



**TUGAS AKHIR**

**SISTEM INFORMASI GEOGRAFIS LOKASI HOTEL DI KOTA  
PADANG PANJANG BERBASIS ANDROID**

*Diajukan Kepada Jurusan Manajemen Informatika D.III  
Sebagai Salah Satu Syarat Memperoleh Gelar Ahli Madya (A.Md)  
Dalam Ilmu Manajemen Informatika*

**DIVIKA ROZALIA**

**NIM. 14 205 031**

**JURUSAN MANAJEMEN INFORMATIKA FAKULTAS EKONOMI  
DAN BISNIS ISLAM INSTITUT AGAMA ISLAN NEGERI (IAIN)  
BATUSANGKAR**

**2018**

## SURAT PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini :

Nama : DIVIKA ROZALIA  
NIM : 14 205 031  
Tempat / Tanggal Lahir : Titian Panjang / 01 Februari 1995  
Fakultas : Ekonomi Dan Bisnis Islam  
Jurusan : Manajemen Informatika

Menyatakan dengan sesungguhnya bahwa Tugas Akhir saya yang berjudul **“SISTEM INFORMASI GEOGRAFIS LOKASI HOTEL DI KOTA PADANG PANJANG BERBASIS ANDROID “** adalah benar karya saya sendiri bukan plagiat kecuali yang dicantumkan sumbernya.

Apabila di kemudian hari terbukti bahwa karya ilmiah ini plagiat, maka saya bersedia menerima sanksi sesuai dengan ketentuan perundang-undangan yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya untuk digunakan sebagaimana mestinya.

Batusangkar, 15 Februari 2018  
Saya yang Menyatakan



**DIVIKA ROZALIA**  
**Nim.14 205 031**

## PERSETUJUAN PEMBIMBING

Pembimbing Penulis Tugas Akhir atas Nama : **DIVIKA ROZALIA**, Nim : **14 205 031** dengan Judul, "**SISTEM INFORMASI GEOGRAFIS LOKASI HOTEL DI KOTA PADANG PANJANG BERBASIS ANDROID**" memandang bahwa Tugas Akhir yang bersangkutan telah memenuhi persyaratan ilmiah dan dapat disetujui untuk dilanjutkan ke Sidang Munaqasyah.


Dengan persetujuan ini diberikan untuk dapat dipergunakan sebagaimana mestinya.

Batusangkar, Februari 2018

**Ketua Jurusan  
Manajemen Informatika,**


**Pembimbing,**

  
**Iswandi, M.Kom**  
NIP. 19700510 200312 1 004

  
**Lita Sari Muchlis, M.Kom**  
NIP.19780122 200801 2 017

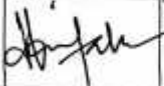
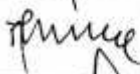
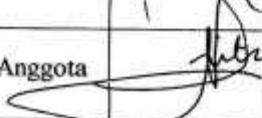
Mengetahui,

**Dekan Fakultas Ekonomi dan Bisnis Islam  
Insitut Agama Islam Negeri Batusangkar**

  
**Dr. H. Atsani, S.H., M.Hum**  
NIP. 19750303 199903 1 004

### PENGESAHAN TIM PENGUJI


Tugas Akhir yang berjudul "SISTEM INFORMASI GEOGRAFIS LOKASI HOTEL DI KOTA PADANG BERBASIS ANDROID" oleh **DIVIKA ROZALIA**, NIM. 14 205 031, telah diajukan pada sidang munaqasyah Institut Agama Islam Negeri (IAIN) Batusangkar, Senin 19 Februari 2018 dan dinyatakan telah diterima sebagai salah satu syarat untuk mencapai gelar Ahli Madya Program Diploma III (D.III) Manajemen Informatika.

No.	Nama Penguji	Jabatan	Tanda Tangan	Tanggal
1.	Lita Sari Muchlis, M.Kom NIP. 19780122 200801 2 017	Ketua Sidang		27/02 2018
2.	Iswandi, M.Kom NIP. 19700510 200312 1 004	Anggota		27/2-18
3.	Fitra Kasma Putra, M.Kom NIP. 19850207 201503 1 004	Anggota		26/2 - 2018

Mengetahui,

Dekan Fakultas Ekonomi dan Bisnis Islam  
IAIN Batusangkar



  
Dr. Ulya Atsani, S.H., M.Hum  
NIP. 19750303 199903 1 004

## HALAMAN PERSEMBAHAN



*Sesungguhnya sesudah kesulitan itu ada kemudahan.  
Maka apabila kamu Telah selesai ( dari suatu urusan),  
kerjakanlah dengan sungguh-sungguh (urusan) yang lain.  
(Q.S. Alam Nasyrah: 6-7)*

*“Dia memberikan hikmah (ilmu yang berguna)  
kepada siapa yang dikehendaki-Nya.  
Barang siapa yang mendapat hikmah itu  
Sesungguhnya ia telah mendapat kebajikan yang banyak.  
Dan tiadalah yang menerima peringatan  
melainkan orang- orang yang berakal”.  
(Q.S. Al-Baqarah: 269)*

*“...kaki yang akan berjalan lebih jauh,  
tangan yang akan berbuat lebih banyak,  
mata yang akan menatap lebih lama,  
leher yang akan lebih sering melihat ke atas,  
lapisan tekad yang seribu kali lebih keras dari baja,  
dan hati yang akan bekerja lebih keras,  
serta mulut yang akan selalu berdoa...” - 5cm.*

*Yaa Allah.....  
Syukur Alhamdulillah.....  
Kini aku tersenyum dalam iradat-mu  
Kau jadikan diriku seperti apa yang aku pikirkan  
Sepahit apapun itu aku tau itulah caramu memuliakan aku.*

*Ku persembahkan karya sederhana ini kepada Ayah dan Ibu  
tercinta...*

*Hari ini setitik kesuksesan telah ku raih  
Awal untukku memulai kehidupan yang sebenarnya  
Dipenuhi hempasan beban mental yang kuat  
Menuntutku untuk harus tetap berdiri  
Seperti batu karang yang tak henti diterjang ombak  
Dalam rangkaian do'a, ku wujudkan mimpi  
Dalam setiap langkah, ku gapai cita-cita*

*Untukmu Ayahanda tercinta...*

*Ribuan terimakasih atas segala do'a dan pengorbananmu  
Panas terik dan hujan badai yang tak pernah menghentikanmu  
Kasih yang kau berikan menguatkanmu  
Sayang dalam candamu menegarkanku  
Dari mu kuraih kesabaran dan kebaikan ini  
Dalam dirimu kutemukan indahnya kesederhanaan  
Kaulah yang mengerti akan hati saat yang lain tak memahami  
Karna cinta mu terpancar dalam diam.*

*Ibunda tersayang...*

*Tak henti ku ucap terimakasih atas kasih sayang dan do'amumu  
Tangan mulusmu telah mengangkatku menjadi pribadi yang ikhlas  
Lembut hatimu menenangkan jiwaku  
Teduhnya pandanganmu adalah penyejuk hatiku  
Kaulah yang tetap ada saat yang lain meninggalkanku  
Surga kecilku adalah senyumanmu ibu.*

*Terimalah karya kecilku ini  
Sebagai tanda baktiku kepada Ayah dan Ibu  
Jika kemarin adalah asa yang tertunda,  
Sekarang perjuangan dan esok adalah pengabdian.*

*Ttd. Anakmu*

*Untuk Kakak tersayang dan Adik terbaik...  
Trimakasih atas segala pengorbananmu selama ini  
Baik dukungan moril ataupun materil  
Nasehat yang selalu kalian tuturkan dalam perjalanan menuju  
pencapaian akhir study ini.  
Sekalipun aku tak selalu bercerita banyak padamu  
Sekalipun aku terus-menerus hanya mengeluh padamu  
Hanya terkadang ungkapkan segala kecamuk rasaku  
Terima kasih, karena membuatku lebih baik.  
Rentetan do'a terbaik tak henti kuucap untuk kebahagiaanmu  
Aku sungguh bersyukur memiliki kalian semua  
Meski kepribadian dan cita-cita kita jelas berbeda  
Trimakasih Kakakku **Ovi Aprilia Amd. Kep** dan Adikku **Novaldo**  
Serta Adikku **Fakhrani Zilvayana** dan **Fakhreïn Alzikra**  
Kupersembahkan keberhasilan ini untuk kalian...*

*“When we pray, God hears more than we say,  
answers more than we ask  
gives more than we imagine  
In His own time and in His own way”*

....*Special Thanks*....

*Terimakasih kepada **Ibuk Lita Sari Muchlis, M.Kom** yang telah meluangkan waktunya untuk bimbingan, sehingga saya bisa menyelesaikan Tugas Akhir ini. Dan juga terimakasih banyak kepada seluruh dosen Manajemen Informatika atas ilmu yang bermanfaat yang telah diberikan selama perkuliahan.*

*Buat teman-teman MI- B atas Kebersamaan yang tak akan terlupa dan tak akan terulang lagi. Persahabatan yang terukir lewat lawakan, bulian dan sindiran. Kadang terselip rasa marah pada kalian, namun takkan berubah dendam karna begitu banyak hal konyol yang kalian berikan ^^ Selamat berkarir sobat, semoga kelak kita akan bertemu lagi dihari dan waktu yang tepat.  
6 semester plus 1 semester jilid 2 :D*

*Buat teman-teman satu bimbingan  
Terimakasih atas dukungannya,  
lelah dan letih selama bimbingan kita lalu bersama-sama.  
Selamat atas gelarnya ( A.Md juo jadinya).*

*Rekan-rekan MI'14 yang tak mungkin disebutkan satu per satu  
Terimakasih yang sedalam-dalamnya atas semangat dan dukungan dalam penyelesaian tugas akhir ini.*

**Never Give Up**



## ABSTRAK

**Judul Tugas Akhir** : **SISTEM INFORMASI GEOGRAFIS  
LOKASI HOTEL DI KOTA PADANG  
PANJANG BERBASIS ANDROID**

**Nama Mahasiswa** : **Divika Rozalia**

**Nomor Induk Mahasiswa** : **14 205 031**

**Jurusan** : **Manajemen Informatika**

**Dosen Pembimbing** : **Lita Sari Muchlis, M.Kom**

Perkembangan teknologi informasi juga dapat dirasakan pada Sistem Informasi Geografis (SIG). Terdapat sekitar 29 buah hotel/penginapan yang terdapat di Kota Padang Panjang. Sebagian besar masyarakat terutama yang berasal dari luar Kota Padang Panjang yang memiliki keperluan dengan hotel tertentu biasanya mengetahui lokasi dan informasinya hanya dari brosur, peta konvensional dan sebagainya. Penggunaan peta digital Google Map untuk membangun Sistem Informasi Geografis (SIG) dapat menjadi salah satu sarana yang efektif untuk penyebaran informasi hotel di Kota Padang Panjang.

Perancangan aplikasi lokasi hotel di Kota Padang Panjang berbasis android ini menggunakan visualisasi pemodelan UML (*Unified Modelling Language*) dengan menggunakan beberapa diagram visualisasi yang berorientasikan objek.

Perancangan aplikasi lokasi hotel di Kota Padang Panjang berbasis android ini menghasilkan sebuah sistem informasi geografis yang dapat memberikan informasi hotel memudahkan masyarakat yang memiliki keperluan dengan hotel tertentu.

**Kata kunci:** Sistem Informasi Geografis, *Unified Modelling Language*, Hotel, Android.

## KATA PENGANTAR



Puji syukur penulis haturkan atas kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya sehingga pembuatan tugas akhir ini dapat terselesaikan. Shalawat dan salam buat junjungan umat, yakni Nabi Muhammad SAW, yang telah membawa manusia dari alam kebodohan ke alam yang penuh dengan ilmu pengetahuan seperti kita rasakan seperti sekarang ini.

Dalam penulisan tugas akhir ini penulis banyak mendapat bimbingan dan bantuan baik moril maupun materil dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. H. Kasmuri, MA selaku Rektor IAIN Batusangkar.
2. Bapak Dr. Ulya Atsani, S.H., M.Hum., selaku Dekan Fakultas Ekonomi dan Bisnis Islam.
3. Bapak Iswandi, M.Kom selaku ketua Jurusan Manajemen Informatika IAIN Batusangkar.
4. Ibu Lita Sari Muchlis, M.Kom sebagai pembimbing Tugas Akhir sekaligus Penasehat Akademik yang telah memberikan bimbingan dan arahan kepada penulis dalam pembuatan tugas akhir ini.
5. Kepala Dinas Pariwisata Hendri Fauzan, AP.M.Si yang telah membantu penulis selama mengumpulkan data yang diperlukan dalam penyusunan tugas akhir ini.
6. Ibunda tercinta Rosmaniar dan Ayahanda tercinta M.nur yang telah banyak memberikan dorongan, motivasi, semangat, bantuan moril maupun materil, nasehat serta doa yang tiada henti-hentinya.

7. Kakak tercinta Ovi Aprilia M dan semua sanak famili yang telah banyak memberikan semangat dan dorongan kepada penulis hingga penulis bisa menyelesaikan tugas akhir ini.
8. Teman-teman seperjuangan angkatan (2014) yang telah memberikan saran-saran yang bermanfaat dalam menyelesaikan tugas akhir ini.
9. Pihak-pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan motivasi dan semangat serta sumbangan pemikiranya kepada penulis sehingga selesainya tugas akhir ini.

Penulis sadar bahwasanya tugas akhir ini jauh dari kesempurnaan, oleh karena itu kritikan dan saran yang sifatnya membangun sangat penulis harapkan untuk kesempurnaan tugas akhir ini. Penulis juga berharap semoga penulisan tugas akhir ini memberikan manfaat kepada kita semua. Amin...

Akhirnya kepada Allah SWT jualah penulis bermohon dan bersujud semoga keikhlasan yang diberikan akan dibalas-Nya. *Amin Ya Robbal'alamin*.

Batusangkar, Februari 2018

Penulis

## DAFTAR ISI

<b>ABSTRAK .....</b>	<b>2</b>
<b>KATA PENGANTAR.....</b>	<b>ii</b>
<b>DAFTAR ISI.....</b>	<b>iv</b>
<b>DAFTAR TABEL .....</b>	<b>vii</b>
<b>DAFTAR GAMBAR.....</b>	<b>viii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
A. Latar Belakang .....	1
B. Identifikasi Masalah.....	2
C. Rumusan Masalah.....	2
D. Batasan Masalah .....	3
E. Tujuan .....	3
F. Manfaat Penelitian .....	3
G. Metodologi Penelitian.....	3
H. Sistematika Penulisan .....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>5</b>
A. Kota Padang Panjang .....	5
B. Hotel.....	5
C. Sistem Informasi Geografis (SIG) .....	7
1. Pengertian Sistem Informasi Geografis .....	7
2. Disiplin Ilmu Pendukung SIG.....	8
3. Perangkat Lunak SIG.....	9
4. Keuntungan menggunakan SIG .....	10
5. Aplikasi-aplikasi SIG.....	11
6. Subsistem SIG.....	14

7.	Komponen SIG .....	16
D.	Tool Pengembangan Sistem.....	19
1.	Android .....	19
2.	Java .....	20
3.	SDK (Software Development Kit).....	20
4.	ADT (Android Development Tools).....	21
5.	Google Maps.....	21
6.	Android Studio.....	23
7.	PHP (Hypertext Preprocessor).....	30
8.	<i>Xampp</i> .....	31
E.	Desain User Interface.....	32
F.	Unified Modelling Language (UML) .....	33
1.	Use Case Diagram .....	33
2.	Sequence Diagram .....	36
3.	Activity Diagram .....	36
4.	Class Diagram.....	37
<b>BAB III ANALISIS DAN PERANCANGAN SISTEM.....</b>		<b>40</b>
A.	Analisis Masalah.....	40
B.	Analisis Aplikasi.....	40
C.	Analisis Tools .....	41
D.	Perancangan Sistem .....	42
1.	Use Case Diagram .....	42
2.	Sequence Diagram .....	43
3.	Activity Diagram .....	45
4.	Class Diagram.....	46

E.	Struktur Program.....	47
F.	Desain Output Sistem .....	48
1.	Desain Form Awal .....	48
2.	Desain Halaman Utama .....	48
3.	Desain Menu <i>Drawer</i> .....	49
4.	Desain Mencari Hotel pada Menu Utama.....	49
5.	Desain Lihat Peta pada Menu Utama .....	50
6.	Desain Menu Tentang.....	51
G.	Desain Input .....	51
1.	Input Data Hotel.....	51
2.	Input Berita .....	52
3.	Input Kamar .....	53
H.	Struktur Tabel .....	53
1.	Tabel Hotel .....	53
2.	Tabel Berita.....	54
3.	Tabel Berita.....	54
<b>BAB IV PENUTUP .....</b>		<b>55</b>
A.	Kesimpulan .....	55
B.	Saran .....	55

## **DAFTAR PUSTAKA**

## **STATECHART DIAGRAM**

## **LISTING PROGRAM**

## DAFTAR TABEL

Tabel 1. Simbol-simbol dalam Use case .....	34
Tabel 2. Simbol-Simbol Sequence Diagram .....	36
Tabel 3. Notasi Pada Activity Diagram .....	37
Tabel 4. Notasi Pada Class Diagram.....	38
Tabel 5. Tabel Peran Aktor .....	42
Tabel 6. Tabel Hotel.....	53
Tabel 7. Tabel Berita.....	54
Tabel 8. Tabel Kamar.....	54

## DAFTAR GAMBAR

Gambar 1. Disiplin ilmu SIG .....	9
Gambar 2. Subsystem SIG .....	15
Gambar 3. Use case diagram.....	43
Gambar 4. Sequence diagram Admin .....	44
Gambar 5. Sequence diagram User (Pengguna) .....	44
Gambar 6. Activity diagram Admin.....	45
Gambar 7. Activity diagram user .....	46
Gambar 8. Class diagram .....	47
Gambar 9. Struktur Program .....	47
Gambar 10. Desain awal form .....	48
Gambar 11. Desain halaman utama .....	48
Gambar 12. Menu drawer .....	49
Gambar 13. Desain Mencari Hotel pada Menu Utama .....	50
Gambar 14. Desain Lihat Peta pada Menu Utama .....	50
Gambar 15. Desain Menu Tentang .....	51
Gambar 16. Input Data Hotel .....	52
Gambar 17. Input Berita.....	52
Gambar 18. Input Kamar.....	53



# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Hotel merupakan salah satu jenis akomodasi yang sangat dikenal oleh masyarakat. Fungsi utama dari hotel adalah sebagai sarana untuk memenuhi kebutuhan tamu sebagai tempat tinggal sementara selama jauh dari tempat asalnya. Pada umumnya kebutuhan utama para tamu dalam hotel adalah istirahat, tidur, mandi, makan, minum, hiburan dan lain-lain. Namun dengan perkembangan dan kemajuan hotel sekarang ini, fungsi hotel bukan saja sebagai tempat menginap atau istirahat bagi para tamu, namun fungsinya bertambah sebagai tujuan konferensi, seminar, lokakarya, musyawarah nasional dan kegiatan lainnya semacam itu yang tentunya menyediakan sarana dan prasarana yang lengkap. Dengan demikian fungsi hotel sebagai suatu sarana komersial berfungsi bukan hanya untuk menginap, beristirahat, makan dan minum tetapi juga sebagai tempat melangsungkan berbagai macam kegiatan sesuai dengan tujuan pasar hotel tersebut.

Kota Padang Panjang merupakan salah satu daerah yang mempunyai potensi wisata yang bagus untuk dikembangkan. Kebanyakan pengunjung menyukai kebudayaan dan tempat wisata yang alami dan terjaga kelestariannya. Akan tetapi untuk tempat beristirahat untuk tidur belum banyak yang mengetahui lokasi tersebut terutama letak hotel-hotel yang ada di wilayah kota padang panjang. Salah satu penyebabnya, dikarenakan kurangnya informasi tentang lokasi hotel tersebut.

Berdasarkan hasil wawancara dengan salah satu pegawai dinas pariwisata kota Padang Panjang mengatakan bahwa saat ini penyampaian informasi hotel-hotel yang ada di Padang Panjang dilakukan dengan cara pemberian brosur, poster, dan peta geografis yang masih berupa peta konvensional (menggunakan media kertas) sehingga akan membuat para wisatawan kesulitan mencari lokasi hotel mana yang dekat dengan

posisinya tersebut dan tentunya kenyamanan wisatawan yang berkunjung akan berkurang.

Oleh karena itu dibutuhkan sebuah aplikasi untuk memperoleh informasi hotel dalam bentuk titik lokasi maupun informasi kedekatan jarak pengguna dengan lokasi tempat, rute jalan, penunjuk arah jalan menuju lokasi hotel yang akan dituju melalui *google maps*, sehingga dapat menghemat waktu dan biaya dalam menemukannya. Sistem informasi ini dirancang dalam bentuk sebuah aplikasi pencarian hotel di Kota Padang Panjang berbasis android. Sistem ini dapat memberikan informasi mengenai letak wilayah atau lokasi tempat-tempat yang ada dipermukaan bumi, dan dapat memberikan informasi mengenai rute yang harus ditempuh dari suatu lokasi ke lokasi lainnya. Aplikasi ini ditujukan untuk sistem operasi *mobile phone* android yaitu diakses secara mobile dengan menggunakan *handphone* yang menggunakan sistem operasi android. Berdasarkan masalah di atas, maka penulis mengajukan tugas akhir dengan judul **“Sistem Informasi Geografis Lokasi Hotel di Kota Padang Panjang Berbasis Android”**.

## **B. Identifikasi Masalah**

Berdasarkan latar belakang masalah diatas, maka dapat diidentifikasi masalah sebagai berikut :

1. Belum tersedianya sistem pemetaan hotel di Kota Padang Panjang.
2. Sulitnya mencari lokasi hotel yang dibutuhkan oleh wisatawan yang berkunjung.
3. Masih banyaknya wisatawan yang belum mengetahui rute menuju hotel yang ingin dituju.

## **C. Rumusan Masalah**

Berdasarkan beberapa identifikasi masalah maka dapat dirumuskan permasalahan, yaitu ***“Bagaimana merancang sebuah aplikasi lokasi hotel berbasis Android yang dapat memberikan informasi mengenai tempat hotel yang ada di Kota Padang Panjang”***.

#### **D. Batasan Masalah**

Berdasarkan rumusan masalah di atas, dapat dibuat suatu batasan masalah yang mencakup:

1. Aplikasi ini dapat diinstal pada *smartphone* bersistem operasi *android*.
2. *User* harus dapat melihat lokasi hotel yang telah disediakan oleh sistem.

#### **E. Tujuan**

Tujuan dari perancangan tugas akhir ini adalah :

1. Menghasilkan perancangan aplikasi pencarian hotel berbasis android.
2. Mengimplementasikan fitur untuk mengetahui informasi mengenai lokasi hotel yang ada di Kota Padang Panjang disertai dengan deskripsi dari tempat yang ingin dikunjungi tersebut.

#### **F. Manfaat Penelitian**

Manfaat yang dapat diperoleh dari tugas akhir ini adalah :

1. Memudahkan *user* dalam memperoleh informasi mengenai lokasi hotel yang ada di Kota Padang Panjang.
2. Memudahkan *user* dalam melihat deskripsi dari lokasi hotel yang akan dikunjungi beserta posisi geografisnya berdasarkan *google maps*.
3. Menjadikan penggunaan *mobile phone* lebih efisien karena informasi diperoleh dengan mudah dan praktis.

#### **G. Metodologi Penelitian**

Dalam penelitian tugas akhir ini penulis menggunakan beberapa metode penelitian antara lain :

1. Penelitian Lapangan (*field research*)

Dalam penelitian ini penulis mendapat data langsung dari hasil peninjauan lapangan mengadakan wawancara pada bagian yang berhubungan langsung dengan pembahasa yang peneliti teliti.

## 2. Penelitian Perpustakaan (*library Research*)

Penelitian ini dilakukan dengan cara mengumpulkan bahan-bahan dari buku-buku, karya ilmiah, ataupun tulisan yang berhubungan dengan permasalahan yang di bahas dalam penelitian tersebut.

## 3. Penelitian Laboratorium (*laboratory Research*)

Penulis melakukan pengolahan data yang telah dikumpulkan dengan menggunakan komputer sebagai alat bantu pembuatan tugas akhir.

### **H. Sistematika Penulisan**

Sistematika penulisan laporan tugas akhir ini dibagi dalam IV BAB yang disusun sebagai berikut :

#### **BAB I PENDAHULUAN**

Bab ini merupakan penguraian mengenai latar belakang masalah, identifikasi masalah, batasan masalah, rumusan masalah, tujuan penelitian, metodologi penelitian, sistematika penulisan.

#### **BAB II LANDASAN TEORI**

Bab ini berisi teori yang diambil dari buku-buku panduan dan referensi lainnya.

#### **BAB III ANALISA DAN PERANCANGAN**

Bab ini membahas analisa system yang sedang berjalan dan rancangan sistem yang diusulkan.

#### **BAB IV PENUTUP**

Bab ini berisikan kesimpulan yang didapat selama pembuatan laporan tugas akhir serta saran-saran yang akan menjadi masukan bagi perkembangan sistem selanjutnya.

## BAB II

### LANDASAN TEORI

#### A. Kota Padang Panjang

Kota Padang Panjang adalah kota dengan luas wilayah terkecil di Sumatera Barat, Indonesia. Kota ini memiliki julukan sebagai Kota Serambi Mekkah, dan juga dikenal sebagai Mesir van Andalas (Egypte van Andalas). Sementara wilayah administratif kota ini dikelilingi oleh wilayah administratif Kabupaten Tanah Datar.

Kawasan kota ini sebelumnya merupakan bagian dari wilayah Tuan Gadang di Batipuh. Pada masa Perang Padri kawasan ini diminta Belanda sebagai salah satu jalur jalan baru dari kota ini menuju Kota Padang karena lebih mudah dibandingkan melalui kawasan Kubung XIII di kabupaten Solok sekarang. Kota ini pernah menjadi pusat pemerintahan sementara Kota Padang, setelah Kota Padang dikuasai Belanda pada masa agresi militer Belanda sekitar tahun 1947.

Kota ini berada di daerah ketinggian yang terletak antara 650 sampai 850 meter di atas permukaan laut, berada pada kawasan pegunungan yang berhawa sejuk dengan suhu udara maksimum 26.1 °C dan minimum 21.8 °C, dengan curah hujan yang cukup tinggi dengan rata-rata 3.295 mm/tahun. Di bagian utara dan agak ke barat berjejer tiga gunung: Gunung Marapi, Gunung Singgalang dan Gunung Tandikek.

#### B. Hotel

Salah satu penunjang untuk majunya dunia pariwisata adalah hotel, yaitu sebagai sarana akomodasi bagi para wisatawan, baik wisatawan Nusantara maupun wisatawan mancanegara. Agar para wisatawan yang mempergunakann sarana akomodasi ini merasa senang, nyaman, serta mendapatkan kepuasan didalam menerima *service* selama tinggal di hotel, maka hotel harus dikelola secara propesional.

Kata hotel merupakan perkembangan dari bahasa Prancis yaitu *hostel*, diambil dari bahasa Latin *hospes*, dan mulai diperkenalkan kepada masyarakat umum pada tahun 1797. Sebelum istilah hotel digunakan di Inggris, rumah-rumah penginapan bagi orang yang berpergian disebut *inn*. Dalam terminology ( ilmu mengenai definisi dan istilah) resmi, tidak ada perbedaan definisi antara kata hotel dan *inn*. Hotel adalah suatu bentuk akomodasi yang dikelola secara komersial, disediakan bagi setiap orang untuk memperoleh pelayanan dan penginapan berikut makan dan minum. (SK Menteri Perhubungan No: Pm 10/Pw 301/Phb 77) (Dalam Yayuk, 1997:2)

Dalam perkembangannya, hotel didefinisikan sebagai berikut :

1. Hotel adalah bentuk bangunan yang menyediakan kamar-kamar untuk menginap para tamu, makanan dan minuman serta fasilitas-fasilitas lain yang diperlukan, dan dikelola secara profesional untuk mendapatkan keuntungan.
2. Hotel adalah suatu badan usaha yang bergerak dalam bidang jasa akomodasi yang dikelola secara komersial. Di dalam hotel para tamu mendapatkan layanan penginapan, makanan dan minuman, serta fasilitas lainnya.
3. Menurut SK Menteri Perhubungan No. 241/II/1970, hotel adalah perusahaan yang menyediakan jasa dalam bentuk akomodasi serta menyediakan hidangan dan fasilitas lainnya di dalam hotel untuk umum yang memenuhi syarat *comfort* dan bertujuan komersial dalam jasa tersebut (STP, Bandung, 1992:6).
4. Hotel adalah suatu jenis akomodasi yang menggunakan sebagian atau seluruh area bangunan untuk menyediakan jasa pelayanan/penginapan, makan, minum serta jasa lainnya bagi umum yang dikelola secara komersial (SK Menparpostel No. Km. 34/NK.103/MPPT 87).
5. Hotel adalah usaha komersial yang menyediakan tempat menginap, makanan dan pelayanan-pelayanan lain untuk umum.

6. Hotel adalah suatu perusahaan yang dikelola oleh pemiliknya dengan menyediakan pelayanan makanan, minuman dan fasilitas kamar untuk tidur kepada orang-orang yang sedang melakukan perjalanan dan mampu membayar dengan jumlah yang wajar sesuai dengan pelayanan yang diterima tanpa adanya perjanjian khusus.

Hasil dari penelitian yang dilakukan di Kantor Dinas Pariwisata Kota Padang Panjang terdapat sebanyak 29 penginapan yang ada di Kota Padang Panjang

## C. Sistem Informasi Geografis (SIG)

### 1. Pengertian Sistem Informasi Geografis

Sistem Informasi Geografis atau disingkat SIG merupakan suatu sistem berbasis komputer yang digunakan untuk mengumpulkan, menyimpan, menggabungkan, mengatur, mentransformasi, memanipulasi, dan menganalisis data-data geografis. Data geografis yang dimaksud disini adalah data spasial yang ciri-cirinya adalah:

- a) memiliki *geometric properties* seperti koordinat dan lokasi.
- b) Terkait dengan aspek ruang seperti persil, kota, kawasan pembangunan.
- c) Berhubungan dengan semua fenomena yang terdapat di bumi, misalnya data, kejadian, gejala atau objek.
- d) Dipakai untuk maksud-maksud tertentu, misalnya analisis, pemantauan ataupun pengelolaan.

Pengertian informasi geografis adalah informasi mengenai tempat-tempat yang terletak dipermukaan bumi, pengetahuan mengenai posisi dimana suatu objek terletak di permukaan bumi dan informasi mengenai keterangan-keterangan (atribut) yang terdapat di permukaan bumi yang posisinya di ketahui. Objek-objek dan fenomena-fenomena dimana lokasi geografis itu berada penting

dianalisis demi pengambilan keputusan-keputusan atau demi kepentingan-kepentingan tertentu.

Pada dasarnya istilah sistem informasi geografis merupakan gabungan dari tiga unsur pokok: sistem, informasi, geografis. Dengan demikian, pengertian terhadap ketiga unsur-unsur pokok ini akan sangat membantu dalam memahami SIG. Dengan melihat unsur-unsur pokoknya, maka jelas SIG merupakan salah satu sistem informasi dengan tambahan unsur geografis. Atau, SIG merupakan suatu sistem yang menekankan pada unsur informasi geografis.

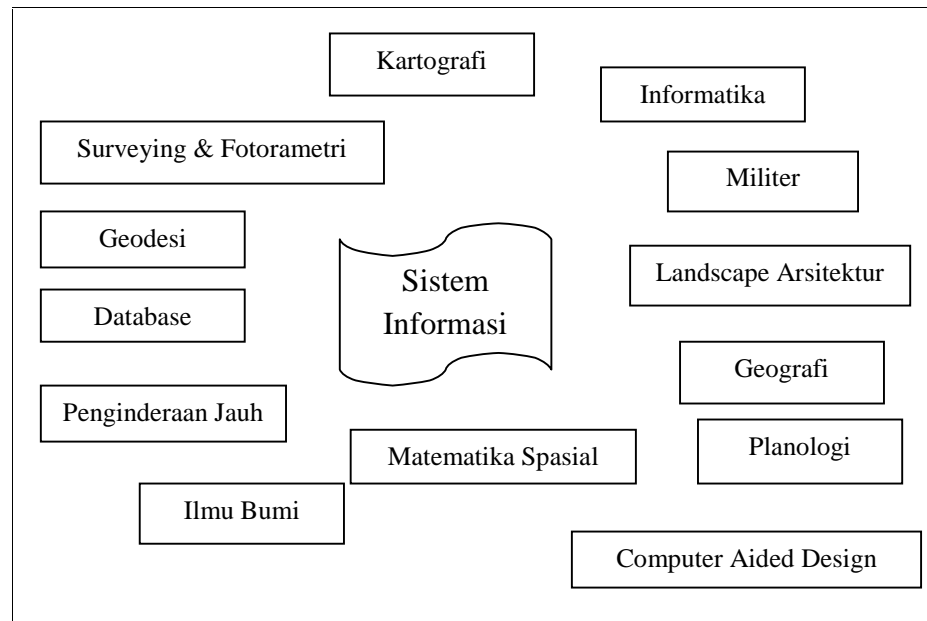
Istilah geografis merupakan bagian dari spasial (keruangan). Kedua istilah ini sering digunakan secara bergantian atau tertukar hingga timbul istilah yang ketiga, geospasial. Ketiga istilah ini mengandung pengertian yang sama di dalam konteks SIG. Penggunaan kata geografis mengandung pengertian suatu persoalan mengenai bumi: permukaan dua atau tiga dimensi.

Dengan memperhatikan pengertian sistem informasi, maka SIG merupakan suatu kesatuan formal yang terdiri dari berbagai sumberdaya fisik dan logika yang berkenaan dengan objek-objek yang terdapat dipermukaan bumi. Jadi, SIG juga merupakan sejenis perangkat lunak yang dapat digunakan untuk pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran informasi geografis.

## **2. Disiplin Ilmu Pendukung SIG**

Disiplin ilmu pendukung SIG antara lain geografis, geodesi, informatika, sistem basis data, kartografi, surveying, fotogrametri, penginderaan jauh, matematika spasial, ilmu bumi, planologi, dan sebagainya. Pengembangan SIG dipengaruhi oleh teori topologi, teori graph dan hitungan geometri karena sebagian besar masalah SIG adalah masalah geometri.





**Gambar 1. Disiplin ilmu SIG**

### 3. Perangkat Lunak SIG

Perangkat lunak Sistem Informasi Geografis berkembang sangat pesat karena dikembangkan oleh beberapa pihak, baik di lingkungan pemerintah, kampus maupun perusahaan.

Di lingkungan pemerintah, perangkat lunak SIG dikembangkan oleh pemerintah Kanada dengan nama CGIS (*Canadian Geographic Information System*), yang merupakan perangkat lunak SIG yang pertama di dunia. Di lingkungan pemerintah Amerika Serikat, dikembangkan pula perangkat lunak yang serupa dengan CGIS yaitu MIDAS, yang digunakan untuk memproses data-data sumber daya alam.

Perangkat lunak SIG juga dikembangkan di lingkungan kampus. Harvard University di Amerika Serikat telah menghasilkan perangkat lunak SIG seperti SYMAP (*Synagraphic Mapping*), CALFORM, SYMVU, GRID, POLYRT dan ODISEY. Di ITC, Belanda dikembangkan perangkat lunak SIG yang dinamai ILWIS (*Integrated Land and Water Information System*). ILWIS

menggabungkan kemampuan pengolahan citra, basis data dan beberapa karakteristik SIG konvensional.

Banyak perusahaan swasta yang juga mengembangkan perangkat lunak SIG, contohnya adalah ESRI (*Environmental System Research Institute*) yang memproduksi ARC/INFO yang mampu menghubungkan teknologi basis data relasional standar (INFO) dengan perangkat lunak khusus yang digunakan untuk menangani objek-objek yang disimpan sebagai data spasial (ARC). ESRI juga mengembangkan ArcView dan ArcGIS untuk komputer desktop yang memiliki tampilan yang lebih menarik dan interaktif.

#### **4. Keuntungan menggunakan SIG**

Berikut ini adalah keuntungan-keuntungan penggunaan SIG:

- a) SIG mempunyai kemampuan untuk memilih dan mencari detail atau tema yang diinginkan, menggabungkan satu kumpulan data dengan kumpulan data lainnya, melakukan pemutakhiran (updating) dengan lebih cepat, memodelkan serta menganalisis suatu keputusan.
- b) SIG dapat membantu pemerintah, perusahaan-perusahaan atau perorangan untuk mengambil keputusan terbaik atau mencari solusi untuk masalah-masalah tertentu yang erat kaitannya dengan objek-objek di permukaan bumi.
- c) SIG dengan mudah menghasilkan peta-peta tematik yang dapat digunakan untuk menampilkan informasi-informasi tertentu. Peta-peta tematik tersebut dapat dibuat dari peta-peta yang sudah ada sebelumnya, hanya dengan memanipulasi atribut-atributnya.
- d) SIG memiliki kemampuan untuk menguraikan unsur-unsur yang terdapat di permukaan bumi menjadi beberapa layer data spasial.
- e) SIG mempunyai tingkat fleksibilitas yang sangat tinggi dalam pengelolaan data, sehingga selain dapat menurunkan/menghasilkan data dalam berbagai bentuk, juga dapat menghasilkan

beberapa macam keluaran (peta dan grafis) serta menampung dan menerima masukan dalam format yang berbeda.

- f) Perangkat lunak SIG terutama desktop SIG mudah digunakan karena dilengkapi dengan bantuan menu-menu dan help, otomatis, cepat, menarik, informatif, dan user friendly.
- g) Perangkat lunak SIG beserta perintah-perintah dalam bahasa skrip yang dimilikinya dapat meng-customize secara efektif dan efisien hampir semua aplikasi SIG, sesuai dengan kebutuhan pengguna.
- h) Perangkat lunak SIG dapat menerima dan bertukar data dengan aplikasi-aplikasi perangkat lunak lainnya, seperti Microsoft Word, Microsoft Excel, Microsoft Access, Oracle maupun dengan perangkat lunak SIG lainnya.\
- i) Produk-produk aplikasi SIG dapat dipublikasikan melalui internet sehingga dapat diakses dengan browser oleh siapa saja yang ingin mempelajari, menggunakan atau mengambil manfaat dari aplikasi-aplikasi SIG tersebut.

## **5. Aplikasi-aplikasi SIG**

Berdasarkan data yang dihimpun Sasitawarih, ( 1994) dalam direktoriat penginderaan jauh & SIG di Indonesia bahwa perangkat lunak/ keras yang terpasang di departemen teknis lebih banyak (16 set), menyusul swasta (13 set) dan kemudian perguruan tinggi (12 set). Pada priode dua tahun (soesilo, 1996) jumlah ini sudah meningkat, menjadi sekitar 150 lembaga yang menggunakan sarana penginderaan jauh SIG sebagai sarana untuk mempermudah pekerjaanya.

Karena kemudahan dalam pengelolaan dan penggunaanya, SIG banyak digunakan di berbagai bidang dan berbagai disiplin ilmu, misalnya:

a) Bidang pendidikan

SIG dapat digunakan untuk menentukan lokasi sekolah-sekolah, untuk sistem informasi pendidikan dan sebagai alat pemahaman pada pembelajaran geografi.

b) Bidang Transportasi dan Perhubungan.

SIG digunakan dalam manajemen pemeliharaan dan perencanaan perluasan jaringan transportasi, penentuan jalur transportasi yang efektif, analisis rawan kemacetan dan bahaya kecelakaan serta inventarisasi jaringan transportasi.

c) Bidang Telekomunikasi

Di bidang ini, SIG dimanfaatkan dalam perencanaan, pemeliharaan dan analisis perluasan jaringan telekomunikasi, pembuatan sistem informasi pelanggan dan fasilitas umum telekomunikasi seperti telepon umum, wartel, warnet, dan lainnya.

d) Bidang Ekonomi, Bisnis, dan Marketing

SIG dimanfaatkan untuk menentukan lokasi-lokasi bisnis yang prospektif untuk bank, pasar swalayan, mal atau supermal, mesin ATM, kantor cabang, show room, counter, outlet, gudang dan sejenisnya, dengan memperhatikan lokasi konsumen atau pelanggan. Selain itu, SIG dapat pula digunakan untuk menganalisis rute tersingkat yang harus dilalui traveling salesman.

e) Bidang Militer

SIG sangat dibutuhkan dalam penyediaan data spasial untuk analisis rute-rute perjalanan logistik dan peralatan perang, pembuatan peta elektronik yang dihubungkan dengan radar yang akan mendeteksi kendaraan-kendaraan ataupun pesawat-pesawat militer musuh maupun untuk usaha pertahanan negara.

f) Bidang Geologi, Pertambangan, dan Perminyakan

SIG digunakan untuk menentukan lokasi-lokasi pertambangan, geologi dan perminyakan yang memperhitungkan keamanan para pekerja tambang dan kelestarian lingkungan. SIG

dapat pula digunakan untuk menganalisis limbah-limbah yang merupakan hasil sampingan dari industri tambang serta menginventarisasi manajemen dan perizinan proyek-proyek pertambangan.

g) Bidang Lingkungan

Analisis dan pemantauan pencemaran udara, limbah berbahaya, dan pencemaran air, sungai, danau, laut, evaluasi pengendapan lumpur atau sedimen baik disekitar danau, sungai atau pantai dapat dilakukan dengan SIG.

h) Bidang Perpajakan

SIG dapat dimanfaatkan untuk menaksir potensi pendapatan dari sektor pajak PBB (Pajak Bumi dan Bangunan), yaitu dengan membuat sistem informasi untuk penarikan pajak dari sektor periklanan yang berasal dari perizinan dan pemasangan papan iklan komersial, plang, billboard yang terkait dengan data posisi, ruang dan masa berlaku.

i) Bidang Perencanaan

SIG dapat menjadi alat yang ampuh dalam perencanaan pemukiman transmigrasi, perencanaan tata ruang wilayah, perencanaan kota, perencanaan pengembangan desa tertinggal, perencanaan lokasi dan relokasi industri, pasar, pemukiman dan sebagai.

j) Bidang Sumber Daya Alam

SIG bermanfaat untuk inventarisasi, manajemen dan kesesuaian lahan untuk pertanian, perkebunan, kehutanan, perencanaan, tata guna lahan, analisis daerah rawan bencana alam, analisis dan pemantauan daerah-daerah kebakaran hutan.

k) Bidang Kesehatan

SIG dapat digunakan untuk menentukan distribusi penderita suatu penyakit, pola atau model penyebaran penyakit. Penentuan distribusi unit-unit rumah sakit ataupun puskesmas-puskesmas,

fasilitas-fasilitas kesehatan maupun jumlah tenaga medis dapat pula dilakukan dengan SIG.

l) Bidang Hidrografi dan Kelautan

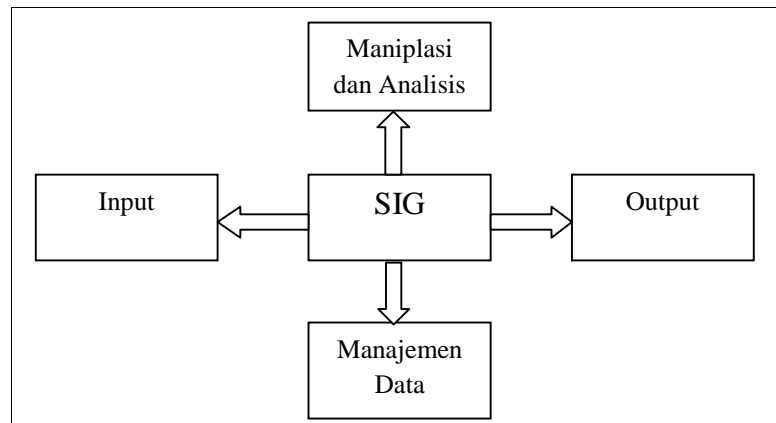
SIG bermanfaat dalam kegiatan inventarisasi dan manajemen stasiun pengamatan pusat, manajemen daerah pesisir pantai, manajemen daerah wisata laut/bahari, taman laut, coastal manajemen dan sebagainya.

m) Bidang Utilitas

SIG dapat digunakan dalam inventarisasi dan manajemen informasi jaringan pipa air minum, sistem informasi pelanggan perusahaan air minum, perencanaan dan perluasan jaringan pipa air minum. Inventarisasi dan manajemen informasi jaringan listrik, pemeliharaan dan perluasan jaringan listrik, penentuan distribusi penempatan tong-tong sampah, WC umum serta fasilitas-fasilitas umum lainnya dapat pula dilakukan.

## 6. Subsistem SIG

Sistem informasi geografis merupakan sistem yang dapat mendukung pengambilan keputusan spasial dan mampu mengintegrasikan deskripsi-deskripsi lokasi dengan karakteristik-karakteristik fenomena yang ditemukan di lokasi tersebut. SIG dapat diuraikan menjadi beberapa subsistem berikut:



Sumber : (Eddy Prahasta, 2005:57).

**Gambar 2. Subsystem SIG**

a) Subsystem masukan (input)

Subsystem ini bertugas mengumpulkan dan mempersiapkan data spasial dan atribut dari berbagai sumber. Subsystem ini juga bertanggungjawab atau mentransformasi format-format data asli ke dalam format yang dapat digunakan oleh SIG.

b) Subsystem manajemen

Subsystem ini mengorganisasikan data spasial maupun atribut ke dalam sistem basisdata sedemikian rupa sehingga data spasial tersebut mudah dicari, di-update dan diedit.

c) Subsystem masukan dan analisis

Subsystem ini menentukan informasi-informasi yang dapat dihasilkan oleh SIG. Selain itu, subsystem ini juga melakukan manipulasi dan pemodelan data untuk menghasilkan informasi yang diharapkan.

d) Subsystem keluaran (output) dan penyajian (display)

Subsystem ini menampilkan atau menghasilkan keluaran seluruh atau sebagian basis data, baik dalam bentuk softcopy maupun hardcopy, dalam format tabel, grafik, peta atau format lainnya.

## 7. Komponen SIG

### a) Perangkat Keras (*hardware*)

Perangkat keras SIG adalah perangkat-perangkat fisik yang merupakan bagian dari sistem komputer yang mendukung analisis geografi dan pemetaan. Perangkat keras SIG mempunyai kemampuan untuk menyajikan citra dengan resolusi dan kecepatan yang tinggi serta mendukung operasi-operasi basis data dengan volume data yang besar secara cepat.

### b) Perangkat Lunak (*software*)

Perangkat lunak digunakan untuk melakukan proses menyimpan, menganalisa, memvisualkan data-data baik data spasial maupun non-spasial. Perangkat lunak yang harus terdapat dalam komponen software SIG adalah:

- 1) Alat untuk memasukkan dan memanipulasi data SIG
- 2) Data Base Management System (DBMS)
- 3) Alat untuk menganalisa data-data
- 4) Alat untuk menampilkan data dan hasil analisa

### c) Data

Pada prinsipnya terdapat dua jenis data untuk mendukung SIG yaitu :

#### 1) Data Spasial

Data spasial adalah gambaran nyata suatu wilayah yang terdapat di permukaan bumi. Umumnya direpresentasikan berupa grafik, peta, gambar dengan format digital dan disimpan dalam bentuk koordinat  $x$ ,  $y$  (vektor) menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan titik-titik, garis-garis atau kurva, atau poligon beserta atribut-atribut atau dalam bentuk image (raster) yang menampilkan, dan menyimpan data spasial dengan menggunakan struktur matriks atau piksel-piksel yang membentuk grid.



Sesuai dengan perkembangan, peta tidak hanya merepresentasikan obyek-obyek yang ada di muka bumi, tetapi berkembang menjadi representasi obyek diatas muka bumi (diudara) dan dibawah permukaan bumi.

## 2) Data Non Spasial (Atribut)

Data non spasial adalah data berbentuk tabel dimana tabel tersebut berisi informasi- informasi yang dimiliki oleh obyek dalam data spasial. Data tersebut berbentuk data tabular yang saling terintegrasi dengan data spasial yang ada.

## d) Manusia (pengguna)

Manusia merupakan inti elemen dari SIG karena manusia adalah perencana dan pengguna dari SIG. Pengguna SIG mempunyai tingkatan seperti pada sistem informasi lainnya, dari tingkat spesialis teknis yang mendesain dan mengelola sistem sampai pada pengguna yang menggunakan SIG untuk membantu pekerjaannya sehari-hari.

## e) Metode

Metode yang digunakan dalam SIG akan berbeda untuk setiap permasalahan. SIG yang baik tergantung pada aspek desain dan aspek realnya.

Ruang Lingkup Sistem Informasi Geografis :

### 1) Input Data

Proses input data digunakan untuk menginputkan data spasial dan data non-spasial. Data spasial biasanya berupa peta analog. Untuk SIG harus menggunakan peta digital sehingga peta analog tersebut harus dikonversi ke dalam bentuk peta digital dengan menggunakan alat digitizer. Selain proses digitasi dapat juga dilakukan proses overlay dengan melakukan proses scanning pada peta analog.

## 2) Manipulasi Data

Tipe data yang diperlukan oleh suatu bagian SIG mungkin perlu dimanipulasi agar sesuai dengan sistem yang dipergunakan. Oleh karena itu SIG mampu melakukan fungsi edit baik untuk data spasial maupun non-spasial.

## 3) Manajemen Data

Setelah data spasial dimasukkan maka proses selanjutnya adalah pengolahan data non-spasial. Pengolahan data non-spasial meliputi penggunaan DBMS untuk menyimpan data yang memiliki ukuran besar.

## 4) *Query* dan Analisis

*Query* adalah proses analisis yang dilakukan secara tabular. Secara fundamental SIG dapat melakukan dua jenis analisis, yaitu:

### a. Analisis *Proximity*

Analisis *Proximity* merupakan analisis geografi yang berbasis pada jarak antar layer. SIG menggunakan proses *buffering* (membangun lapisan pendukung di sekitar layer dalam jarak tertentu) untuk menentukan dekatnya hubungan antar sifat bagian yang ada.

### b. Analisis *Overlay*

*Overlay* merupakan proses penyatuan data dari lapisan layer yang berbeda. Secara sederhana *overlay* disebut sebagai operasi visual yang membutuhkan lebih dari satu layer untuk digabungkan secara fisik.

## 5) Visualisasi

Untuk beberapa tipe operasi geografis, hasil akhir terbaik diwujudkan dalam peta atau grafik. Peta sangatlah efektif untuk menyimpan dan memberikan informasi geografis.

## D. Tool Pengembangan Sistem

### 1. Android

*Android* merupakan generasi baru *platform mobile*, platform yang memberikan pengembang untuk melakukan pengembangan sesuai yang diharapkannya. Sistem operasi yang mendasari *android* dilisensikan di bawah GNU, *general public lisensi* versi 2 (GPLv2), yang lebih dikenal dengan istilah *copyleft*, lisensi di mana setiap perbaikan pihak ketiga harus terus dibawah syarat (terms). *Android* di distribusikan di bawah *Lisensi Apache Software* (ASL/Apache 2), yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen handset khususnya) dapat memilih untuk meningkatkan platform tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apapun yang mereka inginkan. Pengembang aplikasi *android* diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan. (Hendra Nugraha Lengkong, dkk, 2015)

Android dikembangkan bersama oleh perusahaan-perusahaan yang tergabung dalam sebuah konsorsium bernama Open Handset Alliance (OHA). OHA dipimpin oleh Google dan didirikan bersama 34 perusahaan lainnya, dengan tujuan untuk mengembangkan teknologi *mobile device*. OHA juga bergerak di berbagai bidang seperti pembuat *device*, semikonduktor, pembuat aplikasi, komersialisasi, dan *mobile operator* (Alfa Satyaputra, dkk, 2014).

Android lebih banyak digunakan karena memiliki beberapa kelebihan dibanding dengan SO lainnya, diantaranya :

- a) Android lebih accessible karena dapat berjalan di perangkat komputer apapun yang sudah terinstal dengan SDK, Java

Development Kit (JDK), Software Java, dan Android Development Tool (ADT) *Plugin*-nya.

- b) Dalam ketersediaan aplikasi, Android yang berbasis linux memudahkan *programmer* dalam membuat aplikasi baru yang bebas didistribusikan dengan lisensi *Open source*, *Shareware*, atau bahkan *Freeware*.
- c) Android memberikan keringanan biaya dan kemudahan untuk melakukan rilis aplikasi lewat Google Play.
- d) Android bersifat *open platform* atau tidak terikat dengan salah satu produsen perangkat keras atau salah satu operator.
- e) Android bersifat *Cross-Compatibility* yang artinya dapat berjalan dengan banyak ukuran *screen* dan resolusi.

## 2. Java

Java adalah bahasa berorientasi objek yang dapat digunakan untuk pengembangan aplikasi mandiri, aplikasi berbasis internet, serta aplikasi untuk perangkat perangkat cerdas yang dapat berkomunikasi lewat internet atau jaringan komunikasi. Dalam Java ada dua jenis program berbeda, yaitu aplikasi dan *applet*. Aplikasi adalah program yang biasanya disimpan dan dieksekusi dari komputer lokal sedangkan *applet* adalah program yang biasanya disimpan pada komputer yang jauh, yang dikoneksikan pemakai lewat *web browser*. (Hendra Nugraha Lengkong, dkk, 2015)

## 3. SDK (Software Development Kit)

Menurut Safaat (2011), SDK adalah tools API (*Application Programming Interface*). SDK diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

Menurut Alfa Satyaputra, dkk (2014) SDK juga sering disebut sebagai *software* emulator yang mensimulasikan OS Android pada PC. Saat ini SDK sudah berbentuk *bundle*, yang didalamnya terdapat Eclipse + ADT Plugin, Android SDK Tools, Android Platform Tools dan Android *System Image* terbaru untuk emulator.

#### 4. ADT (Android Development Tools)

Menurut Safaat (2011), ADT ada lah *plug-in* yang membuat *eclipse* dapat membuat *project* berbasis Android. ADT harus di install, karena sebagai penghubung antara Android SDK dengan IDE *eclipse* yang akan digunakan sebagai tempat *coding* aplikasi android nantinya. ADT memberikan kemudahan dalam membuat aplikasi *project* android, membuat GUI aplikasi, melakukan *debug* aplikasi dengan menggunakan android SDK tolls dan membuat *package* android (.apk) yang digunakan untuk mendistribusikan aplikasi android yang sudah dirancang. Berikut ini ada beberapa fitur yang disediakan *eclipse* dan ADT :

- a) Pembuatan, pembangunan, pengemasan, instalasi, *debugging*, dan penandaan file aplikasi android yang terintegrasi.
- b) Integrasi SDK tools.
- c) Bahasa pemograman Java dan XML editor.
- d) Dokumentasi yang terintegrasi untuk API Android Framework.

#### 5. Google Maps

- a) Google Maps

*Google Maps* merupakan layanan gratis yang diberikan oleh *Google* dan sangat populer. Fungsinya ialah untuk melihat lokasi, mencari alamat, mendapatkan petunjuk mengemudi dan lain sebagainya. *Google Maps* berbentuk sebuah peta dunia yang dapat digunakan untuk melihat suatu daerah. Dengan kata lain, *Google Maps* adalah suatu peta yang dapat dilihat dengan menggunakan suatu *browser*. Fitur *Google Maps* dapat ditampilkan dalam *web*

atau blog dengan *Google Maps API*. *Google Maps API* (*Application Programming Interface*) adalah suatu *library* yang berbentuk *Javascript* yang memungkinkan *developer* lain untuk memanfaatkan aplikasi ini di aplikasi buatannya. Tetapi sekarang ini *Google Maps API* dapat juga dengan hanya memasang dengan kode atau skrip yang didapat dari *Google Maps*. Tampilan *Google Maps* pun dapat dipilih berdasarkan foto asli atau peta gambar rute saja. Peta *globe virtual Google Maps* ini dapat ditemukan di <http://maps.google.com>. (Anisa Isna Yesiana, 2016)

#### b) Google Maps API

*Google Maps API* merupakan pengembangan teknologi dari google yang digunakan untuk menanamkan *Google Map* di suatu aplikasi yang tidak dibuat oleh Google. *Google Maps API* adalah suatu *library* yang berbentuk *javascript* yang berguna untuk memodifikasi peta yang ada di *Google Maps* sesuai kebutuhan (Elian, 2012). Dalam perkembangannya *Google Maps API* diberikan kemampuan untuk mengambil gambar peta statis. Melakukan *geocoding*, dan memberikan penuntun arah. *Google Maps API* bersifat gratis untuk publik. Penggunaan *Google Maps API* pada pengembangan aplikasi android dengan menggunakan Eclipse dan komputer menggunakan sistem operasi *windows*.

Kekurangan yang ada pada *Google Maps API* yaitu jika ingin melakukan akses harus terdapat layanan internet pada perangkat yang digunakan. Sedangkan kelebihan yang ada pada *Google Maps API* yaitu:

- 1) Dukungan penuh yang dilakukan Google sehingga terjamin dan bervariasi fitur yang ada pada *Google Maps API*.
- 2) Banyak pengembang yang menggunakan *Google Maps API* sehingga mudah dalam mencari referensi dalam pengembangan aplikasi.

## 6. Android Studio

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) untuk *platform* Android. Android Studio ini diumumkan pada tanggal 16 Mei 2013 pada Konferensi Google I/O oleh Produk Manajer Google, Ellie Powers. Android Studio bersifat *free* dibawah *Apache License* 2.0. Android Studio awalnya dimulai dengan versi 0.1 pada bulan mei 2013, kemudian dibuat versi beta 0.8 yang dirilis pada bulan Juni 2014. Pada bulan Desember 2014 dirilis kembali yang lebih baik dari sebelumnya dari versi 1.0. Berbasiskan *JetBrainns' IntelliJ IDEA* Studio didesain khusus untuk *Android Development. Software* ini sudah bisa digunakan untuk Windows, Mac OS X, dan Linux (Budiman Ruliansyah, 2016).

## 7. Arsitektur Android

Secara garis besar Arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut:

### a) Application and Widgets

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

### b) Application Frameworks

Android adalah "*Open Development Platform*" yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alarm, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju API

framework seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*).

Sehingga bisa kita simpulkan *Application Frameworks* ini adalah layer dimana para pembuat aplikasi melakukan pengembangan / pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada *layer* inilah aplikasi dapat dirancang dan dibuat, seperti *content-providers* yang berupa sms dan panggilan telepon.

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

- 1) *Views*
  - 2) *Content Provider*
  - 3) *Resource Manager*
  - 4) *Notification Manager*
  - 5) *Activity Manager*
- c) *Libraries*

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

- 1) *Libraries* media untuk pemutaran media audio dan video
- 2) *Libraries* untuk manajemen tampilan
- 3) *Libraries Graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D
- 4) *Libraries SQLite* untuk dukungan database
- 5) *Libraries SSL* dan *Webkit* terintegrasi dengan web browser dan security
- 6) *Libraries LiveWebcore* mencakup modern web browser dengan engine embeded web view



7) Libraries 3D yang mencakup implementasi OpenGL ES 1.0 API's

d) *Android Run Time*

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

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

e) Fundamental Aplikasi

Aplikasi android ditulis dalam bahasa pemrograman java. Kode java dikompilasi bersama dengan data *file resource* yang dibutuhkan oleh aplikasi, dimana prosesnya dipackage oleh tools yang dinamakan “apt tools” ke dalam paket android sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat diinstall di perangkat seluler.

Ada enam jenis komponen pada aplikasi android yaitu:

1) *Activities*

Suatu activity akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi android bisa jadi hanya memiliki satu

activity, tetapi umumnya aplikasi memiliki banyak activity tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Satu activity biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada user. Untuk pindah dari satu activity ke activity lain kita dapat melakukannya dengan satu event, misalnya *klik* tombol, memilih opsi atau menggunakan triggers tertentu. Secara hirarki sebuah *windowsactivity* dinyatakan dengan *methodActivity setContentView()*.

*ContentView* adalah objek yang berada pada root hirarki.

## 2) Services

Service tidak memiliki *Graphic User Interface* (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, media player sedang memutar lagu dari list yang ada, aplikasi ini akan memiliki dua atau lebih activity yang memungkinkan user untuk memilih lagu misalnya, atau menulis sms sambil player sedang jalan. Untuk menjaga musik tetap di jalankan, activity player dapat menjalankan service. Service dijalankan pada thread utama dari proses aplikasi.

## 3) Broadcast Receiver

*Broadcast receiver* berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh broadcast seperti notifikasi zona waktu berubah, baterai low, gambar telah selesai diambil oleh camera, atau perubahan referensi bahasa yang digunakan. Aplikasi juga dapat menginisiasi broadcast misalnya memberikan informasi pada aplikasi lain bahwa ada

data yang telah diunduh ke perangkat dan siap untuk digunakan.

*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* (getaran) perangkat, dan lain sebagainya.

#### 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*, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta (Map), atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider*.

#### f) Versi Android

Telepon pertam yang memakai sistem operasi android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua vendor seluler didunia menggunakan android sebagai operating system. Adapun versi-versi android yang pernah dirilis adalah sebagai berikut:

##### 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, mengupload 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 diintegrasikan, CDMA/EVDO, 802.1x, VPN, Gestures dan Text-to-speech engine, kemampuan dial kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel, pengadaan resolusi VWGA).

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

Pada 3 Desember 2009 kembali diluncurkan ponsel 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*)

Pada bulan Mei 2010 Android versi 2.2 Rev 1 diluncurkan. Android inilah yang sekarang sangat banyak beredar di pasaran, salah satunya adalah dipakai di Samsung FX tab yang sudah ada di pasaran. Fitur yang tersedia di Android versi ini sudah kompleks diantaranya adalah:

- a. Kerangka aplikasi memungkinkan penggunaan dan penghapusan komponen yang tersedia
  - b. *Dalvik Virtual Machine* dioptimalkan untuk perangkat media
  - c. Grafik: grafik 2D dan grafis 3D berdasarkan libraries OpenGL
  - d. SQLite: untuk penyimpanan data
  - e. Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
  - f. GSM, Bluetooth, EDGE, 3G, dan Wifi (hardware independent)
  - g. Kamera, *Global Positioning System* (GPS), kompas dan *accelerometer* (tergantung hardware)
- 6) Android versi 2.3 (Gingerbread)
- Android versi 2.3 diluncurkan pada Desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan seperti berikut:
- a. *SIP-based VoIP*
  - b. *Near Field Communications* (NFC)
  - c. *Gyroscope* dan *sensor*
  - d. *Multiple cameras support*
  - e. *Mixable audio effects*
  - f. *Download Manager*
- 7) Android versi 3.0 (Honeycomb) dirilis Februari 2011 sebagai android 3.0 revisi 1 serta android versi 3.0 revision 2 telah dirilis pada Juli 2011.
- 8) Android versi 3.1 dirilis Mei 2011, sedangkan android 3.1 revisi 2 juga dirilis Mei 2011, serta android 3.1 revision 3 dirilis pada Juli 2011
- 9) Android versi 3.2 dirilis Juli 2011

- 10) Android versi 4.0 dirilis November 2011
- 11) Android versi 4.1
- 12) Android versi 4.2
- 13) Android versi 4.3
- 14) Android versi 4.4
- 15) Android versi 5.0
- 16) Android versi 6.0
- 17) Android versi 7.0

Android versi 3.0 ke atas adalah generasi *platform* yang digunakan untuk tablet pc. Sementara versi 4.0 sudah merupakan *platform* yang bisa dipakai di smartphone dan tablet pc.

## 8. PHP (Hypertext Preprocessor)

*Hypertext Preprocessor* yaitu bahasa pemrograman web *server-side* yang bersifat *open source* yang merupakan hasil kerja pertama dari Rasmus Lerdorf pada tahun 1995. PHP adalah script yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima oleh *client* selalu yang terbaru (*up to date*). Semua script PHP dieksekusi pada *server* di mana *script* tersebut dijalankan.

PHP menurut Wardana (2010:7) adalah bahasa *scripting* yang menyatu dengan HTML (kode dasar *website*) dan dijalankan pada *server side*. Artinya, semua sintaks PHP yang diberikan akan sepenuhnya dijalankan pada *server*, sedangkan yang dikirimkan ke *browser* hanya hasilnya saja. Sedangkan „menurut Nugroho (2009:113), PHP merupakan bahasa program yang berbentuk *script* yang diletakkan di dalam *server web*. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan di *server*.

## 9. Xampp

Xampp merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh sebuah komunitas *Open Source* (Nugroho, 2009:74). Ada beberapa aplikasi pendukung *Xampp*, yaitu *PhpMyadmin*, PHP dan MySQL *PhpMyadmin* adalah sebuah *software* yang berbentuk seperti halaman situs yang terdapat pada web *server*. Fungsi halaman ini adalah sebagai pengendali database MySQL sehingga pengguna MySQL tidak perlu *report* untuk menggunakan perintah-perintah SQL. Karena dengan adanya halaman ini semua hal tersebut dapat dilakukan hanya dengan mengklik menu fungsi yang ada pada *phpMyadmin*.

Menurut Nugroho (2009:88), “*PhpMyAdmin* adalah suatu *Open Source* yang berbasis *web*, aplikasi ini dibuat menggunakan program PHP, fungsi aplikasi ini adalah untuk mengakses *Database MySQL*”. Dengan adanya aplikasi ini akan sangat mempermudah dan mempersingkat kerja kita dalam mengolah *Database MySQL*.

Menurut Muhammad Sadeli (2014:10) MySQL adalah database yang menghubungkan script php menggunakan perintah *query* dan *escaps character* yang sama dengan PHP. Menurut Nugroho (2009:91), definisi MySQL adalah sebagai berikut : MySQL (*My Structured Query Language*) adalah sebuah program pembuatan dan pengelola *Database* atau yang disering disebut dengan DBMS (*Database Management System*). Kelebihan lain dari MySQL adalah menggunakan bahasa *query* standar SQL. SQL adalah suatu bahasa permintaan yang terstruktur. *MySQL* dapat digunakan untuk mengelola *database* mulai dari yang kecil sampai dengan yang sangat besar. Hingga saat ini, *MySQL* sudah berkembang hingga versi 5. *MySQL 5* sudah mendukung *trigger* untuk memudahkan pengelolaan tabel dalam *database*.

## E. Desain User Interface

Menurut Ben Shneiderman dalam bukunya *Designing the User Interface* terdapat delapan aturan dasar yang dapat membantu merancang desain interface, yang biasa disebut *eight golden rules of interface design*. Kedelapan aturan tersebut antara lain sebagai berikut:

- a) *Strive for consistency*. Konsistensi dilakukan pada urutan tindakan, perintah, dan istilah yang digunakan pada prompt, menu, serta layar bantuan.
- b) *Enable frequent users to use shortcuts*. Ada kebutuhan dari pengguna yang sudah ahli untuk meningkatkan kecepatan interaksi, sehingga diperlukan singkatan, tombol fungsi, perintah tersembunyi, dan fasilitas makro.
- c) *Offer informative feedback*. Untuk setiap tindakan operator, sebaiknya disertakan suatu sistem umpan balik. Untuk tindakan yang sering dilakukan dan tidak terlalu penting, dapat diberikan umpan balik yang sederhana. Tetapi ketika tindakan merupakan hal yang penting, maka umpan balik sebaiknya lebih substansial. Misalnya muncul suatu suara ketika salah menekan tombol pada waktu input data atau muncul pesan kesalahannya.
- d) *Design dialog to yield closure*. Urutan tindakan sebaiknya diorganisir dalam suatu kelompok dengan bagian awal, tengah, dan akhir. Umpan balik yang informatif akan memberikan indikasi bahwa cara yang dilakukan sudah benar dan dapat mempersiapkan kelompok tindakan berikutnya.
- e) *Offer simple error handling*. Sedapat mungkin sistem dirancang sehingga pengguna tidak dapat melakukan kesalahan fatal. Jika kesalahan terjadi, sistem dapat mendeteksi kesalahan dengan cepat dan memberikan mekanisme yang sederhana dan mudah dipahami untuk penanganan kesalahan.
- f) *Permit easy reversal of actions*. Hal ini dapat mengurangi kekuatiran pengguna karena pengguna mengetahui kesalahan yang dilakukan



dapat dibatalkan; sehingga pengguna tidak takut untuk mengeksplorasi pilihan-pilihan lain yang belum biasa digunakan.

- g) *Support internal locus of control*. Pengguna ingin menjadi pengontrol sistem dan sistem akan merespon tindakan yang dilakukan pengguna daripada pengguna merasa bahwa sistem mengontrol pengguna. Sebaiknya sistem dirancang sedemikian rupa sehingga pengguna menjadi inisiator daripada responden.
- h) *Reduce short-term memory load*. Keterbatasan ingatan manusia membutuhkan tampilan yang sederhana atau banyak tampilan halaman yang sebaiknya disatukan, serta diberikan cukup waktu pelatihan untuk kode, mnemonic, dan urutan tindakan.

## **F. Unified Modelling Language (UML)**

UML singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. Menurut Choniles dalam Prabowo dan Herlawati (2011: 6) mengatakan bahwa UML merupakan bahasa yang berarti bahwa UML memiliki sintaks dan semantik.

### **1. Use Case Diagram**

Menurut Hanif (2007: 91) *use case* adalah metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks. *Use case* dikembangkan oleh analis sistem bersama-sama dengan pengguna. Menurut Pooley dalam Prabowo dan Herlawati (2011: 16) model *use case* dapat dijabarkan dalam diagram *use case*, tetapi diagram tidak identik dengan model karena model lebih luas dari diagram. Secara umum *use case* menggambarkan *external view* dari sistem yang akan kita buat modelnya. Disamping itu *use case* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. (Pilone, dalam Prabowo dan Herlawati, 2011: 21).

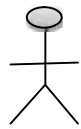

Terdapat beberapa komponen pembentuk diagram *use case* diantaranya :

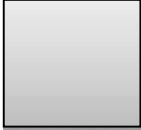
- a) Aktor (*actor*), menggambarkan pihak-pihak yang berperan dalam sistem.
- b) *Use case*, aktivitas atau sarana yang disediakan oleh sistem.
- c) Hubungan (*link*), aktor mana saja yang terlibat dalam *use case* ini.

*Use case* sangat menentukan karakteristik sistem yang kita buat, oleh karena itu Chonoles dalam Probowo dan Herlawati (2011: 22) menawarkan cara untuk menghasilkan *use case* yang baik yakni:

- a) Pilihlah nama yang baik. *Use case* adalah sebuah *behavior* (perilaku), jadi seharusnya dalam frase kata kerja.
- b) Ilustrasikan perilaku dengan lengkap. *Use case* dimulai dari inisiasi oleh aktor primer dan berakhir pada aktor dan menghasilkan tujuan.
- c) Identifikasi perilaku dengan lengkap. Untuk mencapai tujuan dan menghasilkan nilai tertentu dari aktor, *use case* harus lengkap.
- d) Menyediakan *use case* lawan. Kita biasanya membutuhkan *use case* yang membatalkan tujuan, misalnya pada *Use case* pemesanan kamar dibutuhkan pula *use case* pembatalan pemesanan kamar.
- e) Batasi *use case* hingga satu perilaku saja. Guna menghindari kerancuan, jagalah *use case* kita hanya fokus pada satu hal.
- f) Nyatakan *use case* dari sudut pandang aktor. Tulislah *use case* dari sudut pandang aktor bukan sistem.

**Tabel 1. Simbol-simbol dalam Use case**

No	Simbol	Nama	Keterangan
1		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		Use case	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang

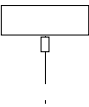


			menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
3		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
4	----- >	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> )
5	←	Generalization	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
6	----- >	Include	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i>
7	←	Extent	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
8	—	Assosiation	Menghubungkan antara objek satu dengan objek yang lainnya

(Nugroho, Adi : 2005)

## 2. Sequence Diagram

*Sequence diagram* menggambarkan interaksi antara objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu, menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. (Komala, 2015). Berikut adalah beberapa simbol didalam *sequence diagram*:






**Tabel 2. Simbol-Simbol Sequence Diagram**

No.	Simbol	Nama	Keterangan
1		<i>Object</i> dan <i>lifeline</i>	Orang, tempat, benda, kejadian atau konsep yang ada dalam dunia nyata yang penting bagi suatu aplikasi yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.
3		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.

## 3. Activity Diagram

Diagram aktivitas (*Activity Diagram*) menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi. Diagram aktivitas ini juga merupakan salah satu cara pemodelan *event-event* yang terjadi dalam *use case*.

**Tabel 3. Notasi Pada Activity Diagram**

Notasi	Nama	Keterangan
	<i>Start State</i>	Menandakan status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan/ <i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	Penggabungan/ <i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	<i>Stop State</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

(Nugroho, Adi : 2005)




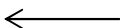
#### 4. Class Diagram

Diagram kelas (*Class Diagram*) merupakan inti dari proses pemodelan objek. Baik *forward engineering* maupun *reverse engineering*. *Forward engineering* adalah proses perubahan model menjadi kode program sedangkan *reverse engineering* adalah proses sebaliknya yaitu merubah kode program menjadi model. Diagram kelas juga merupakan kumpulan kelas-kelas objek. Menurut Whitten dalam Probowo dan Herlawati (2011: 39) kelas adalah satu set objek

yang memiliki atribut dan perilaku yang sama. Secara umum *class* memiliki tiga komponen pokok yaitu:

- a) Nama merupakan nama dari sebuah *class*.
- b) Atribut adalah properti dari sebuah *class*. Atribut melambangkan batas nilai mungkin ada pada *object* dari *class*
- c) *Method* merupakan sesuatu yang bisa dilakukan oleh sebuah *class* lain terhadap sebuah *class*.

**Tabel 4. Notasi Pada Class Diagram**

No	Simbol	Nama	Keterangan
1		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2		Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		Asosiasi	Hubungan statis antar <i>class</i> yang menggambarkan <i>class</i> yang memiliki atribut dengan <i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain.
4		Generalization	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).

5	----- >	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
---	---------	------------	--

(Nugroho, Adi : 2005)

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **A. Analisis Masalah**

Analisis masalah berguna untuk mengetahui berbagai masalah yang ada sehingga dengan adanya aplikasi ini diharapkan dapat menyelesaikan permasalahan yang dihadapi. Adapun beberapa masalah tersebut adalah sebagai berikut:

1. Belum adanya aplikasi lokasi hotel di Kota Padang Panjang berbasis android.
2. Masih banyaknya wisatawan yang belum mengetahui rute menuju hotel yang ingin dituju.

#### **B. Analisis Aplikasi**

Dalam pembuatan aplikasi perangkat lunak dibutuhkan adanya suatu analisa dan perancangan dari sistem yang akan dibuat. Ini bertujuan agar aplikasi yang akan dibuat dapat selesai tepat waktu dan sesuai dengan tujuan awal. Kesalahan dalam analisa akan berimbas pada tahapan proyek selanjutnya dan mengakibatkan aplikasi tidak maksimal atau tidak sesuai dengan tujuan pembuatan aplikasi.

Tahapan analisis ini berisi identifikasi dan evaluasi permasalahan yang ada serta kebutuhan apa saja yang dibutuhkan dalam pembuatan aplikasi perangkat lunak. Dengan adanya analisa dan perancangan maka akan didapatkan suatu gambaran mengenai kebutuhan apa saja yang dibutuhkan dalam program aplikasi ini. Langkah-langkah dalam penggunaan aplikasi dari awal hingga akhir adalah sebagai berikut.

1. user menjalankan aplikasi kehalaman utama
2. Di halaman utama terdapat halaman berita dan 2 menu utama, menu tersebut adalah mencari hotel dan menampilkan peta hotel.



3. Jika user memilih tombol cari hotel maka user akan melihat daftar hotel. Jika user memilih salah satu daftar hotel maka akan muncul keterangan dari hotel yang telah dipilih.
4. Jika user memilih tombol Lihat Peta maka user akan melihat seluruh lokasi hotel yang ada di Kota Padang Panjang.
5. Jika user menekan *icon* di sebelah kiri atas halaman utama. Pada menu ini akan menampilkan menu Home, Tentang, dan keluar.
6. Jika user menekan home maka user akan kembali kehalam utama.
7. Jika user menekan tentang maka user akan melihat penjelasan tentang aplikasi.
8. Jika user menekan Keluar maka user akan keluar dari aplikasi.

### C. Analisis Tools

Analisis *tools* dilakukan untuk menganalisis kebutuhan perangkat yang akan digunakan dalam pengembangan sistem. Terdapat dua kategori perangkat, yaitu analisis kebutuhan perangkat keras dan analisis kebutuhan perangkat lunak.

- a. Analisis Kebutuhan Perangkat Keras
  - 1) Processor Intel Core i3 atau versi lain
  - 2) RAM 2 GB
  - 3) VGA Card
  - 4) USB
  - 5) Mouse dan Keyboard
  - 6) *Smartphone* android untuk pengujian aplikasi yang akan dibuat minimal versi 2.3 (*Gingerbread*)
- b. Analisis Kebutuhan Perangkat Lunak
  - 1) Sistem Operasi Windows 7 atau versi lainnya
  - 2) Android Studio
  - 3) Java Development Kit (JDK)
  - 4) Plugin Android Development Tool (ADT)
  - 5) Android System Development Kit (SDK)

## D. Perancangan Sistem

Perancangan sistem dilakukan untuk menggambarkan, merencanakan, dan membuat sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Perancangan system merupakan hasil transformasi dari analisis ke dalam perancangan yang nantinya akan diimplementasikan. Hal penting yang menjadi perhatian pada tahap ini adalah rancangan yang dibuat diharapkan dapat digunakan dengan mudah oleh semua user.

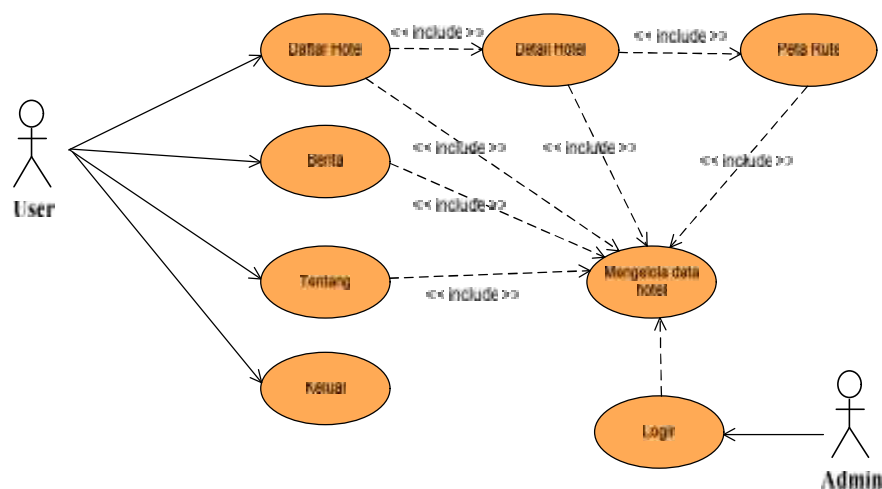
### 1. Use Case Diagram

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use Case Diagram* dari sistem yang dirancang dapat dilihat pada tabel dibawah ini.

**Tabel 5. Tabel Peran Aktor**

No	Actor	Peran
1	Admin	Aktor yang mengelola semua sistem.
2	User (Pengguna)	<ul style="list-style-type: none"> <li>a. Melihat daftar hotel yang ada di Kota Padang Panjang</li> <li>b. Memilih Hotel yang ingin dikunjungi</li> <li>c. Melihat keterangan dari hotel yang dipilih</li> <li>d. Melihat rute yang harus ditempuh user dari lokasi saat ini menuju lokasi hotel yang dipilih</li> <li>e. Melihat Berita hotel</li> <li>f. Melihat tentang aplikasi.</li> </ul>

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use Case Diagram* dari sistem yang dirancang dapat dilihat pada tabel dibawah ini.



**Gambar 3. Use case diagram**

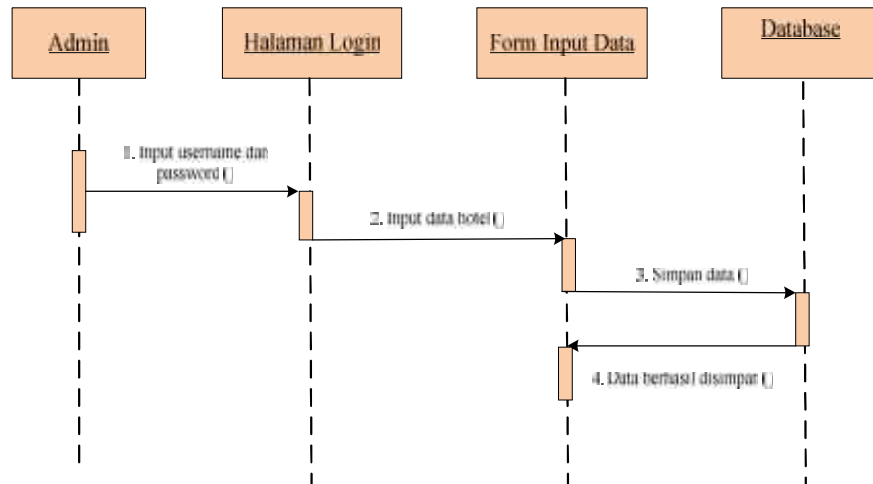
Pada gambar dapat dilihat bahwa *user* dapat melihat informasi mengenai hotel di Kota Padang Panjang, berupa informasi hotel, berita, tentang, serta lokasi geografisnya berdasarkan koordinat yang telah dientrikan pada *google maps*, sedangkan admin melakukan login dan mengelola dan mengupdate semua data hotel.

## 2. Sequence Diagram

### a. Sequence diagram Admin

Pada gambar dibawah menjelaskan alur ketika admin melakukan login. Setelah masuk halaman aplikasi admin mengentrikan username dan password. Setelah login berhasil

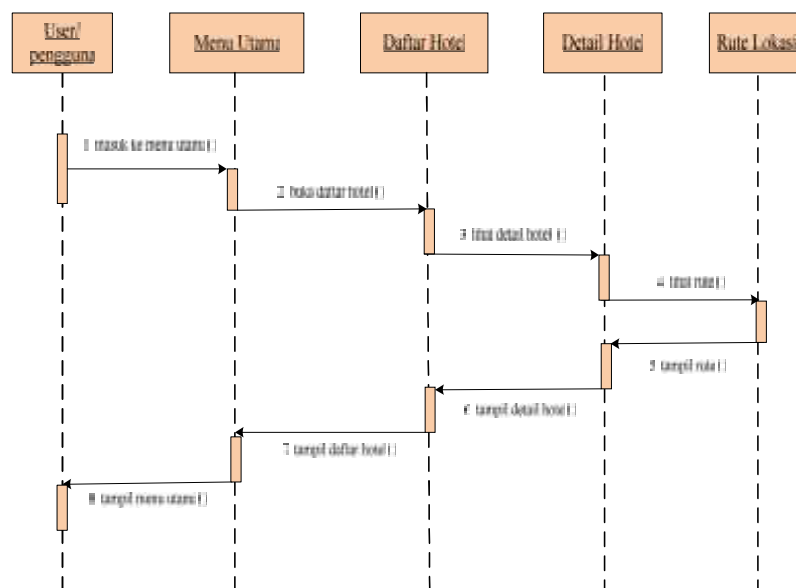
admin dapat menginputkan data-data hotel dan disimpan kedalam database.



**Gambar 4. Sequence diagram Admin**

b. *Sequence diagram User (Pengguna)*

Pada gambar dibawah dapat dilihat bahwa user masuk kehalaman utama, setelah masuk user dapat melihat daftar hotel, detail hotel, serta rute lokasi hotel yang dituju.



**Gambar 5. Sequence diagram User (Pengguna)**

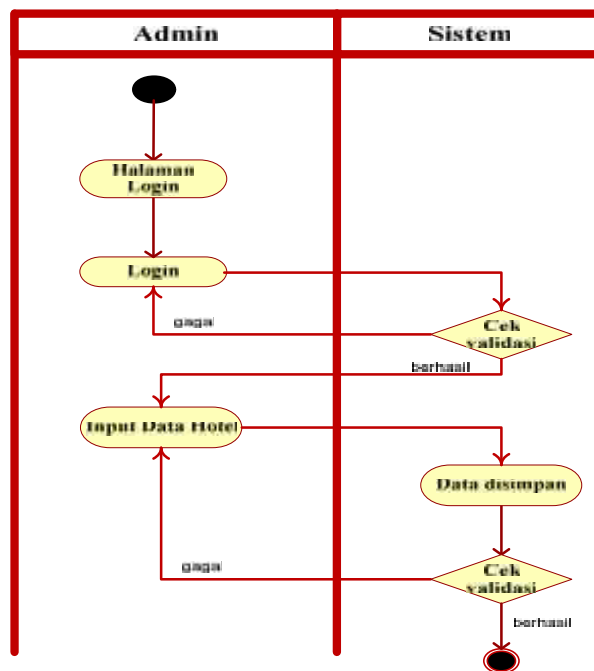
### 3. Activity Diagram

Diagram aktivitas (*Activity Diagram*) menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi. Diagram aktivitas ini juga merupakan salah satu cara pemodelan *event-event* yang terjadi dalam *use case*.

Berikut ini adalah *activity diagram* pada aplikasi lokasi hotel di Kota Padang Panjang.

#### a) *Activity Diagram* Admin

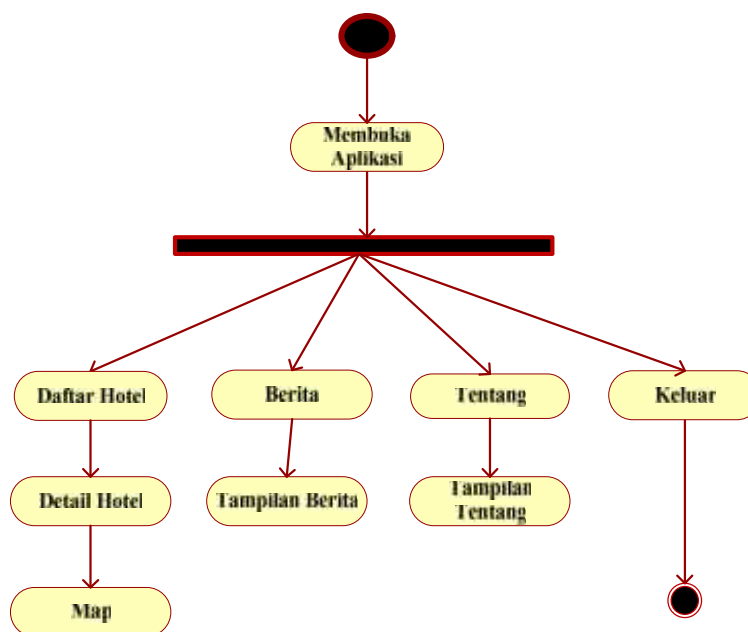
Admin masuk ke halaman utama kemudian login, jika login gagal maka admin harus kembali melakukan login. Jika login berhasil, maka admin dapat mengelola data hotel. Jika data gagal disimpan, maka admin melakukan input data kembali. Jika data berhasil tersimpan, maka data akan tersimpan kedalam database.



**Gambar 6.** *Activity diagram* Admin

b) *Activity Diagram* user

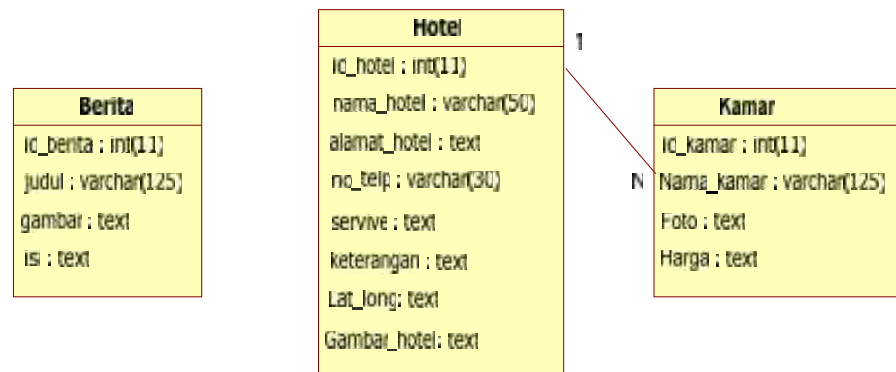
User memulai dari membuka aplikasi. Setelah user dapat melihat daftar hotel, detail hotel, berita, tentang, serta lokasi dari hotel tersebut.



**Gambar 7.** *Activity diagram* user

#### 4. Class Diagram

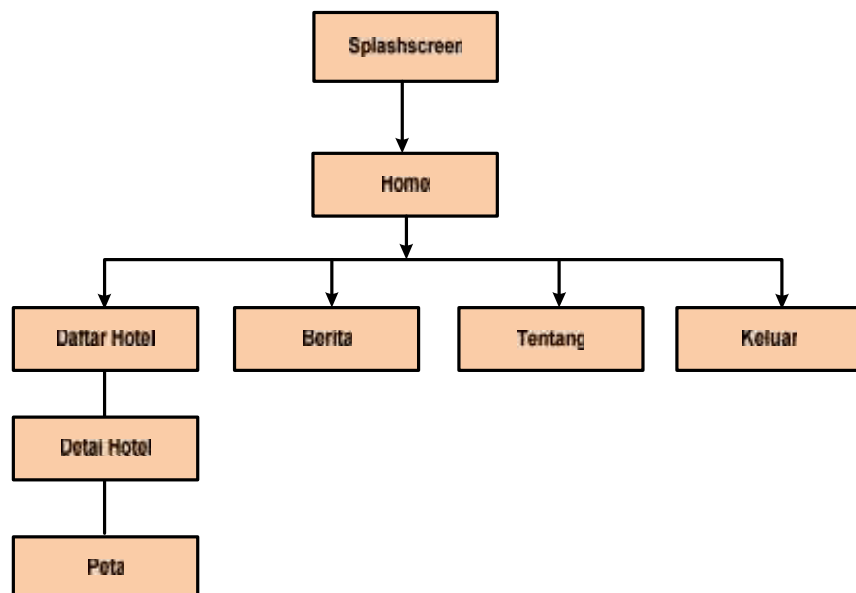
Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut yaitu variabel-variabel yang dimiliki oleh suatu kelas, dan metode atau operasi, yaitu fungsi-fungsi yang dimiliki suatu kelas. Dari gambar di bawah menunjukkan hubungan antar class dengan tabel dalam sistem yang sedang dibangun dan bagaimana saling berkolaborasi untuk mencapai satu tujuan. Perancangan *class diagram* untuk Aplikasi Lokasi Hotel di Kota Padang Panjang, adalah sebagai berikut :



**Gambar 8. Class diagram**

### E. Struktur Program

Perancangan struktur program utama terdiri dari daftar menu atau pilihan yang akan digunakan. Struktur program bertujuan agar pengguna dapat menggunakan sistem dengan mudah dan mengerti apa saja menu yang ada pada sistem. Masing-masing antarmuka mempunyai struktur program tersendiri, berikut ini adalah rancangan struktur antarmuka tersebut.



**Gambar 9. Struktur Program**

## F. Desain Output Sistem

### 1. Desain Form Awal

Dalam form awal terdapat tampilan splash screen yang berisi nama aplikasi dan logo aplikasi, seperti gambar dibawah:



*Gambar 10. Desain awal form*

### 2. Desain Halaman Utama

Halaman utama pada aplikasi ini menampilkan menu-menu untuk pengguna. Menu tersebut adalah mencari hotel dan menampilkan peta hotel, dan berita tentang hotel. Perancangan desain menu utama aplikasi ditunjukkan pada Gambar dibawah.



*Gambar 11. Desain halaman utama*



### 3. Desain Menu *Drawer*

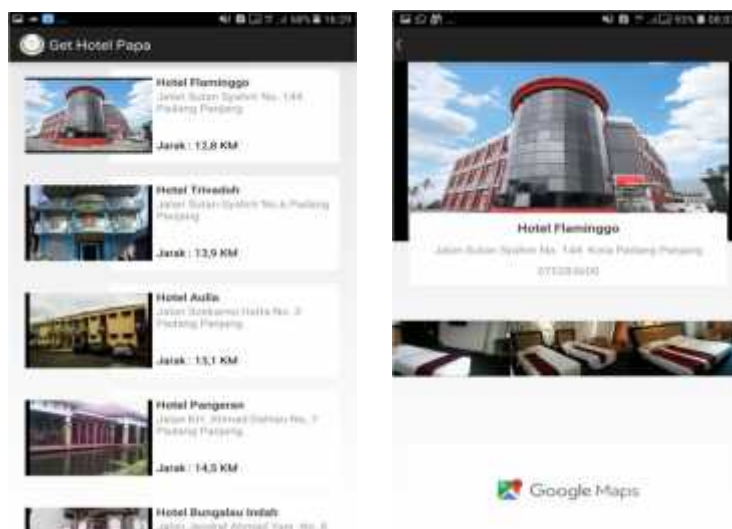
Menu *drawer* dapat ditekan pengguna dengan menekan *icon* di sebelah kiri atas halaman utama. Pada menu ini akan menampilkan menu Home, Tentang, dan Keluar.

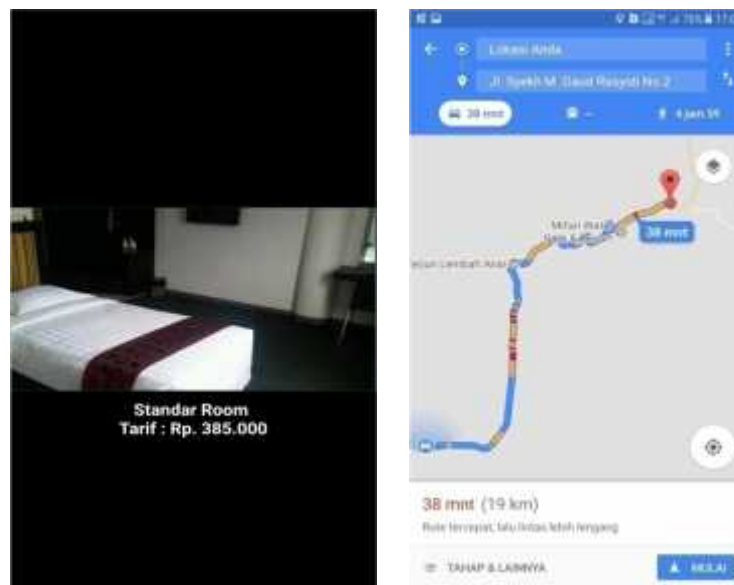


**Gambar 12.** Menu *drawer*

### 4. Desain Mencari Hotel pada Menu Utama

Pada menu ini akan langsung menampilkan daftar hotel di Kota Padang Panjang. Ketika pengguna memilih salah satu hotel maka akan langsung menampilkan deskripsi hotel, alamat hotel, no telp, jalur navigasi dari lokasi pengguna berada menuju lokasi hotel tersebut. Perancangan daftar hotel ditunjukkan pada Gambar dibawah.





*Gambar 13. Desain Mencari Hotel pada Menu Utama*

## 5. Desain Lihat Peta pada Menu Utama

Ketika pengguna memilih menu ini, maka sistem akan menampilkan seluruh lokasi hotel yang ada di padang panjang. Perancangan desain peta hotel ditunjukkan pada Gambar dibawah.



*Gambar 14. Desain Lihat Peta pada Menu Utama*

## 6. Desain Menu Tentang

Pada menu ini menampilkan keterangan tentang aplikasi. Perancangan desain menu tentang aplikasi ditunjukkan pada Gambar dibawah.



*Gambar 15. Desain Menu Tentang*

## G. Desain Input

### 1. Input Data Hotel

Nama Hotel	<input type="text" value="Flamingo"/>
Alamat	<input type="text"/>
Service	<input type="text"/>
Ex: Laundry, Pelayanan 24 jam	
Number Telepon	<input type="text" value="075411345"/>
Keterangan	<input type="text"/>

Latitude dan Longitude

Gambar

 No file selected.

*Gambar 16. Input Data Hotel*

## 2. Input Berita

Judul berita

Gambar

 No file selected.

Isi

*Gambar 17. Input Berita*

### 3. Input Kamar

**Gambar 18. Input Kamar**

## H. Struktur Tabel

Struktur tabel merupakan *database* dari aplikasi lokasi hotel. Tabel-tabel tersebut merupakan tempat penyimpanan semua data yang diperlukan pada sistem ini. Struktur tabel yang sesuai dengan kebutuhan sebagai media penyimpanan dari *entry* data adalah sebagai berikut:

### 1. Tabel Hotel

Database Name : db\_hotelpadangpanjang

Table Name : tb\_hotel

Field Key : id\_hotel

**Tabel 6. Tabel Hotel**

Field	Type	Description
id_hotel	Int	Id Hotel
nama_hotel	Varchar	Nama Hotel
alamat_hotel	text	Alamat Hotel
No_telp	Varchar	No Telp
service	text	Service
keterangan	text	Keterangan
lat, long	text	Latitude, Longitude
gambar_hotel	text	Gambar Hotel

## 2. Tabel Berita

Database Name : db\_hotelpadangpanjang

Table Name : tb\_berita

Field Key : id\_berita

***Tabel 7. Tabel Berita***

Field	Type	Description
Id_berita	int	Id Berita
judul	Varchar	Judul
gambar	text	Gambar
isi	text	Isi

## 3. Tabel Kamar

Database Name : db\_hotelpadangpanjang

Table Name : tb\_kamar

Field Key : id\_kamar

***Tabel 8. Tabel Kamar***

Field	Type	Description
Id_kamar	int	Id Kamar
nama	Varchar	Nama
foto	text	Foto
harga	text	Harga

## **BAB IV**

### **PENUTUP**

#### **A. Kesimpulan**

Berdasarkan hasil perancangan aplikasi lokasi hotel di Kota Padang Panjang berbasis Android dapat disimpulkan sebagai berikut :

1. Dengan adanya aplikasi ini dapat membantu masyarakat dalam pencarian hotel tanpa harus tersesat.
2. Aplikasi ini dapat di akses oleh semua kalangan *user* yang mempunyai *smartphone* dengan sistem operasi *android*.
3. Aplikasi yang di buat ini dapat menampilkan informasi dan peta lokasi geografis letak lokasi hotel yang berada di wilayah kota Padang Panjang pada perangkat *android*.
4. Dengan adanya aplikasi ini dapat membantu pemerintah Kota Padang dalam mempromosikan hotel-hotel yang ada pada wilayah Kota Padang.

#### **B. Saran**

Dalam pembuatan aplikasi lokasi hotel di Padang Panjang pada *Mobile Android* ini tidak semua hotel yang terdapat di Padang Panjang penulis masukkan kedalam aplikasi ini. Masih banyak kekurangan yang perlu Penulis menyarankan :

1. Semoga untuk kedepannya aplikasi ini dikembangkan lebih baik dari sekarang.
2. diharapkan aplikasi ini dapat dikembangkan lagi dalam skala yang lebih luas yaitu provinsi.

## DAFTAR PUSTAKA

- Prahasta, Eddy. 2002. *Konsep-konsep Dasar Sistem Informasi Geografis*. Bandung: Informatika.
- Nugroho, A. 2005. *Analisis Dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek*. Bandung: Informatika.
- Kurniawan, Zaini. 2016. *Sistem Informasi Geografis Pencarian Lokasi Penginapan Dikota Bukittinggi*. Batusangkar: Jurusan Manajemen Informaika, IAIN Batusangkar.
- Ashadi, Eki. 2015. *Sistem Informasi Geografis Parawisata Kabupaten Tanah Datar Berbasis Android Menggunakan Eclipse Indigo*. Batusangkar : STAIN Batusangkar.
- Rusdyanto, Fajar. 2014. *Aplikasi WebGIS Hotel Di Kota Surakarta Memanfaatkan Google Maps API*. Volume 3, Nomor 1, Tahun 2014. Diambil dari : Jurnal Geodesi Undip (17 Januari 2018).
- Santoso, Kartika Imam. 2015. *Implementasi Sistem Informasi Geografis Daerah Pariwisata Kabupaten Temanggung Berbasis Android dengan Global Positioning System (GPS)*. Vol. 2, No. 1, Mei 2015. Diambil dari : <http://journal.unnes.ac.id/nju/index.php/sji> (17 Januari 2018).
- Hati, Gunita Mustika. 2013. *Aplikasi Penanda Lokasi Peta Digital Berbasis Mobile Gis Pada Smartphone Android*. Volume 2, Nomor 4, Tahun 2013. Diambil Dari : Jurnal Geodesi Undip (17 Januari 2018).
- Yesiana, Anisa Isna. 2016. *Aplikasi Sistem Informasi Geografis (Sig) Persebaran Hotel Di Kota Semarang Berbasis Web*. Volume 5, Nomor 2, Tahun 2016. Diambil Dari : Jurnal Geodesi Undip (17 Januari 2018).



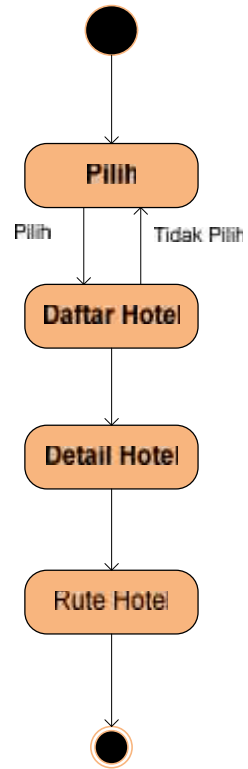
[https://id.wikipedia.org/wiki/Kota\\_Padang\\_Panjang#Geografi](https://id.wikipedia.org/wiki/Kota_Padang_Panjang#Geografi) di ambil 17 Januari  
2018

[https://id.wikipedia.org/wiki/Kota\\_Padang\\_Panjang](https://id.wikipedia.org/wiki/Kota_Padang_Panjang) di ambil 17 Januari 2018

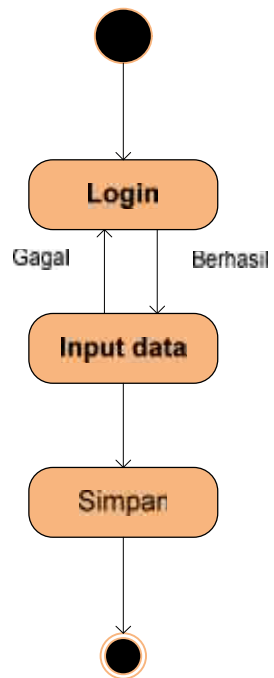
<https://maps.google.com> di ambil 17 Januari 2018

## STATECHART DIAGRAM

### 1. Statechart Diagram User / Pengguna



### 2. Statechart Diagram Admin





**KEMENTERIAN AGAMA REPUBLIK INDONESIA**  
**INSTITUT AGAMA ISLAM NEGERI BATUSANGKAR**  
**LEMBAGA PENELITIAN DAN PENGABDIAN MASYARAKAT**

Jl. Sudirman No.137 Kuburaja Lima Kaum Batusangkar 27213, Telp. (0752) 71150, Ext 135, Fax. (0752) 71879  
Website : www.iainbatusangkar.ac.id e-mail: lppm@iainbatusangkar.ac.id

02 Februari 2018

Nomor : B- 249 /In.27/L.I/TL.00/ 02 /2018

Sifat : Biasa

Lampiran : 1 Rangkap

Perihal : **Mohon Penerbitan Surat Izin Penelitian**

Yth. Walikota Padang Panjang

Up. Kepala Dinas Penanaman Modal dan PTSP Kota Padang Panjang

Padang Panjang

*Assalamu'alaikum Wr. Wb.*

Dengan hormat,

Bersama ini disampaikan kepada Bapak/Ibu bahwa mahasiswa yang tersebut di bawah ini:

Nama/NIM : DIVIKA ROZALIA / 14205031  
Tempat/Tanggal Lahir : Titian Panjang, 01 Februari 1995  
Nomor Induk Keluarga : KTP. 1305044102950002  
Fakultas : Ekonomi dan Bisnis Islam  
Jurusan : Manajemen Informatika  
Alamat : Titian Panjang, Kayu Tanam Nagari Kayu Tanam Kecamatan 2 X 11 Kayu Tanam Kabupaten Padang Pariaman

akan melakukan pengumpulan data untuk proses penulisan laporan hasil penelitiannya sebagai berikut:

Judul Penelitian : **Sistem Informasi Geografis Pemetaan Hotel dan Klasifikasi Hotel di Kota Padang Panjang Berbasis Android**  
Lokasi : Kantor Dinas Pariwisata Kota Padang Panjang dan Hotel di Kota Padang Panjang  
Waktu : 02 Februari 2018 s.d 02 Maret 2018  
Dosen Pembimbing 1 : Lita Sari Muchlis, M.Kom.  
Dosen Pembimbing 2 : -

untuk itu, diharapkan kiranya Bapak/Ibu berkenan menerbitkan surat izin penelitian dalam rangka pelaksanaan penelitian mahasiswa yang bersangkutan.

Demikian disampaikan, atas bantuannya diucapkan terimakasih.



Ketua,

Yusrizal Efendi, S.Ag., M.Ag.

**Tembusan:**

1. Rektor IAIN Batusangkar (Sebagai Laporan).
2. Dekan Fakultas Ekonomi dan Bisnis Islam IAIN Batusangkar (Sebagai Laporan).



**DINAS PENANAMAN MODAL DAN  
PELAYANAN TERPADU SATU PINTU**

Jl. Sultan Syahrir No. 150 Kelurahan Silaing Bawah Kota Padang Panjang Kode Pos 27118  
Email : kppt\_padangpanjang@yahoo.co.id tlp (0752) 485395 fax (0752)485645

**REKOMENDASI MELAKSANAKAN PENELITIAN**  
NOMOR : 070/xy/DPM&PTSP-PP/II-2018

Setelah mempelajari Surat Ketua Lembaga Penelitian dan Pengabdian Masyarakat IAIN Batusangkar Nomor: B-249/In.27/LI/TL.00/02/2018 tanggal 02 Februari 2018 perihal Surat Izin Penelitian, dengan ini kami menyatakan tidak keberatan atas maksud melaksanakan Penelitian/Permintaan Data di Kota Padang Panjang yang dilakukan oleh:

Nama	: Divika Rozalia
Tempat/Tanggal Lahir	: Titian Panjang / 01 Februari 1995
Alamat	: Titian Panjang, Kayu Tanam, Nagari Kayu Tanam Kec. 2 X 11 Kayu Tanam, Kab. Padang Pariaman
No. Kartu Identitas	: 1305044102950002
No. Telepon/HP	: 081374201025
Pekerjaan	: Mahasiswa
Judul Penelitian	: Sistem Informasi Geografis Pemetaan Hotel dan Klasifikasi Hotel di Kota Padang Panjang Berbasis Android
Lokasi Penelitian	: Dinas Pariwisata Kota Padang Panjang dan Hotel di Padang Panjang
Lama Penelitian	: 02 Februari 2018 s.d 02 Maret 2018

Dalam rangka mengadakan penelitian, peneliti diwajibkan/diharuskan memenuhi ketentuan sebagai berikut:

1. Peneliti harus melaporkan diri kepada pihak yang berwenang di tempat penelitian
2. Penelitian dilakukan sesuai dengan ketentuan yang berlaku di tempat penelitian
3. Peneliti diwajibkan menyampaikan laporan hasil penelitian kepada Walikota Padang Panjang melalui Dinas Penanaman Modal dan Pelayanan Terpadu Satu Pintu Kota Padang Panjang.
4. Peneliti dilarang melakukan penelitian diluar hal-hal yang telah ditetapkan oleh pihak yang berwenang di tempat penelitian.
5. Apabila penelitian yang dilakukan tidak sesuai dengan ketentuan tersebut diatas, maka rekomendasi penelitian dicabut kembali.

Demikianlah rekomendasi penelitian ini diberikan kepada yang bersangkutan untuk dapat dipergunakan sebagaimana mestinya.

Padang Panjang, 02 Februari 2018

**An. WALIKOTA PADANG PANJANG**  
Kepala Dinas Penanaman Modal dan  
Pelayanan Terpadu Satu Pintu,



**EWA BGSKA, SH**  
NIP.19750103 200003 1 003



1. Kepala BPBD Kesbangpol Kota Padang Panjang;
2. Kepala Dinas Pendidikan Kota Padang Panjang;
3. Ketua Lembaga Penelitian dan Pengabdian Masyarakat IAIN Batusangkar;



**PEMERINTAH KOTA PADANG PANJANG**  
**DINAS PARIWISATA**

Komplek Pusat Dokumentasi dan Informasi Kebudayaan Minangkabau (PDIKM) Silaing  
Bawah Telepon/Fax. (0752) 84164 Padang Panjang, email [porbudparpp@gmail.com](mailto:porbudparpp@gmail.com)  
Kode Pos 27118

Nomor : 556/ 58 /Pariwisata-PP/II-2018  
Lamp : -  
Hal : Surat Keterangan

Kepada Yth,  
Dekan Fakultas Ekonomi dan Bisnis Islam  
pada IAIN Batusangkar

Dengan hormat,

Menindak lanjuti surat Bapak No.B-249/In.27/L.I/TL.00/02/2018 Perihal pengantar pengambilan data untuk Tugas Akhir, dengan ini menerangkan bahwa :

Nama : DIVIKA ROZALIA  
NIM : 14 205 031  
Jurusan : Manajemen Informatika  
Fakultas : Ekonomi dan Bisnis Islam

Telah mengambil data untuk penyusunan Tugas Akhir dengan judul "**SISTEM INFORMASI GEOGRAFIS PEMETAAN HOTEL DAN KLASIFIKASI HOTEL DI KOTA PADANG PANJANG BERBASIS ANDROID**".

Demikianlah surat keterangan ini disampaikan, atas perhatiannya diucapkan terima kasih.

Padang Panjang, 28 Februari 2018

Kepala



**HENDRI FAUZAN, AP.M.Si**

NIP. 19740409 199501 1 001



KEMENTERIAN AGAMA  
INSTITUT AGAMA ISLAM NEGERI  
BATUSANGKAR

PROGRAM DIPLI MANAJEMEN INFORMATIKA

Jl. Sadirman No. 177 Kubo Rajo Lama Karuh Batusangkar 27213 Telp. (0752) 71150, 574221, Fax. (0752) 71879  
<http://www.iainbatusangkar.ac.id> e-mail: [mb@iainbatusangkar.ac.id](mailto:mb@iainbatusangkar.ac.id)

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

KARTU BIMBINGAN PENULISAN TUGAS AKHIR

NIM : 14.205.031  
Nama : Divika Rozalia  
Jurusan : D.III Manajemen Informatika  
Dosen Pembimbing : Lita Sari Muchlis, M.Kom  
Judul Tugas Akhir : Sistem Informasi Geografis Lokasi Hotel di Kota Padang Panjang berbasis Android

NO	Hari/Tanggal	Materi Bimbingan	Paraf
1.	26/12/2017	Konsultasi BAB I - BAB III	
2.	9/01/2018	Perbaikan BAB I	
3.	23/01/2018	Perbaikan BAB I - BAB II	
4.	28/01/2018	Perbaikan BAB I - BAB II	
5.	04/02/2018	Perbaikan BAB I - BAB III & PROGRAM	
6.	05/02/2018	Perbaikan BAB I - BAB III & PROGRAM	
7.	08/02/2018	Perbaikan PROGRAM	
8.	14/02/2018	ACE TA	

Catatan : Setiap konsultasi dengan dosen pembimbing kartu ini harus dibawa, diisi, dan diparaf oleh dosen pembimbing

Batusangkar, 14 Februari 2018

Mahasiswa

**Divika Rozalia**

Nim. 14.205.031

Dosen Pembimbing Tugas Akhir

**Lita Sari Muchlis, M.Kom**

NIP. 19780112 200901 2 017

Mengetahui,

Dosen Penasehat Akademik

**Lita Sari Muchlis, M.Kom**

NIP. 19780112 200901 2 017

**DAFTAR NAMA HOTEL/PENGINAPAN/WISMA  
DI PADANG PANJANG**

NO	NAMA HOTEL	PEMILIK	PENGURUS	TELP/ALAMAT
1.	Mifan Cottage	H.Nelson Septiadi, STM	Hasan Basri	Komp.Objek Wisata Mifan Waterpark, Jl.St.Syahril Silaing Bawah Padang Panjang, Telp. (0752) 484272
2.	Wisma Pangeran	Nurjanis	Peli Amri	Jl.K.H Ahmad Dahlan No.7 Padang Panjang Telp. 0752 – 82171
3.	Tryvadoh Hotel	Riva	Riva	Jl. St.Syahril Padang Panjang Telp.(0752) 485279 HP.081363354628
4.	Aulia Hotel 1	Lisnur	Denison	Jl. Soekarno Hatta No.3 Padang Panjang Telp.(0752) 485111
5.	Aulia Hotel 2	Lisnur	Denison	Jl. St.Syahril Padang Panjang Telp.(0752) 84448
6.	Wisma Bundo Kandung	Eli Sovia	Leli	Jl. St.Syahril No. 186 Silaing Bawah Padang Panjang HP. 085274460235
7.	Graha Serambi Meloah	H. Bahar Yusuf	Dadan Suyatna	Jl. A. Yani Padang Panjang HP. 085264375636
8.	Wisma Syahidah Dinniyah Puteri	Pauziah Fauzan	Beni	Jl. Abdul Hamid Hakim No. 30 Kel.Pasar Usang Padang Panjang Telp. (0752)485454
9.	Hotel Hasiba	Hj. Zaimar	Munawarah, SH	Jl.K.H Ahmad Dahlan Padang Panjang Telp. (0752) 82434
10.	Homestay Cahaya Nur	Eli Sovia	Eli Sovia	Jl. St.Syahril Padang Panjang Telp.
11.	Wisma Kereta Api	PT. KAI	Karyawan PT. KAI	Stasiun Pasar Usang Jl. AK.Gani No. 125 Padang Panjang HP. 081266041374
12.	Hotel Bungalow	H. Muslim	DediIndrawan	Jl. A.Yani No. 8 Telp. (0752) 485421 HP. 081374200889
13.	Hotel Makmur	(Alm) Zarmiludin	Raimon	Jl. Imam Bonjol No. 234 Padang Panjang Telp. (0752) 82140 HP. 085355171100
14.	Hotel Flaminggo	H. RiwardiArlin	Chatrul Saleh	Jl. St. Syahrir No.144 Kel. Silaing Bawah Telp. (0752) 83600
15.	Wisma Rabbari	H. Oriza Sativa	Raja	Jl. Soekarno Hatta No.88 A Bukit Surungan Telp. (0752) 485706 HP. 082389812020
16.	Wisma Ummi	Arlinda Cornelia Dewi	Mumum Mulyani	Jl. Soekarno Hatta No.97 Bukit SurunganTelp. (0752) 485706 HP. 081266366359
17.	Wisma Keluarga	Suhaimi	Suhaimi	Jl. Soekarno Hatta No.89 Bukit Surungan HP. 08128649436
18.	Wisma Villa Widya	H. Erman, SH	Joni Indra	Jl. M.NasirNo.20 Kel. Silaing Bawah Telp. 082170348007

19.	Wisma Pesona Muslim	H. Ret	Del	Jl. Soekarno Hatta No.70 Kel. Bukit Surungan HP. 085263697500
20.	Wisma Alia Serambi	Anita	Dasneli	Jl. St.Syahri No. 106 Silaing Bawah Padang Panjang Telp. (0752) 83864
21.	Wisma PDIKM	Dinas Pariwisata Padang Panjang	Dinas Pariwisata Padang Panjang	Komplek PDIKM Kota Padang Panjang Telp. (0752) 84164
22.	Wisma Hayati	Jaya wisata	Jaya wisata	Jl. Soekarno Hatta No.1, Kel. Bukit Surungan
23.	Rumah Liburan 1,2	Zuhaida	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.7 Kubu Gadang Kel. Ekor Lubuk
24.	Rumah Baruah 1,2	Agusnir	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.6 Kubu Gadang Kel. Ekor Lubuk
25.	Home Stay Ratna	Nenek Ani	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.8 Kubu Gadang Kel. Ekor Lubuk
26.	Home Stay Fatimah	Nenek Ani	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.8 Kubu Gadang Kel. Ekor Lubuk
27.	Home Stay Oma-oma	Era Sovia	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.8 Kubu Gadang Kel. Ekor Lubuk
28.	Home Stay Makmi	Erniwati	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.9 Kubu Gadang Kel. Ekor Lubuk
29.	Home Stay Lima Bersaudara	Rasidah	Kelompok Sadar Wisata (Pokdarwis) Kubu Gadang	Jl. Hj. Miskin RT.6 Kubu Gadang Kel. Ekor Lubuk



Padang Panjang, September 2017  
Kepala

HERY FAUZAN, A.P.M.Si  
Pepihina Telp. 0752 HP. 19740409 199501 1 001



## LISTING PROGRAM

### 1. SplashScreen

#### a. SplashActivity.java

```
package com. minangsoft.gethotelpapa;

import android.app.ActionBar;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.WindowManager;

/**
 * Created by divika on 01/12/17.
 */
public class SplashActivity extends Activity {
    private static final int SPLASH_TIME = 3000;
    @Override
    protected void onCreate(Bundle savedInstanceState) {

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULL
SCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);
        ActionBar actionBar = getActionBar();
        actionBar.hide();
        super.onCreate(savedInstanceState);
        setContentView(R.layout.splashscreen);

        new Handler().postDelayed(new Runnable(){
            @Override
            public void run(){
                Intent intent = new
Intent(SplashActivity.this, HomeActivity.class);
                startActivity(intent);
                SplashActivity.this.finish();
            }
        }, SPLASH_TIME);
    }
}
```

#### b. SplashScreen.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
```

```

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/hotelpapa"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"/>

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:paddingBottom="20dp"
    android:textSize="50dp"
    android:textStyle="bold" />

<!--
<ProgressBar
    android:padding="10dp"
    android:id="@+id/progressBar1"
style="?android:attr/progressBarStyleLarge"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_below="@+id/textView1"
    android:layout_centerHorizontal="true" />
-->
</RelativeLayout>

```

## 2. Menu Utama

### a. HomeActivity.java

```

package com. minangsoft. gethotelpapa;

import android. app. ActionBar;
import android. app. Activity;
import android. app. Fragment;
import android. app. FragmentManager;
import android. os. Bundle;
import android. support. v4. app. FragmentActivity;
import android. support. v4. app. FragmentTransaction;
import android. support. v4. widget. DrawerLayout;
import android. view. LayoutInflater;
import android. view. Menu;
import android. view. View;
import android. view. ViewGroup;

import com. minangsoft. gethotelpapa. fragment. HomeFragment;
import
com. minangsoft. gethotelpapa. fragment. ProductDetailFragmen
t;
import
com. minangsoft. gethotelpapa. fragment. TentangFragment;

public class HomeActivity extends FragmentActivity

```

```

        implements
NavigationDrawerFragment.NavigationDrawerCallbacks {

    private android.support.v4.app.Fragment
contentFragment;
    HomeFragment homeFragment;
    TentangFragment tentangFragment;
    ProductDetailFragment pdtDetailFragment;

    /**
     * Fragment managing the behaviors, interactions and
     presentation of the navigation drawer.
     */
    private NavigationDrawerFragment
mNavigationDrawerFragment;

    /**
     * Used to store the last screen title. For use in
     {@link #restoreActionBar()}.
     */
    private CharSequence mTitle;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        mNavigationDrawerFragment =
(NavigationDrawerFragment)

getFragmentManager().findFragmentById(R.id.navigation_dra
wer);
        mTitle = getTitle();

        // Set up the drawer.
mNavigationDrawerFragment.setUp(
            R.id.navigation_drawer,
            (DrawerLayout)
findViewById(R.id.drawer_layout));

        android.support.v4.app.FragmentManager
fragmentManager = getSupportFragmentManager();

        if (savedInstanceState != null) {
            if
(savedInstanceState.containsKey("content")) {
                String content =
savedInstanceState.getString("content");
                if
(content.equals(ProductDetailFragment.ARG_ITEM_ID)) {
                    if (fragmentManager

.findFragmentByTag(ProductDetailFragment.ARG_ITEM_ID) !=
null) {
                        contentFragment = fragmentManager

```

```

        .findFragmentByTag(ProductDetailFragment.ARG_ITEM_ID);
    }
}
if
(fragmentManager.findFragmentByTag(HomeFragment.ARG_ITEM_ID) != null) {
    homeFragment = (HomeFragment)
fragmentManager

.findFragmentByTag(HomeFragment.ARG_ITEM_ID);
    contentFragment = homeFragment;
}
} else {
    homeFragment = new HomeFragment();
    switchContent(homeFragment,
HomeFragment.ARG_ITEM_ID);
}

}

@Override
public void onNavigationDrawerItemSelected(int
position) {
    // update the main content by replacing fragments
    FragmentManager fragmentManager =
getFragmentManager();
    fragmentManager.beginTransaction()
        .replace(R.id.container,
PlaceholderFragment.newInstance(position + 1))
        .commit();
}

public void onSectionAttached(int number) {
    switch (number) {
        case 1:
            mTitle =
getString(R.string.title_section1);
            homeFragment = new HomeFragment();
            switchContent(homeFragment,
HomeFragment.ARG_ITEM_ID);
            break;
        case 2:
            mTitle =
getString(R.string.title_section2);
            tentangFragment = new TentangFragment();
            switchContent(tentangFragment,
TentangFragment.ARG_ITEM_ID);
            break;
        case 3:
            mTitle =
getString(R.string.title_section3);
            finish();
            break;
    }
}
}

```

```

        public void restoreActionBar() {
            ActionBar actionBar = getActionBar();

            actionBar.setNavigationMode(ActionBar.NAVIGATION_MODE_STANDARD);
            actionBar.setDisplayShowTitleEnabled(true);
            actionBar.setTitle(mTitle);
        }

        /**
         * A placeholder fragment containing a simple view.
         */
        public static class PlaceholderFragment extends
            Fragment {
            /**
             * The fragment argument representing the section
             * number for this
             * fragment.
             */
            private static final String ARG_SECTION_NUMBER =
                "section_number";

            public PlaceholderFragment() {
            }

            /**
             * Returns a new instance of this fragment for
             * the given section
             * number.
             */
            public static PlaceholderFragment newInstance(int
                sectionNumber) {
                PlaceholderFragment fragment = new
                PlaceholderFragment();
                Bundle args = new Bundle();
                args.putInt(ARG_SECTION_NUMBER,
                    sectionNumber);
                fragment.setArguments(args);
                return fragment;
            }

            @Override
            public View onCreateView(LayoutInflater inflater,
                ViewGroup container,
                Bundle
                savedInstanceState) {
                View rootView =
                inflater.inflate(R.layout.fragment_home_2, container,
                    false);
                return rootView;
            }

            @Override
            public void onAttach(Activity activity) {
                super.onAttach(activity);
            }
        }
    }

```

```

        ((HomeActivity) activity).onSectionAttached(
getArguments().getInt(ARG_SECTION_NUMBER));
    }
}

@Override
protected void onSaveInstanceState(Bundle outState) {
    if (contentFragment instanceof HomeFragment) {
        outState.putString("content",
HomeFragment.ARG_ITEM_ID);
    } else {
        outState.putString("content",
ProductDetailFragment.ARG_ITEM_ID);
    }
    super.onSaveInstanceState(outState);
}

public void
switchContent(android.support.v4.app.Fragment fragment,
String tag) {
    android.support.v4.app.FragmentManager
fragmentManager = getSupportFragmentManager();
    while (fragmentManager.popBackStackImmediate())
        ;

    if (fragment != null) {
        FragmentTransaction transaction =
fragmentManager
            .beginTransaction();
        transaction.replace(R.id.content_frame,
fragment, tag);
        // Only ProductDetailFragment is added to the
back stack.
        if (!(fragment instanceof HomeFragment)) {
            transaction.addToBackStack(tag);
        }
        transaction.commit();
        contentFragment = fragment;
    }
}

@Override
public void onBackPressed() {
    android.support.v4.app.FragmentManager fm =
getSupportFragmentManager();
    if (fm.getBackStackEntryCount() > 0) {
        super.onBackPressed();
    } else if (contentFragment instanceof
HomeFragment
        || fm.getBackStackEntryCount() == 0) {
        finish();
    }
}

@Override

```

```

    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }
}

```

## b. Activity\_home.xml

```

<!-- A DrawerLayout is intended to be used as the top-
level content view using match_parent for both width and
height to consume the full space available. -->
<android.support.v4.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

tools:context="com.minangsoft.gethotelpapa.HomeActivity">

    <!-- As the main content view, the view below
consumes the entire
        space available using match_parent in both
dimensions. -->

    <FrameLayout
        android:id="@+id/content_frame"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <FrameLayout
        android:id="@+id/container"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <!-- The drawer is given a fixed width in dp and
extends the full height of
        the container. -->
    <fragment
        android:id="@+id/navigation_drawer"

android:name="com.minangsoft.gethotelpapa.NavigationDrawe
rFragment"

android:layout_width="@dimen/navigation_drawer_width"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        tools:layout="@layout/drawer_home" />

</android.support.v4.widget.DrawerLayout>

```







```

        mDrawerListView = (ListView) inflater.inflate(
            R.layout.drawer_home, container, false);
        mDrawerListView.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?>
parent, View view, int position, long id) {
                selectItem(position);
            }
        });
        mDrawerListView.setAdapter(new
ArrayAdapter<String>(
            getActionBar().getThemedContext(),

android.R.layout.simple_list_item_activated_1,
            android.R.id.text1,
            new String[]{

getString(R.string.title_section1),

getString(R.string.title_section2),

getString(R.string.title_section3),
            }));

mDrawerListView.setItemChecked(mCurrentSelectedPosition,
true);

        return mDrawerListView;
    }

    public boolean isDrawerOpen() {
        return mDrawerLayout != null &&
mDrawerLayout.isDrawerOpen(mFragmentManagerContainerView);
    }

    /**
     * Users of this fragment must call this method to
     set up the navigation drawer interactions.
     *
     * @param fragmentId The android:id of this
     fragment in its activity's layout.
     * @param drawerLayout The DrawerLayout containing
     this fragment's UI.
     */
    public void setUp(int fragmentId, DrawerLayout
drawerLayout) {
        mFragmentManagerContainerView =
getActivity().findViewById(fragmentId);
        mDrawerLayout = drawerLayout;

        // set a custom shadow that overlays the main
        content when the drawer opens

mDrawerLayout.setDrawerShadow(R.drawable.drawer_shadow,
GravityCompat.START);
        // set up the drawer's list view with items and

```

*click listener*

```
        ActionBar actionBar = getActionBar();
        actionBar.setDisplayHomeAsUpEnabled(true);
        actionBar.setHomeButtonEnabled(true);

        // ActionBarDrawerToggle ties together the the
proper interactions
        // between the navigation drawer and the action
bar app icon.
        mDrawerToggle = new ActionBarDrawerToggle(
            getActivity(),                /* host
Activity */
            mDrawerLayout,                /*
DrawerLayout object */
            R.drawable.ic_drawer,         /* nav
drawer image to replace 'Up' caret */
            R.string.navigation_drawer_open, /*
"open drawer" description for accessibility */
            R.string.navigation_drawer_close /*
"close drawer" description for accessibility */
        ) {
            @Override
            public void onDrawerClosed(View drawerView) {
                super.onDrawerClosed(drawerView);
                if (!isAdded()) {
                    return;
                }

                getActivity().invalidateOptionsMenu(); //
calls onPrepareOptionsMenu()
            }

            @Override
            public void onDrawerOpened(View drawerView) {
                super.onDrawerOpened(drawerView);
                if (!isAdded()) {
                    return;
                }

                if (!mUserLearnedDrawer) {
                    // The user manually opened the
drawer; store this flag to prevent auto-showing
                    // the navigation drawer
automatically in the future.
                    mUserLearnedDrawer = true;
                    SharedPreferences sp =
PreferenceManager
.getDefaultSharedPreferences(getActivity());

                    sp.edit().putBoolean(PREF_USER_LEARNED_DRAWER,
true).apply();
                }

                getActivity().invalidateOptionsMenu(); //
```

```

calls onPrepareOptionsMenu()
    }
};

// If the user hasn't 'learned' about the drawer,
open it to introduce them to the drawer,
// per the navigation drawer design guidelines.
if (!mUserLearnedDrawer &&
!mFromSavedInstanceState) {

mDrawerLayout.openDrawer(mFragmentContainerView);
}

// Defer code dependent on restoration of
previous instance state.
mDrawerLayout.post(new Runnable() {
    @Override
    public void run() {
        mDrawerToggle.syncState();
    }
});

mDrawerLayout.setDrawerListener(mDrawerToggle);
}

private void selectItem(int position) {
    mCurrentSelectedPosition = position;
    if (mDrawerListView != null) {
        mDrawerListView.setItemChecked(position,
true);
    }
    if (mDrawerLayout != null) {
mDrawerLayout.closeDrawer(mFragmentContainerView);
    }
    if (mCallbacks != null) {
mCallbacks.onNavigationDrawerItemSelected(position);
    }
}

@Override
public void onAttach(Activity activity) {
    super.onAttach(activity);
    try {
        mCallbacks = (NavigationDrawerCallbacks)
activity;
    } catch (ClassCastException e) {
        throw new ClassCastException("Activity must
implement NavigationDrawerCallbacks.");
    }
}

@Override
public void onDetach() {
    super.onDetach();
}

```

```

        mCallbacks = null;
    }

    @Override
    public void onSaveInstanceState(Bundle outState) {
        super.onSaveInstanceState(outState);
        outState.putInt(STATE_SELECTED_POSITION,
mCurrentSelectedPosition);
    }

    @Override
    public void onConfigurationChanged(Configuration
newConfig) {
        super.onConfigurationChanged(newConfig);
        // Forward the new configuration the drawer
toggle component.
        mDrawerToggle.onConfigurationChanged(newConfig);
    }

    @Override
    public void onCreateOptionsMenu(Menu menu,
MenuInflater inflater) {
        // If the drawer is open, show the global app
actions in the action bar. See also
        // showGlobalContextActionBar, which controls the
top-left area of the action bar.
        if (mDrawerLayout != null && isDrawerOpen()) {
            inflater.inflate(R.menu.global, menu);
            showGlobalContextActionBar();
        }
        super.onCreateOptionsMenu(menu, inflater);
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (mDrawerToggle.onOptionsItemSelected(item)) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    /**
     * Per the navigation drawer design guidelines,
updates the action bar to show the global app
     * 'context', rather than just what's in the current
screen.
     */
    private void showGlobalContextActionBar() {
        ActionBar actionBar = getActionBar();
        actionBar.setDisplayShowTitleEnabled(true);

actionBar.setNavigationMode(ActionBar.NAVIGATION_MODE_STA
NDARD);
        actionBar.setTitle(R.string.app_name);
    }

```



```

import
com.minangsoft.gethotelpapa.json.CustomListAdapterHotel;
import com.minangsoft.gethotelpapa.json.Hotel;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import static java.lang.StrictMath.toRadians;

public class HotelActivity extends Activity implements
ListView.OnItemClickListener {

    private ListView listView;
    private String JSON_STRING,ttile;
    GPSTracker gps;
    private ProgressDialog pdDialog1;
    double latitude = 0;
    double longitude = 0;
    private List<Hotel> hotelList = new
ArrayList<Hotel>();
    private CustomListAdapterHotel adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hotel);

        gps = new GPSTracker(getBaseContext());

        listView = (ListView)
findViewById(R.id.listView);
        listView.setOnItemClickListener(this);

        adapter = new CustomListAdapterHotel(this,
hotelList);
        listView.setAdapter(adapter);

        load_data();
    }

    private void load_data(){
        showDialog(true,"Loading...");
        StringRequest req = new
StringRequest(Request.Method.POST, AppConfig.URL_API, new
Response.Listener<String>() {
            @Override
            public void onResponse(String s) {
                //Toast.makeText(getApplicationContext(),
s,Toast.LENGTH_SHORT).show();

```

```

        try {
            List<HashMap<String, String>> list =
new ArrayList<>();
            JSONObject data= new JSONObject(s);
            JSONArray data2 =
data.optJSONArray("data");

//Toast.makeText(getApplicationContext(),
data.getString("data"),Toast.LENGTH_SHORT).show();
            hotellist.clear();
            for (int i=0; i<data2.length(); i++){
                JSONObject tmp_data =
data2.getJSONObject(i);

                if(gps.canGetLocation()) {

                    latitude = gps.getLatitude();
                    longitude =
gps.getLongitude();

                    String id =
tmp_data.getString("id_hotel");
                    String name =
tmp_data.getString("nama");
                    String alamat =
tmp_data.getString("alamat");
                    String latlong =
tmp_data.getString("lat_lon");
                    String ser =
tmp_data.getString("service");
                    String no_telp =
tmp_data.getString("no_telp");
                    String gambar =
tmp_data.getString("gambar");
                    String[] lalo =
latlong.split(",");

                    String lat = lalo[0];
                    String longa = lalo[1];

                    //String diskripsi =
diska.substring(0,40);
                    //String img =
tmp_data.getString("foto");

                    double radiang = (6371 *
Math.acos(Math.cos(toRadians(latitude)) *
Math.cos(toRadians(Double.parseDouble(lat))) *
Math.cos(toRadians(Double.parseDouble(longa)) -
toRadians(longitude)) + Math.sin(toRadians(latitude)) *
Math.sin(toRadians(Double.parseDouble(lat)))));

                    //if(radiang <= 10000 ){

                    Hotel hotel = new Hotel();
                    hotel.setid(id);

```



```

        hotel.setTitle(name);
        hotel.setAlamat(alamat);

hotel.setThumbnailUrl(AppConfig.INDUK_SERVER + "hotel/" +
gambar);
        hotel.setJarak("Jarak : " +
papo(radiang) + " KM");
        hotelList.add(hotel);

//
    }

    }
    adapter.notifyDataSetChanged();

} catch (JSONException e) {

Toast.makeText(getApplicationContext(), "Kesalahan dalam
koneksi\nPastikan perangkat telahterkoneksi dengan
benar!", Toast.LENGTH_SHORT).show();
        Log.e("RequestLogin",s);
    }
    showDialog(false, "");
}
}, new Response.ErrorListener() {
    @Override
    public void onResponse(VolleyError
error) {
        Toast.makeText(getApplicationContext(),
"Kesalahan dalam koneksi\nPastikan perangkat
telahterkoneksi dengan benar!",
Toast.LENGTH_SHORT).show();
        showDialog(false, "");
    }
}){
    @Override
    protected Map<String, String> getParams()
throws AuthFailureError {
        Map<String, String> param = new
HashMap<>();
        param.put("action", "hotel");
        param.put("latlong",
String.valueOf(latitude+longitude));
        return param;
    }
};
//Volley.newRequestQueue(this).add(req);
AppData.getInstance().addToRequestQueue(req);
}

public String papo(double value) {
    DecimalFormat myFormatter = new
DecimalFormat("0.0");
    return myFormatter.format(value);
}

```

```

private void showDialog(boolean show,String message){
    if(pdDialogl == null){
        pdDialogl = new ProgressDialog(this);
    }
    pdDialogl.setMessage(message);
    pdDialogl.setCancelable(false);
    if(show) pdDialogl.show();
    else pdDialogl.dismiss();
}

@Override
public void onItemClick(AdapterView<?> parent, View
view, int position, long id) {

    TextView c = (TextView)
view.findViewById(R.id.id);
    String id_send = c.getText().toString();
    Intent intent = new Intent(getBaseContext(),
Detil.class);
    intent.putExtra("id_send", id_send);
    startActivity(intent);

//      HashMap<String,String> data_map
=(HashMap)parent.getItemAtPosition(position);
//      String latlon =
data_map.get("latlon").toString();
//      Intent i = new Intent(Intent.ACTION_VIEW);
//
i.setData(Uri.parse("https://www.google.com/maps/search/?
api=1&query="+latlon));
//      startActivity(i);
}
}
}

```

## b. Hotel.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context="com.minangsoft.gethotelpapa.HotelActivity"
>

<ListView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/listView"
    tools:layout_editor_absoluteY="0dp"
    tools:layout_editor_absoluteX="8dp"
    android:divider="@null"

```

```
        android:dividerHeight="0dp"
        tools:ignore="MissingConstraints" />
```

```
</RelativeLayout>
```

## 5. Peta

### a. Peta.java

```
package com.minangsoft.gethotelpapa;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.ProgressDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Build;
import android.os.Bundle;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.FragmentActivity;
import android.support.v4.content.ContextCompat;
import android.util.Log;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.minangsoft.gethotelpapa.app.AppConfig;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
```

```

import static java.lang.StrictMath.toRadians;

public class PetaActivity extends FragmentActivity
implements OnMapReadyCallback,
           GoogleApiClient.ConnectionCallbacks,
           GoogleApiClient.OnConnectionFailedListener,
           LocationListener {

    private GoogleMap mGoogleMap;
    SupportMapFragment mapFrag;
    LocationRequest mLocationRequest;
    GoogleApiClient mGoogleApiClient;
    Location mLastLocation;
    Marker mCurrLocationMarker;

    MarkerOptions markerOptions = new MarkerOptions();
    CameraPosition cameraPosition;
    LatLng center;
    LatLng latLng;
    String title, id_map, JSON_STRING;

    public static final String ID = "id";
    public static final String TITLE = "nama";
    public static final String LAT = "lat";
    public static final String LNG = "long";

    public static double mlastLat, mlastLong, var;

    private String url = "";

    String tag_json_obj = "json_obj_req";

    GPSTracker gps;

    double latitude = 0;
    double longitude = 0;
    private ProgressDialog pdDialog1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_peta);

        gps = new GPSTracker(getApplicationContext());
        // Obtain the SupportMapFragment and get notified
when the map is ready to be used.
        SupportMapFragment mapFragment =
(SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
    }

    @Override

```

```

    public void onResume() {
        super.onResume();

        if (mGoogleMap != null) { //prevent crashing if
the map doesn't exist yet (eg. on starting activity)
            //mGoogleMap.clear();

            buildGoogleApiClient();
            if
(ActivityCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
                // TODO: Consider calling
                //     ActivityCompat#requestPermissions
                // here to request the missing
permissions, and then overriding
                //     public void
onRequestPermissionsResult(int requestCode, String[]
permissions,
                    //
int[] grantResults)
                // to handle the case where the user
grants the permission. See the documentation
                // for ActivityCompat#requestPermissions
for more details.
                return;
            }
            mGoogleMap.setMyLocationEnabled(true);
            LatLng latLng = new LatLng(mLastLat,
mLastLong);
            MarkerOptions markerOptions = new
MarkerOptions();
            markerOptions.position(latLng);
            //markerOptions.title("Posisi Saya");

markerOptions.icon(BitmapDescriptorFactory.fromResource(R
.drawable.marker_position));
            mCurrLocationMarker =
mGoogleMap.addMarker(markerOptions);

            mGoogleMap.setMyLocationEnabled(true);
            center = new LatLng(mLastLat, mLastLong);
            cameraPosition = new
CameraPosition.Builder().target(center).zoom(15).build();

mGoogleMap.animateCamera(CameraUpdateFactory.newCameraPos
ition(cameraPosition));
            load_data();

            // add markers from database to the map
        }
    }

```

```

@Override
public void onPause() {
    super.onPause();

    //stop location updates when Activity is no
longer active
    if (mGoogleApiClient != null) {

LocationServices.FusedLocationApi.removeLocationUpdates(m
GoogleApiClient, this );
    }
}

@Override
public void onMapReady(GoogleMap googleMap)
{
    mGoogleMap=googleMap;

mGoogleMap.setMapType(GoogleMap.MAP_TYPE_TERRAIN);

    //Initialize Google Play Services
    if (android.os.Build.VERSION.SDK_INT >=
Build.VERSION_CODES.M) {
        if
(ContextCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_FINE_LOCATION)
            == PackageManager.PERMISSION_GRANTED)
        {
            //Location Permission already granted
            buildGoogleApiClient();
            mGoogleMap.setMyLocationEnabled(true);

            latitude = gps.getLatitude();
            longitude = gps.getLongitude();

            center = new LatLng(latitude, longitude);
            cameraPosition = new
CameraPosition.Builder().target(center).zoom(15).build();

            googleMap.animateCamera(CameraUpdateFactory.newCameraPosi
tion(cameraPosition));
                //getMarkers();
                load_data();
            } else {
                //Request Location Permission
                checkLocationPermission();
            }
        }
    } else {
        buildGoogleApiClient();
        mGoogleMap.setMyLocationEnabled(true);
    }
}
}

```

```

        protected synchronized void buildGoogleApiClient() {
            mGoogleApiClient = new
GoogleApiClient.Builder(getBaseContext())
                .addConnectionCallbacks(this)
                .addOnConnectionFailedListener(this)
                .addApi(LocationServices.API)
                .build();
            mGoogleApiClient.connect();
        }

        @Override
        public void onConnected(Bundle bundle) {
            mLocationRequest = new LocationRequest();
            //mLocationRequest.setInterval(1000);
            //mLocationRequest.setFastestInterval(1000);

            //mLocationRequest.setPriority(LocationRequest.PRIORITY_B
            ALANCED_POWER_ACCURACY);
            if
(ContextCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_FINE_LOCATION)
    == PackageManager.PERMISSION_GRANTED) {

                //LocationServices.FusedLocationApi.requestLocationUpdate
                s(mGoogleApiClient, mLocationRequest, this);
            }
        }

        @Override
        public void onConnectionSuspended(int i) {}

        @Override
        public void onConnectionFailed(ConnectionResult
connectionResult) {}

        @Override
        public void onLocationChanged(Location location)
        {
            mLastLocation = location;
            if (mCurrLocationMarker != null) {
                mCurrLocationMarker.remove();
            } else {
                //Place current location marker
                mLastLat = location.getLatitude();
                mLastLong = location.getLongitude();
                LatLng latLng = new
                LatLng(location.getLatitude(), location.getLongitude());
                MarkerOptions markerOptions = new
                MarkerOptions();
                markerOptions.position(latLng);
                //markerOptions.title("Posisi Saya");

                markerOptions.icon(BitmapDescriptorFactory.fromResource(R
                .drawable.marker_position));
                mCurrLocationMarker =

```

```

mGoogleMap.addMarker(markerOptions);

//          Toast.makeText(getContext(), "Hasil : "+
mLastLat,Toast.LENGTH_LONG).show();
//move map camera

//mGoogleMap.moveCamera(CameraUpdateFactory.newLatLngZoom
(latLng,15));
    }

}

    public static final int
MY_PERMISSIONS_REQUEST_LOCATION = 99;
    private void checkLocationPermission() {
        if
(ContextCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {

            // Should we show an explanation?
            if
(ActivityCompat.shouldShowRequestPermissionRationale(this
,
android.Manifest.permission.ACCESS_FINE_LOCATION)) {

                // Show an explanation to the user
                *asynchronously* -- don't block
                // this thread waiting for the user's
                response! After the user
                // sees the explanation, try again to
                request the permission.
                new AlertDialog.Builder(getBaseContext())
                    .setTitle("Location Permission
                Needed")
                    .setMessage("This app needs the
                Location permission, please accept to use location
                functionality")
                    .setPositiveButton("OK", new
                DialogInterface.OnClickListener() {
                    @Override
                    public void
                onClick(DialogInterface dialogInterface, int i) {
                        //Prompt the user once
                        explanation has been shown

                ActivityCompat.requestPermissions((Activity)
                getBaseContext(),
                    new
                String[] {android.Manifest.permission.ACCESS_FINE_LOCATION
                },
                MY_PERMISSIONS_REQUEST_LOCATION );

```



```

        }
    })
    .create()
    .show();

    } else {
        // No explanation needed, we can request
the permission.
        ActivityCompat.requestPermissions(this,
            new
String[]{android.Manifest.permission.ACCESS_FINE_LOCATION
},
            MY_PERMISSIONS_REQUEST_LOCATION
);
    }
}

@Override
public void onRequestPermissionsResult(int
requestCode,
String
permissions[], int[] grantResults) {
    switch (requestCode) {
        case MY_PERMISSIONS_REQUEST_LOCATION: {
            // If request is cancelled, the result
arrays are empty.
            if (grantResults.length > 0
                && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {

                // permission was granted, yay! Do
the
                // location-related task you need to
do.
                if
(ContextCompat.checkSelfPermission(getBaseContext(),
android.Manifest.permission.ACCESS_FINE_LOCATION)
                ==
PackageManager.PERMISSION_GRANTED) {

                    if (mGoogleApiClient == null) {
                        buildGoogleApiClient();
                    }

mGoogleMap.setMyLocationEnabled(true);
                }

            } else {

                // permission denied, boo! Disable
the
                // functionality that depends on this
permission.

```

```

        Toast.makeText(getBaseContext(),
"permission denied", Toast.LENGTH_LONG).show();
    }
    return;
}

// other 'case' lines to check for other
// permissions this app might request
}
}

private void addMarker(final LatLng latlng, final
String title) {
    markerOptions.position(latlng);
    markerOptions.title(title);

markerOptions.icon(BitmapDescriptorFactory.fromResource(R
.drawable.marker_shelter_new));
    mGoogleMap.addMarker(markerOptions);

    mGoogleMap.setOnInfoWindowClickListener(new
GoogleMap.OnInfoWindowClickListener() {
        @Override
        public void onInfoWindowClick(Marker marker)
        {
            // Intent i = new
            Intent(Intent.ACTION_VIEW);
            // String newString =
            String.valueOf(latlng);
            // String lala = newString.substring(10);
            // String lila = lala.replace(" ", "");
            //
            //Toast.makeText(getBaseContext(), lila, Toast.LENGTH_LONG)
            .show();
            //
            i.setData(Uri.parse("https://www.google.com/maps/search/?
            api=1&query="+lila));
            // startActivity(i);
            Intent intent = new
            Intent(getBaseContext(), Detil.class);
            Bundle extras = new Bundle();
            extras.putString("nama_send",
            marker.getTitle());
            intent.putExtras(extras);
            startActivity(intent);
        }
    });
}

private void load_data(){
    showDialog(true, "Loading...");
    StringRequest req = new
StringRequest(Request.Method.POST, AppConfig.URL_API, new
Response.Listener<String>() {
        @Override

```

```

        public void onResponse(String s) {
            //Toast.makeText(getApplicationContext(),
            s,Toast.LENGTH_SHORT).show();
            try {
                List<HashMap<String, String>> list =
new ArrayList<>();
                JSONObject data= new JSONObject(s);
                JSONArray data2 =
data.optJSONArray("data");

                //Toast.makeText(getApplicationContext(),
                data.getString("data"),Toast.LENGTH_SHORT).show();

                for (int i=0; i<data2.length(); i++){
                    JSONObject tmp_data =
data2.getJSONObject(i);

                    if(gps.canGetLocation()){

                    }

                    latitude = gps.getLatitude();
                    longitude = gps.getLongitude();

                    String latlong =
tmp_data.getString("lat_lon");
                    String[] lalo =
latlong.split(",");
                    String lat = lalo[0];
                    String longa = lalo[1];

                    //String diskripsi =
diska.substring(0,40);
                    //String img =
tmp_data.getString("foto");

                    double radiang = ( 6371 *
Math.acos( Math.cos( toRadians(latitude) ) * Math.cos(
toRadians(Double.parseDouble(lat)) ) * Math.cos(
toRadians(Double.parseDouble(longa)) -
toRadians(longitude) ) + Math.sin( toRadians(latitude) )
* Math.sin( toRadians( Double.parseDouble(lat) ) ) ) );

                    //if(radiang <= 10000 ){
                    title =
tmp_data.getString("nama");
                    latLng = new
LatLng(Double.parseDouble(lat),
Double.parseDouble(longa));

                    addMarker(latLng, title);

                    //}

```

```

    }

    } catch (JSONException e) {

Toast.makeText(getApplicationContext(), "Kesalahan dalam
koneksi\nPastikan perangkat telah terkoneksi dengan
benar!", Toast.LENGTH_SHORT).show();
        Log.e("RequestLogin",s);
    }
    showDialog(false, "");
    }
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError
error) {
        Toast.makeText(getApplicationContext(),
"Kesalahan dalam koneksi\nPastikan perangkat telah
terkoneksi dengan benar!", Toast.LENGTH_SHORT).show();
        showDialog(false, "");
    }
}){
    @Override
    protected Map<String, String> getParams()
throws AuthFailureError {
        Map<String, String> param = new
HashMap<>();
        param.put("action", "hotel");
        param.put("latlong",
String.valueOf(latitude+longitude));
        return param;
    }
};
Volley.newRequestQueue(this).add(req);
}

private void showDialog(boolean show,String message){
    if(pdDialogl == null){
        pdDialogl = new ProgressDialog(this);
    }
    pdDialogl.setMessage(message);
    pdDialogl.setCancelable(false);
    if(show) pdDialogl.show();
    else pdDialogl.dismiss();
}

@Override
public void onBackPressed() {
    super.onBackPressed();
    Intent intent = new Intent(getBaseContext(),
HomeActivity.class);
    startActivity(intent);
    finish();
}
}
}

```

## b. Peta.xml

```
<fragment
xmlns:android="http://schemas.android.com/apk/res/android"
"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/map"

    android:name="com.google.android.gms.maps.SupportMapFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="com.minangsoft.gethotelpapa.PetaActivity"
/>
```

