

SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI PEKALONGAN BERBASIS ANDROID

LAPORAN TUGAS AKHIR

Disusun untuk memenuhi Syarat Memperoleh Gelar
Ahli Madya pada Program Diploma III (D3)
Program Studi Manajemen Informatika



Disusun Oleh :

Saifudin

MI.15.00305

**PROGRAM STUDI MANAJEMEN INFORMATIKA
POLITEKNIK MUHAMMADIYAH PEKALONGAN**

2019

LEMBAR PERSETUJUAN
SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI
PEKALONGAN BERBASIS ANDROID

TUGAS AKHIR

Disusun Oleh :

Saifudin (MI.15.00305)

Diajukan untuk Memenuhi Persyaratan Akademis dalam Menyelesaikan
Program Diploma Tiga pada Program Studi Manajemen Informatika
Politeknik Muhammadiyah Pekalongan

Pekalongan, 21 Januari 2019

Pembimbing 1

Pembimbing 2

Aslam Fatkhudin, S.Kom., M.Kom.
NPPM : 100.0306

Edy Subowo, ST.
NPPM : 100.6314

Mengetahui,
Ketua Program Studi

Imam Rosyadi, S.E., M.Si.
NPPM : 100.0406

LEMBAR PENGESAHAN
SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI
PEKALONGAN BERBASIS ANDROID

TUGAS AKHIR

Disusun Oleh :

Saifudin (MI.15.00305)

Diajukan untuk Memenuhi Persyaratan Akademis dalam Menyelesaikan
Program Diploma Tiga pada Program Studi Manajemen Informatika
Politeknik Muhammadiyah Pekalongan

Pekalongan, 29 Januari 2019

Penguji 1

Penguji 2

Teguh Satrio, S.Kom, M.Eng
NPPM : 100.7017

Imam Rosyadi, S.E., M.Si.
NPPM : 100.0406

Mengetahui,
Ketua Program Studi

Imam Rosyadi, S.E., M.Si.
NPPM : 100.0406

MOTTO

“Wa Man Jaahaada fa-innama yujaahidu linafsihi.”

“Barang siapa bersungguh-sungguh, sesungguhnya kesungguhannya itu adalah untuk dirinya sendiri.” (QS. Al-Ankabut: 6)

“Kita melihat kebahagiaan itu seperti pelangi, tidak pernah berada di atas kepala kita sendiri, tetapi selalu berada di atas kepala orang lain.” (Thomas Hardy)

“Kebanyakan dari kita tidak mensyukuri apa yang sudah kita miliki, tetapi kita selalu menyesali apa yang belum kita capai.” (Schopenhauer)

“Banyak kegagalan dalam hidup ini dikarenakan orang-orang tidak menyadari betapa dekatnya mereka dengan keberhasilan saat mereka menyerah.” (Thomas Alva Edison)

Pendidikan mempunyai akar yang pahit, tapi buahnya manis”.
(Aristoteles)

”Melalui kesabaran, seseorang dapat meraih lebih dari pada melalui kekuatan yang dimilikinya”. (Edmund Burke)

PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah, saya persembahkan Tugas Akhir ini untuk :

1. Bapak Imam Rosyadi, SE, M.Si selaku Kepala Jurusan Manajemen Informatika Politeknik Muhammadiyah Pekalongan.
2. Bapak Aslam Fatkhudin dan Edi Subowo selaku dosen pembimbing yang selama pengerjaan tugas akhir ini telah banyak membantu saya.
3. Bapak dan Ibu tercinta yang memberikan semangat dalam segala hal.
4. Teman-teman Manajemen Informatika angkatan 2015 yang memberikan masukan dalam penyusunan laporan Tugas Akhir ini.
5. Semua pihak yang telah membantu penyusunan laporan Tugas Akhir ini.

KATA PENGANTAR

Puji syukur Alhamdulillah, penulis panjatkan atas segala limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan judul **“Sistem Informasi Geografis Wisata Gunung Di Pekalongan Berbasis Android”**.

Pembuatan Tugas Akhir ini disusun sebagai salah satu syarat untuk menyelesaikan pendidikan Program Diploma III, jurusan Manajemen Informatika Politeknik Muhammadiyah Pekalongan.

Penulis menyadari bahwa tersusunnya Tugas Akhir ini tidak terlepas dari bimbingan, dorongan, dan bantuan semua pihak. Untuk itu penulis mengucapkan terima kasih kepada :

1. Aslam Fatkhudin, S.Kom., M.Kom. selaku Direktur Politeknik Muhammadiyah Pekalongan sekaligus dosen pembimbing 1, yang selama ini telah memimpin Politeknik Muhammadiyah Pekalongan dengan baik dan membawa kampus ini ke arah yang lebih baik.
2. Imam Rosyadi, S.E., M.Si. selaku kepala jurusan Manajemen Informatika, yang telah memberikan arahan-arahan yang baik kepada semua mahasiswa jurusan Manajemen Informatika sehingga menciptakan mahasiswanya menjadi sumber daya manusia yang lebih berkualitas.
3. Edy Subowo, S.T selaku Dosen Pembimbing 2 yang telah membimbing dan sabar dalam memberikan saran-saran yang baik demi terwujudnya Tugas Akhir ini dengan baik dan benar.

4. Bapak dan Ibu dosen Politeknik Muhammadiyah Pekalongan yang telah memberikan bekal ilmu pengetahuan yang sangat berguna dalam penyusunan Tugas Akhir ini.
5. Bapak dan Ibu staf karyawan Politeknik Muhammadiyah Pekalongan yang telah memberi dukungan dan semangat demi terselesaikannya penyusunan Tugas Akhir ini.
6. Keluarga besarku yang telah memberikan dorongan moril dan materiil serta do'a sehingga dapat terselesaikannya penyusunan Tugas Akhir ini.
7. Teman-teman dan semua pihak yang secara langsung maupun tidak langsung yang telah membantu dalam penyusunan Tugas Akhir ini.

Penulis menyadari bahwa penyusunan Tugas Akhir ini masih banyak kekurangan, sehingga penulis mengharapkan kritik dan saran yang bersifat membangun demi kesempurnaan Tugas Akhir ini. Besar harapan penulis semoga penyusunan Tugas Akhir ini dapat bermanfaat bagi semua pihak yang berkepentingan.

Pekalongan, 21 Mei 2018

Penulis

ABSTRAKSI

Selain terkenal dengan batiknya, Pekalongan juga terkenal dengan destinasi wisata, keindahan seni dan budayanya. Keindahan alam yang ada di kota yang terletak di kawasan jalur pantai utara jawa ini, juga memiliki segudang destinasi wisata alam yang ditawarkan, seperti curug, pantai dan pegunungan. Pariwisata lokal dapat membuat perekonomian warga meningkat dan juga bisa menjadi media promosi untuk kekayaan lokal. Tetapi dengan banyaknya destinasi wisata tersebut, masih banyak yang belum terekspose dan minim informasi seperti obyek wisata mengenai gunung-gunung di Pekalongan. Padahal di era digital ini seharusnya informasi bisa terbuka lebar kemedi masa baik itu media cetak maupun media elektronik. Selain itu perkembangan teknologi terutama pada ponsel, sekarang ini sudah menjadi *smartphone* dengan menggunakan teknologi sistem operasi Android yang bersifat *open source* sehingga memungkinkan masyarakat untuk ikut mengembangkan sistem informasi tersebut maupun membuat aplikasi yang ada di dalamnya. Berdasarkan permasalahan di atas, maka dibuatlah aplikasi Sistem Informasi Geografis Wisata Gunung Di Pekalongan berbasis *android*. Hasil dari penelitian ini adalah aplikasi Sistem Informasi Geografis Wisata Gunung Di Pekalongan yang dapat memberikan informasi kepada masyarakat tentang wisata gunung, manfaat pendakian, prediksi cuaca, *safery and rescue*, mudah, cepat dan efektif sehingga dapat membantu wisatawan untuk mengenal wisata gunung di Pekalongan.

Kata Kunci : Sistem Informasi Geografis, Wisata Gunung, Aplikasi *Android* dan *Android Studio*

ABSTRACT

Besides being famous for its batik, Pekalongan is also famous for its tourist destinations, the beauty of its art and culture. The natural beauty in the city which is located in the area of the north Java coastline, also has a myriad of natural tourist destinations on offer, such as waterfalls, beaches and mountains. Local tourism can make the people's economy increase and can also be a media campaign for local wealth. But with many tourist destinations, there are still many who have not been exposed and lack information such as tourism objects in the mountains in Pekalongan. Even though in this digital era, information should be widely available both in print and electronic media. In addition, technological developments, especially on cellphones, are now becoming smartphones by using Android operating system technology that is open source so as to enable people to participate in developing the information system as well as making applications in it. Based on the problems above, the application of the Geographic Tourism Information System in Pekalongan is based on Android. The results of this study are the application of Geographic Information System for Mountain Tourism in Pekalongan that can provide information to the community about mountain tourism, the benefits of climbing, weather prediction, safety and rescue, easy, fast and effective so that it can help tourists to know mountain tourism in Pekalongan.

Keywords : *Geographic Information System, Mountain Tourism, Android and Android Studio Applications*

DAFTAR ISI

LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN.....	iii
MOTTO.....	iv
PERSEMBAHAN.....	v
KATA PENGANTAR	vi
ABSTRAKSI.....	viii
ABSTRACT	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah.....	1
1.2. Identifikasi Masalah.....	2
1.3. Rumusan Masalah.....	3
1.4. Batasan Masalah	3
1.5. Tujuan Dan Manfaat Penelitian.....	3
1.5.1. Tujuan	3
1.5.2. Manfaat	3
1.6. Metode Pengumpulan Data	4
1.7. Metode Pengembangan Sistem.....	6
1.8. Sistematika Penulisan	10

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI	11
2.1. Tinjauan Pustaka.....	11
2.2. Landasan Teori	12
2.2.1. Sistem.....	12
2.2.2. Informasi	13
2.2.3. Sistem Informasi.....	14
2.2.4. Sistem Informasi Geografis.....	14
2.2.5. Android	17
2.2.6. Google Map.....	19
2.2.7. Gunung.....	19
2.2.8. Basis Data.....	20
2.2.9. Desain Sistem.....	23
2.2.10. <i>Software</i> Pembangun Aplikasi.....	31
BAB III ANALISIS DAN PERANCANGAN.....	37
3.1. Gambaran Umum DINPORAPAR	37
3.1.1. Sejarah Singkat DINPORAPAR	37
3.1.2. Visi dan Misi DINPORAPAR.....	38
3.1.3. Struktur Organisasi	38
3.2. Analisis Sistem	42
3.2.1. Sistem yang Sedang Berjalan	42
3.2.2. Analisa Sistem yang Sedang Berjalan.....	43
3.3. Perancangan Sistem Baru.....	44
3.4. Perancangan Aliran Data.....	45

3.4.1.	Diagram Konteks	46
3.4.2.	DFD <i>Level 0</i>	47
3.4.3.	DFD <i>Level 1</i> Proses 1 (Pengolahan Data).....	48
3.4.4.	DFD <i>Level 1</i> Proses 3 (Melihat Data).....	49
3.4.5.	Perancangan <i>Entity Relationship Diagram</i> (ERD).....	50
3.5.	Desain <i>Database</i>	50
BAB IV IMPLEMENTASI DAN PEMBAHASAN		55
4.1.	Perancangan Desain Sistem	55
4.1.1.	Tampilan Halaman <i>User</i>	55
4.1.2.	Tampilan Halaman Admin	59
4.2.	Hasil Tampilan Sistem	64
4.2.1.	Tampilan Halaman <i>User</i>	64
4.2.2.	Tampilan Halaman <i>Admin</i>	67
BAB V PENUTUP		73
5.1.	Kesimpulan.....	73
5.2.	Saran	73
DAFTAR PUSTAKA		75
LAMPIRAN		77

DAFTAR GAMBAR

Gambar 1.1 Metode <i>Waterfall</i>	7
Gambar 2.1 <i>File</i> proyek di tampilan Android	32
Gambar 2.2 Jendela utama Android Studio	33
Gambar 3.1 Struktur Organisasi	39
Gambar 3.2 <i>Flowchart</i> Sistem informasi wisata gunung yang sedang berjalan ..	42
Gambar 3.3 <i>Flowchart</i> Sistem informasi geografis yang di usulkan	44
Gambar 3.4 Diagram Konteks	46
Gambar 3.5 DFD <i>Level 0</i>	47
Gambar 3.6 DFD <i>Level 1</i> Proses 1	48
Gambar 3.7 DFD <i>Level 1</i> Proses 3	49
Gambar 3.8 <i>Entity Relationship Diagram</i> (ERD).....	50
Gambar 4.1 Halaman Menu <i>User</i>	55
Gambar 4.2 Halaman Wisata Gunung	56
Gambar 4.3 Detail Wisata	56
Gambar 4.4 Halaman Menuju Lokasi	57
Gambar 4.5 Halaman Galeri.....	57
Gambar 4.6 Halaman Artikel	58
Gambar 4.7 Halaman Detail Artikel	58
Gambar 4.8 Halaman Login	59
Gambar 4.9 Halaman Menu <i>admin</i>	59
Gambar 4.10 Halaman Data Wisata	60
Gambar 4.11 Halaman Tambah Wisata	60

Gambar 4.12 Halaman Artikel (Manfaat Pendakian dan <i>Safety and Resue</i>)	61
Gambar 4.13 Halaman Tambah Artikel	61
Gambar 4.14 Halaman Pos Pendakian	62
Gambar 4.15 Tambah Pos Pendakian	62
Gambar 4.16 Halaman Galeri	63
Gambar 4.17 Tambah Galeri	63
Gambar 4.18 Halaman Menu <i>User</i>	64
Gambar 4.19 Halaman Wisata Gunung	64
Gambar 4.20 Halaman Detail Wisata	65
Gambar 4.21 Halaman Detail Wisata	65
Gambar 4.22 Halaman Galeri	66
Gambar 4.23 Halaman Artikel	66
Gambar 4.24 Halaman Detail Artikel	67
Gambar 4.25 Halaman Login	67
Gambar 4.26 Halaman Menu <i>Admin</i>	68
Gambar 4.27 Halaman Wisata	68
Gambar 4.28 Halaman Tambah Wisata	69
Gambar 4.29 Halaman Artikel	69
Gambar 4.30 Halaman Tambah Artikel	70
Gambar 4.31 Halaman Pos Pendakian	70
Gambar 4.32 Tambah Pos Pendakian	71
Gambar 4.33 Halaman Galeri	71
Gambar 4.34 Halaman Tambah Galeri	72

DAFTAR TABEL

Tabel 2.1 Perkembangan Sistem Operasi Android	18
Tabel 2.2 Simbol <i>Flowchart</i> Sistem	24
Tabel 2.3 Simbol <i>Flowchart</i> Program	27
Tabel 2.4 Komponen-Komponen DFD.....	29
Tabel 2.5 Simbol dalam ERD.....	30
Tabel 3.1 Tabel Admin	51
Tabel 3.2 Tabel Wisata	51
Tabel 3.3 Tabel Manfaat Pendakian	52
Tabel 3.4 Tabel Galeri	53
Tabel 3.5 Tabel <i>Safety and Rescue</i>	53
Tabel 3.6 Tabel Pos Pendakian	54

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Selain terkenal dengan batiknya, Pekalongan terkenal dengan destinasi wisatanya, keindahan seni dan budaya. Keindahan alam yang ada di kota yang terletak di kawasan jalur pantai utara Jawa ini, juga memiliki segudang destinasi wisata alam yang ditawarkan, seperti curug, pantai dan pegunungan (Pekalongan, 2001). Padahal pariwisata lokal dapat membuat perekonomian warga meningkat dan juga bisa menjadi media promosi untuk kekayaan lokal. Tetapi dengan banyaknya destinasi wisata di atas masih banyak yang belum terekspose ke media seperti gunung-gunung di Pekalongan dan informasinya masih minim padahal di era digital ini seharusnya informasi bisa terbuka lebar ke media masa baik itu media cetak, media elektronik dan media siber (Apriadi, 2012).

Perkembangan teknologi pada *smartphone* sekarang sudah berkembang dengan menggunakan teknologi sistem operasi Android yang memungkinkan penggunaan *smartphone* untuk membuat aplikasi berbasis sistem operasi Android yang dapat dijalankan pada *smartphone* yang mendukung Android *platform*. Dengan adanya teknologi Android, pengguna *smartphone* dapat melakukan kreasi sendiri ataupun mengunduh aplikasi Android untuk kemudian digunakan pada telepon selularnya. Penggunaan Android yang semakin luas menjadikan aplikasi android sebagai suatu aplikasi yang mudah diakses oleh semua orang. Pengguna *smartphone* Indonesia juga bertumbuh dengan pesat. Lembaga riset digital

marketing Emarketer memperkirakan pada 2018 jumlah pengguna aktif *smartphone* di Indonesia lebih dari 100 juta orang. Dengan jumlah sebesar itu, Indonesia akan menjadi negara dengan pengguna aktif *smartphone* terbesar keempat di dunia setelah Cina, India, dan Amerika (Kominfo, 2015).

Berdasarkan permasalahan diatas, maka penulis tertarik untuk membuat Sistem Informasi Geografis Wisata Gunung di Pekalongan Berbasis Android yang hanya berfokus pada macam-macam gunung di Pekalongan.

1.2. Identifikasi Masalah

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka dapat ditemukan beberapa permasalahan sebagai berikut :

1. Belum adanya gambaran wilayah gunung di Pekalongan yang dilengkapi rute pendakian dan potensi wisata di DINPORAPAR kabupaten Pekalongan.
2. Belum adanya aplikasi yang memberikan gambaran kepada wisatawan mengenai destinasi wisata gunung dan manfaat wisata gunung pada DINPORAPAR Kabupaten Pekalongan.
3. Belum adanya pendidikan *safety* (keamanan) dan *rescue* (penyelamatan) bagi pendaki gunung di DINPORAPAR kabupaten Pekalongan yang bisa di akses secara digital/*online*.

1.3. Rumusan Masalah

Berdasarkan identifikasi masalah yang telah diuraikan di atas, maka rumusan masalahnya adalah “Bagaimana merancang dan membuat Sistem Informasi Geografis Wisata Gunung di Pekalongan Berbasis Android yang bertujuan untuk mempermudah para wisatawan untuk mencari informasi wisata gunung?”

1.4. Batasan Masalah

Perancangan aplikasi ini dibangun dengan beberapa batasan masalah agar penyusunan tugas akhir ini tidak keluar dari lingkup pembahasan, batasan masalahnya adalah sebagai berikut :

1. Sistem Informasi Geografis ini menampilkan profil gunung di Pekalongan.
2. Informasi tambahan berupa manfaat pendakian, prediksi cuaca, *safety* dan *rescue*.
3. Sistem Informasi Geografis Wisata Gunung di Pekalongan dengan menggunakan *software android studio* dan menggunakan SDK Android 4.2.2 (API 17).

1.5. Tujuan Dan Manfaat Penelitian

1.5.1. Tujuan

Tujuan dari penelitian ini adalah merancang dan membangun Sistem Informasi Geografis Wisata Gunung di Pekalongan Berbasis Android.

1.5.2. Manfaat

Adapun manfaat dalam penelitian ini adalah sebagai berikut :

1. Bagi Penulis

- a. Sebagai sarana untuk mengembangkan ilmu pengetahuan pemrogram penulis yang di dapat di bangku kuliah yang di implementasikan ke dalam lingkungan masyarakat.
- b. Menambah wawasan dan pengalaman dalam pembuatan program aplikasi, khususnya mengenai sistem informasi berbasis android.

2. Bagi Politeknik Muhammadiyah Pekalongan

Sebagai bahan tambahan referensi di perpustakaan Politeknik Muhammadiyah Pekalongan dan dapat dijadikan referensi tugas akhir untuk mahasiswa angkatan selanjutnya.

3. Bagi Wisatawan

Memberikan informasi untuk rekomendasi wisata gunung di Pekalongan khususnya masyarakat Pekalongan maupun dari luar Pekalongan.

4. Bagi Pemerintah Kabupaten Pekalongan

- a. Sebagai media promosi pariwisata.
- b. Mempermudah pihak pengelola dalam mengenalkan objek wisata.

1.6. Metode Pengumpulan Data

Metode penelitian merupakan cara atau strategi menyeluruh untuk menemukan atau memperoleh data yang diperlukan (Soehartono, 2008). Metode

yang akan digunakan untuk merancang dan menganalisa sistem dalam penelitian ini adalah :

1. Metode Observasi

Observasi merupakan teknik atau pendekatan untuk mendapatkan data primer dengan cara mengamati langsung obyek datanya (Jogiyanto, 2008). Metode ini dilakukan dengan melakukan pengamatan secara langsung di Dinas Kepemudaan dan Olahraga dan Pariwisata Kabupaten Pekalongan pada tanggal 23 Juli 2018 sehingga dapat memperoleh data dan informasi yang tepat dan kemudian diambil suatu kesimpulan yaitu berupa data tentang gambaran gunung, ketinggian gunung dan lokasi gunung.

2. Metode Wawancara

Wawancara adalah komunikasi dua arah untuk mendapatkan data dari responden. Wawancara dapat berupa wawancara personal, wawancara dan wawancara telepon (Jogiyanto, 2008). Metode ini dilakukan melalui dialog serta tanya jawab dengan pihak terkait untuk memperoleh data yang dibutuhkan dan menentukan sebuah desain yang akan dibuat. Tentu saja dengan narasumber yang merupakan orang yang berpengalaman dan kompeten dibidangnya, dalam hal ini adalah tanya jawab

- a. kepada Purno Susilo STP, MM. selaku Kasi Pengembangan Pariwisata pada tanggal 23 Juli 2018 untuk mendapatkan data-data yang diperlukan yaitu ketinggian gunung, lokasi gunung, dan profil gunung di Pekalongan.

- b. Pak Supari selaku penjaga gunung rogojembangan pada tanggal 26 agustus 2018 untuk mendapatkan data-data yang diperlukan yaitu tentang profil gunung rogojembangan.

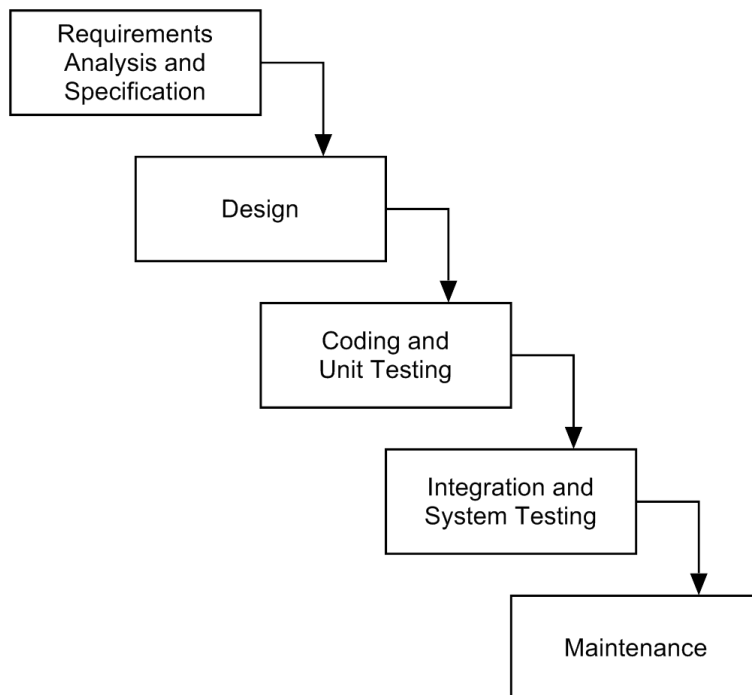
3. Studi Pustaka

Studi pustaka adalah metode pengumpulan data dengan mencari informasi lewat buku, majalah, koran, dan literatur lainnya yang bertujuan untuk membentuk sebuah landasan teori (Arikunto, 2006). Dari definisi tersebut maka penulis mencari sumber-sumber bahan yang dapat digunakan untuk mendukung penelitian ini diantaranya dengan mencari literatur pada penelitian sebelumnya dan membuka website yang ada di internet.

1.7. Metode Pengembangan Sistem

Dalam pengembangan sistem ini, penulis menggunakan metode pengembangan sistem *waterfall*. Metode *waterfall* melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing / verification*, dan *maintenance*. Sehingga pengerjaan dari suatu sistem dilakukan secara berurutan atau secara linier (Mall, 2009). Pada perancangan dan pembangunan Sistem Informasi Geografis ini, penulis hanya sampai tahap *Integration and System Testing* saja.

Secara umum, tahapan-tahapan pengembangan sistem pada metode *waterfall* dapat dilihat pada gambar 1.1.



Gambar 1.1 Metode *Waterfall*

(Sumber : Mall, 2009)

1. *Requirements Analysis and Specification*

Ini adalah langkah pertama yang paling penting dari model *waterfall*. Kegiatan ini terdiri dari mengumpulkan persyaratan dan kemudian menganalisa persyaratan yang terkumpul. Tujuan dari kegiatan pengumpulan persyaratan ini adalah untuk mengumpulkan semua informasi yang relevan mengenai sistem yang akan dikembangkan dengan maksud untuk memahami dengan jelas kebutuhan pengguna sistem.

Disini penulis mengumpulkan semua data yang dibutuhkan seperti nama gunung, foto gunung, lokasi gunung dan pos-pos pendakian. Sebagai persyaratan tahap pertama.

2. *Design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi kedalam bentuk *blueprint software* sebelum *coding* dimulai. *Design* harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Tahapan ini difokuskan pada proses desain struktur data, arsitektur sistem, representasi *interface*, dan algoritma program.

Ketika tahap pertama sudah terpenuhi maka lanjut ketahap kedua, penulis membuat sebuah *design* dari data yang diperoleh yaitu baik membuat relasi database dan tampilan dari aplikasi. Sebagai persyaratan tahap kedua.

3. *Coding and Unit Testing*

Setelah proses desain selesai maka hasilnya harus diterjemahkan kedalam bentuk program komputer yang kemudian menghasilkan suatu sistem. Kemudian program yang dibangun langsung diuji per *unit*. Selama *unit testing*, setiap modul diuji secara terpisah karena ini adalah cara yang paling efisien untuk mengidentifikasi kesalahan pada program.

Ketika tahap kedua sudah terpenuhi maka lanjut ketahap ketiga, penulis membuat aplikasi dari tahap kedua yang di implementasikan ke aplikasi android studio sebagai alat pembuatan aplikasi android. Sebagai persyaratan tahap ketiga.

4. *Integration and System Testing*

Kegiatan ini dilakukan untuk menemukan kesalahan-kesalahan yang memungkinkan terjadi pada proses pengkodean secara keseluruhan, serta memastikan bahwa program yang dibangun memberikan hasil yang sesuai dengan kebutuhan.

Ketika tahap ketiga sudah terpenuhi maka lanjut ketahap ke empat, pada tahap ini aplikasi sudah jadi tetapi aplikasi belum sepenuhnya jadi karena didalamnya belum kita tes disini penulis mengecek dengan klik bagian satu kebagian yang lain apakah ada kesalahan dari aplikasi tersebut. Jika ada kesalahan maka diperbaiki dengan melihat *error* didalam aplikasi sampai mendapatkan hasil yang diinginkan.

5. *Maintenance*

Ini adalah tahap akhir dari model *waterfall* dan dilakukan setelah instalasi sistem. Ketika sistem dijalankan tak jarang ditemukan cacat pada sistem yang tidak ditemukan sebelumnya sehingga mengharuskan *programer* memperbaiki atau meningkatkan kinerja sistem.

Penulis tidak sampai ketahap ini karena biasanya pada tahap *Maintenance* aplikasi sudah dipublis kedia dan perlu menambahkan fitur baru.

1.8. Sistematika Penulisan

Untuk memberikan gambaran secara menyeluruh masalah yang akan dibahas dalam penulisan laporan tugas akhir ini, maka sistematika penulisan dibagi dalam lima bab sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, identifikasi masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metode penelitian, metode pengembangan sistem, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bab ini menguraikan tinjauan pustaka terdahulu dan definisi dari teori-teori yang mendasari pembahasan secara detail.

BAB III ANALISIS DAN PERANCANGAN

Bab ini membahas tentang gambaran umum Dinas Kepemudaan dan Olahraga dan Pariwisata Kabupaten Pekalongan, serta menguraikan analisis sistem informasi wisata yang sedang berjalan dan perancangan sistem informasi geografis wisata yang diusulkan yang mencakup DFD, ERD, dan rancangan basis data.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini membahas tentang perancangan sistem informasi geografis wisata yang meliputi perancangan tampilan sistem dan hasil rancangan sistem.

BAB V PENUTUP

Bab ini mengulas tentang kesimpulan dan saran.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Bab ini akan membahas tinjauan aplikasi-aplikasi terdahulu yang sejenis dengan aplikasi ini, beberapa penelitian tentang Sistem Informasi Geografis sudah pernah dilakukan sebelumnya. Diantaranya penelitian dengan judul:

Sistem Informasi Geografis Objek Wisata Wilayah Puncak Kabupaten Bogor Berbasis Android. Penelitian ini dilatar belakangi oleh sektor pariwisata memberikan kontribusi yang besar untuk kabupaten Bogor khususnya wilayah puncak, akan tetapi informasi tempat wisata puncak sangat minim dengan kemajuan teknologi. Maka dari itu penulis membangun aplikasi peta petunjuk objek pariwisata untuk wilayah Puncak kabupaten Bogor yang berguna dalam memperoleh informasi daerah-daerah wisata dan penunjuk untuk para wisatawan yang menggunakan ponsel sistem operasi android 2.3.3 dan dapat diakses secara online (Saputra, 2012).

Sistem Informasi Geografis Track Gunung Awu Berbasis Web. Penelitian ini dilatar belakangi minimnya informasi tentang destinasi wisata alam di Kabupaten Kepulauan Sangihe dan belum dikelola secara maksimal, salah satunya wisata alam Gunung Awu. Padahal di Kabupaten Kepulauan Sangihe, Sulawesi Utara merupakan salah satu daerah yang memiliki berbagai jenis potensi wisata seperti wisata alam, wisata budaya maupun wisata sejarah. Maka dari itu penulis membangun sebuah Sistem Informasi Geografis Track Gunung Awu Berbasis Web dengan adanya sistem ini, diharapkan dapat membantu untuk memberikan

informasi yang cepat dan akurat, serta memudahkan dalam penentuan jalur yang akan dilalui dari titik *start* hingga sampai ke puncak Gunung Awu (Mua, 2017).

Persamaan dari penelitian-penelitian diatas dengan penelitian ini yaitu untuk memberikan kemudahan bagi wisatawan dalam mendapatkan informasi tentang daerah-daerah wisata.

Perbedaan dari penelitian-penelitian diatas dengan penelitian ini adalah Sistem Informasi Geografis Wisata Gunung di Pekalongan berbasis Android dengan sistem operasi terbaru, sehingga diharapkan memiliki beberapa kelebihan baik dari segi kemudahan operasional maupun kecepatan akses. Selain itu, aplikasi ini juga dilengkapi Sistem Pemosisi Global (*Global Positioning System*) sehingga bisa mengetahui jarak pengguna dengan tempat wisata. Sistem Informasi Geografis Wisata Gunung ini terintegrasi dengan aplikasi *Maps* sehingga bisa memberikan rute langsung ke lokasi tujuan.

2.2. Landasan Teori

2.2.1. Sistem

Terdapat dua kelompok pendekatan didalam pendefinisian sistem. Pendekatan yang menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan sistem sebagai kumpulan elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Sutabri, 2012).

Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Sebagai gambaran, jika dalam sebuah sistem terdapat elemen yang tidak memberikan manfaat dalam mencapai tujuan yang sama, maka elemen tersebut dapat dipastikan bukanlah bagian dari sistem (Kadir, 2014).

Berdasarkan dari dua definisi diatas maka penulis menyimpulkan bahwa sistem adalah sekumpulan elemen yang menekankan pada prosedur yang saling berkaitan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.

2.2.2. Informasi

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya (Sutabri, 2012).

Informasi adalah hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang relevan yang dibutuhkan oleh orang untuk menambah pemahamannya terhadap fakta-fakta yang ada. Informasi bagi setiap elemen akan berbeda satu sama lain sesuai dengan kebutuhannya masing-masing (Oetomo, 2002).

Berdasarkan pengertian-pengertian yang telah disebutkan diatas, dapat disimpulkan bahwa informasi adalah sekumpulan fakta-fakta yang telah diolah menjadi bentuk data, sehingga dapat menjadi lebih berguna dan dapat digunakan

oleh siapa saja yang membutuhkan data-data tersebut sebagai pengetahuan ataupun dapat digunakan dalam pengambilan keputusan.

2.2.3. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu (Sutabri, 2012).

2.2.4. Sistem Informasi Geografis

Sistem Informasi Geografis (SIG) adalah sistem komputer yang digunakan untuk mengumpulkan, memeriksa, mengintegrasikan, dan menganalisa informasi-informasi yang berhubungan dengan permukaan bumi. Pada dasarnya, istilah sistem informasi geografi merupakan gabungan dari tiga unsur pokok yaitu sistem, informasi, dan geografi (Prahasta, 2002). 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. SIG merupakan suatu sistem yang menekankan pada unsur informasi geografi. 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. Istilah “informasi

geografis” mengandung pengertian informasi mengenai tempat-tempat yang terletak di permukaan bumi, pengetahuan mengenai posisi dimana suatu objek terletak di permukaan bumi, dan informasi mengenai keterangan-keterangan (atribut) yang terdapat di permukaan bumi yang posisinya diberikan atau diketahui.

2.2.4.1 Subsistem SIG

Sub sistem yang dimiliki oleh SIG yaitu data input, data output, data management, data manipulasi dan analisis. Sub sistem SIG tersebut dijelaskan dibawah ini:

1. Data Masukan: Sub sistem ini bertugas untuk mengumpulkan dan mempersiapkan data spasial (keruangan) dan data atribut dari berbagai sumber. Sub sistem ini pula yang bertanggung jawab dalam mengkonversi atau mentransformasi format data-data aslinya ke dalam format yang digunakan oleh SIG.
2. Data Keluaran: Sub sistem ini menampilkan atau menghasilkan keluaran seluruh atau sebagian basis data baik dalam bentuk *softcopy* maupun bentuk *hardcopy* seperti: tabel, grafik, peta dan lain-lain.
3. Data *Management*: Sub sistem ini mengorganisasikan baik data spasial maupun atribut ke dalam sebuah basis data sedemikian rupa sehingga mudah dipanggil, dan diubah.
4. Data manipulasi dan analisis: Sub sistem ini menentukan informasi-informasi yang dapat dihasilkan oleh SIG. Selain itu, sub sistem ini juga melakukan

manipulasi dan permodelan data untuk menghasilkan informasi yang diharapkan.

2.2.4.2 Komponen SIG

SIG merupakan sistem kompleks yang biasanya terintegrasi dengan lingkungan sistem-sistem komputer yang lain di tingkat fungsional dan jaringan.

1. Perangkat keras (*Hardware*): Pada saat ini SIG tersedia untuk berbagai platform perangkat keras mulai dari PC *desktop*, *workstations*, hingga *multiuser host* yang dapat digunakan oleh banyak orang secara bersamaan dalam jaringan komputer yang luas, berkemampuan tinggi, memiliki ruang penyimpanan (*harddisk*) yang besar, dan mempunyai kapasitas memori (RAM) yang besar. Walaupun demikian, fungsionalitas SIG tidak terikat secara ketat terhadap karakteristik-karakteristik fisik perangkat keras ini sehingga keterbatasan memori pada PC30 pun dapat diatasi. Adapun perangkat keras yang sering digunakan untuk SIG adalah komputer (PC), *mouse*, *digitizer*, *printer*, *plotter*, dan *scanner*.
2. Data dan Informasi Geografi: SIG dapat mengumpulkan dan menyimpan data dan informasi yang diperlukan baik secara tidak langsung dengan cara mengimport-nya dari perangkat-perangkat lunak SIG yang lain maupun secara langsung dengan cara mendigitasi data spasialnya dari peta dan memasukkan data atributnya dari table-tabel dan laporan dengan menggunakan keyboard.

3. Manajemen: Suatu proyek SIG akan berhasil jika dimanage dengan baik dan dikerjakan oleh orang-orang memiliki keahlian yang tepat pada semua tingkatan.

2.2.5. Android

Android merupakan OS (*Operating System*) *Mobile* yang tumbuh ditengah OS lainnya yang berkembang dewasa ini. OS lainnya seperti *Windows Mobile*, *i-Phone OS*, *Symbian*, dan masih banyak lagi. Akan tetapi, OS yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh karena itu, adanya keterbatasan dari aplikasi pihak ketiga untuk mendapatkan data asli ponsel, berkomunikasi antar proses serta keterbatasan distribusi aplikasi pihak ketiga untuk platform mereka (Susanto, 2011).

Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia*. Pada saat perilisan perdana *Android*, 5 November 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode *Android* di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Berikut adalah rangkaian sejarah perkembangan Android yang resmi diluncurkan oleh Google, dari waktu ke waktu.

Tabel 2.1 Perkembangan Sistem Operasi Android
(Sumber : Dicoding, 2018)

Version	Code name	Release date	API level	DVM/ART	New features	Icon
9	Android P	-	-	-	-	-
8.1	Oreo	October 25, 2017	26 - 27	ART	<ul style="list-style-type: none"> Picture-in-Picture 	
7.1 - 7.1.2	Nougat	August 22, 2016	24 - 25	ART	<ul style="list-style-type: none"> Multi window GIF Keyboard 	
6.0 - 6.0.1	Marshmallow	October 5, 2015	23	ART	<ul style="list-style-type: none"> Now On Tap Permissions Battery (Doze & App Standby) 	
5.1 - 5.1.1	Lollipop	November 12, 2014	21 - 22	ART	<ul style="list-style-type: none"> Material Design Multiscreen Notifications 	
4.4 - 4.4.4	KitKat	October 31, 2013	19 - 20	DVM (and ART 1.6.0)	<ul style="list-style-type: none"> Voice : Ok Google Immersive Design Smart Dialer 	
4.1 - 4.3.1	Jelly Bean	July 9, 2012	18	DVM	<ul style="list-style-type: none"> Google Now Actionable Notifications Account Switching 	
4.0 - 4.6	Ice Cream Sandwich	October 19, 2011	15	DVM	<ul style="list-style-type: none"> Custom Home Screen Data Usage Control Android Beam 	
3.0 - 3.2.6	HoneyComb	February 22, 2011	11 - 13	DVM	<ul style="list-style-type: none"> Tablet-Friendly Design System Bar Quick Settings 	
2.3 - 2.3.7	Gingerbread	February 9, 2011	9 - 10	DVM	<ul style="list-style-type: none"> Gaming APIs NFC Battery Management 	
2.2 - 2.2.3	Froyo	May 20, 2010	8	DVM	<ul style="list-style-type: none"> Voice Action Portable Hotspot Dalvik JIT 	
2.0 - 2.1	Eclair	October 26, 2009	5	-	<ul style="list-style-type: none"> Google Maps Navigation Home Screen Customization Speech-to-Text 	
1.6	Donut	September 15, 2009	4	-	<ul style="list-style-type: none"> Quick Search Box Screen Size Diversity Android Market 	
1.5	Cupcake	April 27, 2009	3	-	-	

2.2.6. Google Map

Google map adalah layanan gratis yang diberikan oleh *Google* dan sangat populer. *Google Maps* adalah suatu peta dunia yang dapat kita gunakan untuk melihat suatu daerah. Dengan kata lain, *Google Maps* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Kita dapat menambahkan fitur *Google Maps* dalam web yang telah kita buat atau pada blog kita yang berbayar maupun gratis sekalipun dengan *Google Maps API*. *Google Maps API* adalah suatu library yang berbentuk *JavaScript* (Yusro, 2013).

2.2.7. Gunung

Gunung adalah bentuk muka bumi yang menonjol dari rupa bumi di sekitar. Gunung biasanya lebih tinggi dan curam dibandingkan bukit. Gunung dan pegunungan terbentuk karena pergerakan kerak bumi yang menjulang naik. Jika kedua kerak bumi menjulang naik, pegunungan dihasilkan, sebaliknya jika salah satu kerak bumi terlipat bawah kerak yang lain, gunung berapi terbentuk. Gunung adalah sebuah bentuk tanah yang menonjol di atas wilayah sekitarnya. Sebuah gunung biasanya lebih tinggi dan curam dari sebuah bukit, tetapi ada kesamaan (Abdi, 2013).

Jenis Pegunungan :

1. Pegunungan rendah, tingginya 200-500 m dari permukaan air laut.
2. Pegunungan menengah, tingginya 500-1.500 m dari permukaan laut.
3. Pegunungan tinggi, tingginya 1.500 m ke atas

2.2.8. Basis Data

2.2.8.1 Pengertian Basis Data

Basis data adalah kumpulan terorganisasi dari data-data yang berhubungan sedemikian rupa sehingga mudah disimpan, dimanipulasi, serta dipanggil oleh pengguna (Nugroho, 2011).

Basis data adalah kumpulan data berelasi yang disusun, diorganisasikan, dan disimpan secara sistematis dalam media simpan komputer mengacu kepada metode-metode tertentu sedemikian rupa sehingga dapat diakses secara cepat dan mudah menggunakan program atau aplikasi komputer untuk memperoleh data dari basis data tersebut (Ichwan, 2011).

Dari beberapa definisi-definisi tersebut, dapat dikatakan bahwa basis data adalah kumpulan data yang dirancang dan dibangun agar dapat digunakan oleh beberapa pengguna untuk berbagai kepentingan.

2.2.8.2 Pengertian DBMS

DBMS (*Database Management System*) adalah sistem yang secara khusus dibuat untuk memudahkan pemakai dalam mengelola basis data. Sistem ini dibuat untuk mengatasi kelemahan sistem pemrosesan yang berbasis berkas (Kadir dan Triwahyuni, 2003).

Untuk menyediakan kemudahan bagi pemakai dan meningkatkan produktivitas, DBMS menyediakan sejumlah perangkat produktivitas seperti pembangkit *query* dan pembangkit laporan.

2.2.8.3 Komponen Lingkungan Basis Data

Komponen yang menyusun lingkungan DBMS terdiri atas :

a. Perangkat Keras

Perangkat keras digunakan untuk menjalankan DBMS beserta aplikasi-aplikasinya. Perangkat keras berupa komputer dan periferal pendukungnya. Komputer dapat berupa PC, minikomputer, *mainframe*, dll.

b. Perangkat Lunak

Komponen perangkat lunak mencakup DBMS itu sendiri, program aplikasi, serta perangkat lunak pendukung untuk komputer dan jaringan. Program aplikasi dapat dibangun dengan menggunakan bahasa pemrograman C++, Pascal, Delphi, atau Visual Basic.

c. Data

Bagi sisi pemakai, komponen terpenting dalam DBMS adalah data karena dari data inilah pemakai dapat memperoleh informasi yang sesuai dengan kebutuhan masing-masing.

d. Prosedur

Prosedur adalah petunjuk tertulis yang berisi cara merancang hingga menggunakan basis data.

e. Orang

Komponen orang dapat dibagi menjadi tiga kelompok :

- 1) Pemakai akhir (*end-user*).
- 2) Pemrogram aplikasi.
- 3) Administrator basis data.

2.2.8.4 Model Basis Data

Model data adalah sekumpulan konsep terintegrasi yang dipakai untuk menjabarkan data, hubungan antar data, dan kekangan terhadap data yang digunakan untuk menjaga konsistensi. Kadang, model data disebut struktur data logis (Kadir, 2014).

Model data yang umum pada saat ini ada empat macam, yaitu :

1. Model Data Hierarkis

Model ini seringkali dijabarkan dalam bentuk pohon terbalik. Di dalam model ini dikenal istilah orang tua dan anak. Masing-masing berupa suatu simpul dan terdapat hubungan bahwa setiap anak hanya dapat memiliki satu orang tua, sedangkan orang tua dapat memiliki sejumlah anak.

2. Model Data Jaringan

Model data jaringan menyerupai model hierarkis, tetapi dengan perbedaan:

- a) Tidak mengenal akar.
- b) Setiap anak bisa memiliki lebih dari satu orang tua.

3. Model Data Relasional

Model data relasional menggunakan sekumpulan tabel berdimensi dua (yang biasa disebut relasi atau tabel), dengan masing-masing tabel tersusun atas sejumlah baris dan kolom.

4. Model Data Berbasis Objek

Model data berbasis objek adalah model data yang menerapkan teknik pemrograman berorientasi objek. Berbeda dengan tiga model yang telah dibahas di

depan, model data berbasis objek mengemas data dan fungsi untuk mengakses data (metode) kedalam bentuk objek.

2.2.9. Desain Sistem

Desain sistem dibagi menjadi dua sub tahapan, yakni perancangan konseptual dan perancangan fisik. Keduanya memiliki sejumlah aktivitas. Target akhir tahapan ini adalah menghasilkan rancangan yang memenuhi kebutuhan yang ditentukan selama tahapan analisis sistem. Hasil akhirnya berupa spesifikasi rancangan yang sangat rinci sehingga mudah diwujudkan pada saat pemrograman (Kadir, 2003).

2.2.9.1 Flowchart (Diagram Alur)

Bagan alir (*flowchart*) adalah suatu bentuk grafik atau diagram dari algoritma dimana simbol-simbol standart mewakili tampilan operasi yang perlu dan memperlihatkan urutan pelaksanaannya (Amsyah, 2001). *Flowchart* menolong analis dan *programmer* untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

Adapun jenis-jenis bagan alir yang digunakan, yaitu :


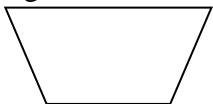
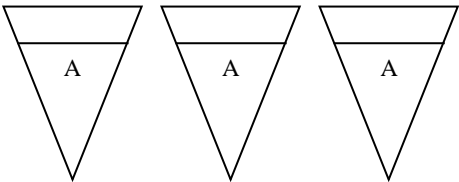

a. Bagan Alir Sistem

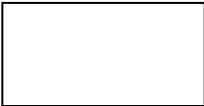

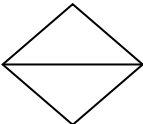
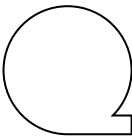


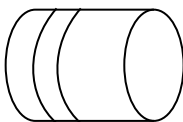
Pendesain program dapat mulai dengan suatu bagan alir sistem, yaitu bagan alir yang memberikan gambaran luas keseluruhan operasi tanpa penguraian semua langkah *input* spesifik, pengolahan, dan output yang sebenarnya akan dilaksanakan.

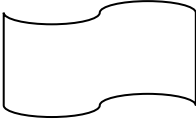

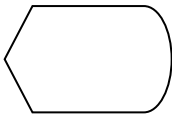
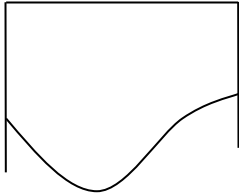

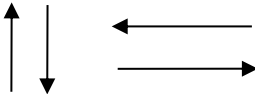
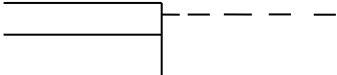
Idenya adalah untuk menampilkan gambaran total tanpa kuatir akan tiap detail yang kecil. Suatu bagan alir sistem memasukkan komponen-komponen dari beberapa program komputer yang terpisah (Amsyah, 2001). Dengan kata lain, *flowchart* ini merupakan deskripsi secara grafik dari urutan prosedur-prosedur yang terkombinasi yang membentuk suatu sistem. Simbol-simbol *flowchart* sistem, yaitu :

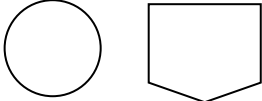
Tabel 2.2 Simbol *Flowchart* Sistem

(Sumber : Kusrini, 2007)

Simbol	Keterangan
<p>Simbol dokumen</p> 	Menunjukkan dokumen input dan output baik untuk proses manual, mekanik atau komputer.
<p>Simbol kegiatan manual</p> 	Menunjukkan pekerjaan manual.
<p>Simbol simpanan <i>offline</i></p> 	File non komputer yang diarsip urut angka (<i>numerical</i>), huruf (<i>alphabetical</i>), tanggal (<i>chronological</i>).
<p>Simbol kartu plong</p> 	Menunjukkan input atau output yang menggunakan kartu plong (<i>punched card</i>).

Simbol	Keterangan
Simbol proses 	Menunjukkan kegiatan proses dari operasi program komputer.
Simbol operasi luar 	Menunjukkan operasi yang dilakukan di luar proses operasi komputer.
Simbol pengurutan <i>offline</i> 	Menunjukkkan proses pengurutan data di luar proses komputer.
Simbol peta magnetik 	Menunjukkan <i>input</i> atau <i>output</i> menggunakan pita magnetik.
Simbol <i>harddisk</i> 	Menunjukkan <i>input</i> atau <i>output</i> menggunakan <i>harddisk</i> .
Simbol <i>diskette</i> 	Menunjukkan <i>input</i> atau <i>output</i> menggunakan <i>diskette</i> .
Simbol drum magnetik 	Menunjukkan <i>input</i> atau <i>output</i> menggunakan drum magnetik.

Simbol	Keterangan
<p>Simbol pita kertas berlubang</p> 	<p>Menunjukkan <i>input</i> atau <i>output</i> menggunakan pita kertas berlubang.</p>
<p>Simbol <i>keyboard</i></p> 	<p>Menunjukkan <i>input</i> yang menggunakan <i>on-line keyboard</i>.</p>
<p>Simbol <i>display</i></p> 	<p>Menunjukkan <i>output</i> yang ditampilkan di monitor.</p>
<p>Simbol pita kontrol</p> 	<p>Menunjukkan penggunaan pita kontrol (<i>control tape</i>) dalam <i>batch control total</i> untuk pencocokan diproses <i>batch processing</i>.</p>
<p>Simbol hubungan komunikasi</p> 	<p>Menunjukkan proses transmisi data melalui <i>channel</i> komunikasi.</p>
<p>Simbol garis alir</p> 	<p>Menunjukkan arus dari proses.</p>
<p>Simbol penjelasan</p> 	<p>Menunjukkan penjelasan dari suatu proses.</p>

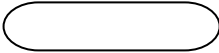

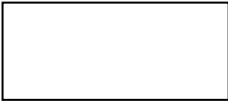
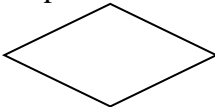
Simbol	Keterangan
Simbol penghubung 	Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.

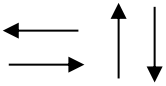

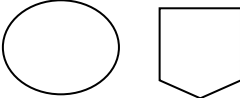

b. Bagan Alir Program

Flowchart Program merupakan keterangan yang lebih rinci tentang bagaimana setiap langkah program atau prosedur sesungguhnya dilaksanakan. *Flowchart* ini menunjukkan setiap langkah program atau prosedur dalam urutan yang tepat saat terjadi. Dan dibuat dari derivikasi bagan alir sistem (Zulkifli, 2001).

Tabel 2.3 Simbol *Flowchart* Program

(Sumber : Kusrini, 2007)

Simbol	Keterangan
Simbol titik terminal 	Simbol ini digunakan untuk menunjukkan awal dan akhir suatu proses.
Simbol <i>input/output</i> 	Simbol ini digunakan untuk mewakili data <i>input</i> atau <i>output</i> .
Simbol proses 	Simbol ini digunakan untuk mewakili suatu proses komputer.
Simbol keputusan 	Simbol ini digunakan untuk suatu penyeleksian kondisi di dalam program.

Simbol	Keterangan
Simbol garis alir 	Simbol ini digunakan untuk menunjukkan arus dari proses.
Simbol proses terdefinisi 	Simbol ini digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
Simbol penghubung 	Simbol ini digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman lainnya.
Simbol persiapan 	Simbol ini digunakan untuk memberi nilai awal suatu besaran.


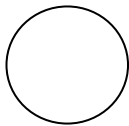
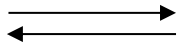

2.2.9.2 DFD (Data Flow Diagram)

Data flow diagram konteks merupakan suatu *network* yang menggambarkan suatu sistem automat/komputerisasi, manualisasi atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya (Sutabri, 2012).

Simbol-simbol DFD (*Data Flow Diagram*) yaitu :

Tabel 2.4 Komponen-Komponen DFD

(Sumber : Sutabri, 2012)

Simbol	Keterangan
External <i>entity</i> 	Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
Proses 	Simbol ini digunakan untuk proses pengolahan atau transformasi data.
<i>Data flow</i> 	Simbol ini digunakan untuk menggambarkan aliran data yang berjalan.
<i>Data store</i> 	Simbol ini digunakan untuk menggambarkan <i>data flow</i> yang sudah disimpan atau diarsipkan.

2.2.9.3 ERD (Entity Relationship Diagram)


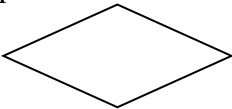
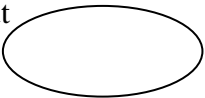

ERD merupakan suatu model data yang dikembangkan berdasarkan objek (Sutanta, 2011). Istilah-istilah dalam ERD :

- Entitas : adalah sesuatu atau objek di dunia nyata (*real word*) yang dapat dibedakan dengan sesuatu atau objek lainnya.
- Atribut : merupakan properti yang menjelaskan kriteria dari entitas. Penentuan atribut-atribut yang relevan bagi sebuah entitas merupakan hal yang penting dalam pembentukan model data.

- c. Relasi : merupakan salah satu kata kunci dalam *database* relasional. Relasi menunjukkan adanya hubungan antara entitas yang berbeda.
- d. Derajat relasi / kardinalitas : merupakan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Ada beberapa tipe hubungan antar entitas, yaitu *one to one*, *one to many*, *many to one*, dan *many to many*.

Tabel 2.5 Simbol dalam ERD

(Sumber : Nugroho, 2011)

Simbol	Keterangan
Entitas 	Suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
Relasi 	Menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
Atribut 	Berfungsi mendeskripsikan karakter entitas.
Garis 	Sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

2.2.10. *Software* Pembangun Aplikasi

2.2.10.1 Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA (Developers, 2018). Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya:

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

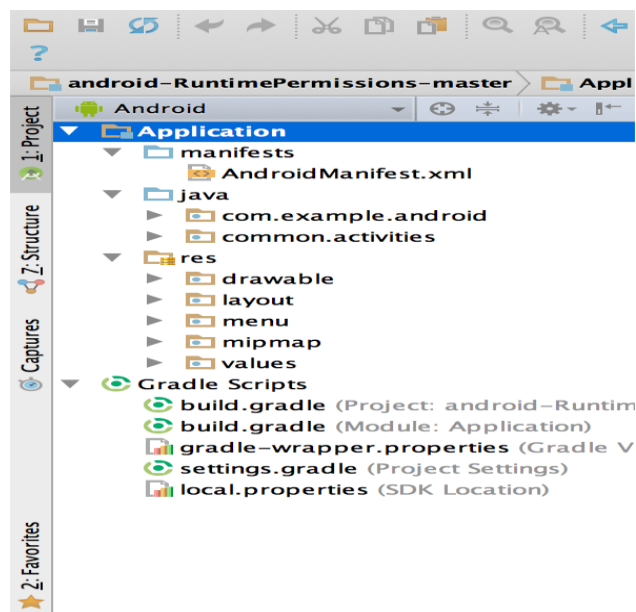
2.2.10.2 Struktur Proyek Android Studio

Setiap proyek di Android Studio berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup:

1. Modul aplikasi Android
2. Modul Pustaka
3. Modul *Google App Engine*

Secara default, Android Studio akan menampilkan file proyek Anda dalam tampilan proyek Android, seperti yang ditampilkan dalam gambar 2.1. Tampilan disusun berdasarkan modul untuk memberikan akses cepat ke file sumber utama proyek Anda. Semua file versi terlihat di bagian atas di bawah *Gradle Scripts* dan masing-masing modul aplikasi berisi folder berikut:

1. *Manifests*: Berisi file *AndroidManifest.xml*
2. *java*: Berisi file kode sumber Java, termasuk kode pengujian JUnit.
3. *res*: Berisi semua sumber daya bukan kode, seperti tata letak *XML*, *string UI*, dan gambar *bitmap*.

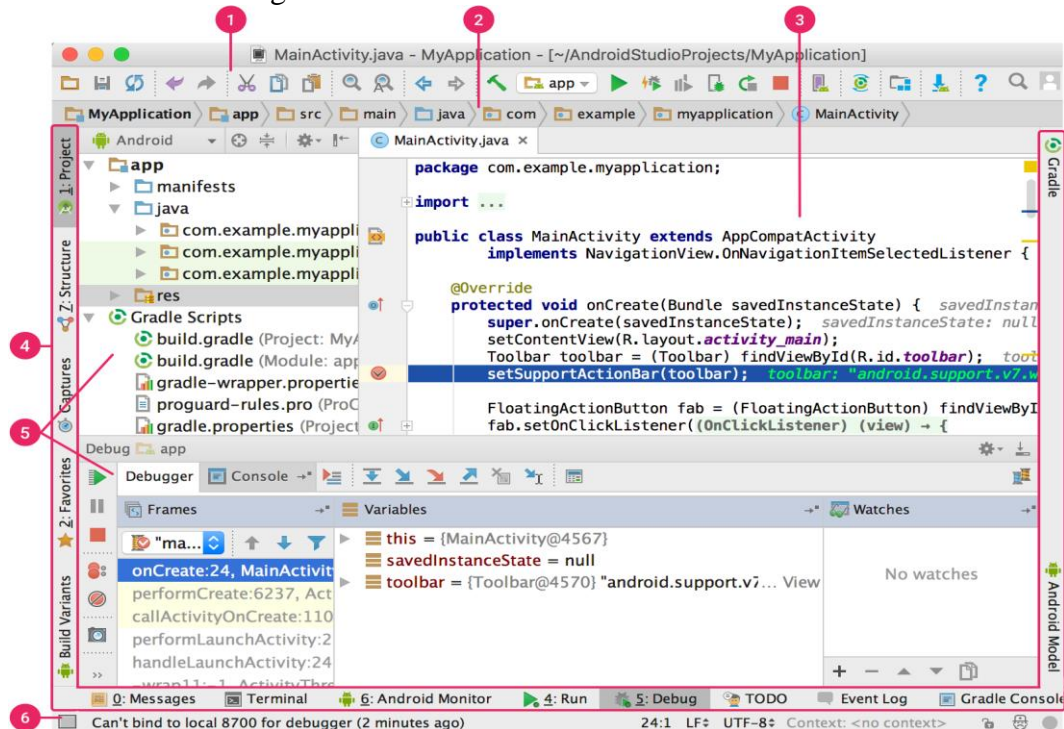


Gambar 2.1 File proyek di tampilan Android

(Sumber : Developers, 2018)

2.2.10.3 Antarmuka Pengguna Android Studio

Jendela utama Android Studio terdiri dari beberapa bidang logika yang diidentifikasi dalam gambar 2.2.



Gambar 2.2 Jendela utama Android Studio

(Sumber : Developers, 2018)

1. Bilah alat memungkinkan Anda untuk melakukan berbagai jenis tindakan, termasuk menjalankan aplikasi dan meluncurkan alat Android.
2. Bilah navigasi membantu Anda bernavigasi di antara proyek dan membuka file untuk diedit. Bilah ini memberikan tampilan struktur yang terlihat lebih ringkas dalam jendela Project.
3. Jendela editor adalah tempat Anda membuat dan memodifikasi kode. Bergantung pada jenis file saat ini, editor dapat berubah. Misalnya, ketika melihat file tata letak, editor menampilkan Layout Editor.

4. Bilah jendela alat muncul di luar jendela IDE dan berisi tombol yang memungkinkan Anda meluaskan atau menciutkan jendela alat individual.
5. Jendela alat memberi Anda akses ke tugas tertentu seperti pengelolaan proyek, penelusuran, kontrol versi, dan banyak lagi. Anda bisa meluaskan dan juga menciutkannya.
6. Bilah status menampilkan status proyek Anda dan IDE itu sendiri, serta setiap peringatan atau pesan.

2.2.10.4 JAVA

Bahasa pemrograman Java awalnya dikembangkan oleh *Sun Microsystem* yang diprakarsai oleh James Gosling dan dirilis pada tahun 1995 sebagai komponen inti dari platform *Java Sun Microsystems* (Java 1.0 [J2SE]) (Tutorialspoint, 2018).

Rilis terbaru dari *Java Standard Edition* adalah Java SE 8. Dengan kemajuan Java dan popularitasnya yang tersebar luas, beberapa konfigurasi dibangun untuk menyesuaikan berbagai jenis platform. Misalnya: *J2EE* untuk Aplikasi *Enterprise*, *J2ME* untuk Aplikasi Seluler.

2.2.10.5 PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen HTML. Pembuatan *web* ini merupakan kombinasi antara PHP sendiri sebagai bahasa pemrograman dan HTML sebagai

pembangun halaman *web*. Ketika seorang pengguna internet akan membuka suatu situs yang menggunakan fasilitas *server-side scripting* PHP, maka terlebih dahulu *server* yang bersangkutan akan memproses format HTML ke *web browser* pengguna internet tadi. Dengan demikian seorang pengguna internet tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman *web* menjadi lebih terjamin.

Tetapi tidak seperti ASP yang juga cukup dikenal sebagai *server-side scripting*, PHP merupakan *software* yang *Open Source* (gratis) dan mampu lintas *Platform*, yaitu dapat digunakan dengan sistem operasi dan *web server* apapun. PHP mampu berjalan di Windows dan beberapa versi Linux. PHP juga dapat dibangun sebagai modul pada *web server Apache* dan sebagai *binary* yang dapat berjalan sebagai CGI (Sunarfrihantono, 2003).

2.2.10.6 MySQL

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar SQL (*Structured Query Language*). MySQL memiliki dua bentuk lisensi, yaitu FreeSoftware dan Shareware. MySQL yang biasa kita gunakan adalah MySQL FreeSoftware yang berada di bawah Lisensi GNU/GPL (*General Public License*). Selain itu anda juga dapat memiliki produk MySQL yang sifatnya komersial, biasa disebut dengan MySQL AB.

MySQL merupakan sebuah *database server* yang *free*, artinya kita bebas menggunakan *database* ini untuk keperluan pribadi atau usaha tanpa harus membeli

atau membayar lisensinya. MySQL pertama kali dirintis oleh seorang programmer *database* bernama Michael Widenius. Selain sebagai *database server*, MySQL juga merupakan program yang dapat mengakses suatu *database* MySQL yang berposisi sebagai *Client*. Jadi MySQL adalah sebuah *database* yang dapat digunakan baik sebagai *Client* maupun *Server* (Nugroho, 2005).

BAB III

ANALISIS DAN PERANCANGAN

3.1. Gambaran Umum DINPORAPAR

3.1.1. Sejarah Singkat DINPORAPAR

Dinas kepemudaan dan olahraga dan pariwisata dibentuk berdasarkan peraturan daerah kabupaten Pekalongan nomor 4 tahun 2016 tentang pembentukan dan susunan perangkat daerah kabupaten Pekalongan di tindak lanjuti dengan penerbitan peraturan bupati Pekalongan nomor 45 tahun 2016 tentang kedudukan, susunan organisasi, tugas dan fungsi serta tata kerja dinas daerah kabupaten Pekalongan. Berpijak dari kebijakan daerah tersebut, kedudukan dinporapar kabupaten Pekalongan adalah merupakan unsur pelaksanaan pemerintah kabupaten yang dipimpin oleh seorang kepala, yang berada di bawah dan bertanggung jawab kepada bupati melalui sekretaris daerah.

Dinas Pemuda Olahraga dan pariwisata mempunyai fungsi :

1. Perumusan kebijakan teknis di bidang Pemuda Olahraga dan Pariwisata
2. Penyelenggaraan urusan pemerintahan dan pelayanan umum bidang pemuda, olahraga dan pariwisata
3. Pemantauan, evaluasi dan pelaporan bidang pemuda, olahraga dan pariwisata
4. Pelaksanaan tugas lain yang diberikan oleh bupati sesuai dengan tugas dan fungsinya

3.1.2. Visi dan Misi DINPORAPAR

3.1.2.1 Visi

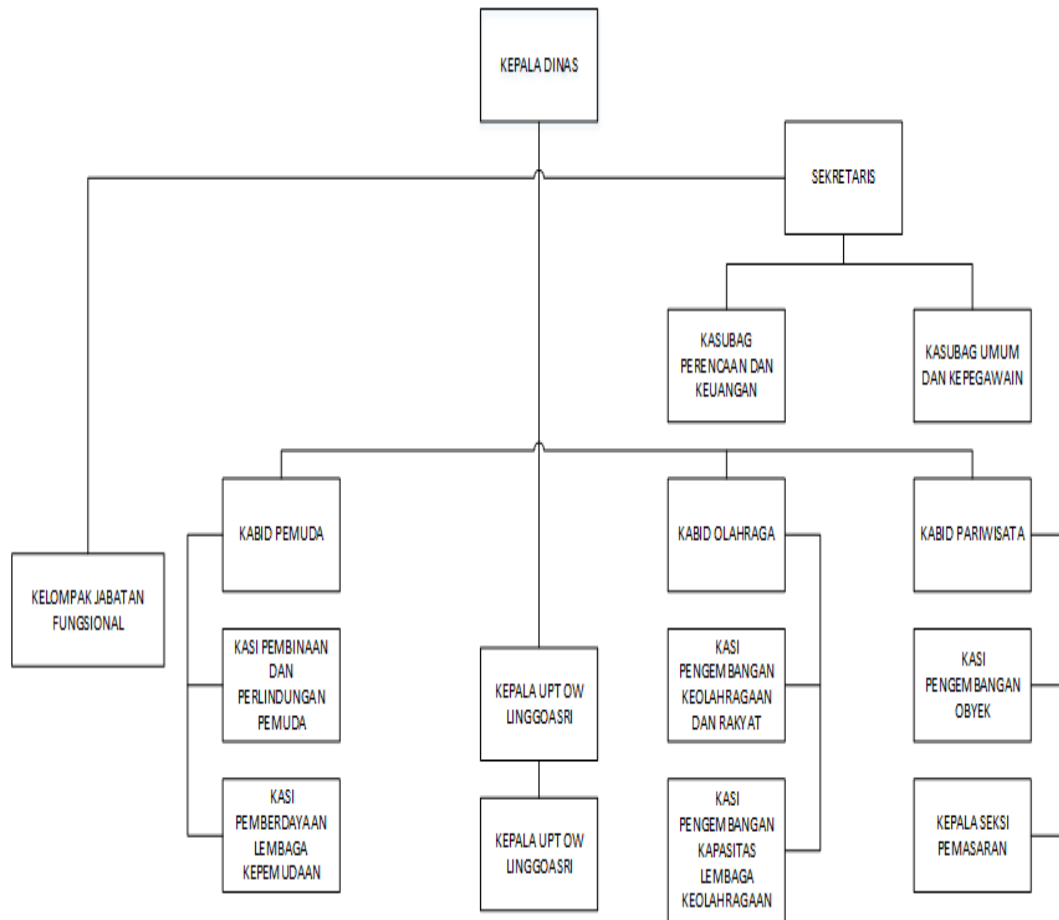
“ Mewujudkan Pemuda Dan Olahraga Unggul dan Berprestasi Optimal Serta Mewujudkan Pariwisata Andalan Demi Kesejahteraan Masyarakat Kabupaten Pekalongan ”

3.1.2.2 Misi

1. Menumbuh kembangkan terwujudnya kreatifitas generasi muda yang berakhlakul karimah, mandiri, demokratis, dan berprestasi.
2. Membangun sistem pembinaan dan pengembangan keolahragaan dengan lembaga terkait secara optimal dalam pencapaian prestasi daerah
3. Meningkatkan potensi pariwisata menuju pemberdayaan masyarakat ,peningkatan pendapatan masyarakat dan peningkatan pendapatan asli daerah

3.1.3. Struktur Organisasi

Struktur organisasi merupakan suatu gambaran yang dapat memberikan informasi tentang bagaimana pola hubungan kerja, fungsi, tugas, tanggung jawab, dan wewenang antar bagian-bagian yang ada dalam suatu organisasi. Adapun struktur organisasi yang berjalan di DINPORAPAR adalah sebagai berikut:



Gambar 3.1 Struktur Organisasi

Penjelasan dari gambar 3.1 adalah sebagai berikut :

1. Kepala Dinas

Kepala Dinas Kepemudaan Dan Olahraga dan Pariwisata Kabupaten Pekalongan mempunyai tugas pokok membantu Bupati dalam melaksanakan penyusunan dan pelaksanaan kebijakan daerah di bidang kepemudaan, olahraga dan pariwisata.

Kepala Dinas Kepemudaan Dan Olahraga dan Pariwisata Kabupaten Pekalongan mempunyai fungsi:

- a. Perumusan kebijakan teknis bidang pemuda, olahraga dan pariwisata
- b. Pembinaan dan pelaksanaan tugas bidang pemuda, olahraga dan pariwisata

- c. Penyelenggaraan urusan pemerintahan dan pelayanan umum bidang pemuda, olahraga dan pariwisata

2. Sekretaris

Sekretariat dipimpin Oleh seorang Sekretaris yang berkedudukan di bawah dan bertanggung jawab kepada Kepala Dinas yang mempunyai tugas pokok Melaksanakan penyusunan program, keuangan, umum dan kepegawaian.

Dalam melaksanakan tugas pokok, Sekretaris mempunyai fungsi:

- a. Pelaksanaan koordinasi, penyusunan rencana dan program, evaluasi serta pelaporan di bidang pemuda, olahraga dan pariwisata
- b. Pengelolaan keuangan dan perbendaharaan
- c. Pengelolaan surat-menyurat, ekspedisi, arsip, perlengkapan, rumah tangga, perjalanan dinas dan ketatalaksanaan

3. Bidang Repemudaan

Bidang Kepemudaan dipimpin oleh seorang Kepala Bidang yang berkedudukan di bawah dan bertanggung jawab kepada Kepala Dinas melalui Sekretaris.

Dalam melaksanakan tugas pokok Kepala Bidang Kcpemudaan mempunyai fungsi:

- a. Pelaksanaan koordinasi, pembinaan, pengembangan dan perlindungan pemuda
- b. Pelaksanaan koordinasi, pembinaan, pengembangan dan pemberdayaan lembaga kepemudaan
- c. Pelaksanaan tugas-tugas lain yang diberikan Oleh atasan sesuai dengan tugas dan fungsinya

4. Bidang Olahraga

Bidang Olahraga dipimpin oleh seorang Kepala Bidang yang berkedudukan di bawah dan bertanggung jawab kepada Kepala Dinas melalui Sekretaris.

Dalam melaksanakan tugas pokok, Kepala Bidang Olahraga mempunyai fungsi:

- a. Pelaksanaan pembinaan, pengembangan dan pemantauan keolahragaan dan bakat prestasi
- b. Pelaksanaan pembinaan, pengembangan dan pemantauan kapasitas lembaga keolahragaan daerah
- c. Pelaksanaan pembinaan, pengembangan dan menyusun perencanaan serta melaksanakan penyediaan prasarana keolahragaan

5. Bidang Pariwisata

Bidang Pariwisata dipimpin oleh seorang Kepala Bidang yang berkedudukan di bawah dan bertanggung jawab kepada Kepala Dinas melalui Sekretaris.

Dalam melaksanakan tugas pokok, Kepala Bidang Pariwisata mempunyai fungsi:

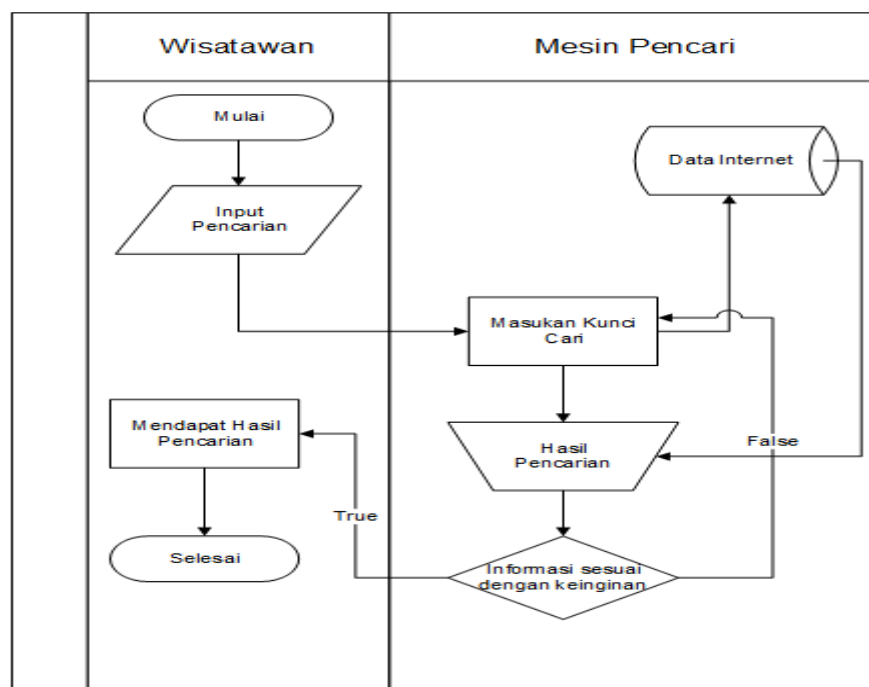
- a. Pelaksanaan pembinaan, pengembangan dan pemantauan sarana obyek wisata
- b. Pelaksanaan pengembangan dan pemantauan daya tarik wisata
- c. Pelaksanaan pembinaan, pengembangan dan pemantauan penyuluhan serta pemasaran wisata

3.2. Analisis Sistem

Analisis sistem merupakan tahap yang bertujuan untuk memahami sistem, mengetahui kekurangan sistem, dan menentukan kebutuhan dari sistem yang akan dibangun. Dengan menganalisis prosedur sistem maka setiap sistem yang akan dibangun dapat di evaluasi sehingga dapat dibuat satu usulan untuk pengembangan sistem atau pembangunan sistem yang baru.

3.2.1. Sistem yang Sedang Berjalan

Setelah melakukan peninjauan langsung diketahui bahwa Sistem yang sedang berjalan saat ini yaitu dengan pencarian di mesin pencari *online* dimana calon wisata mencari info wisata gunung di Pekalongan atau bertanya ke komunitas.



Gambar 3.2 *Flowchart* Sistem informasi wisata gunung yang sedang berjalan

Berikut uraian prosedur Sistem Informasi Geografis Wisata Gunung di Pekalongan pada gambar 3.2 :

1. Mulai
2. Wisatawan menginputkan pencarian wisata
3. Masukan kunci cari dari wisata yang ada di internet
4. Mendapatkan hasil pencarian
5. Di cek apakah hasilnya sesuai dengan apa yang kita inginkan
6. Jika salah (*False*) maka tidak mendapatkan hasil wisata dan kembali menginputkan kunci cari
7. Jika benar (*True*) maka mendapatkan hasil wisata
8. Selesai

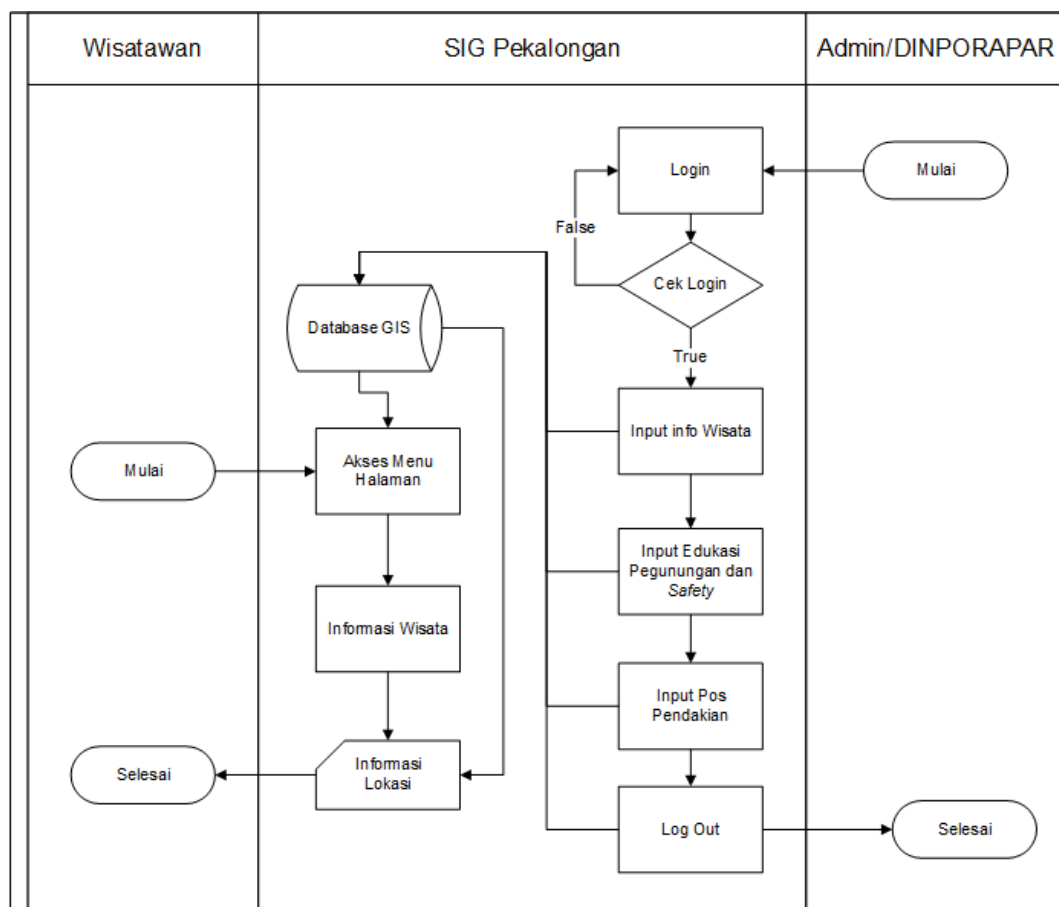
3.2.2. Analisi Sistem yang Sedang Berjalan

Setelah menganalisis prosedur jalannya sistem yang ada, dapat disimpulkan bahwa permasalahan yang ada pada sistem yang berjalan adalah pada proses pencarian informasi wisata, proses pencarian masih dilakukan secara manual yaitu dengan mencari di internet atau dengan bertanya pada komunitas pendaki gunung. Hal itu tentu membuat pencarian informasi menjadi tidak efektif dan efisien karena wisatawan harus mencari informasi dengan cara yang telah di sebutkan di atas. Oleh sebab itu penulis membuat pemecahan masalah berupa perancangan dan pembangunan SIG berbasis Android, sehingga diharapkan memiliki beberapa kelebihan baik dari segi kemudahan operasional maupun kecepatan akses.

3.3. Perancangan Sistem Baru

Perancangan sistem dilakukan untuk memberikan gambaran, perencanaan, dan pembuatan sketsa pengaliran suatu data dalam program. Sehingga memudahkan seseorang dalam pembuatan sistem agar sistem dapat dengan mudah dimengerti oleh orang yang menggunakan sistem tersebut.

Dalam mengantisipasi maupun mengatasi permasalahan yang ada pada SIG wisata Pekalongan, perlu adanya implementasi aplikasi sistem baru berupa aplikasi Android sebagai pengembangan dari sistem yang sudah ada. Dengan pengembangan sistem berupa aplikasi Android diharapkan wisatawan dapat mengakses informasi wisata gunung dengan lebih cepat dan efisien.



Gambar 3.3 Flowchart Sistem informasi geografis yang di usulkan

Berikut uraian prosedur Sistem Informasi Geografis Wisata Gunung di Pekalongan pada gambar 3.3 :

a. Proses dari admin

1. Mulai
2. Admin login dengan memasukan nama dan *password*
3. Jika salah (*False*) maka kembali ke login
4. Jika benar (*True*) maka bisa menginput data, yaitu data wisata, data pos pendakian, dan edukasi gunung dan *safety*
5. Semua data akan tersimpan ke database GIS
6. Selesai

b. Proses dari pengguna

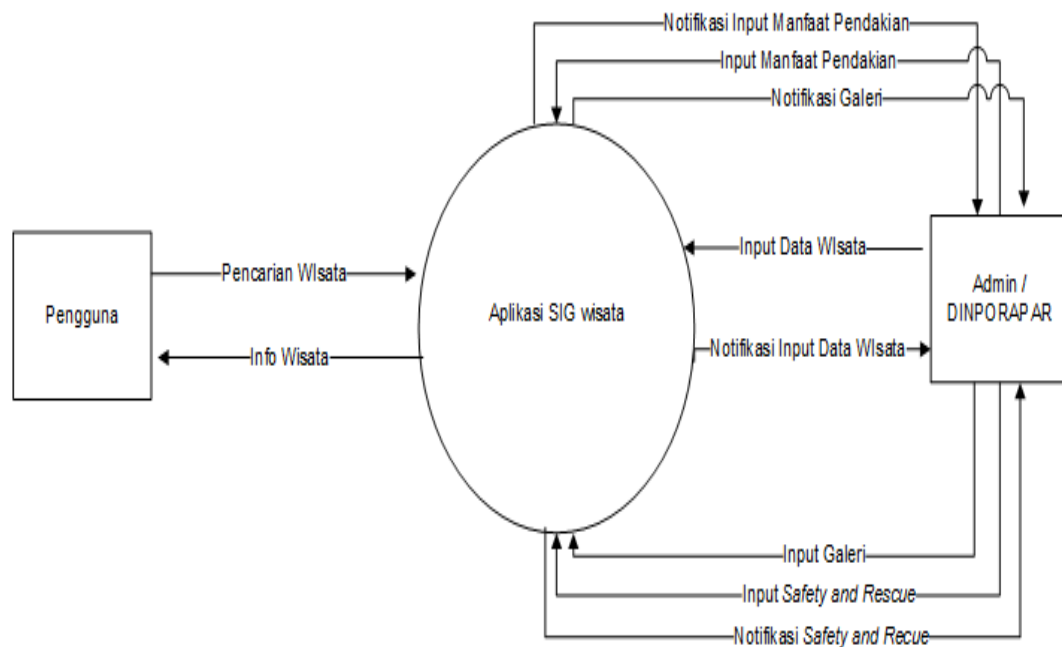
1. Mulai
2. Akses menu halaman
3. Informasi wisata
4. Melihat informasi lokasi dari wisata
5. Selesai

3.4. Perancangan Aliran Data

Sistem informasi yang diusulkan digunakan sepenuhnya untuk memudahkan sistem informasi geografis wisata gunung di Pekalongan. Dalam membuat dan merancang suatu aliran data informasi pariwisata diperlukan gambaran untuk proses-proses yang ada dari *input* proses sampai *output* prosesnya.

Untuk menggambarkan aliran data tersebut digunakan DFD atau *Data Flow Diagram*. DFD adalah suatu *network* yang menggambarkan suatu sistem automat/komputerisasi, manualisasi atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya (Sutabri, 2012).

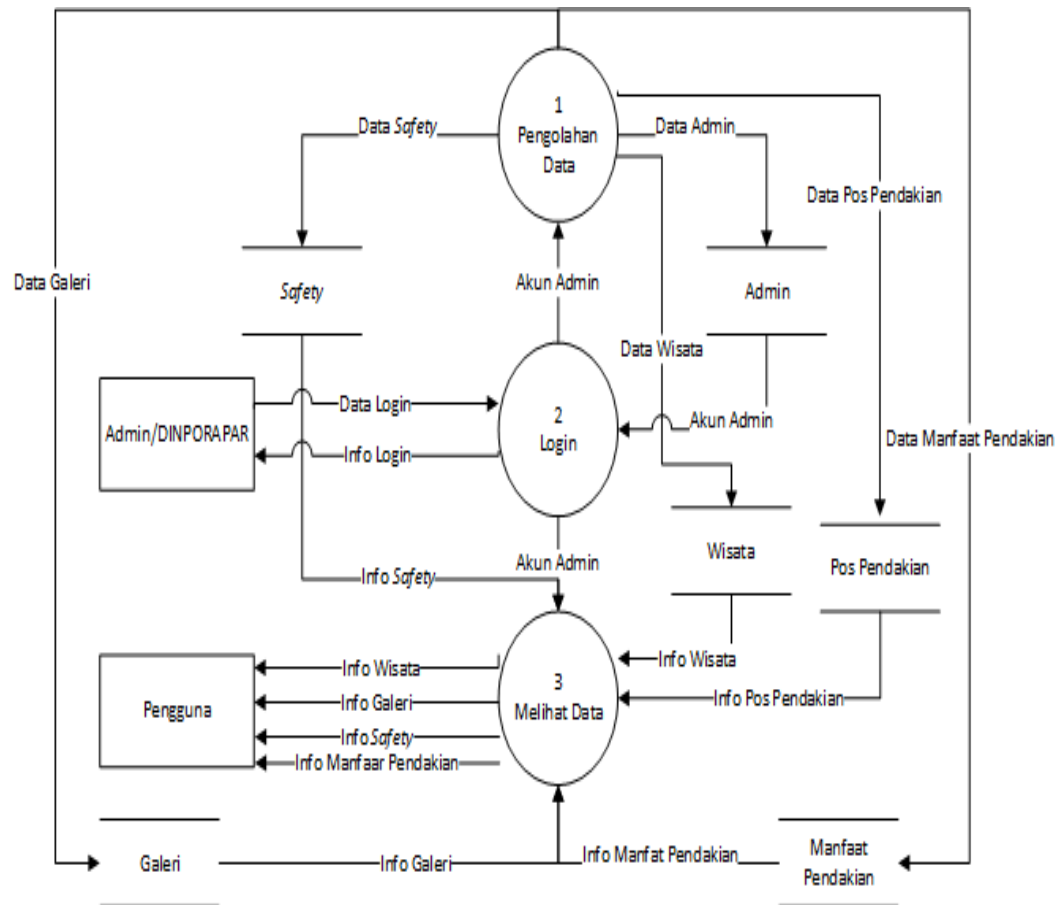
3.4.1. Diagram Konteks



Gambar 3.4 Diagram Konteks

Dari sistem yang diusulkan terdapat dua entitas yang saling berkaitan, yaitu entitas pengguna dan admin. Admin akan menginput data wisata ke sistem yang kemudian diproses menjadi informasi seperti informasi wisata.

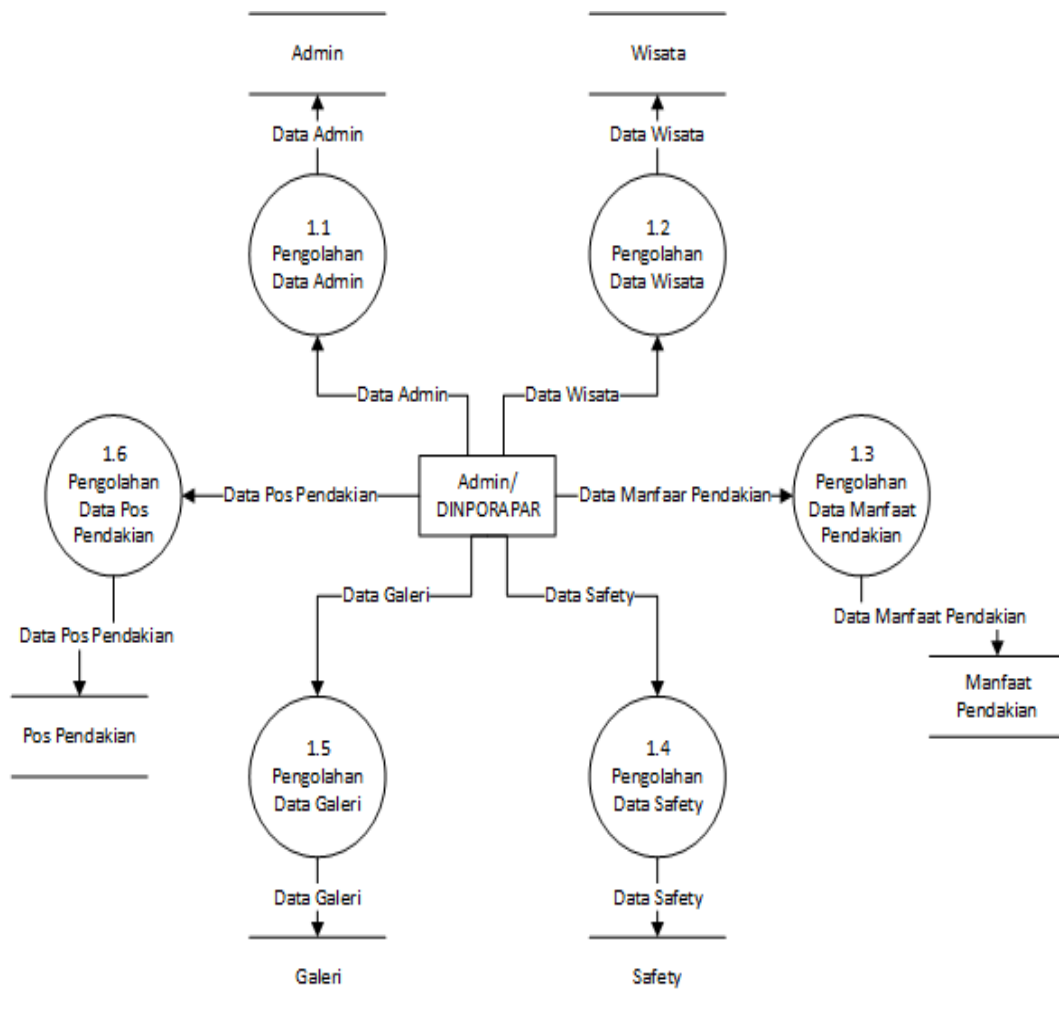
3.4.2. DFD Level 0



Gambar 3.5 DFD Level 0

Dari DFD Level 0 diatas terdapat 3 proses, 6 data store, dan 2 entitas yaitu Admin dan Pengguna. Hanya satu entitas yang harus melewati proses *Login* terlebih dahulu agar bisa melakukan proses yang lainnya.

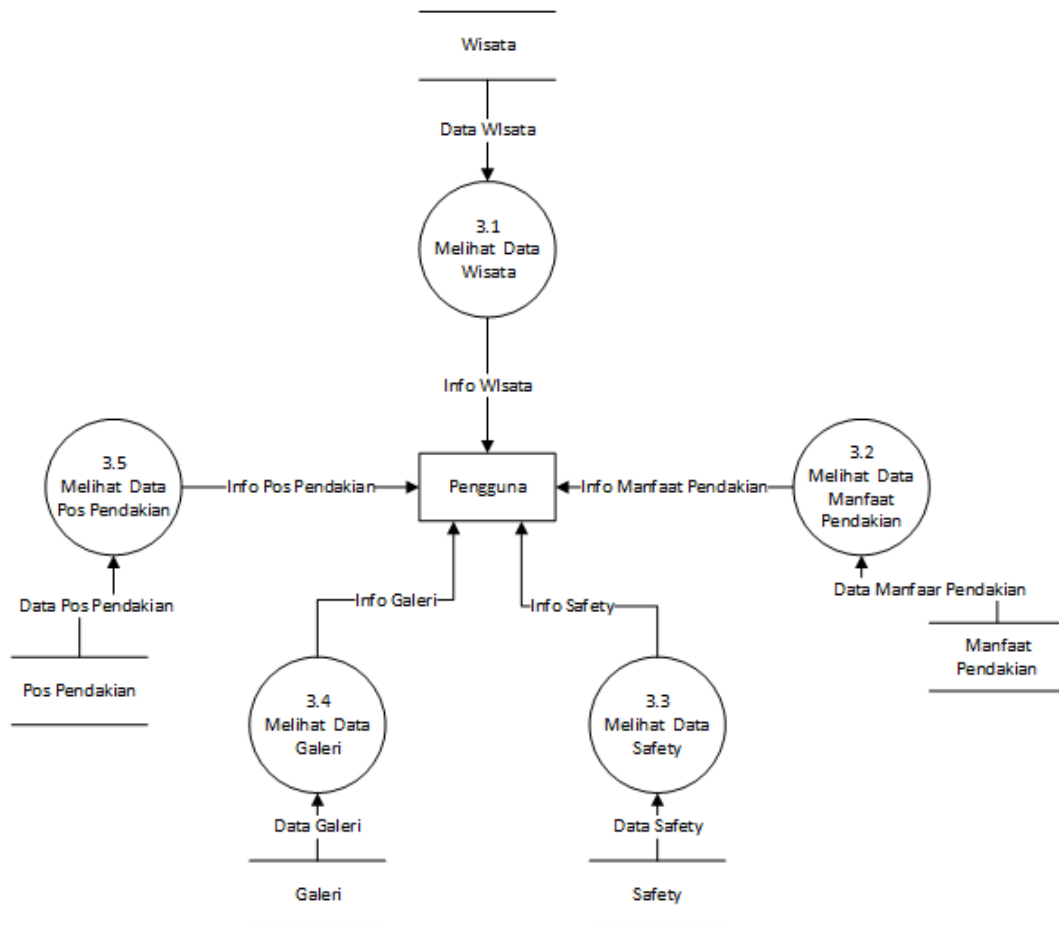
3.4.3. DFD *Level 1* Proses 1 (Pengolahan Data)



Gambar 3.6 DFD *Level 1* Proses 1

Dari DFD *Level 1* Proses 1 diatas terdapat 6 proses, 6 *data store*, dan 1 entitas yaitu Admin. Entitas Admin dapat melakukan semua proses.

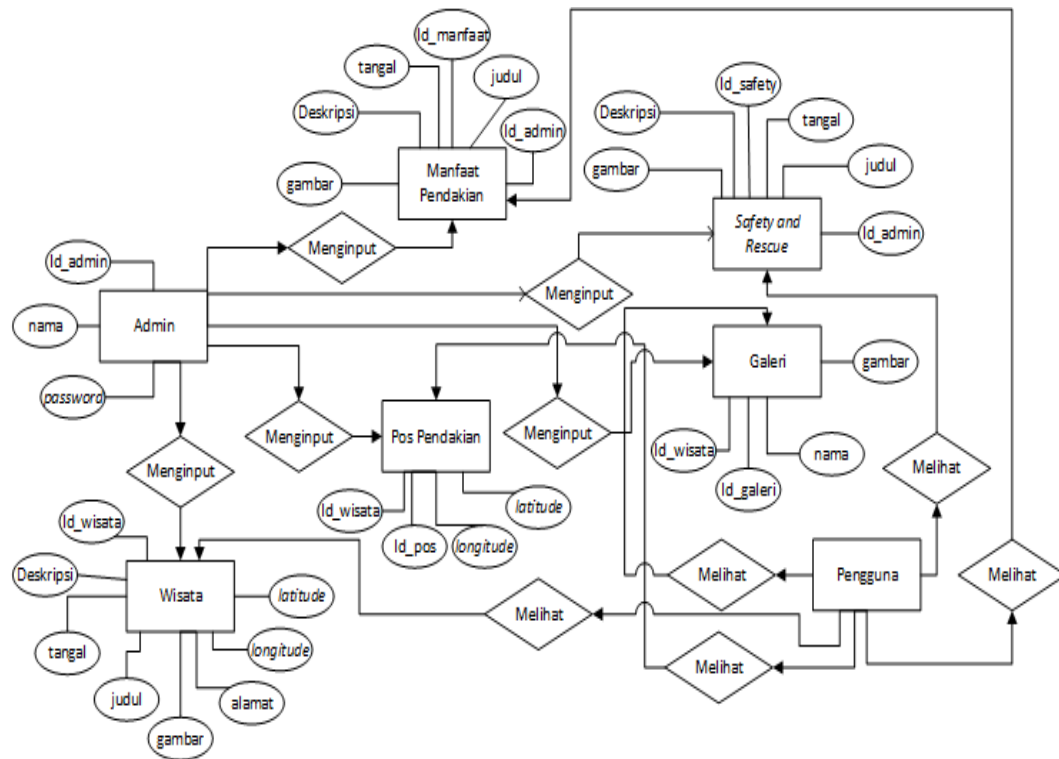
3.4.4. DFD *Level 1* Proses 3 (Melihat Data)



Gambar 3.7 DFD *Level 1* Proses 3

Dari DFD *Level 1* Proses 3 diatas terdapat 5 proses, 5 *data store*, dan 1 entitas yaitu Pengguna. Entitas Pengguna mendapat info dari semua proses dan *data source* yang ada.

3.4.5. Perancangan *Entity Relationship Diagram* (ERD)



Gambar 3.8 *Entity Relationship Diagram* (ERD)

Dari rancangan pada gambar 3.6 terdapat 7 entitas, dimana masing-masing entitas memiliki beberapa atribut. Entitas admin mempunyai 3 atribut, Pos Pendakian dengan 5 atribut, manfaat pendakian memiliki 6 atribut, wisata mempunyai 7 atribut, *safety and rescue* mempunyai 6 atribut, dan galeri memiliki 4 atribut.

3.5. Desain Database

Dalam perancangan dan pembangunan SIG ini, penulis menggunakan MySQL sebagai database. Adapun desain database yang digunakan pada SIG Wisata Gunung Pekalongan menggunakan database MySQL adalah sebagai berikut :

a. Tabel Admin

Tabel admin digunakan untuk menyimpan data admin, terdiri dari 3 *field* dimana id_admin sebagai *primary key*.

Tabel 3.1 Tabel Admin

Field	Type	Size	Key	Keterangan
id_admin	Integer	11	*	Identitas admin
nama	Varchar	30		<i>Username</i> admin
password	Varchar	30		<i>Password</i> admin

b. Tabel Wisata

Tabel wisata digunakan untuk menyimpan data wisata, terdiri dari 7 *field* dimana id_wisata sebagai *primary key*.

Tabel 3.2 Tabel Wisata

Field	Type	Size	Key	Keterangan
id_wisata	Integer	11	*	Identitas wisata
image	Text			Gambar Wisata
judul	Text			Judul Wisata
lokasi	Text			Lokasi Wisata
deskripsi	LongText			Deskripsi Wisata
ketinggian	Integer	11		Tinggi Gunung
fasilitas	Text			Fasilitas Wisata

tipe_tanah	Text			Jenis Tanah Wisata
tipe_gunung	Text			Tipe Gunung
latitude	Varchar	50		Latitude Wisata
longiude	Varchar	50		Longitude Wisata

c. Tabel Manfaat Pendakian

Tabel manfaat pendakian digunakan untuk menyimpan data manfaat pendakian, terdiri dari 6 *field* dimana *id_manfaat* sebagai *primary key*.

Tabel 3.3 Tabel Manfaat Pendakian

Field	Type	Size	Key	Keterangan
id_manfaat	Integer	11	*	Identitas Manfaat Pendakian
Id_admin	Integer	11	**	Identitas Admin
image	Text			Gambar Manfaat Pendakian
judul	Text			Judul Manfaat Pendakian
deskripsi	LongText			Deskripsi Manfaat Pendakian
tanggal	DATETIME			Tanggal Postingan

d. Tabel Galeri

Tabel galeri digunakan untuk menyimpan data manfaat pendakian, terdiri dari 4 *field* dimana *id_manfaat* sebagai *primary key*.

Tabel 3.4 Tabel Galeri

Field	Type	Size	Key	Keterangan
id_galeri	Integer	11	*	Identitas galeri
Id_wisata	Integer	11	**	Identitas Wisata
nama	Text			Nama Galeri
gambar	Text			Gambar Galeri

e. Tabel *Safety and Rescue*

Tabel *Safety and Rescue* digunakan untuk menyimpan data manfaat pendakian, terdiri dari 6 *field* dimana *id_safety* sebagai *primary key*.

Tabel 3.5 Tabel *Safety and Rescue*

Field	Type	Size	Key	Keterangan
id_safety	Integer	11	*	Identitas <i>Safety and Rescue</i>
Id_admin	Integer	11	**	Identitas Admin
image	Text			Gambar <i>Safety and Rescue</i>
judul	Text			Judul <i>Safety and Rescue</i>
deskripsi	LongText			Deskripsi <i>Safety and Rescue</i>
tanggal	DATETIME			Tanggal Postingan

f. Tabel Pos Pendakian

Tabel Pos Pendakian digunakan untuk menyimpan data manfaat pendakian, terdiri dari 5 *field* dimana *id_pos* sebagai *primary key*.

Tabel 3.6 Tabel Pos Pendakian

Field	Type	Size	Key	Keterangan
id_pos	Integer	11	*	Identitas Pos Pendakian
Id_wisata	Integer	11	**	Identitas Wisata
nama	Text			Nama Pos Pendakian
latitude	Varchar	50		Latitude Pos Pendakian
longiude	Varchar	50		Longitude Pos Pendakian

BAB IV

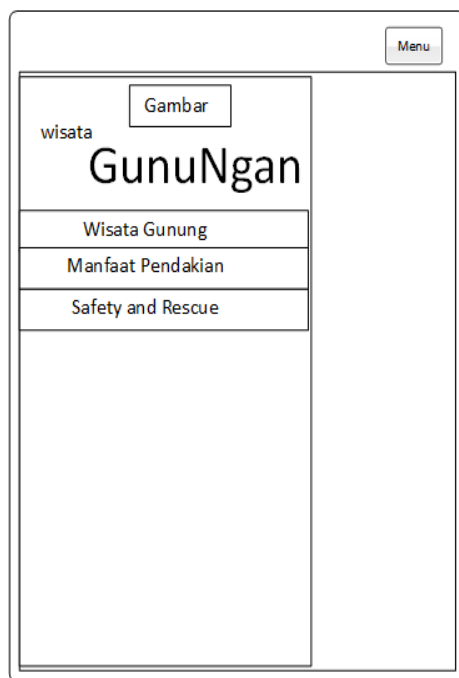
IMPLEMENTASI DAN PEMBAHASAN

4.1. Perancangan Desain Sistem

Perancangan desain berguna untuk menggambarkan halaman-halaman yang dapat digunakan untuk pengelolaan Sistem Informasi Geografis Wisata Gunung di Pekalongan. Perancangan desain sistem ini dibagi menjadi 2 (dua) bagian, yaitu tampilan halaman *user* dan admin. Berikut ini merupakan perancangan desain sistem yang akan dibuat.

4.1.1. Tampilan Halaman *User*

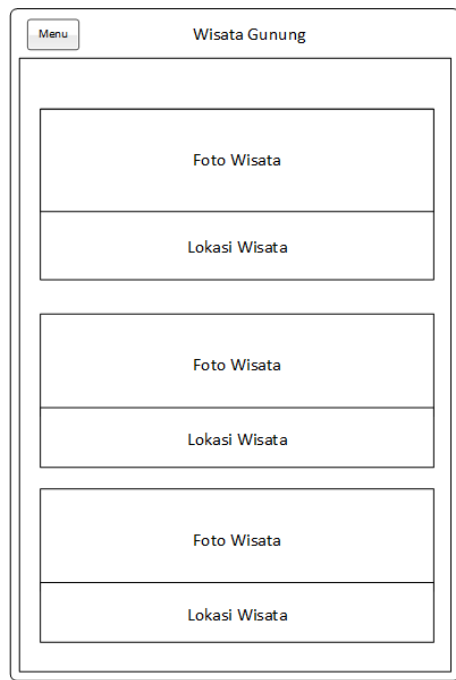
a. Halaman Menu *User*



Halaman ini merupakan tampilan menu utama dari aplikasi untuk *user* terdiri dari Wisata Gunung, Manfaat Pendakian dan *Safety and Rescue* yang digunakan user untuk mencari informasi yang diinginkan.

Gambar 4.1 Halaman Menu *User*

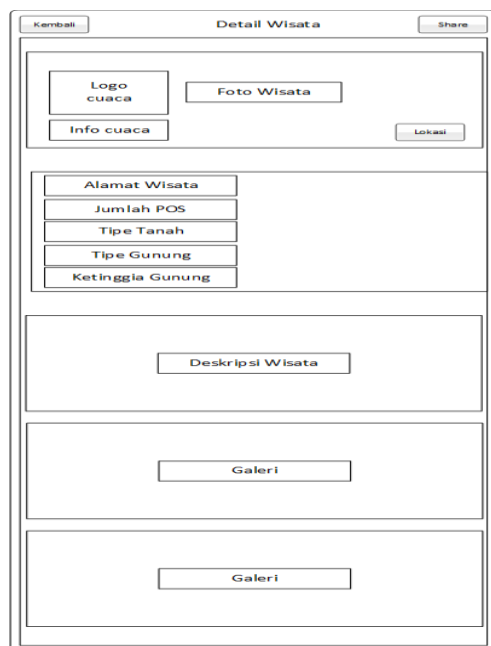
b. Halaman Wisata Gunung



Halaman ini di rancang untuk menampilkan data wisata, lokasi wisata dan fotonya.

Gambar 4.2 Halaman Wisata Gunung

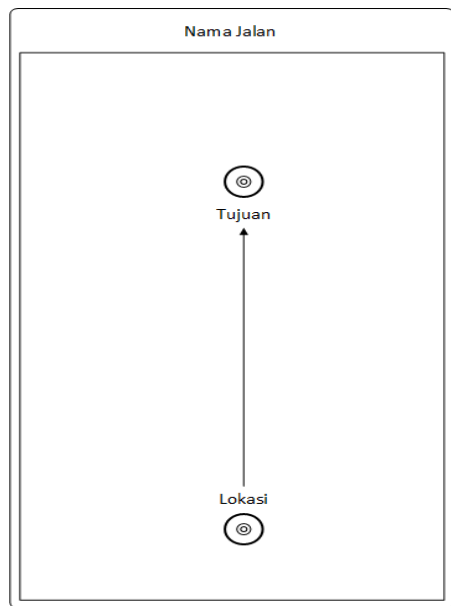
c. Halaman Detail Wisata



Halaman ini untuk menampilkan detail wisata, info prediksi cuaca, foto wisata, pos pendakian dan lokasi.

Gambar 4.3 Detail Wisata

d. Halaman Menuju Lokasi



Halaman ini untuk menampilkan Menuju Lokasi Tujuan.

Gambar 4.4 Halaman Menuju Lokasi

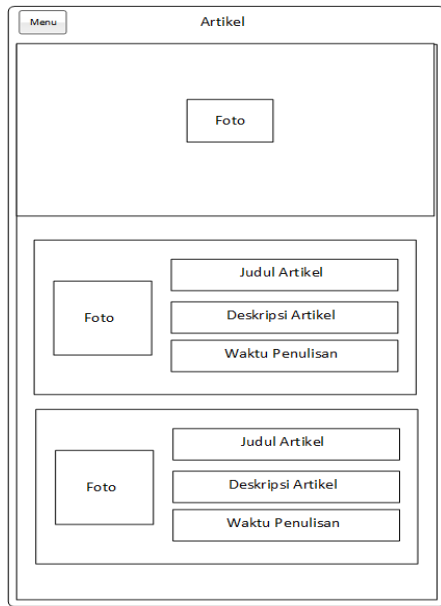
e. Halaman Galeri



Halaman ini untuk menampilkan foto dari *spot* wisata.

Gambar 4.5 Halaman Galeri

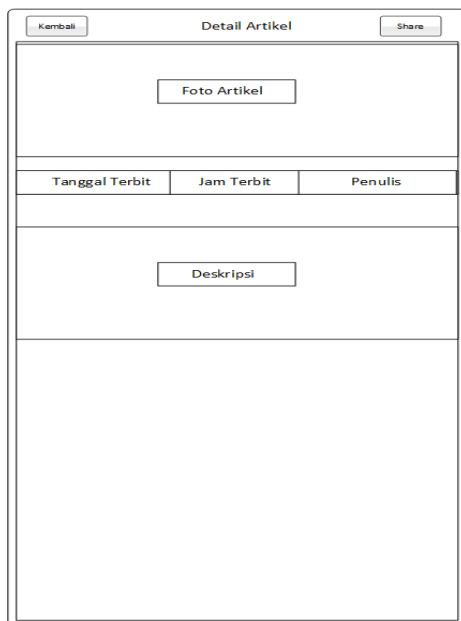
f. Halaman Artikel (Manfaat Pendakian dan *Safety and Resue*)



Halaman ini untuk menampilkan artikel (Manfaat Pendakian dan *Safety and Resue*) yang berisi informasi.

Gambar 4.6 Halaman Artikel

g. Halaman Detail Artikel (Manfaat Pendakian dan *Safety and Resue*)

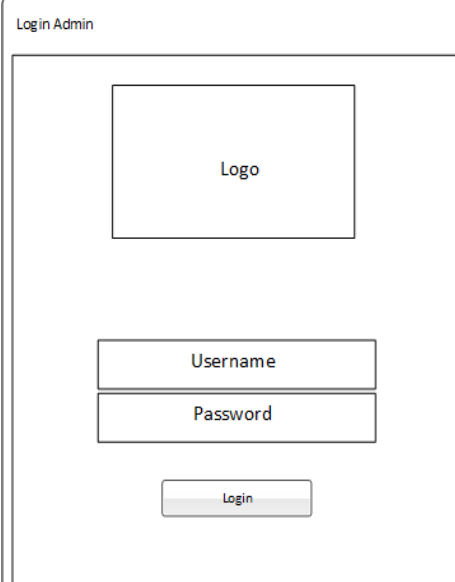


Halaman ini untuk menampilkan detail artikel (Manfaat Pendakian dan *Safety and Resue*) yang berisi informasi lengkap dari artikel tersebut.

Gambar 4.7 Halaman Detail Artikel

4.1.2. Tampilan Halaman Admin

a. Halaman Login

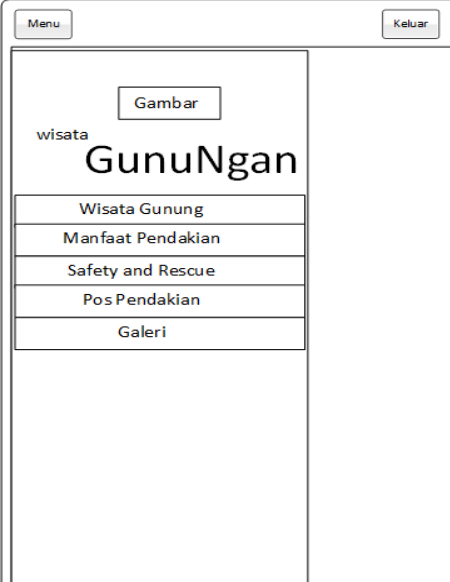


The image shows a login form titled "Login Admin". It contains a "Logo" placeholder, a "Username" input field, a "Password" input field, and a "Login" button.

Gambar 4.8 Halaman Login

Halaman ini akan muncul pertama kali saat *admin* membuka aplikasi.

b. Halaman Menu *Admin*



The image shows the "Menu" page for an admin. It has a "Menu" button in the top left and a "Keluar" button in the top right. The main content area displays a "Gambar" placeholder, the text "wisata", and a large heading "GunuNgan". Below this is a list of menu items: "Wisata Gunung", "Manfaat Pendakian", "Safety and Rescue", "Pos Pendakian", and "Galeri".

Gambar 4.9 Halaman Menu *admin*

Halaman ini merupakan tampilan menu utama dari aplikasi untuk *admin* terdiri dari Wisata Gunung, Manfaat Pendakian, Pos Pendakian, Galeri, *Safety and Rescue*.

c. Halaman Wisata

The screenshot shows a web application window titled 'Wisata Gunung'. It has a 'Menu' button on the top left and a 'Keluar' button on the top right. The main content area displays a list of three travel items. Each item is represented by a table with two rows: 'Foto Wisata' and 'Lokasi Wisata'. To the right of each table are two buttons: 'HAPUS' and 'UBAH'. At the bottom right of the list is a button labeled 'TAMBAH DATA'.

Halaman ini di rancang untuk menampilkan/mengubah data wisata, lokasi wisata dan fotonya.

Gambar 4.10 Halaman Data Wisata

d. Halaman Tambah Wisata

The screenshot shows a web application window titled 'Tambah Wisata'. It has a 'Kembali' button on the top left and a 'Simpan' button on the top right. The main content area contains a form with the following fields: 'Nama Wisata :', 'Alamat Wisata :', 'Deskripsi Wisata :', 'Latitude :', and 'Longitude :'. There is also an 'Upload Foto' button. The form is enclosed in a box with a shadow.

Halaman ini di gunakan untuk menambah data Wisata.

Gambar 4.11 Halaman Tambah Wisata

e. Halaman Artikel (Manfaat Pendakian dan *Safety and Resue*)

The screenshot shows a web interface titled 'Artikel'. It has a 'Menu' button on the top left and a 'Keluar' button on the top right. The main content area displays a list of three articles. Each article entry consists of a 'Foto' placeholder, a 'Judul Artikel' field, a 'Deskripsi Artikel' field, a 'Waktu Penulisan' field, and two buttons: 'HAPUS' and 'UBAH'. At the bottom of the list, there is a 'TAMBAH DATA' button.

Halaman ini di rancang untuk menampilkan/mengubah data dari masing-masing artikel.

Gambar 4.12 Halaman Artikel (Manfaat Pendakian dan *Safety and Resue*)

f. Halaman Tambah Artikel (Manfaat Pendakian dan *Safety and Resue*)

The screenshot shows a web interface titled 'Tambah Artikel'. It has a 'Kembali' button on the top left and a 'Simpan' button on the top right. The main content area contains three input fields: 'Judul Artikel', 'Upload Foto', and 'Deskripsi'.

Halaman ini di gunakan untuk menambah data Artikel (Manfaat Pendakian dan *Safety and Resue*).

Gambar 4.13 Halaman Tambah Artikel

g. Halaman Pos pendakian

The screenshot shows a mobile application interface for 'Pos Pendakian'. At the top, there's a header with 'Menu' on the left and 'Keluar' on the right. Below the header, the page is divided into two main sections. The first section contains a text input field labeled 'Nama Wisata' with a small square icon to its right. The second section contains a text input field labeled 'Nama POS', followed by two text input fields labeled 'Latitude' and 'Longitude'. At the bottom right of the page, there is a button labeled 'TAMBAH DATA'.

Gambar 4.14 Halaman Pos Pendakian

h. Halaman Tambah Pos Pendakian

The screenshot shows a mobile application interface for 'Tambah POS'. At the top, there's a header with 'Kembali' on the left and 'Simpan' on the right. Below the header, the page contains three text input fields. The first is labeled 'Nama POS :', the second is labeled 'Latitude :', and the third is labeled 'Longitude :'. Below these fields is a large, empty rectangular area, likely for additional notes or a description.

Gambar 4.15 Tambah Pos Pendakian

Halaman ini di rancang untuk menampilkan/mengubah data pos wisata.

Halaman ini di gunakan untuk menambah data Pos Pendakian dari masing-masing wisata.

i. Halaman Galeri

The 'Galeri' page interface includes a top navigation bar with 'Menu' and 'Keluar' buttons. Below this is a form for 'Nama Wisata' with a gallery icon. The main area displays a list of three items, each consisting of a 'Foto' placeholder and 'HAPUS' and 'UBAH' action buttons.

Gambar 4.16 Halaman Galeri

j. Halaman Tambah Galeri

The 'Tambah Galeri' page interface includes a top navigation bar with 'Kembali' and 'Simpan' buttons. Below this is a form with a 'Nama Foto :' input field and an 'Upload Foto' button.

Gambar 4.17 Tambah Galeri

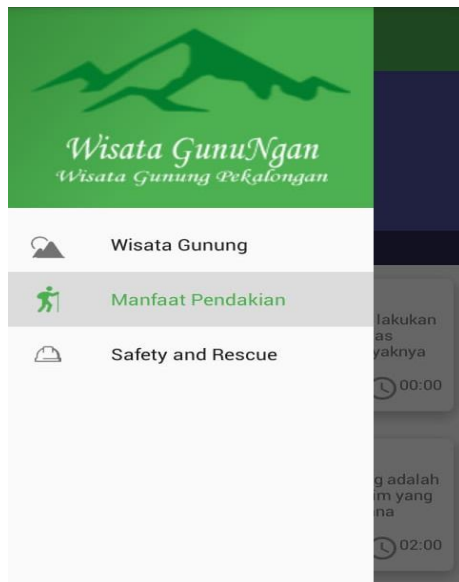
Halaman ini di rancang untuk menampilkan/mengubah data foto wisata.

Halaman ini di gunakan untuk menambah data Foto Wisata dari masing-masing wisata.

4.2. Hasil Tampilan Sistem

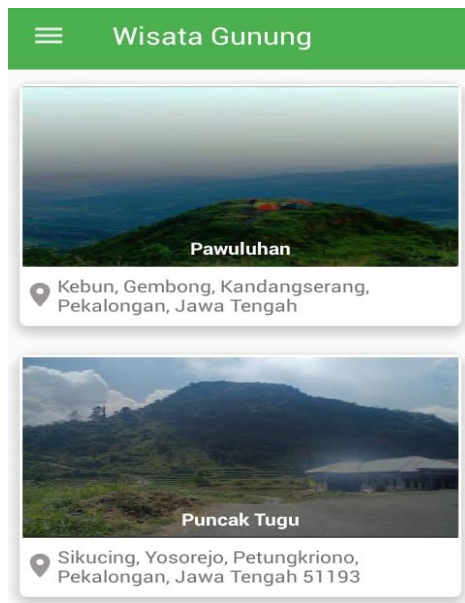
4.2.1. Tampilan Halaman *User*

a. Halaman Menu *User*



Gambar 4.18 Halaman Menu *User*

b. Halaman Wisata Gunung



Gambar 4.19 Halaman Wisata Gunung

Halaman ini merupakan tampilan menu utama dari aplikasi untuk *user* terdiri dari Wisata Gunung, Manfaat Pendakian dan *Safety and Rescue* yang digunakan user untuk mencari

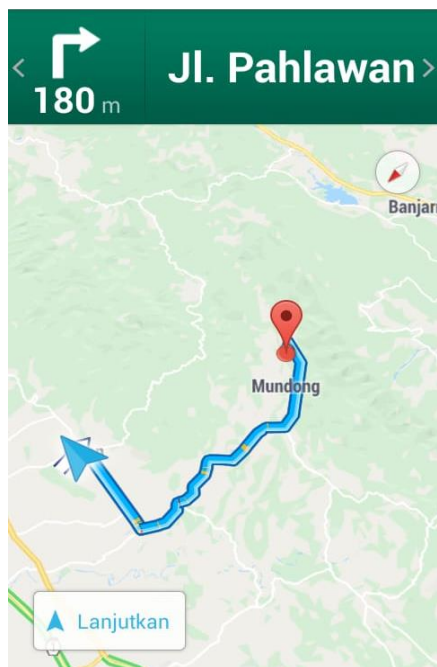
Halaman ini di rancang untuk menampilkan data wisata, lokasi wisata dan fotonya.

c. Halaman Detail Wisata



Gambar 4.20 Halaman Detail Wisata

d. Halaman Menuju Lokasi

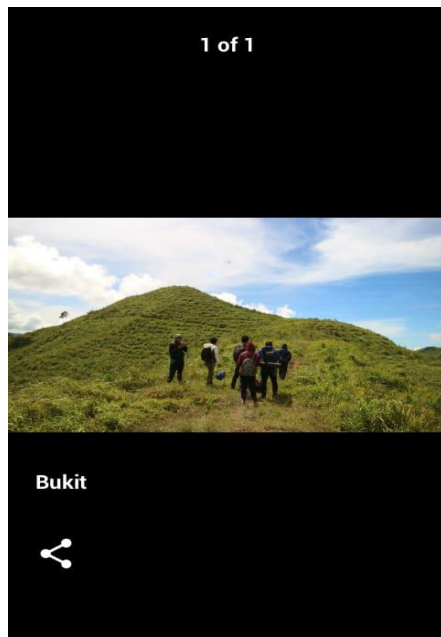


Gambar 4.21 Halaman Detail Wisata

Halaman ini untuk menampilkan detail wisata, info prediksi cuaca, foto wisata, pos pendakian dan lokasi.

Halaman ini untuk menampilkan Menuju Lokasi Tujuan.

e. Halaman Galeri



Gambar 4.22 Halaman Galeri

Halaman ini untuk menampilkan foto dari *spot* wisata.

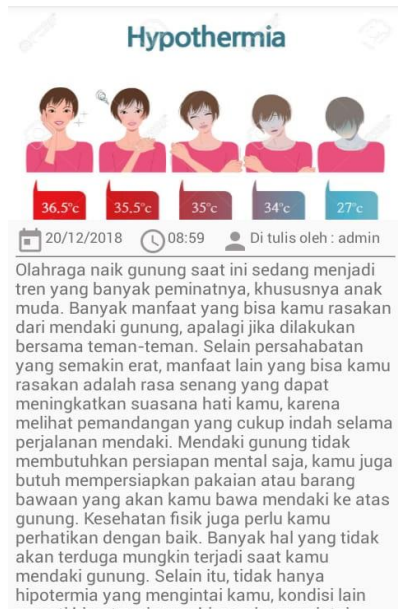
f. Halaman Artikel (Manfaat Pendakian dan *Safety and Resue*)



Gambar 4.23 Halaman Artikel

Halaman ini untuk menampilkan artikel (Manfaat Pendakian dan *Safety and Resue*) yang berisi informasi.

g. Halaman Detail Artikel (Manfaat Pendakian dan *Safety and Resue*)

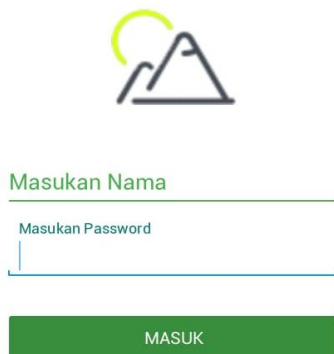


Halaman ini untuk menampilkan detail artikel (Manfaat Pendakian dan *Safety and Resue*) yang berisi informasi lengkap dari artikel tersebut.

Gambar 4.24 Halaman Detail Artikel

4.2.2. Tampilan Halaman *Admin*

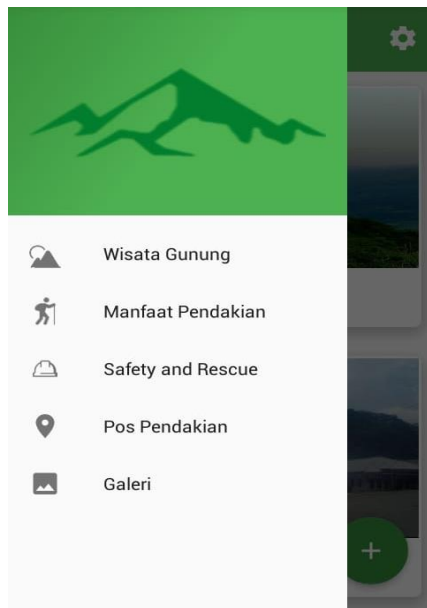
h. Halaman Login



Halaman ini akan muncul pertama kali saat *admin* membuka aplikasi.

Gambar 4.25 Halaman Login

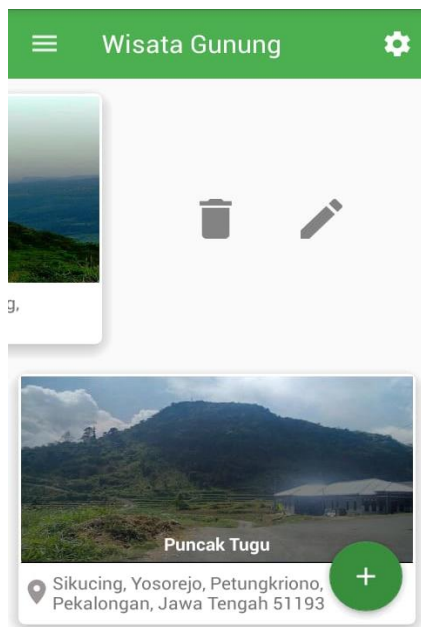
i. Halaman Menu *Admin*



Halaman ini merupakan tampilan menu utama dari aplikasi untuk *admin* terdiri dari Wisata Gunung, Manfaat Pendakian, Pos Pendakian, Galeri, *Safety and Rescue*.

Gambar 4.26 Halaman Menu *Admin*

j. Halaman Wisata



Halaman ini di rancang untuk menampilkan/mengubah data wisata, lokasi wisata dan fotonya.

Gambar 4.27 Halaman Wisata

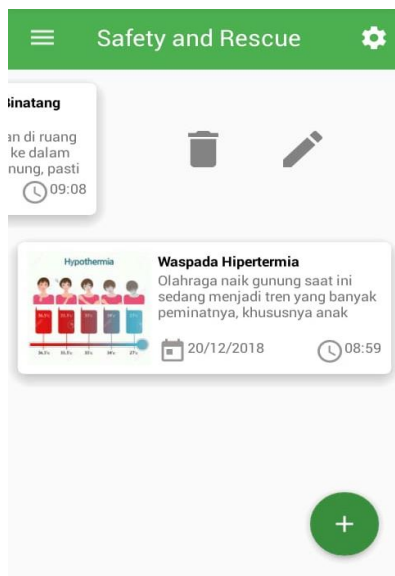
k. Halaman Tambah Wisata



Halaman ini di gunakan untuk menambah data Wisata.

Gambar 4.28 Halaman Tambah Wisata

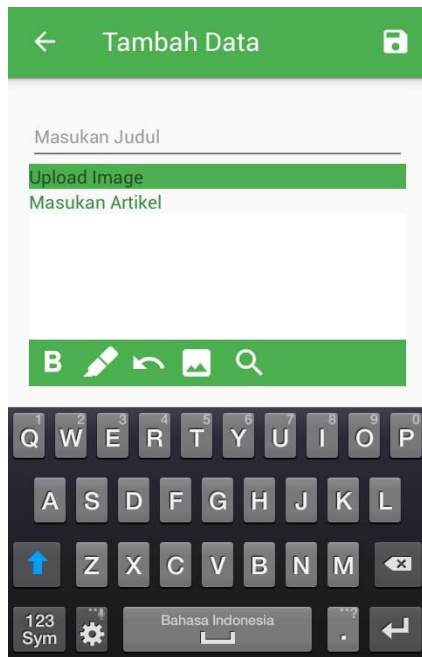
l. Halaman Artikel (Manfaat Pendakian dan *Safety and Resue*)



Halaman ini di rancang untuk menampilkan/mengubah data dari masing-masing artikel.

Gambar 4.29 Halaman Artikel

m. Halaman Tambah Artikel (Manfaat Pendakian dan *Safety and Resue*)



Halaman ini di gunakan untuk menambah data Artikel (Manfaat Pendakian dan *Safety and Resue*).

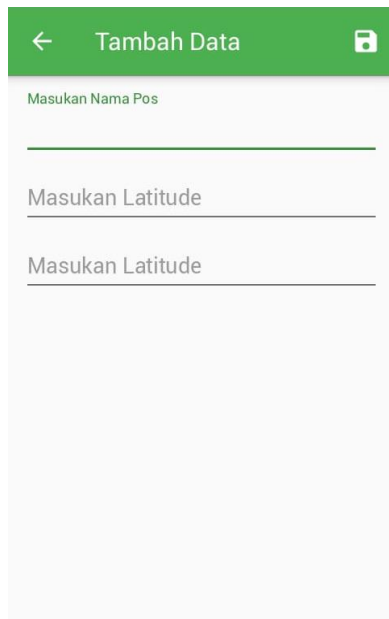
Gambar 4.30 Halaman Tambah
n. Halaman Pos Pendakian



Halaman ini di rancang untuk menampilkan/mengubah data pos wisata.

Gambar 4.31 Halaman Pos Pendakian

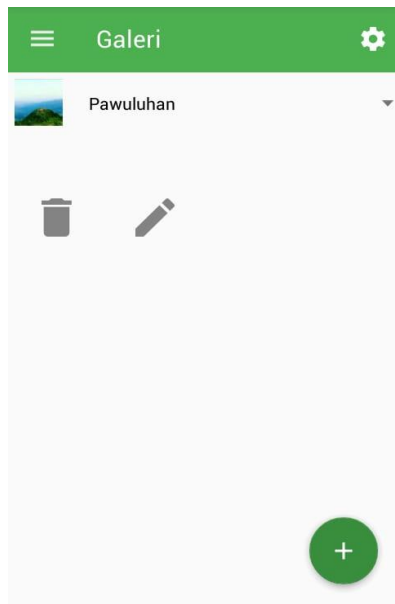
o. Halaman Tambah Pos Pendakian



Gambar 4.32 Tambah Pos Pendakian

Halaman ini di gunakan untuk menambah data Pos Pendakian dari masing-masing wisata.

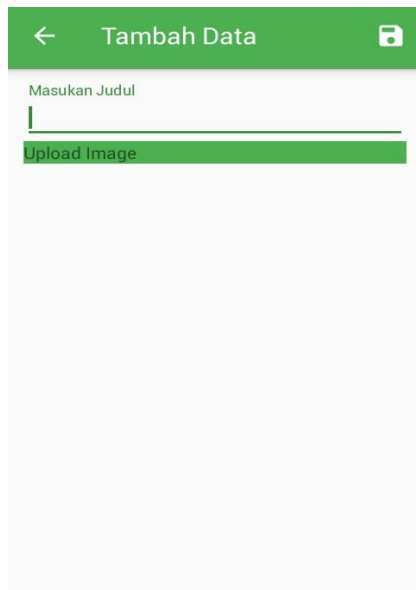
p. Halaman Galeri



Gambar 4.33 Halaman Galeri

Halaman ini di rancang untuk menampilkan/mengubah data foto wisata.

q. Halaman Tambah Galeri



Halaman ini di gunakan untuk menambah data Foto Wisata dari masing-masing wisata.

Gambar 4.34 Halaman Tambah Galeri

BAB V

PENUTUP

5.1. Kesimpulan

Pada bab ini akan dijelaskan kesimpulan dari pembuatan laporan tugas akhir yang berisi uraian singkat terhadap implementasi dan saran untuk pengembangan Sistem Informasi Geografis Wisata Gunung Di Pekalongan Berbasis Android.

Dari penelitian yang saya dapatkan di wisata gunung dapat saya simpulkan sebagai berikut :

1. SIG Wisata Gunung di Pekalongan berbasis Android ini telah berhasil dibuat dengan menggunakan *Android Studio* sebagai *front-end*, PHP sebagai *back-end*, dan MySQL sebagai *database*.
2. Aplikasi Sistem Informasi Geografis Wisata Gunung di Pekalongan Berbasis Android ini berisi informasi tentang wisata gunung, prediksi cuaca, artikel terkait manfaat pendakian, *safety and rescue*.
3. Aplikasi yang telah dibuat berisi menu yang menginformasikan data wisata gunung dan informasi terkait, yaitu menu wisata gunung, manfaat pendakian, *safety and rescue* sebagai *front end*.

5.2. Saran

Berkaitan dengan terselesaikannya penulisan tugas akhir ini, ada beberapa saran yang disampaikan, Sistem Informasi Geografis Wisata Gunung Di Pekalongan Berbasis Android perlu dikembangkan lagi agar dapat lebih maksimal dalam kegunaannya. Untuk kedepannya di sarankan agar

1. Pada program aplikasi ini belum terdapat *room chat* yang bertujuan untuk *sharing* terkait pengalaman para pendaki
2. Sistem masih bersifat pengelolaan admin belum adanya input informasi dari para pendaki terkait artikel *safety and rescue* yang bertujuan untuk memperluas informasi.

DAFTAR PUSTAKA

Abdi, Y. A., 2013. <https://www.academia.edu/7490026/>
[Diakses 25 Desember 2018].

Abdul Kadir dan Terra Ch. Triwahyuni, 2003. *Pengenalan Teknologi Informasi*. Yogyakarta: ANDI.

Amsyah, Z., 2001. *Manajemen Sistem Informasi*. Jakarta: Gramedia Pustaka.

Apriadi, T., 2012. In: *Agenda Setting*. s.l.:s.n.

Developers, G., 2018. *Mengenal Android Studio*.
<https://developer.android.com/studio/intro/>
[Diakses 1 Agustus 2018].

Ichwan, M., 2011. *Pemrograman Basis Data Delphi 7 dan Mysql*. Bandung: Informatika Bandung.

Kadir, A., 2003. *Pengenalan Sistem Informasi*. Yogyakarta: ANDI.

Kadir, A., 2014. *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: ANDI.

Mua, F., 2017. *Sistem Informasi Geografis Track Gunung Awu Berbasis Web*.

Nugroho, A., 2011. *Perancangan dan Implementasi Sistem Basis Data*. Yogyakarta: ANDI.

Nugroho, B., 2005. *atabase Relasional dengan MySQL*. Yogyakarta: ANDI.

Oetomo, B. S. D., 2002. *rencanaan & Pembangunan Sistem*. Yogyakarta: ANDI.

Pekalongan, P. K., 2001. *Kondisi Umum Geografis*.

Prahasta, E., 2002. *Konsep-konsep Dasar Sistem Informasi Geografis*. Bandung: Informatika Bandung.

Saputra, A. E., 2012. Sistem Informasi Geografis Objek Wisata Wilayah Puncak Kabupaten Bogor Berbasis Android.

Sunarfrihantono, B., 2003. *PHP dan MySQL untuk Web*. Yogyakarta: ANDI.

Susanto, S. H., 2011. *udah Membuat Aplikasi Android*. Yogyakarta: ANDI.

Sutabri, T., 2012. *Analisis Sistem Informasi*. Yogyakarta: ANDI.

Sutabri, T., 2017. *Sistem Informasi Geografis berbasis Android*. 1 ed. Yogyakarta: Andi Publihser.

Sutanta, E., 2011. *Basis Data dalam Tinjauan Konseptual*. Yogyakarta: ANDI.

Tutorialspoint, 2018. *Java*.
https://www.tutorialspoint.com/java/java_overview.htm
[Diakses 1 Agustus 2018].

Yusro, 2013. *Pengertian Google Maps API*.
<http://www.myusro.info/2013/02/pengertian-google-maps-api.html>
[Diakses 1 Agustus 2018].

LAMPIRAN 1

PERNYATAAN KEASLIAN KARYA TULIS TUGAS AKHIR



SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI PEKALONGAN BERBASIS ANDROID

PERNYATAAN KEASLIAN KARYA TULIS TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Saifudin
NIM : MI.15.00305
Program Studi : Manajemen Informatika

Menyatakan dengan sesungguhnya bahwa tugas akhir,

Judul : SISTEM INFORMASI GEOGRAFIS
WISATA GUNUNG DI PEKALONGAN
BERBASIS ANDROID

Adalah benar-benar hasil karya saya.

Didalam tugas akhir ini tidak terdapat gagasan orang lain yang saya ambil dengan cara menyalin atau meniru dalam bentuk rangkaian kalimat atau simbol yang saya aku seolah-olah sebagai tulisan saya sendiri tanpa melakukan pengakuan kepada penulis aslinya.

Apabila kemudian terbukti bahwa saya melakukan tindakan menyalin atau meniru tulisan orang lain seolah-olah pemikiran saya sendiri, saya bersedia menerima sanksi sesuai peraturan yang berlaku di Politeknik Muhammadiyah Pekalongan.

Pekalongan, 29 Januari 2019
Yang Memberikan Pernyataan

Saifudin
NIM : MI.15.00305

LAMPIRAN 2

BUKU BIMBINGAN TUGAS AKHIR



**SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI
PEKALONGAN BERBASIS ANDROID**

LAMPIRAN 3

LISTING PROGRAM



**SISTEM INFORMASI GEOGRAFIS WISATA GUNUNG DI
PEKALONGAN BERBASIS ANDROID**

Halaman *User*

a. **activity_main.xml**

```
<android.support.v4.widget.DrawerLayout
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:id="@+id/drawer_layout"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:fitsSystemWindows="true"

    tools:openDrawer="start">

    <include

        layout="@layout/app_bar_main"

        android:layout_width="match_parent"

        android:layout_height="match_parent" />

    <android.support.design.widget.NavigationView

        android:id="@+id/nav_view"

        android:layout_width="match_parent"

        android:layout_height="match_parent"

        android:layout_gravity="start"

        android:fitsSystemWindows="true"

        app:headerLayout="@layout/nav_header_main"

        app:menu="@menu/activity_main_drawer" />

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


b. app_bar_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
    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=".activity.MainActivity">

    <android.support.design.widget.AppBarLayout

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar

            android:id="@+id/toolbar"

            android:layout_width="match_parent"

            android:layout_height="?attr/actionBarSize"

            android:background="?attr/colorPrimary"

            app:popupTheme="@style/AppTheme.PopupOverlay" />

        </android.support.design.widget.AppBarLayout>

        <include layout="@layout/content_main" />

    </android.support.design.widget.CoordinatorLayout>
```

c. content_main.xml

```
<?xml version="1.0" encoding="utf-8"?>

<android.support.constraint.ConstraintLayout
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"

    app:layout_behavior="@string/appbar_scrolling_view_behavior"

    tools:context=".activity.MainActivity"

    tools:showIn="@layout/app_bar_main">

    <FrameLayout

        android:id="@+id/frame"

        android:layout_width="match_parent"

        android:layout_height="match_parent"/>

</android.support.constraint.ConstraintLayout>
```

d. MainActivity.java

```
public class MainActivity extends AppCompatActivity
    implements NavigationView.OnNavigationItemSelectedListener {
    FragmentManager fragmentManager = getSupportFragmentManager();

    @Override

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
        ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
            this, drawer, toolbar, R.string.navigation_drawer_open,
            R.string.navigation_drawer_close);
        drawer.addDrawerListener(toggle);
        toggle.syncState();

        NavigationView navigationView = (NavigationView) findViewById(R.id.nav_view);
        navigationView.setNavigationItemSelectedListener(this);
        fragmentManager.beginTransaction().replace(R.id.frame, new WisataGunung()).commit();
        getSupportActionBar().setTitle(getResources().getString(R.string.wisata_gunung));
    }

    @Override

    public void onBackPressed() {
        DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
        if (drawer.isDrawerOpen(GravityCompat.START)) {
            drawer.closeDrawer(GravityCompat.START);
        } else {
            super.onBackPressed();
        }
    }
}
```

```

    }
}

@SuppressWarnings("StatementWithEmptyBody")
@Override
public boolean onNavigationItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.

    int id = item.getItemId();

    if (id == R.id.nav_camera) {
        fragmentManager.beginTransaction().replace(R.id.frame, new WisataGunung()).commit();
        getSupportActionBar().setTitle(getResources().getString(R.string.wisata_gunung));
    } else if (id == R.id.nav_gallery) {
        fragmentManager.beginTransaction().replace(R.id.frame, new
        ManfaatPegunungan()).commit();
        getSupportActionBar().setTitle(getResources().getString(R.string.manfaat));
    } else if (id == R.id.nav_slideshow) {
        fragmentManager.beginTransaction().replace(R.id.frame, new
        SafetyAndRescue()).commit();
        getSupportActionBar().setTitle(getResources().getString(R.string.safety));
    }

    DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
}
}

```

e. fragment_list.xml

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```

```
android:layout_height="match_parent"
```

```
tools:context=".fragment.WisataGunung">
```

```
<!-- TODO: Update blank fragment layout -->
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical">
```

```
<com.glide.slider.library.SliderLayout
```

```
    android:id="@+id/slider"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="170dp"
```

```
    android:visibility="gone" />
```

```
<FrameLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent">
```

```
<TextView
```

```
    android:id="@+id/txt_notif"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:gravity="center"
```

```
    android:textAlignment="center"
```

```
    android:textSize="20sp"
```

```
    android:textStyle="bold"
```

```
    tools:text="Text Kosong" />
```

```

        <android.support.v7.widget.RecyclerView
            android:id="@+id/rv_list"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            tools:listitem="@layout/list_wisata" />
    </FrameLayout>
</LinearLayout>

<ProgressBar
    android:id="@+id/pd"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center" />
</FrameLayout>

```

f. **WisataGunung.java**

```

public class WisataGunung extends Fragment {
    @BindView(R.id.rv_list)
    RecyclerView rvList;
    Unbinder unbinder;

```

```
@BindView(R.id.txt_notif)
```

```
TextView txtNotif;
```

```
@BindView(R.id.pd)
```

```
ProgressBar pd;
```

```
public WisataGunung() {
```

```
    // Required empty public constructor
```

```
}
```

```
@Override
```

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                          Bundle savedInstanceState) {
```

```
    // Inflate the layout for this fragment
```

```
    View view = inflater.inflate(R.layout.fragment_list, container, false);
```

```
    unbinder = ButterKnife.bind(this, view);
```

```
    LinearLayoutManager layoutManager = new LinearLayoutManager(getContext());
```

```
    //layoutManager.setStackFromEnd(true);
```

```
    rvList.setLayoutManager(new VegaLayoutManager());
```

```
    getData();
```

```
    return view;
```

```
}
```

```
@Override
```

```
public void onDestroyView() {
```

```
    super.onDestroyView();
```

```
    unbinder.unbind();
```

```
}
```

```
private void getData() {
```

```
    pd.setVisibility(View.VISIBLE);
```

```
    ApiService apiService = RetroClient.getApiService();
```

```

Call<ResponseWisata> call = apiService.getWisata();

call.enqueue(new Callback<ResponseWisata>() {

    @Override

    public void onResponse(Call<ResponseWisata> call, Response<ResponseWisata>
response) {

        List<WisataItem> hasilPesan = response.body().getWisata();

        Log.e("Tag", "Hasil List :" + new Gson().toJson(hasilPesan));

        if (response.body().getResponse().equalsIgnoreCase("true")) {

            try {

                WisataAdapter adapterPesan = new WisataAdapter(getContext(), hasilPesan);

                // swipeMain.setRefreshing(false);

                rvList.setAdapter(adapterPesan);

                pd.setVisibility(View.GONE);

            } catch (Exception e){

            }

        } else {

            Log.e("Tag", "Gagal req data ");

            pd.setVisibility(View.GONE);

        }

    }

}

@Override

public void onFailure(Call<ResponseWisata> call, Throwable t) {

    pd.setVisibility(View.GONE);

    Snackbar.make(rvList, "Tidak Ada Koneksi internet",
Snackbar.LENGTH_INDEFINITE)

        .setAction("Ulangi", new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                getData();

            }

        })

```



```

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

```

g. activity_detail_wisata.xml

```

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
    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"
    android:fitsSystemWindows="true"
    tools:context=".activity.detail.DetailWisata">

    <android.support.design.widget.AppBarLayout
        android:id="@+id/app_bar"
        android:layout_width="match_parent"
        android:layout_height="@dimen/app_bar_height"
        android:fitsSystemWindows="true"
        android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.design.widget.CollapsingToolbarLayout
            android:id="@+id/toolbar_layout"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:fitsSystemWindows="true"
            app:contentScrim="?attr/colorPrimary"
            app:layout_scrollFlags="scroll|exitUntilCollapsed"
            app:toolbarId="@+id/toolbar">

```

```
<android.support.constraint.ConstraintLayout
    android:layout_width="match_parent"
    android:background="@drawable/bg_artikel"
    android:layout_height="wrap_content">
```

```
<ImageView
    android:id="@+id/img_cover"
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:background="#000000"
    android:scaleType="fitXY"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    android:src="@drawable/safety" />
```

```
<TextView
    android:id="@+id/bg"
    android:layout_width="0dp"
    android:layout_height="200dp"
    android:background="@drawable/bg_artikel"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="@+id/img_cover"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<TextView
    android:id="@+id/txt_cuaca"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:layout_marginLeft="8dp"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textSize="60sp"
    app:layout_constraintStart_toStartOf="@+id/guideline"
    app:layout_constraintTop_toTopOf="@+id/guideline3"
    tools:background="@drawable/ic_clear" />
```

```
<TextView
    android:id="@+id/txt_ket_cuaca"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="@android:color/white"
    android:textStyle="bold"
    android:textSize="18sp"
    app:layout_constraintStart_toStartOf="@+id/txt_cuaca"
    app:layout_constraintTop_toBottomOf="@+id/txt_cuaca"
    tools:text="Hujan deras" />
```

```
<android.support.constraint.Guideline
    android:id="@+id/guideline3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_begin="48dp" />
```

```
<TextView
    android:id="@+id/txt_derajat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginStart="8dp"
    android:padding="0dp"
    android:textColor="@android:color/white"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="@+id/txt_ket_cuaca"
    app:layout_constraintStart_toEndOf="@+id/txt_ket_cuaca"
    app:layout_constraintTop_toBottomOf="@+id/txt_cuaca"
    tools:text="30 C" />
```

```
<android.support.constraint.Guideline
    android:id="@+id/guideline"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    app:layout_constraintGuide_begin="16dp" />
```

```
<TextView
    android:id="@+id/txt_lokasi"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginStart="8dp"
    tools:text="Gunung kelude"
    android:textStyle="bold"
    app:layout_constraintBottom_toTopOf="@+id/txt_derajat"
    app:layout_constraintStart_toEndOf="@+id/txt_cuaca" />
```

```
<ProgressBar
    android:id="@+id/pd_cuaca"
    style="?android:attr/progressBarStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="@+id/bg" />
```

```
</android.support.constraint.ConstraintLayout>
```

```
<android.support.v7.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    app:layout_collapseMode="pin"
    app:popupTheme="@style/AppTheme.PopupOverlay" />
```

```
</android.support.design.widget.CollapsingToolbarLayout>
```

```
</android.support.design.widget.AppBarLayout>
```

```
<include layout="@layout/content_detail_wisata" />
```

```
<android.support.design.widget.FloatingActionButton
```

```

        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/fab_margin"
        app:layout_anchor="@id/app_bar"
        app:layout_anchorGravity="bottom|end"
        app:srcCompat="@drawable/ic_navigation_black_24dp"
        tools:ignore="VectorDrawableCompat" />

</android.support.design.widget.CoordinatorLayout>

```

h. **DetailWisata.java**

```
public class DetailWisata extends AppCompatActivity implements OnMapReadyCallback {
```

```

    @BindView(R.id.no_image)
    TextView noImage;

    @BindView(R.id.recycler_view)
    RecyclerView recyclerView;

    @BindView(R.id.pd)
    ProgressBar pd;

    @BindView(R.id.no_pos)
    TextView noPos;

    @BindView(R.id.pd_pos)
    ProgressBar pdPos;

    @BindView(R.id.wrapper_map)
    LinearLayout wrapperMap;

    @BindView(R.id.txt_cuaca)
    TextView txtCuaca;

    @BindView(R.id.toolbar_layout)
    CollapsingToolbarLayout collapsingToolbarLayout;

    @BindView(R.id.app_bar)

```

```
AppBarLayout appBarLayout;
@BindView(R.id.txt_ket_cuaca)
TextView txtKetCuaca;
@BindView(R.id.txt_derajat)
TextView txtDerajat;
@BindView(R.id.txt_lokasi)
TextView txtLokasi;
@BindView(R.id.pd_cuaca)
ProgressBar pdCuaca;
@BindView(R.id.img_cover)
ImageView imgCover;
/*@BindView(R.id.txt_deskripsi)
HtmlTextView txtDeskripsi;*/
@BindView(R.id.txt_deskripsi)
WebView txtDeskripsi;
@BindView(R.id.txt_alamat)
TextView txtAlamat;
@BindView(R.id.txt_pos)
TextView txtPos;
@BindView(R.id.txt_tanah)
TextView txtTanah;
@BindView(R.id.txt_gunung)
TextView txtGunung;
@BindView(R.id.txt_tinggi)
TextView txtTinggi;
@BindView(R.id.txt_fasilitas)
WebView txtFasilitas;
private GoogleMap mMap;
private ArrayList<DataItem> images;
WisataItem data;
```

```

//weather icon
Typeface weatherFont;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_detail_wisata);
    ButterKnife.bind(this);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    //get nilai from adapter
    data = getIntent().getParcelableExtra(KEY_DATA);
    if (data != null) {
        getData(data.getIdWisata());
        //cuaca
        //metric = untuk satuan celcius
        getWeather(data.getLatitude(), data.getLongitude(), "metric", API);
        Glide.with(this)
            .load(BASE_URL_IMAGE_WISATA + data.getImage())
            .into(imgCover);
        //txtDeskripsi.setHtml(data.getDeskripsi());
        txtAlamat.setText(data.getLokasi());
        txtTanah.setText(data.getTipe_tanah());
        txtGunung.setText(data.getTipe_gunung());
        txtTinggi.setText(data.getKetinggian() + " mdpl");
        txtDeskripsi.getSettings().setJavaScriptEnabled(true);
        txtDeskripsi.loadData(data.getDeskripsi(), "text/html; charset=utf-8", "UTF-8");
        txtFasilitas.getSettings().setJavaScriptEnabled(true);
        txtFasilitas.loadData(data.getFasilitas(), "text/html; charset=utf-8", "UTF-8");
    }
}

```



```

//list

images = new ArrayList<>();

//back

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

//fonts

weatherFont = Typeface.createFromAsset(getAssets(), "fonts/weathericons-regular-webfont.ttf");

txtCuaca.setTypeface(weatherFont);

FloatingActionButton fab = findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            if (data != null) {
                //maps
                Uri gmmIntentUri = Uri.parse("google.navigation:q=" + data.getLatitude() + "," +
data.getLongitude() + "");
                Intent mapIntent = new Intent(Intent.ACTION_VIEW, gmmIntentUri);
                mapIntent.setPackage("com.google.android.apps.maps");
                startActivity(mapIntent);
            }
        } catch (Exception e) {
            Snackbar.make(view, "Tidak ada aplikasi google maps, silahkan download dulu",
Snackbar.LENGTH_LONG)
                .setAction("Action", null).show();
        }
    }
});

```

```

//recyclerview galery

RecyclerView.LayoutManager mLayoutManager = new LinearLayoutManager(this,
LinearLayoutManager.HORIZONTAL,

    false);

recyclerView.setLayoutManager(mLayoutManager);

recyclerView.setItemAnimator(new DefaultItemAnimator());


recyclerView.setOnItemClickListener(new
GalleryAdapter.RecyclerTouchListener(getApplicationContext(),    recyclerView,    new
GalleryAdapter.ClickListener() {

    @Override

    public void onClick(View view, int position) {

        //kirim data jika ada item yagn di click

        Bundle bundle = new Bundle();

        bundle.putSerializable("images", images);

        bundle.putInt("position", position);


        FragmentTransaction ft = getFragmentManager().beginTransaction();

        SlideshowDialogFragment newFragment = SlideshowDialogFragment.newInstance();

        newFragment.setArguments(bundle);

        newFragment.show(ft, "slideshow");

    }


    @Override

    public void onLongClick(View view, int position) {

        }

    }));


//hide tittle in bar

appBarLayout.addOnOffsetChangedListener(new AppBarLayout.OnOffsetChangedListener()
{

    boolean isShow = true;

```

```
int scrollRange = -1;
```

```
@Override
```

```
public void onOffsetChanged(AppBarLayout appBarLayout, int verticalOffset) {
```

```
    if (scrollRange == -1) {
```

```
        scrollRange = appBarLayout.getTotalScrollRange();
```

```
    }
```

```
    if (scrollRange + verticalOffset == 0) {
```

```
        String title = "Title";
```

```
        if (data != null) {
```

```
            title = data.getJudul();
```

```
        }
```

```
        collapsingToolbarLayout.setTitle(title);
```

```
        isShow = true;
```

```
    } else if (isShow) {
```

```
        collapsingToolbarLayout.setTitle(" "); //carefull there should a space between double  
quote otherwise it wont work
```

```
        isShow = false;
```

```
    }
```

```
}
```

```
});
```

```
// 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 onMapReady(GoogleMap googleMap) {
```

```
    mMap = googleMap;
```

```

    /** Add a marker in Sydney and move the camera
    LatLng sydney = new LatLng(-6.18938381651682, 106.84674588079042);
    mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in Sydney"));
    mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));*/

    if (data != null) {
        getPosPendakian(data.getIdWisata());
    }

    mMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);

    mMap.getUiSettings().setCompassEnabled(true);
    mMap.getUiSettings().setZoomGesturesEnabled(true);
    mMap.getUiSettings().setRotateGesturesEnabled(false);
    mMap.getUiSettings().setZoomControlsEnabled(true);
}

private void getData(String id) {
    recyclerView.setVisibility(View.VISIBLE);
    noImage.setVisibility(View.GONE);
    pd.setVisibility(View.VISIBLE);

    //cek jika data tidak kosong
    //maka di hapus
    if (images != null) {
        images.clear();
    }

    ApiService apiService = RetroClient.getApiService();
    Call<ResponseView> call = apiService.getView(id);
    call.enqueue(new Callback<ResponseView>() {

```

@Override

```
public void onResponse(Call<ResponseView> call, Response<ResponseView> response) {  
    List<DataItem> hasilPesan = response.body().getData();  
    Log.e("Tag", "Hasil image :" + new Gson().toJson(hasilPesan));  
    if (response.body().getResponse().equalsIgnoreCase("true")) {  
        //tambahkan image  
        images.addAll(hasilPesan);  
        GalleryAdapter adapterPesan = new GalleryAdapter(DetailWisata.this, images);  
        recyclerView.setAdapter(adapterPesan);  
        pd.setVisibility(View.GONE);  
    } else {  
        Log.e("Tag", "Gagal req data ");  
        recyclerView.setVisibility(View.GONE);  
        noImage.setVisibility(View.VISIBLE);  
        pd.setVisibility(View.GONE);  
    }  
}
```

@Override

```
public void onFailure(Call<ResponseView> call, Throwable t) {  
    recyclerView.setVisibility(View.GONE);  
    noImage.setVisibility(View.VISIBLE);  
    pd.setVisibility(View.GONE);  
}  
});  
}
```

```
private void getPosPendakian(String id) {  
    wrapperMap.setVisibility(View.VISIBLE);  
    noPos.setVisibility(View.GONE);  
    pdPos.setVisibility(View.VISIBLE);  
}
```

```

ApiService apiService = RetroClient.getApiService();
Call<ResponsePos> call = apiService.getPosPendakian(id);

call.enqueue(new Callback<ResponsePos>() {
    @Override
    public void onResponse(Call<ResponsePos> call, Response<ResponsePos> response) {
        List<PosPendakianItem> hasilPesanan = response.body().getPosPendakian();
        Log.e("Tag", "Hasil List :" + new Gson().toJson(hasilPesanan));
        if (response.body().getResponse().equalsIgnoreCase("true")) {
            initMarker(hasilPesanan);
            txtPos.setText(hasilPesanan.get(0).getPos());
            pdPos.setVisibility(View.GONE);
            wrapperMap.setVisibility(View.VISIBLE);
        } else {
            Log.e("Tag", "Gagal req data ");
            wrapperMap.setVisibility(View.GONE);
            noPos.setVisibility(View.VISIBLE);
            txtPos.setText("0");
            pdPos.setVisibility(View.GONE);
        }
    }
});

@Override
public void onFailure(Call<ResponsePos> call, Throwable t) {
    wrapperMap.setVisibility(View.GONE);
    noPos.setVisibility(View.VISIBLE);
    txtPos.setText("0");
    pdPos.setVisibility(View.GONE);
}
});

```

```

    }

    private void initMarker(List<PosPendakianItem> mListMarker) {
        //iterasi semua data dan tampilkan markernya
        for (int i = 0; i < mListMarker.size(); i++) {
            //set latlng nya
            LatLng location = new LatLng(Double.parseDouble(mListMarker.get(i).getLatitude()),
                Double.parseDouble(mListMarker.get(i).getLongitude()));

            //tambahkan markernya
            mMap.addMarker(new
MarkerOptions().position(location).title(mListMarker.get(i).getNama()));

            //set latlng index 0
            LatLng latLng = new LatLng(Double.parseDouble(mListMarker.get(0).getLatitude()),
                Double.parseDouble(mListMarker.get(0).getLongitude()));

            //lalu arahkan zooming ke marker index ke 0
            mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(new
LatLng(latLng.latitude, latLng.longitude), 17.0f));
        }
    }

    private void getWeather(String latitude, String longitude, String units, String api) {
        pdCuaca.setVisibility(View.VISIBLE);

        ApiService apiService = InitWeather.getApiService();

        Call<ResponseWeather> call = apiService.getWeatherKoordinat(latitude, longitude, units,
api);

        call.enqueue(new Callback<ResponseWeather>() {
            @SuppressWarnings("NewApi")
            @Override
            public void onResponse(Call<ResponseWeather> call, Response<ResponseWeather>
response) {
                //Log.e("Tag", "Hasil List : " + new Gson().toJson(response));

                try {

```

```

        if (response.body().getCod() == 200) {
            if (data != null) {
                Drawable img =
                getResources().getDrawable(R.drawable.ic_location_on_black_24dp);
                //sesuaikan image sesuai isian bisa di start/awal, top/atas,end/akhir, bottom/bawah
                txtLokasi.setCompoundDrawablesRelativeWithIntrinsicBounds(null, null, img,
                null);

                txtLokasi.setText(data.getJudul());
            }

            //keterangan
            txtKetCuaca.setText(cuaca(response.body().getWeather().get(0).getId()));

            //icon

            txtCuaca.setText(Html.fromHtml(Function.setWeatherIcon(response.body().getWeather().get(0).g
            etId(),

                response.body().getSys().getSunrise() * 1000,
                response.body().getSys().getSunset() * 1000)));

            //derajat
            int cuaca = (int) response.body().getMain().getTemp();

            //txtDerajat.setText(String.format("%.2F", response.body().getMain().getTemp()+ "
            °C");

            txtDerajat.setText(String.valueOf(cuaca) + " °C");
            pdCuaca.setVisibility(View.GONE);
        } else {
            Log.e("Tag", "Gagal req data ");
            pdCuaca.setVisibility(View.GONE);
        }
    } catch (Exception e) {
        Toast.makeText(DetailWisata.this, "Cuaca tidak ditemukan",
        Toast.LENGTH_SHORT).show();
        pdCuaca.setVisibility(View.GONE);
    }
}

```



```

@Override

public void onFailure(Call<ResponseWeather> call, Throwable t) {

    pdCuaca.setVisibility(View.GONE);

    Snackbar.make(recyclerView, "Tidak Ada Koneksi internet",
Snackbar.LENGTH_INDEFINITE)

        .setAction("Ulangi", new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                if (data != null) {

                    //cuaca

                    getWeather(data.getLatitude(), data.getLongitude(), "metric", API);

                }

            }

        }).show();

    }

});

}

```

```

@Override

public boolean onCreateOptionsMenu(Menu menu) {

    // Inflate the menu; this adds items to the action bar if it is present.

    getMenuInflater().inflate(R.menu.main, menu);

    return true;

}

```

```

@Override

public boolean onOptionsItemSelected(MenuItem item) {

    // Handle action bar item clicks here. The action bar will

    // automatically handle clicks on the Home/Up button, so long

    // as you specify a parent activity in AndroidManifest.xml.

    int id = item.getItemId();

```

```

//noinspection SimplifiableIfStatement
if (id == R.id.action_settings) {

    //Toast.makeText(this, "Share It", Toast.LENGTH_SHORT).show();

    try {

        Intent sharingIntent = new Intent(Intent.ACTION_SEND);
        sharingIntent.setType("text/plain");
        String shareBody = "";

        if (data != null) {
            shareBody = data.getJudul();

            //hilangkan semua tag html
            //https://stackoverflow.com/questions/240546/remove-html-tags-from-a-string
            String isi = android.text.Html.fromHtml(data.getDeskripsi()).toString();

            //potong string biar tidak panjang
            if (isi.length() > 50) {
                //https://stackoverflow.com/questions/8369708/limiting-the-number-of-characters-
                //in-a-string-and-chopping-off-the-rest/8369746
                isi = isi.substring(0, 100);
            }

            shareBody += "\n" + isi + "...";
        }

        sharingIntent.putExtra(Intent.EXTRA_SUBJECT, getString(R.string.SUBJEK));
        //isi text yang di share
        sharingIntent.putExtra(Intent.EXTRA_TEXT, shareBody);
        startActivity(Intent.createChooser(sharingIntent, getString(R.string.SHARE_VIA)));
    } catch (Exception e) {

    }

    return true;
}

```

```

    } else if (item.getItemId() == android.R.id.home) {
        onBackPressed();
    }

    return super.onOptionsItemSelected(item);
}
}

```

i. **ManfaatPeguunungan.java**

```

public class ManfaatPeguunungan extends Fragment implements
ViewPagerEx.OnPageChangeListener{

```

```

    @BindView(R.id.slider)

```

```

    SliderLayout slider;

```

```

    @BindView(R.id.txt_notif)

```

```

    TextView txtNotif;

```

```

    @BindView(R.id.rv_list)

```

```

    RecyclerView rvList;

```

```

    @BindView(R.id.pd)

```

```

    ProgressBar pd;

```

```

    Unbinder unbinder;

```

```

public ManfaatPeguunungan() {

```

```

    // Required empty public constructor

```

```

}

```

```

@Override

```

```

public View onCreateView(LayoutInflater inflater, ViewGroup container,

```

```

    Bundle savedInstanceState) {

```

```

    // Inflate the layout for this fragment

```

```

    View view = inflater.inflate(R.layout.fragment_list, container, false);

```

```

    unbinder = ButterKnife.bind(this, view);

```

```

        //layoutManager.setStackFromEnd(true);

        rvList.setLayoutManager(new VegaLayoutManager());

        slider.setVisibility(View.VISIBLE);

        //slider

        slider.setPresetTransformer(SliderLayout.Transformer.ZoomOut);

        slider.setPresetIndicator(SliderLayout.PresetIndicators.Center_Bottom);

        slider.setCustomAnimation(new DescriptionAnimation());

        slider.setDuration(4000);

        slider.addOnPageChangeListener(this);

        getData();

        return view;
    }

    @Override
    public void onDestroyView() {
        super.onDestroyView();
        unbinder.unbind();
    }

    private void getData() {
        pd.setVisibility(View.VISIBLE);

        ApiService apiService = RetroClient.getApiService();

        Call<ResponseManfaat> call = apiService.getManfaat();

        call.enqueue(new Callback<ResponseManfaat>() {
            @Override
            public void onResponse(Call<ResponseManfaat> call, Response<ResponseManfaat>
response) {
                List<ManfaatItem> hasilPesanan = response.body().getManfaat();
            }
        });
    }

```

```

Log.e("Tag", "Hasil List :" + new Gson().toJson(hasilPesan));
if (response.body().getResponse().equalsIgnoreCase("true")) {
    try {
        ManfaatAdapter adapterPesan = new ManfaatAdapter(getContext(), hasilPesan);
        // swipeMain.setRefreshing(false);
        rvList.setAdapter(adapterPesan);
        pd.setVisibility(View.GONE);

        for (int i = 0; i < hasilPesan.size(); i++) {
            RequestOptions requestOptions = new RequestOptions();
            requestOptions.centerCrop();

            TextSliderView sliderView = new TextSliderView(getContext());
            // if you want show image only / without description text use DefaultSliderView
instead

            // initialize SliderLayout
            sliderView
                .image(BASE_URL_IMAGE_MANFAAT + hasilPesan.get(i).getImage())
                .description(hasilPesan.get(i).getJudul())
                .setRequestOption(requestOptions)
                .setBackgroundColor(Color.WHITE)
                .setProgressBarVisible(true)
                .setOnSliderClickListener(new BaseSliderView.OnSliderClickListener() {
                    @Override
                    public void onSliderClick(BaseSliderView slider) {
                        Toast.makeText(getContext(), slider.getBundle().get("extra") + "",
Toast.LENGTH_SHORT).show();

                    }
                });

```

```

        //add your extra information
        sliderView.bundle(new Bundle());

        sliderView.getBundle().putString("extra", hasilPesan.get(i).getJudul());

        slider.addSlider(sliderView);
    }
} catch (Exception e){

}

} else {
    Log.e("Tag", "Gagal req data ");
    pd.setVisibility(View.GONE);
}
}

@Override

public void onFailure(Call<ResponseManfaat> call, Throwable t) {
    pd.setVisibility(View.GONE);

    Snackbar.make(rvList, "Tidak Ada Koneksi internet",
Snackbar.LENGTH_INDEFINITE)
        .setAction("Ulangi", new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                getData();
            }
        }).show();
    }
});
}

@Override

public void onPageScrolled(int i, float v, int i1) {
}

```

```

@Override

public void onPageSelected(int i) {

}

@Override

public void onPageScrollStateChanged(int i) {

}

@Override

public void onStop() {
    slider.stopAutoCycle();
    super.onStop();
}
}

```

j. Activity_detail_manfaat.xml

```

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
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"

android:fitsSystemWindows="true"

tools:context=".activity.detail.DetailManfaat">

<android.support.design.widget.AppBarLayout

    android:id="@+id/app_bar"

    android:layout_width="match_parent"

```

```
android:layout_height="@dimen/app_bar_height"
android:fitsSystemWindows="true"
android:theme="@style/AppTheme.AppBarOverlay">
```

```
<android.support.design.widget.CollapsingToolbarLayout
    android:id="@+id/toolbar_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    app:contentScrim="?attr/colorPrimary"
    app:layout_scrollFlags="scroll|exitUntilCollapsed"
    app:toolbarId="@+id/toolbar">
```

```
<ImageView
    android:id="@+id/img_cover"
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:scaleType="fitXY"
    tools:src="@drawable/safety" />
```

```
<android.support.v7.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    app:layout_collapseMode="pin"
    app:popupTheme="@style/AppTheme.PopupOverlay" />
```

```
</android.support.design.widget.CollapsingToolbarLayout>
</android.support.design.widget.AppBarLayout>
```

```
<include layout="@layout/content_detail_manfaat" />
```


</android.support.design.widget.CoordinatorLayout>

k. **DetailManfaat.java**

```
public class DetailManfaat extends AppCompatActivity {

    @BindView(R.id.img_cover)
    ImageView imgCover;

    @BindView(R.id.toolbar_layout)
    CollapsingToolbarLayout collapsingToolbarLayout;

    @BindView(R.id.app_bar)
    AppBarLayout appBarLayout;

    @BindView(R.id.txt_tanggal)
    TextView txtTanggal;

    @BindView(R.id.txt_jam)
    TextView txtJam;

    @BindView(R.id.txt_penulis)
    TextView txtPenulis;

    @BindView(R.id.txt_isi)
    WebView txtIsi;

    //
    ManfaatItem data;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail_manfaat);
        ButterKnife.bind(this);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
    }
}
```

```

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

data = getIntent().getParcelableExtra(KEY_DATA);

if (data != null) {
    txtTanggal.setText(customTanggal(data.getTanggal(), "yyyy-MM-dd HH:mm:ss",
"dd/MM/yyyy"));
    txtJam.setText(customTanggal(data.getTanggal(), "yyyy-MM-dd HH:mm:ss", "HH:mm"));
    txtPenulis.setText("Di tulis oleh : " + data.getNama());

    Glide.with(this)
        .load(BASE_URL_IMAGE_MANFAAT + data.getImage())
        .into(imgCover);
    txtIsi.getSettings().setJavaScriptEnabled(true);
    txtIsi.loadData(data.getDeskripsi(), "text/html; charset=utf-8", "UTF-8");
}

//hide tittle in bar
appBarLayout.addOnOffsetChangedListener(new AppBarLayout.OnOffsetChangedListener()
{
    boolean isShow = true;
    int scrollRange = -1;

    @Override
    public void onOffsetChanged(AppBarLayout appBarLayout, int verticalOffset) {
        if (scrollRange == -1) {
            scrollRange = appBarLayout.getTotalScrollRange();
        }
        if (scrollRange + verticalOffset == 0) {
            String title = "Title";
            if (data != null) {
                title = data.getJudul();
            }
        }
    }
}

```

```

        collapsingToolbarLayout.setTitle(title);

        isShow = true;
    } else if (isShow) {

        collapsingToolbarLayout.setTitle(" "); //carefull there should a space between double
quote otherwise it wont work

        isShow = false;
    }

}

});
}

```

@Override

```

public boolean onCreateOptionsMenu(Menu menu) {

    // Inflate the menu; this adds items to the action bar if it is present.

    getMenuInflater().inflate(R.menu.main, menu);

    return true;
}

```

@Override

```

public boolean onOptionsItemSelected(MenuItem item) {

    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.

    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {

        //Toast.makeText(this, "Share It", Toast.LENGTH_SHORT).show();

        try {

            Intent sharingIntent = new Intent(Intent.ACTION_SEND);

            sharingIntent.setType("text/plain");

            String shareBody = "";

```

```

        if (data != null) {

            shareBody = data.getJudul();

            //hilangkan semua tag html
            //https://stackoverflow.com/questions/240546/remove-html-tags-from-a-string
            String isi = android.text.Html.fromHtml(data.getDeskripsi()).toString();

            //potong string biar tidak panjang

            if (isi.length() > 50) {

                //https://stackoverflow.com/questions/8369708/limiting-the-number-of-characters-
                //in-a-string-and-chopping-off-the-rest/8369746
                isi = isi.substring(0, 100);

            }

            shareBody += "\n" + isi + "...";

        }

        sharingIntent.putExtra(Intent.EXTRA_SUBJECT, getString(R.string.SUBJEK));

        //isi text yang di share
        sharingIntent.putExtra(Intent.EXTRA_TEXT, shareBody);

        startActivity(Intent.createChooser(sharingIntent, getString(R.string.SHARE_VIA)));
    } catch (Exception e) {

    }

    return true;
} else if (item.getItemId() == android.R.id.home) {
    onBackPressed();
}

return super.onOptionsItemSelected(item);
}
}

```

1. SafetyAndRescue.java

```
public class SafetyAndRescue extends Fragment implements
ViewPagerEx.OnPageChangeListener{

    @BindView(R.id.slider)
    SliderLayout slider;

    @BindView(R.id.txt_notif)
    TextView txtNotif;

    @BindView(R.id.rv_list)
    RecyclerView rvList;

    @BindView(R.id.pd)
    ProgressBar pd;

    Unbinder unbinder;

    public SafetyAndRescue() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.fragment_list, container, false);
        unbinder = ButterKnife.bind(this, view);

        //layoutManager.setStackFromEnd(true);
        rvList.setLayoutManager(new VegaLayoutManager());
        slider.setVisibility(View.VISIBLE);

        //slider
        slider.setPresetTransformer(SliderLayout.Transformer.ZoomOut);
        slider.setPresetIndicator(SliderLayout.PresetIndicators.Center_Bottom);
        slider.setCustomAnimation(new DescriptionAnimation());
```

```
slider.setDuration(4000);  
slider.addOnPageChangeListener(this);
```

```
    getData();  
    return view;  
}
```

@Override

```
public void onDestroyView() {  
    super.onDestroyView();  
    unbinder.unbind();  
}
```

```
private void getData() {  
    pd.setVisibility(View.VISIBLE);  
    ApiService apiService = RetroClient.getApiService();  
    Call<ResponseSafety> call = apiService.getSafety();  
    call.enqueue(new Callback<ResponseSafety>() {  
        @Override  
        public void onResponse(Call<ResponseSafety> call, Response<ResponseSafety> response)  
        {  
            List<SafetyItem> hasilPesanan = response.body().getSafety();  
            Log.e("Tag", "Hasil List :" + new Gson().toJson(hasilPesanan));  
            if (response.body().getResponse().equalsIgnoreCase("true")) {  
                try {  
                    SafetyAdapter adapterPesanan = new SafetyAdapter(getContext(), hasilPesanan);  
                    // swipeMain.setRefreshing(false);  
                    rvList.setAdapter(adapterPesanan);  
                    pd.setVisibility(View.GONE);  
  
                    for (int i = 0; i < hasilPesanan.size(); i++) {  
                        RequestOptions requestOptions = new RequestOptions();
```

```

requestOptions.centerCrop();

TextSliderView sliderView = new TextSliderView(getContext());
// if you want show image only / without description text use DefaultSliderView

instead

// initialize SliderLayout
sliderView

    .image(BASE_URL_IMAGE_SAFETY + hasilPesan.get(i).getImage())
    .description(hasilPesan.get(i).getJudul())
    .setRequestOption(requestOptions)
    .setBackgroundColor(Color.WHITE)
    .setProgressBarVisible(true)
    .setOnSliderClickListener(new BaseSliderView.OnSliderClickListener() {
        @Override
        public void onSliderClick(BaseSliderView slider) {
            Toast.makeText(getContext(), slider.getBundle().get("extra") + "",
Toast.LENGTH_SHORT).show();

        }
    });

//add your extra information
sliderView.bundle(new Bundle());

sliderView.getBundle().putString("extra", hasilPesan.get(i).getJudul());
slider.addSlider(sliderView);
}
} catch (Exception e){

}
} else {
    Log.e("Tag", "Gagal req data ");

```

```

        pd.setVisibility(View.GONE);
    }
}

@Override

public void onFailure(Call<ResponseSafety> call, Throwable t) {

    pd.setVisibility(View.GONE);

    Snackbar.make(rvList, "Tidak Ada Koneksi internet",
Snackbar.LENGTH_INDEFINITE)

        .setAction("Ulangi", new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                getData();

            }

        }).show();

    }

});

}

@Override

public void onPageScrolled(int i, float v, int i1) {

}

@Override

public void onPageSelected(int i) {

}

@Override

public void onPageScrollStateChanged(int i) {

}

```



```

@Override

public void onStop() {

    slider.stopAutoCycle();

    super.onStop();

}

}

```

m. **activity_detail_artikel.xml**

```

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout
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"

android:fitsSystemWindows="true"

tools:context=".activity.detail.DetailManfaat">

<android.support.design.widget.AppBarLayout

    android:id="@+id/app_bar"

    android:layout_width="match_parent"

    android:layout_height="@dimen/app_bar_height"

    android:fitsSystemWindows="true"

    android:theme="@style/AppTheme.AppBarOverlay">

<android.support.design.widget.CollapsingToolbarLayout

    android:id="@+id/toolbar_layout"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:fitsSystemWindows="true"

    app:contentScrim="?attr/colorPrimary"

    app:layout_scrollFlags="scroll|exitUntilCollapsed"

```

```
app:toolbarId="@+id/toolbar">
```

```
<ImageView
```

```
    android:id="@+id/img_cover"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="300dp"
```

```
    android:scaleType="fitXY"
```

```
    tools:src="@drawable/safety" />
```

```
<android.support.v7.widget.Toolbar
```

```
    android:id="@+id/toolbar"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="?attr/actionBarSize"
```

```
    app:layout_collapseMode="pin"
```

```
    app:popupTheme="@style/AppTheme.PopupOverlay" />
```

```
</android.support.design.widget.CollapsingToolbarLayout>
```

```
</android.support.design.widget.AppBarLayout>
```

```
<include layout="@layout/content_detail_manfaat" />
```

```
</android.support.design.widget.CoordinatorLayout>
```

n. content_detail_artikel.xml

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<android.support.v4.widget.NestedScrollView
```

```
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"
```

```
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
```

```
tools:context=".activity.detail.DetailManfaat"
tools:showIn="@layout/activity_detail_manfaat">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="@dimen/nav_header_vertical_spacing"
    android:orientation="vertical">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
```

```
<TextView
    android:id="@+id/txt_tanggal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:drawableLeft="@drawable/ic_date"
    android:textSize="@dimen/subsub_judul"
    tools:text="2018/10/01" />
```

```
<TextView
    android:id="@+id/txt_jam"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:drawableLeft="@drawable/ic_access_time_black_24dp"
    android:textSize="@dimen/subsub_judul"
    tools:text="18.00 wib" />
```

```
<TextView
    android:id="@+id/txt_penulis"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:drawableLeft="@drawable/ic_person_black_24dp"
    android:textSize="@dimen/subsub_judul"
    tools:text="Di posting oleh : Ujib" />
```

```
</LinearLayout>
```

```
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@drawable/bg_artikel" />
```

```
<WebView
    android:id="@+id/txt_isi"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    tools:text="@string/large_text" />
```

```
</LinearLayout>
```

```
</android.support.v4.widget.NestedScrollView>
```

o. DetailArtikel.java

```
public class DetailManfaat extends AppCompatActivity {

    @BindView(R.id.img_cover)
    ImageView imgCover;

    @BindView(R.id.toolbar_layout)
    CollapsingToolbarLayout collapsingToolbarLayout;

    @BindView(R.id.app_bar)
    AppBarLayout appBarLayout;

    @BindView(R.id.txt_tanggal)
    TextView txtTanggal;

    @BindView(R.id.txt_jam)
    TextView txtJam;

    @BindView(R.id.txt_penulis)
    TextView txtPenulis;

    @BindView(R.id.txt_isi)
    WebView txtIsi;

    ManfaatItem data;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail_manfaat);
        ButterKnife.bind(this);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        data = getIntent().getParcelableExtra(KEY_DATA);
        if (data != null) {
            txtTanggal.setText(customTanggal(data.getTanggal(), "yyyy-MM-dd HH:mm:ss", "dd/MM/yyyy"));
            txtJam.setText(customTanggal(data.getTanggal(), "yyyy-MM-dd HH:mm:ss", "HH:mm"));
        }
    }
}
```

```

txtPenulis.setText("Di tulis oleh : " + data.getNama());

Glide.with(this)
    .load(BASE_URL_IMAGE_MANFAAT + data.getImage())
    .into(imgCover);
txtIsi.getSettings().setJavaScriptEnabled(true);
txtIsi.loadData(data.getDeskripsi(), "text/html; charset=utf-8", "UTF-8");
}

//hide tittle in bar
appBarLayout.addOnOffsetChangedListener(new AppBarLayout.OnOffsetChangedListener()
{
    boolean isShow = true;
    int scrollRange = -1;

    @Override
    public void onOffsetChanged(AppBarLayout appBarLayout, int verticalOffset) {
        if (scrollRange == -1) {
            scrollRange = appBarLayout.getTotalScrollRange();
        }
        if (scrollRange + verticalOffset == 0) {
            String title = "Title";
            if (data != null) {
                title = data.getJudul();
            }
            collapsingToolbarLayout.setTitle(title);
            isShow = true;
        } else if (isShow) {
            collapsingToolbarLayout.setTitle(" "); //carefull there should a space between double
            quote otherwise it wont work
            isShow = false;
        }
    }
}

```

```

    }
});
}

```

@Override

```

public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

```

@Override

```

public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        //Toast.makeText(this, "Share It", Toast.LENGTH_SHORT).show();
        try {
            Intent sharingIntent = new Intent(Intent.ACTION_SEND);
            sharingIntent.setType("text/plain");
            String shareBody = "";

            if (data != null) {
                shareBody = data.getJudul();
                //hilangkan semua tag html
                //https://stackoverflow.com/questions/240546/remove-html-tags-from-a-string
                String isi = android.text.Html.fromHtml(data.getDeskripsi()).toString();
            }
        } catch (Exception e) {
            // TODO: Handle the exception
        }
    }
}

```

```

        //potong string biar tidak panjang

        if (isi.length() > 50) {
            //https://stackoverflow.com/questions/8369708/limiting-the-number-of-characters-
            in-a-string-and-chopping-off-the-rest/8369746
            isi = isi.substring(0, 100);
        }
        shareBody += "\n" + isi + "...";
    }

    sharingIntent.putExtra(Intent.EXTRA_SUBJECT, getString(R.string.SUBJEK));
    //isi text yang di share
    sharingIntent.putExtra(Intent.EXTRA_TEXT, shareBody);
    startActivity(Intent.createChooser(sharingIntent, getString(R.string.SHARE_VIA)));
} catch (Exception e) {

}

return true;
} else if (item.getItemId() == android.R.id.home) {
    onBackPressed();
}

return super.onOptionsItemSelected(item);
}
}

```


p. **Koneksi.php**

```
<?php

    // definisikan koneksi ke database

    $server = "localhost";

    $username = "root";

    $password = "";

    $database = "db_wisatagunung";

    //cek koneksi

    // Koneksi dan memilih database di server

    $con =mysqli_connect($server,$username,$password,$database) or die("Koneksi gagal");

    //mysqli_select_db($database) or die("Database tidak bisa dibuka");

?>
```

q. **Wisata.php**

```
<?php

    //koneksi

    // definisikan koneksi ke database

    include '../..//koneksi.php';

    $sql = mysqli_query($con,"SELECT * FROM wisata ORDER BY id_wisata DESC");

    //untuk menampung isi data

    $response=array();

    $cek=mysqli_num_rows($sql);

    if($cek >0){

        $response["wisata"]=array();

        //$responsepos["pos"]=array();

        //buat var array yang dibutuhkan

        $responsepos=array();

        $data=array();

        $data2=array();
```

```

//perulangan
//ambil data dr query
while ($row=mysqli_fetch_array($sql)){
    //ambil value sesuai dengan nama tabel dan masukan ke array
    $data["id_wisata"]=$row["id_wisata"];
    $data["image"]=$row["image"];
    $data["judul"]=$row["judul"];
    $data["lokasi"]=$row["lokasi"];
    $data["deskripsi"]=$row["deskripsi"];
    $data["latitude"]=$row["latitude"];
    $data["longitude"]=$row["longitude"];

    //pos pendakian
    //ambil data sesuai dengan id
    $cari = $row["id_wisata"];

    $sql2 = mysqli_query($con,"SELECT * FROM pos WHERE id_wisata = '$cari' ORDER
    BY id_pos DESC");

    while ($pos = mysqli_fetch_array($sql2)){
        //masukan nilai ke array
        $data2["id_pos"]=$pos["id_pos"];
        $data2["id_wisata"]=$pos["id_wisata"];
        $data2["nama"]=$pos["nama"];
        $data2["latitude"]=$pos["latitude"];
        $data2["longitude"]=$pos["longitude"];

        //push data untuk menambahkan array ke array
        array_push($responsepos,$data2);

        // print_r($pos);

        //masukan data ke array menjadi array array
        $data["pos"] = $responsepos;

        // $data["posku"] = array($data2);
    }
}

```

```

        $response["pesan"]="berhasil Mengambil Data";
        $response["response"]="true";
        array_push($response["wisata"],$data);
        // array_push($response["wisata"],$responsepos);
        //print_r($row);
    }
    //mengubah data menjadi JSON
    echo json_encode($response);
    // echo json_encode($responsepos);
} else{
    $response["pesan"]="Gagal Mengambil Data";
    $response["response"]="false";
    echo json_encode($response);
}

?>

```

r. **ManfaatPendakian.php**

```

<?php
    //koneksi
    // definisikan koneksi ke database
    include '../koneksi.php';

    $sql = mysqli_query($con,"SELECT * FROM manfaat t1 INNER JOIN admin t2 ON
t1.id_admin = t2.id_admin ORDER BY id_manfaat DESC");

    //query untuk menampilkan semua data di table
    // $sql=mysqli_query($con,"SELECT * FROM jadwal_tranfusi ORDER BY id_tranfusi desc");
    //untuk menampung isi data
    $response=array();
    $cek=mysqli_num_rows($sql);

```

```

if($cek >0){
    $response["manfaat"]=array();
    //perulangan
    while ($row=mysqli_fetch_array($sql)){
        $data=array();
        $data["id_manfaat"]=$row["id_manfaat"];
        $data["id_admin"]=$row["id_admin"];
        $data["image"]=$row["image"];
        $data["judul"]= utf8_encode($row["judul"]);
        $data["deskripsi"]= utf8_encode($row["deskripsi"]);
        $data["tanggal"]=$row["tanggal"];
        $data["nama"]=$row["nama"];

        $response["pesan"]="berhasil Mengambil Data";
        $response["response"]="true";
        array_push($response["manfaat"],$data);
        // print_r($row);
    }
    //mengubah data menjadi JSON
    echo json_encode($response);
}else{
    $response["pesan"]="Gagal Mengambil Data";
    $response["response"]="false";
    echo json_encode($response);
}
?>

```

s. **SafetyAndRescue.php**

```
<?php

//koneksi

// definisikan koneksi ke database

include '../koneksi.php';

    $sql = mysqli_query($con,"SELECT * FROM safety t1 INNER JOIN  admin t2 ON t1.id_admin
= t2.id_admin ORDER BY id_safety DESC");

    //query untuk menampilkan semua data ditable

    // $sql=mysqli_query($con,"SELECT * FROM jadwal_tranfusi ORDER BY id_tranfusi desc");

    //untuk menampung isi data

    $response=array();

    $cek = mysqli_num_rows($sql);

    if($cek >0){

        $response["safety"]=array();

        //perulangan

        while ($row=mysqli_fetch_array($sql)){

            $data=array();

            $data["id_safety"]=$row["id_safety"];

            $data["id_admin"]=$row["id_admin"];

            $data["image"]=$row["image"];

            $data["judul"]= utf8_encode($row["judul"]);

            $data["deskripsi"]= utf8_encode($row["deskripsi"]);

            $data["tanggal"]=$row["tanggal"];

            $data["nama"]=$row["nama"];

            $response["pesan"]="berhasil Mengambil Data";

            $response["response"]="true";

            array_push($response["safety"],$data);

            // print_r($row);

        }

    }
```

```
//mengubah data menjadi JSON
echo json_encode($response);
}else{
    $response["pesan"]="Gagal Mengambil Data";
    $response["response"]="false";
    echo json_encode($response);
}
```

?>