentukan radius dan jarak terdekat.*

e. ***Class MenuAbout***

*Class ini digunakan untuk menampilkan menu tentang yang berisi deskripsi aplikasi, nama pembuat aplikasi dll.*

**f. Class MenuRadius**

*Class* ini digunakan untuk menampilkan menu radius yang diantaranya terdapat pilihan radius 1 km hingga 10 km.

**g. Class SplashScreen**

*Class* ini digunakan untuk menampilkan splash screen atau tampilan awal pada saat aplikasi di buka yang berisi nama aplikasi, logo aplikasi dan pembuat aplikasi.

**h. Class TampilMap**

*Class* ini digunakan untuk menampilkan user interface dari aplikasi dan menampilkan peta serta menu-menu aplikasi. *Class* ini yang mengakomodir semua class untuk menjalankan menu-menu yang ada di aplikasi.

**i. Class DataLokasi**

*Class* ini digunakan untuk membuat *arraylist* dari lokasi-lokasi Minimarket yang sudah di ambil dari server.

**j. Class Lokasi**

*Class* ini digunakan untuk menampilkan informasi lokasi yang muncul pada peta seperti longitude, latitude, nama dan alamat.

**k. Class AmbilJarak**

*Class* ini berfungsi untuk menampilkan marker jarak yang akan ditampilkan di peta setelah dilakukan kalkulasi haversine formula.

**l. Class ParserTask**

*Class* ini digunakan untuk memarsing *direction Google* untuk dijadikan ke format JSON.

**m. Class AmbilData**

*Class ini digunakan untuk mengambil data minimarket dari server sebelum ditampilkan di peta.*

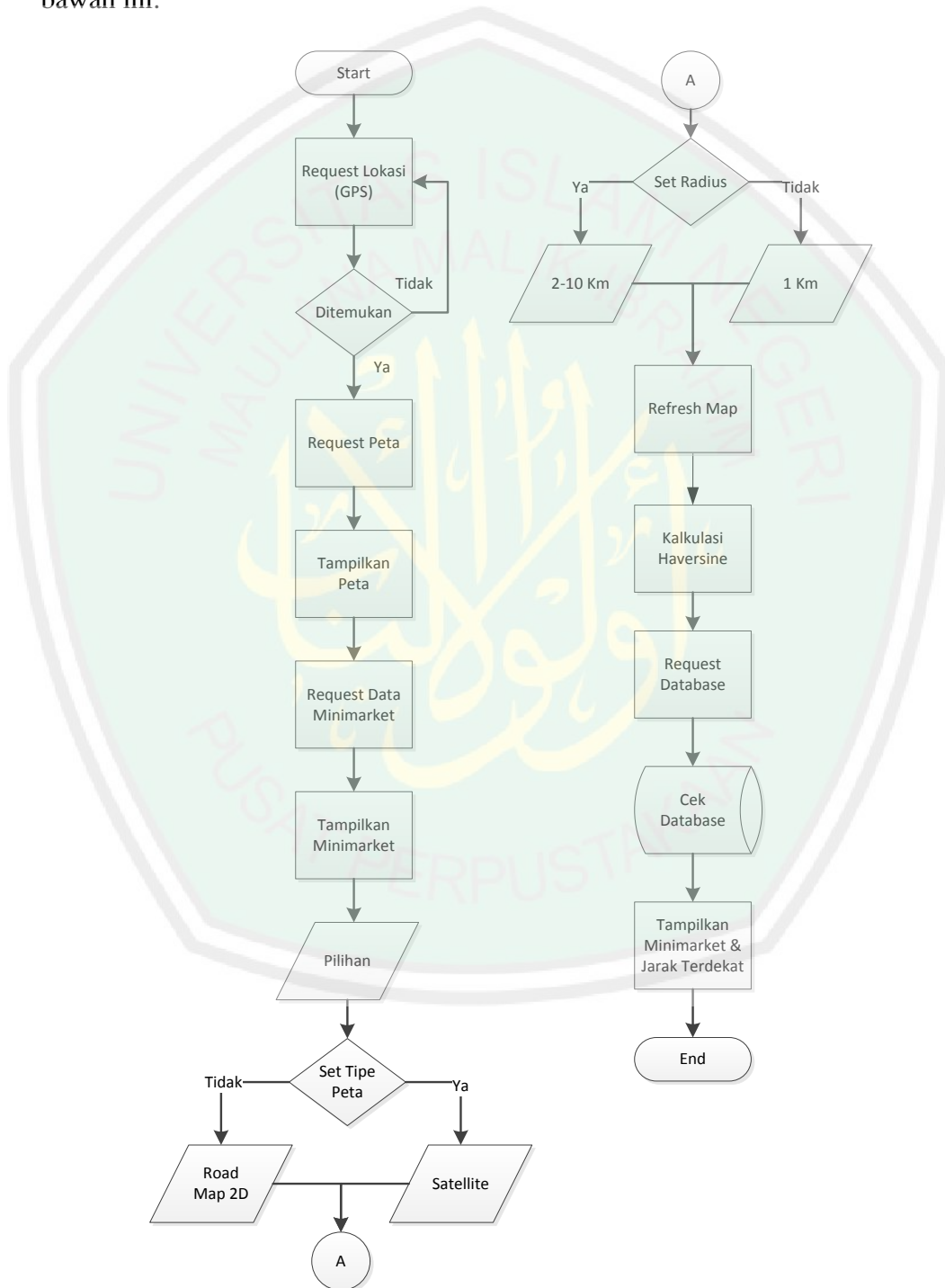
**n. Class DownloadTask**

*Class ini digunakan untuk men-download data dari url google direction.*



### 3.8 Flowchart System

Berikut ini terdapat *flowchart application subsystem*. Lihat gambar 3.18 di bawah ini.



**Gambar 3.18.** *Flowchart Application Subsystem*

Penjelasan *Flowchart Application Subsystem*:

- a. *Start* yaitu awal aplikasi dijalankan/di buka
- b. *Request lokasi GPS*. Pada step ini aplikasi me-request lokasi user menggunakan GPS
- c. *Cek posisi user*. Pada step ini aplikasi mencari posisi lokasi user. Jika belum ditemukan aplikasi kembali me-request lokasi user
- d. *Request peta*. Pada step ini aplikasi me-request peta ke server google maps
- e. *Geospatial*. Pada step ini aplikasi mengambil peta yang akan ditampilkan
- f. *Tampilkan map*. Pada step ini aplikasi menampilkan map setelah request peta ke google maps berhasil dilakukan
- g. *Request informasi minimarket*. Pada step ini aplikasi me-request informasi minimarket dan menampilkannya berupa icon
- h. *Pilihan input*. Pada step ini pada aplikasi terdapat pilihan input untuk menseting aplikasi sesuai kebutuhan sebelum melakukan perhitungan jarak menggunakan haversine formula
- i. *Seting tipe peta*. Pada step ini user menseting tipe peta sesuai kebutuhan. Ada 2 pilihan yaitu modus Satellite dan Road Map 2D
- j. *Seting radius*. Pada step ini user menseting radius sesuai dengan kebutuhan. Ada beberapa pilihan radius yaitu 1 km hingga 10 km
- k. *Refresh map*. Pada step ini *refresh map* digunakan untuk menampilkan kembali icon *minimarket* dan jarak terdekat setelah dilakukan pengaturan atau user pindah lokasi.

- l. *Kalkulasi Haversine*. Pada step ini terdapat proses perhitungan kalkulasi menggunakan Haversine Formula setelah user melakukan perubahan seting yang diinginkan
- m. *Cek request database*. Pada step ini aplikasi me-request ke database untuk mencari kesesuaian data yang kalkulasi dengan Haversine Formula
- n. *Database*. Pada step ini aplikasi mencari data yang sesuai dengan perhitungan Haversine Formula
- o. *Tampilkan informasi Minimarket & jarak terdekat*. Pada step ini aplikasi menampilkan informasi Minimarket dengan jarak terdekat beserta market jalur yang akan dilalui
- p. *End* yaitu akhir dari pencarian mini market menggunakan haversine formula berdasarkan jarak terdekat.

### 3.9 Perancangan Desain Database

Dalam pembuatan aplikasi ini dibutuhkan desain *database* untuk menyimpan data yang akan digunakan dalam proses sistem aplikasi pencarian *minimarket*. Desain *database* ini menjelaskan tabel dan *field* yang akan digunakan. Berikut adalah tabel yang digunakan untuk proses sistem aplikasi pencarian *minimarket*.

#### 3.9.1 Tabel Database Minimarket

Tabel *Database Minimarket* adalah tabel data yang digunakan untuk menyimpan data-data terkait informasi *minimarket*. Lihat tabel 3.2 berikut.

**Tabel 3.2** Perancangan database aplikasi pencarian minimarket terdekat

No.	Nama Field	Type Data	Keterangan
1	Id	Int	Menyimpan kode sebagai primary key dan unik untuk kode minimarket
2	nama	Varchar	Menyimpan data nama minimarket
3	alamat	Text	Menyimpan data alamat minimarket
4	jenis_lokasi	Varchar	Menyimpan data jenis lokasi
5	latitude	Double	Menyimpan lokasi minimarket berdasarkan latitude
6	longitude	Double	Menyimpan lokasi minimarket berdasarkan longitude

### 3.10 Desain Interface

Setelah perancangan UML, *entity relation diagram*, *flowchart* dan perancangan database selesai, maka selanjutnya di buat *desain form* yang menggambarkan sekilas tentang *interface* aplikasi ini. Diantaranya adalah :

#### 3.10.1 Desain Form Awal

Dalam *form* awal terdapat layar untuk menampilkan *Splash Screen* yang berisi nama aplikasi, logo aplikasi, dan pembuat aplikasi. Lihat gambar 3.19 berikut ini.

**Gambar 3.19** Form tampilan awal

### 3.10.2 Desain *Form Tampilan Utama*

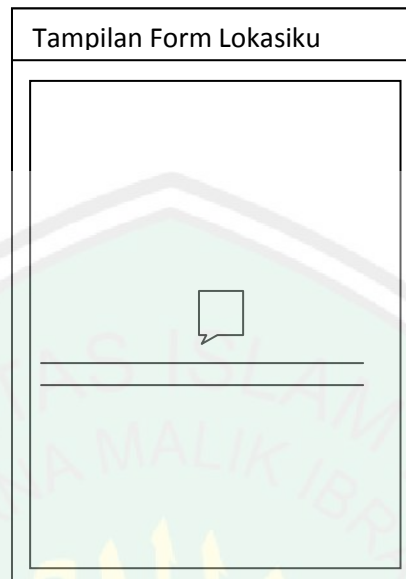
Dalam *form* ini digunakan untuk menampilkan gambar peta dan fasilitas umum yang ada di sekitar posisi pengguna dalam *form* ini terdapat lima buah tombol yaitu tombol lokasiku, tipe peta, seting radius, tentang dan tipe jalur. Kelima tombol ini sebagai menu untuk mencari fasilitas tempat umum dan setting tampilan peta. Lihat gambar 3.20 berikut ini.



**Gambar 3.20** *Form tampilan utama*

### 3.10.3 Desain *Form Menu Lokasiku*

*Form* ini berfungsi untuk mencari lokasi user berada dan menampilkan dalam icon sebelum menampilkan icon Minimarket yang akan di cari jarak terdekat terdekatnya. Lihat gambar 3.21 di bawah ini.

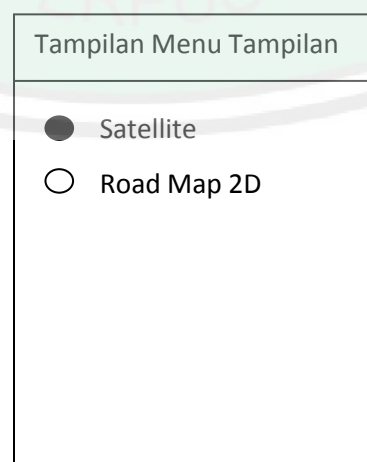


Tampilan Form Lokasiku

**Gambar 3.21** *Form tampilan menu lokasiku*

#### **3.10.4** *Desain Form Tipe Peta*

*Form* ini berfungsi sebagai menu untuk merubah tampilan peta seperti tampilan citra satelit dan gambar map atau peta dua dimensi. Tampilan menu *Satellite* menampilkan pemaparan citra peta seperti citra asli bentukan bumi. Sedangkan menu Road Map 2D menampilkan citra peta dua dimensi. Lihat gambar 3.22 di bawah ini.



Tampilan Menu Tampilan

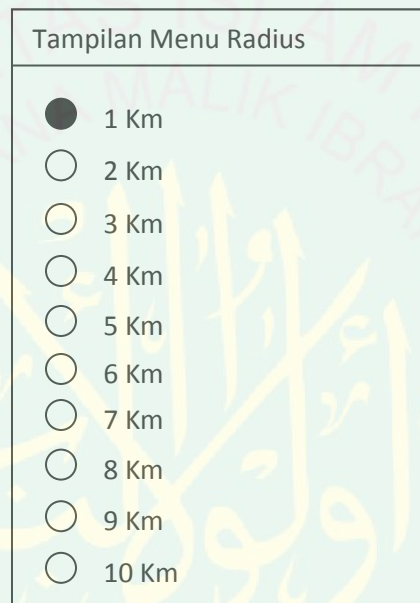
Satellite

Road Map 2D

**Gambar 3.22** *Form tipe peta*

### 3.10.5 Desain Form Setting Radius

*Form* ini berfungsi sebagai menu untuk mengatur radius untuk menampilkan informasi tempat umum sesuai dengan keinginan pengguna (*user*). Ada terdapat sepuluh pilihan radius dari 1 km hingga 10 km. Lihat gambar 3.23 di bawah ini.



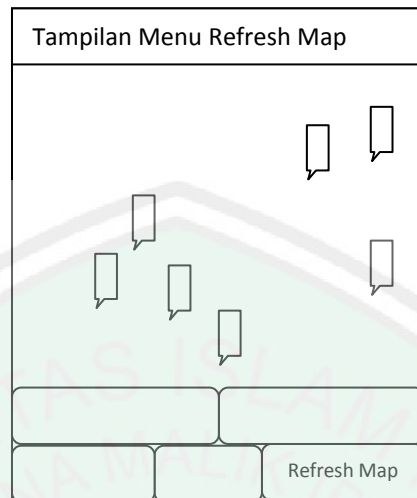
The image shows a screenshot of a form titled "Tampilan Menu Radius". It contains a vertical list of radio buttons, each followed by a radius value in kilometers. The first option, "1 Km", is selected, indicated by a solid black circle. The other options, from "2 Km" to "10 Km", are unselected, indicated by empty white circles with black outlines.

Radio Button	Radius
<input checked="" type="radio"/>	1 Km
<input type="radio"/>	2 Km
<input type="radio"/>	3 Km
<input type="radio"/>	4 Km
<input type="radio"/>	5 Km
<input type="radio"/>	6 Km
<input type="radio"/>	7 Km
<input type="radio"/>	8 Km
<input type="radio"/>	9 Km
<input type="radio"/>	10 Km

**Gambar 3.23** *Form seting radius*

### 3.10.6 Desain Form Menu Refresh Map

*Form* ini digunakan untuk merefresh atau mengambil data minimarket dari server menggunakan *packet data* atau internet dan selanjutnya ditampilkan di peta dalam bentuk icon. *Refresh map* juga digunakan untuk me-refresh icon minimarket pada saat posisi user berpindah-pindah. Lihat gambar 3.24.



**Gambar 3.24** Menu refresh map

### 3.10.7 Desain Form Menu Tentang

*Form* ini dibuat untuk memberi informasi tentang apa aplikasi ini dan tujuan dibuatnya aplikasi ini, siapa pembuatnya, alamat website pembuat, alamat email dan informasi lain yang terkait. Lihat gambar 3.25.

Tampilan Menu Tentang
<p>Aplikasi pencarian minimarket dibuat bertujuan untuk mempermudah pengguna mencari minimarket terdekat dengan posisi user saat itu. Aplikasi ini memanfaatkan teknologi google maps dan haversine formula untuk mendapatkan jarak terdekat dan radius yang optimal.</p> <p>©Alfan Kurniawan www.alfankurniawan.com</p>

**Gambar 3.25** Form tentang

### 3.10.8 Tombol *Zoom In* dan *Zoom Out*

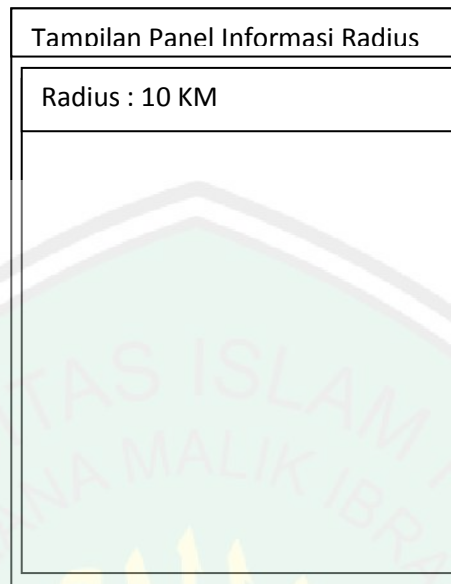
Tombol atau *button zoom in* dan *zoom out* berguna untuk memperbesar maupun memperkecil tampilan citra peta beserta icon-icon didalamnya. Tombol ini digunakan untuk memperjelas *marker* jalan yang menunjukkan minimarket dengan jarak terdekat. Lihat gambar 3.25.



**Gambar 3.26** Tombol *zoom in* dan *zoom out*

### 3.10.9 Desain *Panel Informasi Radius*

*Panel informasi radius* memberikan informasi *radius* dengan satuan kilometer. Rentang kilometer yang digunakan yaitu antara 1 kilometer hingga 10 kilometer. Radius ini sangat diperlukan untuk membatasi jangkauan minimarket yang ingin ditampilkan di peta untuk mempermudah pencarian minimarket terdekat. Lihat gambar 3.26.



**Gambar 3.27** Tampilan panel informasi radius

### 3.11 Uji Kelayakan Aplikasi

Uji kelayakan aplikasi dilakukan untuk menguji apakah sistem pada aplikasi sudah berjalan sesuai dengan fungsinya. Dalam uji kelayakan aplikasi ini terdapat 5 pertanyaan yang disesuaikan untuk menguji sistem pada aplikasi. Berikut pertanyaan pada uji kelayakan:

- a. Apakah aplikasi dapat menemukan & menampilkan lokasi *user*?
- b. Apakah aplikasi dapat menampilkan *icon minimarket* berdasarkan tipe peta?
- c. Apakah aplikasi dapat menampilkan *icon minimarket* berdasarkan *radius*?
- d. Apakah aplikasi dapat menampilkan *minimarket* dengan jarak terdekat berdasarkan satu lokasi *user*?
- e. Apakah aplikasi dapat menampilkan *minimarket* terdekat berdasarkan lokasi *user* yang variatif dan berapa waktu yang dibutuhkan?

### 3.12 Uji Kelayakan Aplikasi Berdasarkan Kuisisioner

Kuisisioner dilakukan untuk menghitung hasil survei kelayakan aplikasi. Survei ditujukan kepada pengguna dengan mengisi pernyataan sesuai dengan pengamatan pada aplikasi. Pengukuran kuisisioner ini menggunakan skala *Likert*, dimana digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang. Dengan skala *Likert*, maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrument yang dapat berupa pernyataan atau pertanyaan (Sugiyono, 2010) Berikut pernyataan pada kuisisioner:

- a. Ketertarikan dengan Aplikasi
- b. Tampilan Aplikasi
- c. Kesesuaian desain warna
- d. Kelengkapan fitur
- e. Kemudahan penggunaan fitur
- f. Penggunaan bahasa
- g. Manfaat Aplikasi
- h. Pengembangan Aplikasi
- i. Publikasi Aplikasi pada masyarakat
- j. Tanggapan pengguna secara keseluruhan

Adapun nilai dari pernyataan diasumsikan dengan huruf, maka keterangan sebagai berikut.

SB = Sangat Baik	diberi skor	5
B = Baik	diberi skor	4
C = Cukup	diberi skor	3
K = Buruk	diberi skor	2
SK = Sangat Buruk	diberi skor	1

Setelah hasil dari kuesioner dianalisis, maka akan dicari tingkat kelayakan produk yang ditentukan dari rata-rata prosentase jawaban responden, berikut tingkatan kelayakan produk:

Sangat layak	=	81% - 100%
Layak	=	61% - 80%
Biasa	=	41% - 60%
Tidak layak	=	21% - 40%
Sangat tidak layak	=	1% - 20%

Berdasarkan tingkatan tersebut, maka akan diperoleh hasil yang nanti akan digunakan dalam menetapkan tingkat kelayakan produk dari hasil kuesioner.

Uji kelayakan aplikasi di atas diharapkan dapat membantu menentukan apakah aplikasi ini sudah berjalan dengan benar sesuai fungsinya atau tidak sebelum digunakan untuk kebutuhan sehari-hari. Hasil dari uji kelayakan aplikasi di atas akan dijelaskan secara lengkap pada bab berikutnya.

## BAB IV

### IMPLEMENTASI HASIL DAN PEMBAHASAN

Pada subbab ini akan dipaparkan implementasi sistem berdasarkan rancangan program. Rancangan yang telah di bangun akan diimplementasikan ke dalam bentuk *source code* dalam bahasa pemrograman Java berbasis Android. Berikut adalah paparan implmenetasi dari perangkat lunak yang telah di bangun.

#### 4.1 Sumber Data

Data yang diperoleh untuk melakukan penelitian lokasi Mini Market di dapat secara manual. Yaitu dengan menggunakan fasilitas GPS pada Smartphone untuk mendapatkan latitude dan longitude lokasi. Jadi penulis mendatangi lokasi langsung dan mendata secara manual terlebih dahulu lalu, baru diinputkan ke database localhost.

#### 4.2 Ruang Lingkup Implementasi

Implementasi merupakan proses transformasi representasi rancangan ke bahasa pemrograman yang dapat dimengerti oleh komputer. Pada bab ini akan di bahas hal-hal yang berkaitan dengan implementasi sistem pencarian mini market.

Lingkungan implementasi yang akan dipaparkan meliputi lingkungan perangkat keras dan lingkungan perangkat lunak.

#### 4.2.1 Ruang Lingkup Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan aplikasi pencarian Mini Market berdasarkan jarak terdekat menggunakan metode *Haversine Formula* ini adalah sebagai berikut:

- a. Laptop HP Pavilion G4 1336TX
  1. Processor Intel(R) Core(TM) i3-2350M CPU @ 2.30 GHz (4 CPUs)
  2. RAM 4096 GB
  3. Harddisk dengan kapasitas 640 GB
  4. LCD 14 inch
- b. Android *device* : Smartfren Andromax C

#### 4.2.2 Ruang Lingkup Perangkat Lunak

Adapun perangkat lunak yang digunakan untuk pengembangan aplikasi ini antara lain:

1. Sistem Operasi Windows 7
2. Eclipse
3. SDK Android
4. MySQL
5. ADT Windows

## 6. Adobe Photoshop

## 7. GPS Status

Perangkat lunak di bangun dengan menggunakan bahasa pemrograman JAVA dengan berbasis Android. Hasil implementasi akan diujikan pada *mobile device* Samsung Galaxy SIII Mini. Lihar gambar 4.1.



Gambar 4.1 Samsung Galaxy SIII Mini ([gsmarena.com](http://gsmarena.com))

### 4.3 Implementasi Sistem

Pada subbab ini akan memaparkan implementasi sistem berdasarkan rancangan program. Rancangan yang telah dibangun akan diimplementasikan ke dalam bentuk *source code* dalam bahasa pemrograman Java berbasis Android. Berikut adalah paparan implementasi dari perangkat lunak yang telah dibangun.

### 4.3.1 Tahap *Request* Lokasi

Pada tahap ini aplikasi melakukan *request* lokasi pengguna menggunakan GPS. Aplikasi akan mengecek apakah GPS sudah aktif. Jika belum aktif maka, aplikasi akan memberikan peringatan dan menanyakan untuk mengaktifkan atau tidak. Jika GPS sudah aktif maka akan muncul lokasi pengguna di *map*. Namun jika GPS tidak diaktifkan maka akan ada peringatan bahwa GPS belum diaktifkan dan lokasi belum di temukan. Lihat *source code* berikut ini.

```
private void getLocation() {
    try {
        Log.v("MAP", "Get My Location");

        /* pengecekan GPS hidup / tidak */
        LocationManager manager = (LocationManager)
            getSystemService(Context.LOCATION_SERVICE);

        If (!manager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
            Log.v("MAP", "GPS Provider Disabled");

            // Ask the user to enable GPS
            AlertDialog.Builder builder = new AlertDialog.Builder(this);
            builder.setTitle("Info");
            builder.setMessage("Anda akan mengaktifkan GPS?");
            builder.setPositiveButton("Ya",
                new DialogInterface.OnClickListener() {

            public void onClick(DialogInterface dialog, int which) {
                // TODO Auto-generated method stub
                Intent i = new Intent(
                    Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                startActivity(i);
            }
            });

            builder.setNegativeButton("Tidak",
                new DialogInterface.OnClickListener() {

            public void onClick(DialogInterface dialog, int which) {
                // TODO Auto-generated method stub
                dialog.dismiss();
            }
            });

            builder.create().show();
        }
    }
}
```

### 4.3.2 Tahap Manampilkan *Map*

Pada tahap ini aplikasi menampilkan *map* yang diambil dari *Google Map*. Pada layar *device* akan ditampilkan *map* dan *icon* lokasi *user* yang sudah didapatkan dari *request* lokasi menggunakan GPS. Namun, jika GPS belum dihidupkan maka ada peringatan bahwa “lokasi belum terdefinisi”. Lihat *source code* berikut.

```
SupportMapFragment fm = (SupportMapFragment)
    getSupportFragmentManager().findFragmentById(R.id.map);
googleMap = fm.getMap();
googleMap.setMyLocationEnabled(true);
googleMap.setOnMarkerClickListener(this);
```

### 4.3.3 Tahap *Request Data Minimarket*

Pada tahap ini aplikasi melakukan *request data minimarket* ke server langsung menuju alamat/*url* yang telah ditetapkan menggunakan jaringan internet pada *device*. Setelah data didapatkan maka data akan di tata ulang dari *hash list* ke *array list*. Lihat *source code* berikut ini.

```

        url = "http://alfankurniawan.com/haversine/haversine.php";
        JSONParser jParser = new JSONParser();
        JSONObject json = jParser.getJSONFromUrl(url);
    try {
        contacts = json.getJSONArray(TAG_DATA);
        for (int i = 0; i < contacts.length(); i++) {
            JSONObject c = contacts.getJSONObject(i);
            HashMap<String, String> map = new HashMap<String, String>();

            // Storing each json item in variable
            id_1 = c.getString("id").trim();
            nama_1 = c.getString(TAG_NAMA).trim();
            String latitude_1 = c.getString(TAG_LATITUDE).trim();
            String longitude_1 = c.getString(TAG_LONGITUDE).trim();
            String akses_1 = c.getString(TAG_JENIS_LOKASI).trim();
            String alamat_1 = c.getString(TAG_ALAMAT).trim();

            map.put("id", id_1);
            map.put("nama", nama_1);
            map.put("latitude", latitude_1);
            map.put("longitude", longitude_1);
            map.put("jenis", akses_1);
            map.put("alamat", alamat_1);

            // adding HashMap to ArrayList
            contactList.add(map);
        }
    }

```

#### 4.3.4 Tahap Kalkulasi *Haversine Formula*

Pada tahap ini aplikasi melakukan perhitungan/kalkulasi menggunakan *Haversine Formula* untuk menentukan lokasi *minimarket* terdekat dengan lokasi *user*. Pada *source code* berikut ini akan ditampilkan yaitu *class Haversine.java* dan *class* tersebut akan dipanggil ke dalam *class TampilMap.java*. Setelah itu akan ditampilkan hasilnya di layar *device*. Lihat *source code* berikut.

```

public class Haversine {
    public static double HaversineFormula(double latPos, double lonPos,
        int index, DataLokasi dataLokasi) {
        final int R = 6371;
        double latFas = dataLokasi.getLokasi(index).getLatitude();
        double lonFas = dataLokasi.getLokasi(index).getLongitude();
        double latDistance = toRad(latFas-latPos);
        double lonDistance = toRad(lonFas-lonPos);
        double a = Math.sin(latDistance / 2) * Math.sin(latDistance / 2) +
            Math.cos(toRad(latPos))*Math.cos(toRad(latFas)) *
            Math.sin(lonDistance / 2) * Math.sin(lonDistance / 2);
        double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
        double distance = R * c;
        return distance;
    }
}

```

#### 4.3.5 Tahap Membuat *Marker/Direction* di Map

Pada tahap ini aplikasi sudah melakukan kalkulasi dan mengetahui jarak *minimarket* terdekat dengan lokasi *user*. Setelah *minimarket* terdekat diketahui maka aplikasi akan menampilkan *marker/rute/direction* berwarna biru untuk menunjukkan jalan lokasi *user* menuju lokasi *minimarket* terdekat. Lihat *source code* berikut.

```

LatLng tambah = new LatLng(point.latitude, point.longitude);
// Creating MarkerOptions
markerku = new MarkerOptions();
markerku.position(tambah);

markerku.icon(BitmapDescriptorFactory.fromResource(R.drawable.lokasiku));
markerku.title(keterangan);
mymark= googleMap.addMarker(markerku);

```

#### 4.3.6 Tahap Menampilkan Jarak Terdekat

Pada tahap terakhir yaitu aplikasi menampilkan jarak terdekat. Jarak terdekat disini ditampilkan pada *pop-up* jika *user* menekan *icon minimarket* terdekat yang telah ditemukan. Di dalam *pop-up* tersebut muncul informasi alamat *minimarket*,

nama *minimarket*, jarak *minimarket* berdasarkan kilometer dan meter. Berikut *source code* dalam tahap ini.

```

if (distance<=Double.parseDouble(teksradius)){

    String alamat= contactList.get(x).get("alamat");
    String keterangan = "Alamat : "+alamat+" \n- Jarak Haversine
                        -"+ "\nJarak (km): "+distance+"
                        km"+ "\nJarak (m): "+distance2+" m";

    lokmark.add(drawMarker(marker, nama, keterangan ));

    if (distance<terdekat){

        latitudeTerdekat = String.valueOf(marker.latitude);
        longitudeTerdekat = String.valueOf(marker.longitude);
        alamatTerdekat =alamat;
        namaTerdekat =nama;
        terdekat = distance;
        to = new LatLng(marker.latitude, marker.longitude);
    }
}

```

#### 4.3.7 Source Code Koneksi Dengan Database

Berikut ini *source code* koneksi *website* dengan *database* pada *web hosting* yang digunakan untuk menampilkan data mentah sebelum di olah dan di tata oleh aplikasi. Pada tahap ini semua data di ambil dari *database* lalu di kirim ke aplikasi, setelah itu baru di olah dan di hitung oleh aplikasi menggunakan *haversine formula*. Lihat *source code* berikut ini.

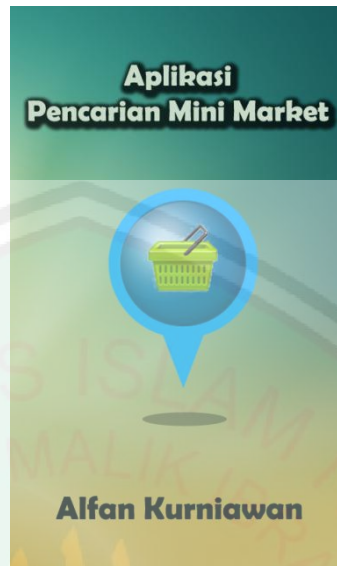
```
<?php
mysql_connect("localhost","alfankur_003","alfan003");
mysql_select_db("alfankur_summer2013");
$q = mysql_query("select * from lokasi");
while($e=mysql_fetch_assoc($q)){
    $output[]=$e;
}
echo "{\"data\":";
print(json_encode($output));
echo "}";
mysql_close();
?>
```

#### 4.4 Implementasi Antarmuka

Setelah semua komponen pendukung telah terinstal maka step selanjutnya yaitu pembuatan program atau aplikasi. Dalam penjelasan program dijelaskan fungsi – fungsi dan kegunaan program yang telah di buat beserta tampilan atau desainnya. Ada 5 menu dalam aplikasi ini antara lain Menu Lokasiku, Menu Tipe Peta, Menu Seting Radius, Menu Tentang, Menu Refresh Peta.

##### 4.4.1 *Splash Screen*

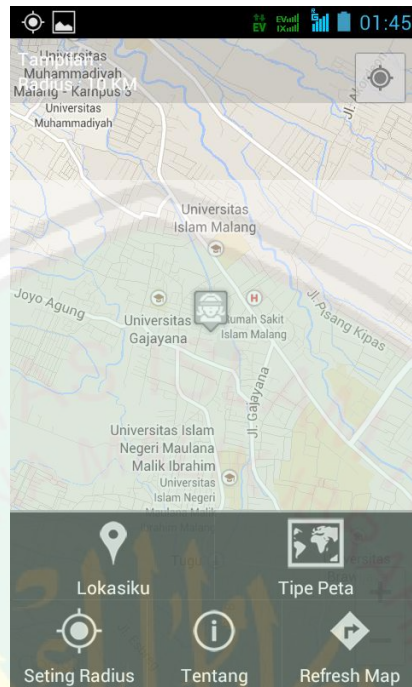
*Splash screen* adalah tampilan awal yang berisi gambar dan atau teks yang muncul ketika aplikasi pertama kali di buka atau dijalankan. Untuk menampilkan splash screen dapat di atur berapa lama waktu yang dibutuhkan untuk tampil sebelum masuk ke menu utama. Splash screen ini digunakan untuk memberikan tampilan awal yang berisi informasi seperti nama aplikasi, logo aplikasi, dan pembuat aplikasi. Pada proses ini, terdapat proses pengambilan data lokasi *user*, lokasi mini market dan menampilkan peta. Lihat gambar 4.2.



**Gambar 4.2** *Splash screen aplikasi*

#### **4.4.2 Form Tampilan Utama**

Pada form ini, berisi tampilan utama aplikasi setelah splash screen. Pada form tampilan utama ini menampilkan peta dan lokasi *user*. Selain menampilkan peta dan marker, di form ini juga menampilkan menu yang akan digunakan untuk mencari mini market di antara lain Menu Lokasiku, Menu Tipe Peta, Menu Seting Radius, Menu Tentang. Lihat gambar 4.3.



**Gambar 4.3** *Tampilan utama aplikasi*

#### 4.4.3 Form Tampilan Peta

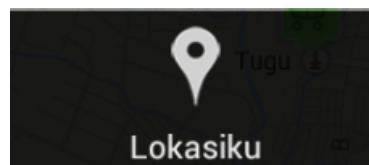
Pada form ini berguna untuk menampilkan peta serta marker dari lokasi *user* dan lokasi mini market. Jika lokasi *user* tidak di dapat maka akan merujuk ke lokasi *default* yang sudah di atur dalam program jika *user* lupa menyalakan GPS. Dalam hal ini ada proses koneksi dengan database secara online yaitu, untuk menampilkan marker lokasi mini market. Lihat gambar 4.4.



**Gambar 4.4** Form tampilan peta

#### 4.4.4 Menu Lokasiku

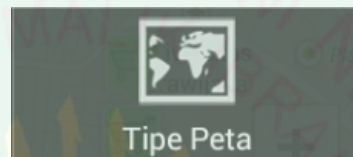
Menu ini berguna untuk menampilkan lokasi *user* berada secara realtime dengan bantuan GPS dan *internet* dan selanjutnya akan ditampilkan di peta. Untuk menggunakan menu ini *user* harus mengaktifkan gps pada *device* terlebih dahulu. Lihat gambar 4.5.



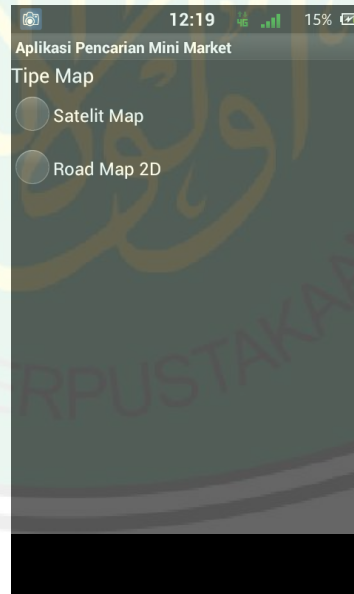
**Gambar 4.5** Menu lokasiku

#### 4.4.5 Menu Tipe Peta

Pada menu ini berguna untuk memilih tampilan dari peta. Yaitu ada 2 pilihan peta antara lain satellite map dan road map 2D. Satellite map akan menampilkan peta sesuai penampakan citra asli bumi. Road map 2D menampilkan peta secara 2D dan lebih ringan daripada satellite map. Lihat gambar 4.6 dan 4.7.



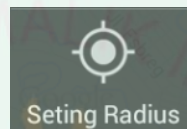
**Gambar 4.6** Menu Tipe peta



**Gambar 4.7** Menu pilihan tipe peta

#### 4.4.6 Menu Seting Radius

Menu ini digunakan untuk mengatur radius lokasi mini market yang ingin ditampilkan. Pada menu ini terdapat menu radius yang dapat di pilih dari 1 Km hingga 10 Km. Radius digunakan untuk membatasi ruang lingkup pencarian agar tidak terlalu lama mencari lokasi terdekat. Lihat gambar 4.8 dan 4.9.



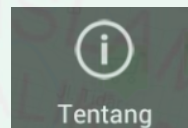
**Gambar 4.8** Menu seting radius



**Gambar 4.9** Menu pilihan seting radius

#### 4.4.7 Menu Tentang

Menu tentang berisi mengenai penjelasan tentang aplikasi yang di buat, nama pembuat dan alamat website / email pembuat dan info lainnya terkait dengan aplikasi dan pembuatnya. Lihat gambar 4.10 dan 4.11.



**Gambar 4.10** Menu tentang



Aplikasi pencarian Minimarket dibuat bertujuan untuk mempermudah pengguna mencari Minimarket terdekat dengan lokasi pengguna. Aplikasi ini memanfaatkan Haversine Formula dan teknologi Google Map untuk mendapatkan radius dan menentukan jarak terdekat yang optimal.

Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri  
Maulana Malik Ibrahim Malang  
2013

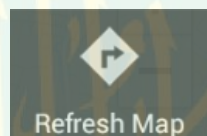
Alfan Kurniawan©2013

[www.alfankurniawan.com](http://www.alfankurniawan.com)

**Gambar 4.11** Form menu tentang

#### 4.4.8 Menu Refresh Map

Pada menu *refresh map* berguna untuk menampilkan *icon minimarket* dari *database* untuk ditampilkan di peta/*map*. Menu ini juga berguna me-*refresh marker/jalan* dengan jarak terdekat jika posisi *user* berpindah tempat dan melakukan pencarian *minimarket* lagi. Menu ini di buat untuk mengatasi duplikat *marker/jalan* dan *icon* posisi *user* yang sudah melakukan pencarian sebelumnya. Lihat gambar 4.12.



Gambar 4.12 Menu refresh map

#### 4.4.9 Tombol Zoom In dan Zoom Out

Tombol ini digunakan untuk memperbesar (*zoom in*) dan memperkecil (*zoom out*) tampilan citra pada peta. Tombol ini berguna untuk memperdetil jalur yang dilalui oleh *marker* jalan setelah di kalkulasi dan ditemukan jarak terdekat posisi *user* dan *minimarket*. Lihat gambar 4.13.



**Gambar 4.13** Tombol *zoom in* dan *zoom out*

#### 4.4 Implementasi Rumus *Haversine Formula*

Untuk membuktikan bahwa rumus haversine dapat menghitung jarak antara dua buah titik dipermukaan bumi maka perlu dilakukan uji coba berikut ini merupakan penjabaran rumus *Haversine Formula* dan penerapan rumus di ke dalam *source code* aplikasi. Berikut rumus *Haversine Formula* yang digunakan.

$$R = \text{Radian Bumi} = 6371 \text{ km.}$$

$$\Delta lat = lat2 - lat1$$

$$\Delta long = long2 - long1$$

$$a = \sin^2(\Delta lat/2) + \cos(lat1) * \cos(lat2) * \sin^2(\Delta long/2)$$

$$c = 2 * \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R * c$$

Dari penjabaran di atas maka didapatkan rumus dalam *query* aplikasi menggunakan *Haversine Formula* seperti di bawah ini.

```

public class Haversine {
    public static double HaversineFormula(double latPos, double lonPos,
        int index, DataLokasi dataLokasi) {
        final int R = 6371;
        double latFas = dataLokasi.getLokasi(index).getLatitude();
        double lonFas = dataLokasi.getLokasi(index).getLongitude();
        double latDistance = toRad(latFas-latPos);
        double lonDistance = toRad(lonFas-lonPos);
        double a = Math.sin(latDistance / 2) * Math.sin(latDistance / 2) +
            Math.cos(toRad(latPos)) * Math.cos(toRad(latFas)) *
            Math.sin(lonDistance / 2) * Math.sin(lonDistance / 2);
        double c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
        double distance = R * c;
        return distance;
    }
    private static Double toRad(Double value) {
        return value * Math.PI / 180;
    }
}

```

Contoh perhitungan rumus *Haversine Formula* :

Lokasi *Minimarket*

Latitude (lat1) : -7.96242 dijadikan radian = -0.138970445

Longitude (long1) : 112.61342 dijadikan radian = 1.965474961

Lokasi *User*

Latitude (lat2) : -7.94271 dijadikan radian = -0.138626441

Longitude (long2) : 112.60615 dijadikan radian = 1.965348075

$R = \text{Radian Bumi} = 6371 \text{ km.}$

$\Delta lat = lat2 - lat1 = -0.138626441 - (-0.138970445) = 0.000344004$

$\Delta long = long2 - long1 = 1.965348075 - 1.965474961 = -0.000126886$

$a = \sin^2 (\Delta lat / 2)$

$= 9.012010004001 * 10^{-12}$

$b = \cos (lat1) * \text{Cos}(lat2)$

$= 0.999994131545609661606336$

$$\begin{aligned}
 c &= \sin^2 (\Delta long / 2) \\
 &= 1.2261 * 10^{-12} \\
 d &= a + b * c \\
 &= 1.02381028086890720060955285696 * 10^{-11} \\
 e &= 2 * \text{atan2}(\sqrt{d}, \sqrt{(1 - d)}) \\
 &= 0.000366659018 \\
 f &= R * e \\
 &= 6371 * 0.000366659018 \\
 &= 2.335984603678 \\
 &= 2.34 \text{ km}
 \end{aligned}$$

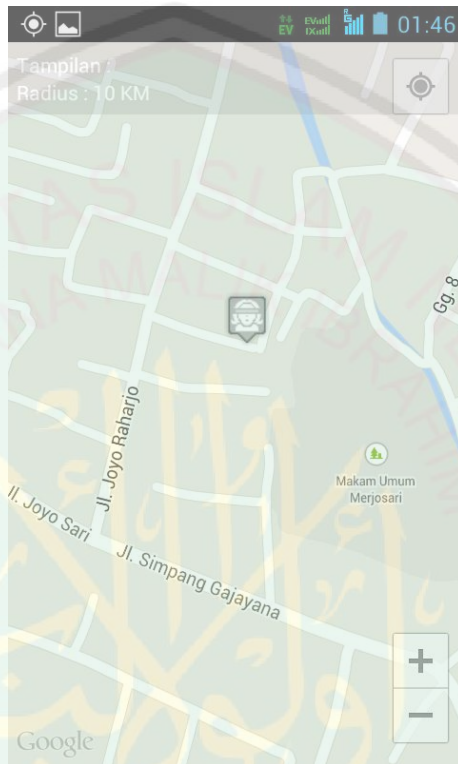
#### 4.5 Evaluasi dan Analisis Hasil Pengujian

Setelah aplikasi selesai di buat maka, aplikasi akan di test dengan beberapa pengujian. Diantaranya pengujian menampilkan lokasi *user*, menampilkan icon minimarket berdasarkan tipe peta yang berbeda, menampilkan icon minimarket dengan radius yang berbeda, menentukan rute terpendek serta pengujian perangkat lunak. Untuk tahap akhir mengevaluasi dan menganalisis hasil dari kuisisioner uji kelayakan produk.

##### 4.5.1 Pengujian Menemukan & Menampilkan Lokasi *User*

Pengujian ini dilakukan untuk mengetahui kemampuan aplikasi ini dalam mencari lokasi *user* dengan bantuan GPS dan internet apakah akurat atau tidak. Pada

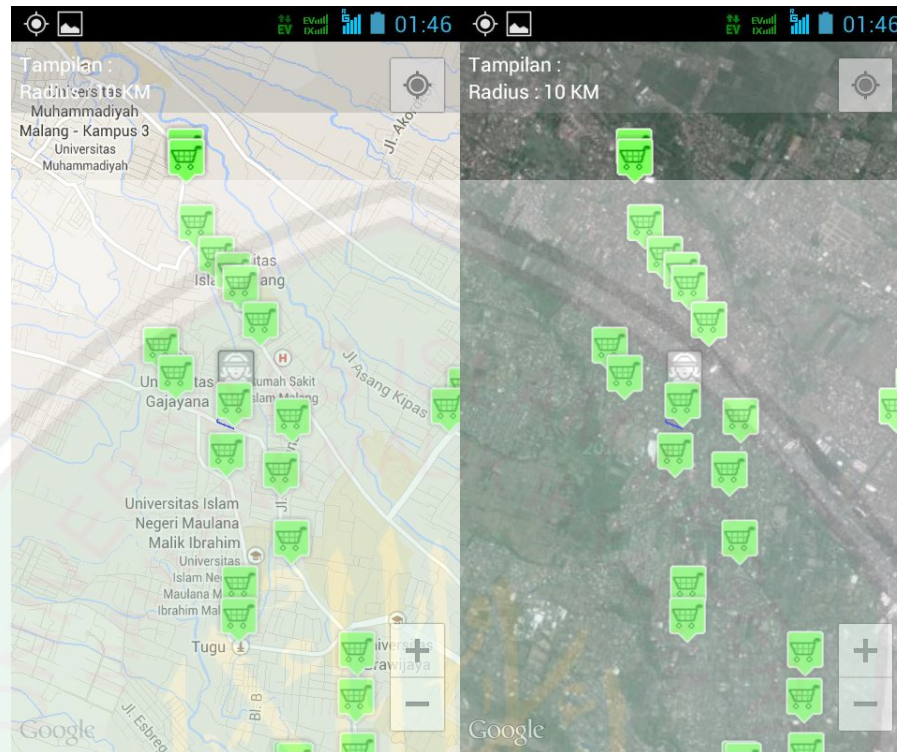
saat di lakukan uji coba, aplikasi dapat menampilkan posisi *user* secara akurat .Lihat gambar 4.14.



**Gambar 4.14** Menampilkan lokasi user

#### 4.5.2 Pengujian Menampilkan Icon Minimarket Berdasarkan Tipe Peta

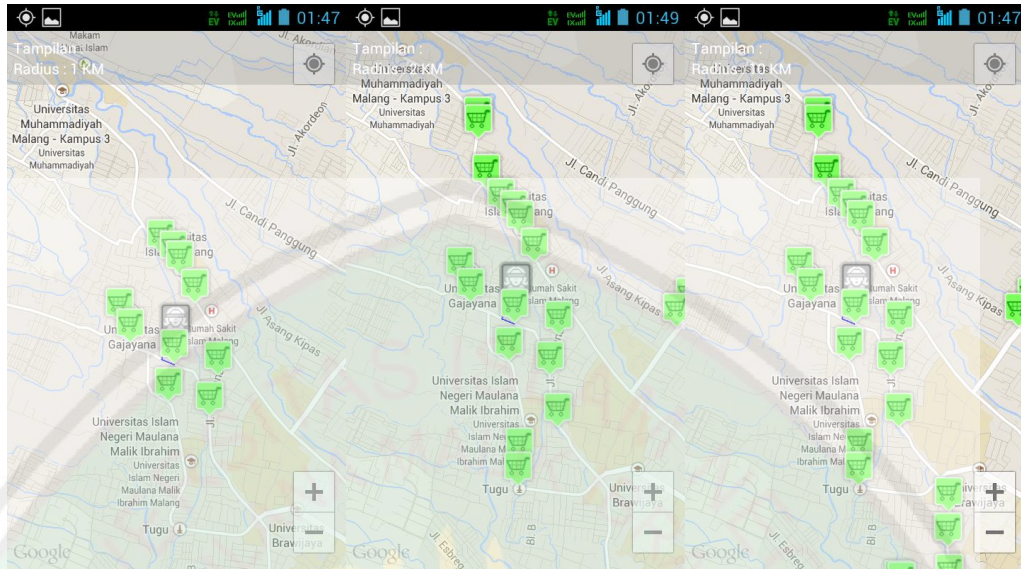
Pada pengujian ini dilakukan untuk mengetahui apakah program dapat menampilkan marker minimarket dengan tipe peta yang berbeda yaitu satellite map dan road map 2D. Pada pengujian ini aplikasi dapat menampilkan icon minimarket berdasarkan tipe peta dengan baik. Lihat gambar 4.15.



**Gambar 4.15** Menampilkan icon minimarket berdasarkan tipe peta

### 4.5.3 Pengujian Menampilkan Icon Minimarket Berdasarkan Radius

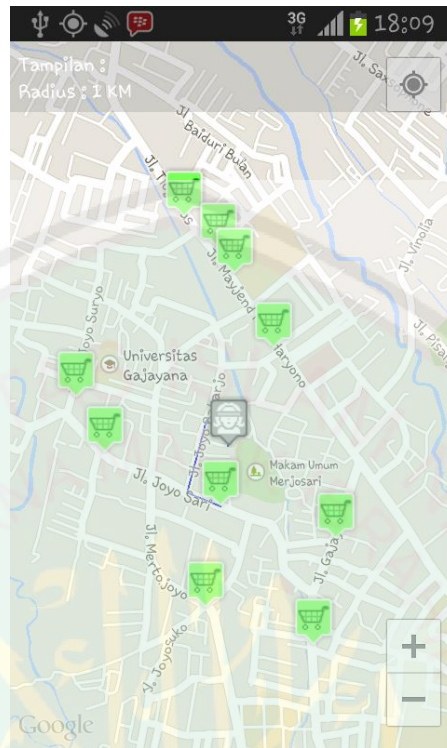
Pada tahap ini aplikasi di uji dengan mengatur radius yang berbeda antara 1 Km sampai dengan 10 Km apakah dapat menampilkan marker minimarket dengan baik. Pada pengujian ini aplikasi dapat menampilkan icon minimarket tanpa kendala walaupun di ubah radiusnya dari 1 Km, 2 Km hingga 10 Km. Lihat gambar 4.16.



**Gambar 4.16** Menampilkan icon minimarket berdasarkan radius yang berbeda

#### 4.5.4 Pengujian Menampilkan *Minimarket* Dengan Jarak Terdekat Berdasarkan Satu Lokasi *User*

Pada tahap ini pengujian yang dilakukan yaitu menguji aplikasi dengan satu lokasi *user*. Dalam pengujian ini aplikasi hanya pada satu lokasi dan di uji apakah lokasi *minimarket* terdekat yang ditemukan akurat atau tidak. Lihat gambar 4.17 berikut ini.



**Gambar 4.17** Menampilkan minimarket terdekat berdasarkan satu lokasi user

Tabel berikut akan menjelaskan lebih detail untuk hasil pencarian *minimarket* terdekat dengan satu lokasi *user*. Lihat tabel 4.1 berikut ini.

**Tabel 4.1** Tabel hasil pengujian aplikasi berdasarkan satu lokasi

No.	Lokasi User	Radius (Km)	Alamat Minimarket	Jarak (Km)
1	Jl. Joyo Raharjo	1	Jl. Joyosuk Timur	0.55
2			Jl. Gajayana	0.48
3			Jl. Gajayana	0.73
4			Jl. Joyo Tambaksari	0.2

5			Jl. Joyo Utomo	0.41
6			Jl. Joyo Suryo	0.54
7			Jl. Mayjen MT Haryono	0.36
8	Jl. Joyo Raharjo	1	Jl. Mayjen MT Haryono	0.57
9			Jl. Mayjen MT Haryono	0.66
10			Jl. Tlogomas	0.78

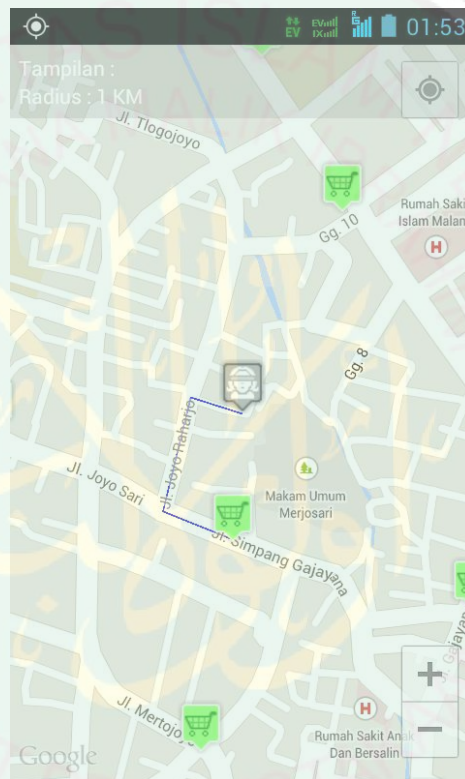
Melihat tabel 4.1 dengan membandingkan jarak setiap *minimarket* di ambil kesimpulan bahwa, aplikasi ini cukup akurat untuk menentukan lokasi *minimarket* terdekat. Dari pengujian dengan satu lokasi *user* didapatkan lokasi *minimarket* terdekat yaitu di Jl. Joyo Tambaksari dengan jarak 0.2 km.

#### 4.5.5 Pengujian Menampilkan *Minimarket* Terdekat Berdasarkan Lokasi *User* Yang Variatif Dan Berapa Waktu Yang Dibutuhkan

Pada tahap ini pengujian dilakukan dengan menguji aplikasi sepenuhnya yaitu menampilkan jarak *minimarket* terdekat dengan posisi *user* yang variatif atau berbeda-beda posisinya dan dapat menampilkan hasil akhir yaitu jarak terdekat serta waktu yang dibutuhkan untuk menemukan *minimarket* terdekat dari awal membuka aplikasi sampai menemukan lokasi dengan radius minimal. Pada tahap ini dilakukan 10 kali uji coba untuk mendapatkan hasil yang maksimal.

## 1. Pengujian 1

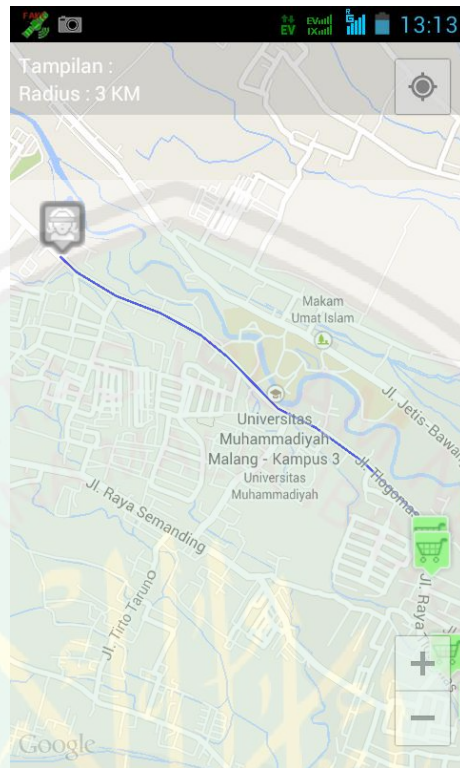
- Lokasi *user* : Jl. Joyo Raharjo
- Radius : 1 Km
- Jarak terdekat minimarket : 0.21 Km
- Lokasi *minimarket* : Jl. Simpang Gajayana



Gambar 4.18 Screenshot hasil pengujian 1

## 2. Pengujian 2

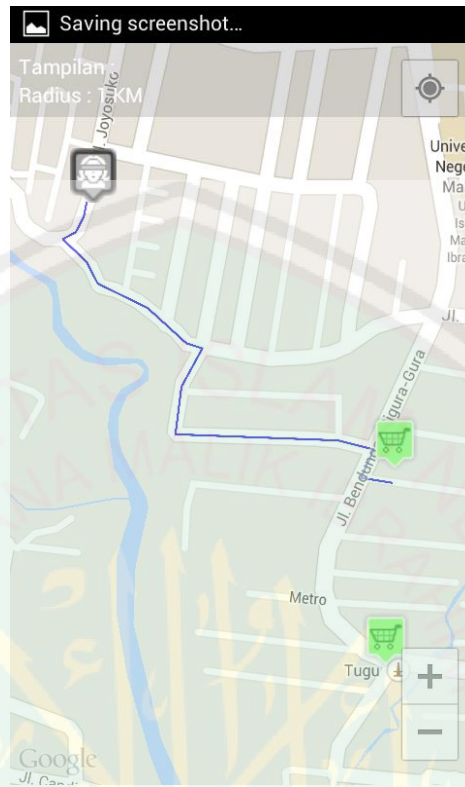
- Lokasi *user* : Jl. Raya Sengkaling
- Radius : 3 Km
- Jarak terdekat minimarket : 2.08 Km
- Lokasi *minimarket* : Jl. Raya Tlogomas



**Gambar 4.19** Screenshot hasil pengujian 2

### 3. Pengujian 3

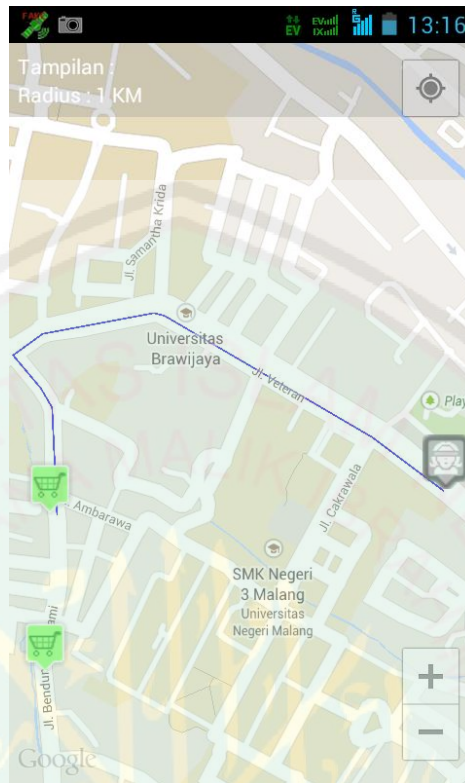
- Lokasi *user* : Jl. Joyosuko Timur
- Radius : 1 Km
- Jarak terdekat : 0.44 Km
- Lokasi *minimarket* : Jl. Bendungan Sigura-gura



**Gambar 4.20** Screenshot hasil pengujian 3

#### 4. Pengujian 4

- Lokasi *user* : Jl. Veteran
- Radius : 1 Km
- Jarak terdekat : 0.78 Km
- Lokasi *minimarket* : Jl. Bendungan Sutami



**Gambar 4.21** Screenshot hasil pengujian 4

## 5. Pengujian 5

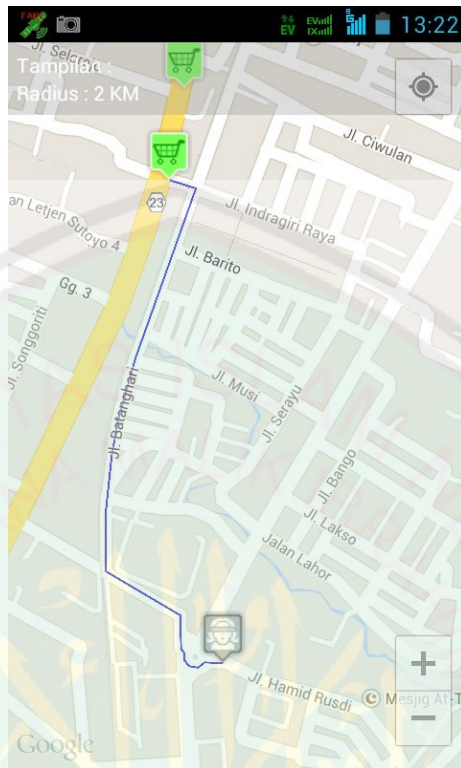
- Lokasi *user* : Jl. Jaksa Agung Suprpto
- Radius : 2 Km
- Jarak terdekat : 1.59 Km
- Lokasi *minimarket* : Jl. Brigjen Katamso



**Gambar 4.22** Screenshot hasil pengujian 5

#### 6. Pengujian 6

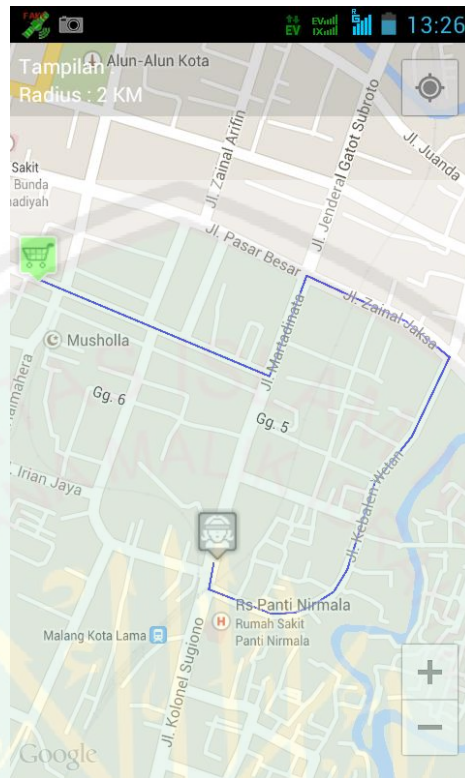
- Lokasi *user* : Jl Hamid Rusdi
- Radius : 2 Km
- Jarak terdekat : 1.03 Km
- Lokasi *minimarket* Jl. Indragiri Raya



**Gambar 4.23** Screenshoot hasil pengujian 6

#### 7. Pengujian 7

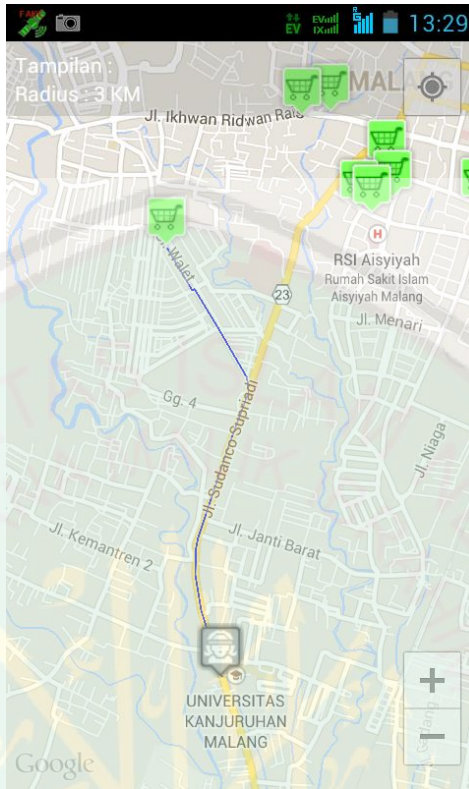
- Lokasi *user* : Jl. Sartono Sh
- Radius : 1 Km
- Jarak terdekat : 0.78 Km
- Lokasi *minimarket* : Jl. Kapten Pierre Tendean



**Gambar 4.24** Screenshoot hasil pengujian 7

#### 8. Pengujian 8

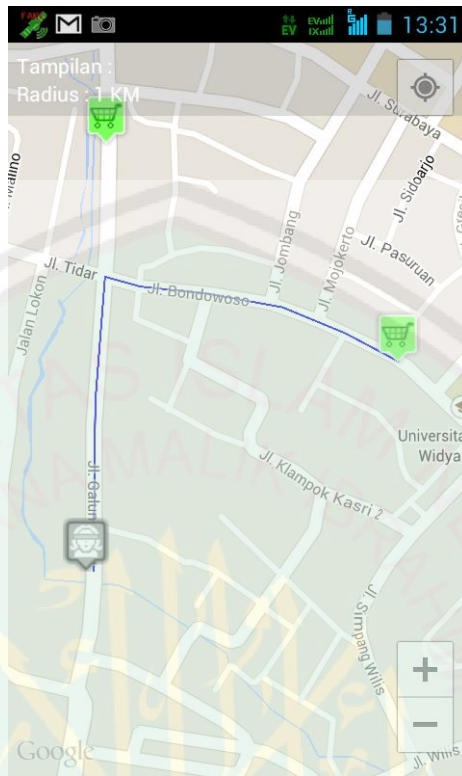
- Lokasi *user* : Jl. Sudanco Supriadi
- Radius : 3 Km
- Jarak terdekat : 2.02 Km
- Lokasi *minimarket* : Jl. Cendrawasih



**Gambar 4.25** Screenshot hasil pengujian 8

### 9. Pengujian 9

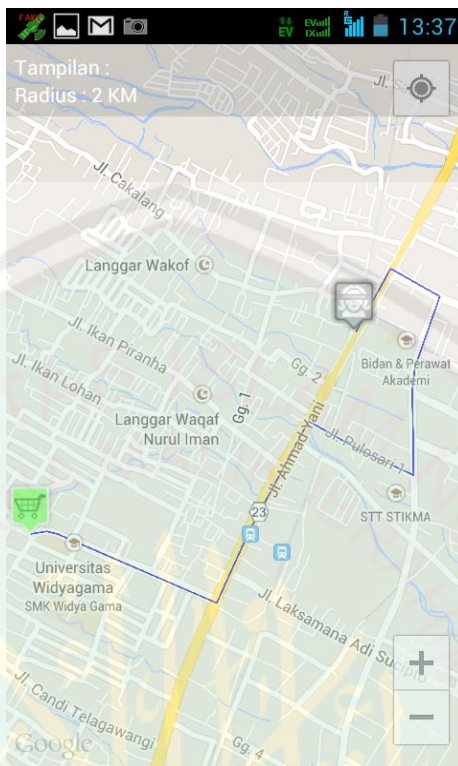
- Lokasi *user* : Jl. Galunggung
- Radius : 1 Km
- Jarak terdekat : 0.46 Km
- Lokasi *minimarket* : Jl. Bondowoso



**Gambar 4.26** Screenshoot hasil pengujian 9

#### 10. Pengujian 10

- Lokasi *user* : Jl. Bale Arjosari
- Radius : 2 Km
- Jarak terdekat : 1.61 Km
- Lokasi *minimarket* : Jl. Borobudur



**Gambar 4.27** Screenshot hasil pengujian 10

**Tabel 4.2** Hasil Pengujian Menampilkan Minimarket Terdekat Berdasarkan Lokasi User Yang Variatif Dan Berapa Waktu Yang Dibutuhkan

No.	Lokasi User	Radius	Lokasi Minimarket	Jarak	Waktu
1	Jl. Joyo Raharjo	1 Km	Jl. Joyo Tambaksari	0.21 Km	23 detik
2	Jl. Raya Sengkaling	3 Km	Jl. Raya Tlogomas	2.08 Km	25 detik
3	Jl. Joyosuko Timur	1 Km	Jl. Sigura-gura	0.44 Km	23 detik
4	Jl. Veteran	1 Km	Jl. Bend. Sutami	0.78 Km	22 detik
5	Jl. J.A. Suprpto	2 Km	Jl. Brigjen Katamso	1.59 Km	28 detik
6	Jl. Hamid Rusdi	2 Km	Jl. Indragiri Raya	1.03 Km	19 detik

7	Jl. Sartono Sh	1 Km	Jl. Kapten Pierre T.	0.78 Km	22 detik
8	Jl. Sudanco Supriadi	3 Km	Jl. Cendrawasih	2.02 Km	25 detik
9	J. Galunggung	1 Km	Jl. Bondowoso	0.46 Km	27 detik
10	Jl. Bale Arjosari	2 Km	Jl. Borobudur	1.61 Km	26 detik
Total					240 detik
Total Rata – Rata					24 detik

Dari 10 pengujian di atas dengan posisi *user* yang berbeda didapatkan hasilnya dapat dilihat pada tabel 4.2 bahwa aplikasi pencarian *minimarket* ini mempunyai akurasi yang cukup baik dalam menentukan jarak terdekat berdasarkan lokasi *user* yang berbeda. Dari 10 pengujian total rata-rata waktu yang dibutuhkan untuk menemukan *minimarket* terdekat cukup cepat yaitu 24 detik. Tidak ada hambatan berarti dalam penggunaan aplikasi ini selama digunakan di *outdoor*/di luar ruangan. Jika digunakan di dalam ruangan maka akurasinya akan menurun atau bahkan tidak akurat karena ada kemungkinan sinyal gps tertutup bangunan.

#### 4.5.6 Pengujian Perangkat Lunak

Setelah program di uji dan bekerja dengan baik pada *emulator* dan *device*, maka selanjutnya program di uji coba dengan *device* yang berbeda. Hal ini bertujuan untuk mengetahui apakah program juga dapat berjalan pada *smartphone*

lainnya yang memiliki resolusi layar dan operating sistem yang berbeda.. Untuk lebih jelasnya lihat table 4.3 berikut.

**Tabel 4.3** *Pengujian aplikasi pada berbagai device*

No.	Tipe	Resolusi Layar	Versi Android	Keterangan
1.	Smartfren Andromax C (Jelly Bean version)	WVGA (480x800 px)	4.1.2	Berjalan dengan baik
2.	LG L3 E400	QVGA (240x320 px)	2.3.6	Berjalan dengan baik
3.	Samsung Galaxy W	WVGA (480x800 px)	2.3.6	Berjalan dengan baik
4.	HTC Desire S	WVGA (480x800 px)	4.0.4	Berjalan dengan baik
5.	Sony Xperia M	WVGA (480x854 px)	4.1.1	Berjalan dengan baik
6.	Samsung Galaxy Ace 3	WVGA (480x800 px)	4.2.2	Berjalan dengan baik
7.	Samsung Galaxy S Advance	WVGA (480x800 px)	4.1.2	Berjalan dengan baik
8.	Oppo Find Muse	WVGA (480x800 px)	4.2.2	Berjalan dengan baik
9.	Samsung Galaxy SIII Mini	WVGA (480x800 px)	4.1.2	Berjalan dengan baik
10.	Sony Xperia E Dual	HVGA (320x480 px)	4.1.2	Berjalan dengan baik
11.	Sony Ericson Live With Walkman	HVGA (320x480 px)	4.0.4	Berjalan dengan baik

#### 4.5.7 Hasil Kuesioner Uji Kelayakan Aplikasi

Hasil perolehan perhitungan dari 50 responden, yang di ambil dari berbagai pihak diantaranya khalayak umum pengguna *device android* dan mahasiswa Jurusan Teknik Informatika dengan *device* dan *operating sistem* yang berbeda. Lihat tabel 4.4.

**Tabel 4.4** Rekapitulasi hasil kuisisioner

No	Pernyataan	Jumlah Penilaian Responden				
		SB	B	C	K	SK
1	Ketertarikan dengan Aplikasi	15	27	6	1	1
2	Tampilan Aplikasi	7	32	8	2	1
3	Kesesuaian desain warna	7	25	14	4	
4	Kelengkapan fitur	5	28	15	1	1
5	Kemudahan penggunaan fitur	10	29	11		
6	Penggunaan bahasa	10	32	8		
7	Manfaat Aplikasi	16	19	13	2	
8	Pengembangan Aplikasi	20	24	5	1	
9	Publikasi Aplikasi pada masyarakat	17	24	7	1	1
10	Tanggapan pengguna secara keseluruhan	10	31	8	1	

Berdasarkan hasil rekapitulasi kuesioner pada tabel 4.4, maka diperoleh hasil untuk masing-masing pertanyaan.

Data interval tersebut, di analisis dengan menghitung rata-rata jawaban berdasarkan skoring setiap jawaban dari responden dan menghitung prosentase total hasil jawaban dengan membagi total jawaban responden dengan total skor ideal (dengan nilai 250, jika semua jawaban mendapat nilai SB). Berdasarkan skor yang telah ditetapkan dapat di hitung, sebagai berikut:



1. Ketertarikan dengan Aplikasi

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
15	5	75	27	4	108	6	3	18	1	2	2	1	1	1

2. Tampilan Aplikasi

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
7	5	35	32	4	128	8	3	24	2	2	4	1	1	1

3. Kesesuaian desain warna

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
7	5	35	25	4	100	14	3	42	4	2	8	0	1	0

4. Kelengkapan fitur

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
5	5	<b>25</b>	28	4	<b>112</b>	15	3	<b>45</b>	1	2	<b>2</b>	1	1	<b>1</b>

5. Kemudahan penggunaan fitur

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
10	5	<b>50</b>	29	4	<b>116</b>	11	3	<b>33</b>	0	2	<b>0</b>	0	1	<b>0</b>

6. Penggunaan bahasa

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
10	5	<b>50</b>	32	4	<b>128</b>	8	3	<b>24</b>	0	2	<b>0</b>	0	1	<b>0</b>

7. Manfaat Aplikasi

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
16	5	<b>80</b>	19	4	<b>76</b>	13	3	<b>39</b>	2	2	<b>4</b>	0	1	<b>0</b>

8. Pengembangan Aplikasi

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
20	5	<b>100</b>	24	4	<b>96</b>	5	3	<b>15</b>	1	2	<b>2</b>	0	1	<b>0</b>

9. Publikasi Aplikasi pada masyarakat

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
17	5	<b>85</b>	24	4	<b>96</b>	7	3	<b>21</b>	1	2	<b>2</b>	1	1	<b>1</b>

10. Tanggapan pengguna secara keseluruhan

SB			B			C			K			SK		
$\sum$ SB (t1)	Skor (t2)	Total (t1*t2)	$\sum$ B (t1)	Skor (t2)	Total (t1*t2)	$\sum$ c (t1)	Skor (t2)	Total (t1*t2)	$\sum$ K (t1)	Skor (t2)	Total (t1*t2)	$\sum$ SK (t1)	Skor (t2)	Total (t1*t2)
10	5	<b>50</b>	31	4	<b>124</b>	8	3	<b>24</b>	1	2	<b>2</b>	0	1	<b>0</b>

Data tersebut merupakan hasil dari perhitungan kuesioner dengan menggunakan skala *Likert*. Pada tabel 4.5 dibawah merupakan hasil perhitungan dengan skala *Likert*.



**Tabel 4.5** Hasil perhitungan kuesioner dengan skala Likert

No	Pernyataan	Hasil Perhitungan					Prosentase
		SB	B	C	K	SK	
1	Ketertarikan dengan Aplikasi	75	108	18	2	1	81.6%
2	Tampilan Aplikasi	35	128	24	4	1	76.8%
3	Kesesuaian desain warna	35	100	42	8	-	74%
4	Kelengkapan fitur	25	112	45	2	1	74%
5	Kemudahan penggunaan fitur	50	116	33	-	-	79.6%
6	Penggunaan bahasa	50	128	24	-	-	80.8%
7	Manfaat Aplikasi	80	76	39	4	-	79.6%
8	Pengembangan Aplikasi	100	96	15	2	-	85.2%
9	Publikasi Aplikasi pada masyarakat	85	96	21	2	1	82%
10	Tanggapan pengguna secara keseluruhan	50	124	24	2	-	80%
<b>Total hasil</b>							<b>793.6</b>
<b>Rata-rata</b>							<b>79.36%</b>

Berdasarkan tabel 4.5, hasil dari analisis kuesioner uji kelayakan aplikasi dari jawaban 50 responden, 79.36% menyatakan bahwa aplikasi pencarian *Minimarket* ini layak berdasarkan tingkatan uji kelayakan, sehingga dapat digunakan

dan dipublikasikan untuk masyarakat umum. Diharapkan dapat membantu dan memberikan solusi dalam pencarian *Minimarket* berdasarkan jarak terdekat sehingga, dapat menghemat waktu, biaya dan tenaga untuk menemukan *Minimarket* dengan menggunakan *mobile devices* berbasis Android.

#### 4.6 Hubungan Aplikasi Pencarian *Minimarket* dengan Perspektif Islam

Dari kuisisioner dan perhitungan kuisisioner menggunakan skala likert bahwa hasil rata-rata perhitungan tersebut menunjukkan nilai 79.36% aplikasi pencarian *Minimarket* ini layak berdasarkan tingkat uji kelayakan. Sedangkan dari 10 pengujian aplikasi diatas bahwa aplikasi ini mempunyai akurasi yang baik dalam menentukan jarak *Minimarket* terdekat. Jadi aplikasi ini memang dapat digunakan dan dimanfaatkan untuk masyarakat luas untuk membantu menemukan *Minimarket* terdekat dengan posisi pengguna. Jadi disini hubungan aplikasi pencarian *Minimarket* dengan perspektif islam yaitu mengamalkan ilmu yang bermanfaat untuk membantu masyarakat luas.

Dijelaskan dalam Al-Qur'an Surat Al-Kahfi ayat 107 sebagai berikut :

﴿ ۱۷ ﴾ إِنَّ الَّذِينَ ءَامَنُوا وَعَمِلُوا الصَّالِحَاتِ كَانَتْ لَهُمْ جَنَّاتُ الْفِرْدَوْسِ نُزُلًا

“Sesungguhnya orang-orang yang beriman dan beramal saleh, bagi mereka adalah surga Firdaus menjadi tempat tinggal” (QS. Al-Kahfi : 107).

Dijelaskan juga dalam Al-Qur'an Surat al-Baqarah ayat 82 sebagai berikut :

وَالَّذِينَ ءَامَنُوا وَعَمِلُوا الصَّالِحَاتِ أُولَٰئِكَ أَصْحَابُ الْجَنَّةِ ۖ هُمْ فِيهَا خَالِدُونَ ﴿٨٢﴾

“Dan orang-orang yang beriman serta beramal saleh, mereka itu penghuni surga; mereka kekal didalamnya” (QS. Al-Baqarah : 82).

Dari 2 ayat Al-Qur'an di atas dijelaskan bahwa Allah memberi manusia karunia akal dan ilmu yang bermanfaat untuk diamankan. Beramal dalam arti tidak harus harta benda atau tenaga, ilmu pun juga harus diamankan untuk kemaslahatan masyarakat luas. Dalam hal ini tujuan pembuatan Aplikasi Pencarian *Minimarket* Berdasarkan Jarak Terdekat Menggunakan Haversine Formula bukan lain untuk beramal saleh. Dengan beramal kita akan dilapangkan segala urusan di dunia maupun di akhirat dan Allah memberi balasan yang besar selain pahala yaitu surga firdaus.

## BAB V

### PENUTUP

#### 5.1. Kesimpulan

Dari penelitian masalah serta analisis dan pembuatan aplikasi ini, dapat di ambil kesimpulan bahwa:

- Aplikasi pencarian Minimarket berdasarkan jarak tempuh terdekat menggunakan metode *Haversine Formula* ini diimplementasikan pada *smartphone* berbasis Android.
- Aplikasi pencarian Minimarket diimplementasikan untuk membantu user dalam mencari minimarket dengan jarak tempuh terdekat guna menghemat waktu, tenaga dan biaya.
- Berdasarkan uji coba kecepatan dalam menentukan lokasi berdasarkan jarak tempuh terdekat, daerah (lingkungan) dan cuaca dapat mempengaruhi kecepatan dalam menentukan lokasi user berdasarkan jarak terdekat.
- Berdasarkan uji coba aplikasi pada beberapa *smartphone/mobile device*, aplikasi dapat berjalan dengan baik dengan resolusi layar dan *operating system* yang berbeda.

- Berdasarkan hasil dari analisis kuesioner uji kelayakan aplikasi, dari 50 responden, 79.36% menyatakan bahwa aplikasi pencarian Minimarket ini layak digunakan dan dipublikasikan untuk masyarakat umum. Meskipun demikian ada beberapa kendala dalam pengembangan aplikasi pencarian Minimarket diantaranya adalah keterbatasan sumberdaya dan keragaman jenis *smartphone* yang digunakan.

## 5.2. Saran

Saran yang disampaikan untuk pengembangan aplikasi ini guna penelitian selanjutnya yaitu dapat di tambah dengan beberapa fitur lainnya, seperti pencarian objek lain, menambah foto untuk mempermudah detil lokasi, menambah suara pada saat melakukan navigasi menggunakan mobil tanpa harus melihat *device mobile*, memberi tanda suara jika lokasi yang di maksud sudah dekat atau sudah sampai, memberikan *history* hasil pencarian agar mempercepat pencarian tanpa harus melakukan pendeteksian ulang lokasi user, memberikan kategori untuk minimarket yang ingin ditampilkan, dan lain-lain. Penggunaan metode lain yang lebih efektif juga bisa di teliti untuk mendapatkan hasil yang lebih baik.

## DAFTAR PUSTAKA

- Boemi Islam.2009. *Tafsir Surat Alam Nasyrah*. [online]. [Accessed 13 September 2013]. Available from World Wide Web: <<http://boemi-islam.net/Ilmu/Aqidah/tafsir-surat-alam-nasyrah/>>
- Bhatia, Shaveta, Saba Hilal.2013.*Mathematical Analysis of Mobile Latitude Software on Location Tracking*.India:International Journal of Computer Applications. ].[Accessed 2 Maret 2014].
- Catur.2010. *Google API* .[online]. [Accessed 9 Mei 2013]. Available from World Wide Web: <<http://catur.dosen.akprind.ac.id/2010/04/22/seri-3-konsep-dasar-uml/>>
- Cyberkomputer.2013.*Pengertian Android, Definisi dan Arti Android serta Versi Android Terbaru*. [online]. [Accessed 21 Januari 2013]. Available from World Wide Web: <<http://cyberkomputer.com/gadget/android/pengertian-android-definisi-dan-arti-android-seerta-versi-android-terbaru/>>
- Douglas Crockford.2008. *JavaScript: The Good Parts*.Sebastopol:O'Reilly Media, Inc.
- G-excess.2009.*Informas: Pengertian GPS* . [online]. [Accessed 21 Januari 2013]. Available from World Wide Web: <<http://www.g-excess.com/343/informasi-pengertian-gps/>>
- Hasanuddin Z, Abidin. 2000.*Penentuan Posisi dengan GPS dan Aplikasinya*. Jakarta :Pradnya Paramita.
- Huda, Nurul, Umi Laili Yuhana, Bilqis Amaliah S.2011.*Rancang Bangun Aplikasi Penugasan pada Blackberry Mobile Device dengan Haversine Formula*.Surabaya:ITS Library. [Journal].[Accessed 8 Maret 2012]. Available from World Wide Web: <<http://digilib.its.ac.id/public/ITS-Undergraduate-17169-paper.pdf>>
- Ibutina.2010.*Tafsir – Surah Al Muzammil*. [online]. [Accessed 16 September 2013]. Available from World Wide Web:< <http://ibutina.com/islamia/alquran/tafsir-surah-al-muzammil/>>
- Ihsan,Muhammad.*Tafsir Surat : Al Muzammil*. [online]. [ Accessed 16 September 2013]. Available from World Wide Web:< [http://users6.nofeehost.com/alquranonline/Alquran\\_Tafsir.asp?SuratKe=73](http://users6.nofeehost.com/alquranonline/Alquran_Tafsir.asp?SuratKe=73)>
- Katsir,Ibnu.2007.*Tafsir Juz Amma, Penerjemah Farizal Tirmizi*.Jakarta:Pustaka Azzam.
- Komputer, Wahana.2013.*Android Programming with Eclipse*.Yogyakarta:Andi.
- Komputer, Wahana.2013.*Step by Step Menjadi Programmer Android*.Yogyakarta:Andi.
- Longitudestore.2008.*Haversine Formula*. [online]. [Accessed 3 September 2013]. Available from World Wide Web: <<http://www.longitudestore.com/haversine-formula.html>>
- Ma'ruf, Hendri.2005.*Pemasaran Ritel*.Jakarta:Gramedia Pustaka Utama.

- Yayasan Indonesia Membaca. 2009. *Al-Quran online Indonesia : Tafsir Depag RI*. [online]. [Accessed 16 September 2013]. Available from World Wide Web: <<http://quran.bacalah.net/content/surat/index.php>>
- Ratsameethammawong, Porpen, M.L. Kulthon Kasemsan.2010.*Mobile Phone Location Tracking by the Combination of GPS, Wi-Fi and Cell Location Technology*.Thailand:IBIMA Publishing.[Journal].[Accessed 4 Maret 2014]. Available from World Wide Web: <<http://www.ibimapublishing.com/journals/CIBIMA/2010/566928/566928.pdf>>
- Riyanto.2010.*Membuat Sendiri Aplikasi Mobile GIS Platform Java ME, Blackberry & Android*. Yogyakarta:Andi.
- Safaat, Nazaruddin.2012.*Pemrograman Aplikasi Mobile Smartphone dan tablet PC berbasis Android*.Bandung:Informatika.
- Quraish Shihab, M..2000.*Tafsir Al Mishbah Volume I*.Jakarta:Lentera Hati.
- Quthb,Sayyid.2008.*Tafsir Fi Zhilalil Qur'an Jilid 1*.Jakarta:Gema Insani
- Solichin, Ahmad.2010. *MySQL 5 : Dari Pemula Hingga Mahir*. [E-book]. [Accessed 4 September 2013]. Available from World Wide Web:<<http://achmatim.net/2010/01/30/buku-gratis-mysql-5-dari-pemula-hingga-mahir/>>
- Sohail, Adnan.2009. *JavaScript Object Notation (JSON)*. [E-Book]. [Accessed 4 September 2013]. Available from World Wide Web: <<http://www.slideshare.net/adnansohailrsm/javascript-object-notation-json-1219489>>
- Steiniger ,Stefan., Neun, Moritz. and, Edwardes, Alistair.2006.*Foundations of Location Based Services.Translation Journal*.Volume 1.0,No.1.[Journal].[Accessed 18 September 2013]. Available from World Wide Web: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.94.1844>>
- Zaki, Ali.2010.*Keliling Dunia Dengan Google Earth dan Google Maps*. Yogyakarta:Andi.

## Lampiran

### Kuesioner Uji Kelayakan Produk

Bapak/Ibu/Saudara yang terhormat,

Demi kepentingan peningkatan dan uji kelayakan aplikasi enkripsi sms, kami mohon bantuan untuk memberikan informasi. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Atas bantuan Bapak/Ibu/Saudara peneliti mengucapkan terima kasih.

Keterangan:

SB = Sangat baik

K = Buruk/Kurang

B = Baik

SK = Sangat buruk/kurang

C = Cukup/Biasa

Nama : .....

No	Pernyataan	Jawaban				
		SB	S	C	K	SK
1	Ketertarikan dengan Aplikasi					
2	Tampilan Aplikasi					
3	Kesesuaian desain warna					
4	Kelengkapan fitur					
5	Kemudahan penggunaan fitur					
6	Penggunaan bahasa					
7	Manfaat Aplikasi					
8	Pengembangan Aplikasi					
9	Publikasi Aplikasi pada masyarakat					
10	Tanggapan pengguna secara keseluruhan					

### Hasil Kuesioner Uji Kelayakan Produk

Keterangan:

SB = Sangat Baik

SK = Sangat buruk/kurang

B = Baik

1-10 = Item pernyataan dalam kuesioner

C = Biasa/Cukup

K = Buruk/Kurang

No	Responden	Jawaban Kuesioner									
		1	2	3	4	5	6	7	8	9	10
1	Angga Nugraha	SB	SB	SB	SB	B	SB	B	SB	SB	SB
2	Bayu Budiarto	B	SB	B	B	C	B	B	SB	B	SB
3	Aluna Kartika Sari	C	B	SB	K	B	SB	B	SB	SB	B
4	Syaiful Arif	C	K	C	C	B	SB	B	C	K	C
5	Muhammad Bustomi	B	C	B	B	B	B	B	B	B	SB
6	Didik Tri Susanto	B	B	C	C	C	C	B	B	B	B
7	Ricky Ardian Rukmana	SB	SB	B	B	B	SB	SB	SB	SB	SB
8	Andri Krismanto	B	B	C	C	B	C	C	B	B	B
9	Amalina Nisail Fikrinah	C	K	K	C	C	C	K	SB	C	K
10	Fauzi	C	B	B	B	B	C	C	B	C	C
11	Yudha	K	C	C	C	C	C	C	B	SB	C
12	Ulwan Irfandi	SB	B	B	B	B	C	C	C	C	B
13	Riesca Maranata	C	SB	C	B	B	B	C	C	C	C
14	Danang Wahyudi	B	B	SB	B	SB	B	C	SB	SB	SB
15	Wildan Gunardi	SK	B	B	C	C	B	C	B	C	B
16	Cahyo Nugroho	SB	C	C	SB	SB	B	SB	SB	B	B
17	Muh Ali Iksanudin	B	B	B	C	B	B	B	B	B	B
18	Ay	B	C	C	C	B	C	B	B	B	C
19	Nanda Reza Andhika	B	B	B	B	B	B	B	B	B	B
20	Nurisul Ubbat	B	B	B	B	B	B	B	B	B	SB
21	Pogal Indra Mussuga	B	SB	B	SB	SB	SB	SB	SB	SB	SB
22	Luqman Hakim	SB	B	B	SB	SB	B	SB	SB	SB	SB
23	M. Fajar Zulfikar	C	SK	B	SK	B	B	SB	K	SK	SB
24	Gilang Kurniawan	B	B	B	B	B	B	B	B	B	B
25	Tri Wahyudianto	SB	B	B	B	SB	B	SB	B	SB	B
26	Ahmad Hariadi	B	B	B	C	C	B	C	B	B	B
27	M. Haris Syafaat	B	SB	C	B	B	B	B	SB	B	B
28	S.S. Mayang Sari	SB	B	B	B	B	B	B	SB	SB	B

29	Anonim 1	B	B	K	C	C	B	C	B	C	B
30	Muvid	B	B	C	B	SB	SB	SB	SB	SB	B
31	Wahyudi	B	C	C	C	B	C	B	SB	B	C
32	Anonim 2	SB	B	B	B	B	B	SB	SB	B	B
33	Habib Abdullah	SB	B	B	B	SB	B	SB	B	SB	B
34	Lailys Suraidah	B	B	C	C	C	B	C	C	C	B
35	Fauzan	B	C	K	B	B	B	C	B	B	C
36	Alfian Pramuditya	B	B	C	B	C	B	B	B	B	C
37	M. Nuril Efendi	SB	B	B	B	B	B	B	SB	B	B
38	Bismo W.P.	B	C	C	C	B	B	SB	SB	B	B
39	Dhany	B	B	B	SB	B	SB	SB	B	SB	B
40	H. Priagung B.	B	B	SB	B	B	SB	C	C	SB	B
41	Andy Ryan P.	B	B	B	B	B	B	B	B	B	B
42	Agus Cahyono	SB	B	B	B	B	SB	B	B	B	B
43	H. Abdul Aziz	SB	B	B	C	C	B	SB	B	B	B
44	Zulfa Ulinuha	B	B	B	B	B	B	SB	B	SB	B
45	Edi	SB	SB	SB	B	B	B	SB	SB	SB	B
46	Sofyan Setiawan	B	C	C	B	B	B	SB	SB	SB	B
47	Ardi Yulianto	SB	B	B	B	SB	B	SB	SB	B	B
48	Imron Arif S.	SB	B	SB	B	SB	B	SB	B	SB	SB
49	Elga Fredianto	B	B	K	C	C	B	C	SB	B	B
50	Bekti Cahyo	B	B	B	B	B	B	B	B	B	B
<b>Total</b>											
	<b>SB</b>	15	7	6	5	9	9	17	20	17	10
	<b>B</b>	27	32	26	28	30	33	19	24	24	31
	<b>C</b>	6	8	14	15	11	8	13	5	7	8
	<b>K</b>	1	2	4	1	0	0	1	1	1	1
	<b>SK</b>	1	1	0	1	0	0	0	0	1	0