BAB III
ANALISI DAN PERANCANGAN SISTEM

III.1 Analisis Sistem

Analisis sistem bertujuan untuk mengidentifikasikan permasalahan-permasalahan yang terdapat pada sistem serta menentukan kebutuhan-kebutuhan dari sistem yang dibangun. Analisis tersebut meliputi analisis masalah, analisis aplikasi sejenis, analisis arsitektur sistem, analisis kebutuhan non fungsional dan analisis data.

III.1.1 Analisis Masalah

Diketahui Berdasarkan hasil survey kuesioner responden pengunjung wisata di kota Bandung setuju jika mereka sering tidak tahu informasi wisata yang ada di kota Bandung dikarenakan kurangnya informasi wisata.

Serta, berdasarkan hasil kuesioner lainnya terhadap wisata di kota Bandung, diperoleh hasil bahwa wisatawan membutuhkan media online tambahan sehingga dapat membantu pihak wisatwan dalam mendapatkan infromasi wisata.. Oleh karena itu, untuk memenuhi kebutuhan yang diinginkan maka peneliti tertarik untuk membangun sebuah aplikasi berbasis mobile (Android) yang bisa menyediakan berbagai informasi wisata yang diharapkan akan mempermudah pengguna dalam mendapatkan informasi objek wisata yang diinginkan dan mempermudah wisatawan dalam mendapatkan informasi wisata yang detail.

1. Bagaimana membangun aplikasi untuk mendeteksi informasi yang ada di dalam objek.
2. Bagaimana membuat aplikasi yang dapat mempermudah user dalam mendapatkan deskripsi suatu wisata secara realtime.

III.1.2 Analisis Aplikasi sejenis

Analisis aplikasi sejenis dimaksudkan untuk menganalisis fungsionalitas dan alur dari aplikasi yang telah ada pada domain yang sama dengan perangkat lunak pada penelitian ini. Hal ini bertujuan untuk memilah fungsionalitas dan alur yang akan diadopsi pada sistem yang diteliti berdasarkan manfaat yang sudah dihasilkan

dari aplikasi yang dianalisis dan mencari *uniqueness* dari perangkat lunak yang akan dibangun pada penelitian ini.

Pada penelitian ini terdapat dua aplikasi yang akan diteliti diantaranya adalah Imageotag dan TouristEye dengan pertimbangan kedua aplikasi tersebut berada pada platform mobile android dan memiliki jumlah unduhan diatas 50.000. Paramater pembanding yang digunakan mengacu kepada komponen-komponen sistem informasi.

1. Imageotag

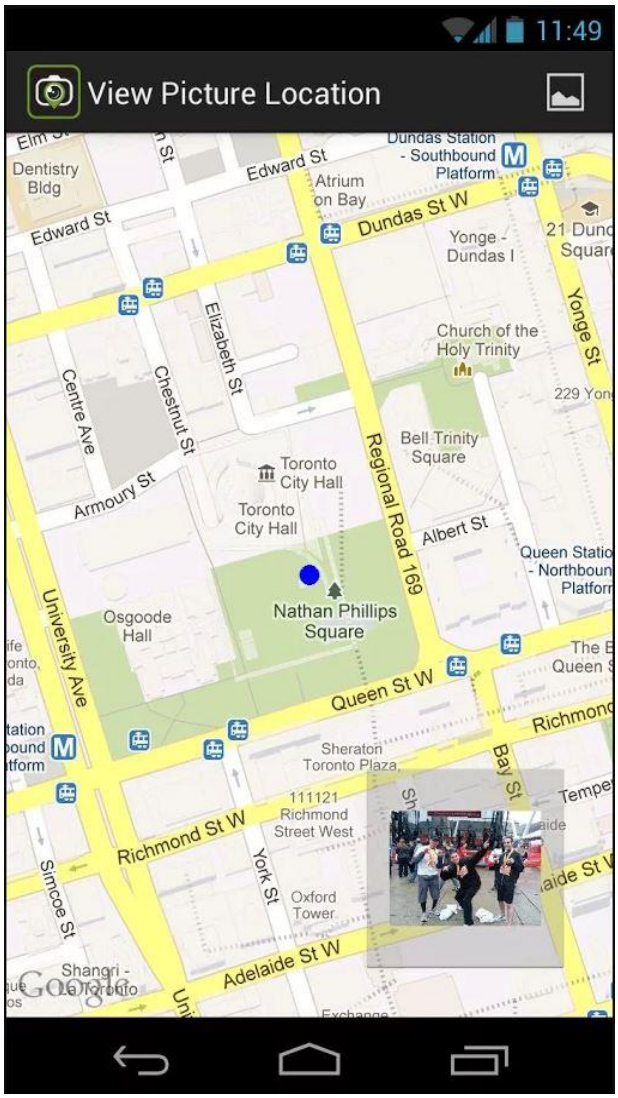
Imgeotag adalah aplikasi kamera ini mampu melakukan geotagging pada setiap foto dan merkamnya. Data yang ditanamkan termasuk GPS yang *di-embed*,lokasi, *accelerometer*, *bearing*, *compas*, *light lux*, *magnetic field*, *pitch*, *roll*, *speed*, *temperature* dan data waktu. Setiap kali pengambilan gambar, kamera ini akan mendekteksi lokasi pengemabilan gambar dengan kamera lain, foto yang diambil bisa dimasukkan ke aplikasi ini lalu pengguna bisa melakukan tag pada foto lainnya yang lebih dulu terekam dalam aplikasi ini. Berikut adalah Gambar III.1 Screenshoot Imageotag.



Gambar III. 1 Sceenshoot Imageotag

2. GeoTag It!

GeoTag It! adalah aplikasi mobile yang dikembangkan dengan berfokus pada DSLR atau point-and-shoot kamera yang memiliki merekam tempat di mana foto-foto indah yang diambil. Berikut adalah Gambar III.2 Screenshot Geotag It



Gambar III. 2 ScreenShot GeoTag

Berdasarkan fungsi pada masing-masing aplikasi tersebut maka dapat dilakukan analisis perbandingan dari kedua aplikasi diatas dengan menggunakan parameter komponen sistem informasi.Berikut adalah Tabel III. Analisi Perbandingan Aplikasi Sejenis.

Tabel III. 1 Analisis Perbandingan Aplikasi Sejenis

Komponen Pembanding	Imageotag	GeoTag It!
Input	Masukan berupa gambar yang diambil langgsung atau gambar yang ada di dalam device	Masukan berupa data gambar yang diambil langsung.
Proses	Data masukan diproses menjadi beberapa fungsionalitas seperti *location*, *temperatur* dan data waktu.	Data masukan diproses menjadi beberapa fungsionalitas seperti *View Trip in 3D* dan *View Picture Location.*
Output	Geo-tagged gambar, embedding GPS & NET lokasi, accelerometer, bantalan, kompas, tanggal, lux cahaya, medan magnet, pitch, kedekatan, roll, kecepatan, suhu & data time. Imageotag menciptakan Google Earth Tours dari urutan gambar GPS	Visualisasikan perjalanan dan berbagi dengan teman, menelusuri kembali langkah anda dalam 3D dan melihat dimana posel atau tablet yang diambil pada peta.
Teknologi	Teknologi kamera geo-tagged dan teknologi accelerometer	Teknologi geo-tagging dan GPS
Software	Aplikasi ini menggunakan sistem operasi android dengan minimal version 2.2 Gingerbread dan lebih tinggi	Aplikasi ini menggunakan sistem operasi android dengan minimal version 2.2

Komponen Pembanding	Imageotag	GeoTag It!
		Gingerbread dan lebih tinggi
Hardware	Aplikasi ini menggunakan beberapa perangkat keras yang sudah terpasang default disetip perangkat android diantaranya: Kamera, GPS, dan A-GPS	Aplikasi ini menggunakan beberapa perangkat keras yang sudah terpasang default disetip perangkat android diantaranya: Kamera, GPS, dan A-GPS
Data	menggunakan penyimpanan internal dan external pada perangkat mobile.	menggunakan penyimpanan internal dan external pada perangkat mobile.
Komponen Kendali	Aplikasi ini akan menampilkan warning untuk melakukan update jika pengguna yang menggunakan aplikasi ini tidak melakukan update secara berkala.	Aplikasi ini akan menampilkan warning untuk melakukan update jika pengguna yang menggunakan aplikasi ini tidak melakukan update secara berkala.

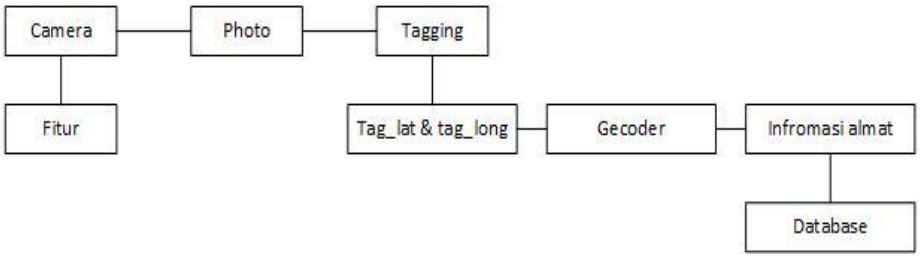
Dari hasil analisis kedua aplikasi sejenis, aplikasi yang akan dibangun akan memiliki beberapa fungsionalitas diantaranya:

1. Memiliki fitur geotagging.
2. Memiliki fitur geofancing.
3. Mengetahui deskripsi suatu objek wisata.

Berdasarkan analisis tersebut juga dapat disimpulkan keunikan perangkat lunak yang dibangun dibandingkan dengan aplikasi sejenis yaitu dapat mnegetahui detail objek wisata yang sedang dikunjungi.

III.1.3 Analisi Teknologi Geotagging

Aplikasi ini memanfaatkan teknologi geotagging yang sudah tertanam di dalam camera smartphone android, data geotagging yang diambil adalah tag_latitude dan tag_longitude. Tag tersebut akan digunakan sebagai data masukan untuk proses geocoder. Hasil dari geocoder adalah berupa informasi alamat tempat pengambilan foto. Dibawah ini merupakan gambar proses pengambilan data geotagging sebagai masukan geocoder hingga penyimpanan data ke database. Berikut adalah Gambar III. 4 Mekanisme Pegambilan Data Geotagging.



Gambar III. 3 Mekanisme Pengambilan Data Geotagging

Mengaktifkan camera pada smartphone android, kemudian pengguna mengambil photo dan proses tagging tersimpan dalam Exif Header yang akan diambil Tag_latitude dan Tag longitude untuk proses perubahan data latitude dan longitude menjadi informasi alamat melalui Geocoder untuk menyimpan data alamat dalam database.

Berikut ini adalah manfaat geotagging:

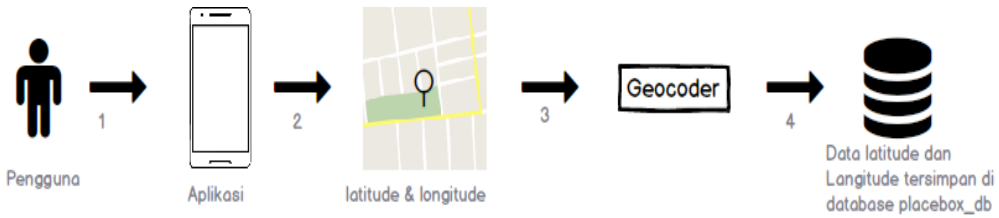
a. Geotagging dapat membantu pengguna menemukan berbagai macam informasi lokasi spesifik. Misalnya, seseorang dapat menemukan gambar-gambar yang diambil di dekat lokasi tertentu dengan memasukkan koordinat lintang dan bujur ke mesin photo geotagging-enabled pencarian. Geotagging diaktifkan layanan informasi juga dapat berpotensi digunakan untuk mencari berita berbasis lokasi, situs web, atau sumber lain. Geotagging dapat memberitahu pengguna lokasi isi gambar yang diberikan atau media lain dari sudut pandang.

Dan sebaliknya pada beberapa platform media menunjukkan media yang relevan dengan lokasi tertentu.

b. Secara umum, proses ini telah disebut geocoding (photo geocode), sebuah istilah yang lebih sering merujuk kepada proses mengambil non-koordinat geografis berdasarkan pengidentifikasi, seperti alamat jalan, dan menemukan koordinat geografis yang terkait atau sebaliknya untuk reverse geocoding, atau penggunaan kamera yang memasukkan koordinat ketika membuat gambar, misalnya menggunakan built dalam penerima GPS.
c. Kamera pada smartphone android sudah dilengkapi GPS yang bisa menambahkan informasi lokasi kedalam metadata photo atau biasa disebut dengan data Exif. Sehingga photo tersebut lokasinya bisa dilihat didalam peta Google map.

III.1.3.1 Proses Mekanisme Pengambilan Data Geotagging

Proses mekanisme pengambilan data geotagging merupakan salah satu proses dimana pengguna/user mengambil data geotagging. Berikut adalah proses pengambilan data geotagging:



Gambar III. 4 Proses Mekanisme Pengambilan Data Geotagging

1. Pengguna melakukan pengambilan foto parkir liar di aplikasi.
2. Sistem mendapatkan data latitude dan longitude yang diambil.
3. Sistem mengambil dan melakukan perubahan data latitude dan longitude melalui proses geocoder menjadi informasi alamat.
4. Sistem akan menyimpan data latitude dan longitude serta informasi alamat ke dalam database dan sistem akan menampilkan informasi laporan parkir liar di dalam aplikasi.

III.1.3.2 Sourece Code Geotagging

Dibawah ini merupakan source code mekanisme pengambilan data geotagging dari perangkat lunak yang dibangun. Berikut adalah Gambar III.6 Source Geotagging.

```
private void getTag() {
    //untuk mengambil tagging waktu, lotitude, dan longitude dalam foto
    try {
        exif = new ExifInterface(mCurrentPhotoPath);
        tag_datetime = exif.getAttribute(ExifInterface.TAG_DATETIME);
        String attrLATITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE);
        String attrLATITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE_REF);
        String attrLONGITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE);
        String attrLONGITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE_REF);
        if ((attrLATITUDE != null)
                && (attrLATITUDE_REF != null)
                && (attrLONGITUDE != null)
                && (attrLONGITUDE_REF != null)) {
            if (attrLATITUDE_REF.equals("N")) {
                latitude = convertToDegree(attrLATITUDE);
                tag_lat = String.valueOf(latitude);
            } else {
                latitude = 0 - convertToDegree(attrLATITUDE);
                tag_lat = String.valueOf(latitude);
            }

            if (attrLONGITUDE_REF.equals("E")) {
                longitude = convertToDegree(attrLONGITUDE);
                tag_long = String.valueOf(longitude);
            } else {
                longitude = 0 - convertToDegree(attrLONGITUDE);
                tag_long = String.valueOf(longitude);
            }
        }
```

```
        GPSTracker tracker = new GPSTracker(this);
        if (!tracker.canGetLocation()) {
            tracker.showSettingsAlert();
        }
        latitude = tracker.getLatitude();
        longitude = tracker.getLongitude();

        Geocoder geocoder = new Geocoder(this, Locale.getDefault());
        List<Address> addresses = null;
        try {
            if (tag_lat ==  null) {
                addresses = geocoder.getFromLocation(latitude, longitude, 1);
                address = addresses.get(0).getThoroughfare();
                lat_user = addresses.get(0).getLatitude();
                long_user = addresses.get(0).getLongitude();
                kota = addresses.get(0).getSubAdminArea();
                aq.id(R.id.tvJalan).text(address);
            } else {
                //untuk geocoder
                addresses = geocoder.getFromLocation(Double.parseDouble(tag_lat), Double.parseDouble(tag_long), 1);
                address = addresses.get(0).getThoroughfare();
                lat_user = Double.valueOf(tag_lat);
                long_user = Double.valueOf(tag_long);
                kota = addresses.get(0).getSubAdminArea();
                aq.id(R.id.tvJalan).text(address);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

Gambar III. 5 Source Geotagging

III.1.4 Analisis Geofencing

Geofencing adalah fitur dalam program perangkat lunak yang menggunakan global positioning system (GPS) untuk menentukan batas-batas geografis. Program yang menggabungkan geofencing memungkinkan untuk mengatur *triggers*, sehingga bila suatu benda dengan menggunakan perangkat tertentu memasuki atau

keluar batas-batas yang ditetapkan, maka akan muncul suatu pemberitahuan (*notifikasi*) yang akan dikirim ke pengguna aplikasi berfitur *geofencing* tersebut.

Secara garis besar, koordinat geografis digunakan untuk membatasi suatu daerah tertentu (*mapping*) sebagai pagar virtual (*geofence*) suatu daerah. Sistem ini menentukan objek yang dilacak terletak di dalam atau di luar wilayah geofenced. Notifikasi akan muncul ketika objek memasuki atau keluar daerah geofence tersebut. Teknologi ini juga dapat memungkinkan untuk pendeteksian kedekatan antara ponsel yang dilacak dengan area *geofenced* tertentu.



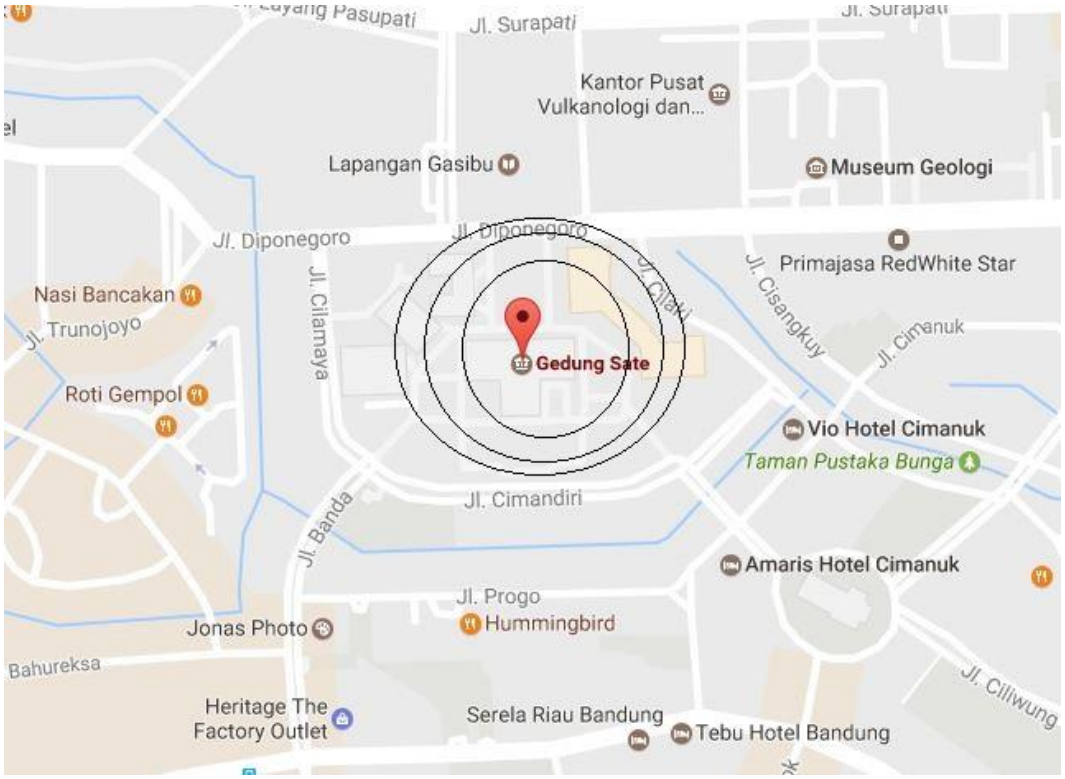
Gambar III. 6 Analisi Geofencing

Pada Gambar III.7 daerah A telah di *geofence*. Sehingga ketika pengguna masuk ke daerah yang telah di *geofence* tersebut, maka akan muncul pemberitahuan pada ponsel berupa informasi tempat-tempat parkir resmi dan bukan tempat parkir resmi.

Geofencing dapat diterapkan untuk berbagai bidang terkait dengan pemantauan orang, kendaraan dan informasi. Hal yang paling utama yaitu dikembangkan untuk sektor informasi tempat-tempat parkir resmi. Akan tetapi aplikasi ini juga dapat diterapkan untuk keselamatan sipil, keamanan, sektor transportasi atau pekerjaan umum.

III.1.4.1 Daerah Geofencing

Untuk ukuran daerah yang di *geofence* berkisar dari beberapa puluhan meter sampai beberapa kilometer. Bentuk daerah yang di geofence tidak beraturan, bentuknya dapat berupa lingkaran, angka geometris, seperti persegi atau persegi panjang, atau lebih rumit, seperti poligon kompleks. Berikut adalah Gambar III.8 Daerah Geofencing.



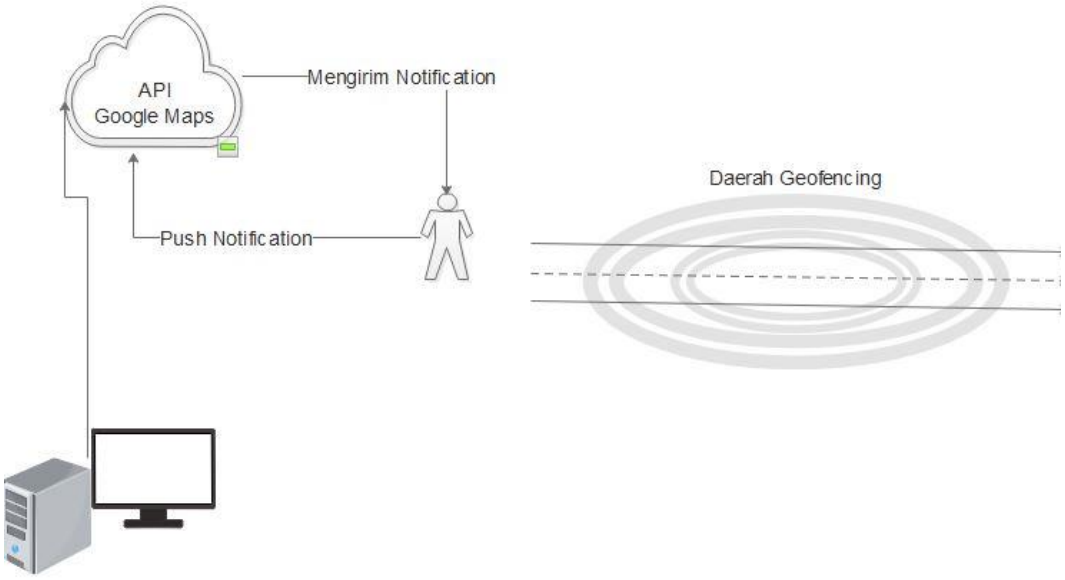
Gambar III. 7 Daerah Geofencing

III.1.4.2 Mekanisme Penentuan Titik

Mekanisme Geofencing untuk menentukan titik yang akan di geofencingkan adalah dengan cara menentukan terlebih dahulu latitude dan longitude yang ingin di beri geofencing, setelah mendapatkan latitude dan longitude, lalu tentukan radius yang ingin dimasukkan, radius yang ditentukan ini dalam jangka permeter.

III.1.4.3 Mekanisme Penerimaan Geofencing

Teknik ini dimaksudkan untuk mendeteksi pengguna yang memasuki daerah geofencing.



Gambar III. 8 Mekanisme Penerimaan Geofencing

Pada Gambar III.9 pengguna masuk ke daerah yang telah di geofence. Kemudian deteksi terjadi ketika smartphone pengguna memasuki daerah yang di geofence, aplikasi yang di tanam di smartphone pengguna memberikan *Push Notification* melalui Google Maps, yang kemudian Google Maps mengirimkan kembali informasi tempat-tempat parkir resmi yang di geofence ke ponsel pengguna. Berikut ini adalah fitur – fitur dalam geofencing :

a. Area (Zone)

   Geofence Area adalah batas-batas virtual / imajiner yang dibuat oleh pemakai untuk membatasi area atau lokasi tertentu. Super Spring GPS Tracker akan menganalisa, membuat laporan dan mengirim alert bila diperlukan, bila kendaraan keluar atau masuk area tersebut.

b. Check Point

   Check Point adalah tempat-tempat yang telah ditentukan oleh pengguna yang harus dituju oleh kendaraan. Super Spring GPS Tracking Device akan mengirimkan alert apabila kendaraan masuk dan keluar dari Check Point.

c. POI (Point Of Interest)

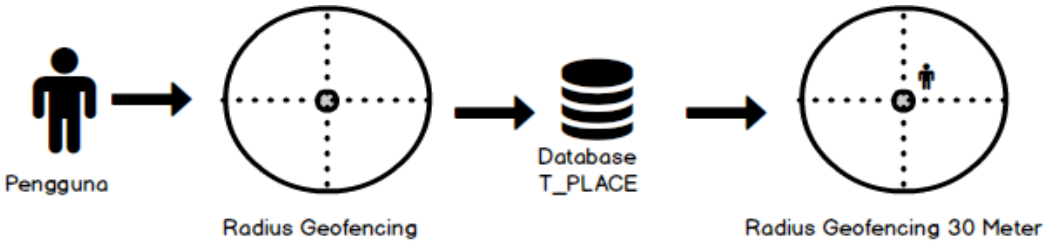
POI adalah lokasi-lokasi yang apabila kendaraan berada di lokasi tersebut, dapat disimpan didalam database, sehingga memudahkan pelacakan dikemudian hari. Untuk memudahkan POI pun dapat dirubah menjadi Check Point.

d. Odometer Tracking Device secara otomatis merekam total jarak yang ditempuh oleh kendaraan. Jarak tersebut dihitung berdasarkan perubahan posisi kendaraan dari waktu ke waktu. Nilai Odometer dilaporkan ke tracking server bersama-sama dengan data posisi, kecepatan, arah dan status I/O. Odometer dapat dimanfaatkan antara lain untuk menganalisa:
    1. Jarak tempuh suatu kendaraan secara keseluruhan /per rit perjalanan.
    2. Menghitung rasio jarak tempuh kendaraan dengan pemakaian BBM.
    3. Membantu perencanaan pemeliharaan kendaraan secara regular.

III.1.4.3.1 Proses Mekanisme Penerimaan Radius Geofencing

Proses mekanisme penerimaan radius geofencing merupakan salah satu proses dimana pengguna/user dapat menentukan radius dari tempat pengguna/user berada.

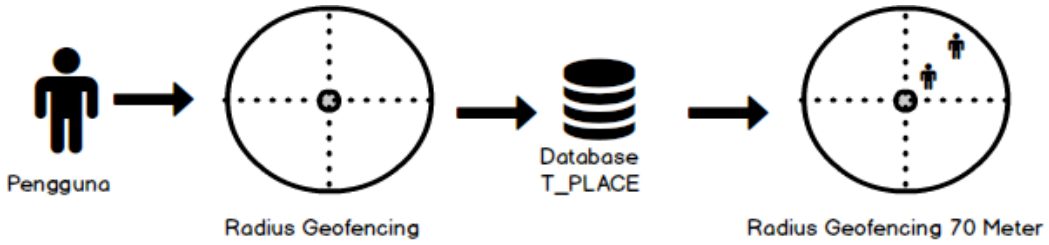
Berikut adalah proses penerimaan radius geofencing 30 meter dari tempat pengguna/user berada.



Gambar III. 9 Proese Penerimaan Radius Geofencing 30 Meter

1. User memilih jarak radius geofencing 30 meter dari posisi lokasi user berada.
2. Sistem mengambil atau membandingkan semua data tempat parkir di database pada tabel tempat parkir dari lokasi user berada.
3. Sistem menampilkan data tempat parkir dari radius 30 meter yang dipilih user.

Berikut adalah proses penerimaan radius geofencing 70 meter dari tempat pengguna/user berada:

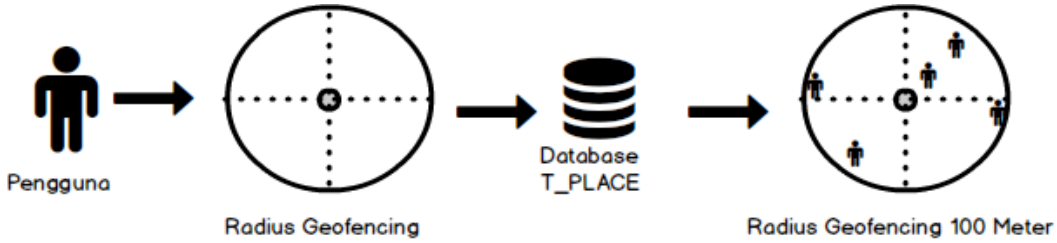


Gambar III. 10 Proese Penerimaan Radius Geofencing 70 Meter

1. User memilih jarak radius geofencing 70 meter dari posisi lokasi user berada.
2. Sistem mengambil atau membandingkan semua data tempat parkir di database pada tabel tempat parkir dari lokasi user berada.

Sistem menampilkan data tempat parkir dari radius 70 meter yang dipilih user

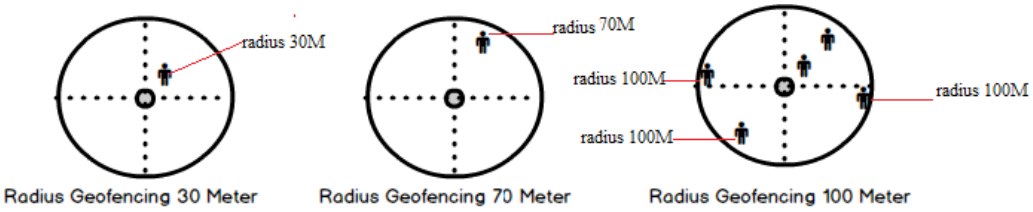
Berikut adalah proses penerimaan radius geofencing 100 meter dari tempat pengguna/user berada:



Gambar III. 11 Proese Penerimaan Radius Geofencing 100 Meter

1. User memilih jarak radius geofencing 100 meter dari posisi lokasi user berada.
2. Sistem mengambil atau membandingkan semua data tempat parkir di database pada tabel tempat parkir dari lokasi user berada.
3. Sistem menampilkan data tempat parkir dari radius 100 meter yang dipilih user.

Berikut adalah perbedaan proses penerimaan radius geofencing yang dipilih user:



Gambar III. 12 Perbedaan Proses Radius Geofencing

III.1.4.3.2 Source Code Geofencing

Dibawah ini merupakan source code fitur dari mekanisme penerimaan geofencing dari perangkat lunak yang dibangun. Berikut adalah Gambar III.14 Source Code Geofencing.

```
@Override
protected void onHandleIntent(Intent intent) {
    GeofencingEvent geofencingEvent = GeofencingEvent.fromIntent(intent);
    if (geofencingEvent.hasError()) {
        String errorMessage = GeofenceErrorMessages.getErrorString(this,
                geofencingEvent.getErrorCode());
        Log.e(TAG, errorMessage);
        return;
    }

    // Get the transition type.
    int geofenceTransition = geofencingEvent.getGeofenceTransition();

    // Test that the reported transition was of interest.
    if (geofenceTransition == Geofence.GEOFENCE_TRANSITION_ENTER ||
            geofenceTransition == Geofence.GEOFENCE_TRANSITION_EXIT) {

        // Get the geofences that were triggered. A single event can trigger multiple geofences.
        List<Geofence> triggeringGeofences = geofencingEvent.getTriggeringGeofences();

        // Get the transition details as a String.
        String geofenceTransitionDetails = getGeofenceTransitionDetails(
                this,
                geofenceTransition,
                triggeringGeofences
        );

        // Send notification and log the transition details.
        sendNotification(geofenceTransitionDetails);
        Log.i(TAG, geofenceTransitionDetails);
    } else {
        // Log the error.
        Log.e(TAG, getString(R.string.geofence_transition_invalid_type, geofenceTransition));
    }
}
```

Gambar III. 13 Source Code Geofencing

III.1.4.3.3 Source Code Radius Geofencing

Dibawah ini merupakan source code mekanisme untuk mengatur jarak radius pada mekanisme penerimaan geofencing dari perangkat lunak yang dibangun. Berikut adalah Gambar III.15 Source Code Geofencing.

```
// set final value listener
rangeSeekbar.setOnSeekbarFinalValueListener(new OnSeekbarFinalValueListener() {
    @Override
    public void finalValue(Number value) {
        String s_jarak = String.valueOf(value);
        double jarak = Double.valueOf(s_jarak);
        double jarak_km = jarak / 1000;
        nilai_meter = s_jarak;
        nilai_kilo = String.valueOf(jarak_km);
        Log.d("CRS=>", String.valueOf(nilai_kilo));
    }
});
```

Gambar III. 14 Source Code Geofencing

III.1.5 Analisi Kedekatan Jarak

Dengan menggunakan fitur geotagging, sistem kemudian akan mengambil informasi latitude dan longitude dari tempat pengambilan foto. Berikut ini adalah script untuk mengambil informasi latitude dan longitude dari geotagging.

```
exif = new ExifInterface(mCurrentPhotoPath);
tag_datetime = exif.getAttribute(ExifInterface.TAG_DATETIME);
String attrLATITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE);
String attrLATITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE_REF);
String attrLONGITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE);
String attrLONGITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE_REF);
if ((attrLATITUDE != null)
        && (attrLATITUDE_REF != null)
        && (attrLONGITUDE != null)
        && (attrLONGITUDE_REF != null)) {
    if (attrLATITUDE_REF.equals("N")) {
        latitude = convertToDegree(attrLATITUDE);
        tag_lat = String.valueOf(latitude);
    } else {
        latitude = 0 - convertToDegree(attrLATITUDE);
        tag_lat = String.valueOf(latitude);
    }

    if (attrLONGITUDE_REF.equals("E")) {
        longitude = convertToDegree(attrLONGITUDE);
        tag_long = String.valueOf(longitude);
    } else {
        longitude = 0 - convertToDegree(attrLONGITUDE);
        tag_long = String.valueOf(longitude);
    }
}
```

Gambar III. 15 Sourcode Informasi Geotagging

Setelah sistem mendapatkan data latitude dan longitude tempat pengambilan foto maka langkah selanjutnya adalah membandingkan antara kedua data lokasi tersebut yaitu lokasi tempat yang akan upload dengan lokasi tempat pengambilkan foto oleh pengguna. Dengan ketentuan di Tabel III.2 Kedektan Jarak dapat di lihat

persentase kecocokan pengguna saat upload foto dilokasi, jika kurang dari 90% pengguna tidak bisa menguplod foto.

Tabel III. 2 Kedekatan Jarak

NO	Keterangan	Interval (Meter)	Persentase Kedekatan	Status
1	Kedektan Jarak	100 M	100%	Bisa di Upload
2	Kedektan Jarak	100 M – 200 M	80%	Tidak bisa di Upload
3	Kedektan Jarak	200 M – 300 M	60%	Tidak bisa di Upload
4	Kedektan Jarak	300 M – 400M	40%	Tidak bisa di Upload
5	Kedektan Jarak	400M	20%	Tidak bisa di Upload

III.1.6 Analisis Fitur

Analisi fitur akan menjelaskan mengenai mekanisme dan cara kerja fitur-fitur yang mendukuang kinerja sistem snap place.

Cara medeteksi gambar yang diambil menggunkan kamera. Pengguna akan memilih salah satu objek yang akan diambil menggunakan kamera, dari hasil kamera diperoleh berupa gambar dan sistem akan mengambil data latitude dan longitude tempat pengambilan gambar tersebut.

Dengan menggunkan fitur geotagging, sistem kemudain akan mengambil informasi latitude dan longitude dari lokasi pengambilan gambar. Berikut adalah script untuk mengambil informasi latitude dan longitude dari geotagging.

```
exif = new ExifInterface(mCurrentPhotoPath);
tag_datetime = exif.getAttribute(ExifInterface.TAG_DATETIME);
String attrLATITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE);
String attrLATITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LATITUDE_REF);
String attrLONGITUDE = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE);
String attrLONGITUDE_REF = exif.getAttribute(ExifInterface.TAG_GPS_LONGITUDE_REF);
if ((attrLATITUDE != null)
        && (attrLATITUDE_REF != null)
        && (attrLONGITUDE != null)
        && (attrLONGITUDE_REF != null)) {
    if (attrLATITUDE_REF.equals("N")) {
        latitude = convertToDegree(attrLATITUDE);
        tag_lat = String.valueOf(latitude);
    } else {
        latitude = 0 - convertToDegree(attrLATITUDE);
        tag_lat = String.valueOf(latitude);
    }

    if (attrLONGITUDE_REF.equals("E")) {
        longitude = convertToDegree(attrLONGITUDE);
        tag_long = String.valueOf(longitude);
    } else {
        longitude = 0 - convertToDegree(attrLONGITUDE);
        tag_long = String.valueOf(longitude);
    }
}
```

Gambar III. 16 Source Code Geotangging

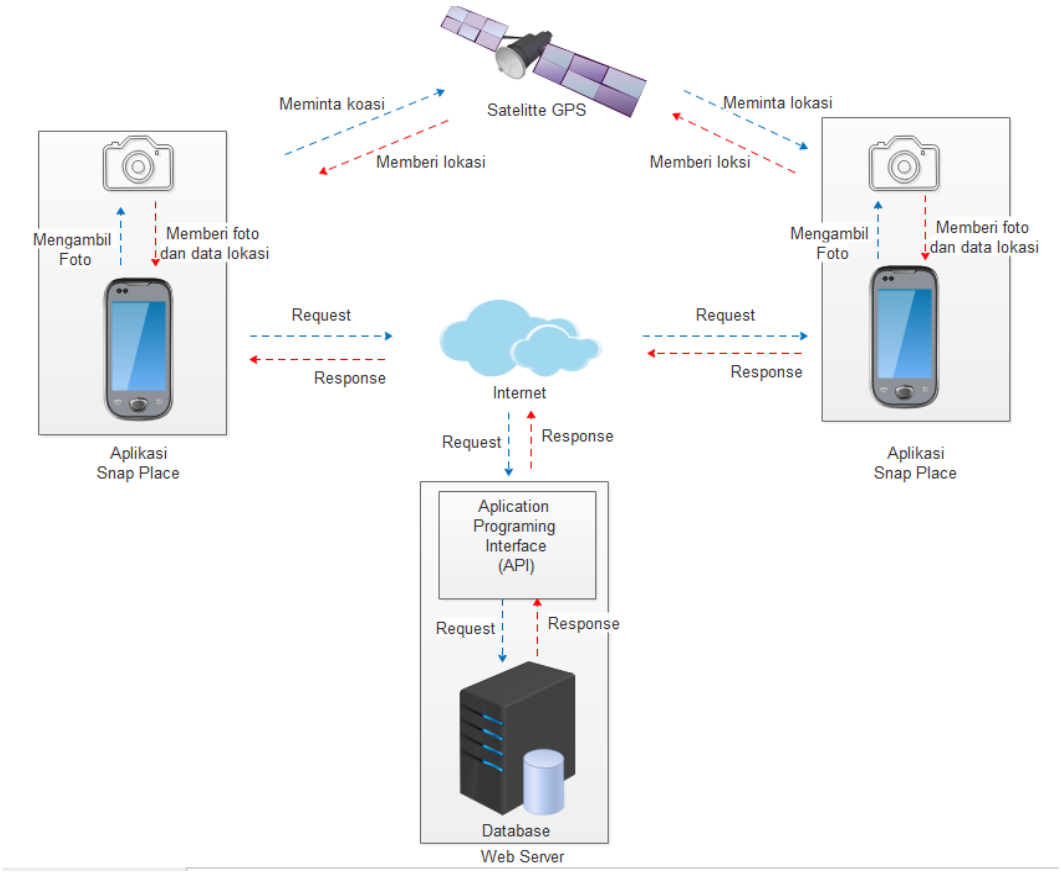
Setelah sistem mendapatkan data latitude dan longitude lokasi pengambilan gambar maka sistem akan mengetahui lokasi yang sedang dikunjungi pengguna.

III.1.7 Analisis Arsitektur Sistem

Analisis arsitektur sistem bertujuan untuk mengidentifikasi arsitektur yang akan dibangun berdasarkan mobile.

Mobile

Platform mobile adalah salah satu subsistem yang dipilih untuk pembangunan dari perangkat lunak ini. Arsitektur perangkat lunak pada platform mobile mengambarkan bagaimana perangkat lunak saling berinteraksi seperti Gambar III. 18 Arsitektur Perangkat Lunak pada Platform mobile. Gambar tersebut menggambarkan keseluruhan arsitektur sistem pada platform mobile.



Gambar III. 17 Arsitektur perangkat lunak pada platform mobile

Berikut adalah deskripsi dari Gambar III.18 Arsiktur Perangkat lunak pada platform mobile :

1. Perangkat mobile pengguna mengambil foto dari kamera.
2. Kamera melakukan request data lokasi ke satellite GPS.
3. GPS mengeirim data lokasi perangkat mobile pengguna.
4. Perangkat mobile pengguna melakukan request data ke server melalui API.
5. Server menerima permintaan data dari API kemudian server akan mengambil data yang ada di database.
6. Setelah server menerimah data yang diminta dari database,data tersebut akan dikembalikan dan diubah dalam bentuk JSON oleh API.
7. Data diambil oleh sistem mobile dalam bentuk JSON untuk diproses perangkat mobile pengguna.

III.1.8 Spesipikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak terbagi kedalam dua kebutuhan yaitu kebutuhan non fungsional dan fungsional. Kebutuhan fungsional dapat dilihat pada Tabel III.2 Spesipikasi Kebutuhan Fungsional.

Tabel III. 3 Spesifikasi Kebutuhan Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-F-001	Sistem mobile menyediakan fasilitas bagi pengguna untuk ambil gambar kedalam sistem
SKPL-F-002	Sistem mobile menyediakan fasilitas bagi pengguna untuk melihat hasil gambar
SKPL-F-003	Sistem mobile menyediakan fasilitas bagi pengguna untuk melihat deskripsi atau sejarah lokasi yang sudah di ambil menggunakan camera
SKPL-F-004	Sistem mobile menyediakan fasilitas bagi pengguna untuk upload photo ke dalam sistem
SKPL-F-005	Sistem mobile menyediakan fasilitas bagi pengguna untuk pencarian lokasi

III.1.9 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional pada sistem ini meliputi analisis perangkat lunak, analisis perangkat keras dan perangkat pikir.

1. Product Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *product requirement*.

a. SKPL-NF-001 Sistem yang dibangun memiliki dua subsistem yaitu web dan mobile.
b. SKPL-NF-002 Sistem yang dibangun minimal menggunakan sistem operasi android versi 5.0 Lolipop dan dibangun dengan dengan kebutuhan perangkat lunak minimum.
    a. Sistem Operasi (platform bebas)

b. Kode editor berupa sublime text 3.
c. Integrated development environment (IDE) berupa Android Studio.
d. Android Development Tool versi 22.1.6 sebagai plugins Android Studio.
e. Android Software Development Kit sebagai fremework android.

a. SKPL-NF-003 Sistem yang dibangun menggunakan perangkat kamera.
b. SKPL-NF-004 Sistem dibangun dengan spesifikasi hardware yang memenuhi standar minimum kebutuhan seperti:
    a. Komputer dengan Vga on board 512 MB.
    b. Layar dengan resolusi 1024x480.
    c. Komputer dengan ram 2 GB.
    d. Perangkat android dengan ram 512.
    e. Device android dengan resolusi 320x480.

2. Organisation Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *Organisational Requirement.*

a. SKPL-NF-005 Desain user interface pada perangkat lunak yang akan dibangun mengacu kepada design guidelines android dari google.

3. External Requirement

Berikut adalah kebutuhan non fungsional berdasarkan klasifikasi *External Requirement.*

a. SKPL-NF-006 Sistem tidak mengizinkan konten yang mengandung unsur sara, pornografi dan norma-norma kesusilaan.
b. SKPL-NF-007 Perangkat lunak yang dibangun harus menghasilkan format file standar (.json) yang bisa digunakan oleh pihak luar yang berkepentingan.

4. Analisis Pengguna

Analisis pengguna sistem dimaksudkan untuk mengetahui siapa saja pengguna yang terlibat dalam menjalankan sistem, sehingga dapat diketahui tingkat dan pemahaman pengguna.Wisatawan memiliki peran untuk menjelakan sistem dapat di lihat dari Tabel III.3 Analisi Pengguna.

Tabel III. 4 Analisis Pengguna

Pengguna	Wisatawan
Tanggung Jawab	Mengambil gamabar
Hak Akses	Mengambil gambar, melihat dekripsi, melihat rute, mengirim photo
Tingkat Keterampilan	a. Memiliki kemampuan dalam mengoperasikan *smartphone* b. Memiliki kemampuan dalam menggunakan internet
Pengalaman	a. Pernah mengakses internet b. Pernah mengoperasikan *smartphone*

Setelah dilakukan analisis, dapat disimpulkan bahwa pengguna pada sistem yang akan dibangun yaitu pemerintah dan masyarakat telah memenuhi syarat untuk menjalankan aplikasi yang akan dibangun ini.

III.1.10 Analisis Data

III.1.10.1 Stuktur Data JSON

Analisis data yang digunakan dalam penelitian ini adalah Object Relational Mapping (ORM). Object relational mapping melakukan pemetaan terhadap tabel-tabel pada basis data relasional dengan suatu class entitas yang ada pada bahasa pemrograman berorientasi objek.

Pemetaan yang dilakukan ORM akan membutuhkan suatu jembatan berupa format data JSON yang dapat menghubungkan ORM dengan database fisik yang ada pada server. Struktur format data JSON yang digunakan pada penelitian ini terbagi menjadi tiga yaitu data objek, data array dan data null.

1. Data Objek

Format pada Tabel III.4 Stuktur JSON data Objek digunakan ketika data yang diterima dari server merupakan data tunggal atau berupa satu objek.

Tabel III. 5 Stuktur JSON data Objek

Struktur data objek

{
“key” : “value”
}

2. Data Array

Format pada Tabel III.5 Struktur JSON data Array digunakan ketika data diterima dari server merupakan data tunggal atau berupa satu objek

Tabel III. 6 Struktur JSON data Array

Struktur data array

{
"result" : "value",
"item" : [{ "key" : "value" },{ "key" : "value" }]
}

3. Data Null

Format pada Tabel III.6 Struktur JSON data Null digunkan ketika data yang diterima dari server merupakan data tunggal atau berupa satu objek

Tabel III. 7 Struktur data Null

Struktur data null

{
"result" : "value",
"message" : "value",
"status_code":value
}

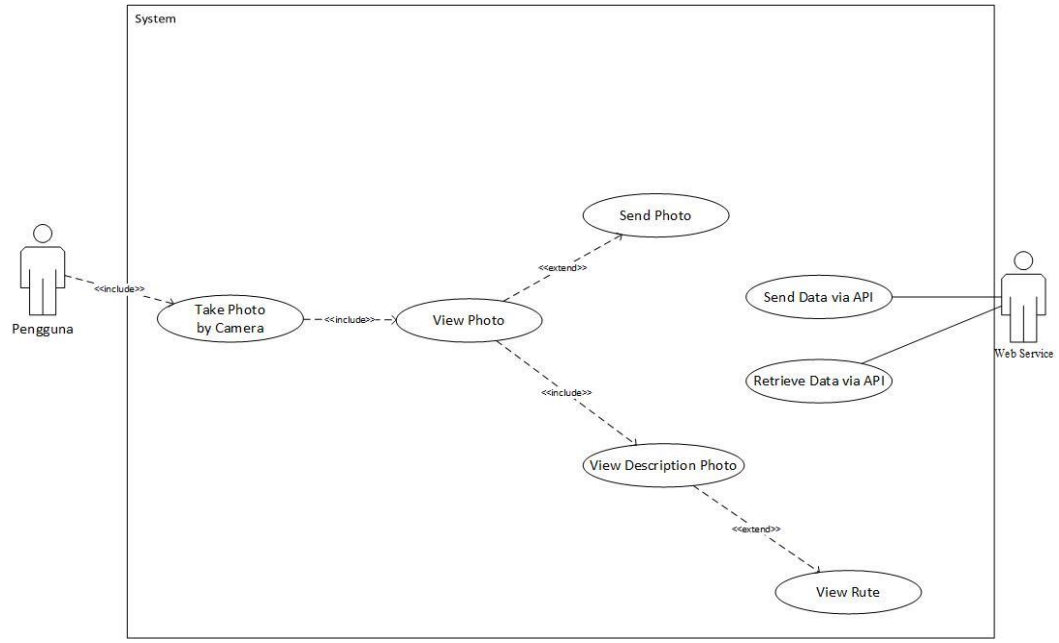
III.1.10.2 Analisi Kebutuhan Fungsional

Berikut ini adalah Analisis Kebutuhan Fungsional dari sistem yang akan dibangun.

III.1.10.3 Analisis Kebutuhan Fungsioanal Mobile

1. Diagram Use Case

Diagram Use Case merupakan pemodelan untuk menggambarkan kelakuan (behavior) perangkat lunak yang akan dibuat. Berikut adalah Gambar III.19. Diagram Use Case.Garis keterhubungan yang dimiliki use case Send data via API dan Retrieve data via API tidak digambarkan untuk mempermudah pembacaan diagram.



Gambar III. 18 Use Case Diagram

2. Definisi Aktor

Pada ini diisi dengan daftar actor dan deskripsi role untuk actor tersebut. Deskripsi role harus menjelaskan wewenang pada role tersebut dalam perangkat lunak. Bisa dibuat dalam bentuk tabel berikut :

Tabel III. 8 Definisi Aktor

No	Actor	Deskripsi
1	Pengguna	Actor dengan role ini memiliki wewenang untuk menggunakan fungsionalitas pada level user interface.
2	Web Service	Actor dengan role ini memiliki wewenang sebagai penyedia API / jembatan sumber data dan berjalan secara *background*.

3. Defenisi Use Case

Pada Tabel III.8 Definisi Use Case berisi daftar use case dan deskripsi singkat mengenai use case tersebut.

Tabel III. 9 Definisi Use Case

No	Use Case	Deskripsi
1	Take Photo by Camera	System memanggil fungsi kamera yanga da di device pengguna.
2	View Photo	System menampilkan gambar yang sudah di ambil menggunakan kamera.
3	View Description Photo	System menampilkan from detail yang sudah di ambil menggunkan snap place.
4	Send Photo	System dapat memasukan data kedalam database
5	View Rute	System menampilkan panduan untuk menuju lokasi.

4. Skenario Use Case

Skenario yang dibuat ditujukan untuk beberapa use case utama, yang menggambarkan urutan interaksi actor dengan use case tersebut, dari awal sampai akhir.

a. Skenario Use Case Take Photo

Berikut adalah Tabel III.9 Sekenario Use Case *Take Photo By Camera* yang menjelaskan skenario dari use case *Take Photo by Camera.*

Tabel III. 10 Skenario Use Case *Take Photo By Camera*

Use Case Name	Take Photo by Camera	
Related Requirements	-	
Goal Context	Melakukan pengambilan take photo	
Preconditions	Form catatan ditampilkan	
Successful End Condition	Photo berhasil diambil	
Failed End Condition	Photo tidak berhasil di ambil	
Primary Actor	Pengguna	
Trigger	Pengguna memilih diambil.	
Main Flow	Step	Action
	1	Sistem menampilkan activity camera
	2	Pengguna memilih take photo
	3	Sistem mengambil gambar dari camera
Extension	Step	Branching Action
	3.1	Sistem gagal mengambil gambar
	3.2	Sistem berhasil mengambil gambar

b. Skenario View Place

Berikut adalah Tabel III.10 Sekenario Use Case View Place yang menjelaskan skenario dari use case view place.

Tabel III. 11 Use Case Skenario *View Photo*

Use Case Name	View Photo
Related Requirements	-
Goal Context	Menampilkan data snap place
Preconditions	Activity utama perangkat lunak ditampilkan
Successful End Condition	Menampilkan *activity* snap place pengguna

Failed End Condition	*Activity* snap place pengguna gagal ditampilkan	
Primary Actor	Pengguna	
Trigger	Pengguna memilih menampilkan	
Main Flow	Step	Action
	1	Pengguna memilih view place
	2	Sistem mengambil data view place pengguna
	3	Activity place pengguna ditampilkan
Extension	Step	Branching Action
	2.1	Sistem gagal mengambil view snap place
	2.2	Sistem berhasil mengambil data view snap place

c. Skenario *View Description Photo*

Berikut adalah Tabel III.11 Skenario Use Case View Detail Place yang menjelaskan skenario dari use case *View Description Photo*

Tabel III. 12 Use Case Skenario *View Description Photo*

Use Case Name	View Description Photo	
Related Requirements	-	
Goal Context	Menampilkan Snap Place	
Preconditions	Activity detail snap place di tampilkan	
Successful End Condition	Menampilkan *activity* detail snap place pengguna	
Failed End Condition	*Activity* form snap place pengguna gagal ditampilkan	
Primary Actor	Pengguna	
Trigger	Pengguna memilih menampilkan	
Main Flow	Step	Action
	1	Pengguna memilih view snap place

	2	Sistem mengambil data detail view snap place
	3	Activity snap place pengguna ditampilkan
Extension	Step	Branching Action
	2.1	Sistem gagal menampilkan detail snap place
	2.2	Sistem berhasil menampilkan detail snap place

f. Skenario Send Photo

Berikut adalah Tabel III.12 Skenario Use Case *Send* Photo yang menejalaskan skenario dari use case *Send* Photo

Tabel III. 13 Skenario Use Case *Send* Photo

Use Case Name	Send Photo	
Related Requirements	-	
Goal Context	Melakukan penambahan data	
Preconditions	Sistem mendapat request upload data	
Successful End Condition	Upload berhasil ditambahkan	
Failed End Condition	Upload gagal ditambahkan	
Primary Actor	Pengguna	
Trigger	Pengguna memilih tombol upload	
Main Flow	Step	Action
	1	Sistem menampilkan from upload
	2	Pengguna memsukan data upload
	3	Sistem mengambil data upload
Extension	Step	Branching Action
	3.1	Sistem gagal menambah data upload
	3.2	Sistem berhasil menambah data upload

g. Skenario View Rute

Berikut Tabel III.13 Skenario Use Case *View Rute* adalah yang menejalaskan skenario dari use case *Rute.*

Tabel III. 14 Skenario Use Case *View Rute*

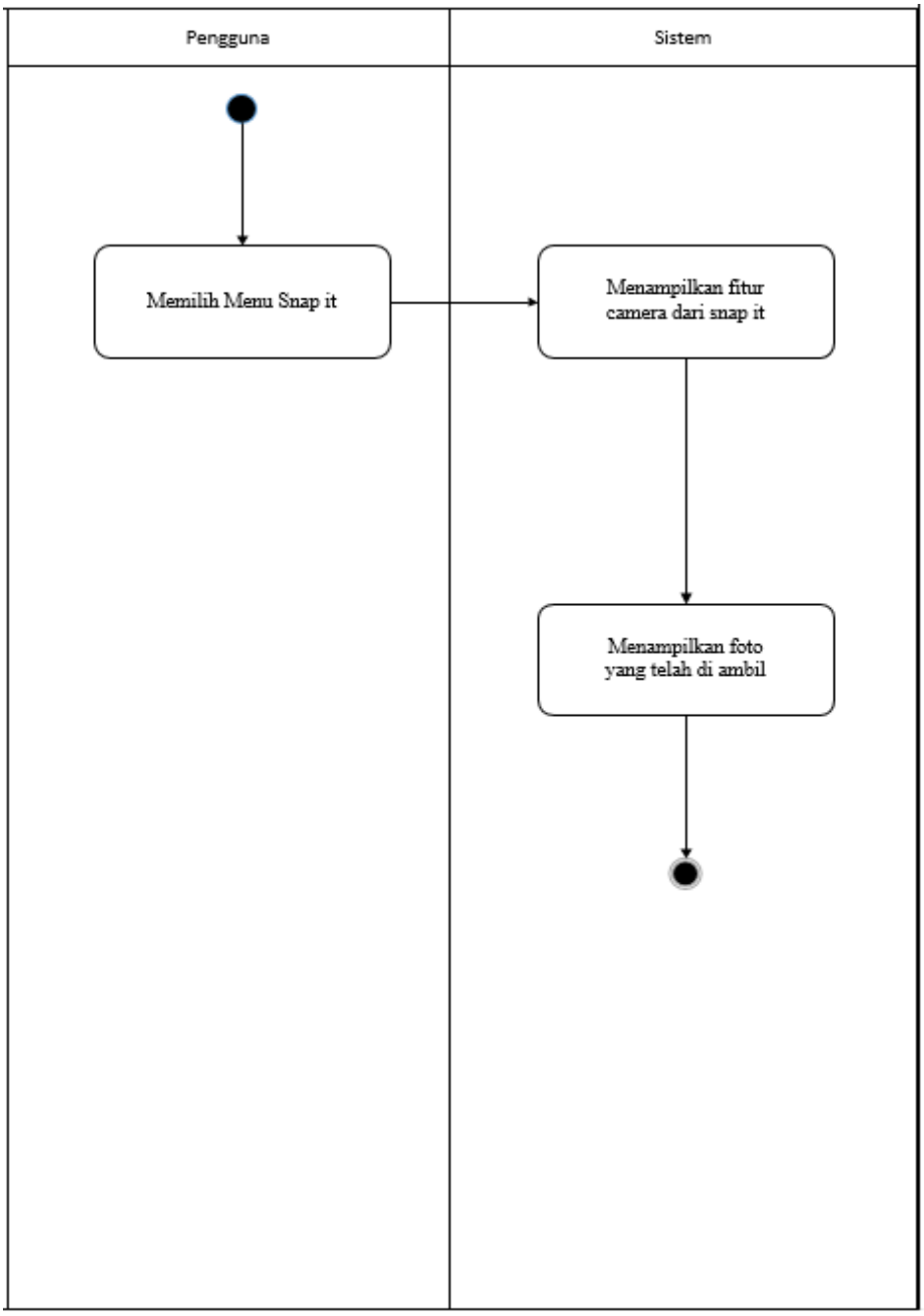
Use Case Name	View Rute	
Related Requirements	-	
Goal Context	Memilih rute plan dipilih pengguna	
Preconditions	Pengguna telah memilih rencana perjalanan	
Successful End Condition	Menampilakn rute berhasil	
Failed End Condition	Menampilkan pesan tidak berhasil	
Primary Actor	Pengguna	
Trigger	Pengguna memilih rute plan	
Main Flow	Step	Action
	1.	System menampilkan rencana perjalanan
	2.	Pengguna memilih rute rencana perjalanan
	3.	Sistem mengvalidasi masukan pengguna
Extension	Step	Branching Action
	3.1	Rute plan gagal
	3.2	Rute plan berhasil

5. Diagram Activity

Activity diagram menjelaskan alarian kerja aktor pada perangkat lunak yang akan dibangun

a. Activity Use Case *Take Photo by Camera*

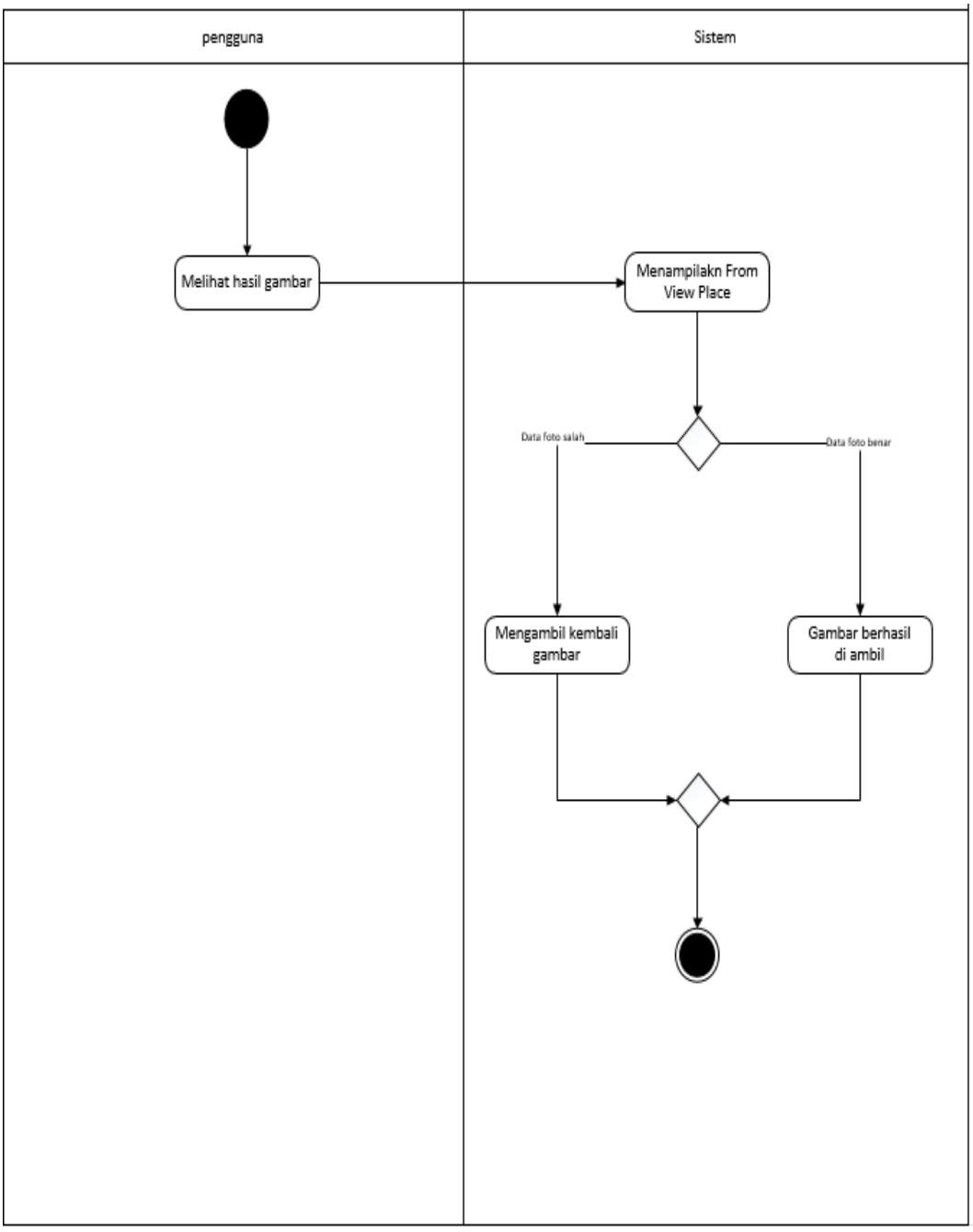
Berikut adalah Gambar III.20 Activity Diagram *Take Photo by Camera* yang menggambarkan activity diagram *take photo by camera*.



Gambar III. 19 Activity Take Photo by Camera

b. Activity View Photo

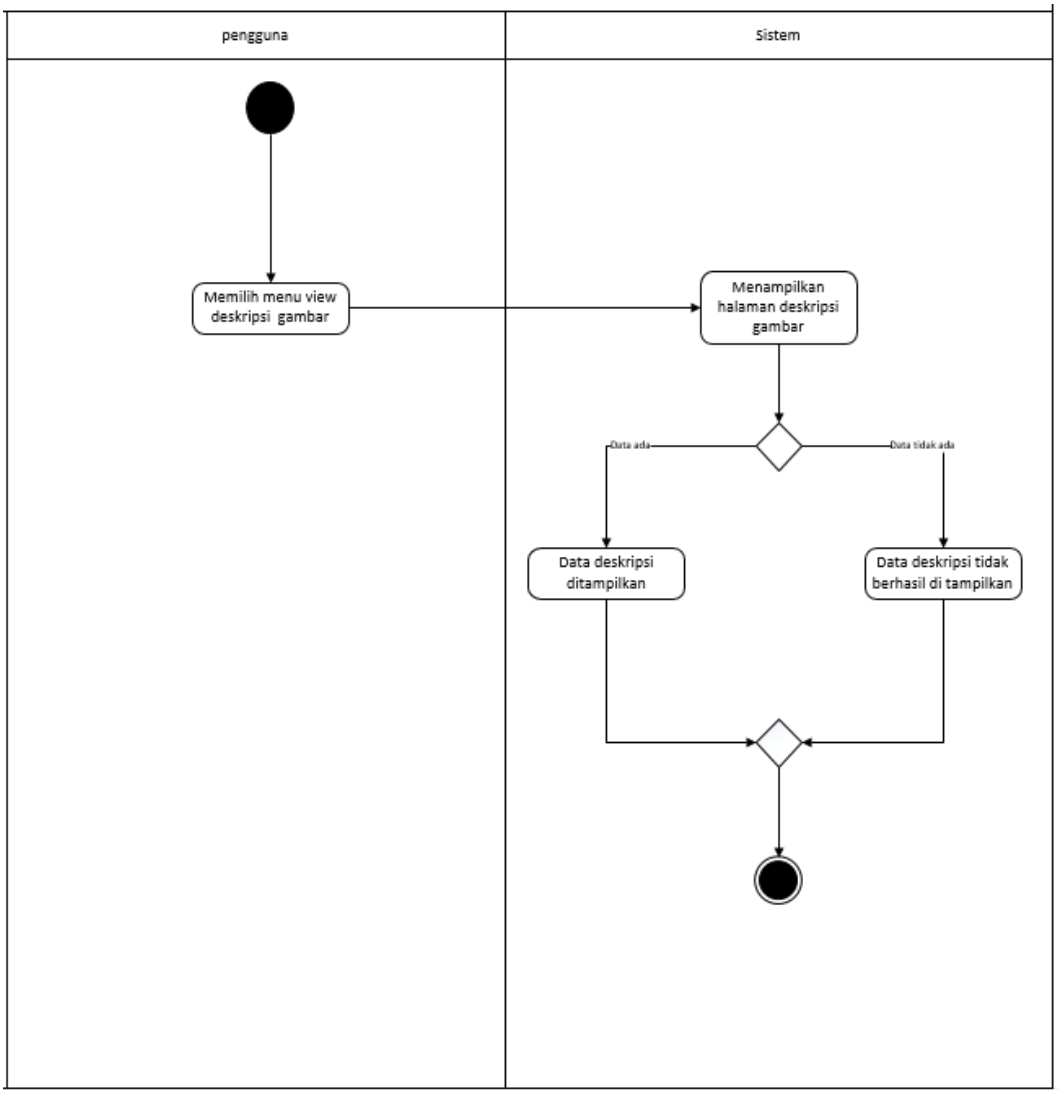
Berikut Gambar III.20 Activity Diagram View Photo yang menggambarkan activity diagram view place.



Gambar III. 20 Activity View Place

c. Activity View Description Photo

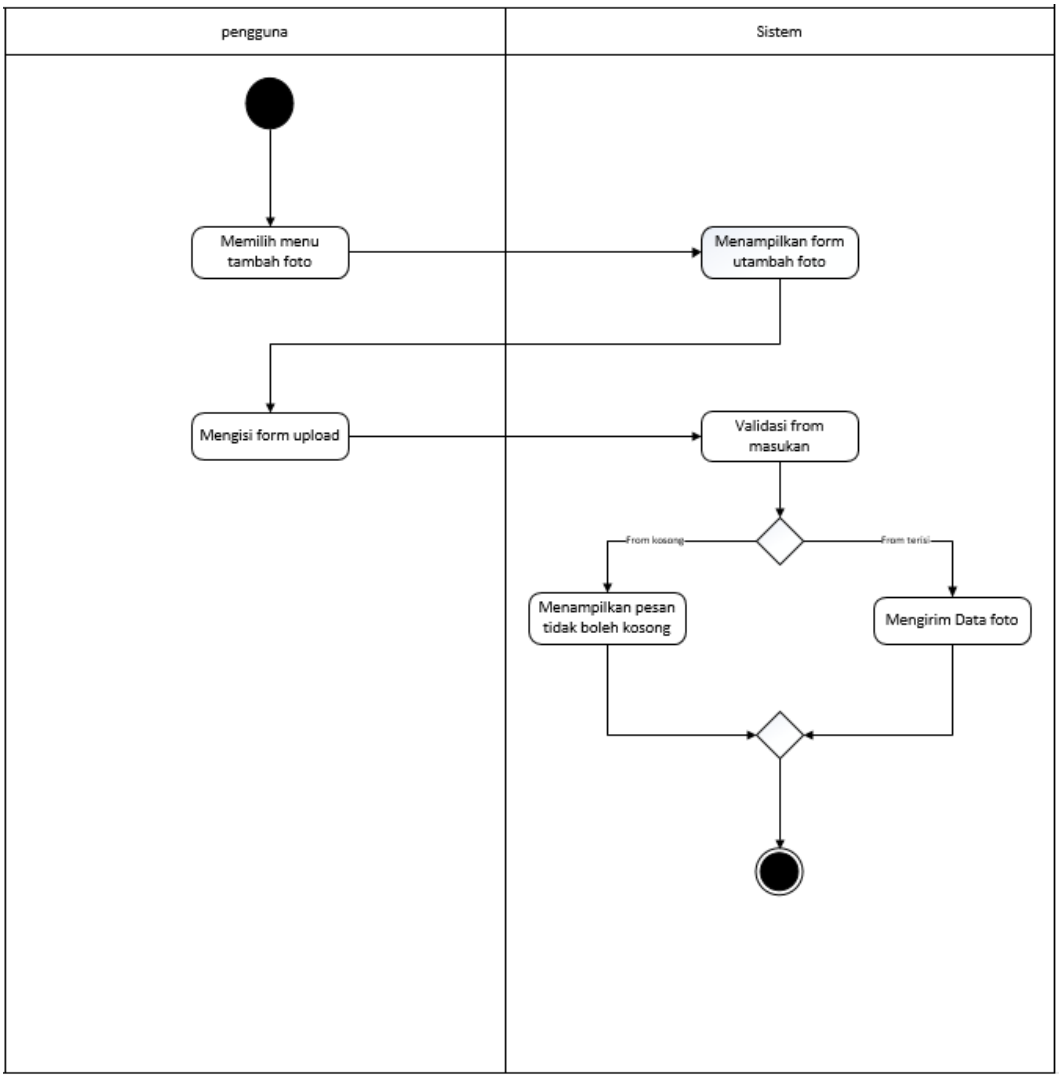
Berikut Gambar III.20 Activity Diagram View Description Photo yang menggambarkan diagram view description Photo.



Gambar III. 21 Activity View Description Photo

d. Activity Send Photo

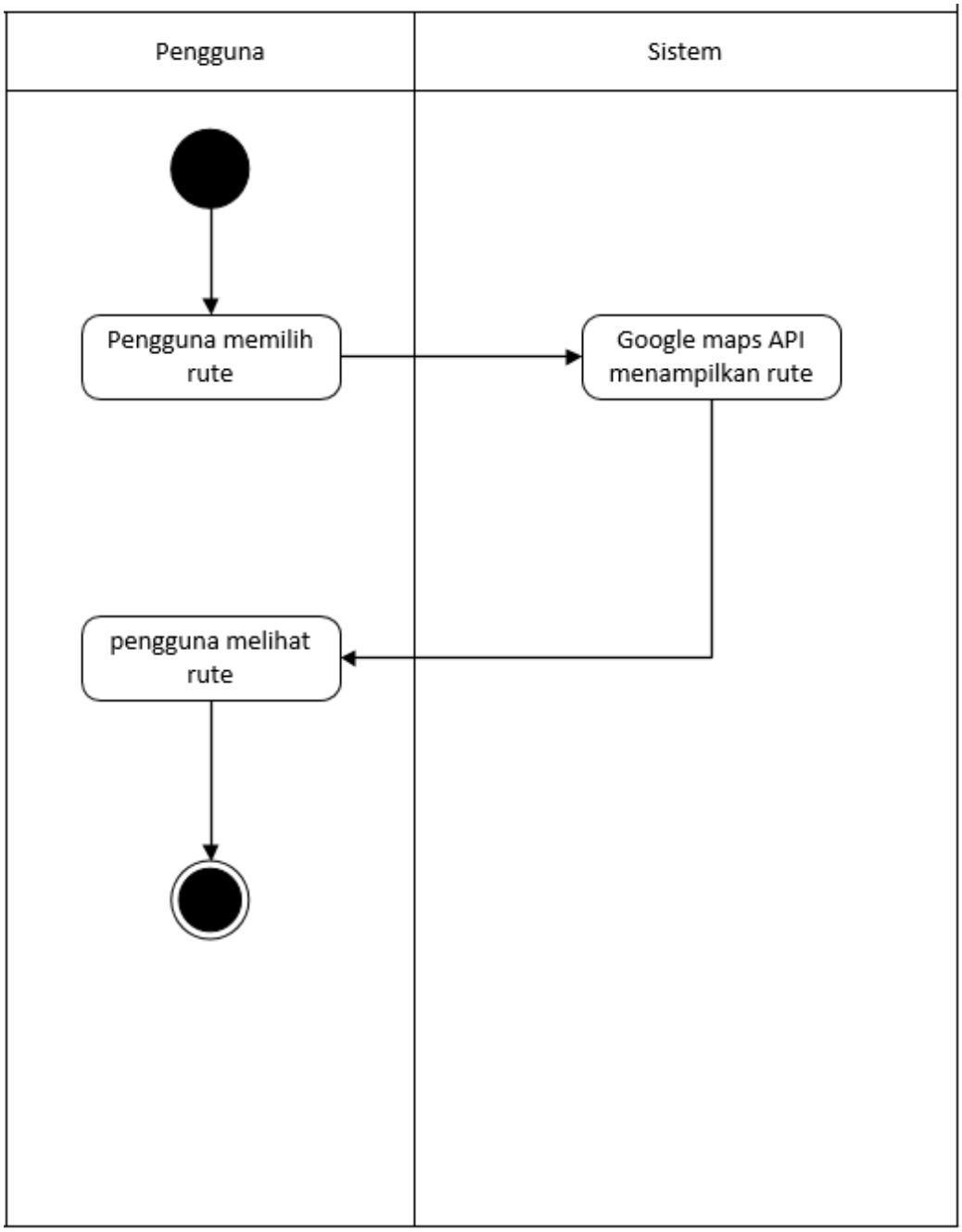
Berikut Gambar III.23 Activity Diagram Send Photo yang menggambarkan activity diagram upload.



Gambar III. 22 Activity Send Photo

e. Activity View Rute

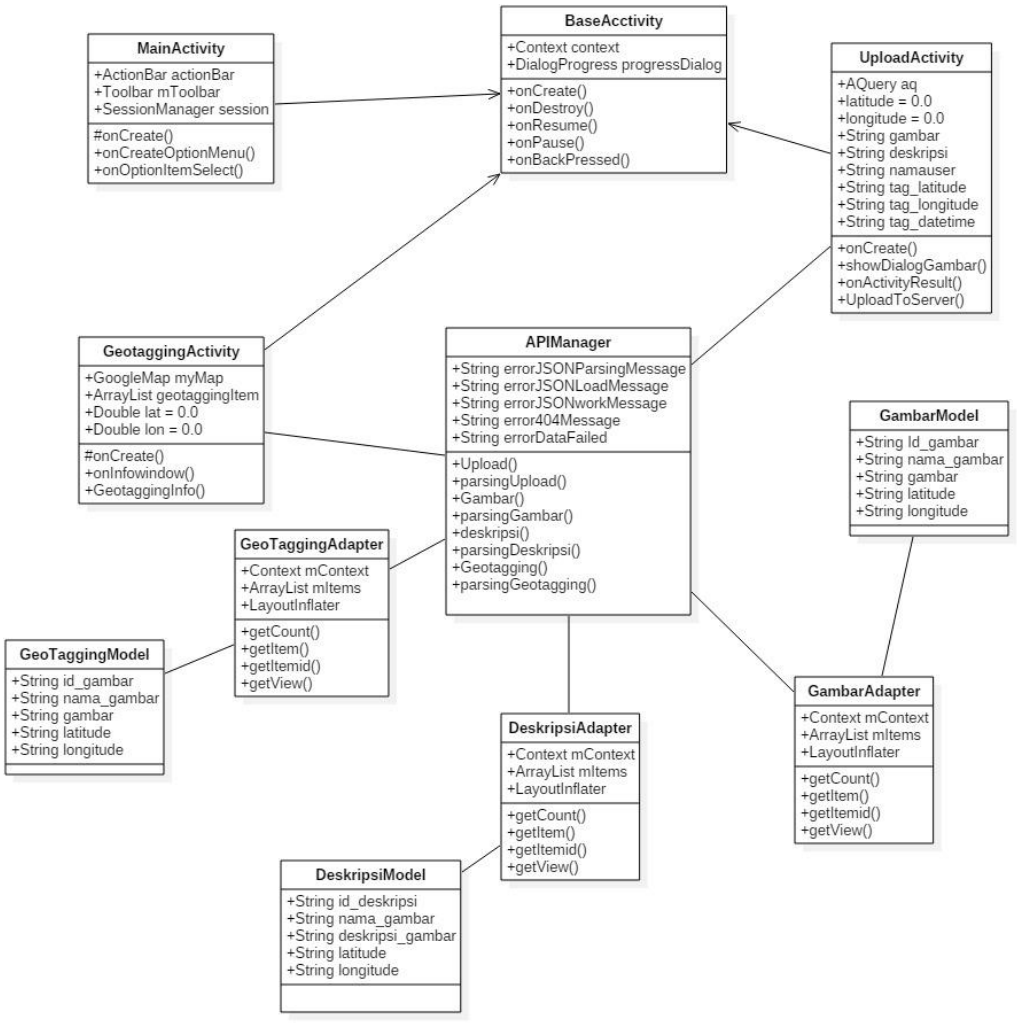
Berikut Gambar III.24 Activity Diagram View Rute yang menggambarkan activity diagram rute



Gambar III. 23 Activity View Rute

6. Diagram Class

Berikut adalah Gambar III.24 Calss Diagram yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu dengan yang lain.



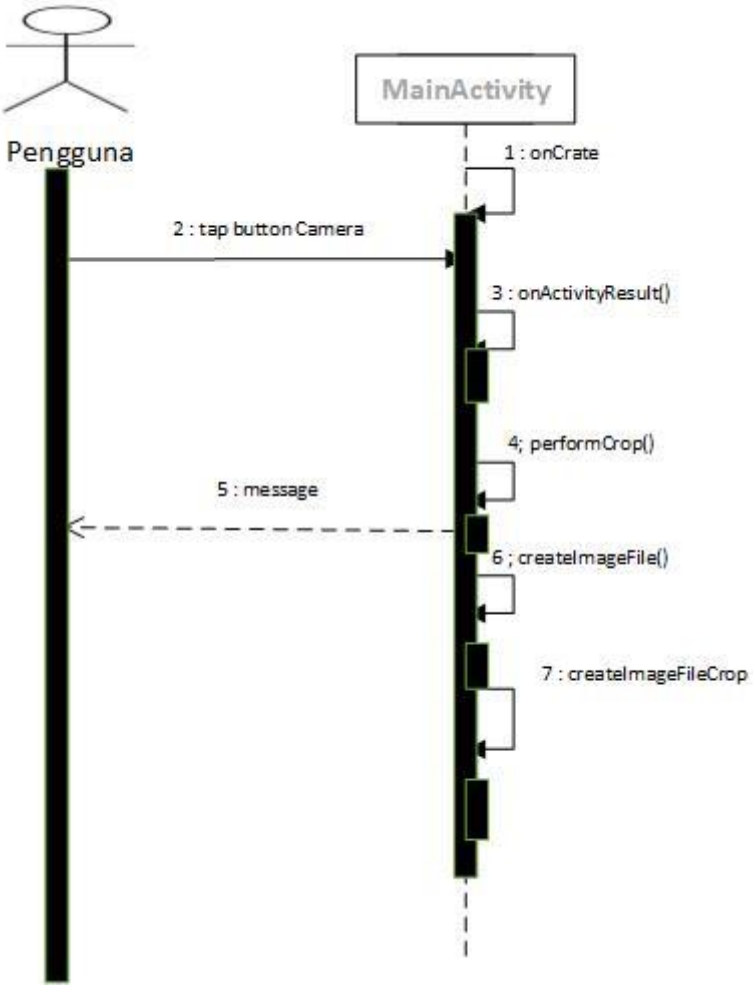
Gambar III. 24 Class Diagram

7. Diagram Sequence

Sequence diagram digunkan untuk menggambarkan arus pekerjaan, pesan yang disampaikan dan bagaimana elemen-elemen di dalamnya bekerja sama dari waktu ke waktu untuk mencapai suatu hasil.

a. Sequence Diargam Use Case Take Photo by Camera

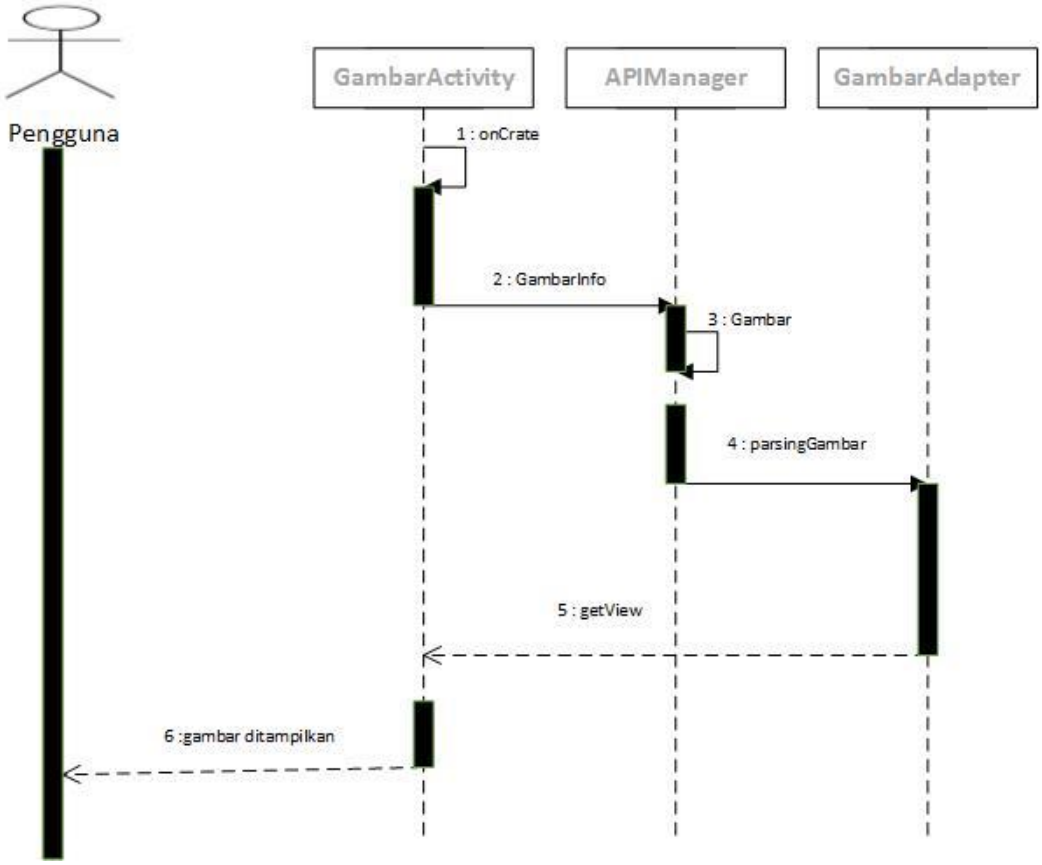
Berikut Gambar III.26 Sequence Take Photo by Camera adalah yang menggambarkan sequence diagram Take Photo by Camera.



Gambar III. 25 Sequence Take Photo by Camera

b. Sequence Diagram Use Case View Place

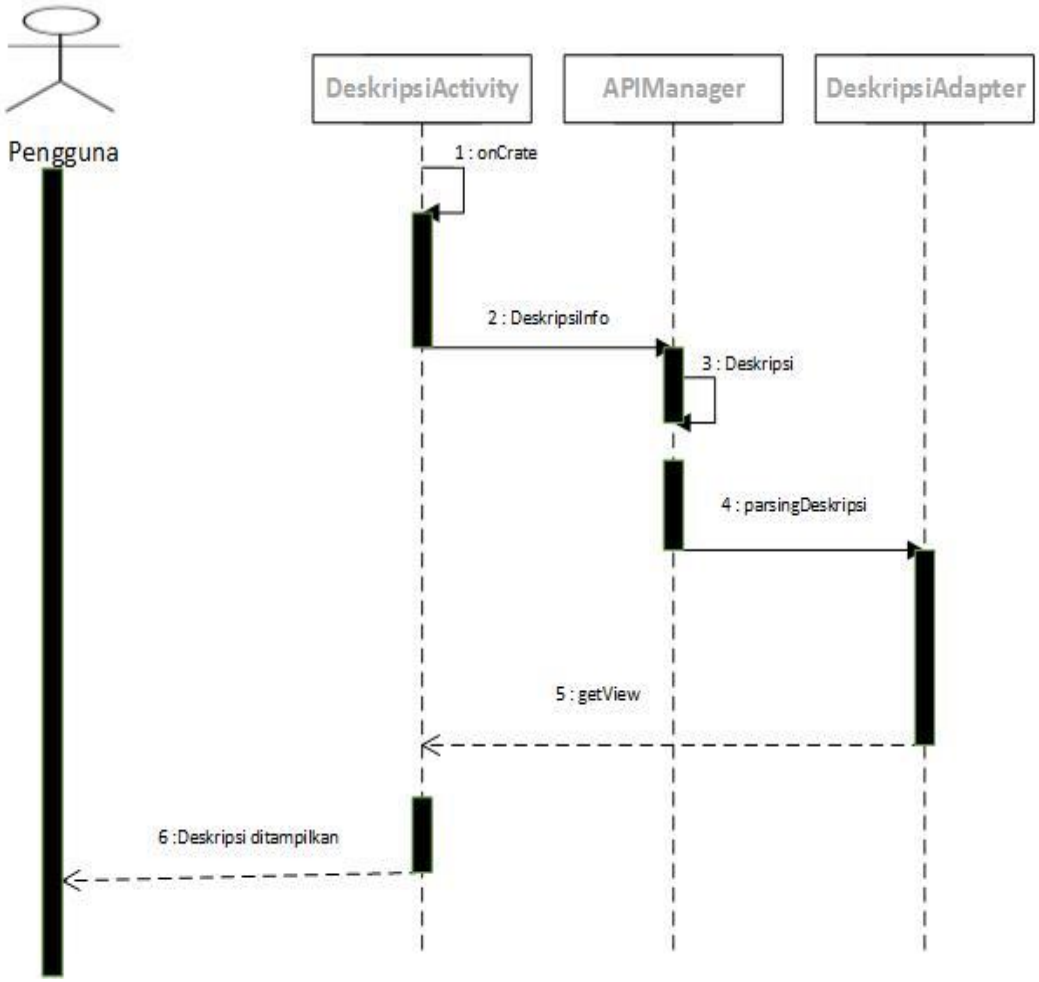
Berikut Gambar III.27 Use Case View Place adalah yang menggambarkan sequence diagram Use Case View Place.



Gambar III. 26 Sequence View Place

c. Sequence View Description Photo

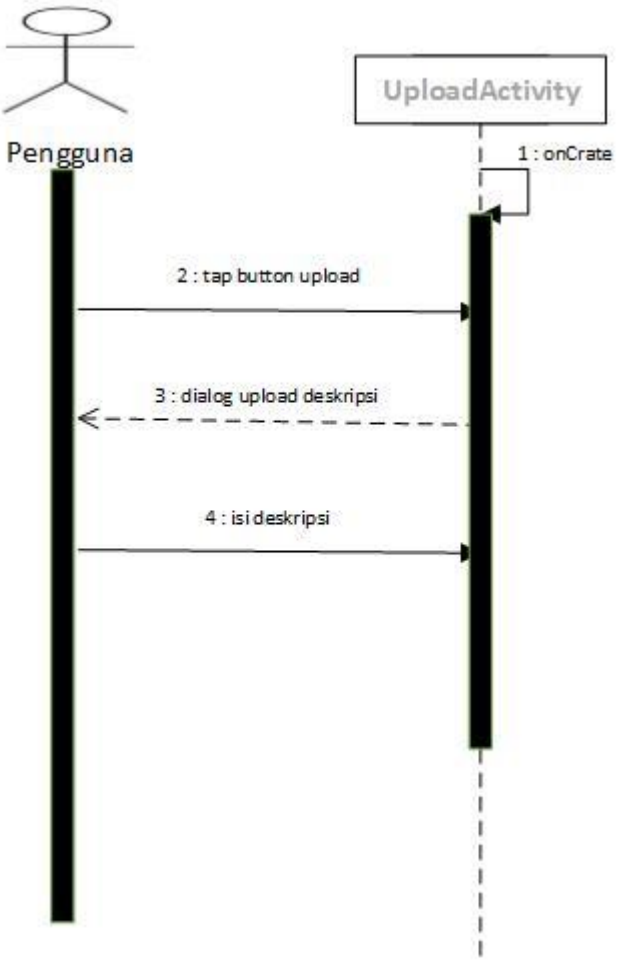
Berikut Gambar III.28 Sequence View Description Photo adalah yang menggambarkan sequence diagram Sequence View Description Photo.



Gambar III. 27 Sequence View Dscription Photo

d. Sequence Send Photo

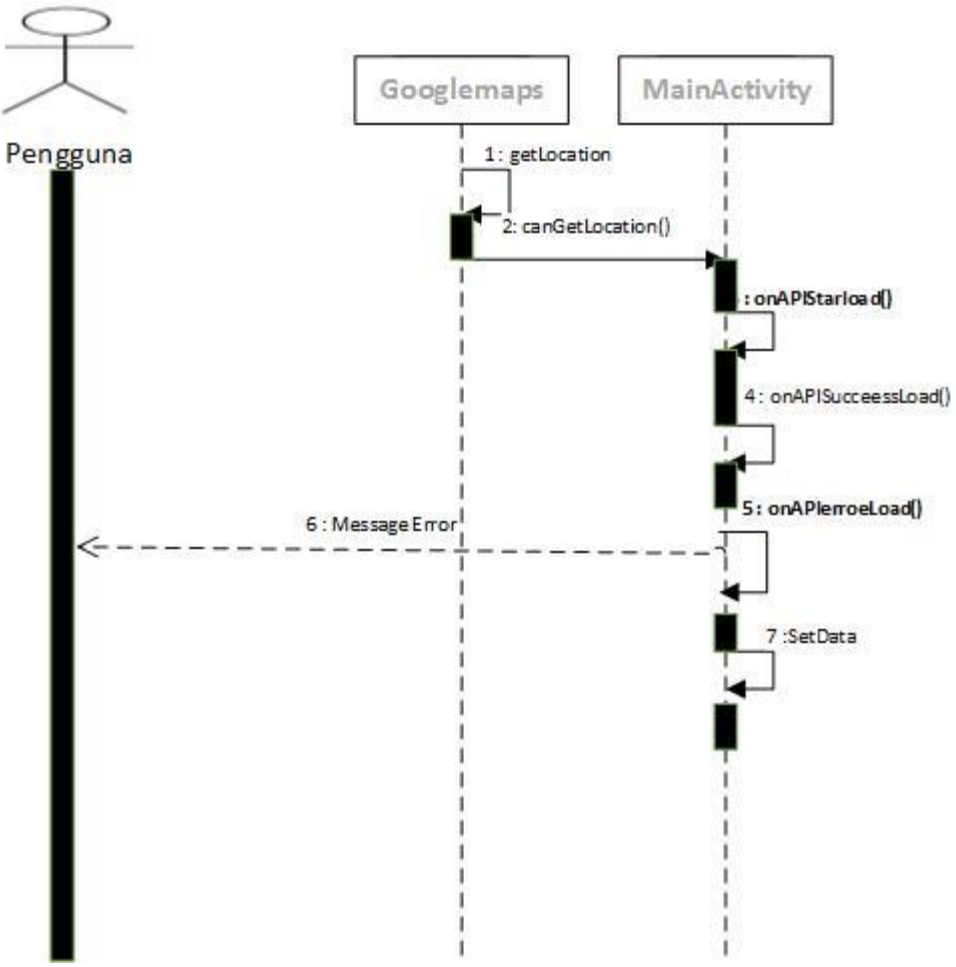
Berikut Gambar III.29 Sequence Send Photo adalah yang menggambarkan sequence diagram Sequence Send Photo.



Gambar III. 28 Sequence Send Photo

e. Sequnce View Rute

Berikut Gambar III.30 Sequence View Rute adalah yang menggambarkan sequence diagram Sequence View Rute.



Gambar III. 29 Sequence View Rute

III.2 Perancangan Sistem

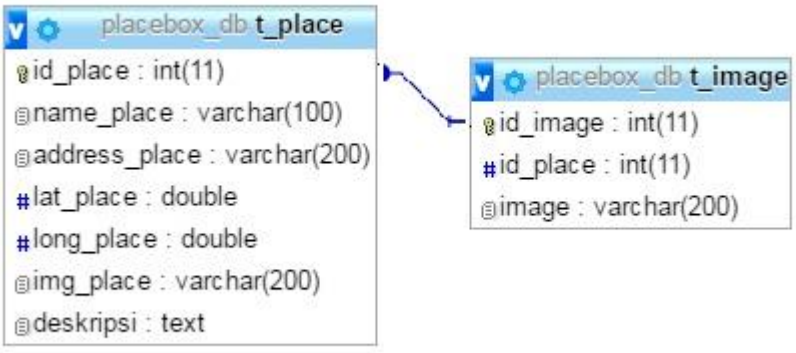
Perancangan sistem adalah penggambaran, perencanaan, dan pembuatan sketsa dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan memiliki fungsi.

III.2.1 Perancangan Data

Berikut adalah perancangan data dari perangkat lunak dibangun

1. Diagram Relasi

Brikut adalah Gambar III.31 Digram Relasi dari perangkat lunak yang dibangun.



Gambar III. 30 Diagram Relasi

2. Stuktur Tabel

Berikut adalah tabel dari perangkat lunak yang dibangun

a. Tabel t_ image.db

Berikut adalah Tabel III.14 t_image yang mendeskripsikan struktur tabel yang digunakan.

Nama file : t_image.db

Tempat penyimpanan : Hard Disk

Tabel III. 15 t_image

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id_image	int	11	primary key	not null, auto increment
Id_place	varchar	100		not null
gambar	varchar			not null

b. Tabel t_ place.db

Berikut adalah Tabel III. 15 t_place yang mendeskripsikan struktur tabel yang digunakan.

Nama file : t_place.db

Tempat penyimpanan : Hard Disk

Tabel III. 16 t_place

Nama Field	Tipe Data	Panjang	Kunci	Keterangan
id_place	int	11	primary key	not null, auto increment
nama_place	varchar	100		not null
Address_place	varchar	200		not null
lat_place	double			null
long_place	double			null
Img_place	varchar	200		not null
deskripsi	text			not null

III.2.2 Parancnagan Web Service Request dan Response

1. t_image

Berikut adalah Tabel III. 16 t_image yang mendeskripsikan rancangan request yang digunakan.

Tabel III. 17 t_image

Parameters	Contoh	Method
Id_image	8	GET
Id_place	8	

Request:
http://place-box.web.id/snaplace_api/snaplace_api_recentuploads.php

Response Berhasil:

Berikut adalah Tabel III. 17 Response Berhasil yang mendeskripsikan rancangan request berhasil yang digunakan.

Tabel III. 18 Response Berhasil


```
{
  "IMAGES": [
    {
      "id_place": "8",
      "id_image": "1",
      "image":     "https://upload.wikimedia.org/wikipedia/commons/f/fc/Tempat-Wisata-di-Bandung-Gambar-Gedung-Sate.jpg"
    },
    {
      "id_place": "8",
      "id_image": "2",
      "image": "http://www.jotravelguide.com/images/bdo-content-gdsate1.jpg"
    }
  ]
}
```

2. t_place

Berikut adalah Tabel III. 18 t_place yang mendeskripsikan rancangan request yang digunakan.

Tabel III. 19 t_place

Parameters	Contoh	Method
Latitude	-6.9025157	Get
Longitude	107.616588	

Request

http://place-box.web.id/snaplace_api/snaplace_api_listresult.php

Response Berhasil:

Berikut adalah Tabel III. 19 Response Berhasil yang mendeskripsikan rancangan request berhasil yang digunakan

Tabel III. 20 Response Berhasil


```
{
  "RESULT": [
    {
      "id_place": "8",
      "name_place": "Gedung Sate",
      "address_place": "Jalan Diponegoro No. 22, Citarum, Bandung Wetan, Citarum, Bandung Wetan, Kota Bandung, Jawa Barat 40115",
      "lat_place": "-6.9025157",
      "long_place": "107.616588",
      "img_place": "gedungsate.jpg",
      "deskripsi": "https://en.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Gedung%20Sate",
      "distance": "0.00"
```

```
    },
    {
      "id_place": "9",
      "name_place": "Braga Street",
      "address_place": "Jl. Braga, Sumur Bandung Kota Bandung Jawa Barat 40111",
      "lat_place": "-6.9182657",
      "long_place": "107.6073503",
      "img_place": "",
      "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Jalan%20Braga",
      "distance": "2.03"
    },
    {
      "id_place": "10",
      "name_place": "Monumen Perjuangan Rakyat Jawa Barat",
      "address_place": "Jl.Dipatiukur No.48,Lebakgede,Coblong, Kota Bandung,Jawa Barat 40132",
      "lat_place": "-6.9051974",
      "long_place": "107.6250667",
      "img_place": "",
      "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Monumen%20Perjuangan%20Rakyat%20Jawa%20Barat",
      "distance": "0.98"
    },
    {
      "id_place": "11",
      "name_place": "Museum Geologi",
```

```
    "address_place": "Jl. Diponegoro No.57,Cihaurgeulis, Cibeunying Kaler,Cihaur Geulis,Cibeunying Kaler,Kota Bandung, Jawa Barat 40122",
    "lat_place": "-6.9007084",
    "long_place": "107.6193019",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Museum_Geologi_Bandung",
    "distance": "0.36"
  },
  {
    "id_place": "12",
    "name_place": "Museum Konferensi Asia Afrika",
    "address_place": "Jalan Asia Afrika No.65,Braga,Sumur Bandung,Braga,Sumur Bandung,Kota Bandung, Jawa Barat 40111",
    "lat_place": "-6.9212134",
    "long_place": "107.6073265",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Museum_Konferensi_Asia_Afrika",
    "distance": "2.32"
  },
  {
    "id_place": "13",
    "name_place": "Museum Wangsit Mandala Sliwangi",
    "address_place": "Jl.Lembong No.38,Braga,Braga,Sumur Bandung,Kota Bandung,Jawa Barat 40181",
    "lat_place": "-6.9177146",
    "long_place": "107.6090696",
```

```
    "img_place": "",
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Museum_Wangsit_Mandala_Siliwangi",
    "distance": "1.88"
   },
   {
    "id_place": "14",
    "name_place": "Museum Pos",
    "address_place": "Jalan Cilaki No.73,Citarum,Bandung Wetan, Kota Bandung,Jawa Barat 40115",
    "lat_place": "-6.9023181",
    "long_place": "107.6177791",
    "img_place": "",
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Museum_Pos",
    "distance": "0.13"
   },
   {
    "id_place": "15",
    "name_place": "Mesjid Raya Bandung",
    "address_place": "Jl. Asia Afrika,Balonggede,Regol,Kota Bandung,Jawa Barat 40251",
    "lat_place": "-6.9217568",
    "long_place": "107.6043132",
    "img_place": "",
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Masjid_Raya_Bandung",
```

```
    "distance": "2.53"
   },
   {
    "id_place": "16",
    "name_place": "Museum Sri Baduga",
    "address_place": "Jl.BKR No.185,Pelindung Hewan,Astanaanyar, Kota Bandung,Jawa Barat 40243",
    "lat_place": "-6.9377645",
    "long_place": "107.6012731",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Museum_Sri_Baduga",
    "distance": "4.27"
   },
   {
    "id_place": "17",
    "name_place": "Taman Lansia",
    "address_place": "Jl.Cisangkuy,Citarum,Bandung Wetan,Kota Bandung,Jawa Barat 40123",
    "lat_place": "-6.9016497",
    "long_place": "107.6183253",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Taman_Lansia_Bandung",
    "distance": "0.21"
   },
   {
    "id_place": "18",
    "name_place": "Cihampelas Walk",
```

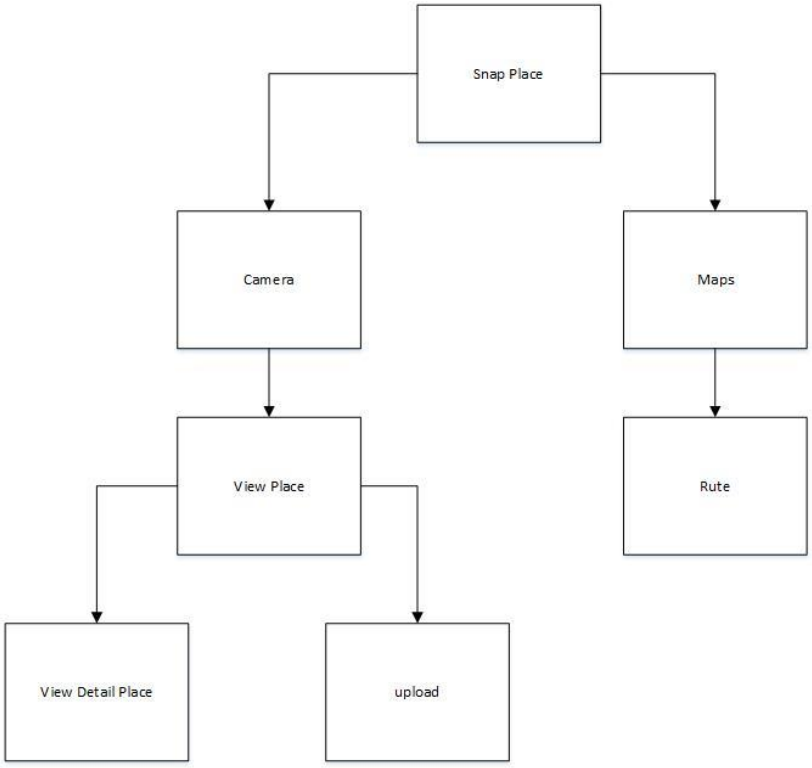
```
    "address_place": "Jl.Cihampelas No.160 Cipaganti Coblong\r\nKota Bandung,Jawa Barat 40131",
    "lat_place": "-6.8933561",
    "long_place": "107.6033405",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&indexpageids=1&action=query&prop=extracts&exintro=&explaintext=&titles=Cihampelas_Walk",
    "distance": "1.78"
  },
  {
    "id_place": "21",
    "name_place": "Saung Angklung Udjo",
    "address_place": "Jl.Padasuka No.118,Pasirlayung, Cibeunying Kidul,Kota Bandung,Jawa Barat 40192",
    "lat_place": "-6.898032",
    "long_place": "107.6527763",
    "img_place": "",
    "deskripsi": "https://id.wikipedia.org/w/api.php?format=json&action=query&prop=extracts&exintro=&explaintext=&titles=Saung_Angklung_Udjo",
    "distance": "4.03"
  },
  {
    "id_place": "22",
    "name_place": "Gedung Merdeka Bandung",
    "address_place": "Jalan Asia Afrika, Braga, Sumur Bandung, Kota Bandung, Jawa Barat 40111",
    "lat_place": "-6.9211169",
    "long_place": "107.6070245",
    "img_place": "",
```

```
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&action=query&prop=extracts
&exintro=&explaintext=&titles=Gedung_Merdeka",
    "distance": "2.32"
   },
   {
    "id_place": "23",
    "name_place": "Taman Film",
    "address_place": "Jalan Layang Pasupati, Bandung Wetan, Tamansari,
Bandung Wetan, Kota Bandung, Jawa Barat 40116",
    "lat_place": "-6.898574",
    "long_place": "107.6054563",
    "img_place": "",
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&action=query&prop=extracts
&exintro=&explaintext=&titles=Taman_Film",
    "distance": "1.30"
   },
   {
    "id_place": "24",
    "name_place": "Trans Studio Bandung",
    "address_place": "Trans Studio Mall, Jalan Jendral Gatot Subroto No.289,
Cibangkong, Kota Bandung, Jawa Barat 40273",
    "lat_place": "-6.9256374",
    "long_place": "107.6342151",
    "img_place": "",
    "deskripsi":
"https://id.wikipedia.org/w/api.php?format=json&action=query&prop=extracts
&exintro=&explaintext=&titles=Trans_Studio_Bandung",
    "distance": "3.22"
   }
```

```
  ]
}
```

III.2.3 Perancangan Arsitektur Menu

Perancangan arsitektur menyediakan software engineer satu gambaran dari stuktur menu program atau blueprint dari perangkat lunak yang akan dibuat. Tujuan perancangan ini adalah untuk membangun struktur program secara moduler dan menggambarkan hubungan kendali diantara modul program. Berikut adalah Gambar III.32 Perancangan Menu.



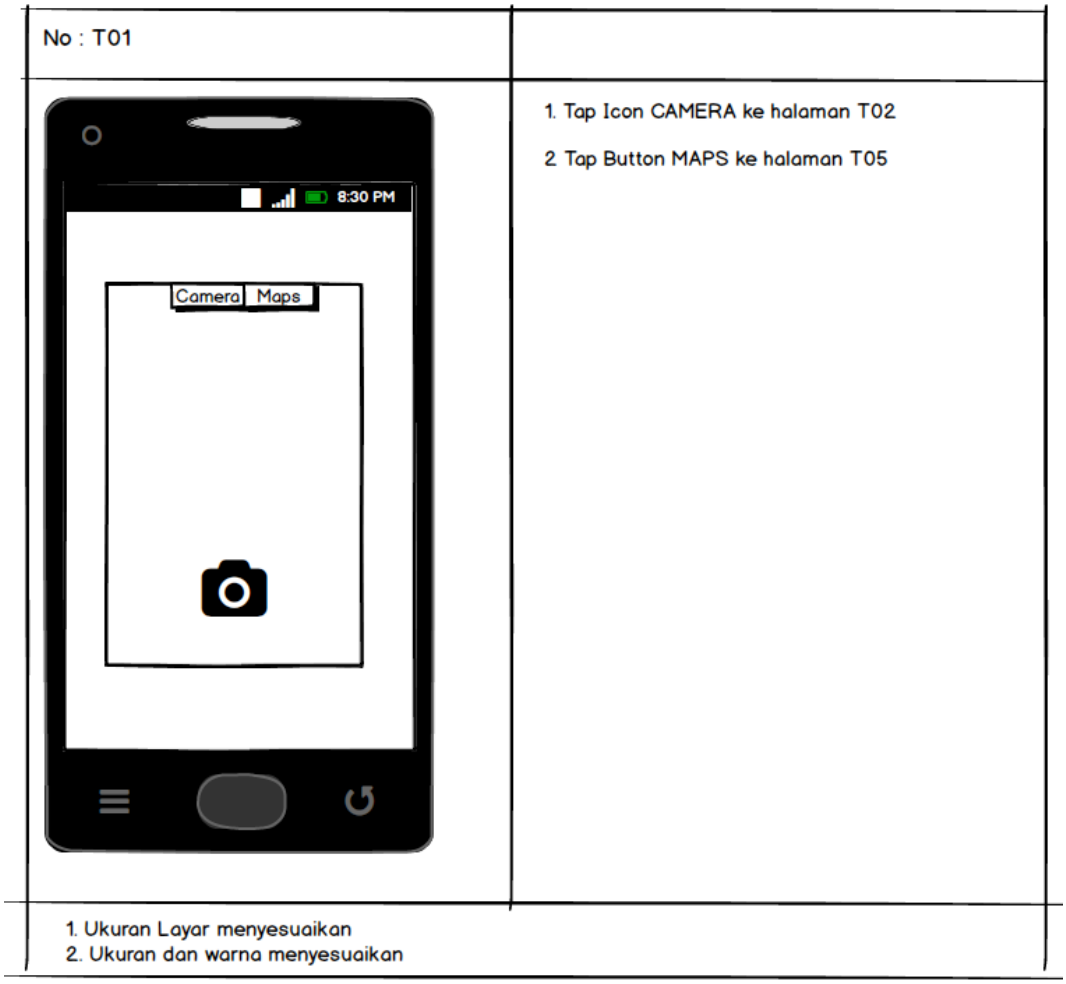
Gambar III. 31 Perancangan Menu

III.2.4 Perancangan Antar Muka

1. Perancangan Antarmuka Mobile

a. Perancangan Antarmuka Camera

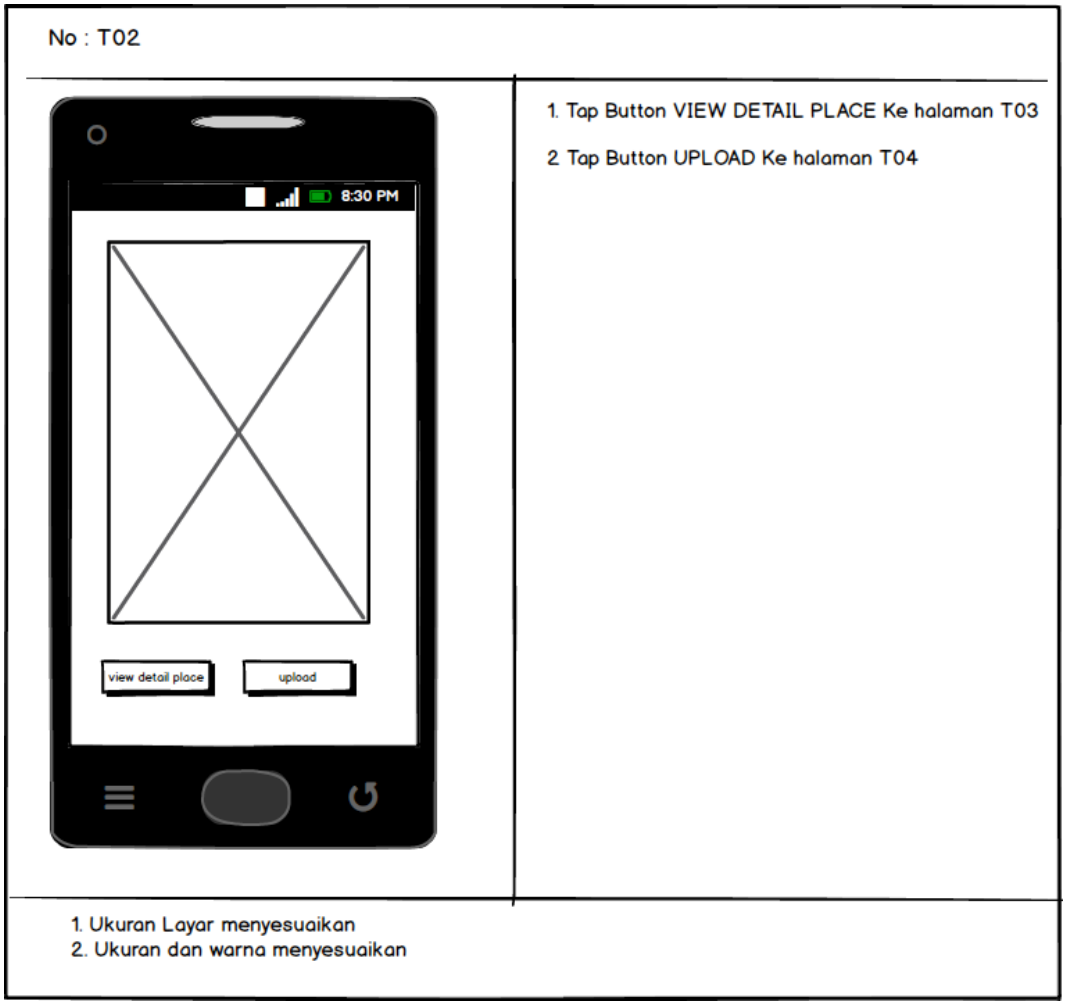
Berikut adalah Gambar III.33 Antarmuka Camera yang menggambarkan rancangan antarmuka Camera.



Gambar III. 32 Antarmuka Camera

b. Perancangan Antarmuka View Place

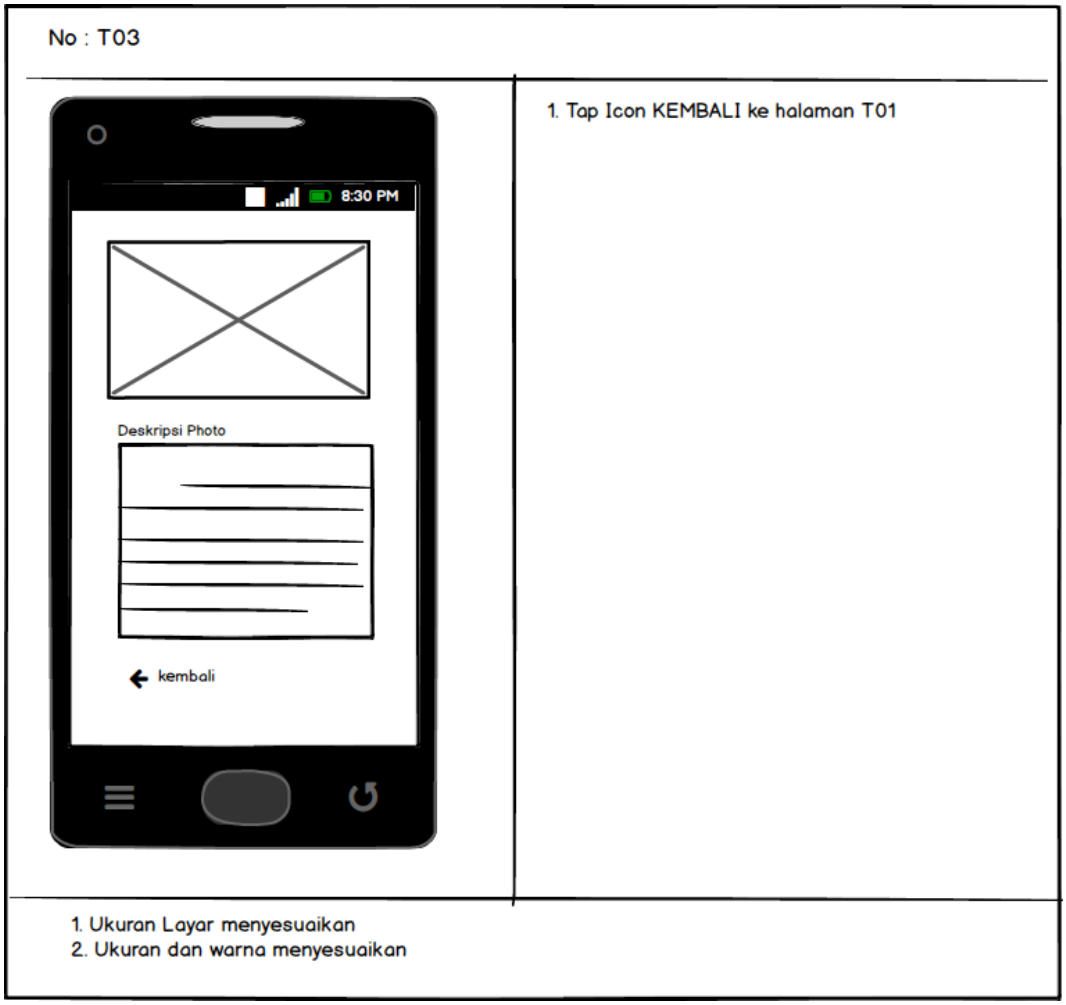
Berikut Gambar III.34 Antarmuka View Place yang menggambarkan rancangan anatarmuka View Place



Gambar III. 33 Antarmuka View Place

c. Perancangan Antarmuka Detail View Place

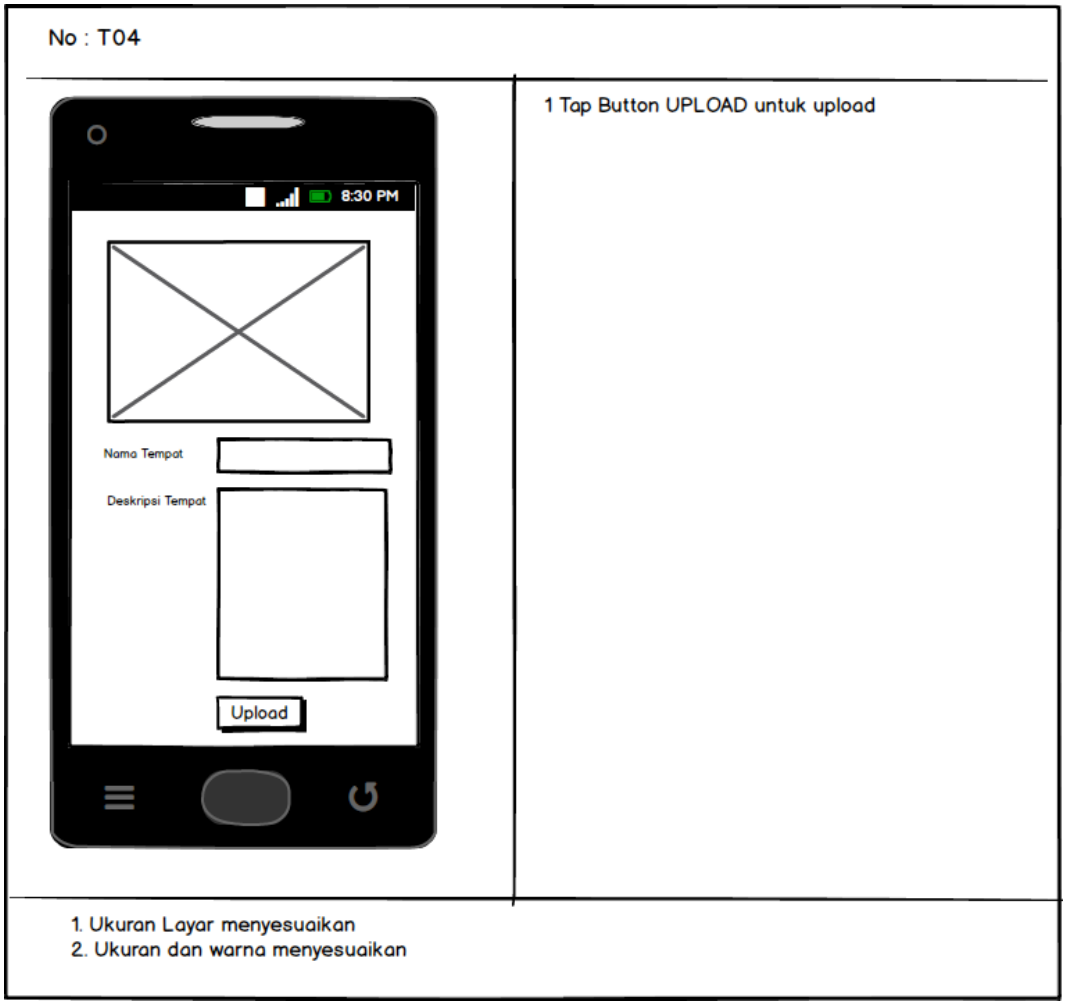
Berikut Gambar III.34 Antarmuka View Description Place yang mengambarkan rancangan antarmuka View description Place



Gambar III. 34 Antarmuka View Description Place

d. Perancangan Antarmuka Send Photo

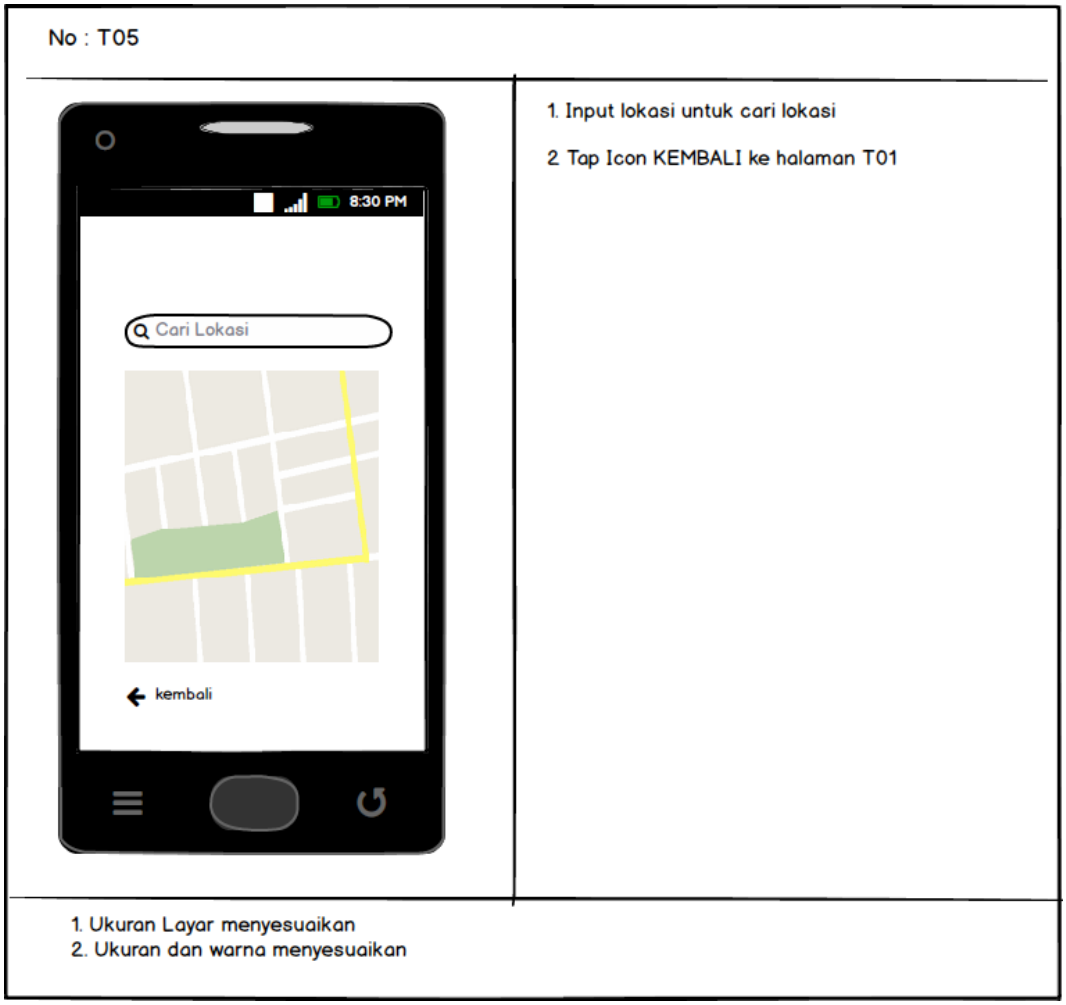
Berikut Gambar III.36 Antarmuka Send Photo yang mengambarkan rancangan antarmuka Send Photo



Gambar III. 35 Antarmuka Send Photo

e. Perancangan Antarmuka View Rute

Berikut adalah Gambar III.37 Antarmuka View Rute yang menggambarkan rancangan antarmuka View Rute



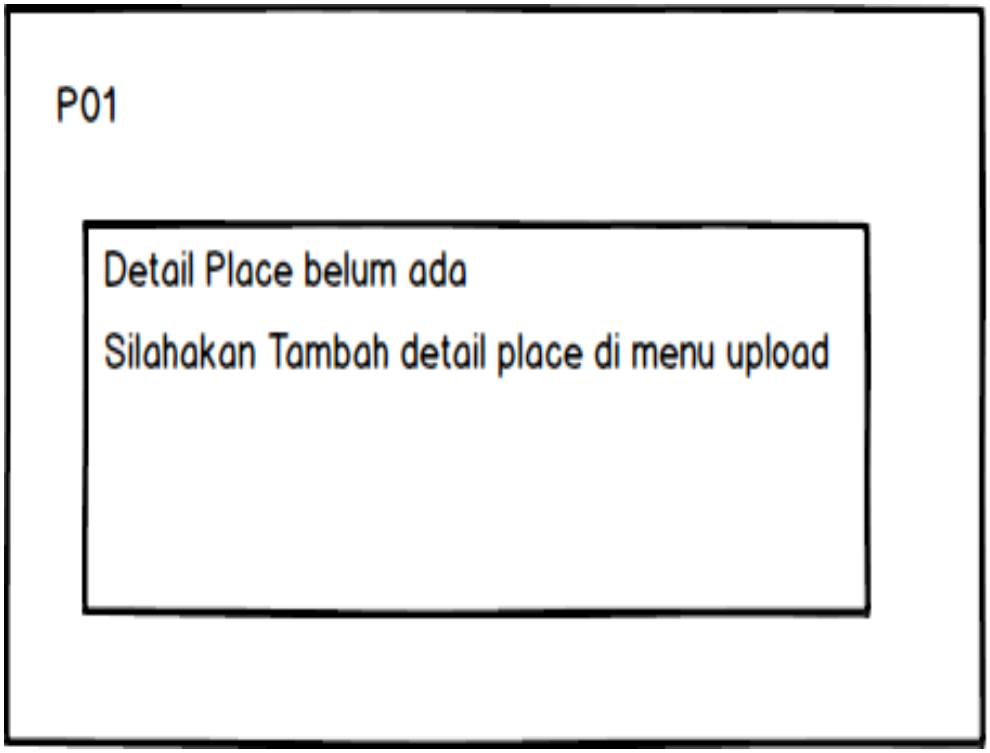
Gambar III. 36 Antarmuka View Rute

III.2.4.1 Perancangan Pesan

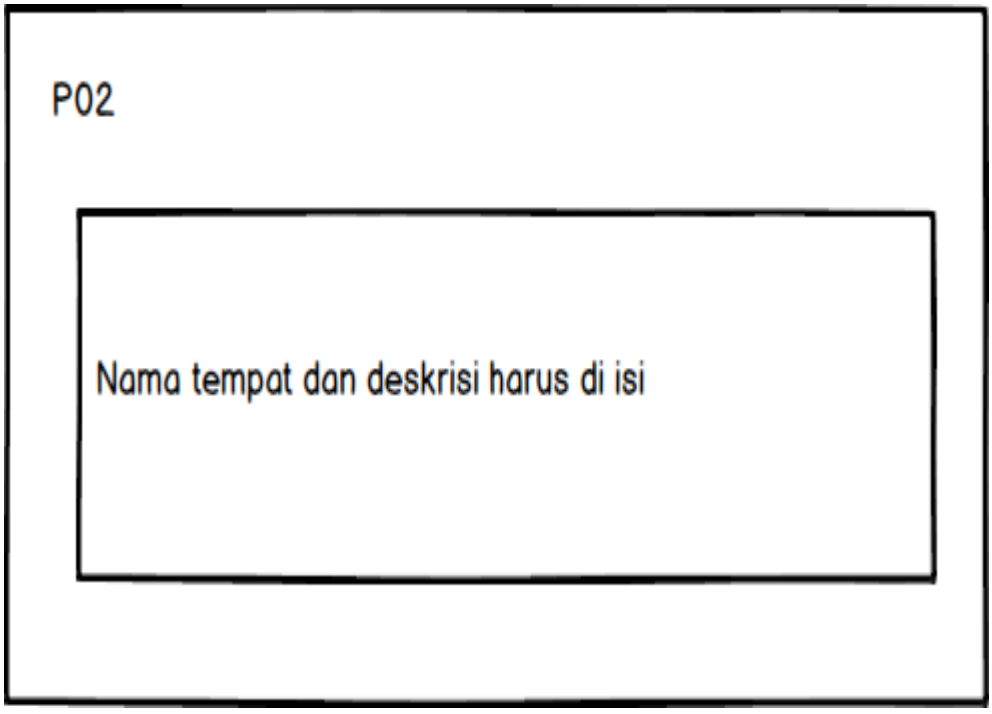
Perancanagn pesan pada perangkat lunk dibangun di peracangan pesan mobile

1. Prancangan Pesan Mobile

Berikut adalah perancangan mobile pada perangkat lunak yang di bangun



Gambar III. 37 Perancangan Pesan P01



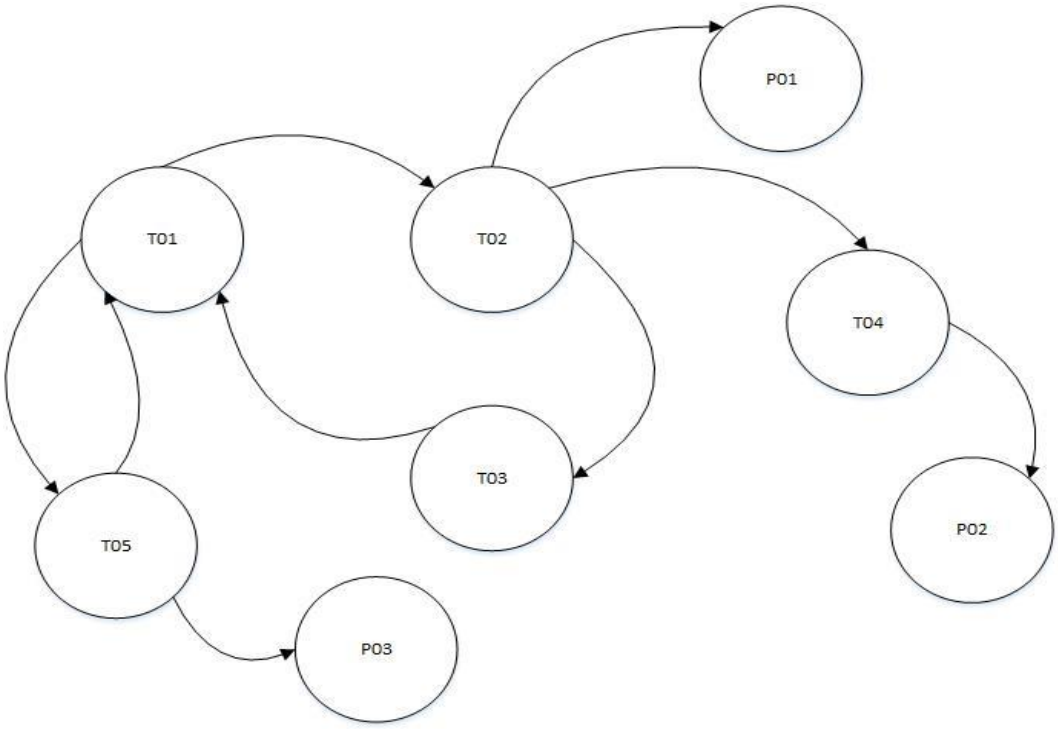
Gambar III. 38 Perancangan Pesan P02



Gambar III. 39 Perancangan Pesan P03

III.2.5 Jaringan Semantik

Jaringan Semantik merupkan gambaran pengetahuan grafis yang menunjukan hubungan antar objek (lingkaran) dan garis yang menggambarkan formasi antar objek tersebut. Adapun jaringan semantik untuk peangkat yang dibangun sebagai berikut.



Gambar III. 40 Jaringan Semantik