

APLIKASI PENGUKURAN LUAS BIDANG PERMUKAAN BUMI
BERBASIS ANDROID

(Skripsi)

Oleh
RUDRA NUGRAHA



JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2018

ABSTRAK

APLIKASI PENGUKURAN LUAS BIDANG PERMUKAAN BUMI BERBASIS ANDROID

Oleh

RUDRA NUGRAHA

Lahan merupakan kebutuhan primer manusia. Lahan digunakan manusia untuk kebutuhan tempat tinggal ataupun bisnis. Luas lahan yang besar serta bentuknya yang tidak beraturan menjadi kendala dalam pendataan lahan. Dengan permasalahan tersebut dibuatlah aplikasi yang dapat mempermudah pendataan tersebut. Aplikasi ini dibuat berbasis android menggunakan bahasa pemrograman Java. Metode perhitungan yang digunakan pada sistem adalah *shoelace formula* dan metode pengembangan sistem yang digunakan dalam aplikasi ini adalah *Waterfall* serta desain menggunakan *Unified Modeling Language (UML)*. Aplikasi dibuat untuk dapat melakukan perhitungan luas dengan bentuk yang beraturan ataupun tidak beraturan. Aplikasi ini memiliki batasan yaitu tidak bisa melakukan perhitungan pada bidang lahan yang miring, hal ini merupakan salah satu kelemahan *shoelace formula*. Aplikasi ini diuji menjadi 3 tahap yaitu pengujian fungsi, akurasi, serta perbandingan aplikasi. Dalam pengujian fungsi aplikasi dapat berfungsi dengan baik, dan dalam pengujian akurasi aplikasi dapat berfungsi dengan baik dengan selisih luas rata-rata 2%, serta dalam pengujian perbandingan aplikasi pun aplikasi dapat berfungsi dengan baik dengan selisih luas rata-rata 0.16%.

**APLIKASI PENGUKURAN LUAS BIDANG PERMUKAAN BUMI
BERBASIS ANDROID**

Oleh

**RUDRA NUGRAHA
1117032056**

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA KOMPUTER

Pada

Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam



**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2018**

Judul Skripsi : **APLIKASI PENGUKURAN LUAS BIDANG
PERMUKAAN BUMI BERBASIS ANDROID**

Nama Mahasiswa : **Rudra Nugraha**

Nomor Pokok Mahasiswa : 1117032056

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam

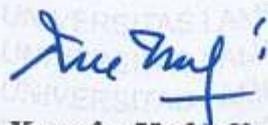
MENYETUJUI

1. Komisi Pembimbing



Febi Eka Febriansyah, M.T.
NIP 19800219 200604 1 001

2. Ketua Jurusan Ilmu Komputer



Dr. Ir. Kurnia Muludi, M.S.Sc.
NIP 19640616 198902 1 001

MENGESAHKAN

1. Tim Penguji

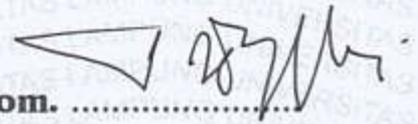
Ketua Penguji : **Febi Eka Febriansyah, M.T.**



Penguji Pembahas : **Dr. Ir. Kurnia Muludi, M.S.Sc.**



Penguji Pembahas : **Drs. Rd. Irwan Adi Pribadi, M.Kom.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Prof. Warsito, S.Si, D.E.A., Ph.D.

NIP 19710212 199512 1 001

Tanggal Lulus Ujian Skripsi : **29 November 2018**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Aplikasi Pengukuran Luas Bidang Permukaan Bumi Berbasis Android”** merupakan karya saya sendiri dan bukan hasil karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah Penulisan Karya Ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 29 November 2018



Rudra Nugraha
NPM. 1117032056

RIWAYAT HIDUP

Penulis dilahirkan di Tanjung Karang, pada tanggal 17 April 1993. Penulis adalah anak Tunggal pasangan Bapak Rudi Joko dan Ibu Neni Rizki Agustiyani. Pendidikan formal yang pernah penulis tempuh Sekolah Dasar (SD) di SD Negeri 1 Semuli Raya, Lampung Utara hingga kelas 5. Kemudian pindah pada tahun 2004 ke SD Negeri 1 Sukamaju, Subang, Jawa Barat dan diselesaikan pada tahun 2005. Sekolah Menengah Pertama (SMP) di SMP Negeri 1 Subang, Jawa Barat diselesaikan pada tahun 2008. Sekolah Menengah Atas (SMA) di SMA Negeri 1 Jalan Cagak, Subang, Jawa Barat diselesaikan pada tahun 2011.

Kemudian pada tahun 2011, penulis menjadi mahasiswa Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung. Selama di Universitas Lampung penulis mengikuti UKM yaitu Himakom. Lalu pada bulan Januari, 2014 penulis melakukan Kuliah Kerja Nyata (KKN) di Kecamatan Dente Teladas, Tulang Bawang. Kemudian pada bulan Januari 2015, penulis melakukan Kerja Praktek di CV. Cipta Teknologi Muda.

PERSEMBAHAN

*Puji syukur kehadirat Allah SWT
dengan ketulusan serta kerendahan hati kupersembahkan
karya ini sebagai tanda baktiku untuk:*

*Teristimewa kedua orang tuaku, papa Rudi Joko dan
mama Neni Rizki Agustiani tercinta yang selalu
memberikan do'a dan harapan di setiap tetes keringatmu
demi tercapainya kesuksesanku. Yang sangat aku sayangi
dan aku hormati yang selalu menanti keberhasilanku.*

Keluarga Keluarga Ilmu Komputer 2011,

Keluarga besar HIMAKOM

*yang telah mengajarkan saya pelajaran hidup yang sangat
berharga terimakasih atas pengalamannya.*

Serta Almamater Tercinta,

Universitas Lampung.

MOTO

*“Waktu bukanlah alasan untuk memulai berkarya
dan waktu bukanlah alasan untuk berhenti
berkarya”*

*“Setiap segala sesuatu itu ada kelebihanannya. Maka
janganlah suka meremehkan dan merendahkan.”*

(Ustadz Jefri Al Buchori)

*“Dan janganlah kamu berputus asa dari rahmat
Allah. Sesungguhnya tiada berputus asa dari
rahmat Allah melainkan orang-orang yang kufur
(terhadap karunia Allah)”*

(Q.S. Yusuf Ayat 87)

SANWACANA

Puji syukur penulis panjatkan kehadirat Allah *Subhanahu Wa Ta'ala* atas berkat rahmat, hidayah, dan kesehatan yang diberikan sehingga penulis dapat menyelesaikan penulisan skripsi ini.

Skripsi ini disusun sebagai syarat untuk memperoleh gelar Sarjana Komputer di Jurusan Ilmu Komputer Universitas Lampung. Judul dari skripsi ini adalah “Aplikasi Pengukuran Luas Bidang Permukaan Bumi Berbasis Android”.

Dalam penyusunan skripsi ini, penulis banyak menghadapi kesulitan. Namun, berkat bantuan dan dorongan dari berbagai pihak, akhirnya penulis dapat menyelesaikan skripsi ini. Untuk itu pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Kedua orang tua tercinta Rudi Joko dan Neni Rizki Agustiani yang telah memberikan doa, dukungan, dan semangat serta memfasilitasi kebutuhan untuk menyelesaikan skripsi ini.
2. Bapak Febi Eka Febriansyah, M.T., sebagai pembimbing penulis yang telah memberikan ide dan masukan dalam pengerjaan skripsi serta memberikan dorongan sehingga skripsi ini dapat diselesaikan.
3. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., sebagai pembahas yang telah memberikan masukan-masukan dan saran yang bermanfaat dalam skripsi ini.

4. Bapak Drs. Rd. Irwan Adi Pribadi, M.Kom., sebagai pembahas yang telah memberikan masukan-masukan dan saran yang bermanfaat dalam skripsi ini.
5. Bapak Dwi Sakheti, M.Kom., selaku dosen pembimbing akademik penulis yang selalu memberikan motivasi, saran, dan arahan dalam menjalani perkuliahan dan menyelesaikan kuliah di Universitas Lampung.
6. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
7. Bapak Didik Kurniawan, S.Si., M.T., selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
8. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu pengetahuan yang bermanfaat bagi penulis.
9. Ade Nora Maela, selaku staf administrasi di Jurusan Ilmu Komputer yang telah membantu segala urusan administrasi selama kuliah.
10. Teman-teman seperjuangan Basir, Fajri, Dimas, Gamma, Adi, Bayu, Okky, Budiman, Rian, Ardhika, Panji, Galih, Tryo, Bobby, Rifki serta seluruh teman-teman Ilkom 2011 yang tidak bisa disebutkan satu persatu. Terimakasih telah memberikan penulis inspirasi dan keceriaan selama perkuliahan.

Akhir kata, semoga Allah SWT selalu memberikan rahmat dan hidayah-Nya kepada kita semua dan semoga skripsi ini bermanfaat bagi kita semua. Amin.

Bandar Lampung, 29 November 2018

Rudra Nugraha

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN	iv
RIWAYAT HIDUP	v
PERSEMBAHAN.....	vi
MOTTO	vii
SANWACANA	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xv
BAB I. PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Manfaat	3
BAB II. TINJAUAN PUSTAKA.....	4
2.1 Sistem Informasi Geografi (SIG).....	4
2.2 <i>Latitude dan Longitude</i>	4
2.3 <i>Shoelace Formula</i>	6

2.4 Penggunaan Koordinat <i>Latitude</i> dan <i>Longitude</i> pada <i>Shoelace Formula</i> ...	9
2.5 Android	12
2.5.1 Android Studio.....	13
2.5.2 Fundamental Aplikasi	14
2.5.3 Versi Android.....	16
2.6 Google Map API	17
2.6.1 Pengertian API.....	17
2.6.2 Pengertian Google Map API.....	17
2.6.3 Google Maps Android API <i>Utility Library</i>	18
2.7 Metode Pengembangan Sistem: Metode <i>Waterfall</i>	18
2.8 Teknik Pengujian Perangkat Lunak	19
2.9 Partisi Ekuivalensi	20
2.10 <i>Unified Modeling Language</i> (UML).....	21
BAB III. METODOLOGI PENELITIAN	26
3.1 Waktu dan Tempat Penelitian	26
3.2 Metodologi Penelitian	26
3.2.1 Alir Penelitian.....	26
3.2.2 Metode Pengembangan Sistem.....	30
3.2.3 Metode Pengumpulan Data	33
3.3 Perancangan Sistem	33
3.3.1 Perancangan UML (Unified Modelling Language).....	34
3.3.2 Perancangan Antarmuka.....	40
3.4 Metode Pengujian Sistem	48
BAB IV. HASIL DAN PEMBAHASAN	51
4.1 Implementasi.....	51
4.2 <i>Layout</i> Aplikasi Pengukuran Luas Bidang Permukaan Bumi	52
4.2.1 <i>Layout</i> Menu Syarat dan Ketentuan	52
4.2.2 <i>Layout</i> Menu Utama.....	52
4.2.3 <i>Layout</i> Menu Hitung Luas Area	53

4.2.4 <i>Layout</i> Menu Bantuan	56
4.2.5 <i>Layout</i> Menu Tentang	56
4.3 Pengujian Fungsional	57
4.3.1 Pengujian Versi Android	58
4.3.2 Pengujian Ukuran Layar	59
4.3.3 Pengujian Fungsi dari Menu Aplikasi.....	60
4.4 Pengujian Akurasi	61
4.5 Pengujian Perbandingan Aplikasi	63
BAB V. SIMPULAN DAN SARAN	66
5.1 Simpulan	66
5.2 Saran	66
DAFTAR PUSTAKA	67

DAFTAR GAMBAR

Gambar	Halaman
2.1 <i>Latitude</i> dan <i>Longitude</i>	5
2.2 Diagram Poligon P	7
2.3 Diagram Trapesium T	8
2.4 Konversi Satuan <i>Latitude</i> Kedalam Satuan Meter	10
2.5 Radius Pada Suatu <i>Latitude</i>	11
2.6 Metode Pengembangan Sistem <i>Waterfall</i> (Kadir, 2002)	19
2.7 <i>Use Case</i> Diagram (Fowler, 2004).....	23
2.8 Diagram Aktivitas (Fowler, 2004)	24
2.9 Diagram Sekuensial (Fowler, 2004)	25
3.1 Diagram Alir Metodologi Penelitian.....	27
3.2 <i>Use Case</i> Diagram.....	34
3.3 <i>Activity</i> Diagram Perhitungan Luas Lokasi Ini	35
3.4 <i>Activity</i> Diagram Perhitungan Luas Lokasi Lain	36
3.5 <i>Activity</i> Diagram Bantuan	37
3.6 <i>Activity</i> Diagram Tentang	37
3.7 <i>Sequence</i> Diagram Perhitungan Luas Lokasi Ini	38
3.8 <i>Sequence</i> Diagram Perhitungan Luas Lokasi Lain	39
3.9 <i>Sequence</i> Diagram Bantuan	39
3.10 <i>Sequence</i> Diagram Tentang.....	40
3.11 <i>Layout</i> Menu Syarat dan Ketentuan.....	41
3.12 <i>Layout</i> Menu Utama.....	41
3.13 <i>Layout</i> Menu Hitung Luas Area.....	42

3.14	<i>Layout</i> Menu Hitung Luas Area Lokasi Ini	42
3.15	<i>Layout</i> Menu Hitung Luas Area Ini Ketika Pengguna Menempatkan Koordinat Titik Simpul	43
3.16	<i>Layout</i> Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol Checklist	44
3.17	<i>Layout</i> Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol Hapus	44
3.18	<i>Layout</i> Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol Hitung	45
3.19	<i>Layout</i> Menu Hitung Luas Area Lokasi Lain	46
3.20	<i>Layout</i> Menu Hitung Luas Area Lain Ketika Pengguna Menempatkan Koordinat Titik Simpul	46
3.21	<i>Layout</i> Menu Bantuan	47
3.22	<i>Layout</i> Menu Tentang	47
4.1	<i>Layout</i> Menu Syarat dan Ketentuan	52
4.2	<i>Layout</i> Menu Utama	53
4.3	<i>Layout</i> Menu Hitung Luas Area	53
4.4	<i>Layout</i> Menu Hitung Luas Area Lokasi Ini dan Penggunaannya	54
4.5	<i>Layout</i> Menu Hitung Luas Area Lokasi Lain dan Penggunaannya	55
4.6	<i>Layout</i> Menu Bantuan	56
4.7	<i>Layout</i> Menu Tentang	57

DAFTAR TABEL

Tabel	Halaman
2.1 Tabel Iterasi Perhitungan Luas Poligon P	7
2.2 Tabel Iterasi Perhitungan Luas Trapesium T	8
3.1 Daftar Pengujian <i>Equivalence Partiotioning</i> (EP)	48
4.1 Pengujian Versi Android	58
4.2 Pengujian Ukuran Layar	59
4.3 Pengujian Fungsi dan Menu Aplikasi.....	60
4.4 Data Pengujian Akurasi	61
4.5 Tabel Iterasi Perhitungan Akurasi	62
4.6 Data Pengujian Perbandingan Aplikasi	63
4.7 Tabel Iterasi Perhitungan Akurasi Luas <i>Area Measurement</i> Terhadap Luas <i>Measure Map Lite</i>	64
4.8 Tabel Iterasi Perhitungan Akurasi Luas <i>Area Measurement</i> Terhadap Luas <i>Field Area Measure</i>	65

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Lahan merupakan kebutuhan primer manusia. Lahan digunakan manusia untuk kebutuhan tempat tinggal ataupun bisnis. Pendataan lahan mengenai luas lahan serta bentuk lahan saat ini sudah dilakukan. Perhitungan luas lahan tidak hanya dilakukan untuk individu, melainkan untuk wilayah juga seperti Desa, Kecamatan, Kabupaten, Provinsi, atau bahkan Negara. Luas lahan yang besar serta bentuknya yang tidak beraturan menjadi kendala dalam perhitungan luas tersebut.

Di sisi lain perkembangan teknologi berkembang dengan sangat pesat. Salah satunya dari perkembangan ponsel yang semakin canggih. Ponsel yang dulu hanya dapat digunakan untuk komunikasi saja seperti telepon dan SMS (*Short Message Service*) sekarang sudah menjadi komputer yang canggih. Fitur dan spesifikasi ponsel saat ini bahkan dapat menyaingi *laptop*. Hal ini membuat penggunaan ponsel ini menjadi serbaguna.

Dilatarbelakangi hal tersebut, peneliti tertarik untuk membuat aplikasi *smartphone* yang bisa digunakan untuk menghitung luas lahan suatu tempat.

1.2 Rumusan Masalah

Berdasarkan apa yang telah dijelaskan dalam latar belakang, maka didapatkan rumusan masalah tentang bagaimana cara membuat aplikasi *smartphone* yang dapat melakukan perhitungan luas lahan yang bentuknya beraturan atau pun yang tidak beraturan.

1.3 Batasan Masalah

Batasan masalah dalam perancangan dan pembuatan aplikasi ini adalah:

1. Aplikasi dapat melakukan perhitungan luas lahan pada bidang yang datar dengan bentuk yang beraturan atau pun tidak beraturan dengan memasukan minimal tiga titik simpul.
2. Aplikasi tidak dapat melakukan perhitungan luas lahan pada bidang yang miring.

1.4 Tujuan

Tujuan dari perancangan dan pembuatan aplikasi ini adalah untuk mempermudah dalam melakukan perhitungan luas lahan yang bentuknya beraturan atau pun yang tidak beraturan.

1.5 Manfaat

Manfaat penelitian dalam pembuatan sistem aplikasi ini adalah:

1. Hasil penelitian dapat mempermudah melakukan perhitungan luas lahan.
2. Hasil penelitian dapat menambah pengetahuan mengenai bagaimana cara menghitung luas lahan dengan bentuk yang tidak beraturan bagi penulis serta pembaca.
3. Hasil penelitian dapat dijadikan acuan untuk pengembangan atau pembuatan aplikasi dalam penelitian yang sama.

BAB II TINJAUAN PUSTAKA

2.1 Sistem Informasi Geografi (SIG)

Sistem Informasi Geografi (SIG) merupakan ilmu pengetahuan yang berbasis pada perangkat lunak komputer yang digunakan untuk memberikan bentuk digital dan analisa terhadap permukaan geografi bumi sehingga membentuk suatu informasi keruangan yang tepat dan akurat (Suryantoro, 2009).

2.2 *Latitude dan Longitude*

2.2.1 Latitude

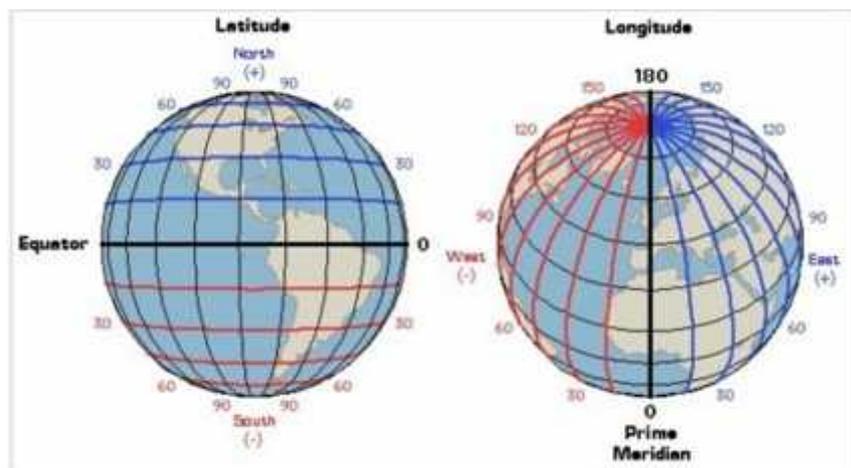
Latitude (lintang) merupakan garis yang melintang di antara Kutub Utara dan Kutub Selatan, yang menghubungkan antara sisi timur dan sisi barat bagian bumi. Garis ini memiliki posisi membentangi bumi, sama halnya seperti garis *equator* (khatulistiwa), tetapi dengan kondisi nilai tertentu. Garis lintang inilah yang dijadikan ukuran dalam mengukur sisi utara-selatan koordinat suatu titik di bumi (Suryantoro, 2009).

2.2.2 Longitude

Longitude (bujur) merupakan garis membujur yang menghubungkan antara sisi utara dan sisi selatan bumi. Garis bujur ini digunakan untuk mengukur sisi barat-timur koordinat suatu titik di bumi (Suryantoro, 2009).

2.2.3 Satuan *Longitude* dan *Latitude*

Longitude dan *latitude* menggunakan satuan derajat. *Longitude* sendiri memiliki pusat di Greenwich yang disebut *prime meridian* dimana *longitude* memiliki nilai 0° , nilainya akan semakin bertambah hingga 180° kearah barat dan 180° kearah timur. Sedangkan *latitude* memiliki pusat di *equator* dimana *latitude* memiliki nilai 0° , nilainya akan semakin bertambah hingga 90° ke arah utara dan berakhir di Kutub Utara serta 90° ke arah selatan dan berakhir di Kutub Selatan. Untuk lebih jelasnya dapat dilihat pada Gambar 2.1.



Gambar 2.1 *Latitude* dan *Longitude*

2.3 Shoelace Formula

Shoelace Formula atau formula tali sepatu (juga dikenal sebagai formula *Gaus's area formula* dan *Surveyor's formula*) adalah algoritma matematis untuk menentukan area poligon sederhana yang titik simpulnya digambarkan oleh koordinat *Cartesian*.

Rumus ini dijelaskan oleh Meister (1724-1788) pada tahun 1769 dan oleh Gauss pada tahun 1795. Rumus ini dapat diverifikasi dengan membagi poligon menjadi segitiga. *Shoelace Formula* dapat dituliskan sebagai berikut (Braden, 1986):

$$A = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right|$$

$$= \frac{1}{2} |x_1 y_2 + x_2 y_3 + \dots + x_{n-1} y_n + x_n y_1 - x_2 y_1 - x_3 y_2 - \dots - x_n y_{n-1} - x_1 y_n|$$

Dengan : A = Luas poligon

n = Banyaknya sudut poligon

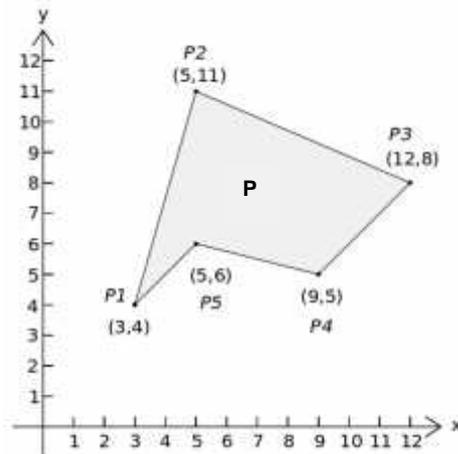
i = Sudut ke- i poligon

x_i = Nilai x pada sudut ke- i poligon

y_i = Nilai y pada sudut ke- i poligon

Sebagai contoh:

Digambarkan poligon P dengan titik simpul dengan nilai sebagai berikut (3,4), (5,11), (12,8), (9,5), dan (5,6), ilustrasi diagram dapat dilihat pada Gambar 2.2.



Gambar 2.2 Diagram Poligon P

Dengan menggunakan *Shoelace Formula* kita dapat melakukan perhitungan sebagai berikut:

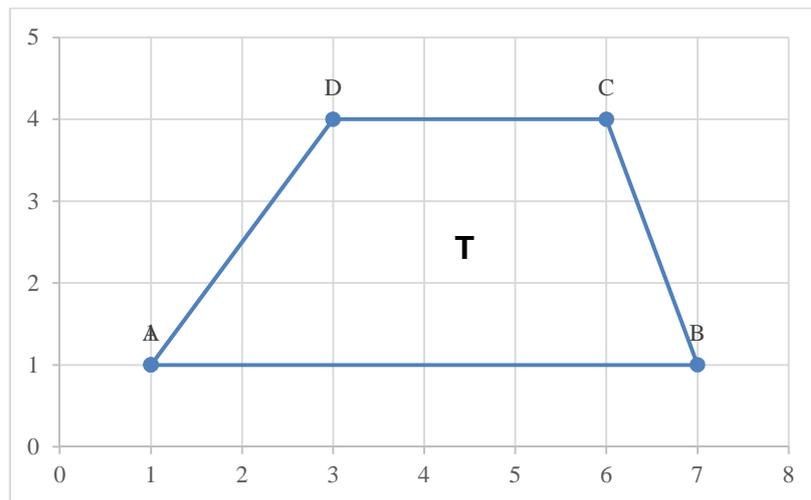
Tabel 2.1 Tabel Iterasi Perhitungan Luas Poligon P

Iterasi	x^i	y^i	$x^i y_{i+1}$	$\sum_{i=1}^{n-1} x_i y_{i+1}$	$x^n y_1$	$x_{i+1} y_i$	$\sum_{i=1}^{n-1} x_{i+1} y_i$	$x_1 y_n$
1	3	4	33	33	20	20	20	18
2	5	11	40	73	20	132	152	18
3	12	8	60	133	20	72	224	18
4	9	5	54	187	20	25	249	18
5	5	6			20			18

$$\begin{aligned}
 A &= \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| \\
 &= \frac{1}{2} |187 + 20 - 249 - 18| \\
 &= \frac{1}{2} |-60| \\
 &= 30
 \end{aligned}$$

Contoh lain:

Digambarkan trapesium T dengan titik simpul dengan nilai sebagai berikut (1,1), (7,1), (6,4), dan (3,4) dengan ilustrasi diagram yang dapat dilihat pada Gambar 2.3.



Gambar 2.3 Diagram Trapesium T

Dengan menggunakan *Shoelace Formula* kita dapat melakukan perhitungan sebagai berikut:

Tabel 2.2 Tabel Iterasi Perhitungan Luas Trapesium T

i	x_i	y_i	$x_i y_{i+1}$	$\sum_{i=1}^{n-1} x_i y_{i+1}$	$x_n y_1$	$x_{i+1} y_i$	$\sum_{i=1}^{n-1} x_{i+1} y_i$	$x_1 y_n$
1	1	1	1	1	3	7	7	4
2	7	1	28	29	3	6	13	4
3	6	4	24	53	3	12	25	4
4	3	4			3			4

$$A = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right|$$

$$\begin{aligned}
 &= \frac{1}{2} |53 + 3 - 25 - 4| \\
 &= \frac{1}{2} |27| \\
 &= 13,5
 \end{aligned}$$

Hasil ini dapat dibuktikan dengan menggunakan rumus dasar trapesium dengan persamaan yang dapat dituliskan sebagai berikut:

$$L = \frac{(a+b)t}{2}$$

Dengan : L = Luas trapesium

a = Panjang sisi yang sejajar dengan AB

b = Panjang sisi yang sejajar dengan CD

t = Tinggi (jarak tegak lurus antara AB dan CD)

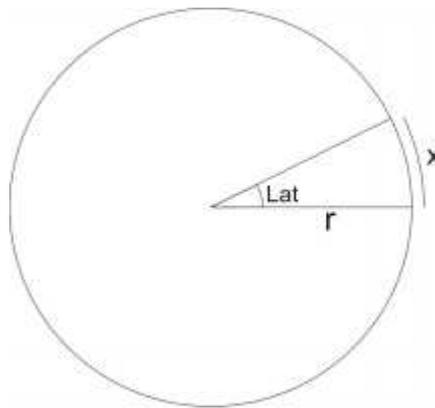
$$\begin{aligned}
 L &= \frac{(AB+CD)t}{2} \\
 &= \frac{(6+3)3}{2} \\
 &= \frac{27}{2} \\
 &= 13,5
 \end{aligned}$$

2.4 Penggunaan Koordinat *Latitude* dan *Longitude* pada *Shoelace Formula*

Koordinat *latitude* dan *longitude* memiliki satuan sudut (derajat), sedangkan persamaan *Shoelace Formula* ini menggunakan satuan bidang. Oleh karena itu kita harus mengkonversi satuan *latitude* dan *longitude* kedalam satuan bidang (meter). Untuk mengkonversinya dapat dilakukan cara berikut:

1. *Latitude*

Untuk mengkonversi *latitude* dapat dengan mudah dilakukan karena tidak ada perbedaan radius bumi disetiap derajat *longitude* nya, dapat diperhatikan pada Gambar 2.4.



Gambar 2.4 Konversi Satuan *Latitude* Kedalam Satuan Meter

Dengan melihat Gambar 2.4 dapat dibuat persamaan sederhana sebagai berikut:

$$\frac{x}{\text{keliling } r} = \frac{\text{lat}}{360}$$

$$x = \frac{\text{lat}}{360} \cdot \text{keliling } r$$

$$x = \frac{\text{lat}}{360} \cdot 2 \cdot \frac{22}{7} \cdot r$$

Dengan: x = Nilai *latitude* dalam satuan meter

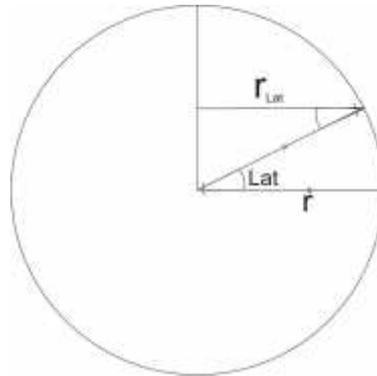
lat = Nilai *latitude* dalam satuan derajat

r = Radius bumi

keliling r = keliling bumi

2. Longitude

Untuk *longitude* sendiri sedikit berbeda dikarenakan radius bumi pada suatu *latitude* berbeda-beda. Dapat diperhatikan pada Gambar 2.5.



Gambar 2.5 Radius Pada Suatu *Latitude*

Dengan melihat Gambar 2.5 kita dapat mencari nilai radius pada suatu *latitude* dengan persamaan trigonometri sebagai berikut:

$$\cos lat = \frac{r}{r_{lat}}$$

$$r_{lat} = r \cdot \cos lat$$

Dengan: r_{lat} = Radius pada x *latitude*

r = Radius bumi

lat = Nilai *latitude* dalam satuan derajat

Setelah mengetahui radiusnya kita dapat mengkonversi satuan *longitude* kedalam meter seperti persamaan pada *latitude*, dapat dituliskan sebagai berikut:

$$\frac{y}{keliling r_{lat}} = \frac{long}{360}$$

$$y = \frac{long}{360} \cdot keliling r_{lat}$$

$$y = \frac{long}{360} \cdot 2 \cdot \frac{22}{7} \cdot r \cdot \cos lat$$

Dengan: y = Nilai *longitude* dalam satuan meter

Long = Nilai *longitude* dalam satuan derajat

r_{lat} = Radius pada x latitude

keliling r_{lat} = Keliling pada x latitude

2.5 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* yang menyertakan *middleware* (*virtual machine*) dan sejumlah aplikasi utama. Android merupakan modifikasi dari Kernel Linux (Andry, 2011).

Tujuan pembuatan sistem operasi ini adalah untuk menyediakan *platform* yang terbuka, yang memudahkan orang mengakses Internet menggunakan telepon seluler. Android juga dirancang untuk memudahkan pengembang membuat aplikasi dengan batasan yang minim sehingga kreativitas pengembang menjadi lebih berkembang (Andry, 2011).

Sebagai *Open Source* dan bebas dalam memodifikasi, di dalam Android tidak ada ketentuan yang tetap dalam konfigurasi *Software* dan *Hardware*. Fitur- fitur yang didapat dalam Android antara lain (Lee, 2011):

- *Storage* - Menggunakan SQLite, *database* yang ringan, untuk sebuah penyimpanan data.
- *Connectivity* - Mendukung GSM/EDGE, IDEN, CDMA, EV-DO, UMTS.
- *Bluetooth* (termasuk A2DP dan AVRCP), WiFi, LTE, dan WiMax.
- *Messaging* - Mendukung SMS dan MMS.

- *Web Browser* – Berbasiskan *open-source* WebKit, bersama mesin.
- Chrome's V8 JavaScript.
- *Media support* – Termasuk mendukung untuk beberapa media berikut : H.263, H.264 (dalam bentuk 3GP or MP4), MPEG-4 SP, AMR, AMRWB (dalam bentuk 3GP), AAC, HE-AAC (dalam bentuk MP4 atau 3GP), MP3, MIDI, Ogg Vorbis, WAV, JPEG, GIF, dan BMP.
- *Hardware support* – Sensor akselerasi, Kamera, Kompas Digital, Sensor Kedekatan, GPS.
- *Multi-touch* – Mendukung *multi-touch screens*.
- *Multi-tasking* – Mendukung aplikasi *multi-tasking*.
- *Flash-support* – Android 2.3 mendukung *Flash* 10.1.
- *Tethering* – Mendukung pembagian dari koneksi Internet sebagai *wired/wireless hotspot*.
- *Play store* – katalog aplikasi yang dapat di-*download* dan diinstal pada telepon seluler secara *online*, tanpa menggunakan PC (*Personal Computer*).
- Lingkungan pengembangan yang kaya, termasuk *emulator*, peralatan *debugging*, dan *plugin*.

2.5.1 Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - *Integrated Development Environment* (IDE) untuk pengembangan aplikasi Android, berdasarkan *IntelliJ* IDEA. Selain merupakan editor kode *IntelliJ* dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih

banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya (*Developers*, 2018):

- Sistem versi berbasis Gradle yang fleksibel.
- *Emulator* yang cepat dan kaya fitur.
- Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android.
- *Instant Run* untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru.
- *Template* kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh.
- Alat pengujian dan kerangka kerja yang ekstensif.
- Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain.
- Dukungan C++ dan NDK.
- Dukungan bawaan untuk Google *Cloud Platform*, mempermudah pengintegrasian Google *Cloud Messaging* dan *App Engine*.

2.5.2 Fundamental Aplikasi

Aplikasi Android ditulis dalam bahasa pemrograman Java, kode Java dikompilasi bersama dengan data *file resource* yang dibutuhkan oleh aplikasi dimana prosesnya di-*package* oleh *tools* yang dinamakan “*apt tools*” ke dalam paket Android sehingga menghasilkan *file* dengan ekstensi apk (*Android Package*). File apk itulah yang sebenarnya kita sebut dengan

aplikasi yang dapat diinstal di perangkat *mobile* nantinya. Ada empat jenis komponen pada aplikasi Android yaitu (Nazruddin, 2012):

1. *Activities*

Suatu *activity* akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan transaksi. Sebuah aplikasi Android bisa jadi hanya memiliki satu *activity*, tetapi umumnya aplikasi memiliki banyak *activity* tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Untuk pindah dari satu *activity* ke *activity* lain kita dapat melakukannya dengan satu *event*, misalkan *click* tombol, memilih opsi atau menggunakan *triggers* tertentu.

2. *Service*

Service tidak memiliki GUI, tetapi *service* berjalan secara *background*, sebagai contoh dalam memainkan musik, *service* mungkin memainkan musik atau mengambil data dari jaringan, tetapi setiap *service* harus berada dalam kelas induknya. Misalnya, *mediaplayer* sedang memutar lagu dari *list* yang ada, aplikasi ini akan memiliki dua atau lebih *activity* yang memungkinkan *user* untuk memilih lagu atau menulis sms sambil *player* sedang berjalan. *Service* dijalankan pada *thread* utama dari proses aplikasi.

3. *Broadcast Receiver*

Broadcast Receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. *Broadcast Receiver* tidak memiliki *user interface* (UI), tetapi memiliki sebuah *activity* untuk merespon informasi yang mereka terima, atau mungkin menggunakan *Notification Manager*

untuk memberitahu kepada pengguna, seperti lampu latar atau *vibrating* perangkat.

4. *Content Provider*

Content Provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam *file* sistem seperti *database* SQLite. *Content Provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*.

2.5.3 Versi Android

Sejak pertama kali muncul sampai sekarang, Android telah memiliki sejumlah pembaharuan. Pembaharuan ini dilakukan untuk memperbaiki *bug* dan menambah fitur-fitur yang baru. Versi-versi yang ada pada Android, yaitu (*Developers*, 2018):

1. Android versi 1.1
2. Android versi 1.5 (*Cupcake*)
3. Android versi 1.6 (*Donut*)
4. Android versi 2.0/2.1 (*Eclair*)
5. Android versi 2.2 (*Froyo: Frozen Yoghurt*)
6. Android versi 2.3 (*Gingerbread*)
7. Android versi 3.0/3.1 (*Honeycomb*)
8. Android versi 4.0 (*ICS: Ice Cream Sandwich*)
9. Android versi 4.1 (*Jelly Bean*)
10. Android versi 4.4 (*Kit Kat*)

11. Android versi 5.0 (*Lollipop*)
12. Android versi 6,0 (*Marshmallow*)
13. Android versi 7.0 (*Nougat*)
14. Android versi 8.0 (*Oreo*)

2.6 Google Maps API

2.5.1 Pengertian API

Application Programming Interface (API) merupakan beberapa *set class* dan *method* atau fungsi dan *signature* yang sederhana. API sendiri bertujuan utama untuk mengatasi “*clueless*” dalam membangun *software* yang berukuran besar, berawal dari sesuatu yang sederhana sampai ke yang kompleks dan merupakan perilaku komponen yang sulit dipahami (Halim, 2011).

2.5.2 Pengertian Google Maps API

Google *maps* API sendiri merupakan beberapa *set class* dan *method* atau fungsi dan *signature* yang sederhana yang dikeluarkan oleh *Google* untuk digunakan dalam aplikasi pemetaan. Beberapa tujuan dari penggunaan *Google Maps API* adalah untuk melihat lokasi, mencari alamat, mendapatkan petunjuk mengemudi dan lain sebagainya.

2.5.3 *Google Maps Android API Utility Library*

Google Maps Android API Utility Library adalah *open-source class library* yang berguna untuk berbagai aplikasi. Dalam *utility library* terdapat *spherical geometry utilities* yang berguna untuk menghitung jarak dan area. Fungsi ini berguna untuk bahan pertimbangan dalam pembuatan aplikasi.

2.6 Metode Pengembangan Sistem: Metode *Waterfall*

Tahapan pada metode *Waterfall* adalah sebagai berikut (Kadir, 2002):

1. Analisis Sistem

Tahap ini dimulai karena adanya permintaan terhadap sistem baru. Tujuan utama tahapan ini adalah untuk menentukan hal detail tentang yang akan dikerjakan oleh sistem yang diusulkan. Analisis sistem mencakup studi kelayakan dan analisis kebutuhan.

2. Desain Sistem

Target dari tahap ini adalah menghasilkan rancangan yang memenuhi kebutuhan yang ditentukan selama tahap analisis sistem. Hasil akhir berupa spesifikasi rancangan yang sangat rinci sehingga mudah diwujudkan pada saat pemrograman. Desain sistem dibagi menjadi dua subtahap, yaitu perancangan konseptual dan perancangan fisik.

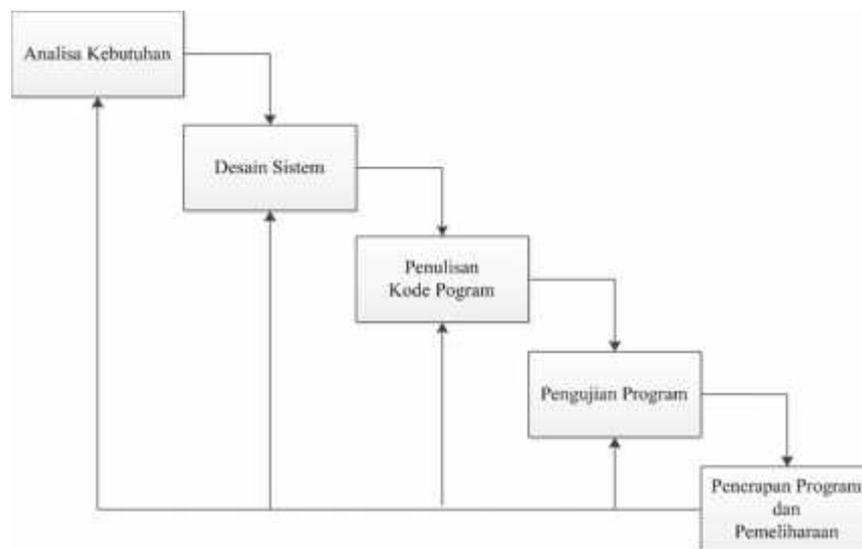
3. Impelementasi Sistem

Pada tahap ini aktivitas yang dilakukan adalah pemrograman dan pengujian, instalasi perangkat keras dan perangkat lunak, pelatihan kepada pemakai, pembuatan dokumentasi dan konversi.

4. Operasi dan Pemeliharaan

Sistem memasuki tahap operasi dan pemeliharaan setelah sistem baru berjalan sepenuhnya menggantikan sistem lama. Selama sistem beroperasi, pemeliharaan tetap diperlukan karena mungkin sistem masih menyisakan masalah-masalah yang tidak terdeteksi selama pengujian sistem. Alasan lainnya adalah pemeliharaan diperlukan karena kebutuhan bisnis atau lingkungan, atau adanya permintaan kebutuhan baru oleh pemakai dan pemeliharaan juga bisa dipicu karena kinerja sistem yang menjadi menurun sehingga seringkali adanya perubahan dalam penulisan program.

Metode *waterfall* digambarkan pada Gambar 2.6.



Gambar 2.6 Metode Pengembangan Sistem *Waterfall* (Kadir, 2002)

2.7 Teknik Pengujian Perangkat Lunak

Ada dua macam pendekatan kasus uji yaitu *white-box* dan *black-box*. Pendekatan *white-box* adalah pengujian untuk memperlihatkan cara kerja dari produk secara rinci sesuai dengan spesifikasinya. Pendekatan *black-box* merupakan pendekatan pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan (Jiang, 2012).

Kasus uji ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya. Teknik pengujian ini berfokus pada *domain* informasi dari perangkat lunak, yaitu melakukan kasus uji dengan mempartisi *domain input* dan *output* program. Metode *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian ini berusaha menemukan kesalahan dalam kategori fungsi-fungsi yang tidak benar atau hilang, kesalahan *interface*, kesalahan dalam struktur data atau akses basis data eksternal, kesalahan kinerja, dan inisialisasi dan kesalahan terminal (Pressman, 2001).

Teknik pengujian yang digunakan dalam penelitian ini terbagi atas pengujian fungsional dan pengujian non fungsional. Salah satu pengujian non fungsional menggunakan angket yang penyusunan bentuk jawaban dari pertanyaan menggunakan *Equivalence Partitioning*.

2.8 Partisi Ekuivalensi

Equivalence Partitioning (EP) merupakan metode *black box testing* yang membagi domain masukan dari program kedalam kelas-kelas sehingga *test case* dapat diperoleh. *Equivalence Partitioning* berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. Kasus uji yang didesain untuk *Equivalence Partitioning* berdasarkan pada evaluasi dari kelas ekuivalensi untuk kondisi masukan yang menggambarkan kumpulan keadaan yang valid atau tidak. Kondisi masukan dapat berupa spesifikasi nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi *Boolean*.

Kesetaraan kelas dapat didefinisikan menurut panduan berikut (Pressman, 2001):

1. Jika masukan kondisi menentukan kisaran, satu sah dan dua diartikan tidak valid kesetaraan kelas.
2. Jika masukan membutuhkan nilai, kondisi tertentu satu sah dan dua tidak valid kesetaraan kelas diartikan.
3. Jika masukan kondisi menentukan anggota dari set, satu sah dan satu tidak valid kesetaraan kelas diartikan.
4. Jika kondisi yang *input*, *Boolean* satu sah dan satu tidak valid kelas diartikan.

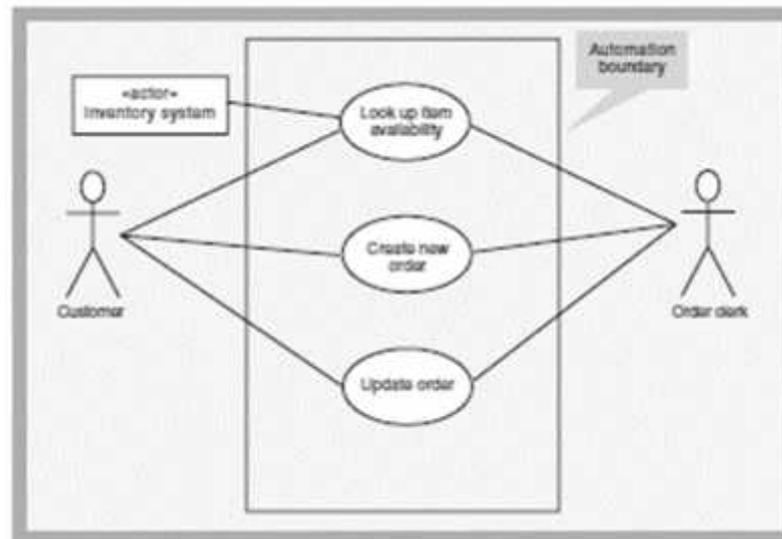
2.9 *Unified Modeling Language (UML)*

Menurut Martin Fowler dalam bukunya yang berjudul “*UML Distilled* Panduan Singkat Bahasa pemodelan Objek Standar”, UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain system perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO). Definisi ini merupakan definisi yang sederhana. Pada kenyataannya, pendapat orang – orang tentang UML berbeda satu sama lain. Hal ini dikarenakan oleh sejarahnya sendiri dan oleh perbedaan persepsi tentang apa yang membuat sebuah proses rancang bangun perangkat lunak efektif. UML lahir dari penggabungan banyak bahasa permodelan grafis berorientasi objek yang berkembang pesat pada akhir 1980-an dan awal 1990-an. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corp. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. UML dideskripsikan oleh beberapa diagram, yaitu sebagai berikut (Fowler, 2004):

1. *Use Case Diagram*

Use case Diagram digunakan untuk menggambarkan sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case diagram* lebih dititikberatkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Sebuah *use case diagram*

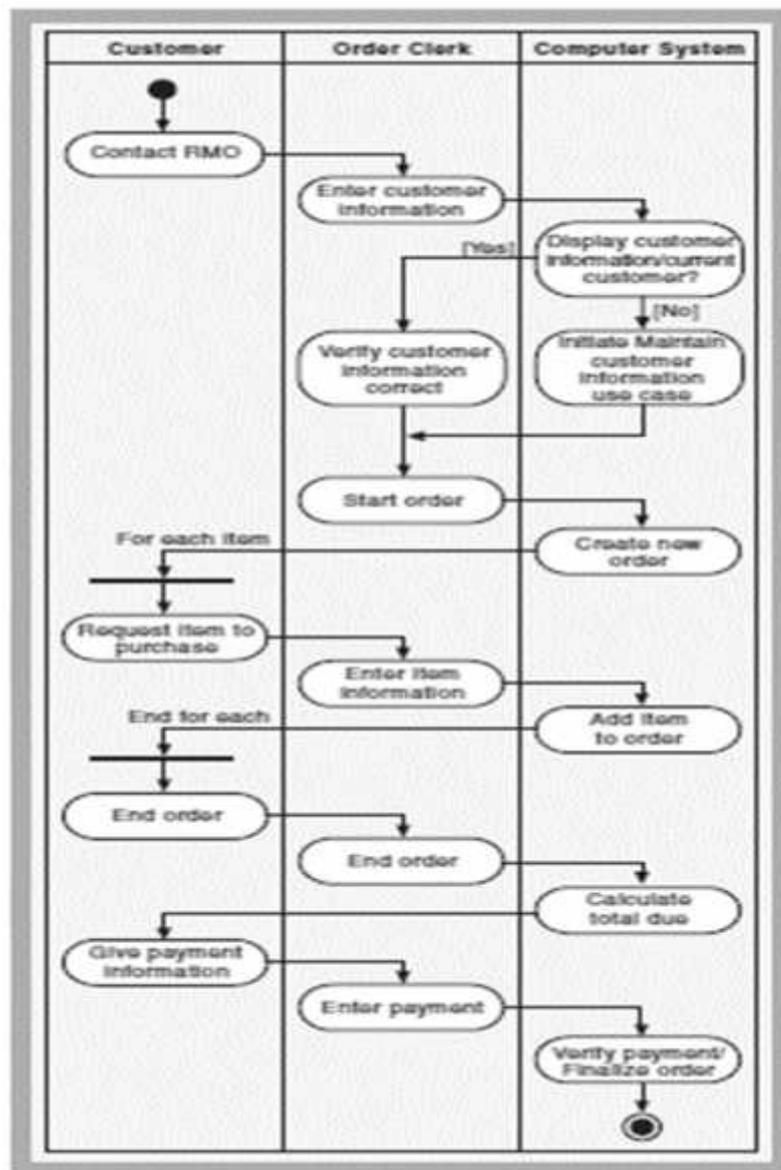
merepresentasikan sebuah interaksi antara aktor dengan sistem. Contoh *use case* disajikan pada Gambar 2.7.



Gambar 2.7 Use Case Diagram (Fowler, 2004).

2. Activity Diagram

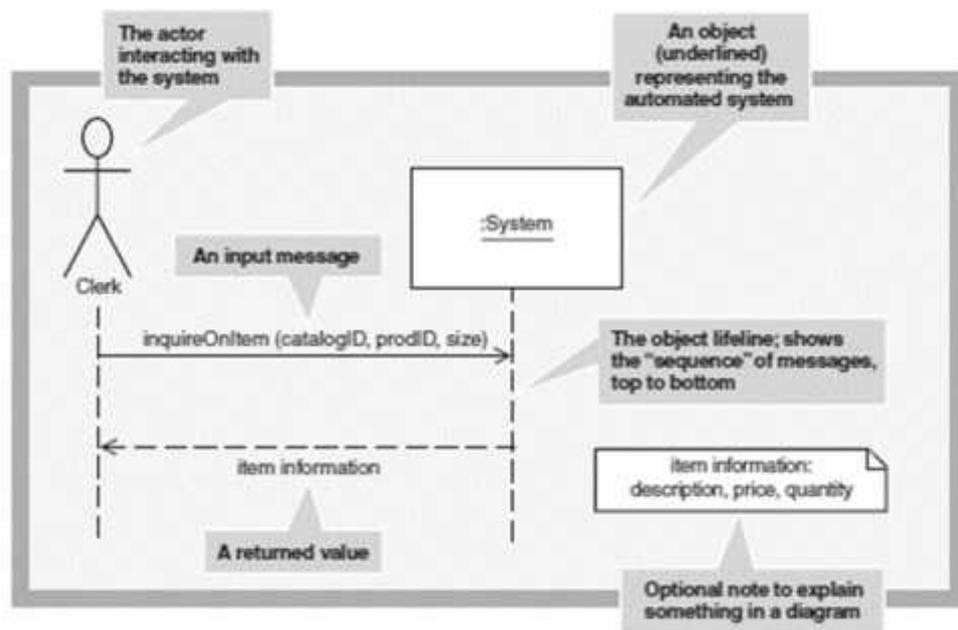
Diagram aktivitas atau *Activity Diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*) serta dapat digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. Aktivitas dalam diagram dipresentasikan dengan bentuk bujursangkar bersudut tidak lancip, yang didalamnya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja. Ada kedudukan awal (*start state*) yang menunjukkan dimulainya aliran kerja, dan sebuah kedudukan akhir (*end state*) yang menunjukkan akhir diagram, titik keputusan dipresentasikan dengan *diamond* (Fowler, 2004). Contoh diagram aktivitas disajikan pada Gambar 2.8.



Gambar 2.8 Diagram Aktivitas (Fowler, 2004).

3. Sequence Diagram

Diagram sekuensial atau *sequence diagram* digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Misalkan, pada *use case* “menarik uang” mempunyai beberapa kemungkinan, seperti penarikan uang secara normal, percobaan penarikan uang tanpa kecukupan ketersediaan dana, penarikan dengan penggunaan PIN yang salah, dan lainnya. Contoh diagram sekuensial disajikan pada Gambar 2.9 (Fowler, 2004).



Gambar 2.9 Diagram Sekuensial (Fowler, 2004).

BAB III METODOLOGI PENELITIAN

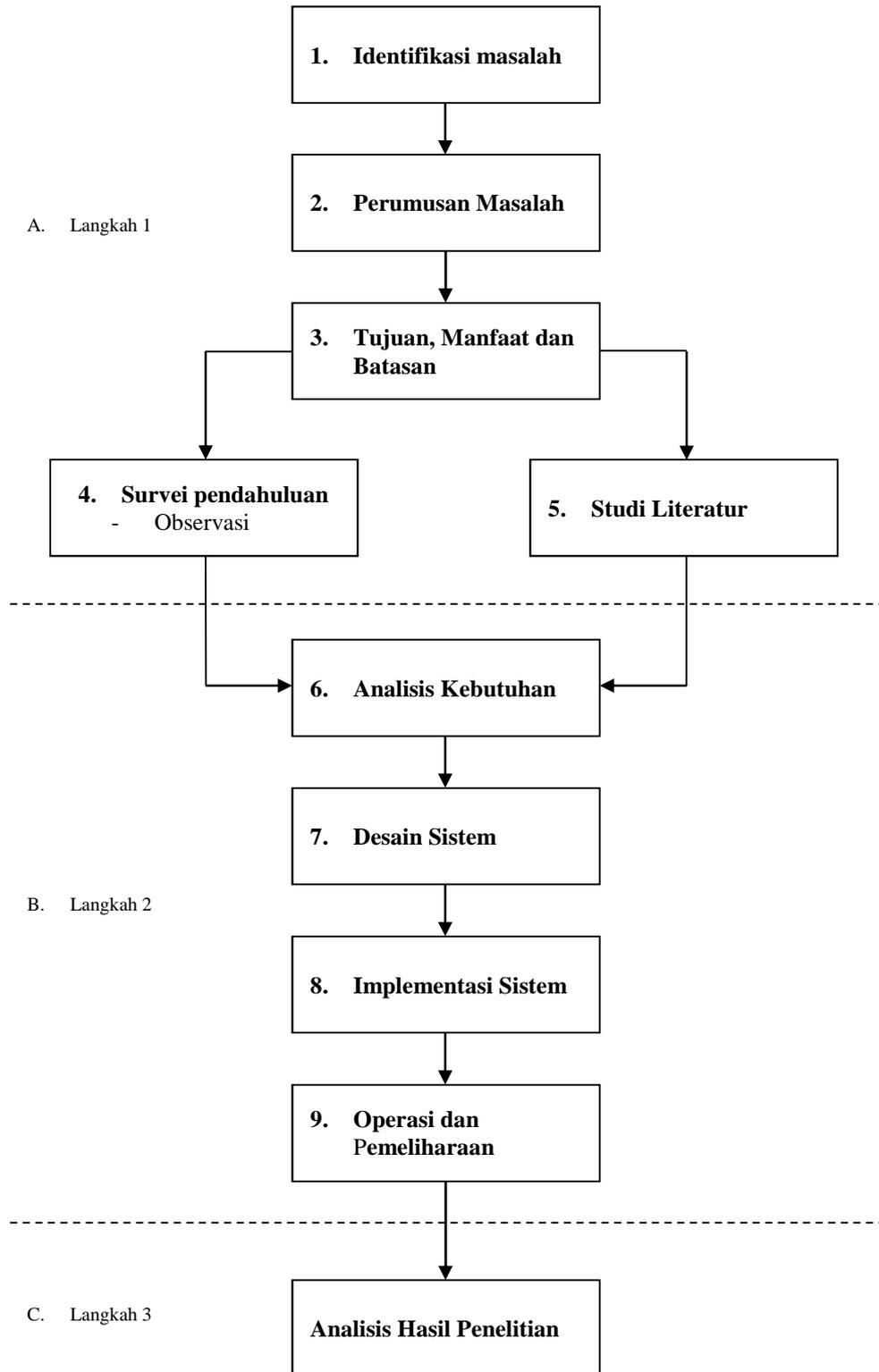
3.1 Waktu dan Tempat Penelitian

Penelitian dilaksanakan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di jalan Soemantri Brojonegoro No. 1 Gedung Meneng, Bandar Lampung. Penelitian ini dilakukan pada Semester Ganjil Tahun Ajaran 2017-2018.

3.2 Metodologi Penelitian

3.2.1 Alir Penelitian

Penelitian dilakukan berdasarkan diagram alir metodologi penelitian yang terdapat pada Gambar 3.1



Gambar 3.1 Diagram Alir Metodologi Penelitian

Penjelasan dari diagram alir metodologi penelitian pada Gambar 3.2 adalah sebagai berikut:

A. Langkah I

Pada langkah I dijelaskan secara umum mengenai penelitian yang dilakukan. Terdapat lima tahap pada langkah ini yaitu sebagai berikut:

1. Tahap pertama adalah identifikasi masalah. Identifikasi masalah adalah pengenalan masalah yang akan dipecahkan dalam penelitian.
2. Tahap kedua adalah perumusan masalah. Perumusan masalah adalah pertanyaan penelitian, yang umumnya disusun dalam bentuk kalimat tanya.
3. Tahap ketiga adalah menentukan tujuan, manfaat dan batasan. Target pencapaian dalam tahap ini adalah deketahuinya tujuan dan manfaat dari aplikasi Android pengukuran luas bidang permukaan bumi. Sedangkan batasan digunakan untuk membatasi pembahasan dan ruang lingkup penelitian.
4. Tahap keempat adalah survei pendahuluan. Survei pendahuluan ini dapat dimaksudkan untuk mengetahui hal-hal penting yang berhubungan dengan penelitian yang dikaji.
5. Tahap kelima adalah studi literatur. Studi literatur ini dapat dimaksudkan untuk mendapatkan gambaran menyeluruh tentang apa yang sudah dikerjakan sebagai teori-teori yang akan dijadikan landasan pada penelitian.

B. Langkah II

Pada langkah II dijelaskan mengenai pengembangan sistem pada penelitian yang dilakukan. Metode pengembangan sistem yang digunakan pada penelitian ini adalah *Waterfall*. Terdapat empat tahap pada langkah ini yaitu sebagai berikut:

6. Tahap keenam adalah analisis kebutuhan. Pada tahap ini akan dilakukan analisis kebutuhan dasar sistem dan analisis kebutuhan pengguna.
7. Tahap ketujuh adalah desain sistem. Pada tahap ini akan dilakukan perancangan sistem dan perancangan *interface* yang menggunakan *Unified Modelling Language* (UML), pada tahap ini juga akan diketahui semua entitas input ataupun output dalam sistem serta *usecase*, *class diagram*, *activity diagram*, *sequence diagram* yang digunakan dalam analisis sistem.
8. Tahap kedelapan adalah implementasi sistem. Pada tahap ini akan dilakukan pembuatan program yang akan dilanjutkan dengan implementasi ke Android. Kemudian dilakukan pengujian menggunakan *black box* (kotak hitam) dengan metode *Equivalence Partitioning* (EP), jika sistem tidak bekerja sesuai analisis maka akan kembali ke tahap sebelumnya sedangkan jika sistem bekerja sesuai analisis maka dilakukan ke tahap selanjutnya.
9. Tahap kesembilan adalah operasi dan pemeliharaan. Tahap ini akan dilakukan setelah sistem berjalan sepenuhnya. Pada tahap ini sistem akan dioperasikan dan dilakukan pemeliharaan karena mungkin

sistem masih menyisakan masalah-masalah yang tidak terdeteksi selama pengujian sistem.

C. Langkah III

Pada langkah III dijelaskan mengenai analisis hasil penelitian. Pada tahap terakhir ini menjelaskan berhasil atau tidaknya dalam suatu penelitian dengan permasalahan yang telah dideskripsikan di awal.

3.2.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Waterfall*. Tahap-tahapnya adalah sebagai berikut:

1. Analisis Sistem

Pada tahap ini akan dilakukan analisis tentang apa yang dibutuhkan dalam sistem dan pada tahap ini juga dilakukan analisis kebutuhan dasar sistem serta analisis kebutuhan pengguna.

a. Analisis kebutuhan dasar sistem

Analisis kebutuhan dasar sistem dilakukan untuk menentukan apa saja yang dapat dilakukan oleh sistem, antara lain:

- i. Menghitung luas dari titik simpul yang diberikan oleh user.
- ii. Menghitung jarak antar titik simpul yang diberikan oleh user.
- iii. Menghitung keliling dari titik simpul yang diberikan oleh user.
- iv. Menampilkan map beserta bentuk lokasi yang diinginkan user.

b. Analisis kebutuhan pengguna

Analisis kebutuhan pengguna dilakukan untuk menentukan apa saja yang dapat dilakukan oleh pengguna, antara lain:

- i. Memasukan titik simpul pada map yang ingin dihitung luas wilayahnya.
- ii. Melihat hasil perhitungan luas, jarak, dan keliling.

2. Desain Sistem

Pada tahap ini akan dilakukan perancangan dan desain sistem. Pada penelitian ini prancangan sistem menggunakan *Unified Modelling Language* (UML). Diagram-diagram UML yang dibuat dalam penelitian ini adalah sebagai berikut:

- a. *Use case* diagram.
- b. *Activity* diagram.
- c. *Sequence* diagram.

Aplikasi yang akan dibuat pada penelitian ini dengan interface yang dirancang sebagai berikut:

- i. *Layout* menu syarat dan ketentuan
- ii. *Layout* menu utama
- iii. *Layout* menu hitung luas area
- iv. *Layout* menu hitung luas area manual
- v. *Layout* menu hitung luas area auto
- vi. *Layout* menu bantuan
- vii. *Layout* menu tentang aplikasi

3. Implementasi Sistem

Pada tahap ini dilakukan proses pembuatan program. Pembuatan program dilakukan dengan menggunakan bahasa pemrograman Java dengan bantuan aplikasi Android Studio. Pada aplikasi ini dibuat beberapa *class* java yang menjelaskan jalannya aplikasi khususnya pada fungsi perhitungan luas bidang dan jarak permukaan bumi pada lokasi yang telah ditentukan. Perhitungan luas bidang dan jarak pada permukaan bumi menggunakan *library* API dari *Goggle Maps* dimana peta yang digunakan bersumber dari *goolge*. Aplikasi ini membutuhkan akses internet serta GPS. Akses internet digunakan untuk membuka *Goggle Maps* sedangkan GPS itu sendiri digunakan untuk menentukan lokasi di mana user berada. Setelah pembuatan program selesai dilakukan pengujian *black box* dengan metode *Equivalence Partitioning* (EP).

4. Operasi dan Pemeliharaan

Pada tahap ini akan dilakukan setelah sistem berjalan sepenuhnya. Pada tahap ini sistem akan dioperasikan dan dilakukan pemeliharaan karena mungkin sistem masih menyisakan masalah-masalah yang tidak terdeteksi selama pengujian sistem.

3.2.3 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan pada penelitian ini adalah:

1. Studi Literatur

Studi literatur didapatkan dari buku-buku, jurnal, prosiding dan internet yang menyediakan informasi tentang GIS, Android, dan bahasa pemrograman java.

2. Metode Spasial

Metode spasial digunakan untuk mendapatkan informasi dengan pengamatan yang dipengaruhi oleh lokasi.

3.3 Perancangan Sistem

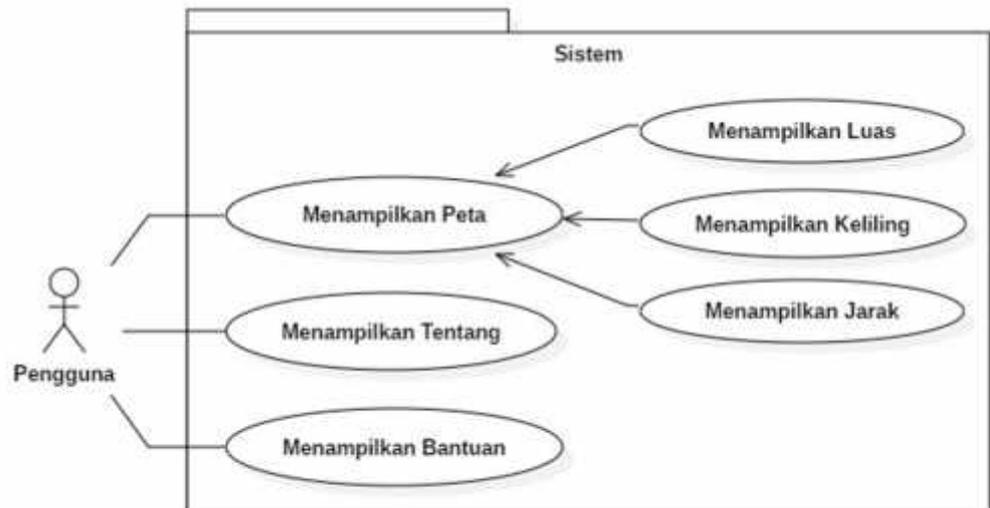
Perancangan sistem adalah tahap setelah analisis dari siklus pengembangan sistem. Perancangan sistem ini berupa penggambaran, perencanaan, dan pembuatan sketsa sistem yang nanti akan dibuat oleh pengembang. Perancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang harus diselesaikan. Tahapan ini termasuk mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah dilakukan instalasi akan benar-benar sesuai dengan rancangan sistem awal.

3.3.1 Perancangan UML (*Unified Modelling Language*)

Perancangan sistem atau desain sistem dilakukan dengan memodelkan permasalahan dalam bentuk diagram-diagram UML sebagai berikut:

1. *Use Case Diagram*

Use case diagram di bawah ini menjelaskan fungsionalitas dari aplikasi pengukuran luas bidang permukaan bumi berbasis Android. Pembuatan *use case* diagram ini lebih dititik beratkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. *Usecase* diagram aplikasi pengukuran luas bidang permukaan bumi berbasis Android dapat dilihat pada Gambar 3.2.



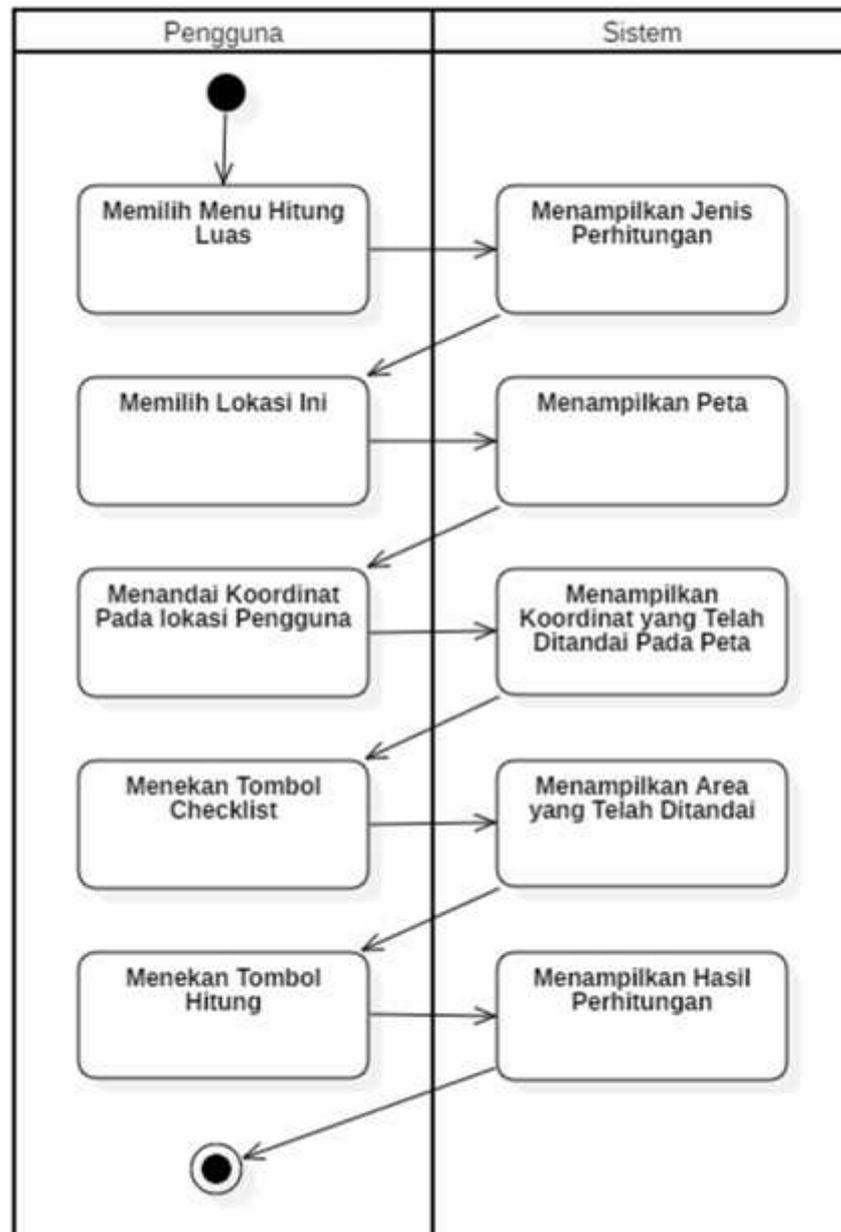
Gambar 3.2 *Usecase Diagram*

2. *Activity Diagram*

Activity diagram digunakan untuk menggambarkan aliran kerja (*workflow*) dari kejadian *usecase* diagram. *Activity* diagram pada sistem ini terbagi atas 4 bagian yaitu sebagai berikut:

a. *Activity* Diagram Perhitungan Luas Lokasi Ini

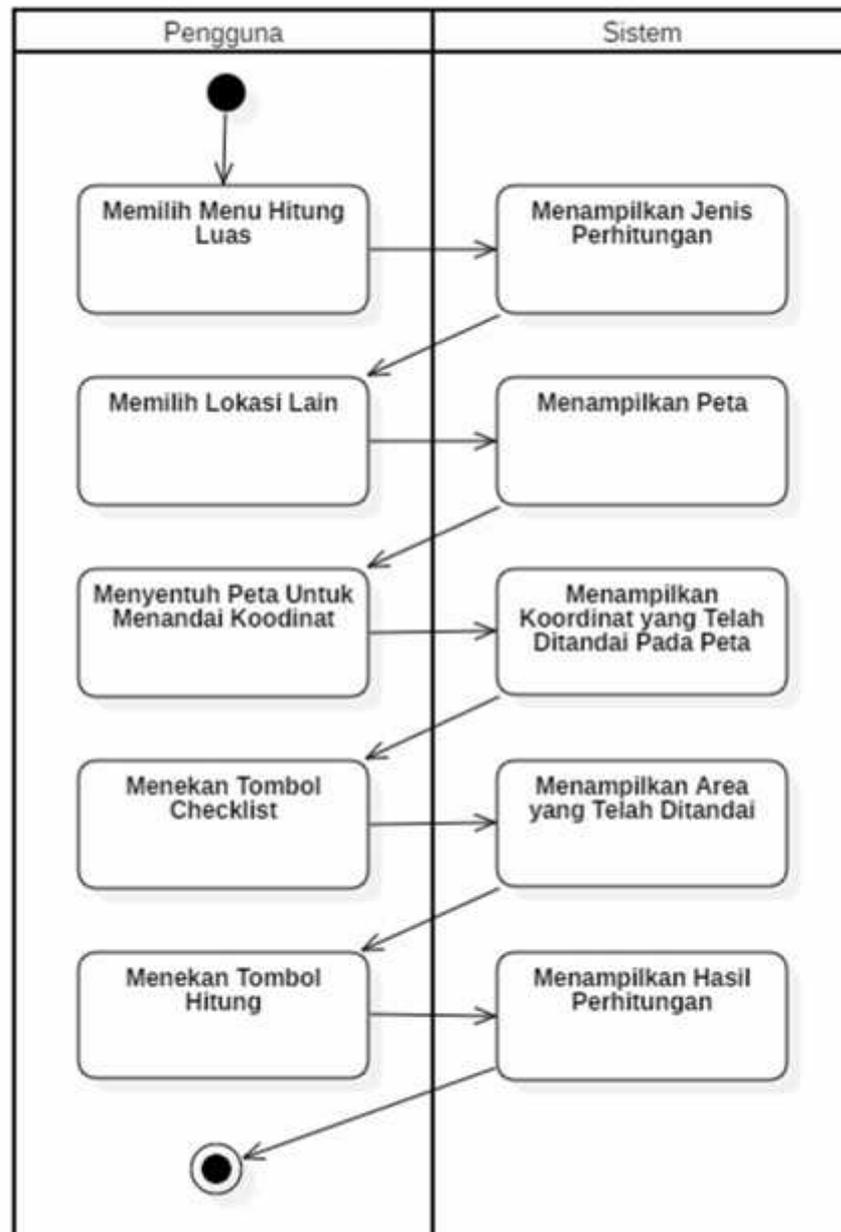
Activity diagram perhitungan luas lokasi ini digunakan untuk menggambarkan aliran fungsionalitas pada fungsi perhitungan luas lokasi ini. *Activity* diagram perhitungan luas lokasi ini dapat dilihat pada Gambar 3.3.



Gambar 3.3 *Activity* Diagram Perhitungan Luas Lokasi Ini

b. *Activity* Diagram Perhitungan Luas Lokasi Lain

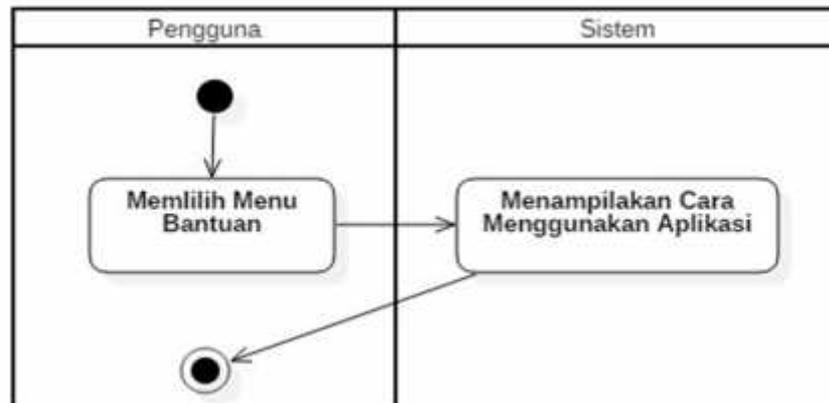
Activity diagram perhitungan luas lokasi lain digunakan untuk menggambarkan aliran fungsionalitas pada fungsi perhitungan luas lokasi lain. *Activity* diagram perhitungan luas lokasi lain dapat dilihat pada Gambar 3.4.



Gambar 3.4 Activity Diagram Perhitungan Luas Lokasi Lain

c. *Activity* Diagram Bantuan

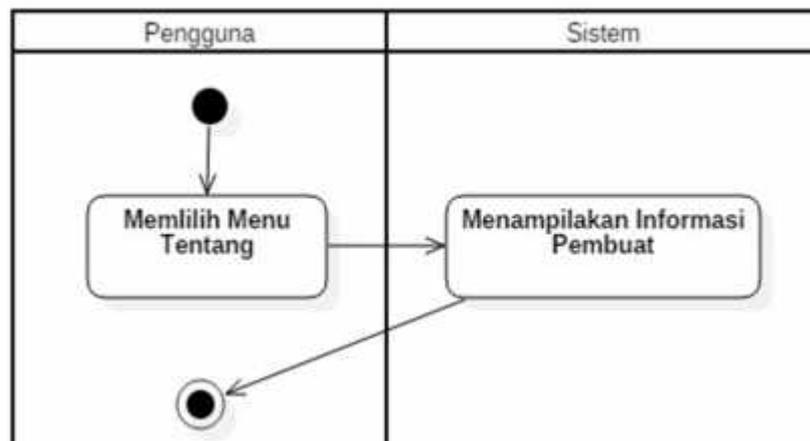
Activity diagram bantuan digunakan untuk menggambarkan aliran fungsionalitas pada menu bantuan. *Activity* diagram bantuan dapat dilihat pada Gambar 3.5.



Gambar 3.5 *Activity* Diagram Bantuan

d. *Activity* Diagram Tentang

Activity diagram tentang digunakan untuk menggambarkan aliran fungsionalitas pada menu tentang. *Activity* diagram tentang dapat dilihat pada Gambar 3.6.



Gambar 3.6 *Activity* Diagram Tentang

3. Sequence Diagram

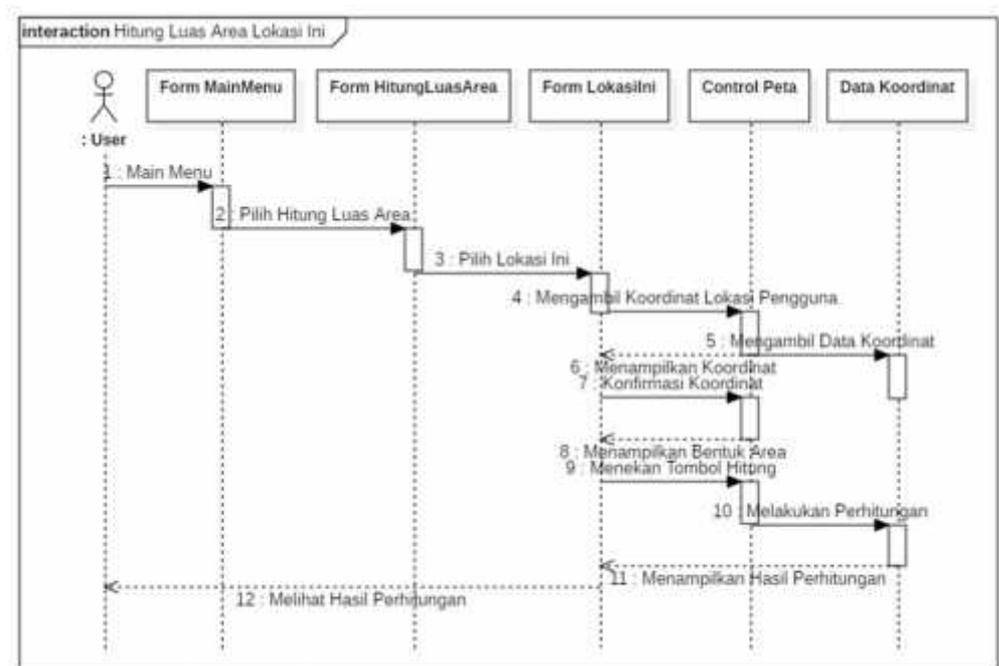
Sequence diagram digunakan untuk menggambarkan aliran fungsionalitas dari kejadian *usecase* diagram. *Sequence* diagram pada sistem ini terbagi atas 3 bagian yaitu sebagai berikut:

a. *Sequence* Diagram Perhitungan Luas

Sequence diagram perhitungan luas terbagi menjadi 2 yaitu:

- *Sequence* Diagram Perhitungan Luas Lokasi Ini

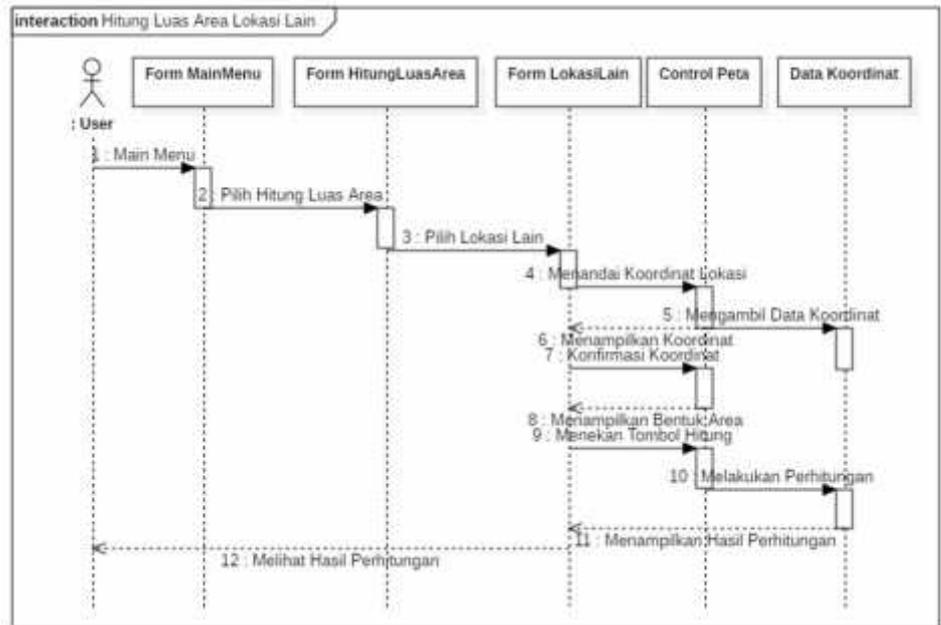
Sequence diagram perhitungan luas lokasi ini dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Sequence* Diagram Perhitungan Luas Lokasi Ini

- *Sequence* Diagram Perhitungan Luas Lokasi Lain

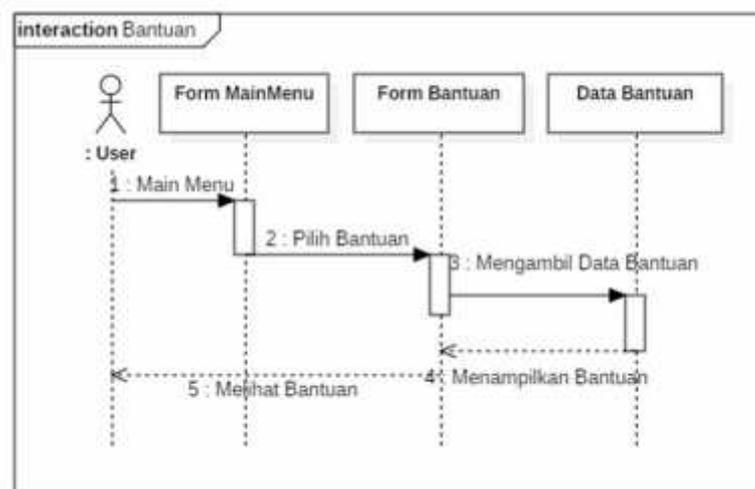
Sequence diagram perhitungan luas lokasi ini dapat dilihat pada Gambar 3.8.



Gambar 3.8 Sequence Diagram Perhitungan Luas Lokasi Lain

b. *Sequence Diagram Bantuan*

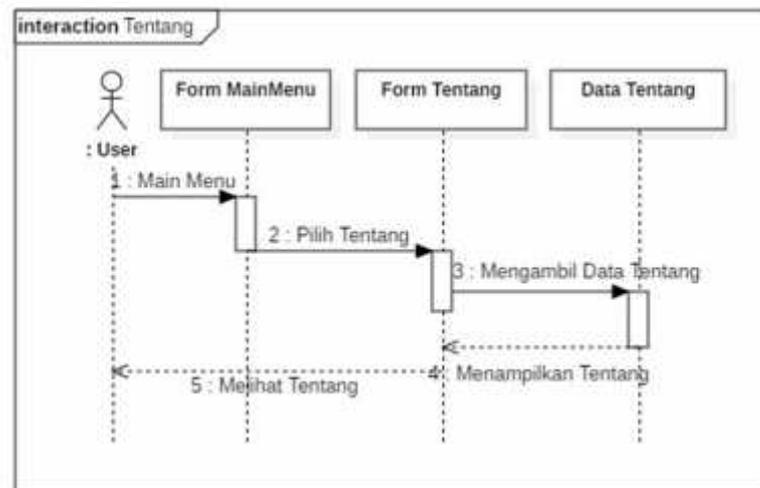
Sequence diagram bantuan dimulai ketika user memilih menu bantuan. Kemudian sistem akan membawa user pada *form* “bantuan”. *Form* bantuan terdiri dari *textbox* yang bersisi tentan cara penggunaan aplikaisi ini. *Sequence* diagram bantuan dapat dilihat pada Gambar 3.9.



Gambar 3.9 *Sequence Diagram* Bantuan

c. *Sequence Diagram* Tentang

Sequence diagram tentang tidak jauh berbeda dengan *sequence* diagram bantuan, hanya isinya saja yang berbeda. *Sequence* diagram bantuan dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Sequence Diagram* Tentang

3.3.2 Perancangan Antarmuka

Perancangan antarmuka merupakan proses penggambaran bagaimana sebuah *interface* sistem dibentuk. Aplikasi ini dirancang dengan tampilan user friendly, sehingga diharapkan pengguna dapat menggunakan aplikasi ini dengan mudah. Dalam aplikasi ini terdapat beberapa *layout* antara lain:

1. *Layout* Menu Syarat dan Ketentuan

Layout ini menampilkan syarat dan ketentuan penggunaan *system*, pengguna harus menyetujui syarat dan ketentuan sebelum menggunakan aplikasi. Perancangan *layout* menu syarat dan ketentuan dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Layout* Menu Syarat dan Ketentuan

2. *Layout* Menu Utama

Layout menu utama pada aplikasi ini menampilkan menu-menu yang dapat digunakan oleh pengguna. Menu tersebut berisi hitung luas area, bantuan, dan tentang. Perancangan *layout* menu utama dapat dilihat pada Gambar 3.12.



Gambar 3.12 *Layout* Menu Utama

3. *Layout* Menu Hitung Luas Area

Pada menu ini aplikasi menampilkan menu lain yang di gunakan pengguna untuk menentukan cara perhitungan luas area. Menu ini berisi menu hitung luas area lokasi ini dan menu hitung luas area lokasi lain.

Perancangan *layout* menu hitung luas area dapat dilihat pada Gambar 3.13.



Gambar 3.13 *Layout* Menu Hitung Luas Area

4. *Layout* Menu Hitung Luas Area Lokasi Ini

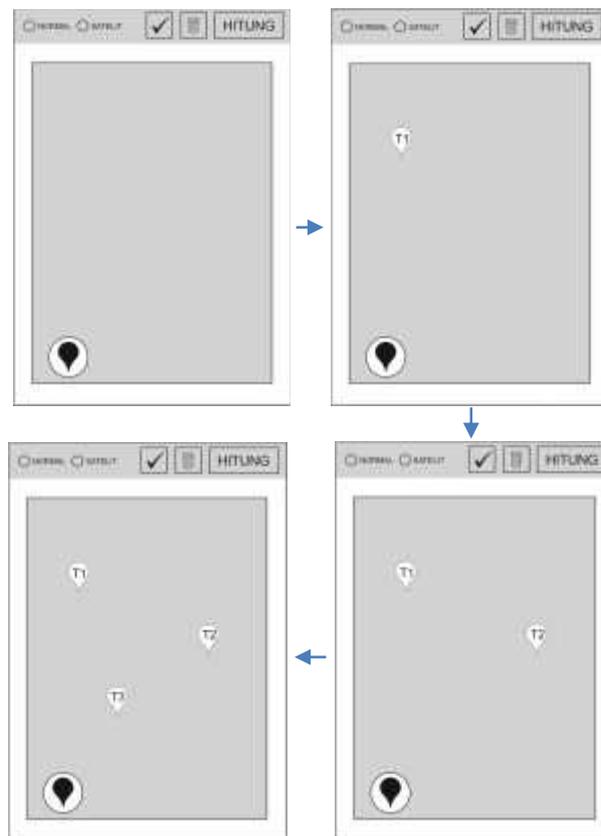
Pada menu ini menampilkan peta dan menu perhitungan yang dapat *user* gunakan untuk menghitung luas yang ada pada permukaan bumi berdasarkan koordinat *user*. Perancangan *layout* menu hitung luas area lokasi ini dapat dilihat pada Gambar 3.14.



Gambar 3.14 *Layout* Menu Hitung Luas Area Lokasi Ini

a. Peta

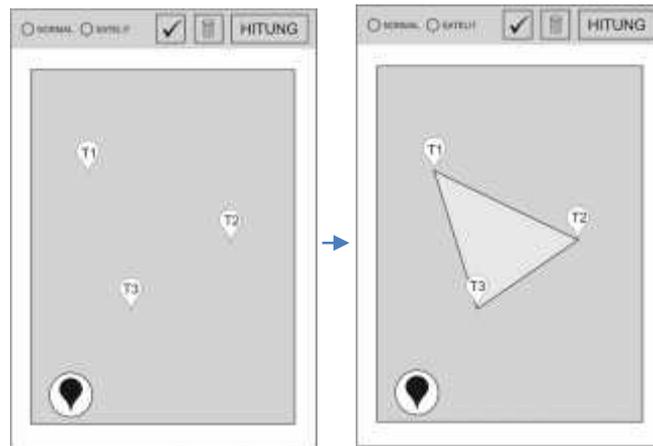
Pada peta terdapat tombol yang dapat digunakan pengguna untuk menandai titik simpul pada lokasi pengguna. Pengguna dapat menandai minimal tiga titik simpul yang nantinya titik simpul tersebut digunakan untuk menghitung luas area tersebut. Sebagai contoh dapat dilihat pada Gambar 3.15.



Gambar 3.15 *Layout* Menu Hitung Luas Area Ini Ketika Pengguna Menempatkan Koordinat Titik Simpul

b. Tombol *Checklist*

Tombol *checklist* digunakan pengguna untuk mengkonfirmasi titik simpul yang telah ditandai. Sistem juga akan menampilkan bentuk area. Sebagai contoh dapat dilihat pada Gambar 3.16.



Gambar 3.16 *Layout* Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol *Checklist*

c. Tombol Hapus

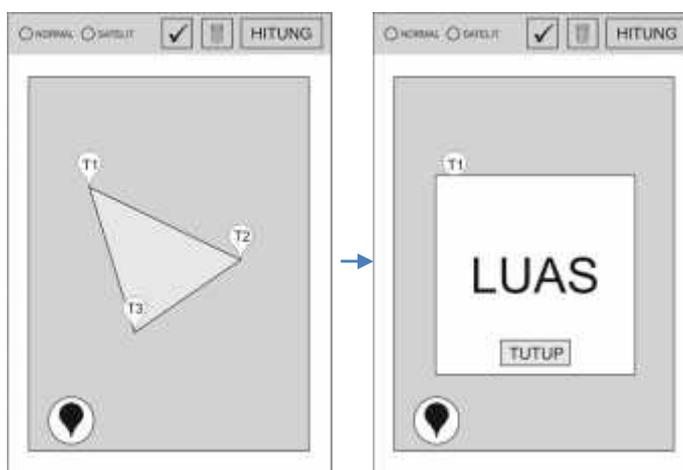
Tombol hapus digunakan pengguna untuk menghapus titik simpul yang sudah ada sebagai contoh dapat dilihat pada Gambar 3.17.



Gambar 3.17 *Layout* Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol Hapus

d. Tombol Hitung

Pengguna dapat menekan tombol hitung untuk melakukan perhitungan luas terhadap area yang sudah ditandai dengan titik simpul sebelumnya. Tombol ini hanya akan berfungsi ketika titik simpul telah dikonfirmasi. Saat ditekan sistem akan menampilkan menu yang merupakan hasil perhitungan. Sebagai contoh dapat dilihat pada Gambar 3.18.



Gambar 3.18 *Layout* Menu Hitung Luas Area Ini Ketika Pengguna Menekan Tombol Hitung

5. *Layout* Menu Hitung Luas Area Lokasi Lain

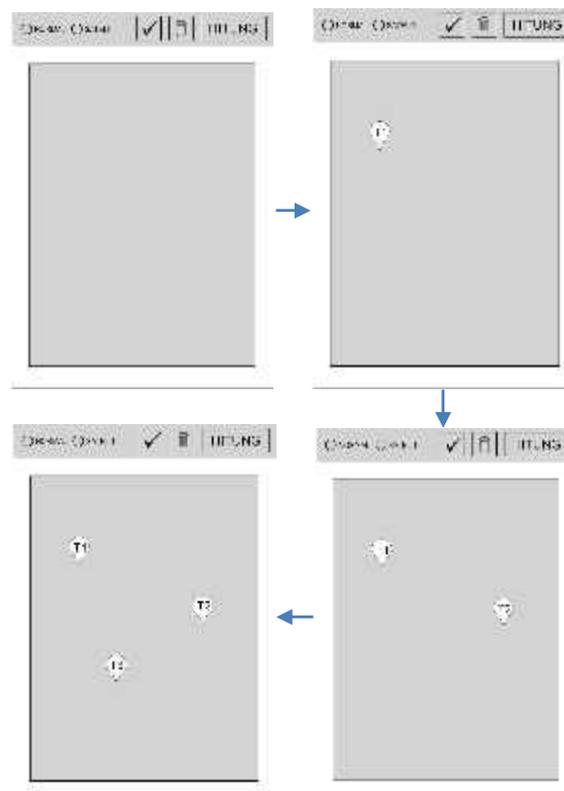
Pada menu ini menampilkan peta dan menu perhitungan yang dapat *user* gunakan untuk menghitung luas yang ada pada permukaan bumi berdasarkan koordinat yang *user* tandai. Perancangan *layout* menu hitung luas area lokasi lain dapat dilihat pada Gambar 3.19.



Gambar 3.19 *Layout* Menu Hitung Luas Area Lokasi Lain

a. Peta

Pada peta pengguna dapat melakukan sentuhan untuk menempatkan koordinat titik simpul yang akan dihitung luas areanya. Sebagai contoh dapat dilihat pada Gambar 3.20.



Gambar 3.20 *Layout* Menu Hitung Luas Area Lain Ketika Pengguna Menempatkan Koordinat Titik Simpul

b. Tombol *Checklist*, Hapus, dan Hitung

Fungsi tombol checklist, hapus, dan hitung sama seperti fungsi tombol checklist, hapus, dan hitung pada menu hitung luas area lokasi ini.

6. *Layout* Menu Bantuan

Menu bantuan ini berguna untuk memberitahu pengguna tentang bagaimana cara menggunakan aplikasi ini. Perancangan *layout* menu bantuan dapat dilihat pada Gambar 3.21.



Gambar 3.21 *Layout* Menu Bantuan

7. *Layout* Menu Tentang

Menu tentang aplikasi ini berfungsi untuk memberitahu pengguna tentang pembuat aplikasi ini. *Layout* menu tentang dapat dilihat pada Gambar 3.22.



Gambar 3.22 *Layout* Menu Tentang

3.4 Metode Pengujian Sistem

Metode pengujian sistem dalam penelitian ini adalah pengujian Black Box dengan metode Equivalence Partiotioning (EP). Pengujian ini berguna untuk membuktikan semua fungsi-fungsi pada aplikasi ini berjalan dengan baik.

Pada pengujian ini berguna untuk mencocokkan bahwa masukan dan respon yang diterima sesuai dengan analisis sistem. Metode ini digunakan karena metode ini dapat mencari kesalahan pada fungsi aplikasi, interface aplikasi dan kesalahan pada struktur data aplikasi. Rancangan daftar pengujian disajikan pada Tabel 3.1.

Tabel 3.1 Daftar Pengujian Equivalence Partiotioning (EP)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Realisasi yang Diharapkan
1	Versi Android	Pengujian kompatibilitas versi <i>operating system</i> Android	Pengujian pada Android versi 2.2 (<i>Froyo</i>)	Tidak kompatibel dengan Android versi 2.2 (<i>Froyo</i>)
			Pengujian pada Android versi 2.3 (<i>Gingerbread</i>)	Tidak Kompatibel dengan Android versi 2.3 (<i>Gingerbread</i>)
			Pengujian pada Android versi 3.0/3.1 (<i>Honeycomb</i>)	Tidak Kompatibel dengan Android versi 3.0/3.1 (<i>Honeycomb</i>)
			Pengujian pada Android versi 4.0 (<i>Ice Cream Sandwich</i>)	Kompatibel dengan Android versi 4.0 (<i>Ice Cream Sandwich</i>)
			Pengujian pada Android versi 4.1 (<i>Jelly Bean</i>)	Kompatibel dengan Android versi 4.1 (<i>Jelly Bean</i>)
			Pengujian pada Android versi 4.4 (<i>Kit Kat</i>)	Pengujian pada Android versi 4.4 (<i>Kit Kat</i>)
			Pengujian pada Android Versi 5.0 (<i>Lollipop</i>)	Kompatibel dengan Android Versi 5.0 (<i>Lollipop</i>)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Realisasi yang Diharapkan
			Pengujian pada Android Versi 6.0 (<i>Marshmallow</i>)	Kompatibel dengan Android Versi 6.0 (<i>Marshmallow</i>)
			Pengujian pada Android Versi 7.0 (<i>Nougat</i>)	Pengujian pada Android Versi 7.0 (<i>Nougat</i>)
			Pengujian pada Android Versi 8.0 (<i>Oreo</i>)	Pengujian pada Android Versi 8.0 (<i>Oreo</i>)
2	Ukuran Layar	Pengujian Ukuran Layar pada Android	Pengujian pada Android dengan ukuran layar 4 inch	Kompatibel pada pada Android dengan ukuran layar 4 inch
			Pengujian pada Android dengan ukuran layar 5 inch	Kompatibel pada Android dengan ukuran layar 5 inch
			Pengujian pada Android dengan ukuran layar 6 inch	Kompatibel pada Android dengan ukuran layar 6 inch
			Pengujian pada Android dengan ukuran layar 7 inch	Kompatibel pada Android dengan ukuran layar 7 inch
			Pengujian pada Android dengan ukuran layar 8 inch	Kompatibel pada Android dengan ukuran layar 8 inch
3	Fungsi <i>layout</i> Menu Utama	Pengujian Pada <i>layout</i> Menu Utama	Klik tombol menu “Menu Hitung Luas Area ”	Menampilkan <i>layout</i> Menu Hitung Luas Area
			Klik tombol menu “Bantuan”	Menampilkan <i>layout</i> Bantuan
			Klik tombol menu “Tentang Aplikasi”	Menampilkan <i>layout</i> Tentang Aplikasi
4	Fungsi <i>layout</i> Menu Hitung Luas Area	Pengujian pada <i>layout</i> peta	Melakukan <i>touch</i> pada peta	Menampilkan POI pada peta yang merupakan titik simpul yang digunakan untuk menghitung luas area pada peta
		Pengujian tombol pada <i>menu</i>	Klik tombol “Hapus”	Menghapus seluruh POI yang ada pada peta
			Klik tombol “Hitung”	Menampilkan luas area pada peta yang telah ditandai. Menu ini dapat dilakukan jika minimal ada tiga POI

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Realisasi yang Diharapkan
5	Fungsi pada menu Tentang Aplikasi	Pengujian pada menu Tentang Aplikasi	Klik menu Tentang Aplikasi	Menampilkan <i>layout</i> Tentang aplikasi
6	Fungsi pada menu Bantuan	Pengujian pada menu Bantuan	Klik menu Bantuan Aplikasi	Menampilkan <i>layout</i> Bantuan aplikasi

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Hasil penelitian yang dilakukan, dapat diambil simpulan sebagai berikut:

1. Aplikasi pengukuran luas bidang permukaan bumi dapat digunakan sebagai media perhitungan luas tanah yang bentuknya tidak beraturan.
2. Hasil pengujian dapat disimpulkan bahwa semua fungsi pada aplikasi dapat dijalankan dengan baik.
3. Hasil pengujian akurasi dapat disimpulkan bahwa aplikasi dapat berfungsi dengan baik dengan selisih luas rata-rata 2%.
4. Hasil pengujian perbandingan aplikasi dapat disimpulkan bahwa aplikasi dengan menggunakan *Shoelace Formula* dapat berfungsi dengan baik dengan selisih luas rata-rata 0.16%.

5.2 Saran

Saran dalam aplikasi ini dapat dikembangkan supaya dapat melakukan perhitungan luas pada bidang tanah yang miring. Fungsi ini berguna untuk pengguna sebagai bahan pertimbangan dalam melakukan perhitungan luas tanah.

DAFTAR PUSTAKA

- Abdul Kadir, 2002. *Pengenalan Sistem Informasi*, Yogyakarta: Penerbit Andi.
- Andry. 2011. *Android A sampai Z*. Jakarta: PCplus.
- Barden, Bard. *The Surveyor's Area Formula*, Washington: *The College Mathematics Journal*.
- Developers, Android. 2018. *Android Developers*. [Online]. Tersedia: <http://developer.android.com/index.html>. Diakses pada tanggal 27 Februari 2018.
- Developers, Android. 2018. *Android Studio Developers*. [Online]. Tersedia: <https://developer.android.com/studio/index.html>. Diakses pada tanggal 27 Februari 2018.
- Fowler, Martin. 2004. *UML Distilled Panduan Singkat Bahasa pemodelan Objek Standar, Edisi 3*. Yogyakarta: Andi Publishing.
- Halim, J I. 2011. *Framework Pemetaan Data Berbasis Peta dengan Menggunakan Google Maps API (Skripsi)*. Universitas Bina Nusantara. Jakarta.
- Jiang, F., Y. Lu. 2012. *Software testing model selection research based on yin-yang testing theory*. In: *IEEE Proceeding of International Conference on Computer Science and Information Processing (CISP)*, pp. 590-594.
- Lee, W. M. 2011. *Beginning Android Application Development*. Wiley Publishing, Inc.

Suryantoro, Agus. 2013. *Integrasi Aplikasi Sistem Informasi Geografis*. Yogyakarta: Penerbit Ombak.

Nazruddin, Safaat H. 2012. *(Edisi Revisi) Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.

Pressman, Roger S. 2001. *Software Engineering A Practitioner's Approach Fifth Edition*. New York: McGraw-HillCompanies, Inc.