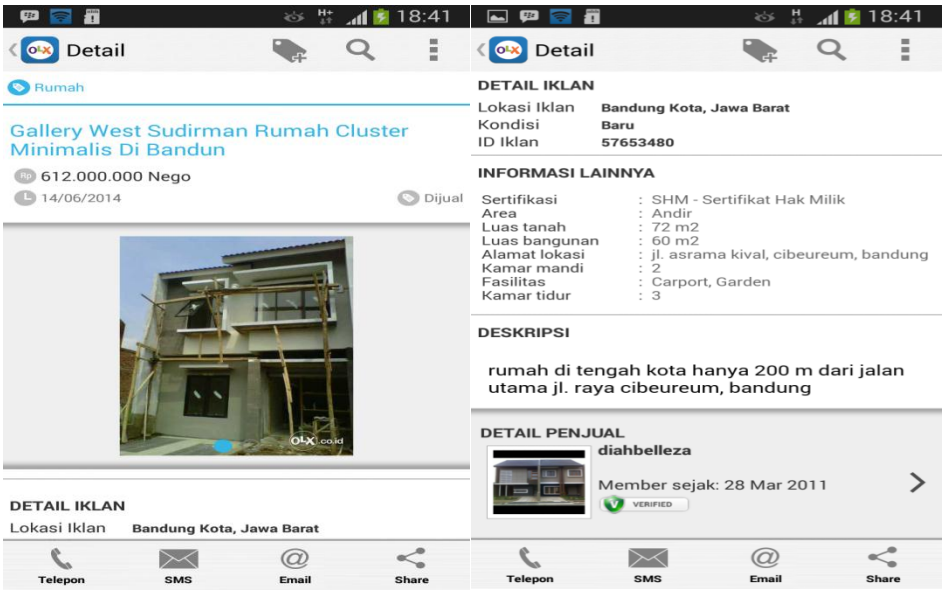
BAB III
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis dan perancangan sistem memerlukan tahapan yang sistematis untuk mendapatkan aplikasi yang baik dan bersesuaian dengan kegunaan dan tujuannya.Tahap awal dari analisis adalah menganalisis kebutuhan-kebutuhan sistem mulai dari kubutuhan pengguna, kubutuhan non fungsional, dan kebutuhan fungsional.Sedangkan untuk tahap perancangan aplikasi yaitu perancangan database, dan perancangan antarmuka.

3.1.1 Analisis Masalah

Sulitnya mendapatkan informasi mengenai rumah tinggal yang sesuai dengan kriteria kita menjadi sesuatu yang sulit bagi semua orang yang sedang mencari informasi mengenai lokasi dan bentuk fisik dari rumah tinggal tersebut.Sudah banyak media yang mengiklankan tentang informasi tempat tinggal tersebut seperti website-website penyedia layanan jual beli seperti olx.co.id, berniaga.com dan rumah123.com, namun informasi yang disajikan dalam menentukan lokasi dan alur rute belum ada.Cara yang dapat dilakukan untuk mendapatkan informasi tersebut adalah dengan melihat peta yang tersedia atau bertanya kepada orang sekitar mengenai lokasi tempat tinggal yang dicari.



Gambar 3.1. Tampilan olx.co.id



Gambar 3.2 Tampilan Berniaga.com

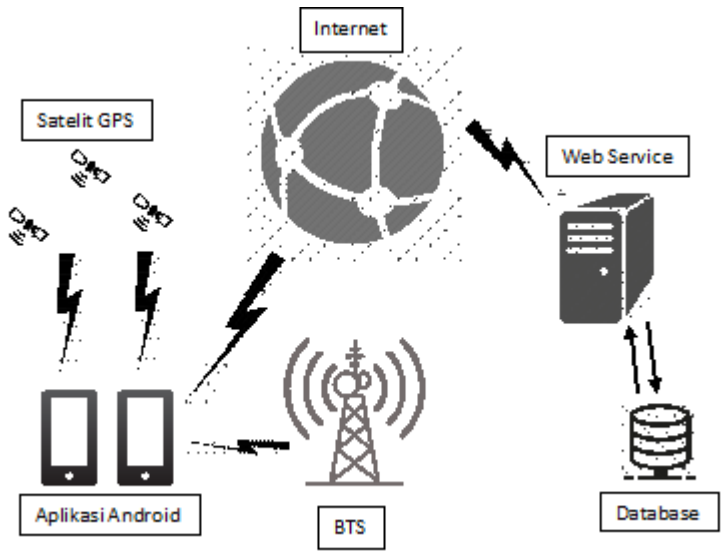


Gambar 3.3 Tampilan Rumah123.com

Namun cara tersebut bukanlah cara yang paling efektif dan efisien, mengingat bahwa orang di sekitar belum tentu mengetahui informasi yang dibutuhkan dan pastinya membutuhkan waktu yang lama. Oleh karena itu dibutuhkan suatu sistem yang dapat menjawab kebutuhan ini, yaitu sistem yang memanfaatkan fitur peta digital dan pembangkit rute serta cepat dan mudah digunakan dalam memberikan informasi tersebut yaitu informasi yang disajikan melalui media aplikasi *mobile*.Karena menggunakan perangkat *mobile* dapat digunakan kapan saja dan dimana saja dengan jaringan internet.

3.1.2 Deskripsi Sistem

Arsitektur fisik sistem mengacu kepada model arsitektur aplikasi two tier. Arsitektur fisik sistem terdiri dari tiga komponen utama yaitu *Client* (Frontend), application *Server* (Web Admin dan Web Service), dan database *Server*. Arsitektur sistem dalam pembangunan aplikasi Tempat Tinggal ini ditunjukan oleh gambar 3.4.



Gambar 3.4 Arsitektur Sistem

Aplikasi *mobile* haruslah terhubung dengan jaringan internet, sehingga aplikasi *mobile* dapat berkomunikasi dengan satelit GPS yang berguna untuk memberikan informasi dimana perangkat *mobile* yang pengguna gunakan sedang berada, dimana informasi lokasinya adalah informasi bujur dan lintang. Dan juga sebagai alternatif, aplikasi *mobile* dapat juga menggunakan BTS yang memiliki fungsi yang sama dengan GPS namun memiliki tingkat akurasi yang tidak seunggul GPS tetapi memiliki jaringan yang baik bila dibandingkan dengan GPS. Tujuan lain aplikasi *mobile* harus terhubung dengan internet adalah agar aplikasi *mobile* dapat berkomunikasi dengan webserver dimana aplikasi dapat melakukan request kepada webserver dan webserver dapat melakukan response kepada aplikasi *mobile*. Dan juga karena database yang digunakan adalah database yang terpusat, maka database dapat diakses dari aplikasi *mobile* melalui webservice, sehingga disini webservice berperan sebagai pengengah antara aplikasi *mobile* dan database.

Pengguna menggunakan aplikasi ini melalui handphonenya, dalam hal ini aplikasi *mobile*-nya adalah front-end dari sistem yang dibuat, dimana front-end yang terdapat pada aplikasi *mobile* ini memiliki fungsi-fungsi yang bekerja dengan cara melakukan request kepada back-end yaitu web service dan back-end melakukan

query pada database dimana hasil dari query ini akan webservice berikan sebagai response kepada aplikasi *mobile*. Dalam sistem ini aplikasi *mobile* dijalankan pada sistem operasi Android, webservice dibangun dengan bahasa pemrograman PHP dan database dibuat dengan menggunakan MySQL.

3.1.2.1 Analisis Penentuan Lokasi Dengan GPS

Dalam menentukan posisi suatu lokasi pada penelitian ini metode yang digunakan yaitu Cell-ID (Cell Identification).Yaitu metode yang didasarkan pada daerah geografis yang tercakup oleh sebuah cell yang berhubungan dengan daerah cakupan dari sinyal GPS atau radio.Untuk itu cell yang dikenali dari base station merupan sebuah nilai koordinat berupa Latitude dan Longitude. Untuk mengukur jarak *mobile* dari base station tidak dapat diketahui dengan pasti, Oleh karena itu, untuk lebih meningkatkan lagi akurasi hasil pencarian, metode Cell ID dapat dikombinasikan dengan metode Timing Advanced (TA), dengan menggunakan TA ini, metode Cell ID akan ditambahkan sebuah fungsionalitas untuk menghitung Round Trip Time (RTT), yaitu waktu transmisi sebuah frame (dari base station ke handphone) dan waktu penerimaan sebuah frame (dari *mobile* ke base station). Dengan tambahan metode ini, jarak antara *mobile* dan base station dapat ditentukan dengan keakuratan 50 m.

Berikut ini salah satu rumus yang sering digunakan untuk menentukan jarak lokasi pada metode CELL ID dengan mengabaikan bentuk geografis baik bukit maupun lembah.

```
Jarak = Arc Cos(Cos(Lat1) x Cos(Lat2) x Cos(Long2-Long1) + Sin(Lat1)
x Sin(Lat 2) ) x R
```

Dimana

Lat1 = Nilai Latitude dari Lokasi Awal

Long1 = Nilai Longitude dari Lokasi Awal

Lat2 = Nilai Latitude dari Lokasi Tujuan

Long2 = Nilai Latitude dari Lokasi Tujuan

R = Jari-jari rata-rata Bumi (6378.2 KM)

Catatan : Nilai Lat dan Long dalam satuan radian.

Contoh :

Misalkan untuk mencari jarak antara Bandara Sultan Iskandar Muda di Banda Aceh ke Bandara Sentani di Jayapura Jika diketahui :

1. Latitude 1 (Banda Aceh) 5°31'1"N dan Longitude 1 (Banda Aceh) 95°25'12"E
2. Latitude 2 (Jayapura) 2°34'19"S dan Longitude 2 (Jayapura)140°30'41"E

Koordinat diatas kan masih dalam bentuk degree minute secon dirubah ke bentuk degree decimal

1. Latitude 1 (Banda Aceh) 5.516944 dan Longitude 1 (Banda Aceh) 95.42
2. Latitude 2 (Jayapura) -2.571944 dan Longitude 2 (Jayapura) 140.511389

Perhitungan :76

Jarak = Arc Cos (Cos(5.516944 * π / 180) *
Cos(-2.571944 * π / 180) *
Cos((140.511389 * π / 180) – (95.42* π / 180)) +
Sin(5.516944 * π / 180) *
Sin(-2.571944 * π / 180)) * 6378.2
= Arc Cos (0.6976857) * 6.371
= 5093.849 KM

Sementara metode untuk penentuan lokasi dapat dilakukan dengan 3 macam cara [9], yaitu :

1. Metode *Basic Postioning*

   Pada metode ini, penentuan lokasi didasarkan pada daerah geografis yang tercakup oleh sebuah sel. Ketika ponsel terhubung ke BTS, ponsel diasumsikan berada dalam sel BTS.

2. Metode *Enchanced Positioning*

   Pada metode ini, sebuah ponsel harus akti terhadap 3 BTS.Jarak ponsel terhadap masing-masing BTS ditentukan berdasarkan waktu yang tempuh oleh sinyal dari ponsel ke masing-masing BTS.Selanjutnya dengan triangulasi, posisi ponsel dapat ditentukan.

3. Metode *Advanced Postioning*

   Pada metodi ini dilakukan pengukuran waktu tiba dari sebuah sinyal yang dikirim dari tiga buah satelit GPS.

3.1.2.2 Elemen Google Maps API

Berikut ini elemen dan objek penting pada Google Maps API yang diterapkan pada aplikasi Pencarian Tempat Tinggal.

1. *Markers*

   Marker adalah fungsionalitas simbol yang menandakan suatu titik koordinat lokasi.Sebagai contoh pada penelitian *Gmarker* digunakan untuk menandai lokasi pada peta *Google Maps*.

2. *Directions*

   Adalah *utility* yang menangani masalah pencarian rute.Sebagai contoh pada penelitian *Gdirections* digunakan untuk memandu rute jalan dari pengguna menuju tempat lokasi pada peta Google Maps.

3. *GeoCoding and Reverse GeoCoding*

   Adalah *utility* untuk memproses konversi alamat (seperti "07 Tubagus Ismail Dalam, Bandung") ke koordinat geografis (seperti lintang dan bujur 37,423021 -122,083739) atau fungsionalitas sebaliknya, yang dapat digunakan untuk menempatkan penanda atau posisi peta.

4. *Location Manager*

   Adalah API yang paling penting, dalam memberikan akses ke layanan sistem lokasi dengan pemetaan dan GPS. Dalam hal ini *Location Manager*berfungsi untuk mendapatkan update periodik lokasi geografis perangkat dengan memetakan nilai latitude dan longitude kedalam *Google Maps*.

5. *Overlays Overview*

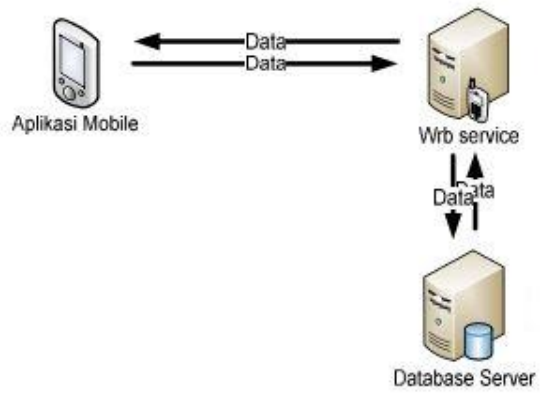
   Adalah objek di *Google Maps* yang terkait dengan koorddinat lintang dan bujur, sehingga objek ini bergerak saat ditarik atau zoom pada peta.*Overlays* mencerminkan benda yang ada pada peta untuk menunjukan titik, garis, area, atau *markers* benda.

6. *Loading Google Maps API* koneksi script yang dibuat keserver Google Map API dengan menggunakan API *key* yang didapatkan pada saat mendaftar ke *Google Maps* API.

7. *Map DOM Elements*

   Untuk menampilkan map pada halaman web, umumnya menggunakan *div tag* untuk menampungnya. Diharuskan membuat sebuah *div tag* dengan nama *"map_canvas",* dan mereferensikan element ini ke *Document Object Model*(DOM).

3.1.3 Analisis Alur Data Sistem



Gambar 3.5. Tampilan Cara Kerja Aplikasi

Analisis alur data sistem pada gambar 3.5 merupakan analisis yang berfungsi untuk menggambarkan secara rinci bagaimana sistem ini dapat bekerja. Analisis ini meliputi :

1. Aplikasi *Mobile*

Aplikasi *mobile* dalam hal ini merupakan analisis fungsionalitas – fungsionalitas aplikasi untuk mengakses data dari database *server* melalui web service. Aplikasi ini dibangun diatas platform android dan bekerja dengan cara memparsing data dari web service yang bertipe JSON untuk diolah pada platform android.

2. Web service

   Web service pada penelitian ini berfungsi sebagai jembatan antara aplikasi *mobile* platform android dengan database *server*.Cara kerja web service ini yaitu dengan mengambil data dari database *server* dan kemudian mengkonversikan data tersebut kedalam format pertukaran data JSON.

3. Protokol HTTP

   Protokol HTTP merupakan protokol jaringan lapisan aplikasi yang digunakan untuk sistem informasi terdistribusi, kolaboratif, dan menggunakan hipermedia.Ada beberapa operasi HTTP yang salah satu contohnya adalah GET dan POST.Operasi GET digunakan untuk meminta sumber dari *server*.Operasi POST digunakan untuk mengirim informasi ke sumber yang ada di *server*.*Server*akan merespon dengan header HTTP yang diikuti dengan bagian data. Pada kasus operasi pemberian GET, respon dari *server*akan berisi sumber yang diminta klien. Jika *server* menerima operasi POST, *server*akan melewatkan data yang ada pada operasi POST ke sumber yang dapat dieksekusi.

   Setelah *server* menerima kembali hasil dari sumber tersebut, *server*akan memasukkan hasil tersebut ke HTTP respon yang dikirim kembali ke klien Pada penelitian ini HTTP digunakan aplikasi *mobile* sebagai protokol yang dapat mendistribusikan data yang bersumber dari database *server*.

3.1.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan system yang menitikberatkan pada properti prilaku yang dimiliki oleh sistem, diantaranya

kebutuhan perangkat keras, perangkat lunak, serta *user* sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.1.4.1 Analisis Kebutuhan Perangkat Keras

Berikut ini adalah spesifikasi perangkat keras minimum yang mendukung aplikasi fasilitas umum dengan baik, yaitu :

1. Perangkat Keras Aplikasi Backend

   Processor : Dengan Kecepatan 2.0 GHZ
   Memori : 4GB
   VGA : 2GB
   Hardisk : 500 GB

2. Perangkat Keras Aplikasi Frontend

   Resolusi : 1080 x 720 pixels
   Memori : 1 GB
   CPU : 1400 MHz
   WLAN : Wi-fi 802.11 b/g/n
   Jaringan 3G : UMTS
   GPS : GLONASS

3.1.4.2 Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak terdiri dari spesifikasi minimum perangkat yang dibutuhkan dan elemen google maps api.

a. Spesifikasi Minimum Perangkat Lunak

Berikut ini adalah spesifikasi minimum perangkat lunak yang digunakan dalam membangun dan mengimplementasikan aplikasi fasilitas umum ini.

1. Sistem Operasi Windows 7 Proffesional
2. Aplikasi IDE Eclipse ADT (Andriod Development Tools)

3. JDK v 1.8 dan Android SDK windows
4. Blue stack versi 2.0

3.1.4.3 Analisis Pengguna Sistem (User)

Analisis pengguna sistem dimaksudkan untuk mengetahui siapa saja actor yang terlibat dalam menjalankan sistem. Sistem aplikasi pencarian tempat tinggal ini sendiri menggunakan platform teknologi bergerak . Pengguna sistem ini sendiri, yaitu :Pengguna aplikasi tempat tinggal atau enduser yaitu para pengguna platform android. Dalam menggunakan sistem ini, pengguna diharuskan memiliki koneksi GRPS/EDGE/UMTS dan fasilitas GPS untuk bisa menggunakan aplikasi ini.
Berikut ini karakteristik pengguna sistem yang ditunjukan oleh tabel 3.1.

Tabel 3.1 Karakteristik Pengguna Sistem

Pengguna	Usia	Hak Akses	Tingkat Pendidikan	Pengalaman
Pembeli	Lebih dari 17 tahun	Mengakses data-data informasi	-	Minimal membuka aplikasi pada *mobile* Android
Penjual	Lebih dari 17 tahun	Mengakses dan mengelola data – data informasi	-	Minimal membuka aplikasi pada *mobile* Android
Admin	Lebih dari 17 tahun	Menyetujui permohonan penambahan	-	Minimal membuka aplikasi pada

	data penjualan	mobile Android

3.1.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan. Analisis kebutuhan fungsional ini meliputi analisi kebutuhan sistem, analisis kebutuhan data, spesifikasi sistem dan pemodelan sistem.

Pemodelan sistem dimodelkan dengan menggunakan UML (Unified Modeling Language). Tahap-tahap pemodelan dalam analisis tersebut antara lain Usecase diagram, Class Diagram, State Diagram, Activity Diagram, Sequence Diagram, Collaboration Diagram, Component Diagram dan Deployment Diagram.

3.1.5.1 Analisis Kebutuhan Sistem

Sebelum melakukan perancangan sistem, terlebih dahulu dilakukan analisis kebutuhan sistem yaitu dengan cara melakukan wawancara terhadap responden. Hal ini dimaksudkan agar dapat mengatasi ketidaksesuaian Antara aplikasi yang dirancang dengan kebutuhan pengguna. Adapun kebutuhan system yang diperlukan antara lain :

1. Menampilkan informasi mengenai keberadaan tempat tinggal.
2. Memberikan informasi data tempat tinggal dengan lengkap dan rinci.
3. Menampilkan peta secara keseluruhan dengan menunjukan posisi lokasi tempat tinggal.
4. Dapat melakukan Pencarian tempat tinggal dengan cepat.

3.1.5.2 Analisis Kebutuhan Data

Setelah melakukan proses analisis kebutuhan sistem, maka analisis selanjutnya adalah analisis kebutuhan data. Analisis ini bertujuan untuk memudahkan

dalam perancangan informasi. Adapun analisis kebutuhan data yang diperlukan antara lain :

1. Data Kategori

   Kriteria data kategori yang akan ditampilkan adalah dengan memilih salah satu dari 4 kategori yaitu rumah, kontrakkan, kostan,apartemen atau semuanya.

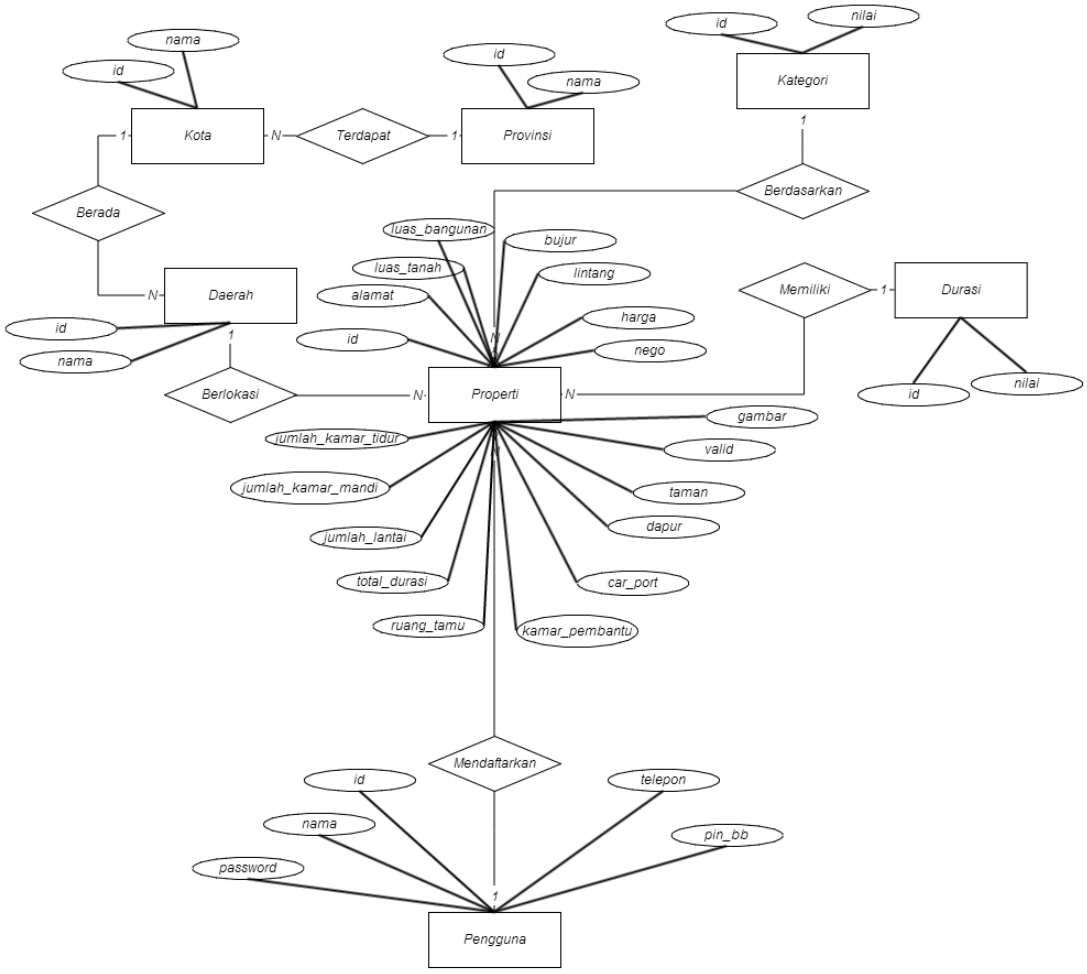
2. Data Lokasi Pembeli

   Kriteria data lokasi pembeli yang akan ditampilkan adalah informasi rumah yang terdekat dengan lokasi pembeli.

3. Data Provinsi

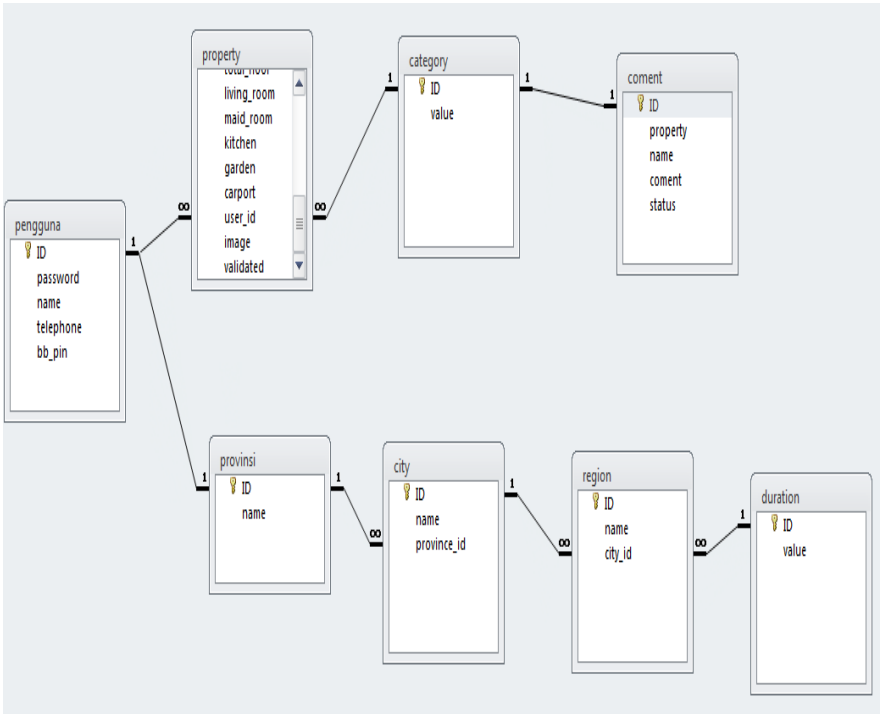
   Kriteria data provinsi yang akan ditampilkan adalah informasi rumah yang berada pada provinsi yang dipilih oleh *user*.

3.1.5.2.1 Entity Relationship Diagram



Gambar 3.6 Tampilan Entity Relationship Diagram

3.1.5.2.2Skema Relasi



Gambar 3.7Skema Relasi

3.1.5.2.3Struktur Tabel

Tabel 3.2Struktur Tabel Pengguna

Nama Field	Tipe	Deskripsi
id	varchar(11)	*Primary key*, id
password	varchar(50)	Password pengguna
name	varchar(100)	Nama pengguna
telephone	varchar(15)	No telp pengguna
Bb_pin	varchar(8)	Pin bb pengguna

Tabel 3.3Struktur Tabel Property

Nama Field	Tipe	Deskripsi
id	varchar(11)	*Primary key*, id
category_id	Int(11)	Id kategori
address	varchar(255)	Alamat property
city_id	Int(11)	*Foreign key*, city_id
region_id	Int(11)	*Foreign key*, region_id
land_area	Int(11)	-
building_area	Int(11)	-
latitude	*double*	-
longitude	*double*	-
price	Bigint(20)	Harga property
negotiable	Tyniint(1)	-
duration_id	Int(11)	-
total_duration	Int(11)	-
total_bed_room	Int(11)	-
total_bath_room	Int(11)	-
total_floor	Int(11)	-
living_room	Tyniint(1)	-
maid_room	Tyniint(1)	-
kitchen	Tyniint(1)	-
garden	Tyniint(1)	-
carport	Tyniint(1)	-

user_id	varchar(50)	-
image	varchar(255)	-
validated	Tyniint(1)	-

Tabel 3.4Struktur Tabel Province

Nama Field	Tipe	Deskripsi
id	int(11)	*Primary key*, id
name	varchar(100)	Nama propinsi

Tabel 3.5Struktur Tabel City

Nama Field	Tipe	Deskripsi
id	int(11)	*Primary key*, id
Name	varchar(100)	Nama kota
Province_id	Int(11)	*Foreign key*, province_id

Tabel 3.6Struktur Tabel Region

Nama Field	Tipe	Deskripsi
id	varchar(11)	*Primary key*, id
name	varchar(100)	Nama daerah
City_id	varchar(11)	*Foreign key*, city_id

Tabel 3.7Struktur Tabel Category

Nama Field	Tipe	Deskripsi
id	int(11)	*Primary key*, id

value	varchar(50)	Nama kategori

Tabel 3.8Struktur Tabel Duration

Nama Field	Tipe	Deskripsi
id	int(11)	*Primary key*, id
value	varchar(50)	durasi

Tabel 3.9Struktur Tabel City

Nama Field	Tipe	Deskripsi
id	int(11)	*Primary key*, id
property	Int(11)	*Foreign key*, property
Name	varchar(100)	Nama pengguna
comment	varchar(255)	komentar
Status	Int(11)	status

3.1.5.2.4 Implementasi Basis Data

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana aplikasi pemograman yang digunakan adalah MySQL, Implementasi basis datanya dalam bahasa SQL adalah sebagai berikut :

1. **Table user**

```
CREATE TABLE IF NOT EXISTS `user` (
```

```
  `id` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `name` varchar(50) NOT NULL,
  `telephone` varchar(50) NOT NULL,
  `bb_pin` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Table property

```
CREATE TABLE IF NOT EXISTS `property` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `category_id` int(11) NOT NULL,
  `address` varchar(255) NOT NULL,
  `city_id` int(11) NOT NULL,
  `region_id` int(11) NOT NULL,
  `land_area` int(11) NOT NULL,
  `building_area` int(11) NOT NULL,
  `latitude` double NOT NULL,
```

```
  `longitude` double NOT NULL,
  `price` bigint(20) NOT NULL,
  `negotiable` tinyint(1) NOT NULL,
  `duration_id` int(11) NOT NULL,
  `total_duration` int(11) NOT NULL,
  `total_bed_room` int(11) NOT NULL,
  `total_bath_room` int(11) NOT NULL,
  `total_floor` int(11) NOT NULL,
  `living_room` tinyint(1) NOT NULL,
  `maid_room` tinyint(1) NOT NULL,
  `kitchen` tinyint(1) NOT NULL,
  `garden` tinyint(1) NOT NULL,
  `carport` tinyint(1) NOT NULL,
  `user_id` varchar(50) NOT NULL,
  `image` varchar(255) NOT NULL,
  `validated` tinyint(1) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=11 ;
```

3. Table province

```
CREATE TABLE IF NOT EXISTS `province` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=35 ;
```

4. Table city

```
CREATE TABLE IF NOT EXISTS `city` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(50) NOT NULL,
  `province_id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=45 ;
```

5. **Table region**

```
CREATE TABLE IF NOT EXISTS `region` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(50) NOT NULL,
  `city_id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=27 ;
```

6. **Table duration**

```
CREATE TABLE IF NOT EXISTS `duration` (
  `id` int(11) NOT NULL,
  `value` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. **Table comment**

```
CREATE TABLE IF NOT EXISTS `comments` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `property` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `comment` varchar(255) NOT NULL,
  `status` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=34 ;
```

8. Table category

```
CREATE TABLE IF NOT EXISTS `category` (
  `id` int(11) NOT NULL,
  `value` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3.1.5.3 Spesifikasi Sistem

Adapun spesifikasi sistem yang dibutuhkan antara lain :

1. Sistem dikembangkan dengan model arsitektur *two tier*. Aplikasi Pencarian Tempat Tinggal dikembangkan disisi *client* atau *frontend* sebagai aplikasi *OnDevice Portal* (ODP). Sedangkan disisi *server* atau *backend* akan dikembangkan aplikasi web.
2. Konektivitas antara aplikasi Pencarian Tempat Tinggal pada *platform* android ini dengan aplikasi *backend* di s*erver* menggunakan koneksi protocol HTTP dengan memanfaatkan jaringan GPRS/EDGE/UMTS.
3. Sistem informasi yang disediakan dibatasi pada informasi tempat tinggal dan lokasi.
4. Spesifikasi sistem disisi *frontend* yaitu :
    a. Perangkat yang digunakan harus bisa digunakan dimana saja dan kapan saja serta memiliki akses internet.
    b. Aplikasi ini menyediakan informasi tentang lokasi-lokasi yang berhubungan dengan informasi tempat tinggal beserta dengan informasi pendukungnya seperti lokasi yang ditampilkan dalam google map.
    c. Aplikasi ini memiliki ukuran yang minimum sehingga mampu disimpan dalam perangkat seluler yang memiliki keterbatasan.
5. Spesifikasi sistem di *backend* yaitu:
    a. Sistem bertanggung jawab untuk menerima request dan meresponse setiap permintaan *user*.
    b. Sistem menjadi penghubung antara *user* dengan basis data.

3.1.5.4 Pemodelan Sistem

Pada tahap ini dilakukan pemodelan sistem untuk menentukan fungsi-fungsi yang dapat dilakukan oleh sistem aplikasi serta menentukan kelas yang dibutuhkan untuk

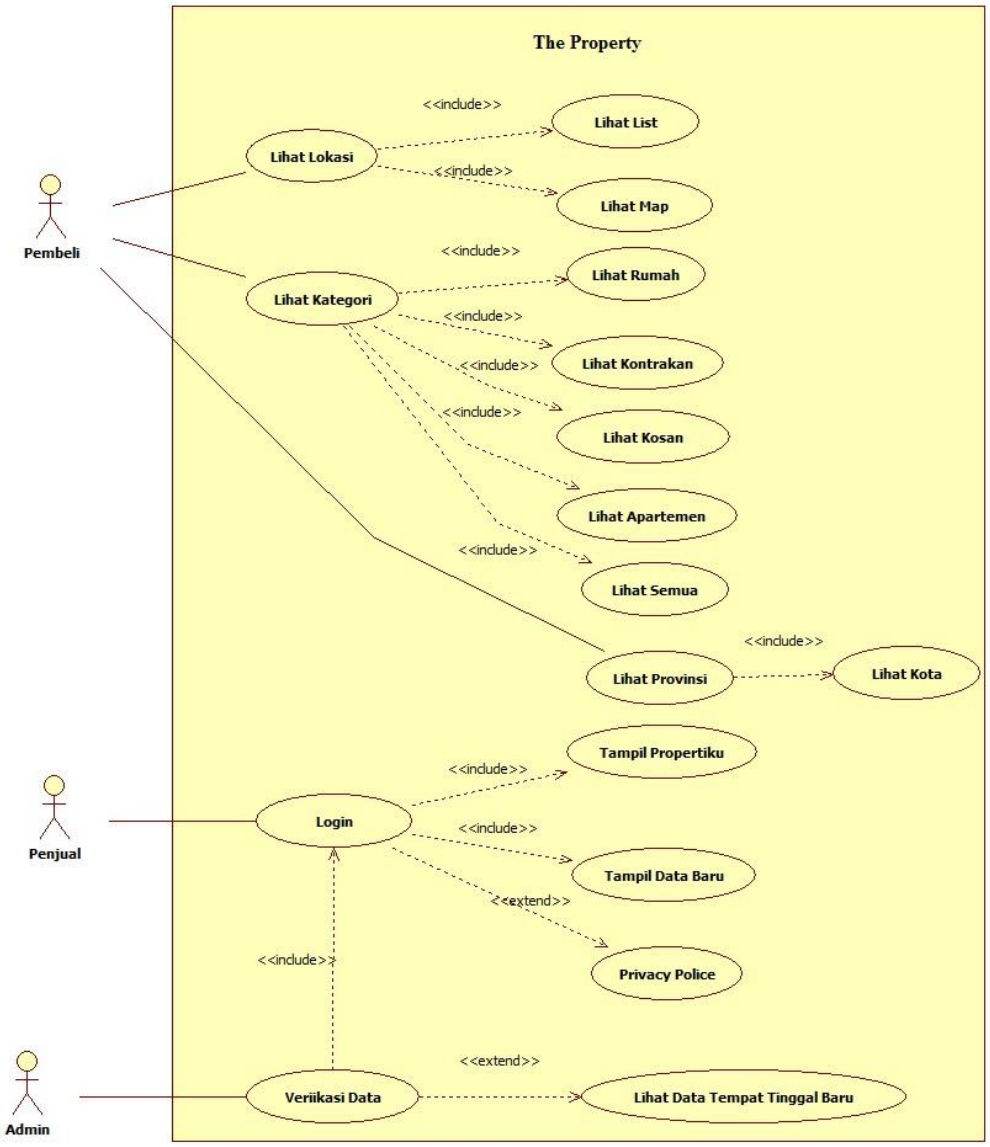
realisasi fungsi-fungsi sistem yang telah dianalisis sebelumnya dan mendeskripsikannya kedalam bentuk diagram.

a. *Use Case* Diagram

Diagram *Use Case* merupakan bagian tertinggi dari fungsionalitas yang dimiliki sistem yang akan menggambarkan bagaimana seseorang atau aktor akan menggunakan dan memanfaatkan sistem. Diagram ini juga mendeskripsikan apa yang akan dilakukan oleh sistem. *Use Case* terdiri dari tiga bagian yaitu identifikasi aktor, identifikasi *Use Case* dan skenario *Use Case*.

1. Identifikasi Aktor

Aktor yang berperan dalam menjalankan sistem dapat dilihat pada gambar 3.7.



Gambar 3.7 Use Case Diagram

2. Identifikasi *Use Case*

Identifikasi *use case* yang terdapat dalam sistem dapat dilihat pada table 3.2.

Tabel 3.10 Deskripsi Use Case

No.	Use Case	Deskripsi
1.	Lihat list	Fungsionalitas untuk menampilkan data tempat tinggal yang diurutkan berdasarkan jarak
2.	Lihat map	Fungsionalitas untuk melihat data tempat tinggal pada peta digital
3.	Lihat rumah	Fungsionalitas untuk menampilkan semua data rumah yang dijual
4.	Lihat kontrakan	Fungsionalitas untuk menampilkan semua data rumah yang dikontrakan
5.	Lihat kostan	Fungsionalitas untuk menampilkan semua data kostan
6.	Lihat apartemen	Fungsionalitas untuk menampilkan semua data apartemen
7.	Lihat semua	Fungsionalitas untuk menampilkan semua data tempat tinggal
8.	Lihat kota	Fungsionalitas untuk menampilkan daftar tempat tinggal berdasarkan kota yang dipilih
9.	Lihat lokasi	Fungsionalitas untuk menampilkan data tempat tinggal berdasarkan lokasi yang dipilih
10.	Lihat kategori	Fungsionalitas untuk menampilkan data tempat tinggal berdasarkan kategori yang dipilih
11.	Lihat provinsi	Fungsionalitas untuk menampilkan data tempat tinggal berdasarkan

		provinsi yang dipilih
12.	Tampil Propertiku	Fungsionalitas untuk menampilkan daftar tempat tinggal yang sedang dijual oleh penjual
13.	Tampil data baru	Fungsionalitas untuk menambahkan data tempat tinggal baru yang ingin diumumkan kepada publik
14.	Privacy Policy	Fungsionalitas untuk menampilkan kebijakan dan peraturan transaksi
15.	Lihat data tempat tinggal baru	Fungsionalitas untuk menampilkan daftar tempat tinggal baru di sisi admin
16.	Verifikasi data	Fungsionalitas untuk mengesahkan data tempat tinggal yang dimasukkan oleh penjual
17.	Login	Fungsionalitas untuk masuk ke menu utama penjual dan admin

3. Skenario *Use Case*

Skenario proses-proses yang terdapat didalam *Use Case* dapat dilihat pada tabel-tabel dibawah ini.

Tabel 3.11 Skenario Use Case Lihat List

Identifikasi	
Nama Use Case	**Lihat List**
Aktor	**Pembeli**
Tujuan	**Menampilkan data tempat tinggal pada list**

Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih kategori	2. Menampilkan data berdasarkan kategori

Tabel 3.12 Skenario Use Case Lihat Map

Identifikasi	
Nama Use Case	**Lihat Map**
Aktor	**Pembeli**
Tujuan	**Menampilkan data tempat tinggal pada peta**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih menu lihat map	2. Menampilkan data pada map

Tabel 3.13 Skenario Use Case Lihat Lokasi

Identifikasi	
Nama Use Case	***Lihat Lokasi***
Aktor	**Pembeli**
Tujuan	**Menampilkan lokasi**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Pilih menu Lokasi	2. Menampilkan lihat list dan lihat map

Tabel 3.14 Skenario Use Case Lihat Kategori

Identifikasi	
Nama Use Case	**Lihat Kategori**
Aktor	**Pembeli**
Tujuan	**Menampilkan data kategori**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih kategori	2. Menampilkan data berdasarkan kategori

Tabel 3.15 Skenario Use Case Lihat Rumah

Identifikasi	
Nama Use Case	**Lihat Rumah**
Aktor	**Pembeli**
Tujuan	**Menampilkan data rumah**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih menu lihat data rumah	2. Menampilkan data rumah

Tabel 3.15 Skenario Use Case Lihat Kontrakan

Identifikasi	
Nama Use Case	***Lihat Kontrakan***
Aktor	**Pembeli**
Tujuan	**Menampilkan kontrakan**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	

Aksi Aktor	**Reaksi Sistem**
1. Pilih menu kontrakan	2. Menampilkan lihat kontrakan

Tabel 3.16 Skenario Use Case Lihat Kosan

Identifikasi	
Nama Use Case	**Lihat kosan**
Aktor	**Pembeli**
Tujuan	**Menampilkan data kosan**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih Kosan	2. Menampilkan data kosan

Tabel 3.17 Skenario Use Case Lihat Apartment

Identifikasi	
Nama Use Case	**Lihat Apartment**
Aktor	**Pembeli**
Tujuan	**Menampilkan data apartement**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih menu lihat apartement	2. Menampilkan data apartement

Tabel 3.18 Skenario Use Case Lihat Semua

Identifikasi	
Nama Use Case	***Lihat semua***
Aktor	**Pembeli**

Tujuan	**Menampilkan semua**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Pilih menu semua	2. Menampilkan data lihat semua

Tabel 3.19 Skenario Use Case Lihat Provinsi

Identifikasi	
Nama Use Case	**Lihat provinsi**
Aktor	**Pembeli**
Tujuan	**Menampilkan data provinsi**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih provinsi	2. Menampilkan data provinsi

Tabel 3.20 Skenario Use Case Lihat Kota

Identifikasi	
Nama Use Case	**Lihat kota**
Aktor	**Pembeli**
Tujuan	**Menampilkan data kota**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih menu lihat kota	2. Menampilkan data pada kota

Tabel 3.21 Skenario Use Case Tampil Properti

Identifikasi	
Nama Use Case	***Lihat properti***
Aktor	**Penjual**
Tujuan	**Menampilkan propertiku**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Pilih menu tampil propertiku	2. Menampilkan data propertiku

Tabel 3.22 Skenario Use Case Tampil Data Baru

Identifikasi	
Nama Use Case	**Tampil data baru**
Aktor	**Penjual**
Tujuan	**Menampilkan data baru**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih tampil data baru	2. Menampilkan data baru

Tabel 3.23 Skenario Use Case Login

Identifikasi	
Nama Use Case	**Login**
Aktor	**Admin dan Penjual**
Tujuan	**Menampilkan hak akses ke dalam sistem**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	

Aksi Aktor	**Reaksi Sistem**
1. Memilih menu login	2. Menampilkan data login

Tabel 3.24 Skenario Use Case Privacy Policy

Identifikasi	
Nama Use Case	**Privacy Policy**
Aktor	**Penjual**
Tujuan	**Menampilkan privacy policy**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
3. Memilih menu privacy policy	4. Menampilkan data privacy policy

Tabel 3.25 Skenario Use Case Lihat Data Tempat Tinggal Baru

Identifikasi	
Nama Use Case	***Lihat data tempat tinggal baru***
Aktor	**Admin**
Tujuan	**Menampilkan data tempat tinggal baru**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Pilih menu lihat data tempat tinggal baru	2. Menampilkan lihat data tempat tinggal baru

Tabel 3.26 Skenario Use Case Verifikasi Data

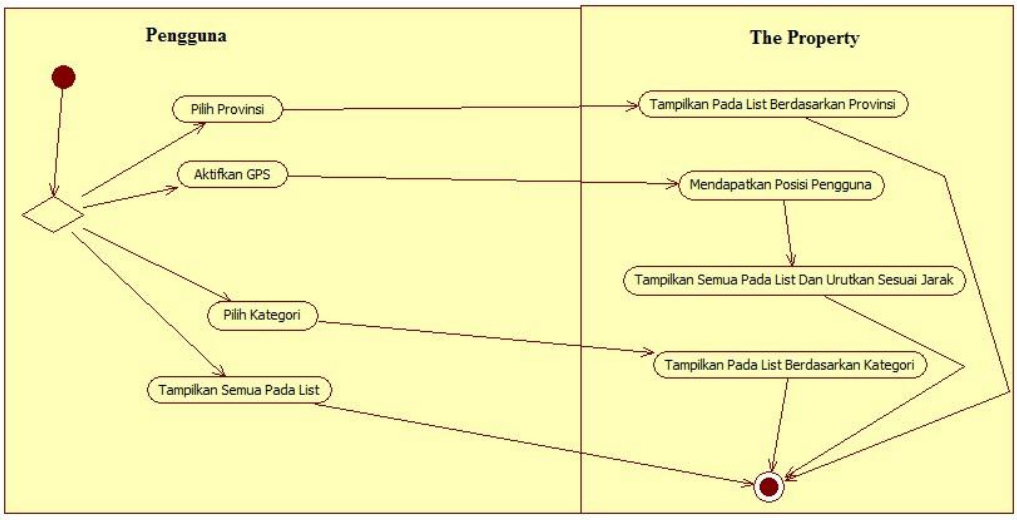
Identifikasi	
Nama Use Case	**Verifikasi data**

Aktor	**Pembeli**
Tujuan	**Menampilkan data verifikasi**
Keadaan Awal	**Sistem menampilkan menu**
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
1. Memilih verifikasi data	2. Menampilkan verifikasi data

b. *Activity Diagram*

Diagram activity merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program.*Activity Diagram* memodelkan event-event yang terjadi didalam suatu *Use Case* dan digunakan untuk pemodelan aspek dinamis dari sistem.

1. **Activity Diagram Lihat Tempat Tinggal**

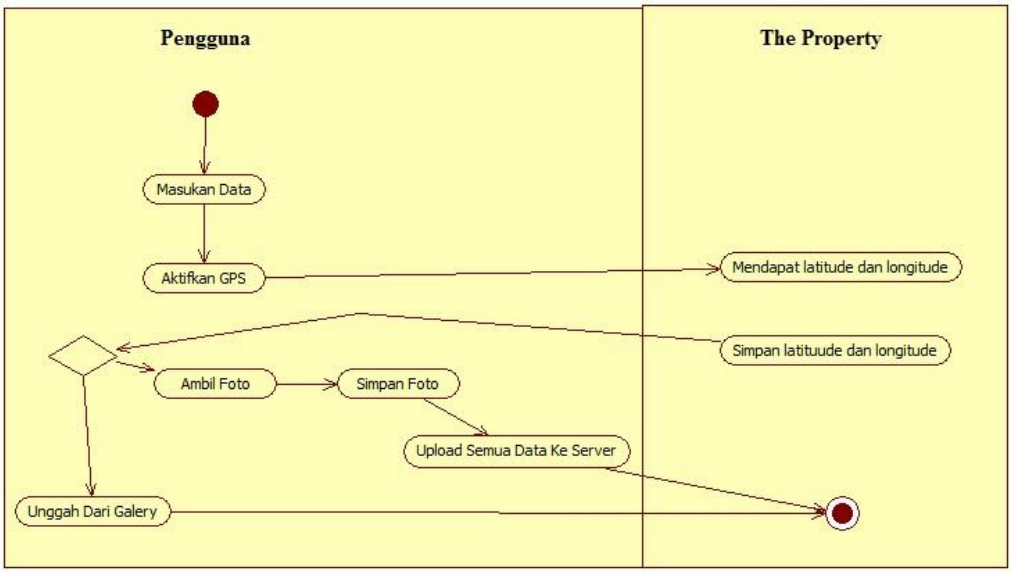


Gambar 3.8 *Activity Diagram* Lihat Tempat Tinggal

Activity diagram yang menjelaskan tentang alur dari proses pencarian data tempat tinggal, dimana pencarian ini dapat dilakukan dengan memilih salah satu dari

kategori – kategori pencarian yang tersedia, yaitu berdasarkan provinsi, berdasarkan kategori, berdasarkan lokasi pengguna atau tampilkan semua data tempat tinggal.

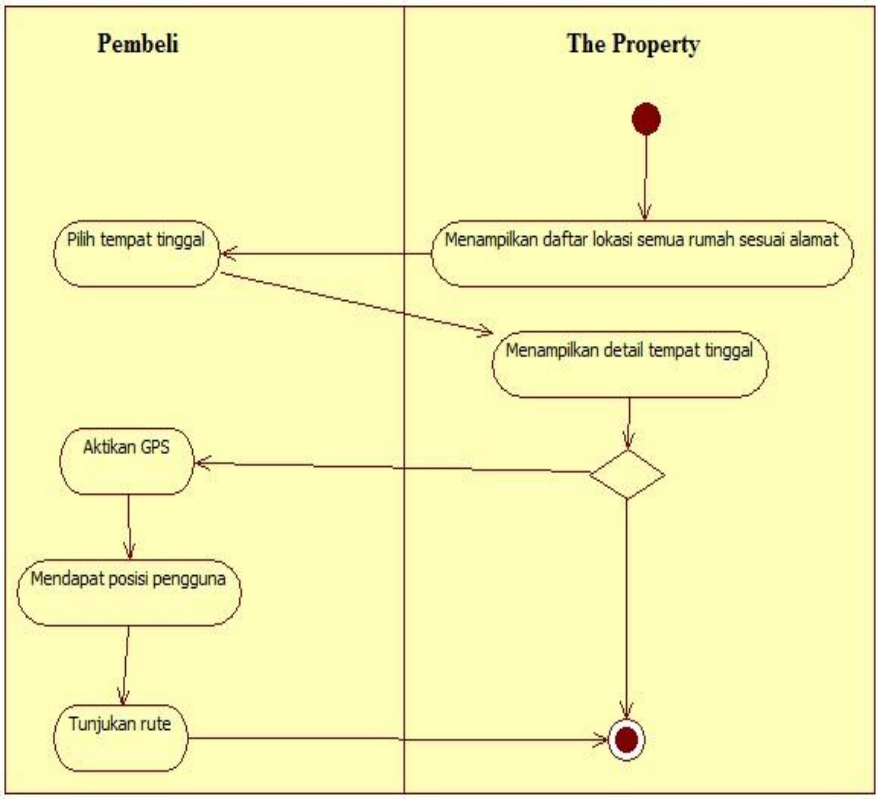
2. Activity Diagram Tambah Tempat Tinggal



Gambar 3.9 *Activity Diagram* Tambah Tempat Tinggal

Activity ini merepresentasikan tentang alur pada saat pengguna ingin menambahkan data tempat tinggal ke dalam database pada aplikasi ini.Dimana ada beberapa tahap yang harus dilewati. Pertama, pengguna harus memasukan data yang paling mendasar, seperti alamat dan harga, lalu setelah itu GPS diaktifkan oleh system, yang bertujuan untuk mendapatkan bujur dan lintang agar lokasi tempat tinggal tersebut diketahui sehingga dapat disajikan pada peta digital, dan yang terakhir adalah pengambilan foto tempat tinggal tersebut. Foto dapat diambil dari camera, atau dari gallery foto.

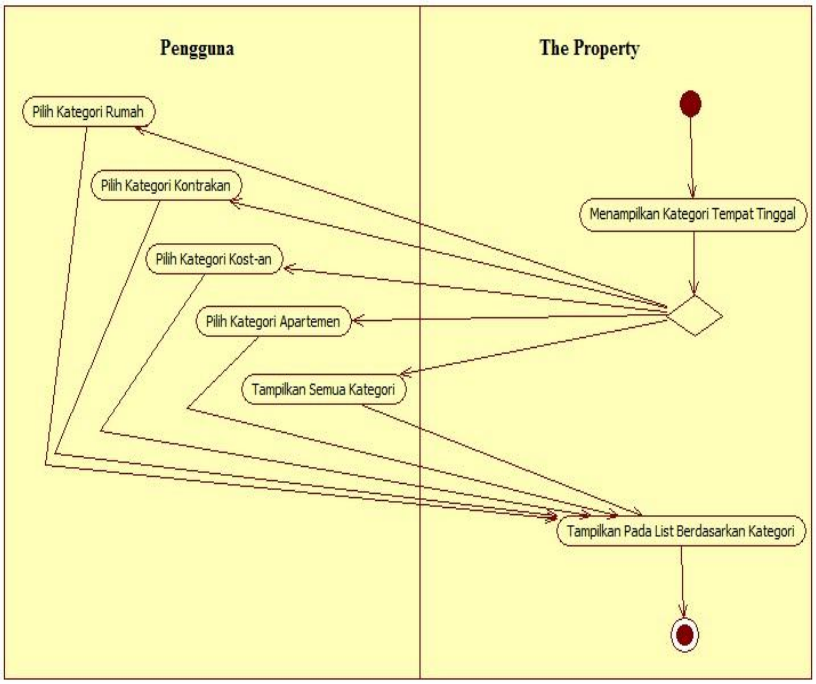
3. Activity Diagram Lokasi



Gambar 3.10*Activity Diagram* Lokasi

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal secara keseluruhan berdasarkan lokasi yang disajikan pada peta digital. Dan juga setelah informasi – informasi ini dihasilkan, maka akan terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

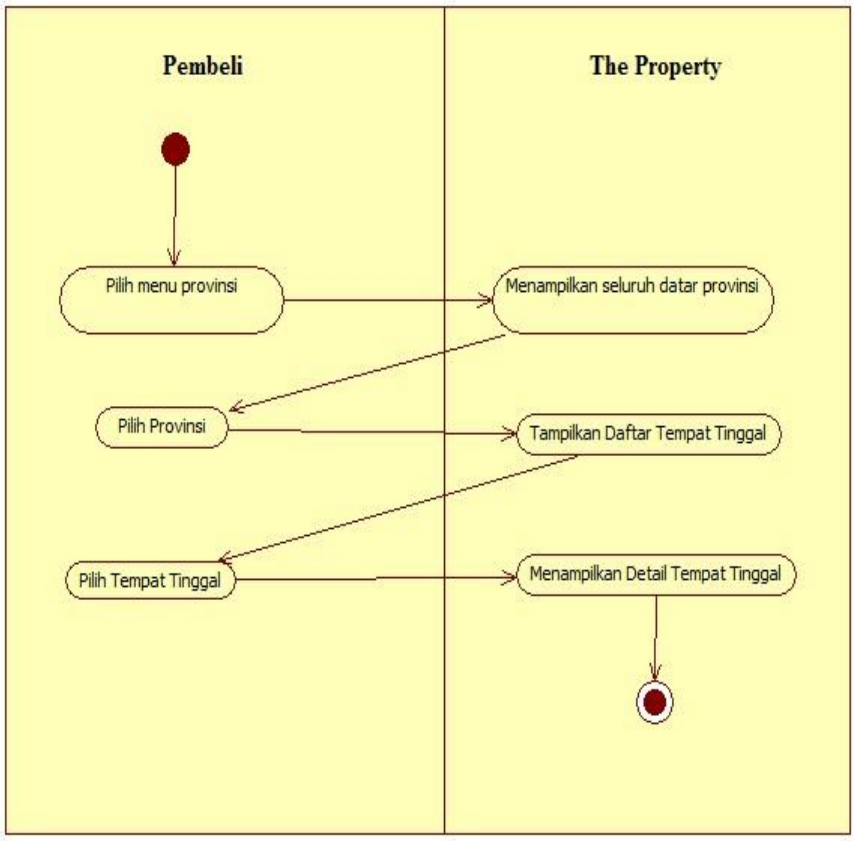
4. Activity Diagram Kategori



Gambar 3.11*Activity Diagram* Kategori

Activity diagram yang menjelaskan tentang alur dari proses pencarian data tempat tinggal, dimana pencarian ini dapat dilakukan dengan memilih salah satu dari kategori – kategori pencarian yang tersedia, yaitu berdasarkan kategori rumah, berdasarkan kategori kontrakan, berdasarkan kategori kost-an atau tampilkan berdasarkan semua kategori.

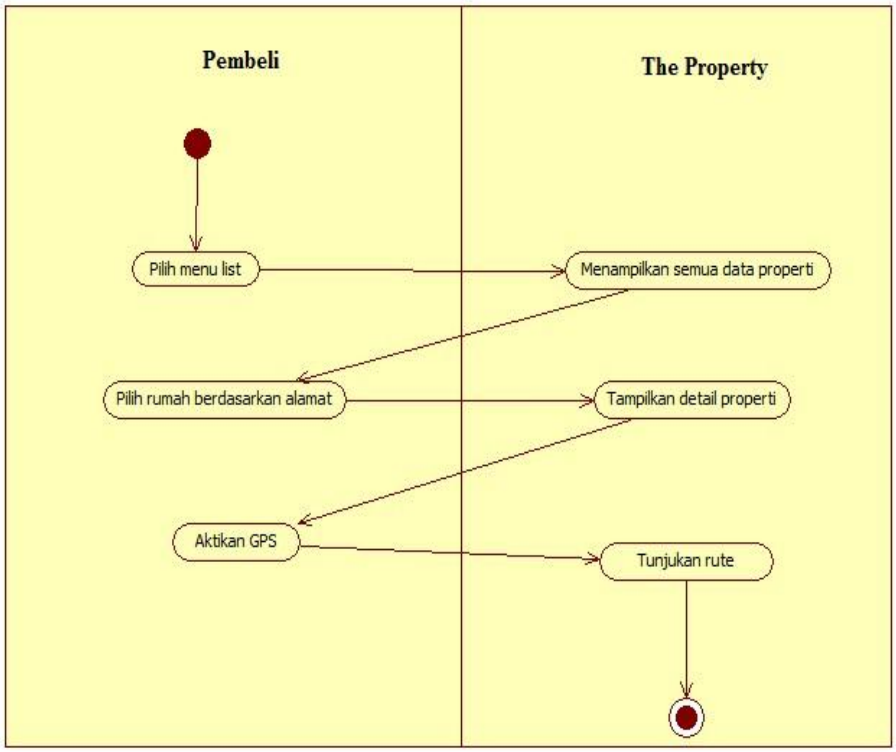
5. Activity Diagram Provinsi



Gambar 3.12*Activity Diagram* Provinsi

Activity diagram yang menjelaskan tentang alur dari proses pencarian data tempat tinggal yang dikelompokan berdasarkan provinsi tempat tinggal.

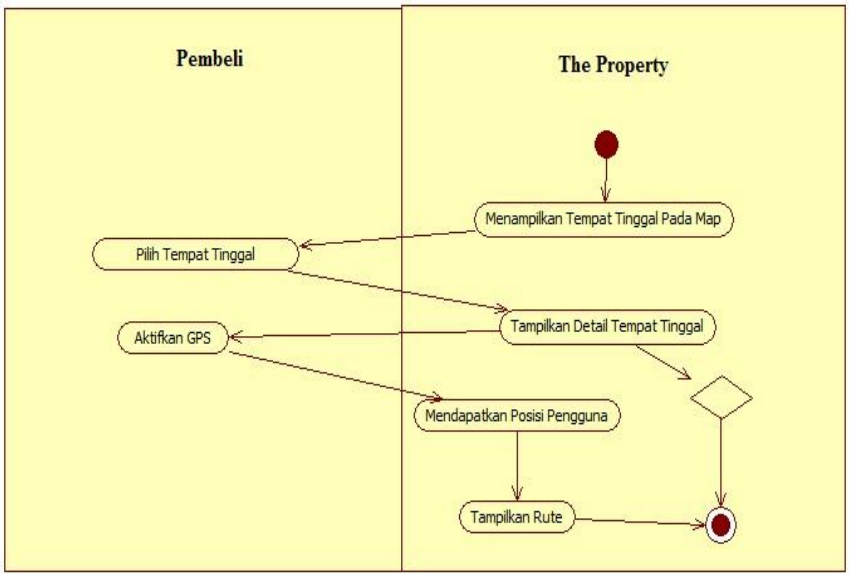
6. Activity Diagram List



Gambar 3.13Activity Diagram List

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal secara menyeluruh berdasarkan lokasi yang disajikan pada peta digital serta detail dari masing-masing tempat tinggal. Dan juga setelah informasi – informasi ini dihasilkan maka akan terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

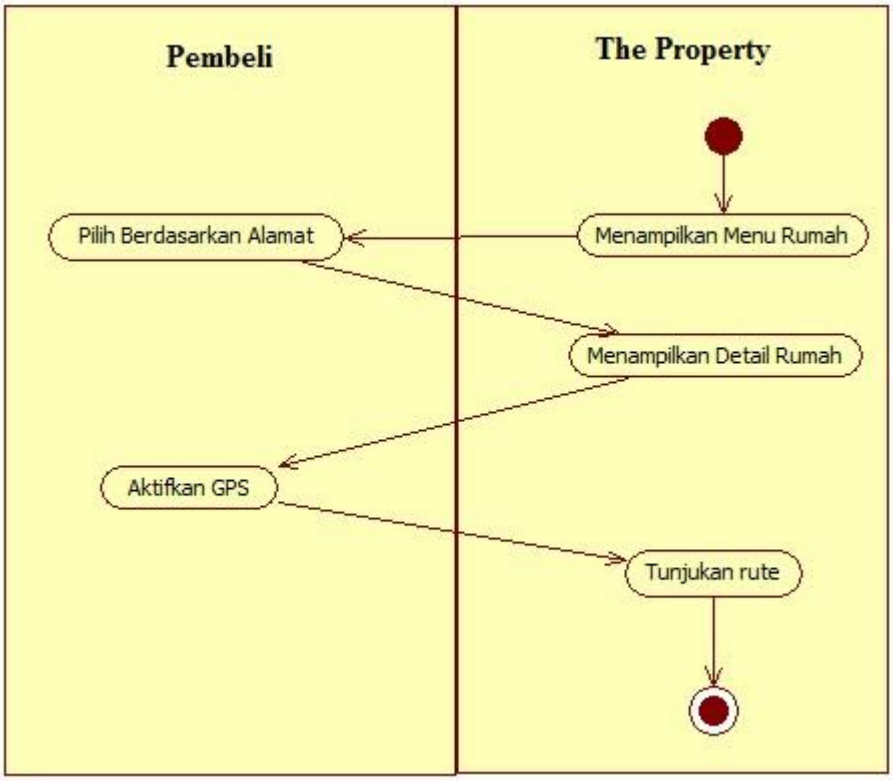
7. Activity Diagram Map



Gambar 3.14Activity Diagram Map

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal yang disajikan pada peta digital. Dan juga setelah informasi – informasi ini dihasilkan, maka akan terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

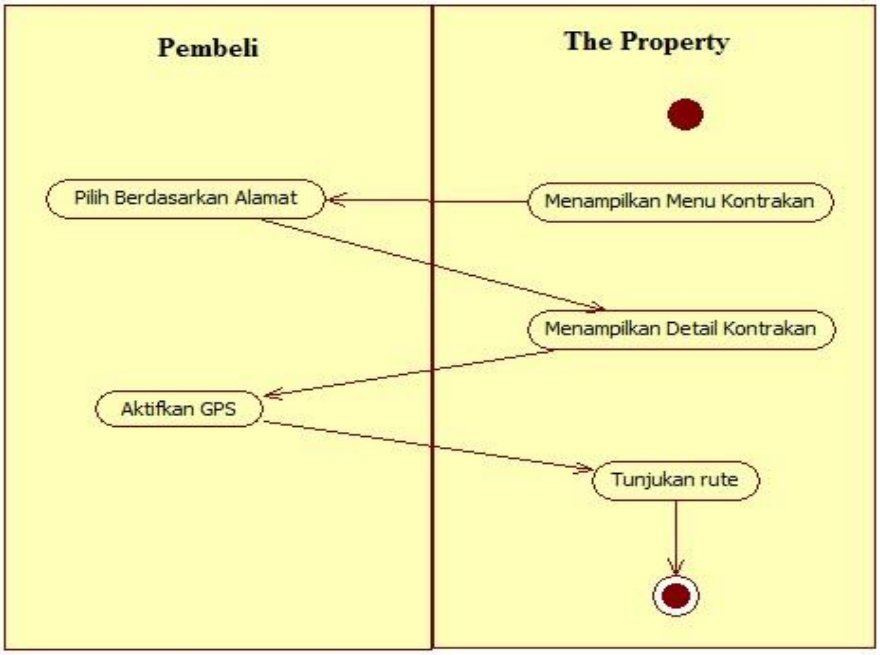
8. Activity Diagram Rumah



Gambar 3.15Activity Diagram Rumah

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal berdasarkan kategori tempat tinggal jenis rumah yang disajikan pada peta digital.Kemudian terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

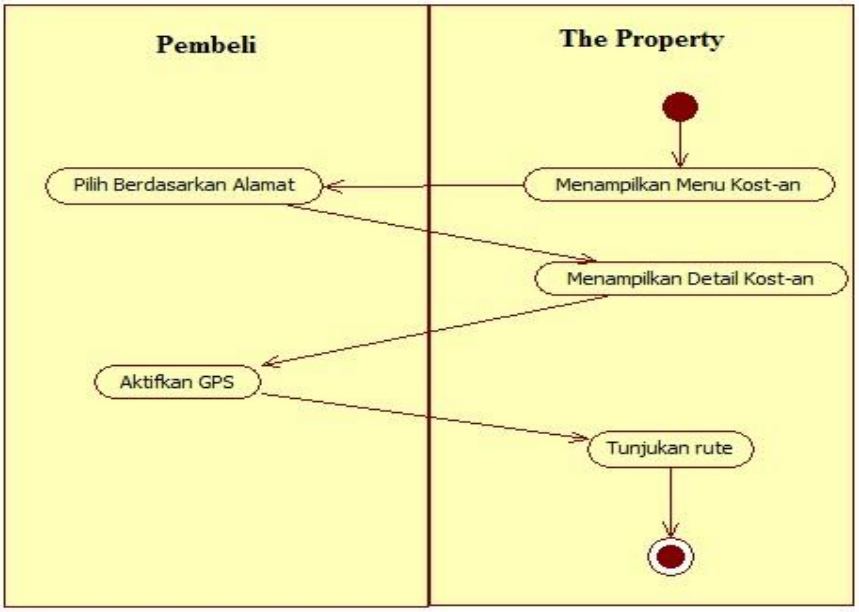
9. Activity Diagram Kontrakan



Gambar 3.16*Activity Diagram* Kontrakan

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal berdasarkan kategori tempat tinggal jenis kontrakan yang disajikan pada peta digital.Kemudian terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

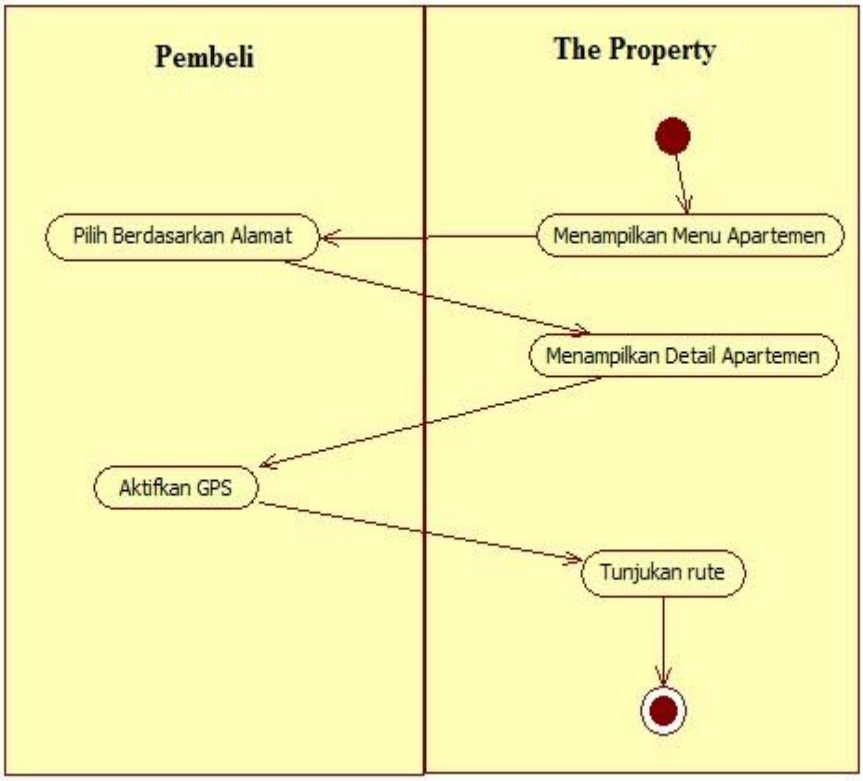
10. Activity Diagram Kost-an



Gambar 3.17*Activity Diagram* Kost-an

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal berdasarkan kategori tempat tinggal jenis kost-anyang disajikan pada peta digital. Kemudian terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

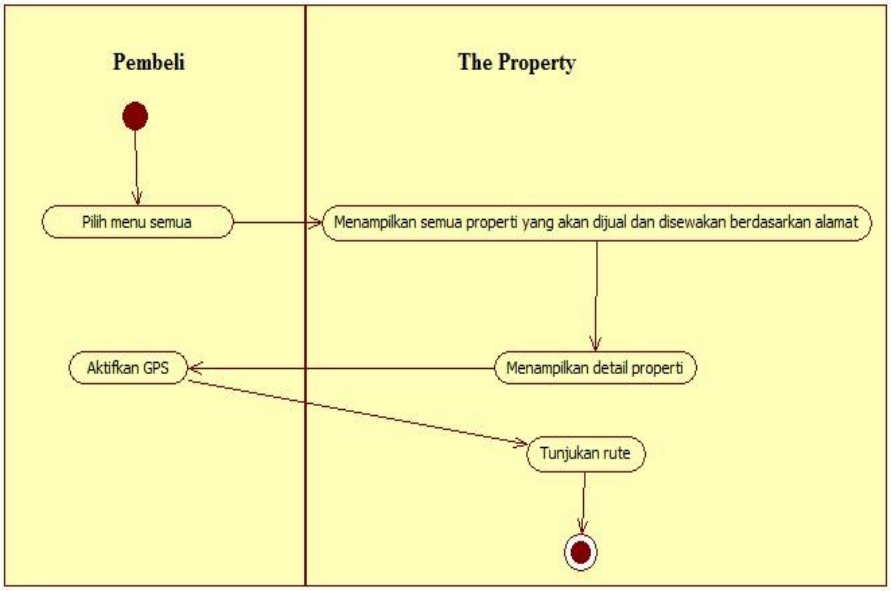
11. Activity DiagramApartemen



Gambar 3.18*Activity Diagram* Apartemen

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal berdasarkan kategori tempat tinggal jenis apartemen yang disajikan pada peta digital.Kemudian terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

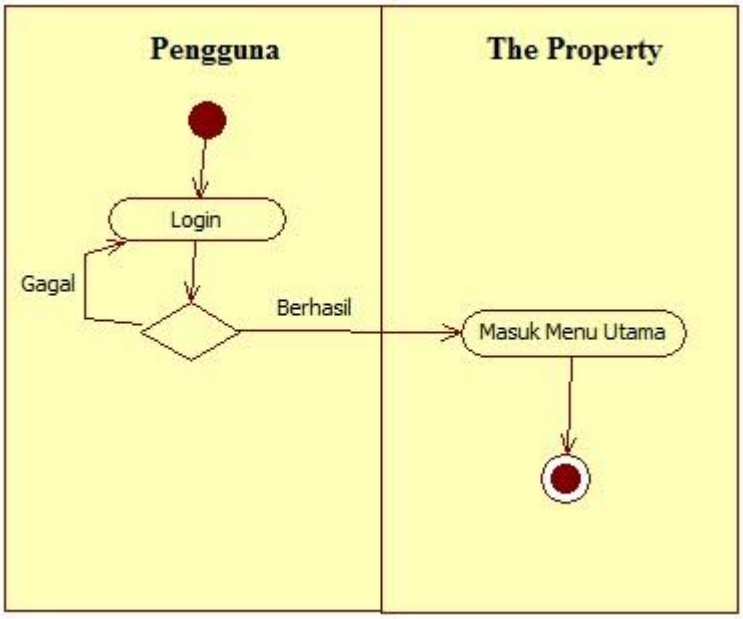
12. Activity Diagram Semua



Gambar 3.19*Activity Diagram* Semua Kategori Properti

Activity diagram ini menjelaskan tentang alur penyajian informasi-informasi tempat tinggal berdasarkan semua kategori tempat tinggal baik itu jenis rumah, kontrakan atau kost-anyang disajikan pada peta digital. Kemudian terdapat menu tambahan yaitu pencarian rute terdekat dari posisi pengguna dan posisi tempat tinggal tujuan.

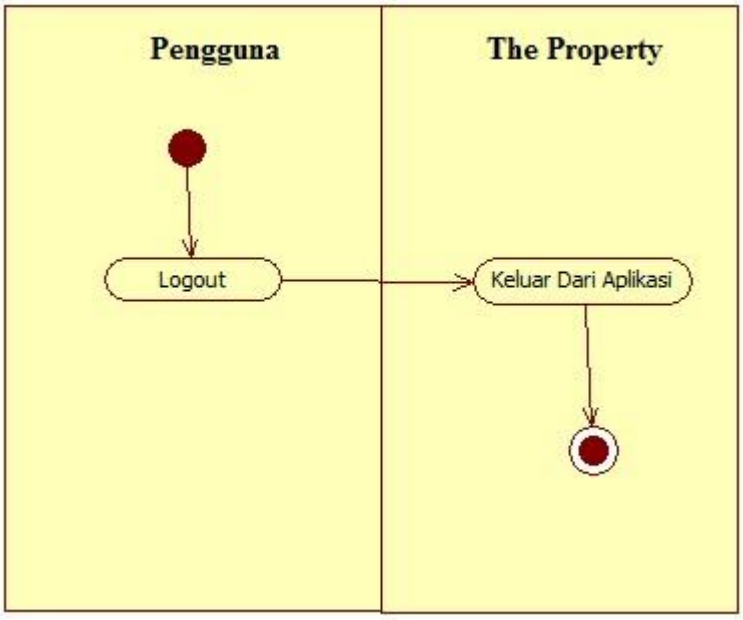
13. Activity Diagram Login



Gambar 3.20 Activity Diagram Login

Activity diagram ini menjelaskan tentang proses login dimana pengguna yang telah login, dia dapat melakukan aktivitas penjualan yaitu menambah data tempat tinggal baru.

14. Activity Diagram Logout

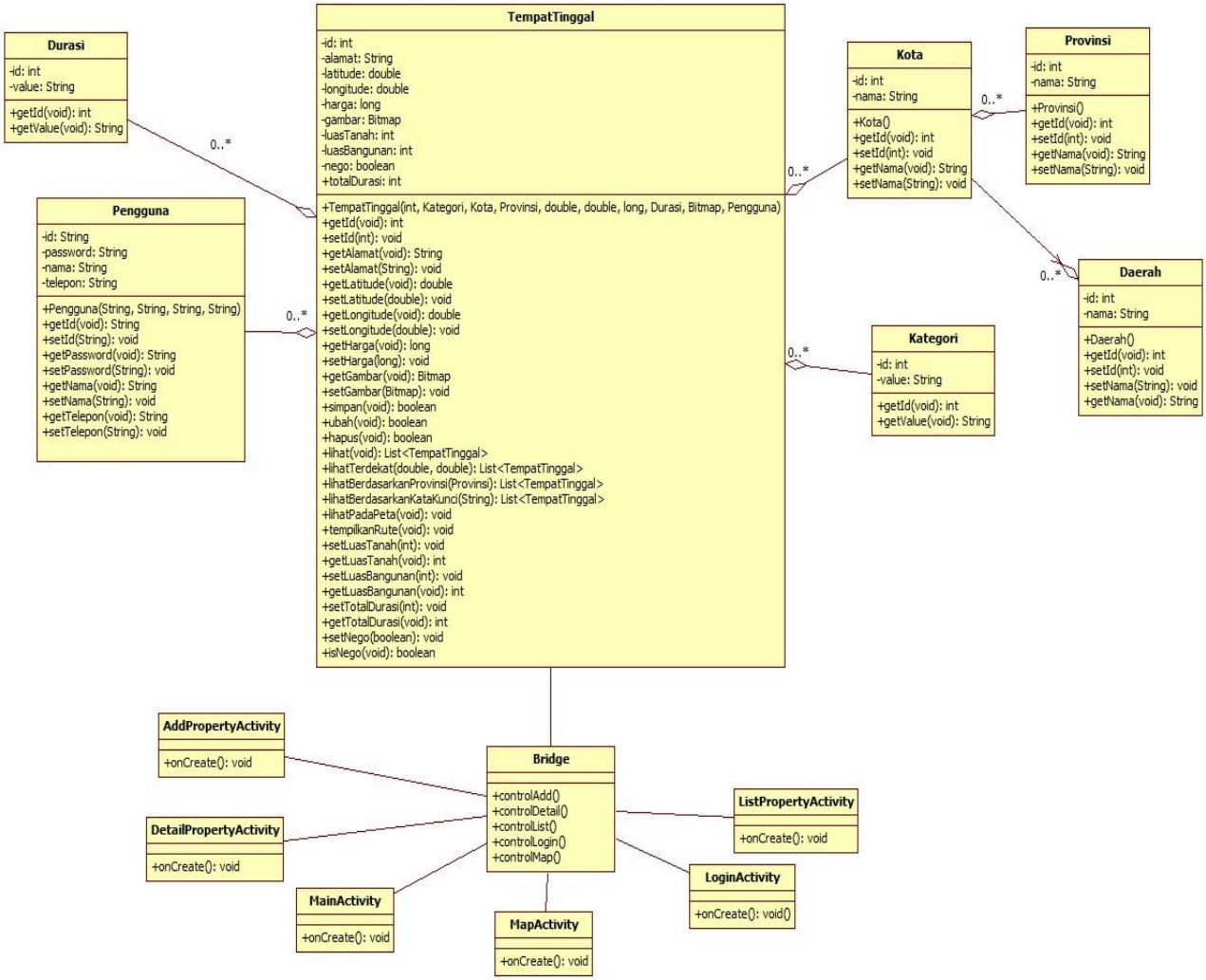


Gambar 3.21 Activity Diagram Logout

Activity diagram ini menjelaskan tentang proses berlangsungnya aktivitas logout, dimana ketika pengguna logout artinya dia keluar dari menu penjual sehingga tidak dapat menambah data penjualan baru.

c. *Class Diagram*

Class Diagram adalah sebuah spesifikasi dari fungsionalitas yang menghasilkan objek dan merupakan inti dari pengembangan aplikasi ini.Diagram ini juga merupakan gambaran keadaan atribut atau properti dari sistem yang melakukan manipulasi fungsi atau metode. Berikut ini gambaran *class* diagram dari sistem ini yang terdapat pada gambar 3.13.



Gambar 3.22*Class Diagram* Aplikasi Pencarian Tempat Tinggal

Tabel 3.20 Deskripsi *Class Diagram*

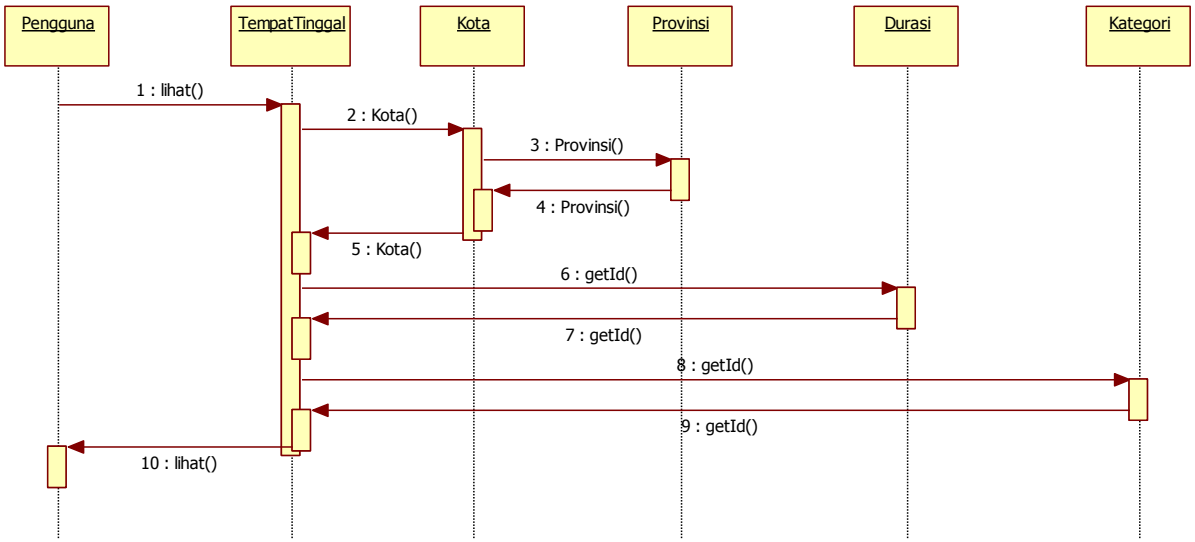
DESKRIPSI CLASS DIAGRAM		
Nama	Jenis	Deskripsi
TempatTinggal	*Model*	Struktur data dari informasi properti
Durasi	*Model*	Struktur data dari informasi durasi dari sebuah

		properti seperti tahun atau bulan
Pengguna	*Model*	Struktur data dari pengguna yang mendaftarkan diri di aplikasi
Kota	*Model*	Struktur data dari informasi kota
Provinsi	*Model*	Struktur data dari informasi provinsi
Daerah	*Model*	Struktur data dari informasi daerah yang terletak di sebuah kota
Kategori	*Model*	Struktur data dari kategori atas properti
Bridge	*Controller*	Menjadi jembatan antara model dan view agar model dan view tidak berkomunikasi secara langsung
AddPropertyActivity	*View*	Antar muka untuk menambah data properti
DetailPropertyView	*View*	Antar muka untuk melihat detil dari sebuah properti
MapActivity	*View*	Antar muka untuk peta digital yang dapat menampilkan lokasi properti
LoginActivity	*View*	Antar muka untuk login ke dalam aplikasi
ListPropertyActivity	*View*	Antar muka untuk menampilkan daftar properti dalam bentuk list
MainActivity	*View*	Antar muka untuk menampilkan menu utama

d. *Sequence Diagram*

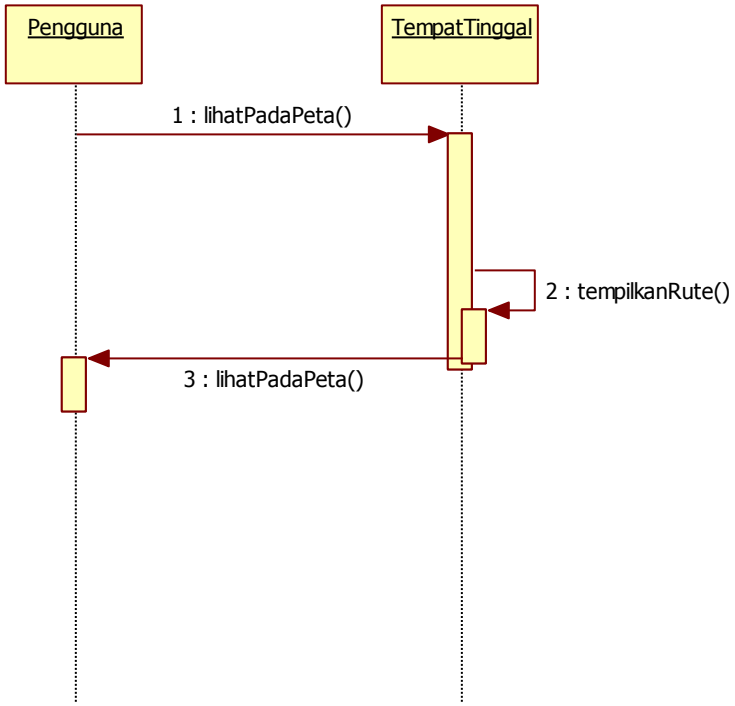
Pada tahap ini *Sequence Diagram* menjelaskan secara detil urutan proses yang dilakukan system untuk mencapai tujuan dari *Use Case*. interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.

1. Sequence Diagram Lihat Tempat Tinggal



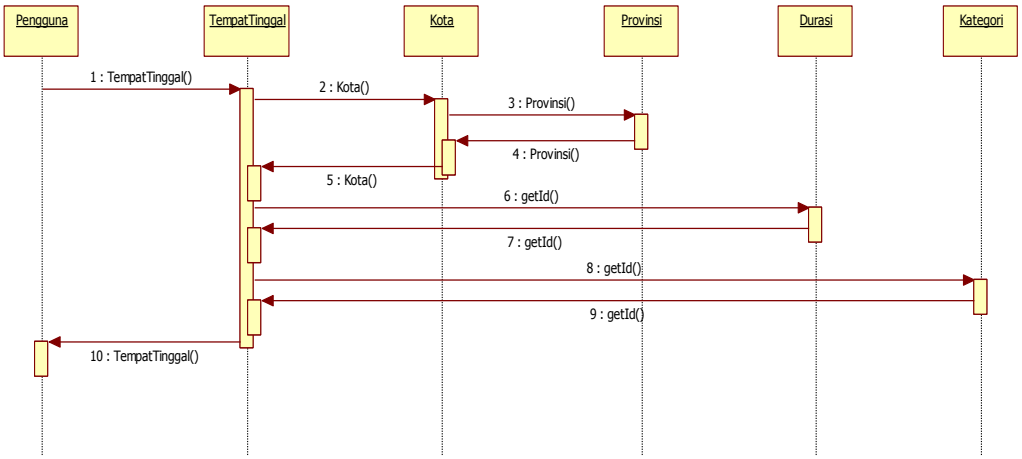
Gambar 3.23*Sequence Diagram* Lihat Tempat Tinggal

2. Sequence Diagram Lihat Tempat Tinggal Pada Peta



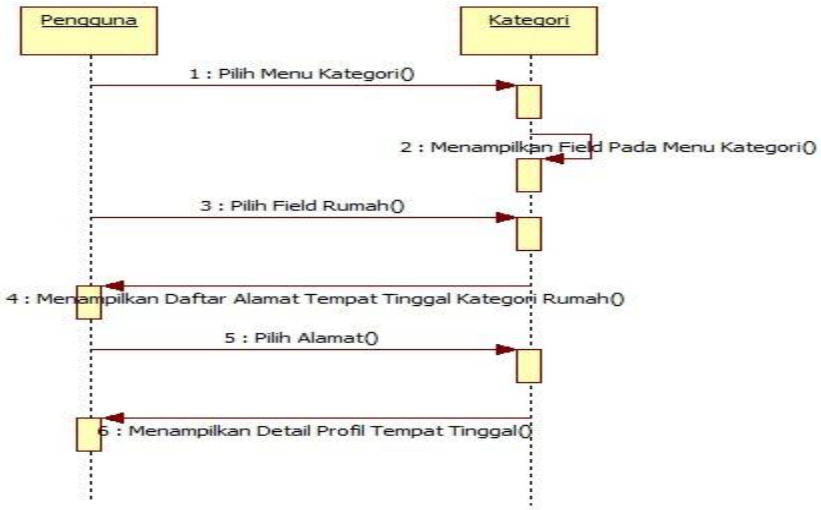
Gambar 3.15 *Sequence Diagram* Lihat Tempat Tinggal Pada Peta

3. Sequence Diagram Tambah Tempat Tinggal



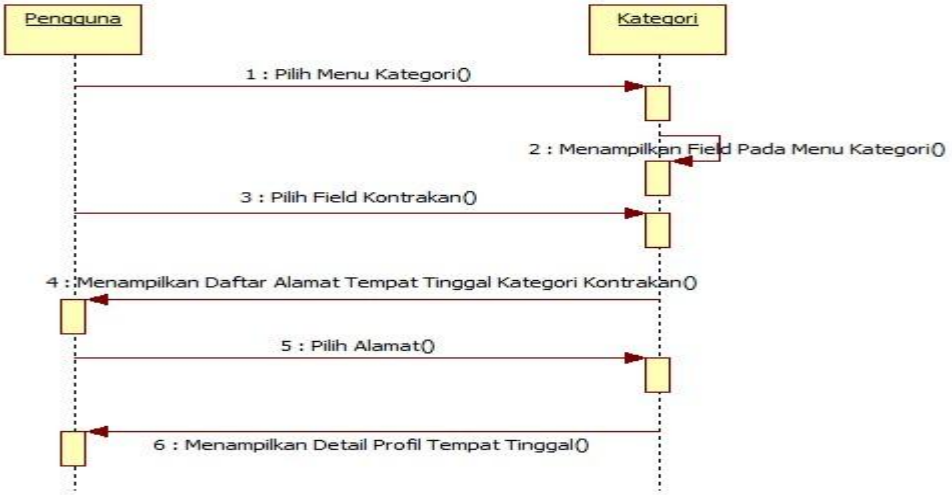
Gambar 3.24*Sequence Diagram* Tambah Tempat Tinggal

4. Sequence Diagram Menampilkan Kategori Rumah



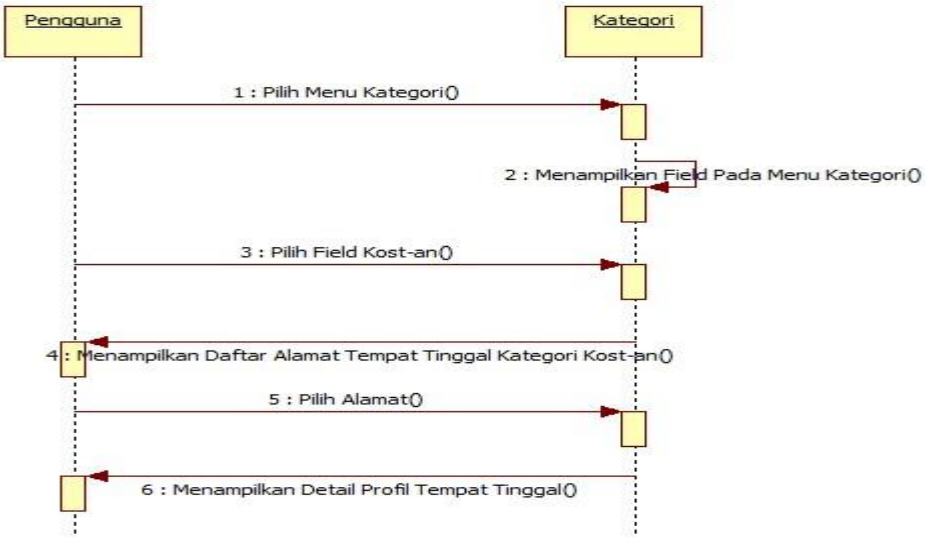
Gambar 3.25*Sequence Diagram* MenampilkanKategori Rumah

5. Sequence Diagram Menampilkan Kategori Kontrakan



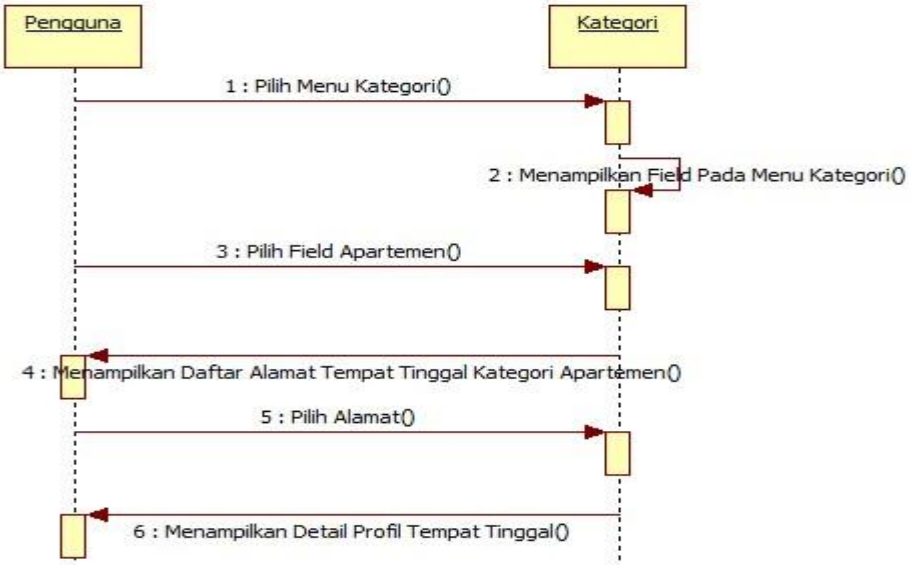
Gambar 3.26*Sequence Diagram* Menampilkan Kategori Kontrakan

6. Sequence Diagram MenampilkanKategori Kost-an



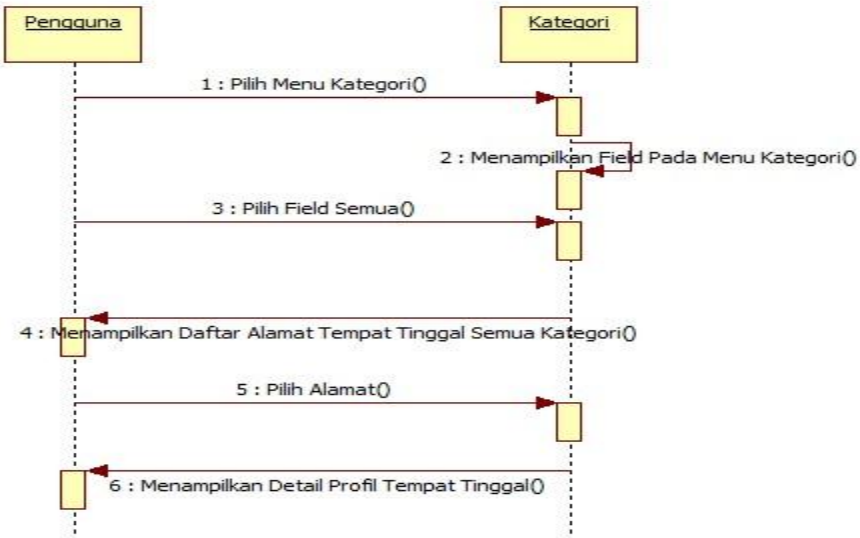
Gambar 3.27*Sequence Diagram* Menampilkan Kategori Kost-an

7. Sequence Diagram Menampilkan Kategori Apartemen



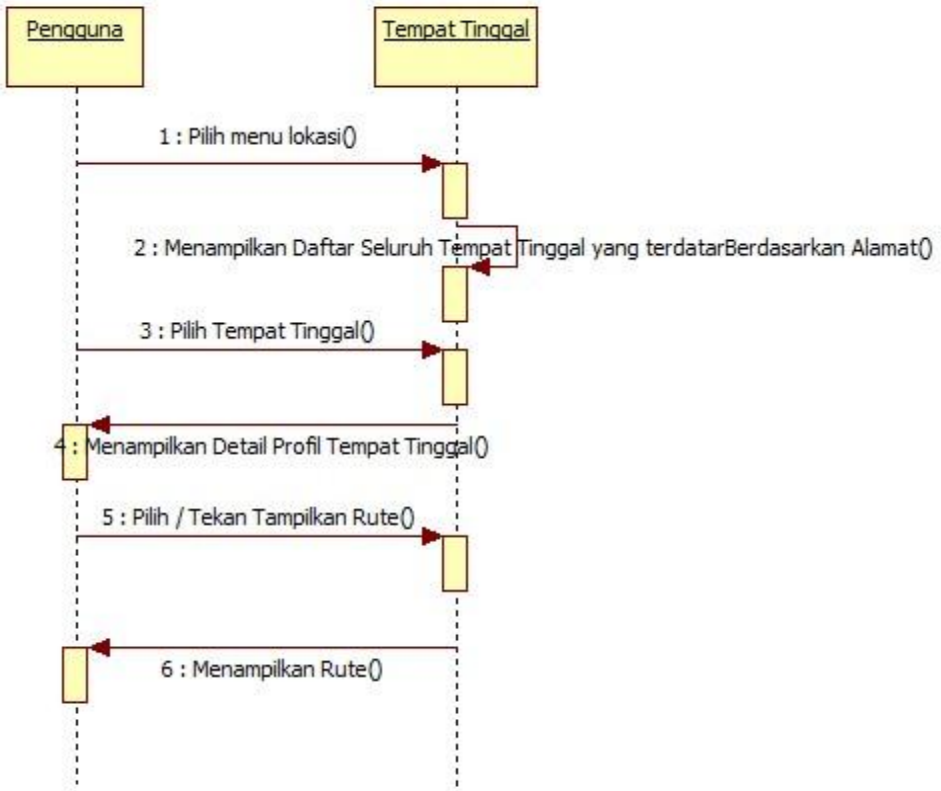
Gambar 3.28*Sequence Diagram* Menampilkan Kategori Apartemen

8. Sequence Diagram Menampilkan Semua Kategori



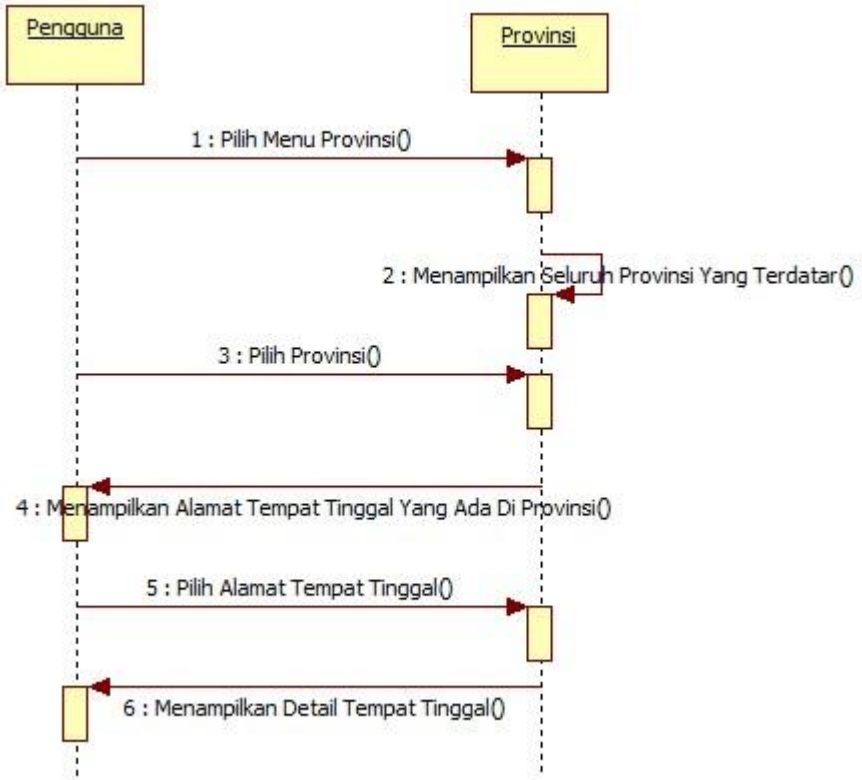
Gambar 3.29*Sequence Diagram*MenampilkanSemua Kategori

9. Sequence Diagram Menampilkan Semua List



Gambar 3.30*Sequence Diagram* Menampilkan Semua List

10. Sequence Diagram Menampilkan Berdasar Provinsi

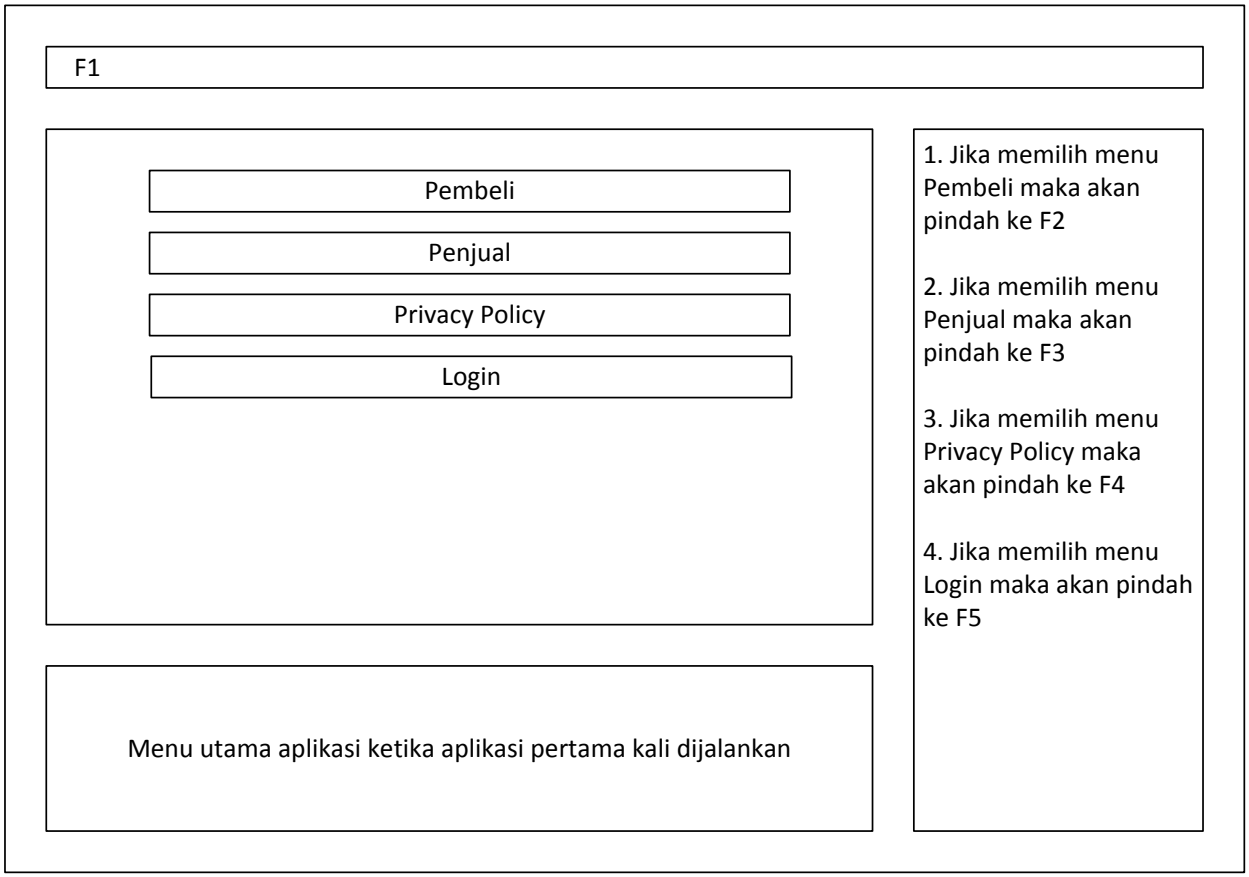


Gambar 3.31*Sequence Diagram* Menampilkan Berdasar Provinsi

3.2 Perancangan Grafis dan Antarmuka

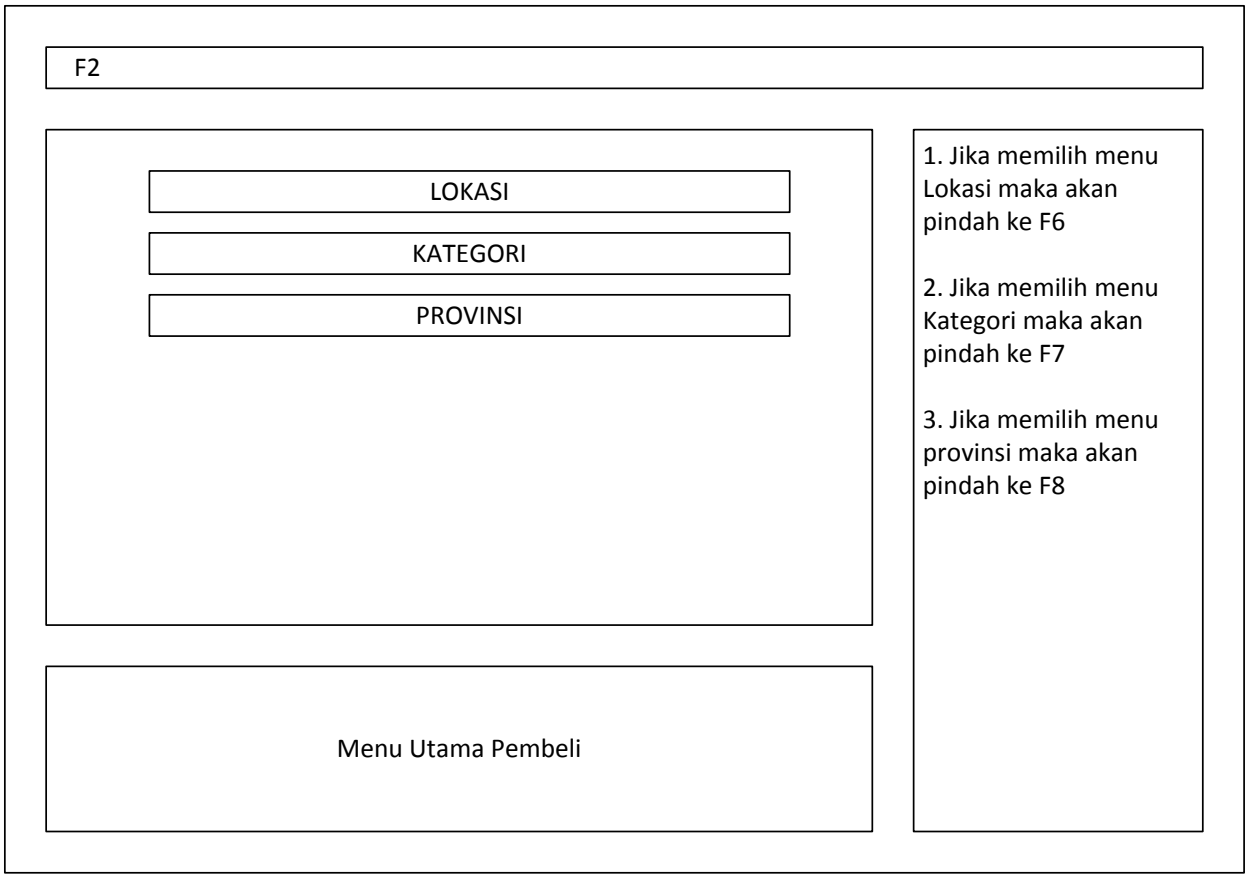
Pada tahap perancangan ini bertujuan untuk menyajikan suatu media komunikasi yang baik antara end-*user* dengan system.Dimana setiap objek-objek antar muka yang disajikan, dipertimbangkan sesuai dengan permasalahan dan kebutuhan *user*.

a. Perancangan Antar Muka Menu Utama



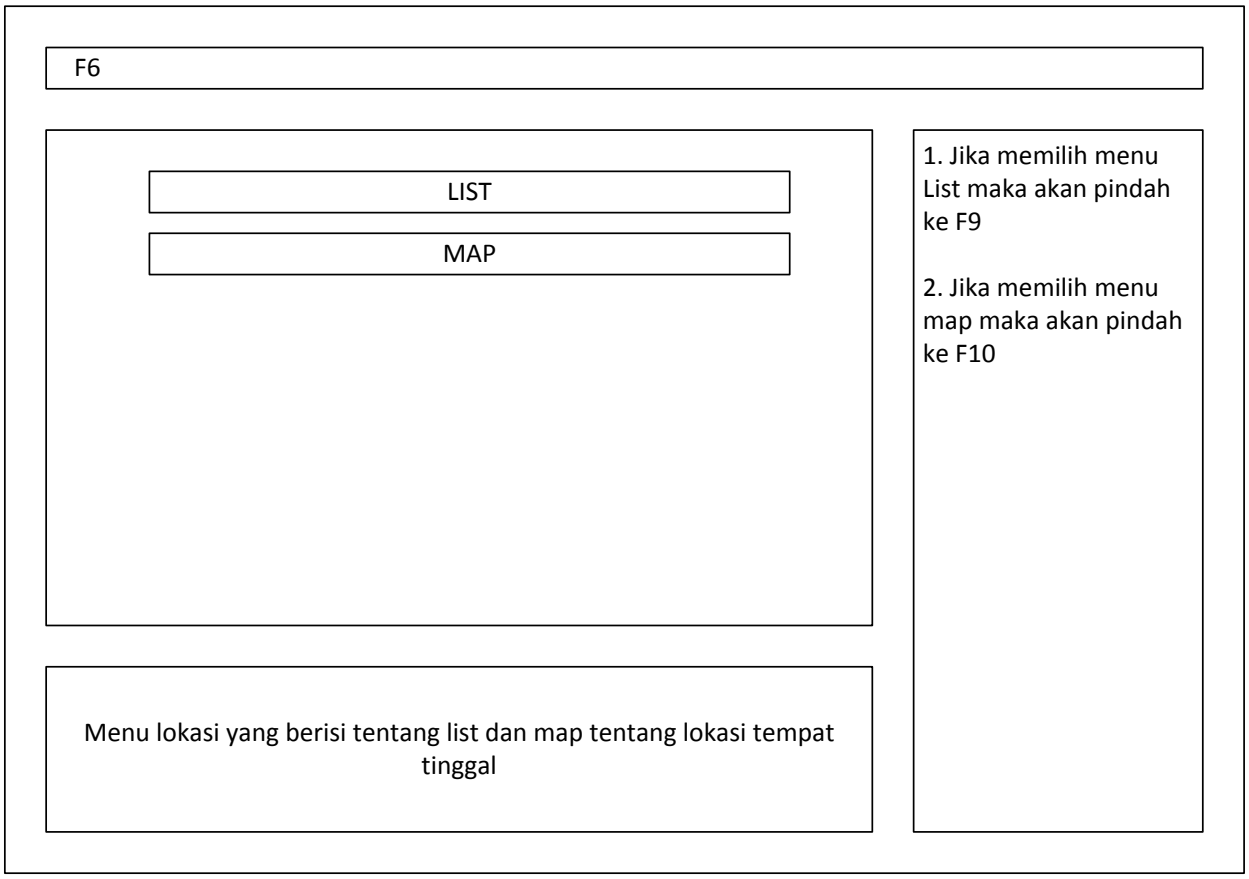
Gambar 3.32 Perancangan Antar Muka Menu Utama

b. Perancangan Antar Muka Menu Pencarian



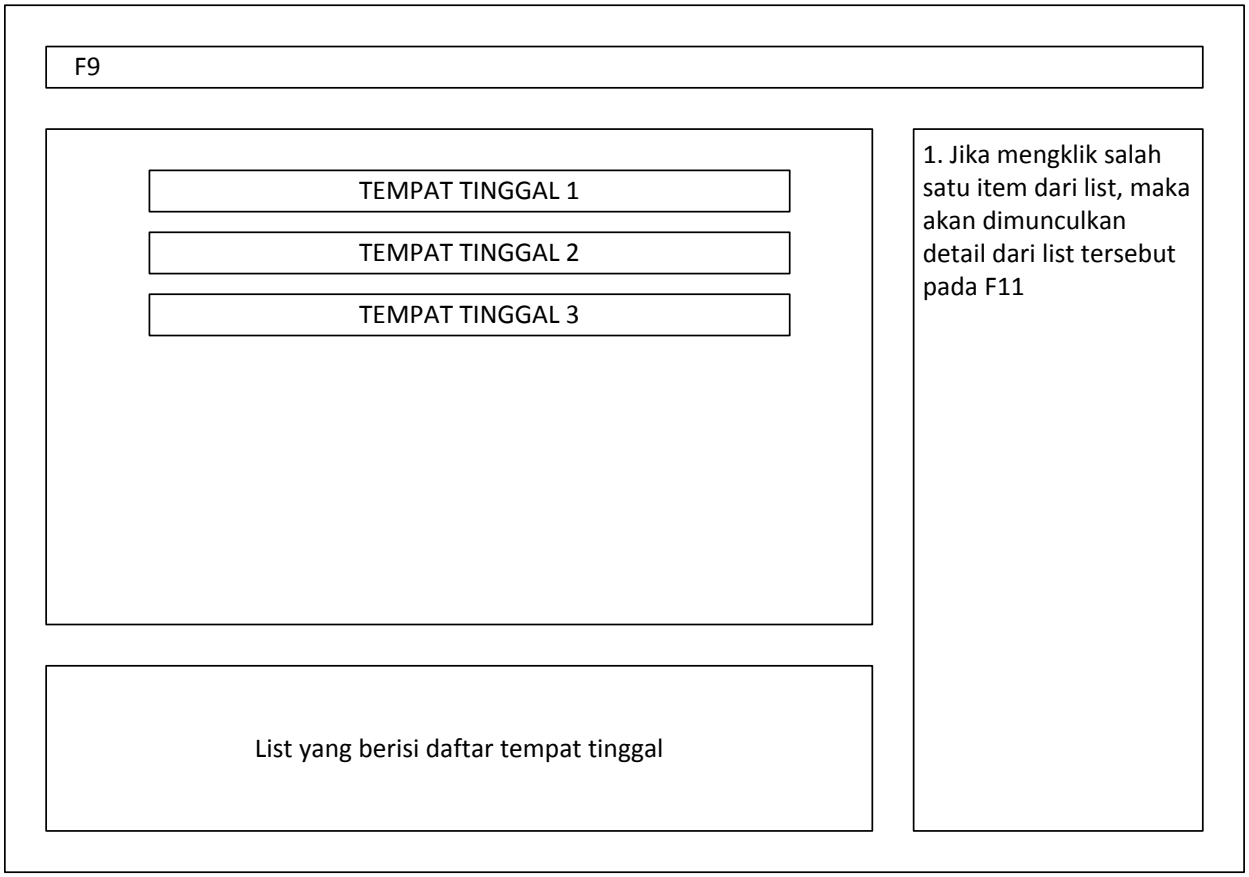
Gambar 3.33 Perancangan Antar Muka Menu Pencarian

c. Perancangan Antar Muka Jenis Media Pencarian



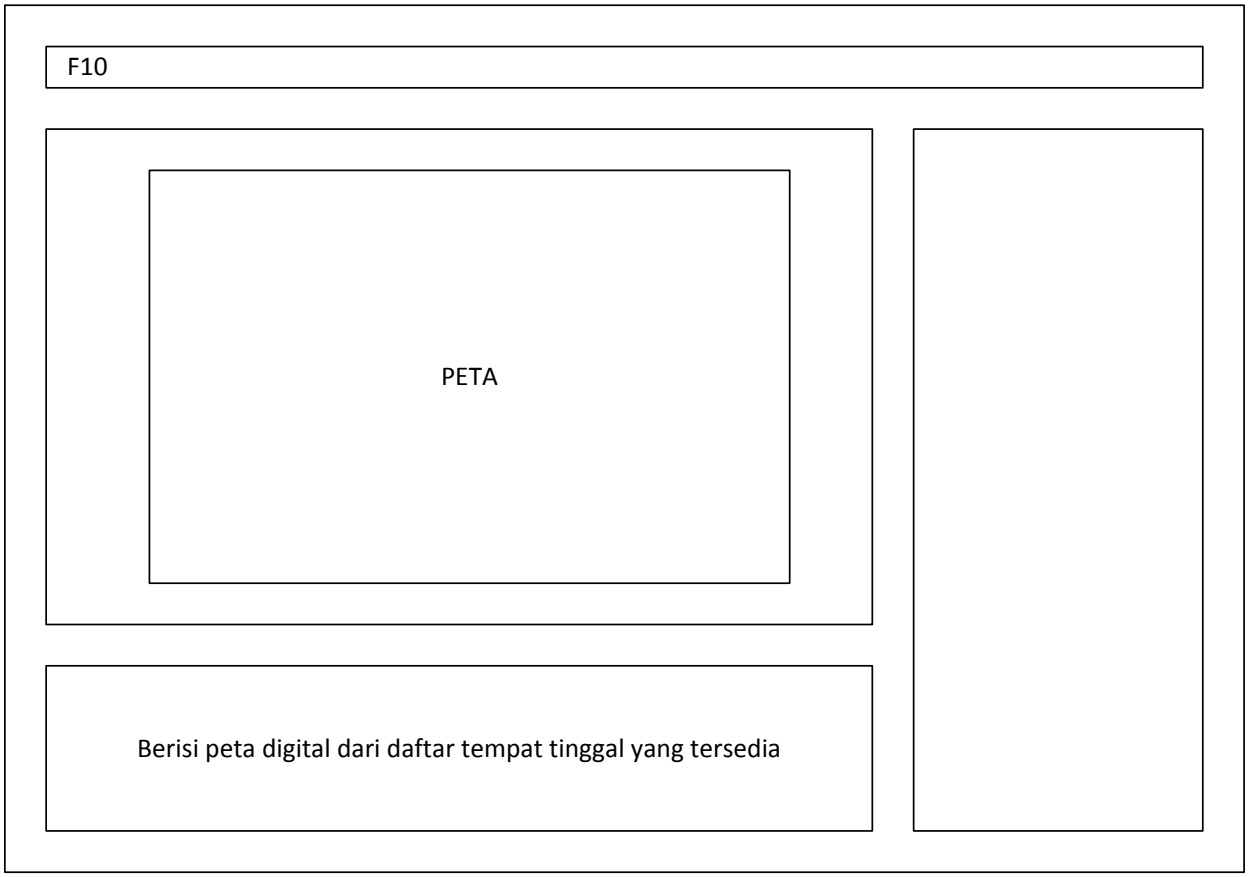
Gambar 3.34 Perancangan Antar Muka Jenis Media Pencarian

d. Perancangan Antar Muka List Tempat Tinggal



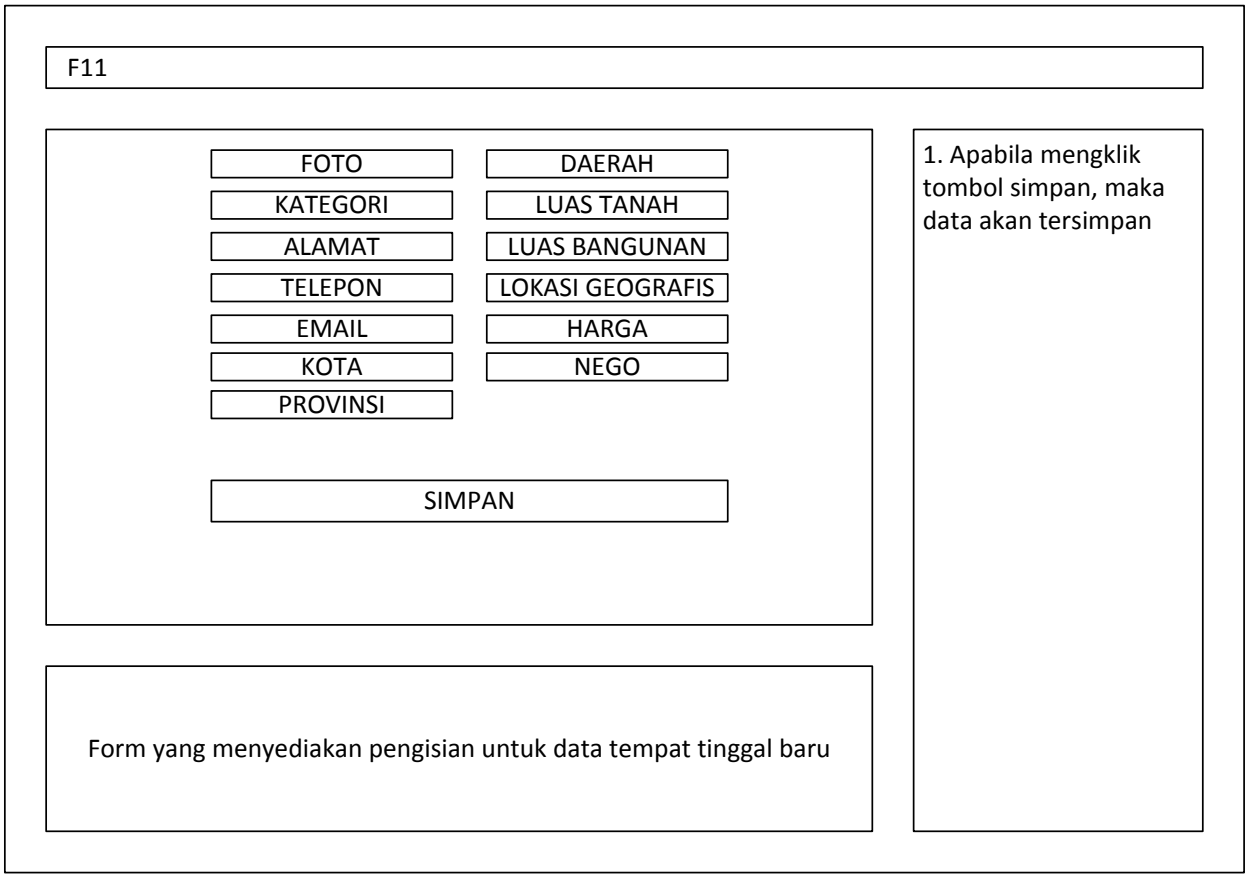
Gambar 3.35 Perancangan Antar Muka List Tempat Tinggal

e. Perancangan Antar Muka Map



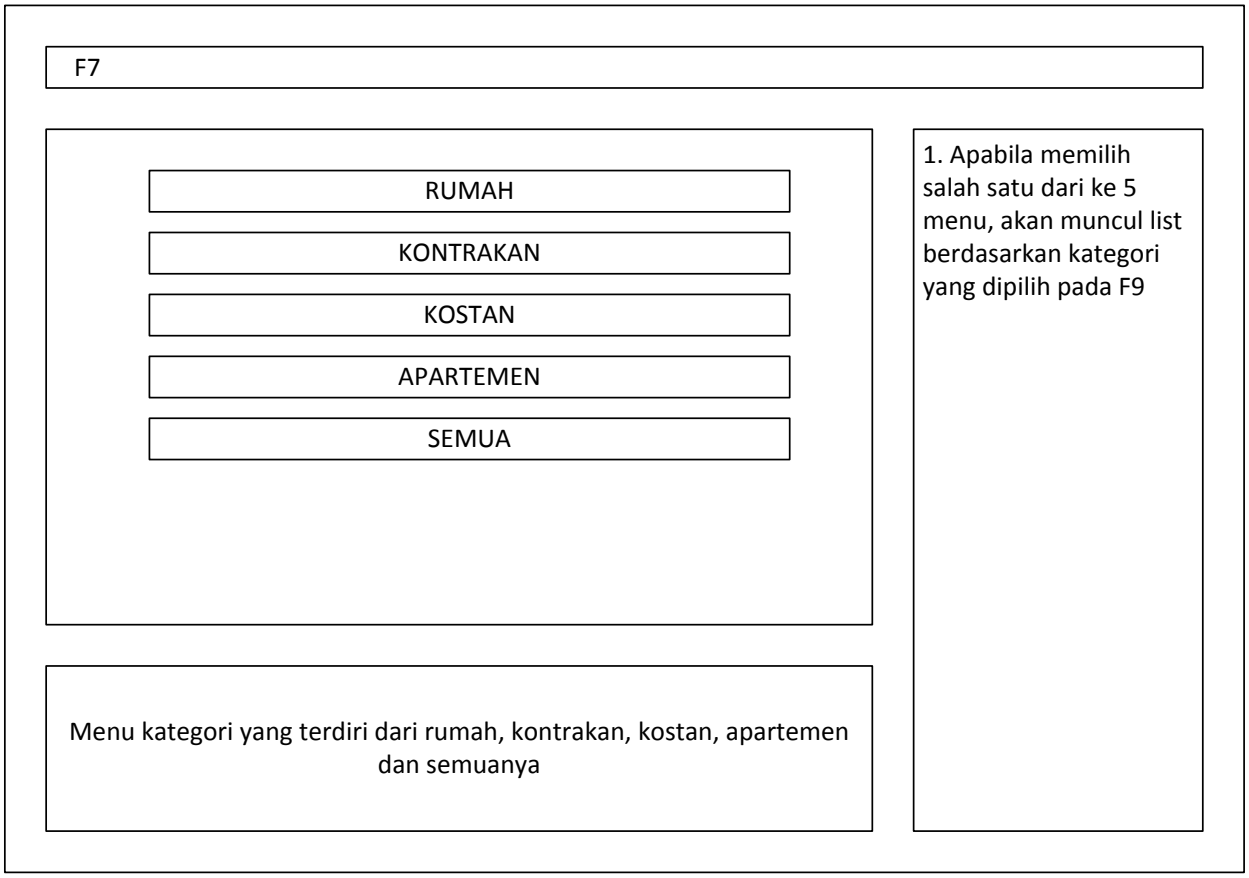
Gambar 3.36 Perancangan Antar Muka Map

f. Perancangan Antar Muka Menambah Data Baru



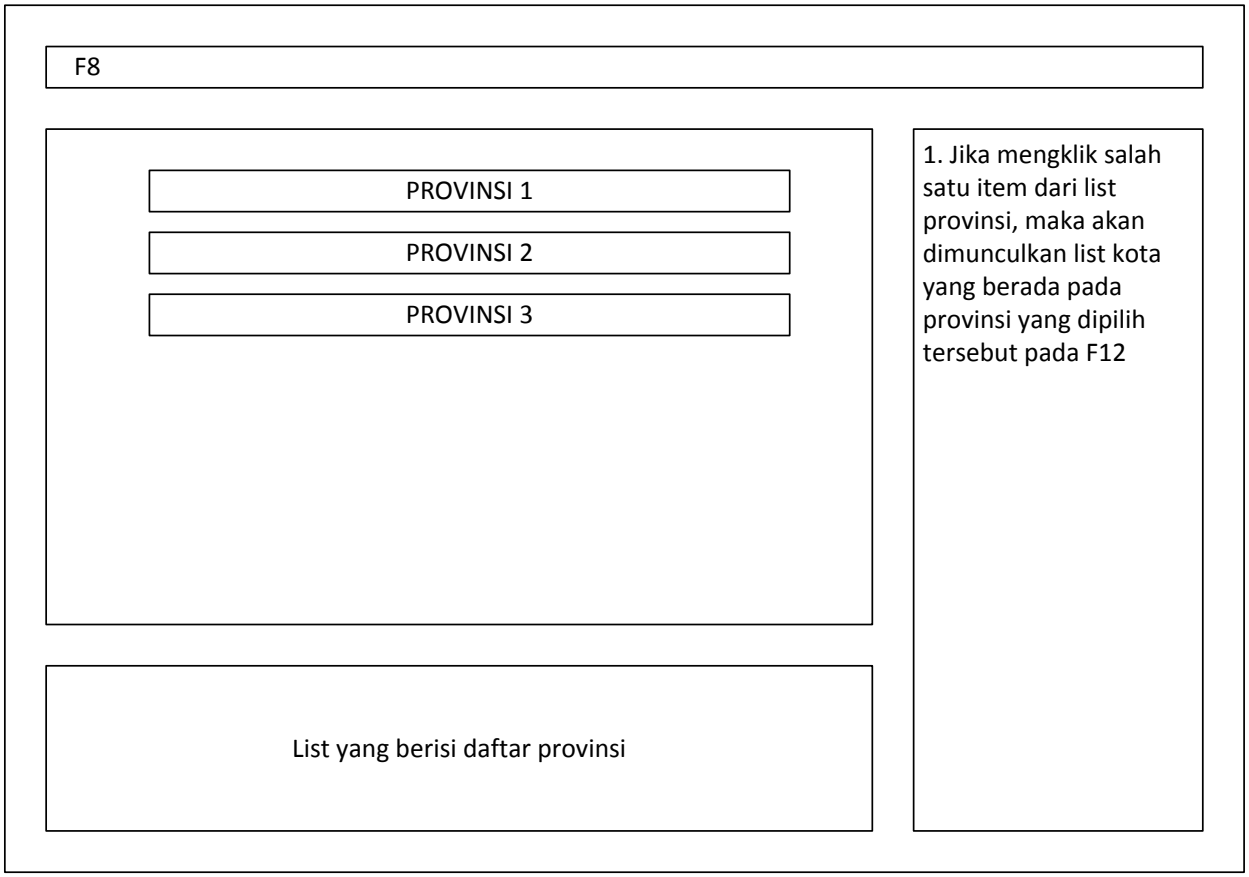
Gambar 3.37 Perancangan Antar Muka Menambah Data Baru

g. Perancangan Antar Muka Kategori



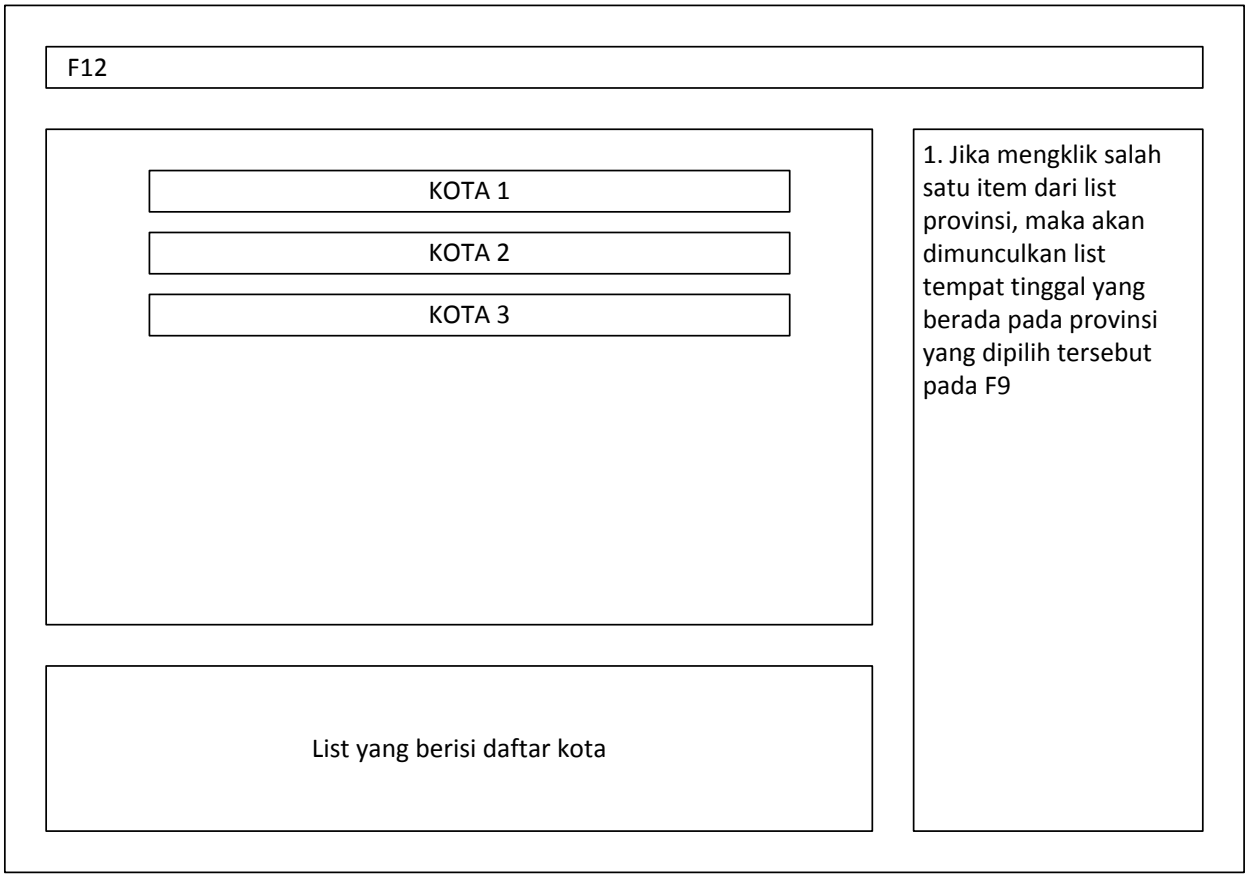
Gambar 3.38 Perancangan Antar Muka Kategori

h. Perancangan Antar Muka Pencarian Berdasarkan Provinsi



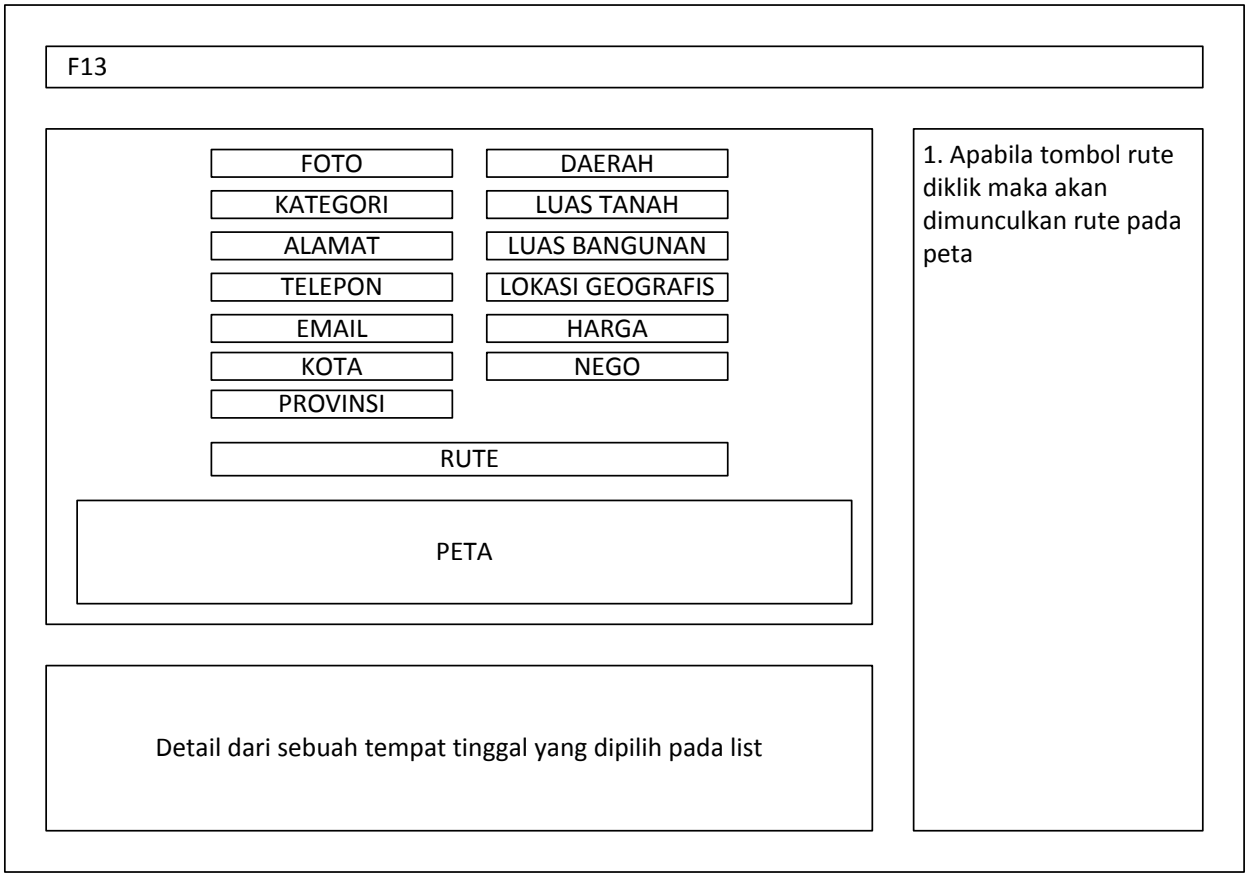
Gambar 3.39 Perancangan Antar Muka Pencarian Berdasarkan Provinsi

i. Perancangan Antar Muka Pencarian Berdasarkan Kota



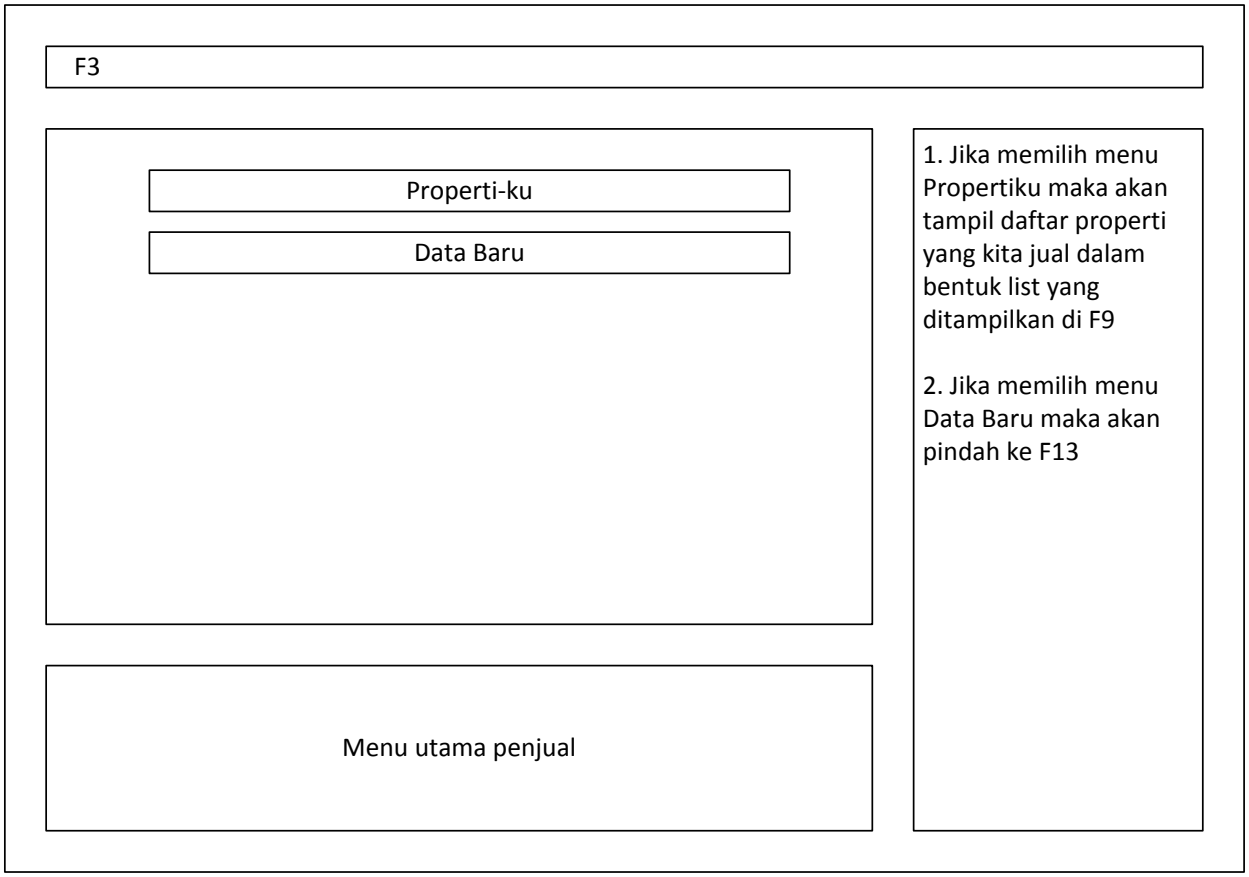
Gambar 3.40 Perancangan Antar Muka Pencarian Berdasarkan Kota

j. Perancangan Antar Muka Detail Tempat Tinggal



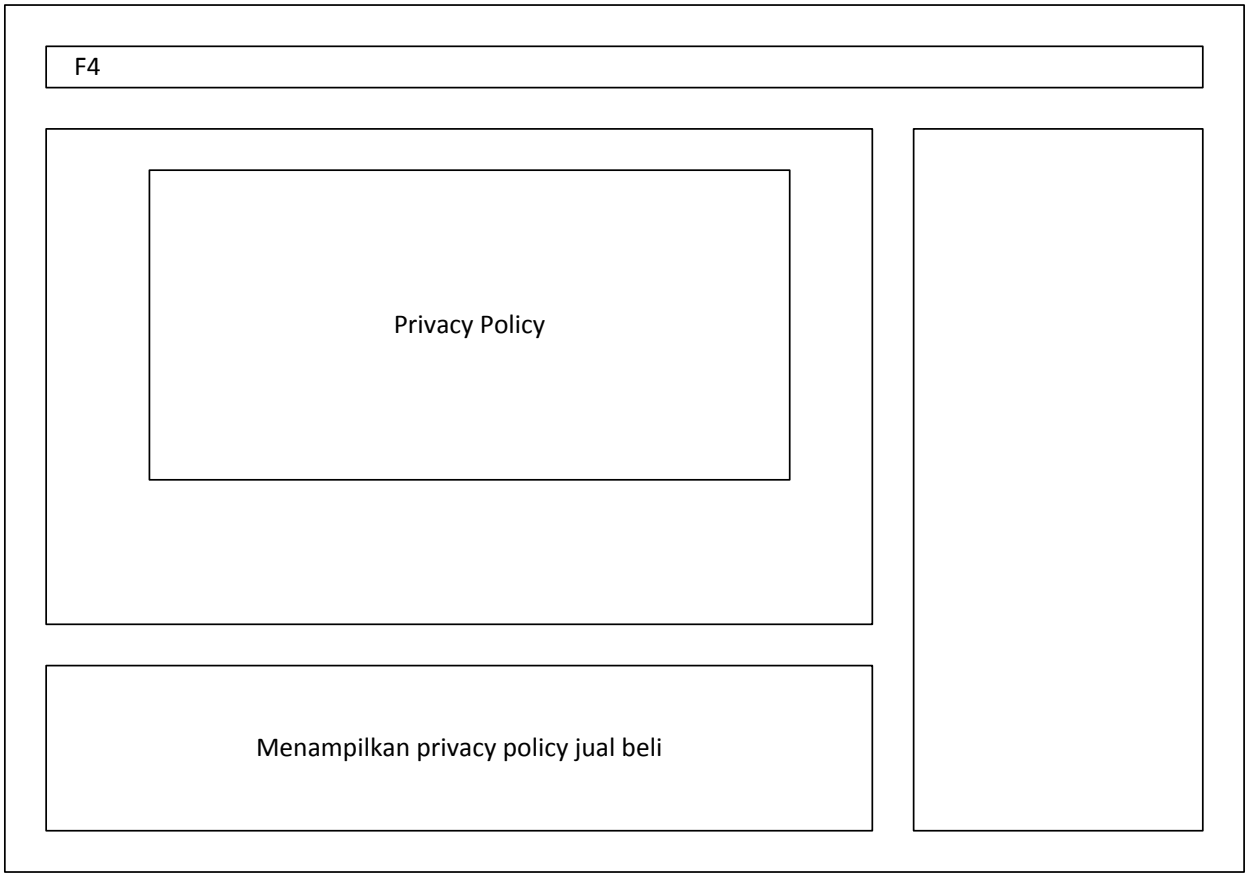
Gambar 3.41 Perancangan Antar Muka Detail Tempat Tinggal

k. Perancangan Antar Muka Menu Penjual



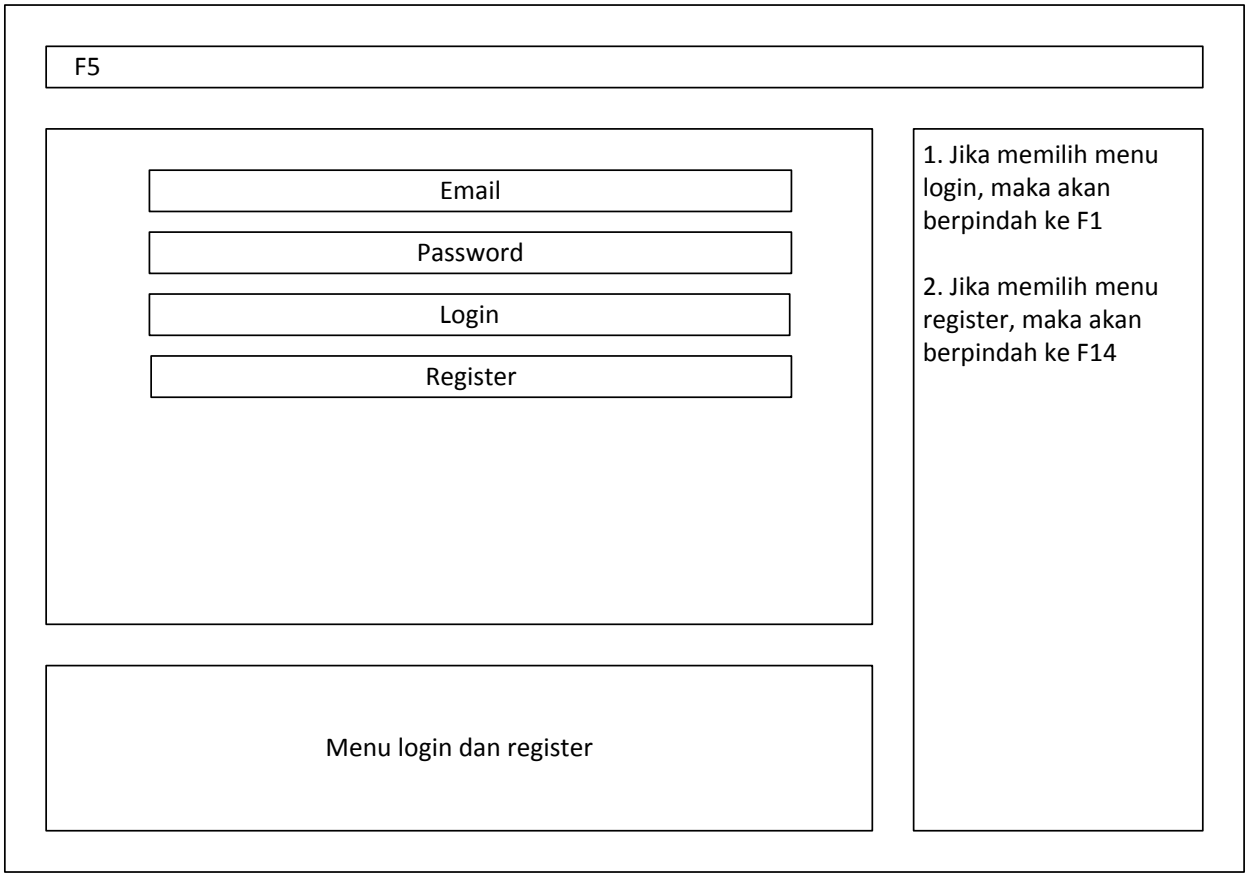
Gambar 3.42 Perancangan Antar Muka Menu Penjual

l. Perancangan Antar Muka Privacy Policy



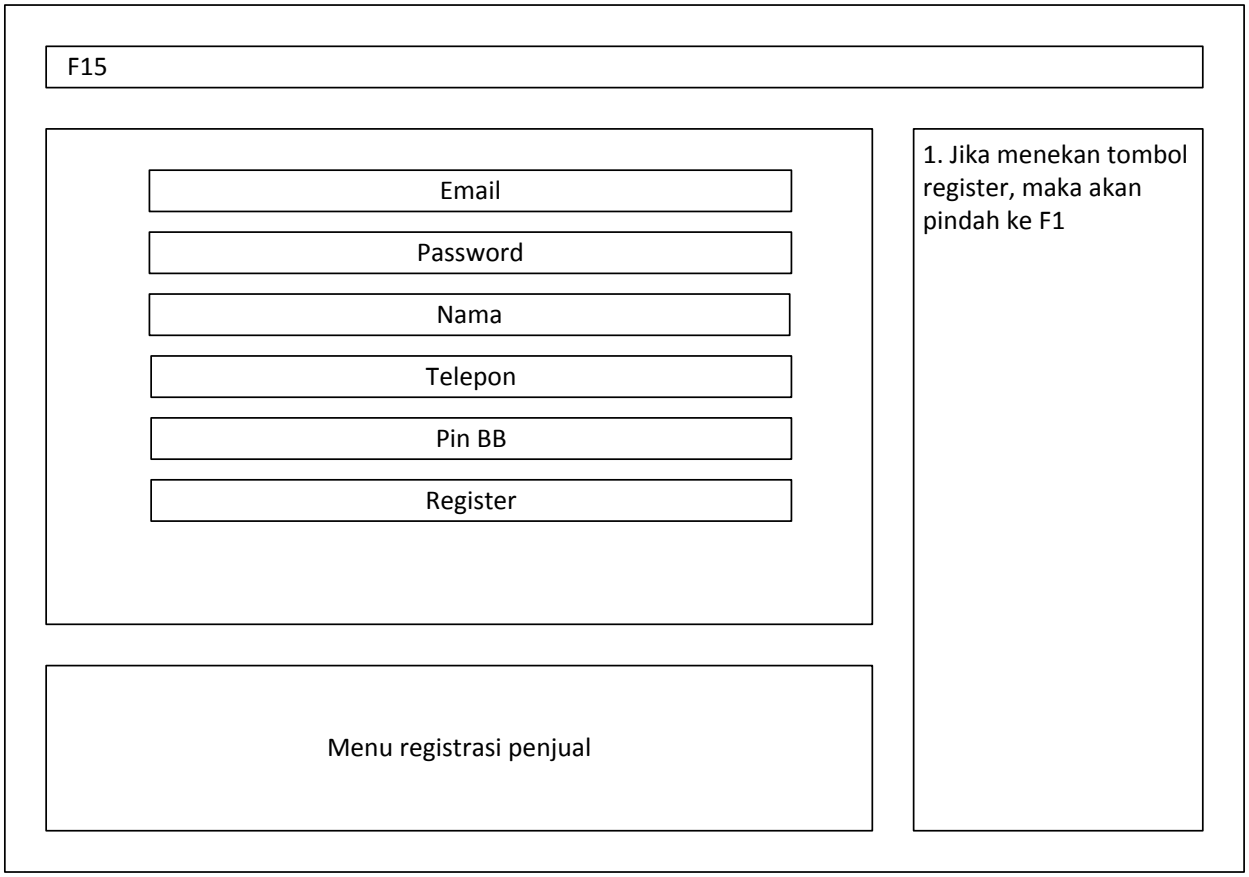
Gambar 3.43 Perancangan Antar Muka Privacy Policy

m. Perancangan Antar Muka Login



Gambar 3.44 Perancangan Antar Muka Login

n. Perancangan Antar Muka Registrasi



Gambar 3.45 Perancangan Antar Muka Registrasi