

**RANCANG BANGUN APLIKASI LAYANAN BERBASIS LOKASI  
DENGAN PENERAPAN *AUGMENTED REALITY* MENGGUNAKAN  
METODE *MARKERLESS* BERBASIS ANDROID  
(Studi Kasus : Pencarian Perangkat Daerah Kota Bengkulu)**

**SKRIPSI**



**Oleh :**

**JULIA PURNAMA SARI**

**G1A009041**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS BENGKULU  
2014**

**RANCANG BANGUN APLIKASI LAYANAN BERBASIS LOKASI  
DENGAN PENERAPAN *AUGMENTED REALITY* MENGGUNAKAN  
METODE *MARKERLESS* BERBASIS ANDROID  
(Studi Kasus : Pencarian Perangkat Daerah Kota Bengkulu)**

**SKRIPSI**

**Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana pada  
Fakultas Teknik Program Studi Teknik Informatika  
Universitas Bengkulu**



**Oleh :**  
**JULIA PURNAMA SARI**  
**G1A009041**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS BENGKULU  
2014**

## MOTTO DAN PERSEMBAHAN

### Motto :

- Aku yakin Allah SWT selalu bersamaku, di setiap kesulitan dan kebahagiaanku. Allah tidak akan pernah tidur. Allah akan memberikan yang terbaik untukku.
- Tidak ada yang tidak mungkin bagi Allah jika Allah berkehendak. Hargai kekuranganmu. Tetap berusaha, berdoa, lakukan yang terbaik, dan hasilnya serahkan pada Allah.
- Di dalam kesulitan pasti ada kemudahan, ada jalan keluar, bersabarlah.
- Kalau saja aku tidak melihat indahnya garis *finish*, mungkin aku akan berhenti sekarang. *Keep Fighting!*

### Persembahan :

```
public void setDedication(People everyOne){  
    ▪ Allah SWT.  
    ▪ Nabi Muhammad SAW.  
  
    ▪ Ayah dan ibu tercinta, Bakarudin dan Suriani.  
    ▪ Saudara-saudara tersayang, Alharis Suryanto, Lina Marleni, Eva Juanarti, Suparno, Diana, Leni, Abdi, dan Febri.  
    ▪ Keponakanku tersayang, Muhammad Khapiz Annur, Waridaniah Assyifah, Naurah Rayani Abdillah, Muhammad Ma'ruf Harlen Pratama, Safira, Aulia, Soko Gandhi, Fahri Hakki.  
  
    ▪ Teman-teman, Reza R, Dhita, Inu, Disa, Sherly, Mifta, Chilla, Abdur, Dery, Lian, Randi, Gita J, Dwinda Etika, Ratna, Mulya, Fadli, Arry, Mirza, Rio N, Juki, Ryza, Sostri, Apni, Handrie, Odie, Abner, Dian, Roro,Azhar, Bobby, Ejo, Eko, Ferry, Firdaus, Ghufron, Indri,Linda, Ariansyah, Rinov, Munadi, Oni, Rozy, Samuel, Tri, Yessica, Zulmi, Denny, Egi, Eleo, Fitri, Aji, Irawan, Leni, Rewa, Meilia, Radias, Robbie, Yoggy, Praja, Tara, Bimo, kak Willy, kak Dean, kak Deva, dll.  
    ▪ SD Negeri 41 Kota Bengkulu, SMP Negeri 4 Kota Bengkulu, SMA Negeri2 Kota Bengkulu dan Almamater UNIB.  
}
```

## KATA PENGANTAR



*Assalamu'alaikum Warahmatullahi Wabarakatuh*

*Alhamdulillahirabbil'alamin*, puji syukur penulis ucapkan kepada Allah SWT yang telah memberikan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “**Rancang Bangun Aplikasi Layanan Berbasis Lokasi Dengan Penerapan Augmented Reality Menggunakan Metode Markerless Berbasis Android (Studi Kasus: Pencarian Perangkat daerah Kota Bengkulu)**” dengan lancar.

Skripsi ini merupakan tugas akhir untuk memenuhi salah satu syarat dalam menyelesaikan studi di Fakultas Teknik Universitas Bengkulu. Selain itu, skripsi ini juga merupakan suatu pembelajaran bagi penulis dalam banyak hal, baik ilmu pengetahuan, praktik ilmu yang didapat selama kuliah di Teknik Informatika dan bekal ilmu ke depannya. Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan. Oleh karena itu, penulis masih perlu banyak belajar lagi dan membutuhkan saran yang membangun dalam skripsi ini. Penulis juga menyadari bahwa tanpa bantuan, saran, dan bimbingan dari pihak-pihak yang telah membantu, penulis tidak akan mampu menyelesaikan tugas akhir ini. Untuk itu penulis mengucapkan terima kasih kepada yang terhormat:

1. Ayah (Bakarudin), Ibu (Suriani) dan saudara-saudaraku tersayang (Alharis Suryanto, Lina Marleni, Eva Juanarti, Suparno, Diana, Leni Andriani, Zahman Abdi, dan Febri) yang selalu memberikan semangat dan dukungan dalam penyelesaian skripsi ini. Terutama ibu yang selalu mengingatkan penulis dan memberikan dukungan moril kepada penulis.
2. Bapak Khairul Amri, S.T., M.T. sebagai Dekan Fakultas Teknik Universitas Bengkulu yang telah memberikan kesempatan kepada penulis untuk menyelesaikan pendidikan di Fakultas Teknik Universitas Bengkulu.

3. Ibu Ernawati, S.T., M.Cs. selaku Pembimbing Utama yang penuh kesungguhan, kesabaran dan bersedia meluangkan waktu yang cukup banyak untuk membimbing, mengarahkan dan memotivasi penulis dalam menyelesaikan skripsi ini.
4. Bapak Aan Erlansari, S.T., M.Eng. selaku Pembimbing Pendamping yang selalu memberikan waktu, arahan dan saran kepada penulis sehingga penulis menjadi lebih termotivasi dan bersemangat untuk menyelesaikan skripsi ini.
5. Bapak Rusdi Efendi, S.T., M.Kom sebagai dosen Pengaji Utama yang telah menyediakan waktu untuk mengarahkan dan memberikan masukan-masukan demi penyempurnaan skripsi ini.
6. Ibu Desi Andreswari, S.T., M.Cs., selaku ketua Prodi Teknik Informatika Universitas Bengkulu dan Pengaji Pendamping yang telah menyediakan waktu untuk mengarahkan dan mengaji skripsi ini.
7. Seluruh dosen dan staf karyawan, serta seluruh civitas akademika Program Studi Teknik Informatika Fakultas Teknik Universitas Bengkulu.
8. Mbak Fenty Adri sebagai staf administrasi Prodi Teknik Informatika yang telah membantu proses administrasi sehingga tugas akhir ini dapat selesai tepat waktu.
9. Teman-teman terutama Reza, Dhita, Wisnu, Disa, Sherly, Mifta, Chilla, Yody, Abdur, Randi, Nopiliansyah, Dery, dan Gita J yang selalu memberikan saran dan dukungan.
10. Teman-teman satu jurusan Teknik Informatika baik kakak tingkat maupun adik tingkat yang selalu memberikan dukungan.
11. Semua pihak yang sudah membantu dan tidak dapat penulis sebutkan satu persatu.

Penulis berharap semoga Allah SWT menganugerahkan rahmat dan hidayah-Nya kepada mereka semua. Semoga skripsi ini dapat bermanfaat bagi kita semua,aamiin.

*Wassalamu'alaikum Warahmatullahi Wabarakatuh*

Bengkulu, Juni 2014

Penulis

**RANCANG BANGUN APLIKASI LAYANAN BERBASIS LOKASI  
DENGAN PENERAPAN *AUGMENTED REALITY* MENGGUNAKAN  
METODE *MARKERLESS* BERBASIS ANDROID  
(Studi Kasus : Pencarian Perangkat Daerah Kota Bengkulu)**

Oleh  
JULIA PURNAMA SARI  
G1A009041  
E-mail: julia.purnamasari09@gmail.com

**ABSTRAK**

Penelitian ini bertujuan untuk membangun suatu aplikasi yang dapat membantu pengguna dalam mencari dan memberikan informasi mengenai perangkat daerah Kota Bengkulu pada *android* dengan menerapkan *augmented reality* menggunakan metode *markerless*. Pada penerapannya *augmented reality* bisa diimplementasikan pada perangkat yang memiliki kamera, akselerometer, kompas dan GPS. Data koordinat dan data perangkat daerah Kota Bengkulu di *download* dari *server*. Aplikasi ini juga memanfaatkan peta dari *google maps* dalam menunjukkan lokasi pengguna ke lokasi perangkat daerah Kota Bengkulu yang dicari. Aplikasi ini dibangun dengan menggunakan *Wikitude* dan bahasa pemrograman *Java for Android* dengan IDE Eclipse 3.5. Metode pengembangan sistem yang digunakan untuk membangun aplikasi ini adalah model *sekuensial linier* dan *Unified Modeling Language* (UML) sebagai perancangan sistem. Penelitian ini menghasilkan sebuah aplikasi layanan berbasis lokasi dengan penerapan *augmented reality* menggunakan metode *markerless* berbasis *android* pada pencarian perangkat daerah Kota Bengkulu.

**Kata kunci:** *Augmented Reality*, *Markerless*, Perangkat Daerah Kota Bengkulu, *Android*, *Google Maps*, GPS.

**BUILDING DESIGN OF LOCATION BASED SERVICE APPLICATION BY  
IMPLEMENTING AUGMENTED REALITY USING ANDROID-BASED  
MARKERLESS METHOD**

**(A Case Study : The Search of Region Government of Bengkulu City)**

By  
JULIA PURNAMA SARI  
G1A009041  
E-mail: julia.purnamasari09@gmail.com

**ABSTRACT**

*This research aims to create an application that can help the users in searching and giving information about the region government of Bengkulu City on android by implementing the augmented reality using markerless method. On the implementation, augmented reality can be implemented on the device that has a camera, an accelerometer, a compass, and GPS. The data of the coordinate and the region government are downloaded from the server. The application also makes use maps from google maps in showing the users' location to the location of the region government looked for. This application is created by using Wikitude and the programming language Java for Android with IDE Eclipse 3.5. The system development method used to build the application is linear sequential model and Unified Modeling Language (UML) as the system designing. This research produced a location-based service application with the implementation of augmented reality using android-based markerless method basing on the search of the region government Bengkulu City.*

**Keywords:** Augmented Reality, Markerless, The Region Government Bengkulu City, Android, Google Maps, GPS.

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERSETUJUAN.....	iii
LEMBAR PENGESAHAN SKRIPSI .....	Error! Bookmark not defined.
MOTTO DAN PERSEMPAHAN .....	v
KATA PENGANTAR .....	vi
ABSTRAK .....	ix
ABSTRACT .....	xi
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiv
DAFTAR TABEL.....	xvi
DAFTAR LAMPIRAN.....	xvii
BAB I PENDAHULUAN.....	1
1.1    Latar Belakang.....	1
1.2    Rumusan Masalah.....	3
1.3    Batasan Masalah .....	3
1.4    Tujuan Penelitian.....	4
1.5    Manfaat Penelitian .....	4
1.6    Sistematika Penelitian.....	5
BAB II LANDASAN TEORI.....	7
2.1    Konsep Dasar Sistem Informasi.....	7
2.1.1    Sistem .....	7
2.1.2    Informasi .....	7
2.1.3    Sistem Informasi.....	8
2.2    Pengertian Layanan Berbasis Lokasi (LBS).....	8
2.3 <i>Augmented Reality</i> .....	11
2.4 <i>Markerless Augmented Reality</i> .....	12
2.5    Prinsip Kerja <i>Markerless Augmented Reality</i> .....	14
2.6    Android .....	16
2.6.1    Sejarah Android .....	16
2.6.2    Arsitektur Android.....	17

2.6.3 Versi Android .....	20
2.7 Google Maps API .....	23
2.7.1 Pengertian API.....	23
2.7.2 Pengertian Google Maps API .....	23
2.8 JSON ( <i>Java Script Object Notation</i> ).....	24
2.9 Metode Pengembangan Sistem .....	25
2.10 <i>Unified Modeling Language (UML)</i> .....	27
2.11 Pengujian Sistem.....	31
2.11.1 Teknik Pengujian Sistem .....	31
2.11.2 Pengukuran Tingkat Kepuasan Pengguna .....	32
2.12 Penelitian Terkait.....	33
<b>BAB III METODOLOGI PENELITIAN.....</b>	<b>36</b>
3.1 Jenis Penelitian.....	36
3.2 Sarana Pendukung dan Sarana Pengujian .....	36
3.3 Jenis dan Sumber Data.....	37
3.4 Teknik Pengumpulan Data.....	38
3.5 Metode Perancangan dan Pengembangan Sistem.....	38
3.6 Metode Pengujian .....	41
3.7 Metode Uji Kelayakan Sistem .....	42
3.8 Jadwal Penelitian .....	43
<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM .....</b>	<b>44</b>
4.1 Identifikasi Permasalahan .....	44
4.2 Analisis Sistem.....	45
4.2.1 Pemahaman Kerja Sistem yang Ada.....	45
4.2.2.1 Alur Sistem Aplikasi.....	45
4.2.2.2 Alur Sistem Server Pada <i>Web Browser</i> .....	47
4.2.2 Analisis Fungsional.....	48
4.2.3 Analisis Non-Fungsional .....	48
4.3 Perancangan Sistem .....	49
4.3.1Perancangan Model UML ( <i>Unified Modeling Language</i> ).....	49
4.3.2Perancangan Antarmuka .....	66

BAB V HASIL PEMBAHASAN .....	73
5.1 Implementasi Sistem.....	75
5.1.1 Pengujian Sistem Untuk Admin Pada <i>Web Browser</i> .....	77
5.1.2 Pengujian Sistem Pada Emulator .....	80
5.1.2.1 <i>Splash Screen</i> .....	80
5.1.2.2 Halaman Utama .....	83
5.1.2.3 Menu Petunjuk.....	86
5.1.2.4 Menu Info .....	88
5.1.2.5 Menu MAP AR .....	90
5.1.2.6 Menu Tentang.....	92
5.2 Implementasi Pada <i>Smartphone</i> Android .....	94
5.3 Uji Ketepatan Titik Lokasi Pada Peta dan AR.....	98
5.4 Uji Jarak Maksimal Objek Muncul di Kamera .....	108
5.5 Uji Kelayakan Sistem .....	111
5.6 Uji Jenis Android Berbeda dan Versi Berbeda.....	137
5.7 Kelebihan dan Kekurangan Aplikasi .....	139
5.7.1 Kelebihan .....	139
5.7.2 kekurangan.....	140
BAB VI PENUTUP .....	142
6.1 Kesimpulan .....	142
6.2 Saran .....	143
DAFTAR PUSTAKA .....	144
LAMPIRAN .....	146

## DAFTAR GAMBAR

Gambar 2.1 Prinsip Kerja <i>Markerless AR</i> .....	15
Gambar 2.2 Model <i>Sekuensial Linier</i> .....	25
Gambar 2.3 <i>Three Complementary Views or Sets of Diagram</i> .....	29
Gambar 4.1 Diagram Alir Aplikasi .....	45
Gambar 4.2 Diagram Alir Sistem Server Pada <i>Web Browser</i> .....	47
Gambar 4.3 <i>Use Case Diagram</i> Aplikasi .....	50
Gambar 4.4 <i>Use Case Diagram</i> Sistem Server .....	50
Gambar 4.5 <i>Activity Diagram</i> Manajemen POI Pada Sistem Server .....	51
Gambar 4.6 <i>Activity Diagram</i> Menu Petunjuk .....	52
Gambar 4.7 <i>Activity Diagram</i> Menu Info .....	52
Gambar 4.8 <i>Activity Diagram</i> Menu MAP AR .....	53
Gambar 4.9 <i>Activity Diagram</i> Menu Tentang .....	54
Gambar 4.10 <i>Activity Diagram</i> Keluar.....	54
Gambar 4.11 <i>Sequence Diagram</i> Manajemen POI.....	55
Gambar 4.12 <i>Sequence Diagram</i> Manajemen Admin.....	56
Gambar 4.13 <i>Sequence Diagram</i> Menu Petunjuk .....	56
Gambar 4.14 <i>Sequence Diagram</i> Menu Info .....	57
Gambar 4.15 <i>Sequence Diagram</i> Cari .....	58
Gambar 4.16 <i>Sequence Diagram</i> Peta .....	58
Gambar 4.17 <i>Sequence Diagram</i> AR .....	59
Gambar 4.18 <i>Sequence Diagram</i> Menu Tentang .....	60
Gambar 4.19 <i>Class Diagram</i> .....	60
Gambar 4.20 <i>Object Diagram</i> .....	63
Gambar 4.21 <i>State Chart Diagram</i> <i>Web Browser</i> .....	64
Gambar 4.22 <i>State Chart Diagram</i> Aplikasi .....	64
Gambar 4.23 <i>Collaboration Diagram</i> .....	65
Gambar 4.24 <i>Deployment Diagram</i> .....	66
Gambar 4.25 <i>User Interface</i> <i>Web Browser Update</i> POI .....	67
Gambar 4.26 <i>User Interface</i> <i>Web Browser</i> Tampilan Data POI .....	67
Gambar 4.27 <i>User Interface</i> <i>Web Browser Update</i> Admin.....	68
Gambar 4.28 <i>User Interface</i> <i>Web Browser</i> Tampilan Data Admin.....	68
Gambar 4.29 <i>User Interface</i> <i>Splash Screen</i> .....	69
Gambar 4.30 <i>User Interface</i> Menu Utama .....	70
Gambar 4.31 <i>User Interface</i> Petunjuk .....	70
Gambar 4.32 <i>User Interface</i> Daftar Menu Info .....	71
Gambar 4.33 <i>User Interface</i> Tampil Informasi Info .....	71
Gambar 4.34 <i>User Interface</i> Menu MAP AR.....	72
Gambar 4.35 <i>User Interface</i> Peta .....	73
Gambar 4.36 <i>User Interface</i> AR .....	73
Gambar 4.37 <i>User Interface</i> Tentang .....	74
Gambar 4.38 <i>User Interface</i> Keluar Aplikasi.....	74
Gambar 5.1 Halaman <i>Login</i> <i>Web Browser</i> .....	77
Gambar 5.2 Halaman Utama <i>Web Browser</i> .....	78
Gambar 5.3 Halaman Tampil Data POI .....	78
Gambar 5.4 Halaman Manajemen POI.....	79
Gambar 5.5 Halaman Tambah POI .....	80
Gambar 5.6 Halaman Menu Administrator .....	80
Gambar 5.7 Tampilan <i>Splash Screen</i> .....	82

Gambar 5.8 Tampilan Halaman Utama .....	85
Gambar 5.9 Tampilan Menu Petunjuk .....	87
Gambar 5.10 Tampilan Menu Info .....	89
Gambar 5.11 Tampilan Informasi .....	90
Gambar 5.12 Tampilan Menu MAP AR Pencarian.....	91
Gambar 5.13 Tampilan Tentang .....	93
Gambar 5.14 Tampilan <i>Splash Screen</i> .....	94
Gambar 5.15 Tampilan Menu Utama Aplikasi .....	95
Gambar 5.16 Tampilan Daftar Informasi .....	96
Gambar 5.17 Tampilan Informasi Perangkat Daerah Kota Bengkulu.....	96
Gambar 5.18 Tampilan Menu MAP AR .....	97
Gambar 5.19 Tampilan Pencarian Lokasi Ketika Dimasukkan Nama Lokasi Yang Di Cari .....	97
Gambar 5.20 Tampilan Pencarian Lokasi Ketika Lokasi Dipilih Dari Daftar .....	98
Gambar 5.21 Tampilan Detail Data Lokasi.....	99
Gambar 5.22 Tampilan Pesan <i>Error</i> Perangkat Daerah Tidak Terisi.....	100
Gambar 5.23 Tampilan Pesan <i>Error</i> Perangkat Daerah Tidak Tersedia .....	100
Gambar 5.24 Tampilan Pesan <i>Error</i> Perangkat Daerah Tidak Tersedia Atau Dimasukkan Sembarang .....	101
Gambar 5.25 Rute Pada <i>Google Maps</i> .....	102
Gambar 5.26 Pencarian Dinas Tenaga Kerja, Pemuda, dan Olahraga .....	103
Gambar 5.27 Rute Posisi Pengguna ke Lokasi Yang Dicari Pada <i>Google Maps</i> .....	104
Gambar 5.28 Tampilan AR Pada Menu MAP AR .....	104
Gambar 5.29 Tampilan Menu Petunjuk .....	105
Gambar 5.30 Tampilan Tentang Aplikasi .....	105
Gambar 5.31 Tampilan Menu Keluar Aplikasi .....	106
Gambar 5.32 Lokasi Awal Pengguna .....	107
Gambar 5.33 Lokasi Akhir Pengguna .....	107
Gambar 5.34 Titik Lokasi Pada Kamera .....	108
Gambar 5.35 Jarak Objek Muncul Pada Kamera Di Lihat Dari Mega Mall .....	109
Gambar 5.36 Jarak Objek Muncul Pada Kamera Di Lihat Dari Bakso Simpang Lima .....	110
Gambar 5.37 Daerah Rata-Rata Persetujuan Komposisi Warna .....	114
Gambar 5.38 Daerah Rata-Rata Persetujuan Kejelasan Teks Yang Ada .....	116
Gambar 5.39 Daerah Rata-Rata Persetujuan Variasi Tampilan .....	117
Gambar 5.40 Daerah Rata-Rata Persetujuan Kualitas Tampilan.....	119
Gambar 5.41 Grafik Persentase Hasil Angket Variabel Tampilan.....	120
Gambar 5.42 Daerah Rata-Rata Persetujuan Kemudahan Menginstall Sistem.....	122
Gambar 5.43 Daerah Rata-Rata Persetujuan Kemudahan Mengoperasikan Sistem .123	
Gambar 5.44 Daerah Rata-Rata Persetujuan Kemudahan Memahami Informasi Yang Di Berikan.....	125
Gambar 5.45 Grafik Persentase Hasil Angket Kemudahan Pengguna.....	126
Gambar 5.46 Daerah Rata-Rata Persetujuan Tujuan Sistem .....	128
Gambar 5.47 Daerah Rata-Rata Persetujuan Fitur-Fitur Sistem .....	130
Gambar 5.48 Daerah Rata-Rata Persetujuan Urutan Penyajian Pada Sistem.....	131
Gambar 5.49 Daerah Rata-Rata Persetujuan Kecepatan Akses Sistem.....	133
Gambar 5.50 Daerah Rata-Rata Persetujuan Ketepatan Informasi Sistem.....	134
Gambar 5.51 Daerah Rata-Rata Persetujuan Kesesuaian Hasil Informasi .....	136
Gambar 5.52 Grafik Persentase Hasil Angket Kinerja Sistem .....	137

## **DAFTAR TABEL**

Tabel 3.1 Jadwal Penelitian yang Akan Dilaksanakan .....	43
Tabel 4.1 Kelas, Atribut dan <i>Method</i> .....	61
Tabel 5.1 Daftar <i>Layout</i> Aplikasi.....	75
Tabel 5.2 Daftar <i>Class</i> Aplikasi.....	76
Tabel 5.3 Pengujian <i>Black Box Splash Screen</i> .....	83
Tabel 5.4 Pengujian <i>Black Box</i> Halaman Utama .....	85
Tabel 5.5 Pengujian <i>Black Box</i> Menu Petunjuk.....	88
Tabel 5.6 Penguji Black Box Menu Info .....	90
Tabel 5.7 Pengujian <i>Black Box</i> Menu MAP AR.....	91
Tabel 5.8 Penguji <i>Black Box</i> Menu Tentang.....	94
Tabel 5.9 Kategori Penilaian.....	112
Tabel 5.10 Hasil Penilaian Variabel Tampilan .....	113
Tabel 5.11 Hasil Penilaian Variabel Kemudahan Pengguna .....	120
Tabel 5.12 Hasil Penilaian Variabel Kinerja Sistem.....	126
Tabel 5.13 Uji Menggunakan Perangkat Yang Berbeda dan Versi Berbeda.....	137

## **DAFTAR LAMPIRAN**

Lampiran A-1 Surat Izin Penelitian .....	A-1
Lampiran B-1 Pola Organisasi Perangkat Daerah .....	B-1
Lampiran C-1 Data Perangkat Daerah Kota Bengkulu.....	C-1
Lampiran D-1 Data Koordinat Perangkat Daerah Kota Bengkulu .....	D-1
Lampiran E-1 Hasil Pengujian <i>Method</i> Sistem (Uji <i>WhiteBox</i> ). ....	E-1
Lampiran E-2 Hasil Pengujian <i>Atribut</i> Sistem (Uji <i>WhiteBox</i> ) .....	E-2
Lampiran F-1 Tahap Pengujian Fungsional Sistem (Uji <i>BlackBox</i> ).....	F-1
Lampiran F-2 Hasil Pengujian Fungsional Sistem (Uji <i>BlackBox</i> ) .....	F-3
Lampiran G-1 Angket Uji Kelayakan Sistem .....	G-1
Lampiran H-1 Tabulasi Data Hasil Perhitungan Angket Uji Kelayakan.....	H-1
Lampiran I-1 Hasil Uji Coba Aplikasi Pada <i>Smartphone</i> Android .....	I-1
Lampiran I-2 Hasil Uji Coba Aplikasi Untuk <i>Augmented Reality</i> .....	I-7

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Wilayah provinsi terdiri atas beberapa kabupaten/kota. Pemerintah provinsi merupakan pemerintah yang melakukan pembinaan dan pengawasan terhadap pemerintah kabupaten/kota. Pemerintah kabupaten/kota mempunyai kewajiban dalam melaksanakan urusan yang menyangkut pemberdayaan daerah. Berdasarkan Peraturan Pemerintah Republik Indonesia Nomor 8 Tahun 2003 Tentang Pedoman Organisasi Perangkat Daerah Presiden Republik Indonesia, Perangkat Daerah adalah organisasi/lembaga pada Pemerintah Daerah yang bertanggung jawab kepada Kepala Daerah dalam rangka penyelenggaraan pemerintahan yang terdiri dari Sekretariat Daerah, Dinas Daerah dan Lembaga Teknis Daerah, Kecamatan, dan Satuan Polisi Pamong Praja sesuai dengan kebutuhan daerah.

Kota Bengkulu di pimpin oleh walikota dan didampingi wakil walikota. Perangkat daerah Kota Bengkulu terdiri dari staf ahli, Setda (Sekretariat Daerah) dengan 3 asisten, DPRD, 15 dinas daerah (unsur pelaksana), 8 badan dan 4 kantor lembaga teknis daerah (unsur pendamping), 3 lembaga lain, sekretariat DPRD, camat, dan lurah (Pola organisasi perangkat daerah kota Bengkulu,2013). Perangkat daerah Kota Bengkulu berupaya dalam memenuhi kebutuhan masyarakat yaitu dengan meningkatkan pelayanan masyarakat yang merupakan salah satu masalah dan tantangan yang dihadapi di bidang kelembagaan.

Informasi mengenai perangkat daerah Kota Bengkulu merupakan suatu hal yang dibutuhkan untuk memberikan informasi kepada masyarakat mengenai perangkat daerah Kota Bengkulu. Masih banyak masyarakat yang belum mengetahui mengenai perangkat daerah Kota Bengkulu seperti lokasi setiap perangkat daerah Kota Bengkulu, gedung, kepala dari setiap perangkat daerah Kota Bengkulu, dan informasi terkait yang dibutuhkan. Ini disebabkan karena tidak semuanya mengetahui pasti jalan, lokasi, dan informasi mengenai perangkat daerah Kota Bengkulu, terutama untuk orang yang baru pertama kali mendatangi ke salah satu perangkat daerah Kota Bengkulu tersebut. Dalam pencarian lokasi perangkat daerah Kota Bengkulu masih dilakukan secara manual yaitu dengan bertanya secara langsung kepada orang lain, ada juga dengan melihat peta dalam bentuk *hardcopy*, atau mencari informasi lokasi melalui internet. Namun, pencarian lokasi dalam bentuk *hardcopy* sudah jarang ditemui. Begitu juga dengan bertanya kepada orang lain, informasi yang kita dapatkan terbatas hanya kepada pengalaman orang tersebut saja. Sedangkan melakukan pencarian lewat *browsing* internet, informasi yang diberikan mengenai lokasi yang dicari juga kurang membantu.

*Augmented Reality* dapat diterapkan pada aplikasi perangkat *mobile* Android karena sistem pada *augmented reality* menganalisa secara *real-time* obyek yang ditangkap dalam kamera yang bisa diimplementasikan pada perangkat yang memiliki GPS, akselerometer, kompas, dan kamera. Menurut Mulyadi dalam Putra (2011), Android merupakan *platform* lengkap mulai dari Sistem Operasi, Aplikasi, *Tool Developing*, *Market* Aplikasi, dukungan *vendor* industri *mobile*, bahkan dukungan dari komunitas *Open*

*System*. Tentu ini merupakan keunggulan yang tidak dimiliki oleh *platform* lain. Dengan melihat perkembangan saat ini, Android telah menjadi kekuatan yang luar biasa. Pada tahun 2009, dilaporkan oleh *canalys Estimates, market smartphone* untuk Android tumbuh 1073,5% disaat *platform* lain tidak ada yang mencapai pertumbuhan 100 %.

Dari uraian di atas, maka diperlukan sebuah aplikasi berbasis android yang dapat membantu masyarakat dalam menemukan lokasi perangkat daerah kota Bengkulu. Oleh karena itu, penelitian ini berjudul “Rancang Bangun Layanan Berbasis Lokasi Dengan Penerapan *Augmented Reality* Menggunakan Metode *Markerless* Berbasis Android (Studi Kasus : Pencarian Perangkat Daerah Kota Bengkulu)”.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah bagaimana merancang dan membangun aplikasi layanan berbasis lokasi dengan penerapan *Augmented Reality* menggunakan metode *Markerless* berbasis Android (studi kasus : pencarian perangkat daerah kota Bengkulu)?

## **1.3 Batasan Masalah**

Dalam layanan berbasis lokasi dengan penerapan *Augmented Reality* menggunakan metode *Markerless* berbasis Android pada pencarian perangkat daerah kota Bengkulu ini, ditetapkan beberapa batasan sebagai berikut:

1. Lokasi perangkat daerah kota Bengkulu yang akan di cari antara lain dinas daerah (Unsur Pelaksana), Lembaga Teknis Daerah, Lembaga lain, dan Sekretariat DPRD. Adapun pola organisasi perangkat daerah Kota Bengkulu terlampir.
2. Informasi yang diperbaharui hanya pada titik koordinat lokasi perangkat daerah kota Bengkulu, dan data perangkat daerah kota Bengkulu (nama perangkat daerah Kota Bengkulu, alamat, dan kepala perangkat daerah Kota Bengkulu).
3. Gedung tiga dimensi yang ditampilkan tidak termasuk interior atau tidak dapat melihat ruangannya.

#### **1.4 Tujuan Penelitian**

Tujuan yang ingin dicapai dari penelitian ini adalah merancang dan membangun aplikasi layanan berbasis lokasi dengan penerapan *Augmented Reality* menggunakan metode *Markerless* berbasis Android pada pencarian perangkat daerah kota Bengkulu.

#### **1.5 Manfaat Penelitian**

Manfaat dari penelitian ini adalah menambah fasilitas teknologi baru berupa aplikasi android bagi pemerintah Kota Bengkulu dalam memberikan pelayanan kepada masyarakat mengenai layanan berbasis lokasi pencarian perangkat daerah Kota Bengkulu.

## **1.6 Sistematika Penulisan**

Dalam penyusunan tugas akhir ini, sistematika penulisan sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini berisikan pembahasan masalah umum yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab ini memuat landasan teori yang berfungsi sebagai sumber atau referensi dalam memahami permasalahan yang berkaitan dengan aplikasi yang akan dibangun.

### **BAB III METODE PENELITIAN**

Bab ini berisi tentang metode-metode yang digunakan dalam penelitian, seperti jenis penelitian, teknik pengumpulan data, jenis dan sumber data, metode pengembangan sistem, metode pengujian dan jadwal penelitian.

### **BAB IV ANALISIS DAN DESAIN PERANGKAT LUNAK**

Bab ini menjelaskan setiap tahapan analisis dan perancangan sistem aplikasi yang akan dibangun dalam penelitian meliputi analisis sistem dan perancangan sistem.

### **BAB V HASIL DAN PEMBAHASAN**

Bab ini merupakan bab yang berisi hasil dan pembahasan yang menguraikan hasil perancangan sistem dan implementasinya.

## BAB VI PENUTUP

Ini merupakan bab penutup yang merupakan bab terakhir yang berisi kesimpulan dan saran dari pembuatan tugas akhir sampai ke pengembangan perangkat lunak kedepannya.

## BAB II

### LANDASAN TEORI

#### 2.1 Konsep Dasar Sistem Informasi

##### 2.1.1 Sistem

Terdapat dua kelompok pendekatan di dalam mendefinisikan sistem yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Menurut Jogiyanto, suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponennya didefinisikan oleh Jogiyanto sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Jogiyanto, 2005).

Kedua kelompok definisi tersebut adalah benar dan tidak bertentangan, yang berbeda adalah cara pendekatannya. Pendekatan sistem yang merupakan kumpulan elemen-elemen atau komponen-komponen atau subsistem-subsistem merupakan definisi yang lebih luas. Definisi ini lebih banyak diterima, karena kenyataannya suatu sistem dapat terdiri dari beberapa subsistem atau sistem bagian.

##### 2.1.2 Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang

nyata yang digunakan untuk pengambilan keputusan. Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan (Jogiyanto, 2005).

### **2.1.3 Sistem Informasi**

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dengan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan (Jogiyanto,2005).

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya (Jogiyanto,2005).

## **2.2 Pengertian Layanan Berbasis Lokasi (LBS)**

Menurut Purbojati dalam M.Iqbal (2013), Layanan Berbasis Lokasi atau *Location-Based Services* merupakan sebuah layanan pada sebuah telepon genggam atau *mobile device* lain yang bergantung pada lokasi dari *device*. Dengan menggunakan *location-based service*, perangkat akan menentukan lokasinya, dan informasi ini digunakan untuk mendapatkan informasi berguna lainnya untuk *user*.

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

Terdapat dua unsur utama dari *Location Based Service* adalah:

1. *Location Manager* (API Maps):

Menyediakan perangkat bagi sumber atau *source* untuk LBS, *Application Programming Interface* (API) *Maps* menyediakan fasilitas untuk menampilkan atau memanipulasi peta.

2. *Location Providers* (API Location)

Menyediakan teknologi pencarian lokasi yang digunakan oleh perangkat. API Location berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time*.

Terdapat lima komponen pendukung utama dalam teknologi Layanan Berbasis Lokasi, antara lain:

1. *Piranti Mobile*, adalah salah satu komponen penting dalam LBS. Piranti ini berfungsi sebagai alat bantu (*tool*) bagi pengguna untuk meminta informasi. Hasil dari informasi yang diminta dapat berupa teks, suara, gambar dan lain sebagainya. *Piranti mobile* yang dapat digunakan bisa berupa PDA, *smartphone*, laptop. Selain itu, *piranti mobile* dapat juga

berfungsi sebagai alat navigasi di kendaraan seperti halnya alat navigasi berbasis GPS.

2. Jaringan Komunikasi, komponen ini berfungsi sebagai jalur penghubung yang dapat mengirimkan data-data yang dikirim oleh pengguna dari *peranti mobile*-nya untuk kemudian dikirimkan ke penyedia layanan dan kemudian hasil permintaan tersebut dikirimkan kembali oleh penyedia layanan kepada pengguna.
3. Komponen *Positioning* (Penunjuk Posisi/Lokasi), setiap layanan yang diberikan oleh penyedia layanan biasanya akan berdasarkan pada posisi pengguna yang meminta layanan tersebut. Oleh karena itu, diperlukan komponen yang berfungsi sebagai pengolah/pemroses yang akan menentukan posisi pengguna layanan saat itu. Posisi pengguna tersebut bisa didapatkan melalui jaringan komunikasi *mobile* atau juga menggunakan *Global Positioning System (GPS)*.
4. Penyedia layanan dan aplikasi, merupakan komponen LBS yang memberikan berbagai macam layanan yang bisa digunakan oleh pengguna. Sebagai contoh ketika pengguna meminta layanan agar bisa tahu posisinya saat itu, maka aplikasi dan penyedia layanan langsung memproses permintaan tersebut, mulai dari menghitung dan menentukan posisi pengguna, menemukan rute jalan, mencari data di *Yellow Pages* sesuai dengan permintaan, dan masih banyak lagi yang lainnya.
5. Penyedia data dan konten, Penyedia layanan tidak selalu menyimpan seluruh data dan informasi yang diolahnya. Karena bisa jadi berbagai

macam data dan informasi yang diolah tersebut berasal dari pengembang/pihak ketiga yang memang memiliki otoritas untuk menyimpannya. Sebagai contoh basis data geografis dan lokasi bisa saja berasal dari badan-badan milik pemerintah atau juga data-data perusahaan/bisnis/industri bisa saja berasal dari *Yellow Pages*, maupun perusahaan penyedia data lainnya.

### **2.3 Augmented Reality**

Menurut Ronald T. Azuma dalam Rachman (2013), realitas bertambah atau dikenal dengan *Augmented Reality (AR)* sebagai penggabungan benda-benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi antar benda dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata. Penggabungan benda nyata dan maya dimungkinkan dengan teknologi tampilan yang sesuai, interaktivitas dimungkinkan melalui perangkat-perangkat *input* tertentu, dan integrasi yang baik memerlukan penjejakan yang efektif.

Ada beberapa metode yang digunakan pada *Augmented Reality* yaitu *marker based tracking* dan *markerless*.

#### *1. Marker based tracking*

*Marker based tracking* adalah AR yang menggunakan *marker* atau penanda objek dua dimensi yang memiliki suatu pola yang akan dibaca komputer melalui media webcam atau kamera yang tersambung dengan komputer, biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih.

## *2. Markerless*

Dengan metode *markerless*, pengguna tidak perlu lagi mencetak sebuah *marker* untuk menampilkan elemen-elemen digital. Dalam hal ini, *marker* yang dikenali berbentuk posisi perangkat, arah, maupun lokasi.

### ***2.4 Markerless Augmented Reality***

Salah satu metode *Augmented Reality* yang saat ini sedang berkembang adalah metode *Markerless Augmented Reality*, dengan metode ini pengguna tidak perlu lagi mencetak sebuah *marker* untuk menampilkan elemen-elemen digital. Dalam hal ini, *marker* yang dikenali berbentuk posisi perangkat, arah, maupun lokasi. Ketika aplikasi AR ini mengenali tanda yang sudah dikenali, biasanya akan memvisualisasikan video maupun gambar. Total Immersion dan Qualcomm adalah salah satu perusahaan yang mengembangkan *Augmented Reality* dengan berbagai macam teknik *Markerless Tracking* diantaranya adalah *Face Tracking*, *3D Object Tracking*, *Motion Tracking* dan *GPS Based Tracking* (Akbar,2012).

#### *a. Face Tracking*

Dengan menggunakan algoritma yang mereka kembangkan, komputer dapat mengenali wajah manusia secara umum dengan cara mengenali posisi mata, hidung, dan mulut manusia, kemudian akan mengabaikan objek-objek lain di sekitarnya seperti pohon, rumah, dan benda-benda lainnya. Teknik ini pernah digunakan di Indonesia pada Pekan Raya Jakarta 2010 dan *Toy Story 3 Event*.

#### b. *3D Object Tracking*

Berbeda dengan *Face Tracking* yang hanya mengenali wajah manusia secara umum, teknik *3D Object Tracking* dapat mengenali semua bentuk benda yang ada disekitar, seperti mobil, meja, televisi, dan lain-lain.

#### c. *Motion Tracking*

Pada teknik ini komputer dapat menangkap gerakan, *Motion Tracking* telah mulai digunakan secara ekstensif untuk memproduksi film-film yang mencoba mensimulasikan gerakan. Contohnya pada film Avatar, di mana James Cameron menggunakan teknik ini untuk membuat film tersebut dan menggunakannya secara *real time*.

#### d. *GPS Based Tracking*

Pengembangan teknik ini lebih diarahkan pada *smartphone*, karena teknologi GPS dan kompas yang tertanam pada *smartphone* tersebut. Dengan memanfaatkan fitur GPS yang berfungsi sebagai penentu lokasi pengguna pada saat itu berada sehingga lokasi terdekat yang ingin dituju dapat dilihat melalui implementasi *augmented reality*.

Teknik ini berguna sebagai pemandu selayaknya fungsi GPS, namun dilengkapi dengan *marker* informasi arah yang dituju. Dalam implementasinya, teknik ini mengharuskan tersambungnya koneksi GPS dan kebutuhan paket data yang ada pada *smartphone*, karena data-data lokasi yang dimiliki GPS memiliki akses langsung dari satelit agar cepat mendeteksi wilayah yang telah dijadikan sebuah objek *marker* informasi pada *Augmented Reality*. Akses internet memiliki fungsi sebagai

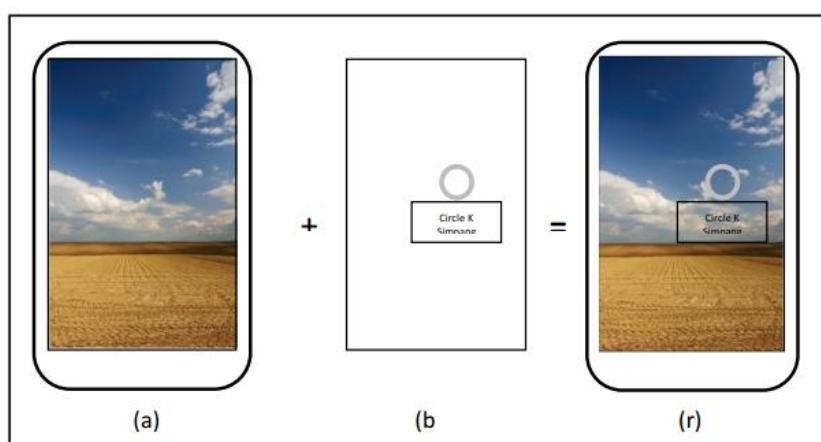
pemanggilan data-data berupa *latitude*, *longitude*, serta informasi yang mendukung setiap lokasi yang disimpan pada *server* sehingga beban ukuran aplikasi dapat diminimalisir.

Teknik GPS *based tracking* sebenarnya membutuhkan peran kompas dan akselerometer sebagai pengatur ukuran layar secara horizontal dan vertikal agar *marker* lokasi dapat dilihat ketika kamera *handset* berada posisi yang sesuai dengan lokasi tersebut. Namun ketika *handset* tidak berada dalam sudut pandang lokasi tersebut maka *marker* tersebut tidak akan tampak.

## 2.5 Prinsip Kerja *Markerless Augmented Reality*

*Augmented reality* adalah teknologi yang menggabungkan dunia maya dengan objek virtual dalam waktu nyata (*real-time*), sehingga menghasilkan suatu pandangan baru terhadap lingkungan dunia nyata. Pelacakan lokasi yang digunakan adalah pelacakan GPS untuk mencari atau memindai titik koordinat setiap lokasi yang di cari pada suatu wilayah yang sebelumnya telah terdaftar dalam *database server*. Dari setiap koordinat (*latitude* dan *longitude*) lokasi tersebut, akan di *tag* sebuah gambar dalam hal ini penanda (*marker*) pada setiap titik koordinat lokasi yang di cari yang nantinya melalui kamera akan ditampilkan dalam layar *handphone*. Untuk menampilkan titik koordinat lokasi yang dicari yang telah di *tag* dengan sebuah penanda dalam layar *handphone*, digunakan *Wikitude AR browser* (*augmented reality engine*). Wikitude AR *browser* merupakan aplikasi perangkat lunak yang bertujuan untuk mengambil, menampilkan, dan melintaskan sumber informasi melalui antarmuka *augmented reality*.

Ketika posisi keberadaan pengguna di suatu wilayah telah ditentukan, aplikasi akan memindai atau melacak titik koordinat lokasi yang dicari yang sebelumnya telah terdaftar dalam *database server*, kemudian akan mencocokkan data informasi dengan wilayah dimana pengguna berada, sehingga informasi yang ditampilkan relevan. Jadi, informasi lokasi yang dicari yang akan ditampilkan telah ditentukan, selanjutnya setiap titik koordinat lokasi yang dicari akan ditempelkan (*tag*) sebuah penanda dimana prosesnya menghasilkan objek maya (*virtual object*). Objek maya kemudian akan dilapiskan atau digabungkan dengan layar dunia nyata yang ditangkap oleh pengguna melalui kamera pada *handphone*, tahap terakhir dilakukan kalibrasi *compass* (mencocokkan orientasi posisi kamera terhadap lingkungan disekitar pengguna). Proses ini menghasilkan tampilan objek maya yang muncul di dunia nyata pada kamera. Tahap ini merupakan hasil akhir penggabungan yang akan pengguna lihat dalam layar *handphone* yang digunakan, sebuah tampilan penyisipan informasi terhadap lokasi yang dicari dalam dunia nyata.



Gambar 2.1 Prinsip Kerja *Markerless AR* : (a) Layar dunia nyata yang ditangkap kamera, (b) Layar hasil proses *augmented reality*, (c) Tampilan *Augmented Reality*: Menggabungkan layar dunia nyata dan layar *augmented reality*.

## 2.6 Android

### 2.6.1 Sejarah Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* yang berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk *smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia (Safaat, 2011).

Pada saat peluncuran perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari *Google* atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung *Google* atau dikenal sebagai *Open Handset Distribution* (OHD) (Safaat, 2011).

Android merupakan generasi baru *platform mobile* dan merupakan *platform* yang memberikan pengembang untuk melakukan sesuai dengan yang diharapkannya. Pengembang aplikasi Android diperbolehkan untuk

mendistribusikan aplikasi mereka dibawah skema lisensi apapun yang mereka inginkan. Pengembang memiliki beberapa pilihan ketika membuat aplikasi berbasis Android.

Kebanyakan pengembang menggunakan *Eclipse* yang tersedia secara bebas untuk merancang dan mengembangkan aplikasi Android. Eclipse adalah IDE yang paling popular untuk mengembangkan Android, karena memiliki Android *Plug-in* yang tersedia untuk memfasilitasi pengembangan Android. Aplikasi Android dapat dikembangkan pada sistem operasi berikut (Safaat, 2011):

1. Windows XP/Vista/Seven
2. Mac OS (Mac OS X 10.4.8 atau yang lebih baru)
3. Linux

## **2.6.2 Arsitektur Android**

Secara garis besar arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut (Safaat, 2011):

1. *Applications* dan *Widgets*

*Application* dan *Widgets* ini adalah *layer* dimana berhubungan dengan aplikasi saja, dimana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut, di *layer* inilah terdapat aplikasi inti termasuk *klien email*, program SMS, kalender, peta, browser, kontak dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

## 2. *Applications Frameworks*

*Applications Frameworks* ini adalah *layer* dimana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem Android, karena pada *layer* inilah aplikasi dapat dirancang, seperti *content-providers* yang berupa SMS dan lain sebagainya.

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

- a. *Views*
- b. *Content Provider*
- c. *Resource Manager*
- d. *Notification Manager*
- e. *Activity Manager*

## 3. *Libraries*

*Libraries* ini adalah *layer* dimana *feature-feature* Android berada, biasanya para pembuat aplikasi kebanyakan mengakses *Libraries* untuk menjalankan aplikasinya. Berjalan di atas kernel, *layer* ini meliputi berbagai *library* C/C++ inti seperti Libc dan SSL, serta:

- a. *Libraries* media untuk pemutaran media audio dan video.
- b. *Libraries* untuk manajemen tampilan.
- c. *Libraries Graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D.
- d. *Libraries SSL* dan *WebKit* terintegrasi dengan *web browser* dan *security*.

- e. *Libraries Live Webcore* mencakup modern *web browser* dengan *engine embedded web view*.
- f. *Libraries 3D* yang mencakup implementasi *Open GL ES 1.0 API's*

## 2. *Android Run Time*

*Layer* yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan implementasi linux. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Didalam *Android Run Time* dibagi menjadi dua bagian yaitu:

- a. *Core Libraries*: Aplikasi Android yang dibangun dalam bahasa java, sementara Dalvik sebagai virtual mesinya bukan virtual mesin java, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa java/c yang ditangani oleh *core libraries*.
- b. *Dalvik Virtual Machine*: Virtual mesin yang berbasis *register* yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat linux kernel untuk *threading* dan manajemen tingkat rendah.

## 3. Linux Kernel

Linux Kernel adalah *layer* dimana inti dari *operating system* dari Android itu sendiri, berisi file-file sistem yang mengatur *system processing, memory, resource, drivers*, dan sistem-sistem operasi Android lainnya.

### **2.6.3 Versi Android**

Versi android yang pernah dirilis adalah sebagai berikut (Safaat, 2011) :

#### **1. Android versi 1.1**

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan *email*.

#### **2. Android versi 1.5 (*Cupcake*)**

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan *Bluetooth A2DP*, kemampuan terhubung secara otomatis ke *headset Bluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan sistem.

#### **3. Android versi 1.6 (*Donut*)**

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol *applet VPN*. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder* dan galeri yang dintegrasikan, CDMA/EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*, kemampuan dial

kontak, teknologi *text to change speech* (tidak tersedia pada semua *handphone*) dan pengadaan resolusi VWGA.

#### 4. Android versi 2.0/2.1 (*Eclair*)

Pada 3 Desember 2009 kembali diluncurkan *smartphone* Android dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan Google Maps 3.1.2, perubahan UI dengan *browser* baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan *Bluetooth* 2.1.

#### 5. Android versi 2.2 (*Froyo: Frozen Yoghurt*)

Pada bulan Mei 2010 di luncurkan Android versi 2.2 (*Froyo: Frozen Yoghurt*) dengan 2 kali revisi, revisi pertama bulan Mei 2010 dan revisi kedua bulan Juli 2010. Android 2.2 adalah rilis minor *platform* termasuk fitur pengguna, fitur pengembang, perubahan API, dan perbaikan *bug*.

#### 6. Android Versi 2.3 (*Gingerbread*)

Android versi 2.3 diluncurkan pada desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan sebagai berikut:

1. *SIP-based VoIP*
2. *Near Field Communication (NFC)*
3. *Gyroscope dan sensor*
4. *Multiple cameras support*
5. *Mixable Audio effect*
6. *Download Manager*

## 7. Android versi 3.0/3.1 (*Honeycomb*)

Android *Honeycomb* dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet. *Honeycomb* juga mendukung *multi processor* dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Tablet pertama yang dibuat dengan menjalankan *Honeycomb* adalah Motorola Xoom.

## 8. Android versi 4.0 (*Ice Cream Sandwich*)

*Ice Cream Sandwich* telah diperkenalkan dalam acara Google I/O pada bulan Mei 2011, dan resmi dirilis pada tanggal, 19 Oktober 2011. Dengan fitur *Honeycomb* yang telah disempurnakan seperti pengunci dengan pengenal muka (*facial recognition unlock*), *control* dan pengawasan penggunaan data jaringan (*network data usage monitoring*), penggabungan kontak dengan jaringan sosial (*social networking contacts*), peningkatan kualitas fotografi, pencarian email *offline* (*offline email searching*), dan berbagi informasi menggunakan NFC (*Near Field Communication*). *Smartphone* pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

## 9. Android Versi 4.1 (*Jelly Bean*)

Android versi 2.3 diluncurkan pada Juli 2012. Pada versi terbaru ini terdapat beberapa perubahan, yaitu terjadinya peningkatan *input keyboard*, desain baru dan fitur pencarian yang sudah disempurnakan UI yang baru dan pencarian melalui *voice search* yang lebih cepat. Terdapat pula aplikasi terbaru dari Google, yaitu *Google Now*. *Google Now*

memberikan informasi yang tepat pada waktu yang terjadi sekarang. Salah satu andalannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. OS Android *Jelly Bean* 4.1 muncul pertama kali dalam produksi tablet Asus, yakni Google Nexus 7.

## **2.7 Google Maps API**

### **2.7.1 Pengertian API**

Menurut Tulach dalam Halim dkk (2011), API atau *Application Programming Interface* bukan hanya satu set *class* dan *method* atau fungsi dan *signature* yang sederhana. API yang 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.

### **2.7.2 Pengertian Google Maps API**

Seperti yang tercatat oleh Svennerberg dalam Halim dkk (2011) Google Maps API adalah API yang paling populer di internet. Pencatatan yang dilakukan pada bulan Mei tahun 2010 ini menyatakan bahwa 43% mashup (aplikasi dan situs web yang menggabungkan dua atau lebih sumber data) menggunakan Google Maps API. Beberapa tujuan dari penggunaan Google Maps API adalah untuk melihat lokasi, mencari alamat, mendapatkan petunjuk mengemudi dan lain sebagainya.

## 2.8 JSON (*Java Script Object Notation*)

JSON adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer.

Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3–Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. JSON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai

Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.

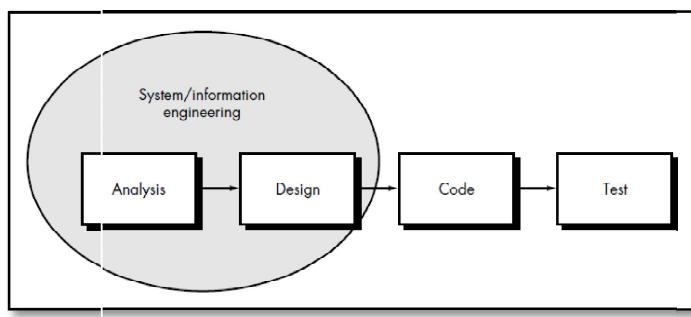
2. Daftar nilai terurutkan (*an ordered list of values*)

Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman modern mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini (Akbar,2012).

## 2.9 Metode Pengembangan Sistem

Model sekuensial linier merupakan salah satu dari metode yang digunakan untuk pengembangan sistem. Sekuensial linier sering disebut juga dengan “siklus kehidupan klasik” atau “model air terjun”. Model sekuensial linier mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak sistematik dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian dan pemeliharaan (Pressman, 2002).



Gambar 2.2 Model Sekuensial Linier (Pressman, 2002)

Pada Gambar 2.2 menggambarkan model pengembangan model sekuensial linier. Model sekuensial linier melingkupi aktivitas-aktivitas sebagai berikut (Pressman, 2002):

1. Rekayasa dan pemodelan sistem

Karena sistem merupakan bagian dari sebuah sistem yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa subset dari kebutuhan ke *software* tersebut. Pandangan sistem ini penting ketika *software* harus berhubungan dengan elemen-elemen yang lain seperti *software*, manusia, dan *database*. Rekayasa dan anasisis sistem menyangkut pengumpulan kebutuhan pada tingkat sistem dengan sejumlah kecil analisis serta desain tingkat

puncak. Rekayasa informasi mencakup juga pengumpulan kebutuhan pada tingkat bisnis strategis dan tingkat area bisnis.

## 2. Analisis kebutuhan *software*

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada *software*. Untuk memahami sifat program yang dibangun, analis harus memahami domain informasi, tingkah laku, unjuk kerja, dan interface yang diperlukan. Kebutuhan baik untuk sistem maupun *software* didokumentasikan dan dilihat lagi dengan pelanggan.

## 3. Desain

Desain *software* sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda, struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Proses desain menterjemahkan syarat/kebutuhan ke dalam sebuah representasi *software* yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode. Sebagaimana persyaratan, desain didokumentasikan dan menjadi bagian dari konfigurasi *software*.

## 4. Generasi kode

Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

## 5. Pengujian

Sekali program dibuat, pengujian program dimulai. Proses pengujian berfokus pada logika internal *software*, memastikan bahwa semua

pernyataan sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa *input* yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

## 6. Pemeliharaan

*Software* akan mengalami perubahan setelah disampaikan kepada pelanggan (pengecualian yang mungkin adalah *software* yang dilekatkan). Perubahan akan terjadi karena kesalahan-kesalahan ditentukan, karena *software* harus disesuaikan untuk mengakomodasi perubahan-perubahan di dalam lingkungan eksternalnya (contohnya perubahan yang dibutuhkan sebagai akibat dari perangkat peripheral atau sistem operasi yang baru), atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja. Pemeliharaan *software* mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.

## 2.10 *Unified Modeling Language (UML)*

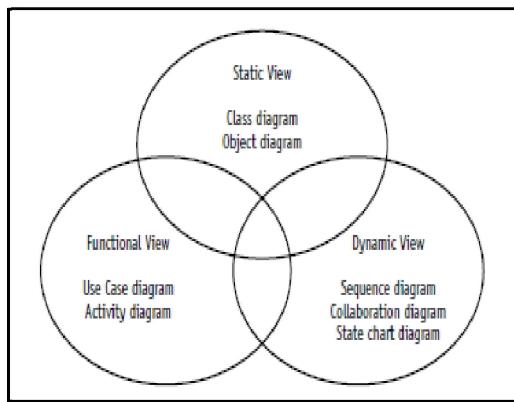
Secara umum *Unified Modeling Language (UML)* merupakan “bahasa” untuk visualisasi, spesifikasi, konstruksi, serta dokumentasi. Dalam kerangka visualisasi, para pengembang menggunakan UML sebagai suatu cara untuk mengkomunikasikan idenya kepada para pemrogram serta calon pengguna sistem/perangkat lunak. Dengan adanya “bahasa” yang bersifat standar, komunikasi perancang dengan pemrogram (lebih tepat lagi komunikasi antar anggota kelompok pengembang) serta calon pengguna diharapkan menjadi mulus (Nugroho, 2005).

Dalam kerangka spesifikasi, UML menyediakan model-model yang tepat, tidak ambigu, serta lengkap. Secara khusus, UML menspesifikasikan langkah-langkah penting dalam pengambilan keputusan analisis, perancangan, serta implementasi dalam sistem yang sangat bernuansa perangkat lunak (*software intensive system*). Dalam hal ini, UML bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman, sehingga adalah mungkin melakukan pemetaan (*mapping*) langsung dari model-model yang dibuat dengan UML ke bahasa-bahasa pemrograman berorientasi objek.

Secara umum UML diterapkan dalam pengembangan sistem atau perangkat lunak berorientasi objek sebab metodologi UML ini umumnya memiliki keunggulan-keunggulan dibawah ini (Nugroho, 2005):

1. *Uniformity*. Dengan metodologi UML (atau metodologi berorientasi objek pada umumnya), para pengembang cukup menggunakan 1 metodologi dari tahap analisis hingga perancangan. Hal ini tidak bisa dilakukan dalam metodologi pengembangan terstruktur. Dengan perkembangan masa kini ke arah aplikasi GUI (*Graphical User Interface*), UML juga memungkinkan kita merancang komponen antarmuka pengguna (*User Interface*) secara terintegrasi bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basis data.
2. *Understandability*. Dengan metodologi ini kode yang dihasilkan dapat di organisasi kedalam kelas-kelas yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami siapa pun juga.

3. *Stability*. Kode program yang dihasilkan relatif stabil sepanjang waktu sebab sangat mendekati permasalahan sesungguhnya di lapangan.
4. *Reusability*. Dengan metodologi berorientasi objek, dimungkinkan penggunaan ulang kode, sehingga pada gilirannya akan sangat mempercepat waktu pengembangan perangkat lunak (sistem informasi).



Gambar 2.3 *Three complementary views or sets of diagrams* (Pender, 2002)

Pada gambar 2.3 menggambarkan sifat komplementer dari tiga pandangan dan diagram yang membuat setiap tampilan UML terdiri atas pengelompokan diagram-diagram sistem. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model. Salah satu cara untuk mengatur diagram UML adalah dengan menggunakan *view*. *View* adalah kumpulan dari diagram yang menggambarkan aspek yang sama dari proyek yang terdiri dari *Static View*, *Dinamis View*, dan *Fungsional View* (Pender, 2002).

### 3. *Class Diagram*

Bersifat statis, diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun

bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif (Nugroho, 2005).

#### 4. *Object Diagram*

Bersifat statis, diagram ini memperlihatkan objek-objek serta relasi-relasi antar objek. Diagram objek memperlihatkan instansiasi statis dari segala sesuatu yang dijumpai pada diagram kelas (Nugroho, 2005).

#### 5. *Activity Diagram*

Bersifat dinamis, diagram ini adalah tipe khusus dari diagram *state* yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek (Nugroho, 2005).

#### 6. *Usecase Diagram*

Bersifat statis, diagram ini memperlihatkan himpunan *usecase* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Nugroho, 2005).

#### 7. *Sequence Diagram*

Bersifat dinamis, diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu (Nugroho, 2005).

#### 8. *State Chart Diagram*

Bersifat dinamis, diagram ini memperlihatkan *state-state* pada sistem, memuat *state*, transisi, *event*, serta aktivitas. Diagram ini terutama penting

untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem reaktif.

#### 9. *Collaboration Diagram*

Bersifat dinamis, diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan (*message*).

#### 10. *Deployment Diagram*

*Deployment diagram* memodelkan bagian-bagian perangkat lunak sistem ke perangkat keras yang akan mengeksekusinya.

### 2.11 Pengujian Sistem

Pengujian pada dasarnya adalah menemukan serta menghilangkan ‘bug’ (kesalahan-kesalahan) yang ada di sistem/perangkat lunak. Kesalahan-kesalahan itu dapat diakibatkan beberapa hal utama, antara lain kesalahan saat penentuan spesifikasi sistem, kesalahan saat melakukan analisis permasalahan, kesalahan saat perancangan, serta kesalahan saat implementasi (Nugroho, 2005).

#### 2.11.1 Teknik Pengujian Sistem

Menurut Bahrawi dalam Nugroho (2005), konsep kualitas sangat penting demi kepuasan pengguna (juga pengembang). Untuk mencapai kualitas yang diharapkan dari sistem yang kita kembangkan pada umumnya ada beberapa strategi pengujian yang dapat dilakukan. Strategi-strategi itu adalah :

1. *Black-Box Testing*. Pada pengujian ini kita tidak perlu tahu apa sesungguhnya terjadi pada sistem/perangkat lunak. Yang kita uji adalah

masukkan serta keluarannya. Artinya, dengan berbagai masukkan yang kita berikan, apakah sistem memberikan keluaran seperti yang kita harapkan?

2. *White-Box Testing*. Pengujian jenis ini mengasumsikan bahwa spesifikasi logika adalah penting dan perlu dilakukan pengujian untuk menjamin apakah sistem berfungsi dengan baik. Tujuan utama dari strategi pengujian ini adalah pengujian berbasis kesalahan.
3. *Top-Down Testing*. Pengujian ini berasumsi bahwa logika utama atau interaksi antar objek perlu diuji lebih lanjut. Strategi ini seringkali dapat mendeteksi cacat/kesalahan/kekurangan yang serius. Pendekatan ini sesuai dengan strategi pengujian berbasis skenario.
4. *Bottom-Up Testing*. Strategi ini mulai dengan rincian sistem kemudian beranjak ke peringkat yang lebih tinggi. Dalam metodologi berorientasi objek, kita mulai dengan menguji metoda-metoda dalam kelas, menguji kelas-kelas serta interaksi antar kelas, dan selanjutnya hingga pada peringkat yang paling tinggi.

### **2.11.2 Pengukuran Tingkat Kepuasan Pengguna**

Mengenai kepuasan pengguna kita masih harus meninjau seberapa jauh sistem yang kita kembangkan memuaskan pengguna. Beberapa cara yang dapat ditempuh untuk mengetahui kepuasan pengguna adalah:

1. **Wawancara**. Wawancara diperlukan untuk mengetahui tingkat kepuasan pengguna dengan mengajukan beberapa pertanyaan.

2. **Kuestioner.** Kuestioner merupakan daftar pertanyaan yang diajukan pada seorang responden untuk mencari jawaban dari permasalahan yang diteliti (Hasibuan, 2007).
3. **Pengamatan Langsung.** Pengamatan langsung bisa juga dilakukan untuk mengetahui apakah sistem yang kita kembangkan sesuai dengan kebutuhan serta harapan pengguna.

## 2.12 Penelitian Terkait

Hasil penelitian yang relevan dengan penelitian yang peneliti lakukan adalah:

1. Pencarian Tempat Kos Dengan Teknologi *Augmented Reality* Berbasis *Smartphone* Android (Hanif,2013). Program Studi Teknik Informatika, Universitas Islam Negeri Sunan Kalijaga Yogyakarta. Aplikasi ini mencari tempat tinggal sementara atau kos di Yogyakarta dengan memanfaatkan *Google Maps* dan mempadukan dengan teknologi Augmented Reality. Aplikasi ini menggunakan metode *markerless*. Kelebihan pada aplikasi menampilkan informasi lebih detail dan data bisa di *update* melalui server. Kekurangan pada aplikasi ini yaitu tidak ada fitur pencarian terdekat lokasi yang dicari,tidak adanya rute lokasi pengguna sekarang dengan lokasi yang di tuju.
2. Sistem Pencarian Fasilitas Umum Terdekat Menggunakan *Augmented Reality* Dengan *Minimum Spanning Tree* (Riesvicky dkk,2012) . Jurusan Sistem Informasi, Fakultas Ilmu Komputer, Universitas Gunadarma. Penelitian ini membangun sebuah aplikasi yang

membantu pengguna untuk mengenali daerah di sekitarnya, mencari fasilitas umum dengan menggunakan metode *minimum spanning tree* sebagai algoritma penunjang pencarian informasi tersebut. Pada aplikasi ini memiliki kelebihan yaitu dapat mencari fasilitas umum terdekat, mendapatkan lokasi saat ini, melakukan *tracking*, mendapatkan penunjuk arah, simpan dan *load route*, serta menyimpan tempat atau fasilitas umum ketika berada ditempat baru. Kekurangannya yaitu objek yang ditampilkan masih dalam bentuk dua dimensi.

3. *Markerless Augmented Reality* Pada Perangkat Android (Rizki,2011). Jurusan Teknik Elektro, Fakultas Teknik Industri, ITS Surabaya. Pada aplikasi ini menggunakan teknologi *markerless augmented reality* yang mengimplementasikan image target dan multi target sebagai obyek pelacakan serta fitur interaktif *virtual button* pada perangkat Android. Sebuah gambar dideteksi menggunakan kamera dari perangkat Android secara *real time*, lalu memunculkan informasi lain secara *virtual* pada layar perangkat tersebut. Kelebihan pada aplikasi ini objek yang ditampilkan sudah dalam bentuk 3 dimensi. Sedangkan kekurangan pada aplikasi ini adalah kurang sempurnanya pelacakan yang telah dilakukan dari penggunaan target dengan parameter fitur alami yang berjumlah kurang banyak, kurang tersebar merata, serta dengan pola kurang yang unik dan berulang.
4. Implementasi *Augmented Reality* Memanfaatkan Sensor Akselerometer, Kompas dan GPS Pada Penentuan Lokasi Masjid

Berbasis Android (Hendrianto dkk,2011). Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Surabaya. Pada aplikasi ini mencari informasi lokasi masjid dengan memanfaatkan sensor yang terintegrasi dalam *device* berupa akselerometer, kompas dan GPS. Aplikasi ini menggunakan metode *markerless*. Kelebihan pada aplikasi ini mendapatkan koordinat masjid yang lebih akurat dengan melakukan perhitungan sudut azimuth dan sudut inklinasi. Kekurangan pada aplikasi ini yaitu pada marker menampilkan objek dua dimensi dan tidak adanya rute lokasi pengguna sekarang dengan lokasi yang di tuju.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Jenis Penelitian

Jenis penelitian yang akan dilakukan adalah penelitian terapan (*applied research*). Penelitian ini bertujuan merancang dan membangun aplikasi layanan berbasis lokasi dengan penerapan *Augmented Reality* menggunakan metode *markerless* berbasis android (studi kasus: pencarian perangkat daerah Kota Bengkulu).

#### 3.2 Sarana Pendukung dan Sarana Pengujian

Dalam penelitian ini, sarana pendukung yang dibutuhkan berupa perangkat keras (*hardware*) dan perangkat lunak (*software*).

1. Perangkat keras (*hardware*)
  - a. Laptop asus dengan spesifikasi Intel Core i3-370M processor, RAM 2GB DDR3,
  - b. *Keyboard* dan *mouse*,
  - c. Modem,
  - d. GPS (Global Positioning System)
2. Perangkat lunak (*software*)
  - a. Sistem operasi : Windows Seven,
  - b. Bahasa Pemrograman : PHP dan Java Development Kit (JDK) 7,
  - c. Eclipse 3.5,
  - d. Google Chrome,
  - e. MySQL,

f. Autodesk 3ds Max 8.0

g. Wikitude 3D Encoder

Sedangkan sarana pengujian yang dibutuhkan berupa perangkat keras (*hardware*) dan perangkat lunak (*software*).

1. Perangkat keras (*hardware*)

a. *Smartphone* Android Samsung Galaxy Grand seri GT-I9082.

3. Perangkat lunak (*software*)

a. *Emulator* pada eclipse 3.5.

### 3.3 Jenis dan Sumber Data

Jenis data yang dibutuhkan dan digunakan dalam penelitian ini adalah data kualitatif. Data kualitatif merupakan jenis data yang dapat diukur secara langsung atau dapat dihitung. Data-data yang dikumpulkan dalam penelitian ini berasal dari 2 (dua) sumber data yaitu:

1. Data Sekunder

Data sekunder yang dibutuhkan berupa data perangkat daerah Kota Bengkulu dari Kantor Walikota Bengkulu bagian Organisasi Kepegawaian antara lain nama-nama perangkat daerah Kota Bengkulu, alamat, nama kepala setiap perangkat daerah Kota Bengkulu, dan informasi mengenai setiap perangkat daerah Kota Bengkulu.

2. Data Primer

Data primer pada penelitian ini adalah titik koordinat setiap perangkat daerah Kota Bengkulu. Peneliti melakukan observasi secara langsung ke tempat lokasi dengan menggunakan alat GPS untuk mendapatkan data koordinat tersebut. Data koordinat diambil di halaman setiap perangkat

daerah Kota Bengkulu tepatnya dekat dengan pintu masuk utama perangkat daerah Kota Bengkulu. Data koordinat setiap lokasi dapat dilihat pada lampiran.

### **3.4 Teknik Pengumpulan Data**

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah:

#### 1. Survei Lapangan

Peneliti mengamati langsung lokasi untuk mengumpulkan data *latitude* dan *longitude* setiap lokasi dengan GPS. Data koordinat diambil di halaman perangkat daerah Kota Bengkulu tepatnya di dekat pintu masuk utama perangkat daerah Kota Bengkulu.

#### 2. Studi Pustaka

Studi pustaka dilakukan dengan mengumpulkan data dan informasi yang dijadikan sebagai acuan perancangan Aplikasi layanan berbasis lokasi dengan penerapan *Augmented Reality* menggunakan metode *markerless* berbasis android pada pencarian perangkat daerah Kota Bengkulu ini. Data dan informasi tersebut diantaranya: tutorial pemrograman android, *Augmented Reality*, desain diagram UML, penggunaan *google maps*, data perangkat daerah Kota Bengkulu.

### **3.5 Metode Perancangan dan Pengembangan Sistem**

Metode yang akan digunakan dalam pengembangan sistem adalah model pengembangan sistem *sekuensial linier* yang bersifat sistematis dan berurutan. Adapun penjelasan tahap-tahap model *sekuensial linier* dalam penelitian ini adalah sebagai berikut:

## 1. Rekayasa dan Pemodelan sistem

Langkah awal yang dilakukan dalam penelitian ini adalah mengidentifikasi masalah yang ada untuk dijadikan suatu sistem sebagai solusi. Peneliti melakukan identifikasi masalah ke kantor Walikota Bengkulu untuk mengidentifikasi permasalahan yang ada. Dalam pencarian lokasi perangkat daerah kota Bengkulu, masyarakat masih mengalami kesulitan dalam menemukan lokasi yang dicari. Pencarian lokasi masih dilakukan secara manual yaitu dengan bertanya secara langsung kepada orang lain, ada juga dengan melihat peta dalam bentuk *hardcopy*, atau mencari informasi lokasi melalui internet. Namun, pencarian informasi lokasi dalam bentuk *hardcopy* sudah jarang ditemui. Begitu juga dengan bertanya kepada orang lain, informasi yang kita dapatkan terbatas hanya kepada pengalaman orang tersebut saja. Sedangkan melakukan pencarian lewat *browsing* internet, informasi yang diberikan mengenai lokasi yang dicari juga kurang membantu. Oleh karena itu, diperlukan suatu aplikasi yang dapat memberikan informasi kepada masyarakat mengenai lokasi perangkat daerah kota Bengkulu.

## 2. Analisis kebutuhan sistem

Adapun analisis kebutuhan sistem yang diperlukan antara lain:

- a. Input : nama perangkat daerah Kota Bengkulu, alamat, nama kepala perangkat daerah Kota Bengkulu, informasi mengenai setiap perangkat daerah Kota Bengkulu, data *longitude*, dan data *latitude*.
- b. Ouput : nama perangkat daerah kota Bengkulu, alamat, nama kepala perangkat daerah Kota Bengkulu, informasi mengenai

perangkat daerah Kota Bengkulu, dan gedung perangkat daerah kota Bengkulu dalam bentuk 3 dimensi.

### 3. Desain Sistem

Perancangan sistem dikerjakan setelah tahap analisis dan definisi kebutuhan selesai dikumpulkan secara lengkap. Kegiatan yang dilakukan di tahap ini adalah menerjemahkan analisis ke dalam bentuk rancangan antarmuka (*interface*), rancangan basis data, dan diagram-diagram *unified modeling language* (UML) yang membantu dalam penganalisan sistem. Tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilan aplikasi layanan berbasis lokasi pada pencarian perangkat daerah Kota Bengkulu yang akan dibangun.

### 4. Generasi Kode

Hasil perancangan sistem akan diubah menjadi bentuk yang dimengerti oleh mesin yaitu ke dalam bahasa pemrograman yang telah ditentukan melalui proses penulisan program (*coding*). Dalam penelitian ini, bahasa pemrograman yang digunakan adalah bahasa pemrograman Java dan platform yang digunakan yaitu Eclipse IDE Galileo 3.5, serta PHP.

### 5. Integrasi dan Pengujian Sistem

Sistem yang sudah dibangun akan dilakukan pengujian untuk melihat apakah sistem tersebut sesuai dengan perencanaan dan perancangan. Pada penelitian ini akan dilakukan dengan menggunakan *Black-Box* dan *White-Box* sebagai metode pengujian sistem.

## 6. Operasi dan Pemeliharaan

Tahap ini adalah tahap akhir pengembangan dan implementasi sistem yaitu pengoperasian sistem secara nyata. Namun dalam pengoperasianya tetap dibutuhkan dukungan agar sistem dapat digunakan dalam jangka panjang dengan melakukan pemeliharaan sistem. Pemeliharaan sistem dilakukan bukan hanya sekedar proses memperbaiki kesalahan program tetapi proses yang memiliki karakteristik memperbaiki kesalahan yang tidak ditemukan pada tahapan sebelumnya atau menambahkan fungsi baru yang belum ada pada program tersebut.

### 3.6 Metode Pengujian

Setiap produk perangkat lunak dapat diuji melalui pengujian pendekatan, yang pertama disebut sebagai *black-box testing* dan kedua disebut sebagai *white-box testing*. Ketika perangkat lunak komputer sudah dipertimbangkan maka *black-box testing* dilakukan untuk menguji antarmuka perangkat lunak. *Input* dan *output* dengan benar diterima dengan proses produksi yang benar pula dan bahwa integritas informasi eksternal (misalnya basis data) tetap terjaga. *Black-box testing* mengkaji beberapa aspek dari sistem tanpa memperhatikan struktur logika internal perangkat lunak. Hal yang akan diuji pada percobaan *black-box* antara lain:

1. Pengujian fungsional sistem dari aplikasi layanan berbasis lokasi pencarian perangkat daerah Kota Bengkulu.
2. Pengujian kebenaran aplikasi dan pemberian informasi mengenai pencarian perangkat daerah Kota Bengkulu.

3. Pengujian performa sistem dari aplikasi layanan berbasis lokasi pencarian perangkat daerah Kota Bengkulu pada OS *Android*.

Penggunaan *white-box testing* dan pengembang perangkat lunak dapat memperoleh *test case* bahwa jaminan semua jalur independen dalam sebuah modul telah di eksekusi minimal sekali, melaksanakan semua keputusan logis pada sisi benar dan salah sistem, mengeksekusi semua *loop* pada batas-batas sistem dan dalam batas-batas operasional sistem dan melaksanakan data internal struktur untuk validitasnya.

Hal yang akan diuji pada percobaan *white-box* antara lain:

1. Pengujian antarmuka aplikasi layanan berbasis lokasi pencarian perangkat daerah Kota Bengkulu,
2. Pengujian *method* dan *attribute* aplikasi layanan berbasis pencarian lokasi perangkat daerah Kota Bengkulu.

### **3.7 Metode Uji Kelayakan Sistem**

Uji Kelayakan dilakukan untuk mendapatkan penilaian langsung terhadap sistem yang dihasilkan. Target dari pengujian kelayakan sistem ini adalah responden (calon pemakai sistem). Adapun tahapan dari uji kelayakan ini adalah:

1. Kuisioner

Kuestioner merupakan daftar pertanyaan yang diajukan pada seorang responden untuk mencari jawaban dari permasalahan yang diteliti (Hasibuan, 2007).

## 2. Tabulasi Data.

Proses perhitungan data angket menggunakan skala *likert*. Sebelum melakukan perhitungan dengan menggunakan skala *likert*, maka terlebih dahulu dicari interval kelas dengan Persamaan (3.1):

$$i = \frac{m-n}{k} \dots\dots (\text{Persamaan 3.1})$$

Keterangan: i = Interval kelas                    m= angka tertinggi skor

k= banyak kelas                                  n= angka terrendah skor

Skala *likert* adalah perhitungan skor pada tiap-tiap interval dari pernyataan yang diberikan ke responden. Hasil dari proses perhitungan disajikan dalam bentuk tabel. Sehingga didapatkan nilai uji kelayakan terhadap sistem (Hasibuan, 2007).

### 3.8 Jadwal Penelitian

Berikut ini adalah jadwal penelitian yang dilaksanakan:

Tabel 3.1 Jadwal Penelitian yang Akan Dilaksanakan

No	Kegiatan	Feb-14				Mar-14				April-14				Mei-14				Juni-13			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1.	Identifikasi Masalah																				
	Latar belakang penelitian																				
	Tujuan dan ruang lingkup penelitian																				
2.	Definisi kebutuhan																				
	Pengumpulan Data																				
	Analisis data yang dibutuhkan																				
4.	Perancangan sistem																				
5.	<i>Coding</i>																				
6.	Pengujian dan evaluasi																				
7.	Analisis hasil																				
8.	Pembuatan laporan																				