

PENERAPAN PERSAMAAN *Haversine* BERDASARKAN *Location Based Service (LBS)* PADA APLIKASI ANDROID UNTUK PENCARIAN RUMAH SAKIT DI WILAYAH BANDAR LAMPUNG

(Skripsi)

Oleh
RATIH NINGTIYAS



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2020**

ABSTRAK

PENERAPAN PERSAMAAN *Haversine* BERDASARKAN *Location Based Service (LBS)* PADA APLIKASI ANDROID UNTUK PENCARIAN RUMAH SAKIT DI WILAYAH BANDAR LAMPUNG

Oleh

RATIH NINGTIYAS

Konsep dasar *Location Based Service* adalah menghasilkan sebuah layanan yang dapat memberikan informasi lokasi keberadaan suatu objek yang ingin diketahui oleh *user* pengguna. Konsep tersebut bertujuan untuk membuat aplikasi *Location Based* tentang keberadaan rumah sakit beserta jam praktik dokter. Aplikasi yang dibuat menggunakan program MIT *App Inventor* yang ada pada Android, karena aplikasi yang dibuat belum ada di *Playstore*. Metode yang digunakan dalam perhitungan jarak terdekat ini merupakan penjabaran dari rumus *Haversine* di dalam *query My SQL*. Sedangkan, aplikasi dikembangkan dengan menggunakan program secara *visual block*, dengan cara beberapa *block* disesuaikan dengan kebutuhan pengguna. Data aplikasi ke sistem *user* merupakan informasi yang diinginkan tanpa menggunakan jaringan internet. Pengkodean aplikasi digunakan untuk menghitung jarak dari koordinat posisi *user* ke semua koordinat lokasi rumah sakit pilihan secara satu persatu, dan menyimpannya secara temporal dengan sistem *block Haversine*. Hasil pengujian di aplikasi dengan posisi user di titik latitude (-5.384522) dan longitude (105.267349) menghasilkan jarak tempuh pada masing-masing rumah sakit yaitu (2.2 km, 3.8 km, 6.9 km, 4.1 km, 1.1 km, 9.5 km, 8.2 km, 1.4km), sedangkan pada rumus *Haversine* di dalam *query My SQL* yaitu (3.5 km, 2.3 km, 8.8 km, 1.7 km, 1.8 km, 9.2 km, 7.2 km, 1.4 km), maka didapatkan jarak antara perhitungan dan aplikasi (1.3 km, 1.5 km, 1.9 km, 2.4 km, 0.7 km, 0.3 km, 0.4 km, 0). Perbandingan dapat terlihat bahwa dari hasil keduanya memiliki akurasi yang cukup baik, hampir mendekati sama, tetapi nilai selisih jarak pada aplikasi diperoleh 2.4 km lebih jauh, akurasi jarak terdekat diperoleh dari rumus *Haversine*.

Kata kunci: Aplikasi, LBS, *Haversine*, *App inventor*, *Google Map's*, RS Bandar Lampung

ABSTRACT

THE IMPLEMENTATION OF HAVERSINE EQUATION BASED ON LOCATION-BASED SERVICE (LBS) FOR ANDROID APPLICATION HOSPITALS SEARCH IN BANDAR LAMPUNG

By

RATIH NINGTIYAS

A basic concept of location-based services is to provide a service that can give information about the location of an object that user want to know. By using the concept, one can develop a location-based application to search the hospital location and doctors working hours. The application was developed by MIT App inventor in Android, due to the unavailability of the application in Playstore. The method used in the calculation of the closest distance is using the explanation of the Haversine equation in query MY SQL. While, the programming application was created by the visual block, by which the blocked are adjusted by user requirement. The Application data to the user system will works offline. The application coding is used to calculate the distance of user position to the chosen hospitals location it is conducted one by one, which is kept it temporary with Haversine blocks system. The result for user in the position of (-5.384522) latitude, and (105.267349) longitude are (2.2 km, 3.8 km, 6.9 km, 4.1 km, 1.1 km, 9.5 km, 8.2 km, 1.4 km), meanwhile to the Haversine equation in query my SQL are (3.5 km, 2.3 km, 8.8 km, 1.7 km, 1.8 km, 9.2 km, 7.2 km, 1.4 km), therefore the result comparison are (1.3 km, 1.5 km, 1.9 km, 2.4 km, 0.7 km, 0.3 km, 0.4 km, 0). From the comparison, we can see that both have good accuracy that close to the same value but has a difference in the application by 2,4 km further. So the closest accurate distance is the haversine equation.

Keywords: Application, LBS, Haversine, APP Inventor, google maps, RS Bandar Lampung

PENERAPAN PERSAMAAN *HAVERSINE* BERDASARKAN *LOCATION BASED SERVICE (LBS)* PADA APLIKASI ANDROID UNTUK PENCARIAN RUMAH SAKIT DI WILAYAH BANDAR LAMPUNG

Oleh
RATIH NINGTIYAS

(Skripsi)

Sebagai Salah Satu Syarat Untuk Mencapai Gelar
SARJANA TEKNIK

Pada

Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2020

Judul Skripsi : **PENERAPAN PERSAMAAN *HAVERSINE*
BERDASARKAN *LOCATION BASED*
SERVICE (LBS) PADA APLIKASI
ANDROID UNTUK PENCARIAN RUMAH
SAKIT DI WILAYAH BANDAR LAMPUNG**

Nama Mahasiswa : **Ratih Ningtias**

Nomor Pokok Mahasiswa : 1215031061

Program Studi : Teknik Elektro

Fakultas : Teknik



1. Komisi Pembimbing


Yetti Yuniati, S.T., M.T.
NIP 19800113 200912 2 002


Dr. Ing. Ardian Ulvan, S.T., M.Sc.
NIP 19731128 199903 1 005

2. Ketua Jurusan Teknik Elektro


Dr. Eng. Khairudin, S.T., M.Sc.
NIP 19700719 200012 1 001

MENGESAHKAN

1. Tim Penguji

Ketua : **Yetti Yuniati, S.T., M.T.**



Sekretaris : **Dr. Ing. Ardian Ulvan, S.T., M.Sc.**



Penguji
Bukan Pembimbing : **Misfa Susanto, S.T., M.Sc., Ph.D.**



2. Dekan Fakultas Teknik



 **Prof. Drs. Ir. SUHARNO, Ph.D., IPU., ASEAN. Eng.**
NIP 19620717 198703 1 002

Tanggal Lulus Ujian Skripsi : **10 Desember 2019**

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang pernah dilakukan orang lain dan sepanjang sepengetahuan saya tidak terdapat atau diterbitkan oleh orang lain, kecuali secara tertulis diacu dalam naskah ini sebagaimana yang disebutkan dalam daftar pustaka. Selain itu saya menyatakan pula bahwa skripsi ini dibuat oleh saya sendiri.

Apabila pernyataan saya tidak benar, maka saya bersedia dikenai sanksi sesuai dengan hukum yang berlaku.

Bandar Lampung, 18 Desember 2019



Ratih Ningtyas
NPM 1215031061



RIWAYAT HIDUP



Penulis dilahirkan di Bandar Lampung pada tanggal 24 Februari 1994, sebagai anak pertama dari dua bersaudara, dari bapak Sulaiman dan ibu Eka Susilawati. Pendidikan formal penulis dimulai dari Taman kanak - kanak Taruna Jaya pada tahun 1999 - 2000, kemudian lanjut pada Sekolah Dasar (SD) Al-Azhar 1 Bandar Lampung pada tahun 2000 - 2005. Kemudian penulis melanjutkan sekolah dasar di Sekolah Dasar Negeri (SDN) 1 Way Halim Permai pada tahun 2005 - 2006, dan lanjut pada Sekolah Menengah Pertama Negeri (SMPN) 24 Bandar Lampung pada tahun 2006 - 2009, kemudian lanjut di Sekolah Menengah Atas (SMA) Gajah Mada Bandar Lampung pada tahun 2009 - 2012. Tahun 2012, penulis terdaftar sebagai mahasiswa Jurusan Teknik Elektro Universitas Lampung melalui jalur PMPAP. Selama menjadi mahasiswa, penulis aktif di Himpunan Mahasiswa Teknik Elektro (Himatro) sebagai Anggota di Departemen Sosial-Ekonomi pada tahun 2014 - 2015. Penulis juga telah melaksanakan kerja praktik (KP) di PT Telkom IDC, Bandung selama 35 hari yakni pada tanggal 3 Agustus – 11 September 2015. Pada pelaksanaan Kerja Praktik penulis ditempatkan pada bagian *machine to machine* di Telkom IDC Bandung.

PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillah kupanjatkan puji dan syukur kehadiran Allah SWT atas segala rahmatnya, dan juga kesempatan dalam menyelesaikan tugas akhir skripsi saya dengan segala kekurangannya. Kupersembahkan karya sederhanaaku ini untuk,

Ayah dan Ibuku Tercinta

Sulaiman dan Eka Susilawati

Zakak - Adikku Tersayang

Andika Wiryia Kurniawan

Dwi Kurniawan

Teman-teman kebanggaanaku

Rekan-rekan Jurusan Teknik Elektro



MOTTO

"Barang siapa yang bersungguh-sungguh, sesungguhnya kesungguhan tersebut untuk kebaikan dirinya sendiri."

(Al-Qur'an, Surat Al-Ankabut ayat 6)

"Terasa sulit ketika aku merasa harus melakukan sesuatu. Tetapi menjadi mudah ketika aku menginginkannya,-"

(Annie gottlier)

"jadikanlah sabar dan shalat sebagai penolongmu. Dan sesungguhnya yang demikian itu sungguh berat, kecuali bagi orang-orang yang khusyu'."

(Al-Qur'an, Surat Al-Baqarah ayat 45)



SANWACANA

Puji dan syukur penulis ucapkan kehadirat Allah SWT, karena atas rahmat dan hidayah-Nya skripsi ini dapat diselesaikan. Shalawat serta salam tak lupa penulis sanjungkan kepada Nabi Muhammad SAW yang selalu dinantikan syafa'atnya di Yaumul Qiyamah kelak.

Skripsi dengan judul **“PENERAPAN PERSAMAAN *Haversine* BERDASARKAN *Location Based Service* (LBS) PADA APLIKASI ANDROID UNTUK PENCARIAN RUMAH SAKIT DI WILAYAH BANDAR LAMPUNG”** adalah salah satu syarat untuk memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.

Selama menjalani perkuliahan dan penelitian, penulis banyak mendapatkan bantuan moril, materil, bimbingan serta nasihat dari berbagai pihak baik secara langsung maupun tidak langsung. Oleh karena itu, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Allah SWT yang telah memberikan kesehatan jasmani maupun rohani sehingga penulis dapat menyelesaikan skripsi ini;
2. Kedua Orang Tua penulis yang senantiasa memberikan doa, dukungan cinta dan kasih sayang sehingga penulis dapat menyelesaikan skripsi ini;
3. Bapak Prof. Dr. Karomani, M.Si selaku Rektor Universitas Lampung;

4. Bapak Prof. Dr. Suharno, M.Sc., Ph.D. selaku Dekan Fakultas Teknik;
5. Bapak Khairudin, S.T., M.Sc., Ph.D.Eng. selaku Ketua Jurusan Teknik Elektro;
6. Ibu Herlinawati, S.T., M.T. selaku Sekretaris Jurusan Teknik Elektro;
7. Ibu Yetti Yuniati, S.T., M.T. selaku Pembimbing Utama atas kesediaannya meluangkan waktu untuk memberikan bimbingan, saran dan kritik serta semangat dalam penyelesaian skripsi ini;
8. Bapak Dr. Ing. Ardian Ulvan, S.T., M.Sc selaku Pembimbing pendamping atas kesediaannya meluangkan waktu untuk memberikan bimbingan, saran dan kritik serta semangat dalam penyelesaian skripsi ini;
9. Bapak Misfa Susanto, S.T., M.Sc., Ph.D selaku Penguji utama atas kesediaannya meluangkan waktu untuk memberikan bimbingan, saran dan kritik serta semangat dalam penyelesaian skripsi ini;
10. Seperjuangan Gusti, Bella, Windy, Risda, Desi, Dika dan Elektro 12 yang luar biasa solid sampai akhir, terimakasih atas kebersamaan dan kekeluargaan yang telah dibangun selama ini;
11. Seluruh staff admin Jurusan Teknik Elektro Mbking, Mbking Stevi yang banyak membantu penulis dalam memfasilitasi kegiatan administrasi.
12. Semua pihak yang tidak dapat penulis sebut satu persatu yang telah membantu serta mendukung dari awal kuliah sampai dengan selesai.
13. Almamater tercinta, atas kisah hidup yang penulis dapatkan semasa kuliah. Semoga kebaikan, kemurahan hati dan bantuan yang telah diberikan semua pihak mendapat balasan yang setimpal dari Allah SWT.

Penulis menyadari bahwa skripsi ini tidak terlepas dari kesalahan dan jauh dari kesempurnaan, oleh karena itu masukan serta saran dan kritik yang membangun sangat penulis harapkan demi perbaikan dimasa yang akan datang.

Bandar Lampung,

Penulis,

RATIH NINGTIYAS

DAFTAR ISI

	Halaman
DAFTAR ISI	i
DAFTAR GAMBAR	iii
DAFTAR TABEL	vi
 BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Tujuan Penelitian	2
1.3 Manfaat Penelitian	2
1.4 Rumusan Masalah	2
1.5 Batasan Masalah	3
1.6 Sistematika Penulisan	3
 BAB II TINJAUAN PUSTAKA	
2.1 Kajian Pustaka	5
2.2 <i>Global Positioning System (GPS)</i>	6
2.3 <i>Location Based Service (LBS)</i>	8
2.4 Komponen <i>Location Based Service (LBS)</i>	9
2.5 Cara Kerja <i>Location Based Service (LBS)</i>	11
2.6 Android	12
2.7 <i>Google Maps</i>	15
2.8 Pemrograman Aplikasi	16
2.9 <i>App Inventor</i>	17
2.10 <i>Haversine</i>	26
2.11 Kerangka Penelitian	28
 BAB III METODE PENELITIAN	
3.1 Waktu dan Tempat Penelitian	30
3.2 Alat dan Bahan	31
3.3 Prosedur Penelitian	32
3.4 Metode Pengumpulan Data	35
3.5 Perancangan Antarmuka	39

3.6	Awal <i>Project</i>	40
3.7	Pengkodean Aplikasi	43
3.8	Pengujian Aplikasi	52
2.9	Tabel Perhitungan Jarak	53
2.10	Analisa Hasil dan Pembahasan	54
 BAB IV HASIL DAN PEMBAHASAN		
4.1	Hasil	55
4.2	Instalasi Aplikasi	55
4.3	Implementasi Rumus <i>Haversine</i>	61
 BAB V KESIMPULAN DAN SARAN		
5.1	Kesimpulan	60
5.2	Saran.....	60
 DAFTAR PUSTAKA		61
 LAMPIRAN		63

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Segmen GPS.....	8
Gambar 2.2 Komponen Pendukung utama	10
Gambar 2.3 Cara Kerja <i>Location Based Sensor</i>	11
Gambar 2.4 Tampilan <i>App Inventor</i>	17
Gambar 2.5 Arsitektur <i>App Inventor</i>	18
Gambar 2.6 Alur Diagram <i>Sequensi</i>	19
Gambar 2.7 Tampilan <i>Palette</i> di <i>Designer</i>	20
Gambar 2.8 Tampilan <i>Viewer</i> di <i>Designer</i>	21
Gambar 2.9 Tampilan <i>Components</i> di <i>Designer</i>	22
Gambar 2.10 Tampilan <i>Properties</i> di <i>Designer</i>	22
Gambar 2.11 Tampilan <i>Main Menu</i> pada <i>Designer</i>	23
Gambar 2.12 Tampilan <i>Block Editor</i>	23
Gambar 2.13 Tampilan <i>Block Palette</i> di <i>Block Editor</i>	24
Gambar 2.14 Tampilan <i>Block Designer</i> di <i>Block Editor</i>	24
Gambar 2.15 Tampilan <i>Zoom Panel</i> di <i>Block Editor</i>	25
Gambar 3.1 Metodologi Penelitian	31
Gambar 3.2 Diagram Alir Proses Admin.....	36
Gambar 3.3 Diagram Alir Proses <i>User</i>	36
Gambar 3.4 Perancangan Antarmuka.....	37

Gambar 3.5 Tampilan halaman <i>sign in</i> akun <i>Gmail</i>	38
Gambar 3.6 <i>project App Inventor</i>	39
Gambar 3.7 <i>Project Baru di App Inventor</i>	40
Gambar 3.8 <i>Design App Inventor</i>	40
Gambar 3.9 Halaman <i>Block Editor App Inventor</i>	41
Gambar 3.10 Penyimpanan <i>Database</i> pada Aplikasi	42
Gambar 3.11 <i>Block</i> Pembacaan Menggunakan <i>Location sensor</i>	43
Gambar 3.12 Pengaturan pada <i>Properties Maps Activity Starter</i>	44
Gambar 3.13 <i>Block Prosedur Direction</i>	45
Gambar 3.14 Pengaturan Integrasi <i>Button Webview</i>	45
Gambar 3.15 <i>Block</i> dan <i>List Database</i> pada <i>Button Info</i>	47
Gambar 3.16 <i>Block Distance Value</i> dan <i>Index Data</i>	48
Gambar 3.17 <i>Block Procedure Distance List</i>	49
Gambar 3.18 <i>Block Haversine</i>	50
Gambar 4.1 Tampilasn <i>Splash Screen</i> Aplikasi <i>RS Balam Map</i>	54
Gambar 4.2 Tampilan <i>Screen Home</i> Aplikasi <i>Rumah Sakit Balam Map</i>	55
Gambar 4.3 Tampilan <i>Button</i> Tentang Kami.....	56
Gambar 4.4 Tampilan <i>List</i> dari <i>Button</i> Pilihan.....	57
Gambar 4.5 Tampilan konten dari <i>button</i> salah satu <i>RS</i>	57
Gambar 4.6 Tampilan dari <i>Sub-Button</i> <i>RS Advent</i>	56
Gambar 4.7 Tampilan dari <i>Google Map</i> <i>RS Advent</i>	61

DAFTAR TABEL

	Halaman
Tabel 2.1 Perkembangan Sistem Operasi Android	14
Tabel 3.1 Jadwal Kegiatan Penelitian	29
Tabel 3.2 <i>List</i> Daftar <i>Home Url</i> Peta Layanan Lokasi RS Bandar Lampung ..	46
Tabel 3.3 Hasil perhitungan <i>Haversine</i> dengan perhitungan <i>Google maps</i>	52
Tabel 4.1 Tabel Hasil Perhitungan Manual.....	61
Tabel 4.2 Hasil perhitungan <i>Haversine</i> dengan perhitungan <i>Google maps</i>	62

BAB I PENDAHULUAN

1.1 Latar Belakang

Informasi dan telekomunikasi berkembang semakin pesat, seperti pengguna internet *mobile* di Indonesia sekitar 61,88% [1]. Hal ini menjadikan peluang untuk membuat aplikasi-aplikasi yang sesuai dengan kebutuhan masyarakat. Saat ini, aplikasi yang akan dibuat dalam sebuah informasi alamat rumah sakit, profil, jam praktek dokter dan fasilitas lainnya. Sasaran pembuatan aplikasi ini akan diberikan kepada seorang pasien. Pada lokasi yang dibutuhkan dengan posisi geografis suatu daerah, akan menggunakan peta yang banyak dikenal oleh masyarakat adalah *Google Maps*.

Maps diakses sesuai dengan info berupa *web/mobile* dan dibuat berdasarkan *Location Based Service* (LBS). Sedangkan, LBS ini menghasilkan layanan informasi mengenai lokasi keberadaan *user*. Hal ini menyebabkan peningkatan nilai informasi, dikarenakan penerima dapat mengasosiasikan pengetahuan atau informasi yang didapat dengan keberadaannya. Peningkatan nilai informasi ini terkait secara erat dengan potensi nilai komersial bagi penyedia layanan (*provider*) LBS. *Provider* dapat menyediakan informasi secara instan pada saat konsumen memerlukan dan relevan terhadap lokasi konsumen [2].

Skripsi ini membahas aplikasi yang dikembangkan berbasis *android* menggunakan MIT App Inventor. App Inventor juga terdapat *Location Based Service* (LBS) dan komponen *Global Positioning System* (GPS) pada ponsel pintar. *Based Transceiver Station* (BTS) terdapat di dalamnya yang menentukan posisi dengan jaringan internet dari operator seluler. Sedangkan rumus yang akan dipakai adalah *Haversine*, rumus ini akan membandingkan jarak antara user pengguna dengan jarak rumah sakit terdekat.

1.2 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Tersedianya aplikasi lokasi Rumah Sakit dengan program MIT App Inventor yang ada pada Android.
2. Mendapatkan hasil perhitungan jarak akses *user* ke rumah sakit terdekat dengan rumus *Haversine* secara manual dan membandingkannya dengan jarak yang ada pada aplikasi.

1.3 Manfaat Penelitian

Manfaat dari penelitian ini yaitu tersedia aplikasi pada ponsel pintar yang dapat mengetahui jarak rumah sakit terdekat, serta informasi jam praktik dokter pada MIT App Inventor.

1.4 Rumusan Masalah

Permasalahan yang dibahas pada penelitian skripsi antara lain:

1. Bagaimana merancang aplikasi *mobile* yang berisi informasi rumah sakit di Bandar Lampung,
2. Bagaimana proses perhitungan rute terdekat rumah sakit menggunakan rumus *Haversine* secara manual, dan apakah hasil perhitungan sama atau hampir mendekati dengan hasil yang ada pada aplikasi.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah:

1. Aplikasi ini memberikan letak lokasi hanya meliputi beberapa rekomendasi rumah sakit terdekat.
2. Peta yang akan digunakan dalam aplikasi ini memanfaatkan *Google Maps*.
3. Menggunakan akses internet dalam mengakses informasi yang tersedia pada aplikasi.
4. Fitur aplikasi meliputi navigasi, deskripsi rumah sakit, galeri foto dan informasi lainnya.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN

Membahas pendahuluan yang terdiri dari latar belakang penelitian, perumusan masalah, batasan masalah yang akan dibatasi pada penelitian, tujuan dan sistematika pada penulisan skripsi.

BAB II TINJAUAN PUSTAKA

Bab ini meninjau beberapa hasil penelitian terdahulu yang berhubungan dengan topik skripsi ini. Membahas tentang teori dasar teknologi *Location Based Service* (LBS), cara kerja LBS, aplikasi MIT *Invertor*, dan rumus *Haversine*.

BAB III METODOLOGI PENELITIAN

Bab ini memuat tentang langkah-langkah penelitian yang dilakukan mulai dari prosedur penelitian sampai pengkodean aplikasi.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memuat implementasi sistem tentang analisa kebutuhan dari sistem yang telah dibuat dan bagaimana proses analisisnya hingga dapat diitampilkan kedalam perangkat *mobile*.

BAB V KESIMPULAN DAN SARAN

Bab ini memuat kesimpulan dari aplikasi yang telah dibuat dengan metode navigasi persamaan *Haversine*, dan saran perbaikan di waktu yang akan datang.

BAB II **TINJAUAN PUSTAKA**

2.1 Kajian Pustaka

Penulis [1] dengan judul aplikasi lokasi fasilitas kesehatan penerima BPJS di kota Semarang berbasis android. Penelitian ini bertujuan untuk memudahkan pengguna dalam mengakses informasi spasial. Fasilitas kesehatan penerimaan BPJS kesehatan dilakukan dengan menggunakan GPS Navigasi *Handheld* adalah data spasial. Hasil penelitian tersebut berupa sistem informasi geografis perangkat *mobile* Android yang menampilkan fasilitas kesehatan penerimaan BPJS menggunakan program MIT App *Invertor*.

Penulis [2] dengan judul “Sistem Informasi Geografis Layanan Hotel Jarak Terdekat dengan Posisi Pengguna Menggunakan *Haversine* Formula serta *Google Maps* sebagai Pendukung”. Tujuan dari penelitian tersebut adalah mempermudah akses informasi kepada orang awam yang membutuhkan informasi Layanan Hotel Kota Malang. Penelitian tersebut adalah aplikasi yang dibuat berbasis *Client&server* serta *Web Service* untuk memudahkan *update* map tanpa harus menginstal ulang aplikasi.

Penulis [3] dengan judul “Pembangunan Aplikasi *Location Based Service* (LBS) di Makassar”. Metode yang digunakan adalah *mobile* LBS dengan menggunakan IDE *Eclipse* dan *Google API*. Hasil yang diharapkan pada aplikasi ini untuk memberikan informasi kepada pengguna tentang info fasilitas umum ATM, SPBU dan Rumah Sakit, beserta rute jalan.

Penulis [4] dengan judul “Pencarian Rumah Sakit Bermitra dengan BPJS Kesehatan Berbasis Android”. Penulis menggunakan *Eclipse*, sebagai *Android Developer Tools* (ADT). Metode pengembangan perangkat lunak menggunakan metode *waterfall*. Adanya Sistem Informasi Geografis Rumah Sakit, dengan menerapkan LBS dan sistem android untuk mencari Rumah Sakit.

Skripsi yang akan dikerjakan adalah sebuah aplikasi berdasarkan *Location Based Service* (LBS) pada android dengan persamaan *Haversine*. Penulis akan meninjau terlebih dahulu sebuah konsep dasar suatu wilayah di provinsi Lampung tepatnya di Bandar Lampung, dan akan menjelaskan juga *Location Based Service* (LBS) pada android dengan persamaan *Haversine*.

2.2 *Global Positioning System* (GPS)

Global Positioning System (GPS) adalah sebuah sistem satelit navigasi dan penentuan posisi dengan menggunakan satelit. GPS dapat memberikan informasi tentang posisi, kecepatan, dan waktu secara cepat, akurat, murah, di mana saja di bumi ini pada setiap saat tanpa tergantung cuaca. Pada dasarnya GPS terdiri atas tiga segmen utama, yaitu segmen angkasa yang terdiri dari satelit-satelit GPS,

segmen sistem kontrol yang terdiri dari stasiun-stasiun pengamat dan pengendali satelit, dan segmen pemakai yang terdiri dari pemakai GPS termasuk alat-alat penerima dan pengolah sinyal dan data GPS [5]. Berikut adalah penjelasan mengenai tiga segmen utama pada GPS, yaitu segmen angkasa, segmen sistem kontrol, dan segmen pengguna.

a. Segmen Angkasa

Satelit GPS dapat dianalogikan sebagai stasiun radio angkasa yang dilengkapi dengan antena-antena, sebagai pengirim dan penerima sinyal-sinyal gelombang. Sinyal-sinyal akan diterima oleh *receiver* GPS di dekat permukaan bumi dan digunakan untuk menentukan informasi posisi, kecepatan maupun waktu.

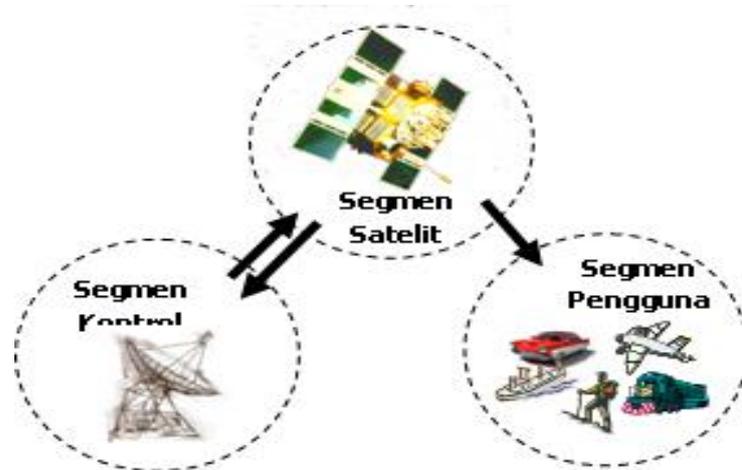
b. Segmen Sistem Kontrol

Segmen ini terdiri atas tiga bagian yaitu *Ground Control Station* (GCS), *Monitor Station* (MS), *Prelaunch Control Station* (PCS). Fungsi dari segmen ini adalah mengontrol dan memantau operasional satelit serta memastikan bahwa satelit berfungsi sebagaimana mestinya.

c. Segmen Pengguna

Segmen ini terdiri dari para pengguna satelit GPS, dalam hal ini *receiver* GPS diperlukan untuk menerima dan GPS memproses sinyal-sinyal dari satelit. Ada tiga tipe *receiver* GPS dengan masing-masing tingkat ketelitian/posisi yang berbeda-beda. Tipe alat GPS pertama adalah tipe navigasi (*handheld*) dengan ketelitian 3-6 meter. Tipe alat yang kedua adalah tipe geodetik *single* frekuensi (tipe pemetaan), yang biasa digunakan dalam survey dan pemetaan yang membutuhkan ketelitian

posisi sekitar sentimeter sampai dengan beberapa desimeter. Tipe terakhir adalah tipe geodetik *dual* frekuensi yang dapat memberikan ketelitian posisi mencapai milimeter



Gambar 2.1 Segmen GPS [6].

2.3 *Location Based Service (LBS)*

LBS memiliki kemampuan mencari lokasi geografis dari *mobile device* dan menyediakan layanan berdasarkan lokasi yang diperolehnya [7]. Konsep LBS ini menghasilkan layanan informasi mengenai lokasi keberadaan *user*. Hal ini menyebabkan peningkatan nilai informasi, dikarenakan penerima dapat mengasosiasikan pengetahuan atau informasi yang didapat dengan keberadaannya. Peningkatan nilai informasi ini terkait secara erat dengan potensi nilai komersial bagi penyedia layanan (*provider*) layanan LBS. *Provider* dapat menyediakan informasi secara instan pada saat konsumen memerlukan dan relevan terhadap lokasi konsumen. LBS dapat diklasifikasikan menjadi tiga jenis, yaitu *local information*, *traffic and tracking information* dan *general services*.

a. *Local Information*

Memungkinkan pengguna untuk mencari layanan di sekitar mereka.

b. *Traffic and Tracking Information*

Berfokus kepada pelacakan aset atau manusia.

c. *General Services*

Tidak tersedia informasi ke pengguna, namun menggunakan data lokasi pengguna.

2.4 **Komponen *Location Based Service (LBS)***

Memiliki empat komponen pendukung utama dalam teknologi layanan berbasis lokasi, antara lain:

a. *Perangkat Mobile*

Fungsinya sebagai alat bagi pengguna untuk mendapatkan informasi. Informasi yang didapat dapat berupa teks, suara, gambar dan lain sebagainya. Perangkat *mobile* yang dapat digunakan bisa berupa PDA, *smartphone*, *laptop*.

b. *Jaringan Komunikasi*

Fungsinya sebagai jalur penghubung yang dapat mengirimkan data pengguna dari perangkat *mobile*, kemudian dikirimkan ke *provider* dan informasi tersebut dikirimkan kembali oleh *provider* kepada *user*. Setiap layanan yang diberikan oleh *provider* biasanya, berdasarkan pada posisi *user* yang meminta layanan tersebut. Oleh karena itu, diperlukan komponen pengolah/pemroses menentukan posisi *user* saat itu. Posisi *user*

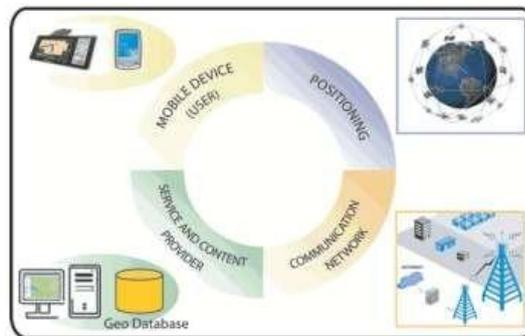
tersebut bisa didapatkan melalui jaringan komunikasi *mobile* atau *Global Positioning System (GPS)*.

c. *Provider* dan Aplikasi

Provider adalah komponen LBS, memberikan berbagai macam layanan yang bisa digunakan oleh *user*. Sebagai contoh ketika pengguna meminta layanan agar bisa tahu posisinya saat itu, maka aplikasi dan penyedia layanan langsung memproses permintaan.

d. Penyedia Data dan Konten

Provider 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.



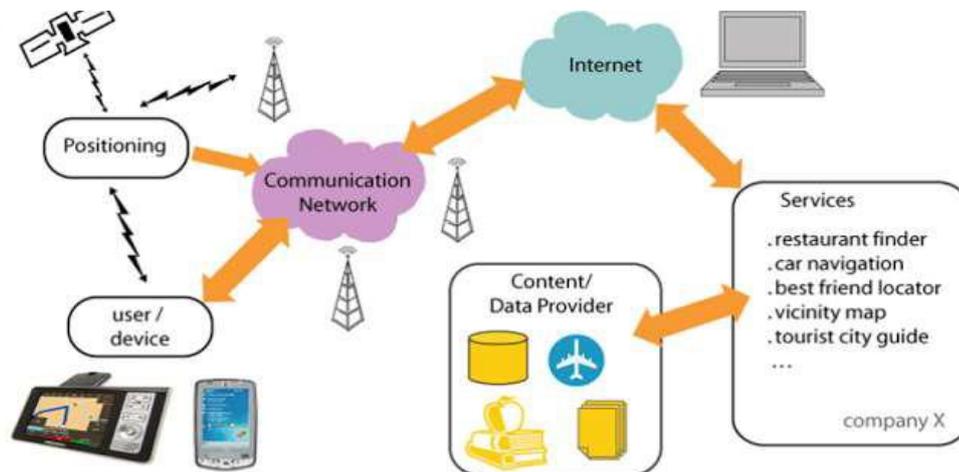
Gambar 2.2 Komponen pendukung utama teknologi LBS [8].

2.5 Cara Kerja *Location Based Service (LBS)*

Cara kerja LBS pada aplikasi pencarian objek terdekat berdasarkan posisi *user*.

- a. *Smartphone* akan membuka aplikasi yang memanfaatkan layanan LBS yang sudah ter-*install*.

- b. Aplikasi akan melakukan sambungan dengan jaringan *provider* yang dipakai oleh *user*.
- c. Selanjutnya informasi posisi *user* pada perangkat *mobile* yang diperoleh dari *Location Sensor*. Hal ini dapat dilakukan oleh perangkat dengan menggunakan GPS sendiri atau layanan posisi jaringan yang berasal dari *provider*.
- d. Perangkat *mobile* pengguna akan mengirimkan permintaan informasi ke satelit untuk menentukan *longitude* dan *latitude* dari pengguna aplikasi tersebut.
- e. *Provider* menghubungkan aplikasi pada *smartphone* dengan *server* LBS dan meminta data yang diinginkan *user* beserta informasi tentang jarak dan cara yang diperlukan dalam menjangkau lokasi tujuan.
- f. Terakhir, *user* mendapatkan data dan ditampilkan di *smartphone* melalui aplikasi.



Gambar 2.3 Cara Kerja *Location Based Service* [9].

2.6 Android

Android adalah *Operating System mobile* berbasis Linux yang tumbuh ditengah *Operating System (OS)* lainnya yang berkembang, seperti *Windows Mobile*, *iphone*, *Symbia* dan lain-lain [10]. *Android* dianggap sebagai *platform* masa depan yang lengkap, terbuka dan bebas sebagai berikut:

2.6.1 Lengkap (*Complete Platform*)

Seorang desainer dapat melakukan pendekatan yang komprehensif ketika sedang mengembangkan *platform* *Android*. *Android* merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.

2.6.2 Terbuka (*Open Source*)

Platform *Android* disediakan melalui lisensi *open source* dan pengembang dengan bebas mengembangkan aplikasi. *Android* sendiri menggunakan Linux kernel 2.6.

2.6.3 Bebas (*Free Platform*)

Android adalah aplikasi yang bebas develop dan tidak ada lisensi/biaya royalti untuk dikembangkan pada *platform* *Android*. *Android* juga tidak ada biaya keanggotaan diperlukan, tidak diperlukan biaya pengujian, tidak ada kontrak yang diperlukan. Aplikasi untuk *android* dapat didistribusikan dan diperdagangkan dalam bentuk apapun. Seiring dengan pengembangannya, *android* memiliki berbagai macam versi, antara lain:

a. Android 2.3 – 2.3.7 (*Gingerbread*)

Versi ini dirilis pada 6 Desember 2010, dengan nama sistem operasi Android versi 2.3. Ada banyak peningkatan pada versi Android yang satu ini dibandingkan dengan versi sebelumnya, seperti memaksimalkan kemampuan aplikasi dan *game*, dukungan layar resolusi WXGA dan di atasnya, serta mulai digunakannya *Near Field Communication* (NFC). Beberapa versi *update* yang dirilis antara lain v.2.3.3 hingga v.2.3.7.

b. Android versi 3.0 -3.2 (*Honeycomb*)

Versi ini dirilis pada 22 Februari 2011 dengan nama sistem Android versi 3.0. Android versi ini merupakan OS yang didesain khusus untuk pengoptimalan penggunaan pada tablet PC.

c. Android versi 4.0 – 4.0.4 (*Ice Cream Sandwich*)

Versi ini dirilis pada 19 Oktober 2011 dengan nama Android versi 4.0. Secara teori semua perangkat seluler yang menggunakan versi android sebelumnya dan dapat di-*update*.

d. Android versi 4.1 – 4-3 (*Jelly Bean*)

Jelly Bean memiliki penambahan baru di antaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Terdapat pula aplikasi *Google Now* yang dapat memberikan informasi yang tepat pada waktu yang tepat pula. Selain itu dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android 4.1 kini sudah mencapai versi 4.3.

e. Android versi 4.4 (KitKat)

KitKat diluncurkan pada Oktober 2013. Penamaan pada versi ini didasarkan dari kerja sama antara Google dengan perusahaan Nestle. Android versi 4.4 ini diprediksikan akan kompatibel untuk digunakan pada *smartphone* Android mulai dari kelas *high-end* hingga *low-end* [10].

Tabel 2.1 Perkembangan Sistem Operasi Android

Versi	Nama Kode	Tanggal Rilis	Level API	Distribusi
4.4	KitKat	Oktober 2013	19	-
4.3	Jelly Bean	24 Juli 2013	18	-
4.2.x	Jelly Bean	13 November 2012	17	6,5%
4.1.x	Jelly Bean	9 Juli 2012	16	34,0%
4.0.3-4.0.4	Ice Cream Sandwich	16 Desember 2011	15	22,5%
3.2	Honeycomb	15 Juli 2011	13	0,1%
3.1	Honeycomb	10 Mei 2011	12	0,0%
2.3.3-2.3.7	Ginger Bread	9 Februari 2011	10	33,0%
2.3-2.3.2	Ginger Bread	6 Desember 2010	9	0,1%
2.2	Froyo	20 Mei 2010	8	2,5%
2.0-2.1	Eclair	26 oktober 2009	7	1,2%
1.6	Donut	15 September 2009	4	0,1%
1.5	Cupcake	30 April 2009	3	0%

2.7 Google Maps

Google Maps adalah suatu peta dunia yang dapat digunakan untuk melihat suatu daerah, bisa juga disebut peta yang dapat dilihat dengan menggunakan suatu

browser. Fitur dalam *web* atau aplikasi yang telah dibuat pada blog yang berbayar maupun gratis sekalipun menggunakan *Google Maps API* [11].

Cara membuat *Google Maps* yang akan ditampilkan pada suatu *web* atau *blog* sangat mudah, hanya membutuhkan pengetahuan mengenai HTML serta *Java Script* dan koneksi. Pembuatan ini dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga dapat fokus hanya pada data-data yang akan ditampilkan. Dengan kata lain, hanya membuat suatu data, sedangkan peta yang akan ditampilkan adalah milik Google. Sehingga tidak dipusingkan dengan membuat peta suatu lokasi, bahkan dunia. Pada *Google Maps API* terdapat empat jenis pilihan model peta yang disediakan oleh Google, di antaranya:

a. *Roadmap*

Menampilkan peta biasa dua dimensi pada tampilan *Google Maps*.

b. *Satellite*

Menampilkan foto satelit pada tampilan *Google Maps*.

c. *Terrain*

Menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai pada tampilan *Google Maps*.

d. *Hybrid*

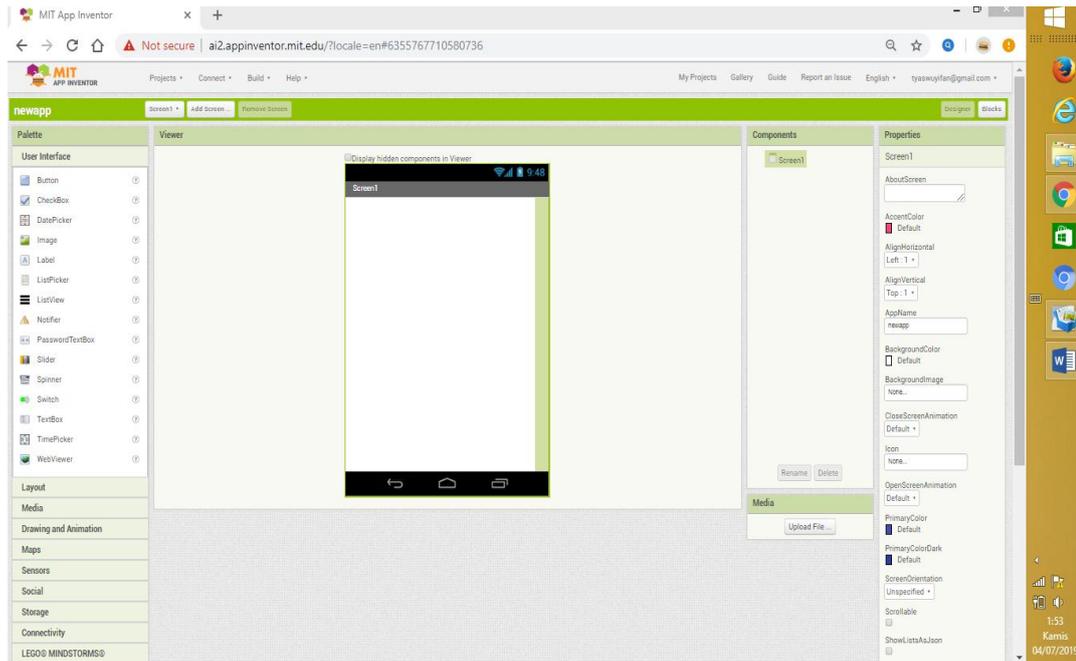
Menunjukkan foto satelit yang di atasnya tergambar pula tampilan pada *roadmap* (jalan dan nama kota) pada tampilan *Google Maps*.

2.8 Pemrograman Aplikasi *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah notasi grafis yang didukung oleh meta-model tunggal, membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OOP). Bahasa UML digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek, namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi [12].

2.9 *App Inventor*

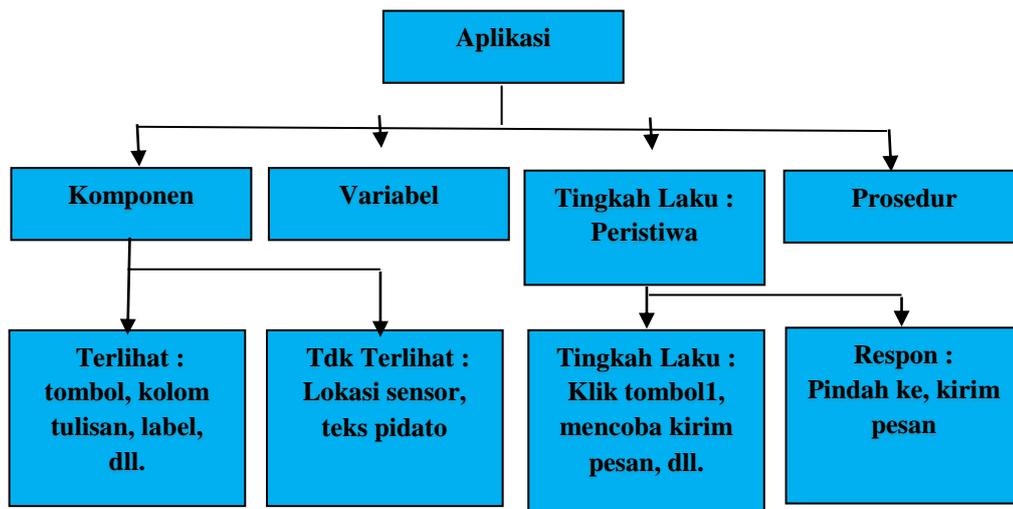
App Inventor adalah sebuah aplikasi yang disediakan oleh *Google*, sedang *maintenance* oleh *Massachusetts Institute of Technology* (MIT). *App Inventor* memungkinkan setiap pengguna untuk menggunakan *software* aplikasi yang disediakan *Android*. Pengguna bisa menggunakan tampilan grafis GUI serta *drag and drop visual* objek untuk membuat aplikasi yang akan dijadikan sistem operasi *Android*. *App Inventor* pengguna dimulai dengan *web-based service* pada *browser* secara *online* <http://ai2.appinventor.mit.edu>, dengan cara mengatur tampilan aplikasi *user interface* pada web *Graphical User Interface* (GUI) *builder*, kemudian menspesifikasikan *behavior* aplikasi yang akan dibuat *block* yang sesuai dengan tampilan Gambar 2.4.



Gambar 2.4. Tampilan *App Inventor*

a. *Arsitektur App Inventor*

App Inventor mempunyai struktur internal yang harus dipahami terlebih dahulu agar dapat dibuat secara efektif. Salah satu cara untuk memahami bagian dalam sebuah aplikasi yaitu dengan memecahnya menjadi dua bagian yaitu komponen dan *behavior*. Kedua bagian ini merupakan sebuah jendela yang saling berhubungan yang nantinya digunakan untuk mengembangkan aplikasi dengan *App Inventor*, kemudian dengan menggunakan komponen *designer* dapat dibuat komponen dari sebuah aplikasi yang nantinya berjalan sesuai dengan perintah dari sistem *block* yang telah dirancang. Arsitektur aplikasi *App Inventor* dapat dilihat pada Gambar 2.6



Gambar 2.5 Arsitektur *App Inventor*

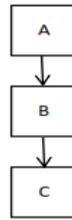
Arsitektur aplikasi *App Inventor* terdiri dari:

- *Komponen App inventor*

App inventor dibagi menjadi dua yaitu *visible* dan *non visible*. Pada komponen *visible*, komponen dijalankan dapat terlihat oleh pengguna, sedangkan *non visible* komponen dijalankan tidak dapat terlihat oleh pengguna. karena *non visible* menyediakan akses membangun fungsional dari perangkat aplikasi. Keduanya di defenisikan oleh *properties*. *Properties* sendiri merupakan memori yang digunakan untuk menyimpan komponen tersebut.

- *Behavior*

Behavior didefinisikan sebagai aplikasi yang merespon sebuah *event*. Pada *App inventor* memiliki *visual block* yang menspesifikasi *behavior*. Oleh karna itu, sebuah aplikasi dapat dikonseptkan secara *event*.



Gambar 2.6 Alur Diagram Sequensi [13].

Berdasarkan Gambar 2.6, *event* terjadi saat aplikasi memanggil fungsi secara berurutan. Fungsinya adalah sesuatu yang akan dilakukan oleh komponen atau tanpa komponen. *Event* dan beberapa set fungsi yang dikerjakan pada sebuah respon yaitu *event handler*.

- *Event*

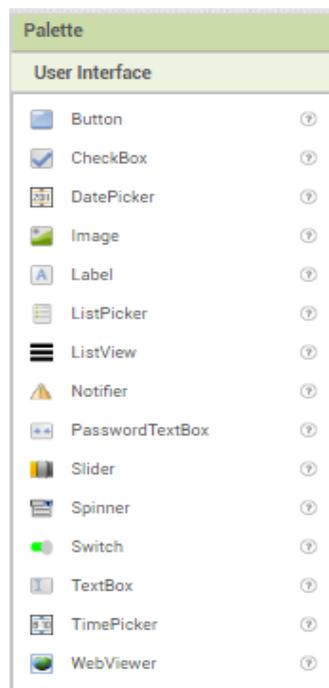
Event memanggil *method* berdasarkan perintah tertentu. Ada beberapa jenis *event* di *App Inventor*, sebagai berikut:

1. *User Initiated Event* merupakan *event input form button*, serta memicu suatu sistem di aplikasi.
2. *Initialization Event* merupakan pengguna *event* saat aplikasi mulai dijalankan oleh pengguna.
3. *Timer Event* merupakan pengguna *event* berdasarkan waktu, dimana membantu pergerakan sebuah *event* dengan waktu tertentu.
4. *Animation Event* merupakan pengaturan sebuah sistem *event* dengan menggunakan sebuah objek grafis.
5. *External Event* merupakan sebuah *event* yang didapat di luar aplikasi, seperti informasi lokasi pengguna pada satelit GPS.

- *Integrated Development Environment*

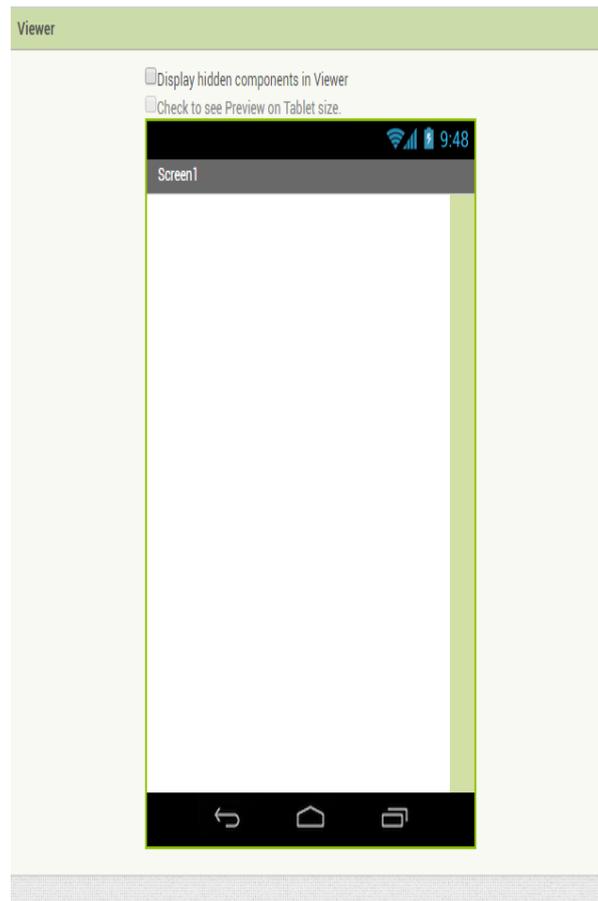
Program *App Inventor* pada dasarnya bekerja secara *online* melalui *browser internet*, Untuk menjalankan program yang di buat pada aplikasi memiliki ketentuan sebagai berikut:

- a. *Palette* adalah sebuah objek yang digunakan oleh *block editor*, serta terdiri dari beberapa komponen *basic*, *media*, *social*, *animation*, *sensor*, *screen arrangement*, *lego mindstrom*, dan *other stuff*.



Gambar 2.7 Tampilan *Palette* di *Designer*

- b. *Viewer* disebut juga *form designer* digunakan, perancangan atau *user interface* dapat diambil melalui *palette*.



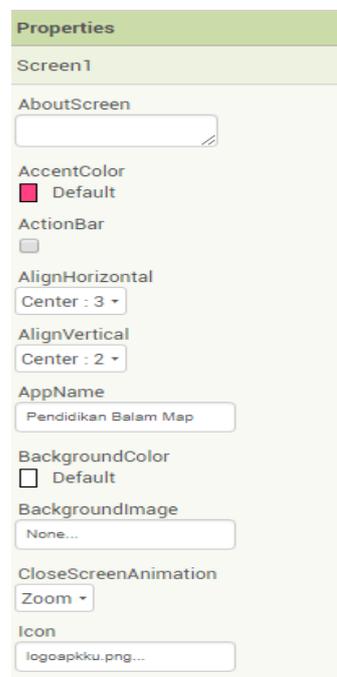
Gambar 2.8 Tampilan *Viewer* di *Designer*

- c. *Components* digunakan untuk melihat daftar komponen yang terdapat pada *screen/form*. *Component* juga dapat digunakan menghapus serta mengubah nama dari komponen yang diletakkan pada *screen*.



Gambar 2.9 Tampilan *Components* di *Designer*

- d. *Properties* berfungsi mengubah tampilan teks serta *upload database*, sehingga dapat mengatur tampilan di *interface* aplikasi.



Gambar 2.10 Tampilan *Properties* di *Designer*

- e. *Main Menu* digunakan sebagai antarmuka dalam mengakses menu utama di *App Inventor*. *Main menu* digunakan sebagai penyimpanan *project*, *debugging project* dan memaket *project* menjadi aplikasi.



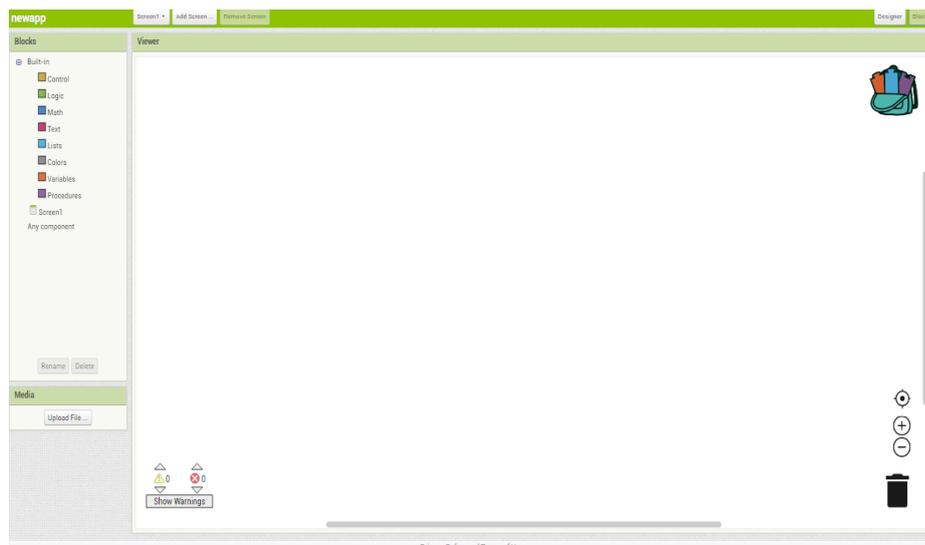
Gambar 2.11 Tampilan *Main Menu* pada *Designer*

- f. *Block Editor*

Block editor digunakan sebagai perancang *behavior* aplikasi yang dibuat. *Block editor* dibagi menjadi beberapa bagian main menu, *block palette*, *block designer*, dan *zoom panel*.

- g. *Main Menu*

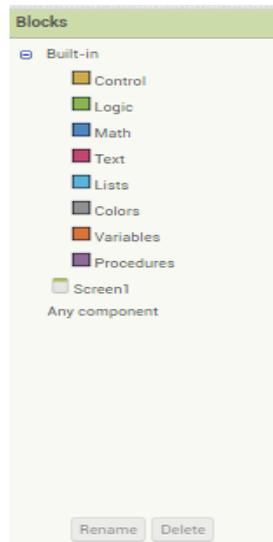
Main Menu digunakan sebagai menu utama di aplikasi *block editor*. *Main menu* juga dapat mengakses *emulator*, serta menyimpan *project* dan mengatur.



Gambar 2.12 Tampilan *Block Editor*

h. Palette

Palette digunakan untuk mengambil *part block* yang letaknya pada *block designer*.



Gambar 2.13 Tampilan *Block Palette* di *Block Editor*

i. Block designer

Block designer digunakan untuk meletakkan *part block* yang digunakan sebagai program utama.



Gambar 2.14 Tampilan *Block Designer* di *Block Editor*

j. Zoom Panel

Zoom Panel digunakan sebagai *block designer* untuk memperbesar dan perkecil tampilan *block editor* terlihat pada gambar + dan -.

$\lambda_1, \lambda_2 = longitude$ dari titik 1 (posisi user), $longitude$ dari titik 2 (lokasi tujuan)
 Fungsi *Haversine* radius, derajat (d/R) dalam rumus akan menjasi *haversine* (180°
 $d/\pi R$). Kemudian untuk nilai d dapat diterapkan *Haversine invers* (jika tersedia)
 atau dengan menggunakan fungsi *arcsin (invers sinus)*:

$$d = r \text{ haversine}^{-1}(h) = 2r \arcsin\sqrt{hr} \dots\dots\dots (2.3)$$

di mana h adalah *Haversine* (d/R)

untuk memperjelas penggunaan rumus *Haversine* diatas, berikut penjabaran
 rumus yang nantinya akan di gunakan pada perhitungan manual, dari rumus di
 atas untuk menghitung jarak antara dua titik dapat ditulis dengan persamaan
 berikut :

$$d = R * 2 * \text{asin}(\sqrt{a + c}) \dots\dots\dots (2.4)$$

Untuk mendapatkan nilai a , digunakan rumus sebagai berikut:

$$a = \sin\left(\frac{\Delta lat}{2}\right)^2 \dots\dots\dots (2.5)$$

dimana Δlat merupakan dari perhitungan $\frac{\pi}{180} * (latitude \text{ tujuan} - latitude \text{ user})$.

$$c = \cos(lat \text{ User}) * \cos(lat \text{ RS}) * \sin\left(\frac{\Delta long}{2}\right) \dots\dots\dots (2.6)$$

Lat 1 merupakan *latitude user*, sedangkan lat 2 adalah posisi *latitude* lokasi tujuan
 pengguna, dan

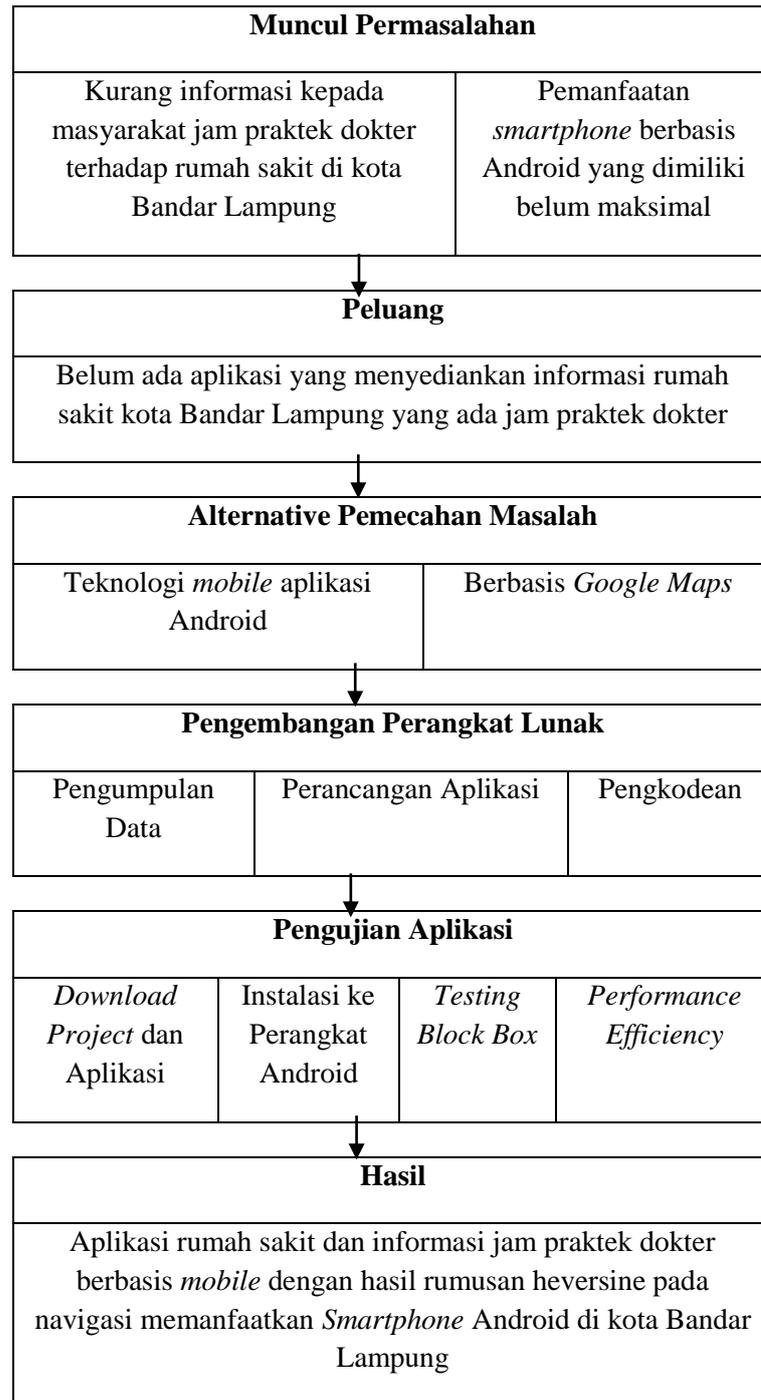
$$\Delta long = \frac{\pi}{180} * (longitude \text{ tujuan} - longitude \text{ user}) \dots\dots\dots (2.7)$$

dari persamaan 2.4 hingga 2.6 adalah untuk menentukan jarak paling dekat lokasi
 yang akan terdapat perbandingan. Jadi, harus terdapat titik lokasi dan ditemukan
 jarak, barulah dibandingkan jarak terpendek. Jalur jalan yang sebenarnya, sesuai
 kondisi memanfaatkan *google maps*, lalu akan ditampilkan antara lokasi user dan
 lokasi tujuan.

2.11 Kerangka Penelitian

Lokasi rumah sakit yang berada di Kota Bandar Lampung tersebar dan memiliki fasilitas yang beragam. Lokasi rumah sakit tersebut menjadikan masyarakat, ingin mendapatkan informasi mengenai jam praktik dokter. Biasanya, seorang pasien untuk mengetahui dokter jaga harus mendatangi rumah sakit. Hal ini dapat terjadi karena belum adanya informasi jam praktek dokter sehingga pasien harus ke rumah sakit menanyakan langsung.

Berdasarkan permasalahan tersebut maka diperlukan alternatif untuk penyelesaian masalah tersebut. Adapun penyelesaian masalah tersebut adalah pembuatan aplikasi yang diperlukan dan diharapkan setelah rancangan ini selesai dapat berguna bagi masyarakat. Selain itu untuk meninjau di mana letak posisi rumah sakit tersebut dengan fasilitas yang telah disiapkan berupa peta di dalam aplikasi langsung dari *smartphone* para pengguna sendiri. Adapun kerangka penelitian dapat dilihat pada Gambar 2.16.



Gambar 2.16 Kerangka Penelitian

BAB III METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Skripsi ini dilaksanakan pada:

Waktu : Agustus 2017 – Desember 2019

Tempat : Laboratorium Teknik Telekomunikasi Jurusan Teknik
Elektro Universitas Lampung

Tabel dari jadwal dan aktifitas penelitian dapat dilihat pada Tabel 3.1:

Tabel 3.1 Jadwal Kegiatan Penelitian

No.	Aktifitas	Agustus	Januari	Maret	April	Jun	Des
1	Studi Literature						
2	Prancangan system pemrograman						
3	Seminar proposal						
4	Pembuatan pemrograman						
5	Uji coba pemrograman						
6	Analisa data dan pembahasan						
7	Seminar Hasil						
8	Perbaikan analisa data dan pembahasan						
9	Komprehensif						

3.2 Alat dan Bahan

Skripsi ini, mulai dari mendesain sampai tahap pemrograman, penulis menggunakan perlengkapan komputer sebagai server untuk menyediakan database aplikasi dan handset Android. Secara lebih spesifik perlengkapan komputer dan handset beserta perangkat pendukung yang digunakan yaitu:

3.2.1 Spesifikasi Komputer Server

Spesifikasi komputer *server* yang digunakan yaitu:

1. Sistem operasi *Microsoft Windows 8.1 pro*
2. *Prosesor AMD E1-2500 APU with radeon (TM) HD Graphics 1.40GHz*
3. Memori RAM 4.00 GB

3.2.2. Spesifikasi Handset Android

Spesifikasi handset android yang digunakan yaitu :

1. vivo Y15
2. Sistem operasi Android versi v4.2 jelly bean
3. CPU Mediatek MT6572
4. Memori RAM 4 GB
5. Memori Eksternal 2 GB

3.2.3. Aplikasi lain yang dipakai penulis dalam penyelesaian skripsi

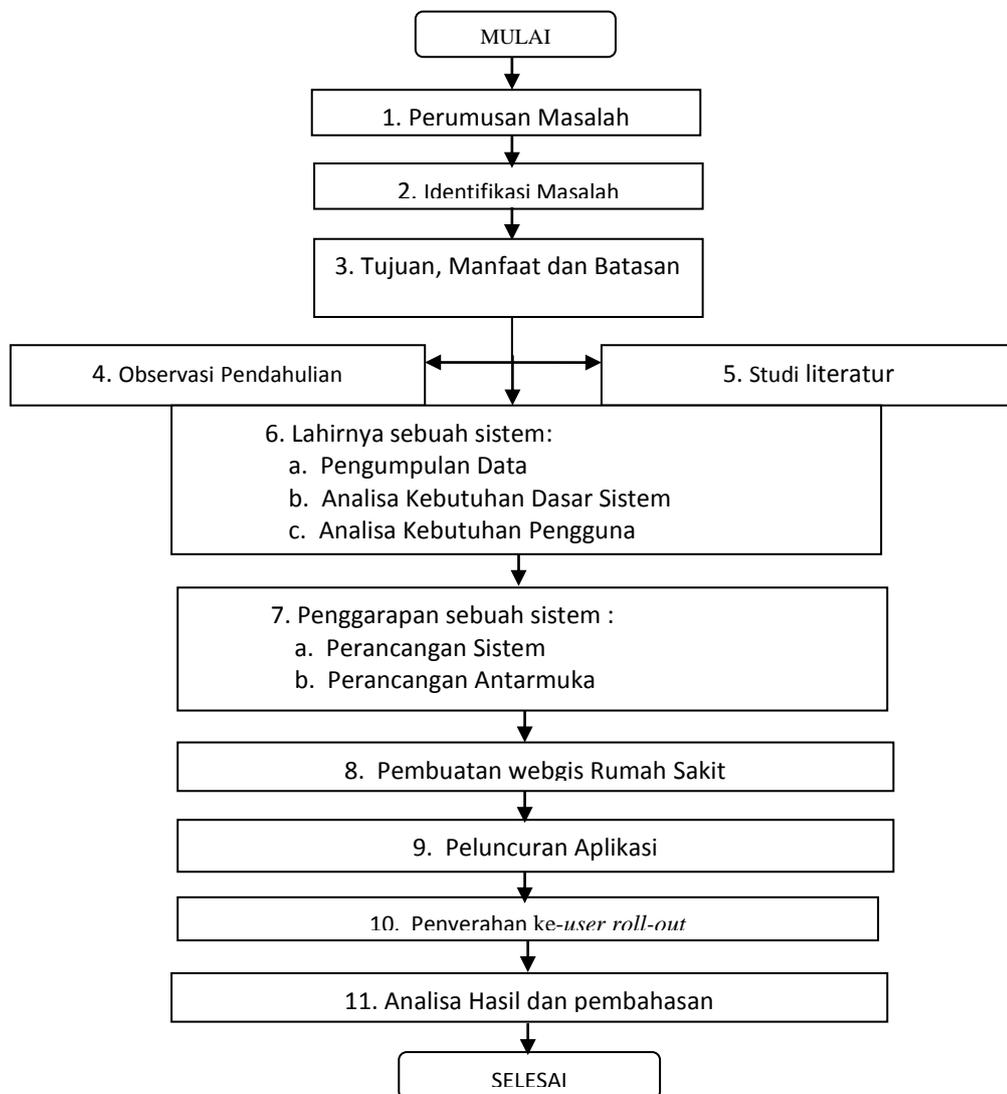
1. *App Inventor (version v.134)*, digunakan sebagai media pembuatan aplikasi.
2. *Google App Engine*, digunakan untuk membuka *App Inventor* pada *browser*.

3. *Google Chrome*, digunakan sebagai media aplikasi *App Inventor* secara *online*.

4. *Microsoft Office 2010*, digunakan untuk perancangan sistem dan atau diagram aplikasi.

3.3 Prosedur Penelitian

Penelitian yang dijalankan memiliki berapa tahapan kerja yang dilakukan berdasarkan diagram alir metodologi penelitian terdapat pada Gambar 3.1



Gambar 3.1 Metodologi Penelitian

Penjelasan diagram alir tersebut adalah :

1. Mengidentifikasi masalah, mulai dengan pengenalan masalah untuk menentukan masalah yang akan dipecahkan dalam skripsi ini.
2. Perumusan masalah, penelitian ini memiliki pertanyaan yang membutuhkan jawaban melalui pengumpulan data yang umumnya disusun dalam bentuk kalimat tanya.
3. Menentukan tujuan, manfaat dan batasan. Pencapaian pada tahap ini diketahuinya tujuan dan manfaat dari rancang bangun aplikasi bergerak *webgis* rumah sakit dengan komparatif studi pada kerja jaringan data dan *wifi*. Sedangkan batasan digunakan untuk membatasi pembahasan dan ruang lingkup penelitian.
4. Survei pendahuluan, yaitu untuk mengetahui hal-hal penting yang berhubungan dengan penelitian yang dikaji.
5. Studi literatur, yaitu untuk mendapatkan gambaran yang menyeluruh tentang apa yang sudah dikerjakan sebagai teori-teori yang akan dijadikan landasan penelitian.
6. Fase ke-dua merupakan pengembangan sistem. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah proses terpadu. Tahap pertama yang dilakukan dalam pengembangan sistem ini adalah melahirkan sebuah sistem pengembang. Pada tahap ini akan dilakukan pengumpulan data berupa data titik koordinat rumah sakit di Kota Bandar Lampung. Pada tahap ini juga akan dilakukan analisis kebutuhan dasar sistem dan analisis kebutuhan pengguna.

7. Penggarapan sistem. Pada tahap ini akan dilakukan perancangan sistem dan perancangan *interface*. Selanjutnya akan dilakukan perancangan *interface*, perancangan ini dilakukan untuk merancang tata letak sistem sesuai dengan analisis kebutuhan sistem.
8. Pembuatan webgis rumah sakit. Pada tahap ini akan dilakukan pembuatan *web webgis* rumah sakit.
9. Peluncuran Aplikasi *appsgeysre*. Pada tahap ini akan dilakukan konvert pada *webgis* rumah sakit menjadi aplikasi android dengan menggunakan aplikasi *launcher* yaitu *appsgeysre*.
10. Penyerahan aplikasi ke-*user roll-out*. Pada tahap ini akan dilakukan penyerahan sistem aplikasi ke-*user roll-out* melalui *PlayStore*.
11. Analisa hasil dan pembahasan. Pada tahap ini menjelaskan berhasil atau tidaknya dalam suatu penelitian dengan permasalahan yang telah dideskripsikan di awal, selain menjelaskan tentang struktur aplikasi, penulis juga akan menjelaskan persamaan heversine dan membandingkan ke dalam bentuk kurva.

3.4 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan pada penelitian ini adalah sebagai berikut:

- a. Studi Literatur

Studi literatur yang digunakan yaitu buku-buku, jurnal, prosiding dan internet yang menyajikan informasi tentang aplikasi *Webgis* rumah sakit.

Arsitektur umum pemetaan aplikasi berbasis web, Android dan bahasa pemrograman *Javascript html*.

b. Metode Spasial

Metode ini digunakan untuk mendapatkan informasi pengamatan yang dipengaruhi efek ruang atau lokasi. Pengaruh efek ruang tersebut disajikan dalam bentuk koordinat lokasi (*logitude, latitude*).

3.4.1 Analisis Kebutuhan Sistem

Berdasarkan identifikasi masalah di atas, maka data yang dibutuhkan sistem adalah sebagai berikut:

- a. Peta Kota Bandar Lampung dengan informasi rumah sakit kota di Bandar Lampung.
- b. Informasi detail mengenai rumah sakit di kota Bandar Lampung.

3.4.2 Analisis User Requirement

Kebutuhan dasar aplikasi ini adalah sebagai berikut.

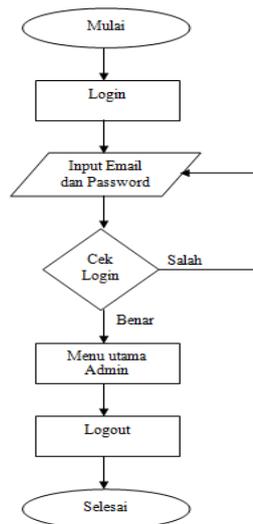
- a. Sistem dapat menampilkan peta rumah sakit di Bandar Lampung dengan menggunakan perangkat Android.
- b. Sistem dapat memberikan *step by step*, lokasi alamat rumah sakit kepada *user* untuk menuju tujuan yang *user* inginkan.
- c. Sistem dapat memberikan informasi mengenai masing-masing rumah sakit di kota Bandar Lampung.
- d. Sistem dapat merespon beberapa aksi yang diberikan oleh *user*.

3.4.3 Perancangan Sistem

Perancangan sistem adalah tahap setelah analisis dari siklus pengembangan sistem. Perancangan sistem di sini berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan dapat berfungsi. Perancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang harus diselesaikan. Tahap ini termasuk mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah dilakukan instalasi akan benar-benar sesuai dengan rancangan awal sistem. *Flowchart* merupakan diagram alur yang menggambarkan urutan logika dari suatu prosedur yang ada dalam suatu sistem. Berikut alur *Flowchart* untuk rumah salit di wilayah Bandar Lampung:

a. *Flowchart* Proses Admin

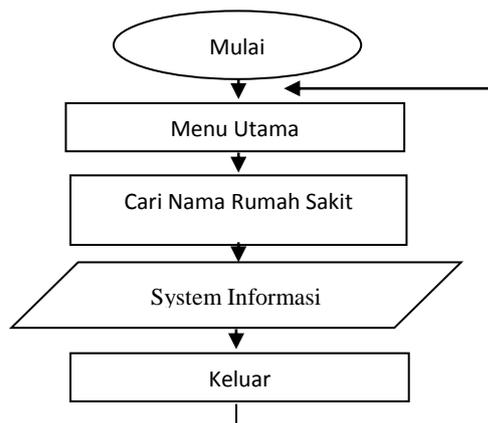
Flowchart proses *admin* merupakan gambaran diagram alur dari prosedur menu login admin sampai akses menu utama. Login untuk mengakses menu-menu yang ada di dalam program, masukkan *email* dan *password*. Email dan password akan dicek jika benar maka admin dapat mengakses menu-menu di dalam program tersebut. jika salah maka ulangi input *email* dan *password* sampai benar. Setelah proses akses selesai dilakukan *logout*. Proses admin dapat dilihat pada Gambar 3.2:



Gambar 3.2 Diagram Alir Proses Admin

b. Flowchart Proses User

Flowchart proses *user* merupakan gambaran diagram alur dari prosedur menu utama di sisi *user*. Menu utama menampilkan peta digital rumah sakit di kota Bandar Lampung. Kemudian cari nama rumah sakit untuk menampilkan informasi lengkapnya klik *link*. Proses di sisi *user* dapat dilihat pada Gambar 3.2



Gambar 3.3 Diagram Alir Proses *User*

3.5 Perancangan Antarmuka

Perancangan antarmuka merupakan proses penggambaran bagaimana sebuah tampilan antarmuka sistem dibentuk. Aplikasi *Webgis* rumah sakit dirancang dengan tampilan yang *user friendly*, sehingga diharapkan dapat mempermudah pengguna dalam menggunakan aplikasi ini. Dalam Aplikasi ini terdapat beberapa *layout* atau *form* dapat dilihat pada Gambar 3.4:



Gambar 3.4 Perancangan Antarmuka

a. *Layout* menu utama

Menu utama pada aplikasi ini menampilkan menu-menu untuk pengguna, menu tersebut adalah *Header*, Peta, Data Rumah Sakit, Tentang kami dan Kontak kami.

b. *Layout Header*

Header merupakan judul pada sebuah menu utama

c. *Layout* menu Peta

Layout menu peta terdapat di bagian awal halaman utama. Di halaman ini user dapat mencari informasi sekolah dengan mengklik simbol sekolah tersebut.

d. *Layout menu About Me.*

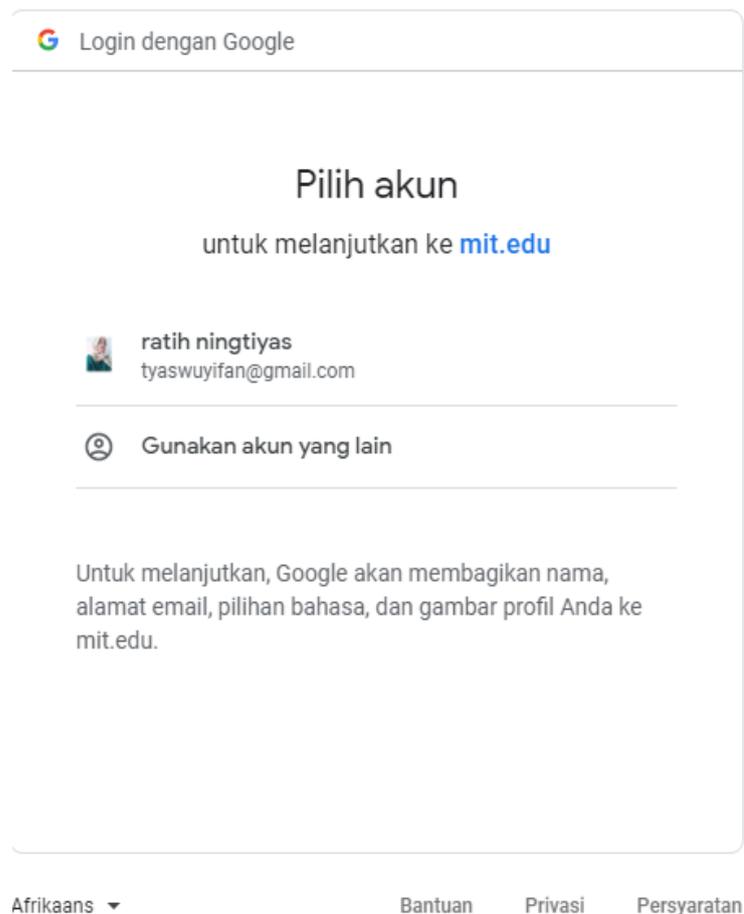
Menu Tentang Kami Aplikasi berisi tentang aplikasi *Webgis* pendidikan serta informasi pengembang.

e. *Layout menu Contact Us.*

Menu kontak Kami Aplikasi berisi tentang kontak admin.

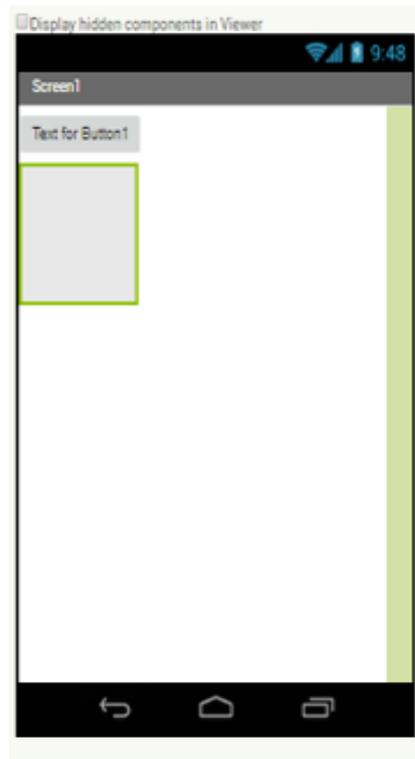
3.6 Awal Project

Langkah awal pembuatan aplikasi adalah dengan membuka web *http://ai2.AppInventor.mit.edu* memulai *browser* yang akan menampilkan halaman *sign in* pada akun gmail yang hanya bisa dibuka melalui *google chrome*.



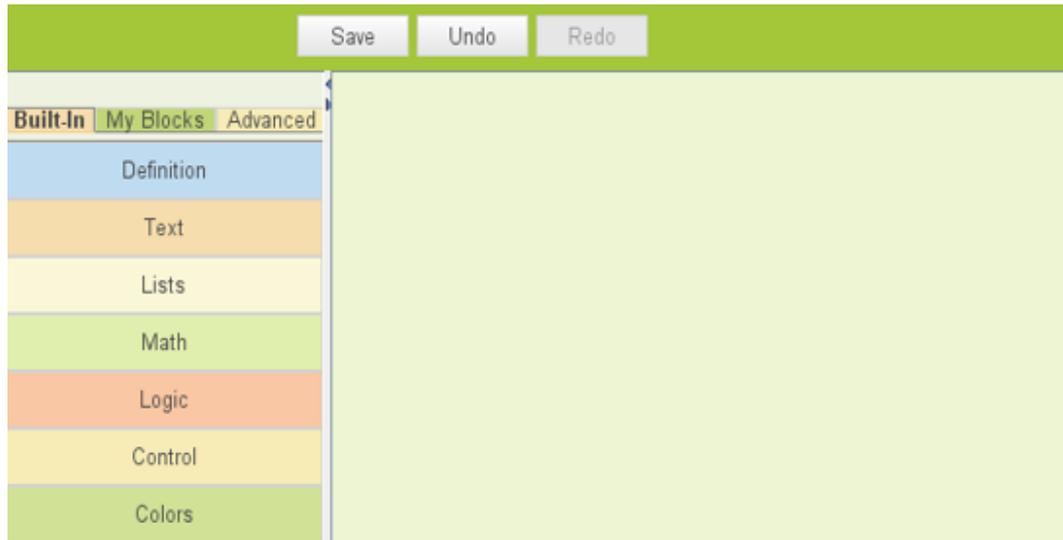
Gambar 3.5 Tampilan halaman *sign in* akun *Gmail*

Gambar 3.5 menunjukkan bahwa user harus *login email* pada *textbox* yang disediakan, lalu *sign in* dan tampilan awal halaman konfirmasi dari *App Inventor*. Setelah itu *browser* menampilkan halaman pembuka *project App Inventor*.



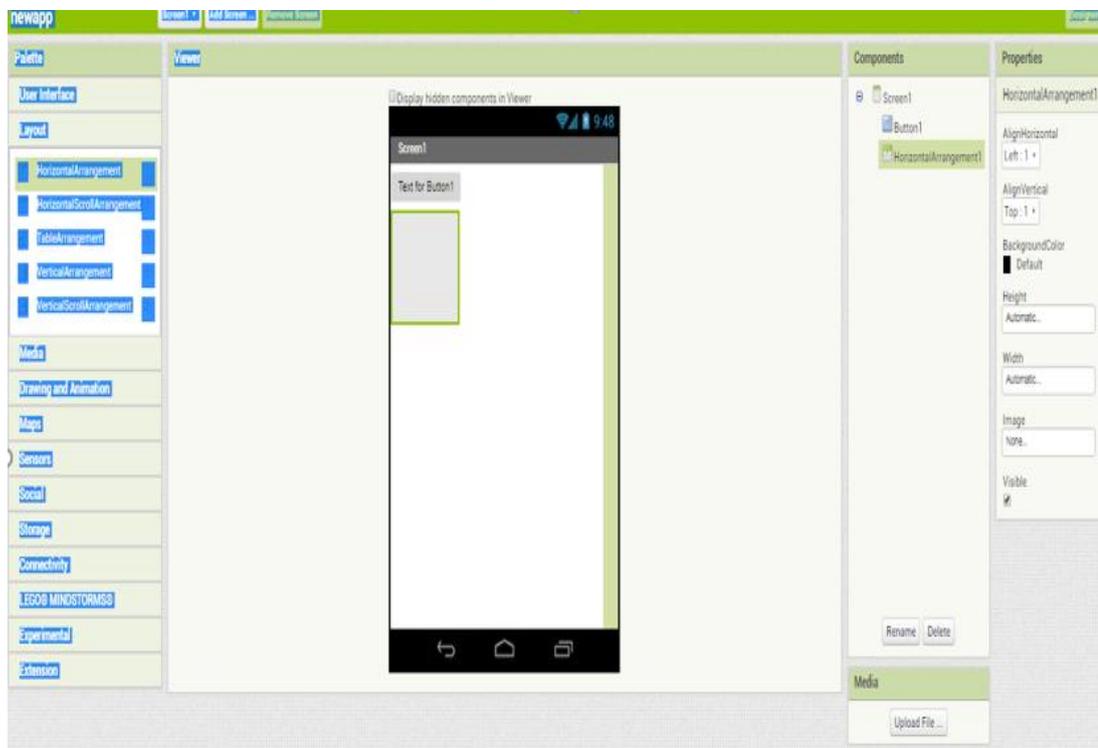
Gambar 3.6 *project App Inventor*

Pembuatan *Project* Aplikasi dengan memilih *New* terlihat pada Gambar 3.6, langkah awal memilih nama *project* aplikasi yang akan dibuat dengan klik tanda *OK*.



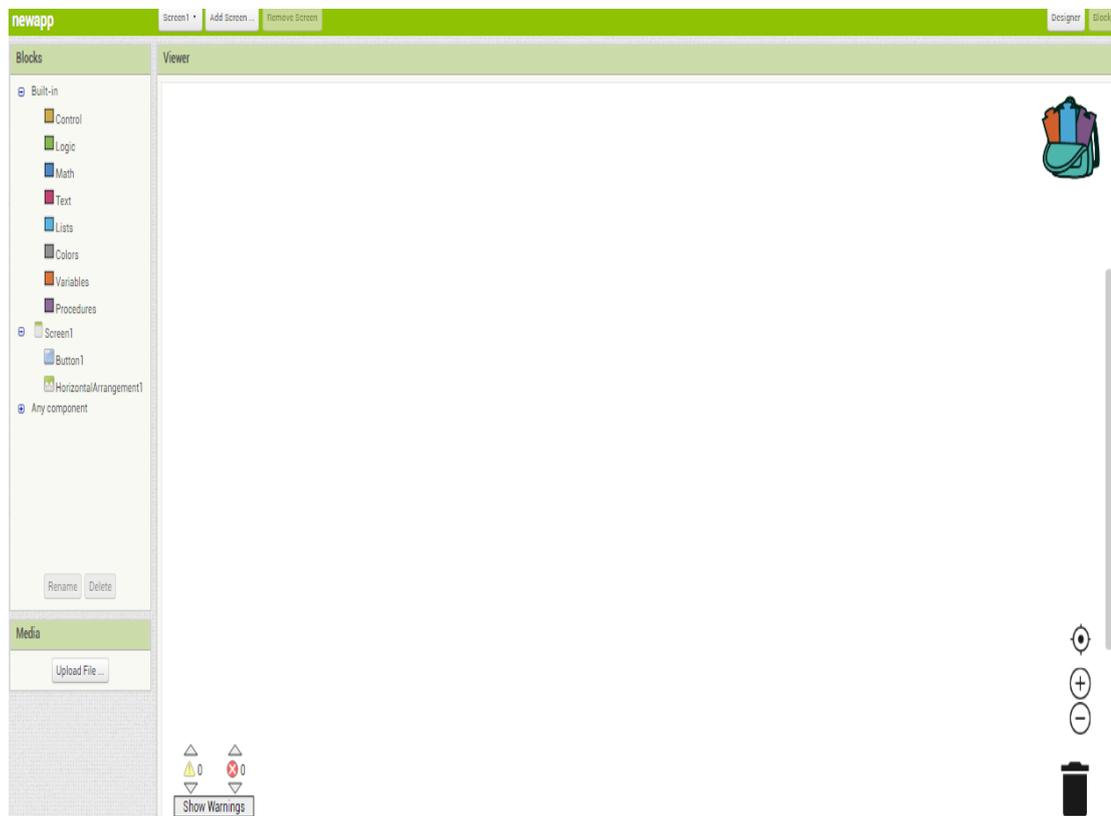
Gambar 3.7 *Project Baru di App Inventor*

Gambar 3.7 adalah *project* baru yang menampilkan halaman *design*. Pada halaman ini dapat dibuat *user interface* dengan menggunakan *drag and drop* komponen dan menyesuaikan terhadap keperluan aplikasi.



Gambar 3.8 *Design App Inventor*

Pada pembuatan pengkodean *App Inventor*, klik *Open The Block Editor* di halaman *block editor* lalu buat rancangan sistem aplikasi dengan bahasa pemrograman *visual block*.



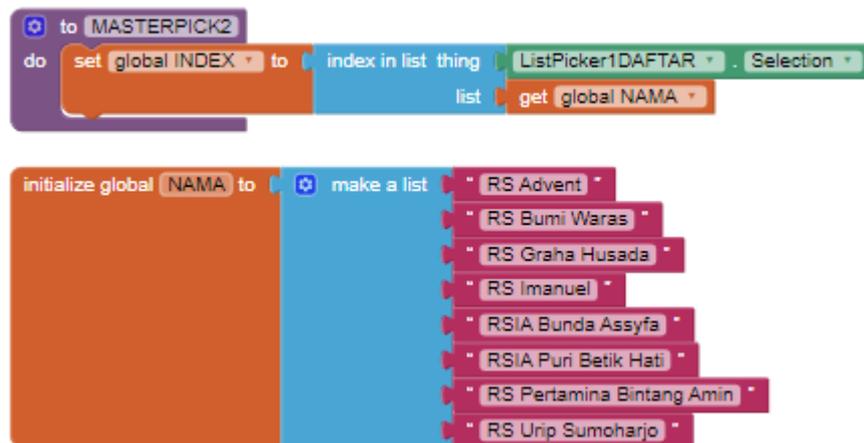
Gambar 3.9 Halaman *Block Editor App Inventor*

3.7 Pengkodean Aplikasi

Pengkodean pada aplikasi ini adalah program secara *visual block*, dengan cara beberapa *block* disesuaikan dengan kebutuhan pengguna. Data aplikasi ke sistem *user* merupakan informasi yang diinginkan tanpa menggunakan jaringan internet. Beberapa pengkodean, seperti algoritma pembacaan posisi, perhitungan jarak tempuh *user*, pembuatan data aplikasi, dan beberapa macam pengkodean lainnya.

- a. Sistem pengkodean *database button* daftar RS Pilihan.

Database pada *App Inventor* akan disimpan oleh sistem aplikasi. Data yang akan disimpan dengan cara membuat daftar dari data variabel yang saling berhubungan, yaitu *index list*. *Index list* mempunyai fungsi penghubung *list database* (terlihat pada gambar 3.10). Caranya, data harus sesuai dan berurutan dengan data yang ada di-*list database* induk.

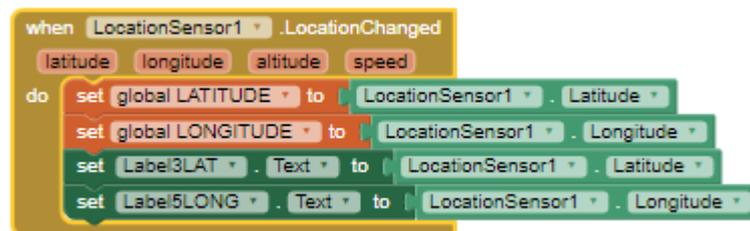


Gambar 3.10 Penyimpanan *Database* pada Aplikasi

Penyimpanan data daftar RS yang dibuat secara variabel, namanya *Initial global NAMA*, *block* variabel digabungkan dengan *block* fungsi *make a list*, *make a list* fungsinya adalah menyimpan data terbaik RS pilihan dalam bentuk *list*, dan data RS yang akan dimasukan ke dalam *list blok Object Name List* nantinya akan ditampilkan dalam bentuk daftar dalam komponen *List Picker*. Data informasi setiap sistem *item* yang akan diambil berdasarkan *index list* yang berhubungan dengan data aplikasi, contohnya *user* memilih daftar RS pilihan yang ada di Bandar Lampung. Maka, sistem aplikasi akan merespon dan menampilkan informasi mengenai RS.

b. Pengkodean pada Sensor Lokasi

Pengkodean ini menggunakan sistem pengkodean dengan membaca posisi koordinat *user* berdasarkan data GPS yang diperoleh langsung dari perangkat GPS *smartphone Android*. Pembacaan posisi kordinat bertujuan untuk navigasi aplikasi, di mana *user* akan diarahkan ke posisi RS yang sudah tersimpan dalam *database* aplikasi. sistem navigasi aplikasi RS Bandar Lampung *Map* akan langsung terintegrasi ke aplikasi *Google Maps* pada perangkat *smartphone Android*. *Google Maps* yang nantinya *user* akan mendapatkan pengarahan langsung ke lokasi RS pilihan. Pada pembuatan algoritma pembacaan di *App Inventor* menggunakan *palette* sensor yang ada pada halaman *designer App Inventor*, dengan men-*drag* komponen *Location sensor* ke dalam *Screen Project*.



Gambar 3.11 Block Pembacaan Menggunakan *Location sensor*

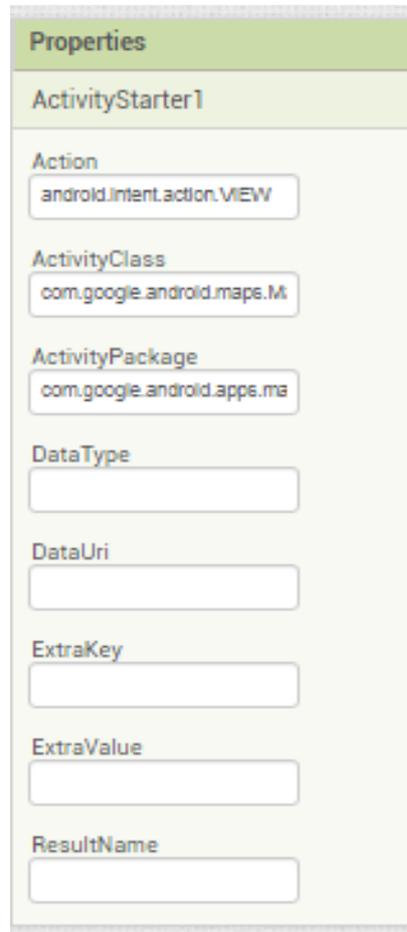
c. Pengaturan pada *Properties Maps ActivityStarter*

Pengaturan menghubungkan RS Balam *Map* ke aplikasi *Google Maps*, dilakukan menggunakan komponen *ActivityStarter* yang terdapat pada *palette other stuff* di halaman *designer* dan *drag* ke *screen project*. Selanjutnya adalah *properties Activity Starter* di halaman *designer* dengan mengisi kolom berikut:

Action: android.intent.action.VIEW

Activity Class: com.google.android.maps.MapActivity

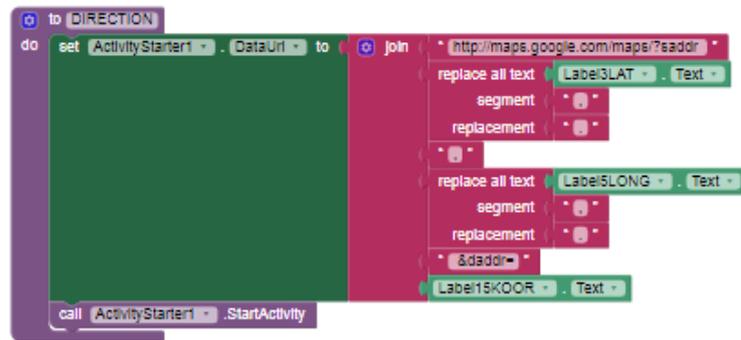
Activity Package: com.google.android.Apps.maps



Gambar 3.12 Pengaturan pada Properties *Maps Activity Starter*

d. Pengkodean *Integrasi button Direction* dengan *Google Maps*

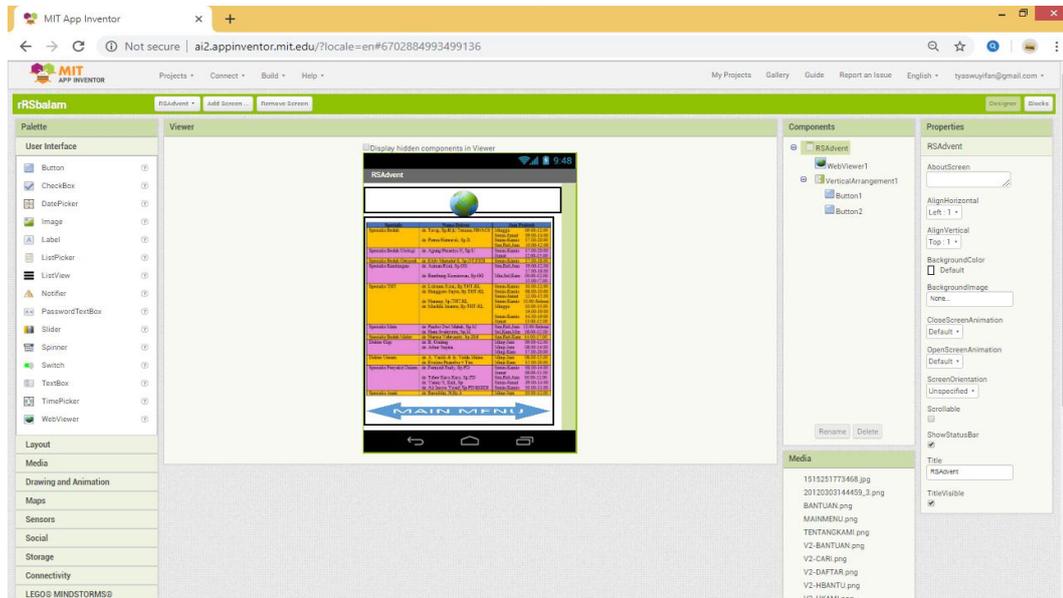
Aplikasi RS Balam *Map* menggunakan pengkodean ke *Google Maps* oleh *block editor* yang dilakukan dengan menghubungkan variabel daftar nama RS pilihan pada *Object Name List* ke variabel koordinat lokasi. Daftar RS Pilihan (aplikasi ini diberi *labelsDataURI*) berdasarkan *index list* melalui sistem *block button direction*. Jadi, maksud dari pengkodean ini adalah pengkodean navigasi ke daftar RS pilihan yang dipilih oleh *user*. Sehingga memberikan gambaran sebuah objek RS.



Gambar 3.13 Block Prosedur Direction

e. Pengkodean Integrasi *button Webview* dengan *Google Mapsengine*

Pada pengkodean RS Balam *Map* ke aplikasi *Google Mapsengine* dengan *block editor* dilakukan oleh *button Webview*, meliputi *button* nama RS pilihan dan jam praktik dokter Balam *Map*. Masud dari pengkodean ini adalah menampilkan peta ke *button-button* yang dipilih oleh *user* berupa *home url* yang didapat dari *Google Mapsengine*. Sehingga yang diharapkan dapat memberikan gambaran lokasi dari objek yang dipilih



Gambar 3.14 Pengaturan Integrasi *Button Webview*

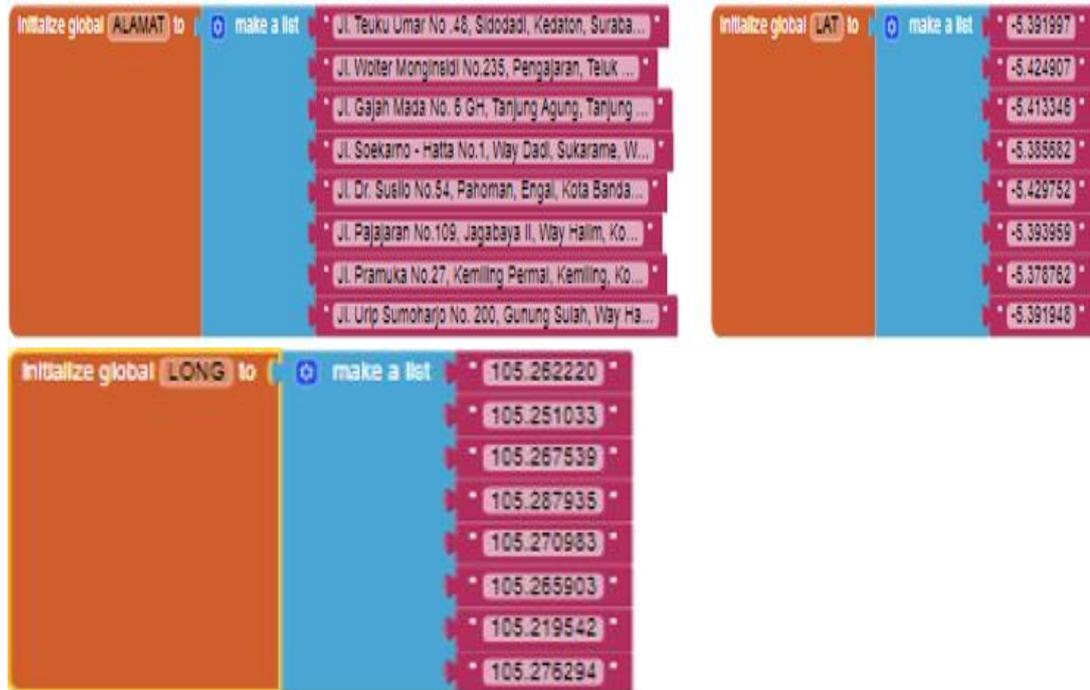
Tabel 3.2 List Daftar *Home Url* Peta Layanan Lokasi RS Bandar Lampung.

No	RS	<i>Home Url</i>
1	RS urip	https://www.google.co.id/maps/place/Rumah+Sakit+Urip+Sumoharjo+Bandar+Lampung/@-5.3912809,105.274264,17z/data=!3m1!4b1!4m5!3m4!1s0x2e411daaaaaaab:0xa5e5de495f003a31!8m2!3d-5.3912862!4d105.2764527
2	RS imanue	https://www.google.co.id/maps/place/Imanuel+Hospital+Way+Halim/@-5.3859524,105.2858064,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40db3fdc59a597:0xc9f5521502764f9!8m2!3d-5.3859577!4d105.2879951
3	RS Assyfa	https://www.google.co.id/maps/place/Ash-Shifa+Maternity+Hospital/@-5.4298322,105.2687997,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40dbcacca086c7:0xc0c1a8ab94d00634!8m2!3d-5.4298375!4d105.2709884
4	RS Graha	https://www.google.co.id/maps/place/Graha+Husada+Hospital/@-5.4134338,105.2653678,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40dbacbf59bccb:0xe8bad2df2ef23ad8!8m2!3d-5.4134391!4d105.2675565
5	RSIA PBH	https://www.google.co.id/maps/place/RSIA+Puri+Betik+Hati/@-5.3940076,105.2637139,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40dae2795e6039:0x6a28e23b02fcd8bf!8m2!3d-5.3940129!4d105.2659026
6	RS Pertamina	https://www.google.co.id/maps/place/RS+Pertamina+Bintang+Amin/@-5.3790773,105.2171061,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40da9fe253ab8d:0xa765533f947a3d51!8m2!3d-5.3790826!4d105.2192948
7	RS Bumi Waras	https://www.google.co.id/maps/place/Rumah+Sakit+Bumi+Waras/@-5.4249076,105.2487358,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40dbdc906f7dd3:0xc6621270b1b1c545!8m2!3d-5.4249129!4d105.2509245
8	RS Advent	https://www.google.co.id/maps/place/RS.+Advent+Bandar+Lampung/@-5.3920376,105.2599564,17z/data=!3m1!4b1!4m5!3m4!1s0x2e40dae3e8978fb3:0x9c508a84b76a1efd!8m2!3d-5.3920429!4d105.2621451

f. Pengkodean Informasi Teks

Pembuatan sistem *block*, memperoleh informasi deskriptif berupa teks dari sistem *database* aplikasi mengenai informasi pendukung dari setiap daftar RS pilihan, yang dilakukan oleh *button* Info. Sistem pengkodean masih dilakukan dengan cara menghubungkan variabel daftar RS ke daftar masing-masing RS lainnya berdasarkan *index list*.

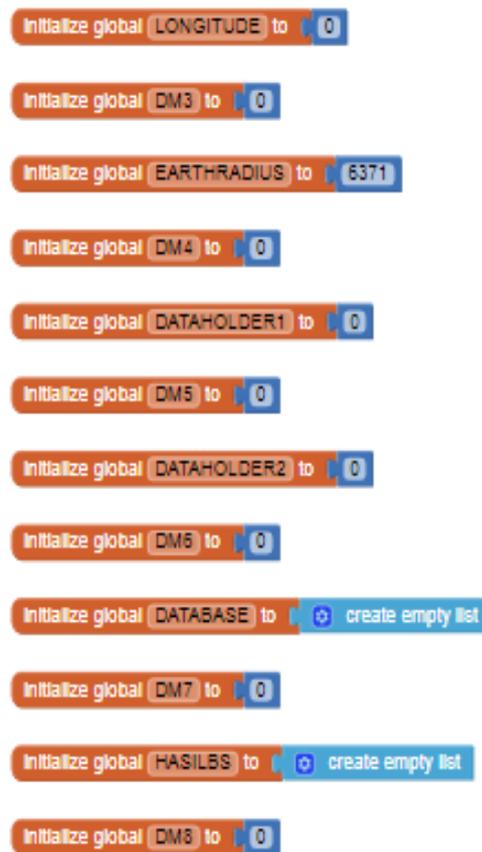




Gambar 3.15 Block dan List Database pada Button Info

g. Pengkodean *Index data*

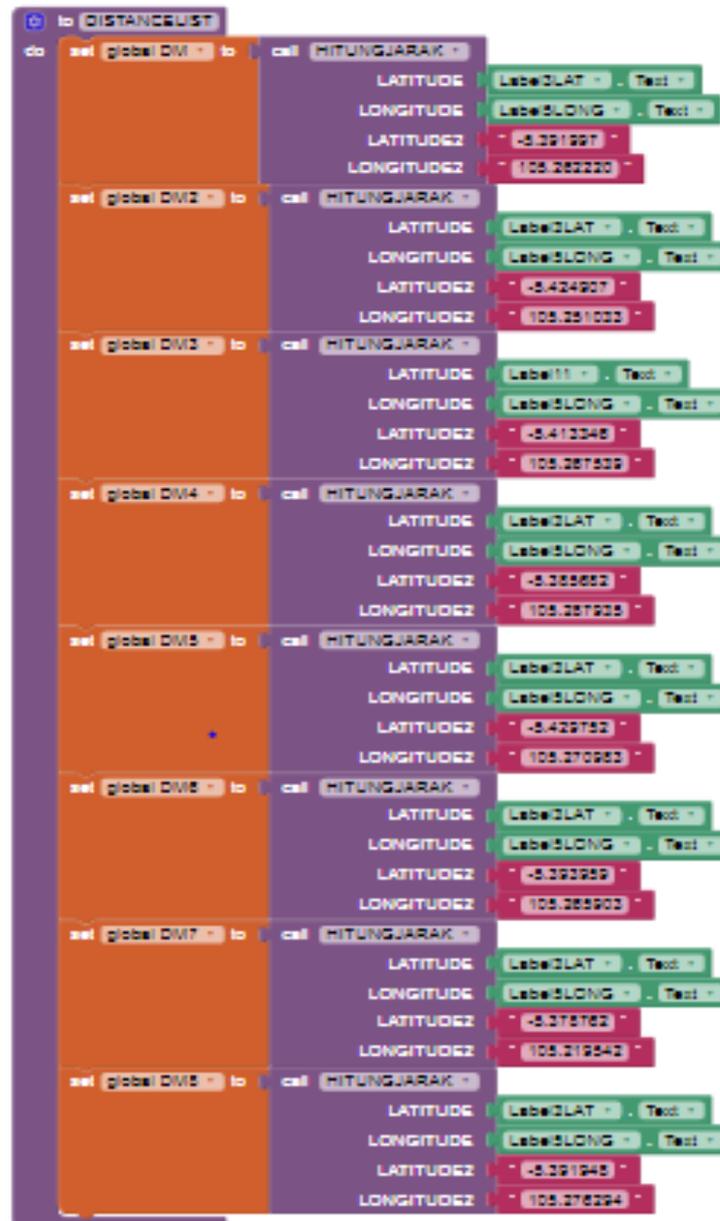
Selain pembuatan sistem pengkodean sebelumnya yang berkaitan dengan *button* Daftar RS pilihan *List*, Selanjutnya melakukan pengkodean *index data* yang akan digunakan untuk perhitungan jarak dari posisi *user* ke setiap koordinat lokasi yang terdapat pada *database*, serta memasukkan nilai dari setiap ketentuan jarak. Rancangan *Block Distance Value* dan *Index Data* dapat dilihat pada Gambar 3.16



Gambar 3.16 *Block Distance Value dan Index Data*

h. Pengkodean *Block Procedure Distance List*

Pengkodean ini berisi *database* koordinat daftar RS pilihan yang dipasang dengan *block Location sensor* dan membaca posisi *user*. Pengkodean ini bertujuan dengan jarak dalam satu variabel *procedure* yang akan dihitung dengan rumus *Haversine*.



Gambar 3.17 Block Procedure Distance List

i. Pengkodean *Haversine*

Perhitungan jarak aplikasi RS Bandar Lampung *Map* menggunakan *Haversine* adalah sebuah persamaan yang digunakan navigasi, dimana rumus tersebut memberikan jarak antara dua titik permukaan bola dari garis bujur dan garis lintang. Pada sistem *block Haversine* akan menghitung jarak dari

koordinat posisi *user* kesemua koordinat lokasi daftar RS pilihan secara satu persatu dan menyimpan secara temporal. Komponen *Lat1* didapat dari *latitude* posisi *user*, sedangkan *Long 1* didapat dari *longitude user*. Keduanya tersebut didapatkan melalui proses pembacaan posisi yang dilakukan *Location Sensor*. Komponen *Lat2* dan *Long2* diperoleh dari *database* koordinat objek RS pilihan.



Gambar 3.18 *Block Haversine*

3.9 Pengujian Aplikasi

Dibagian ini, penulis melakukan empat tahap pengujian aplikasi yaitu *download project* dan aplikasi, *instalasi* aplikasi ke *handset Android*, dan *testing blackbox*, *performance efficiency*.

3.9.1 Download Project dan Aplikasi

Project yang telah dibuat dapat langsung di *download*. *File project* dapat di *download* dalam *format* berbentuk *file Apk* yang hanya bisa diperoleh dari penulis dan penulis belum menaruh aplikasi tersebut ke dalam *play store* di *smartphone*.

3.9.2 Instalasi Aplikasi ke *Handset Android*

Tahap yang akan dilakukan adalah menginstal aplikasi ke *smartphone Android*. Aplikasi ini didapat dengan proses *download* pada *design App Inventor*.

3.9.3 *Testing Blackbox*

Fase implementasi ini adalah *testing* aplikasi, untuk mengetahui aplikasi yang dirancang apakah sesuai dengan *design* dan *activity* setiap sistem *block* yang beroperasi sesuai rancangan. Pengujian ini berdasarkan rancangan *activity* yang telah dibuat.

3.9.4 *Performance Efficiency*

Pengujian aspek *performance efficiency* aplikasi ini dilakukan dengan cara membandingkan jarak pada aplikasi yang tersedia pada *Google maps* dan dengan rumus haversine. Apakah sesuai atau mendekati sama nilai jarak tempuh user ke RS tujuan, maka di dapatkannya perbandingan.

3.5 Tabel Perhitungan Jarak

Tahap pengujian dilakukan dengan menggunakan aplikasi sepenuhnya yaitu menampilkan jarak Rumah Sakit terdekat dengan posisi *user*, pada tahap ini dilakukan perbandingan antara jarak hasil perhitungan haversine formula dengan jarak hasil perhitungan *google maps* untuk mendapatkan hasil yang efektif.

Tabel 3.3 Hasil perhitungan *Haversine* dengan perhitungan *Google maps*

Lokasi <i>User</i>	RS Terdekat	Jarak <i>Haversine</i>	Jarak <i>Google Maps</i>	Selisih
Jl. Kimaja, gg. Pertama, no. 19 kec. Kedaton	RS Urip			
	RS Imanuel			
	RS Assyfa			
	RS Graha			
	RSIA PBH			
	RS Pertamina			
	RS Bumi Waras			
	RS Advent			

3.11 Analisa Hasil dan pembahasan

Dalam tahap ini akan dilakukan analisa dan pembahasan terkait dengan rancang bangun aplikasi *mobile* RS hasil dari aplikasi tersebut akan menentukan garis lintang dan garis bujur yang akan dijelaskan dengan persamaan *Haversine* dan membandingkannya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan dianalisa maka dapat diambil kesimpulan, sebagai berikut:

1. Aplikasi *Location Based* pada program *App Inventor* yang telah dibuat menyediakan beberapa jam praktik dokter di beberapa rumah sakit terdekat melalui Android.
2. Didapatkan hasil yang berbeda disetiap rumah sakit pada rumus *Haversine* yaitu (3.5 km, 2.3 km, 8.8 km, 1.7 km, 1.8 km, 9.2 km, 7.2 km, 1.4 km), dan pada *apk app inventor* yaitu (2.2 km, 3.8 km, 6.9 km, 4.1 km, 1.1 km, 9.5 km, 8.2 km, 1.4 km), dari perbedaan keduanya didapatkan selisih (1.3, 1.5 km, 1.9 km, 2.4 km, 0.7 km, 0.3 km, 0.4 km, 0). Terlihat bahwa hasilnya lebih akurat adalah rumus *Haversine* berdasarkan jarak terdekat, terbukti dari nilai 1.7 km dan 4.1 km dengan selisih 2.4 km jarak yang ada pada aplikasi.

5.2 Saran

Adapun saran pada aplikasi ini yaitu dapat di tambah dengan beberapa fitur lainnya, serta pencarian objek lainnya. Pengguna metode lain yang lebih efektif juga bisa di teliti untuk mendapatkan hasil yang lebih baik.

DAFTAR PUSTAKA

- [1] Istiqomah Sumadikarta, "Perancangan Aplikasi *Location Based Service* Pencarian Rumah Sakit Bermitra dengan BPJS Kesehatan Barbasis Android", *Jurnal Satya Informatika*. Vol 2 No 3, September 2017.
- [2]. Ika Rahayu Wulansari, "Pembuatan Aplikasi Swbaran Lokasi Fasilitas Kesehatan Penerimaan BPJS Kesehatan di Kota Semarang Berbasis Android", *Vol 4 No 4*. (ISSN, 2337-845X), Oktober 2017.
- [3]. Whelly Yulianto, "Menentukan Jarak Hotel dengan Metode *Haversine* Formula", Juni 2017.
- [4]. Imran Djafar, "Pembangunan Aplikasi *Location Based Service* Kota Makassar", *Jurnal Informatika*. (ISSN, 2302-3805), Febuari 2016.
- [5]. Dwi Maryono Endar Suprih Wihidayat, "Pengembangan Aplikasi Android Menggunakan *Integrated Development Environment* (Ide) App Inventor 2," *Jurnal Ilmiah Edutic*, vol. 4, no. 1, November 2017.
- [6]. Edy Budiman, "Pemanfaatan Teknologi *Location Based Service* dalam Pengembangan Aplikasi Profil Kampus Universitas Mulawarman Berbasis Mobile," *Jurnal Ilmiah ILKOM*, vol. 8,3 no. , Desember 2016.
- [7]. Martza Merry Swastikasari, "Rancangan Bangun Aplikasi Mobile E-Kost Menggunakan *Location Based Service* Berbasis Android", *Journal of Computer Science and Information Systemt*, vol. 2, no. 2, Oktober 2018.
- [8]. Fitri Marisa, "Aplikasi Pencarian Pariwisata dan Tempat Oleh-Oleh Terdekat Menggunakan Metode *Haversine* Berbasis Android," *Jurnal Informatika Merdeka Pasuruan*, vol. 3, no. 2, Agustus 2016.

- [9]. Prya Adhi Surya Nugraha, "Pembuatan Aplikasi sebagai Lokasi Wisata Kuliner di Kota Surakarta Berbasis Android," *Jurnal Geodesi Undip*, vol. 6, no. 4, Oktober 2017.
- [10]. Khoirul Umam, "Aplikasi Webgis Pencarian Lokasi Properti Terdekat dengan Metode *Haversine* Formula pada Media Pemasaran Online Agen Properti LJ Hooker Jogja Utara," November 2017.
- [11]. Ratih Kartika Dewi, "Sistem Informasi Menggunakan Teknologi *Location Based Service*" *Jurnal Pengembangan teknologi Informatika dan Ilmu Komputer*, vol. 2, no. 6, 31 Agustus 2017.
- [12]. Agi Priawan D.S Ramdan, "Penerapan *Location Based Service* dan QR-Code dalam Pemetan Lokasi Berbasis Android," *Jurnal Manajemen teknologi Informatika dan Ilmu Komputer*, vol. 1, no. 1, Februari 2017.
- [13]. Muhammad Muslim, "*Mobile APP Menggunakan Location Based Service & Metode Haversine* dalam Pengimplementasian Jarak & Medan", Maret 2018.
- [14]. Sariyun Naja Anwar, "Model Rute dan Peta Interaktif Posyandu di Kota Semarang menggunakan *Geolocation* dan *Haversine* Berbasis Mobile Android" *Jurnal TI DINAMIK*, vol. 20, no. 1, January 2018.
- [15]. Andry Mulyawan Nurul Chafid, "Aplikasi *Location Based Service Wedding Organizer* di Kota Tangerang Selatan Berbasis Android" *Jurnal Satya Informatika*, vol. 2, no. 1, Mei 2017.