# BAB V

# IMPLEMENTASI DAN PENGUJIAN SISTEM

## 5.1. Implementasi

Implementasi merupakan akhir dari sebuah pengembangan sistem informasi. Implementasi sistem tentunya mengacu pada rancangan sistem yang telah dibuat. Pada umumnya untuk implementasi sebuah aplikasi diperlukan konfigurasi pada perangkat lunak serta perangkat keras sesuai kebutuhan sistem supaya sistem informasi bisa berjalan dengan semestinya.

### 5.1.1. Batasan Implementasi

Dalam pengimplementasian perangkat lunak pemantauan dan pengendalian aplikasi pelacakan kiriman berbasis *mobile* pada layanan Pos *Express* di PT. POS INDONESIA (PERSERO) ini terdapat beberapa hal yang menjadi batasan implementasi, yaitu :

1. Penulis merancang sebuah aplikasi pelacakan kiriman berbasis *mobile* menggunakan jaringan *internet* dikarenakan aplikasi yang akan dijalankan di-*device* Android membutuhkan koneksi untuk mengakses informasi kiriman yang ada di *webserver*. Saat perancangan dan pembangunan aplikasi ini basis data yang digunakan dalam pengimplementasian ini adalah MySQL.
2. Pembuatan aplikasi pelacakan kiriman berbasis *mobile* yang berfungsi sebagai *client* pada *device* Android menggunakan bahasa pemrograman

java dan pada Aplikasi Simulasi I-POS yang berfungsi sebagai *server* menggunakan bahasa pemograman PHP.

3. Tampilan pada aplikasi pelacakan kiriman berbasis *mobile* menggunakan bahasa Indonesia dan bahasa Inggris.

### 5.1.2. Implementasi Perangkat Lunak

Dalam Implementasi Aplikasi Pelacakan Kiriman Berbasis *Mobile* pada layanan Pos *Express* harus didukung oleh perangkat lunak yang digunakan agar aplikasi ini berjalan sebagaimana mestinya. Berikut perangkat lunak yang dibutuhkan.

1. Spesifikasi Untuk *Client* (lokal *server* saat perancangan dan pembangunan)
   a. Aplikasi dijalankan *virtual device* atau biasa disebut *emulator*.
   b. Lokal *server* dijadikan sebagai localhost, yaitu Webserver.
   c. Koneksi emulator Andorid versi 2.2 dengan local *server* dengan menggunakan http connection emulator yaitu http://10.0.2.2/pos/seePackage.php.
2. Spesifikasi untuk Client Jaringan Internet (aplikasi sudah siap diimplementasikan)
   a. Aplikasi dijalankan di sistem operasi minimal Android *versi* 2.2
   b. Pengaksesan koneksi menggunakan ISP (*Internet Service Provider).*
   c. Internet sebagai media akses (kependekan dari interconnected-networking) ialah sistem global dari seluruh jaringan komputer yang

saling terhubung menggunakan standar Internet Protocol Suite (TCP/IP) untuk melayani miliaran pengguna di seluruh dunia.

3. Spesifikasi untuk Server
    a. Webhosting sebagai layanan penyewaan ruang simpan untuk menyimpan data-data agar bisa diakses dari mana saja dan kapan saja
    b. *Domain*. Penggunaan domain pada aplikasi yaitu http://simulasi-ipos.net78.net.

### 5.1.3. Implementasi Perangkat Keras

Mengenai implementasi perangkat keras, dalam tahap ini didefinisikan sekumpulan perangkat keras untuk mendukung Aplikasi pelacakan kiriman berbasis  supaya berjalan dengan semestinya. Pada dasarnya tidak ada perangkat keras khusus yang dibutuhkan sistem. Konfigurasi dari perangkat keras pada *device mobile*, diantaranya:

1. Processor 800MHz
2. Memory 4GB
3. RAM 512 MB.
4. Perangkat Jaringan (Speed HSDPA, 7.2 Mbps)

### 5.1.4. Implementasi Basis Data

Pada bagian implementasi basis data akan diuraikan seperti apa sintaksnya hasil dari rancangan basisdata. Nama basis data yang dibuat untuk aplikasi Pelacakan Kiriman Berbasis *Mobile* ini adalah **“pt_pos”**.

**1. Sintaks Pembuatan Database**

```
CREATE  DATABASE pt_pos;
```

**2. Sintaks detail_lokasi_pos**

```
CREATE TABLE `daftar_lokasi_pos` (
    `id_pos` varchar(10) NOT NULL,
    `alamat` text NOT NULL,
     `latitudes` varchar(50) NOT NULL,
    `longitudes` varchar(50) NOT NULL,
     `kode_pos` varchar(15) default NULL,
    `kantor_pos` varchar(50) NOT NULL,
     `propinsi` varchar(100) NOT NULL,
     PRIMARY KEY  (`id_pos`),
     UNIQUE KEY `kode_pos` (`kode_pos`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

**3. Sintaks detail_barang**

```
CREATE TABLE `detail_barang` (
    `id_barang` varchar(10) NOT NULL,
     `nama_barang` varchar(50) NOT NULL,
    `berat` float NOT NULL,
    `bea_kirim` float NOT NULL,
      `id_pengirim` varchar(10) NOT NULL,
    `id_penerima` varchar(10) NOT NULL,
    PRIMARY KEY  (`id_barang`),
     KEY `id_pengirim` (`id_pengirim`),
```

```
  KEY `id_penerima` (`id_penerima`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**4. Sintaks detail_transaksi**

```
CREATE TABLE `detail_transaksi` (
        `idResi` varchar(25) NOT NULL,
      `id_kantung` varchar(25) NOT NULL,
       `waktu_kirim` datetime NOT NULL,
      `waktu_terima` datetime default NULL,
      `id_karyawan` varchar(10) NOT NULL,
      `penerima2` varchar(50) default NULL,
       `id_barang` varchar(10) NOT NULL,
       `status` varchar(20) NOT NULL,
       `id_pos` varchar(10) NOT NULL,
       `latitudes` varchar(50) default NULL,
      `longitudes` varchar(50) default NULL,
      `lokasi2` text,
      PRIMARY KEY  (`idResi`),
       KEY `id_kantung` (`id_kantung`),
        KEY `id_barang` (`id_barang`),
      KEY `id_karyawan` (`id_karyawan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**5. Sintaks karyawan**

```
CREATE TABLE `karyawan` (
    `id_karyawan` varchar(10) NOT NULL,
    `nama_karyawan` varchar(50) NOT NULL,
    `username` varchar(16) NOT NULL,
    `password` varchar(16) NOT NULL,
    `alamat` text,
    `telp` varchar(16) default NULL,
    PRIMARY KEY  (`id_karyawan`),
    UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**6. Sintaks master_transaksi**

```
CREATE TABLE `master_transaksi` (
    `id_kantung` varchar(25) NOT NULL,
    `wilayah` varchar(100) NOT NULL,
    `status_ktg` varchar(15) NOT NULL,
    `id_pos` varchar(10) NOT NULL,
    PRIMARY KEY  (`id_kantung`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**7. Sintaks penerima**

```
CREATE TABLE `penerima` (
    `id_penerima` varchar(10) NOT NULL,
    `penerima` varchar(50) NOT NULL,
    `alamat` text NOT NULL,
    `kota` varchar(30) NOT NULL,
    `telp_fax` varchar(16) NOT NULL,
     PRIMARY KEY  (`id_penerima`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**8. Sintaks pengirim**

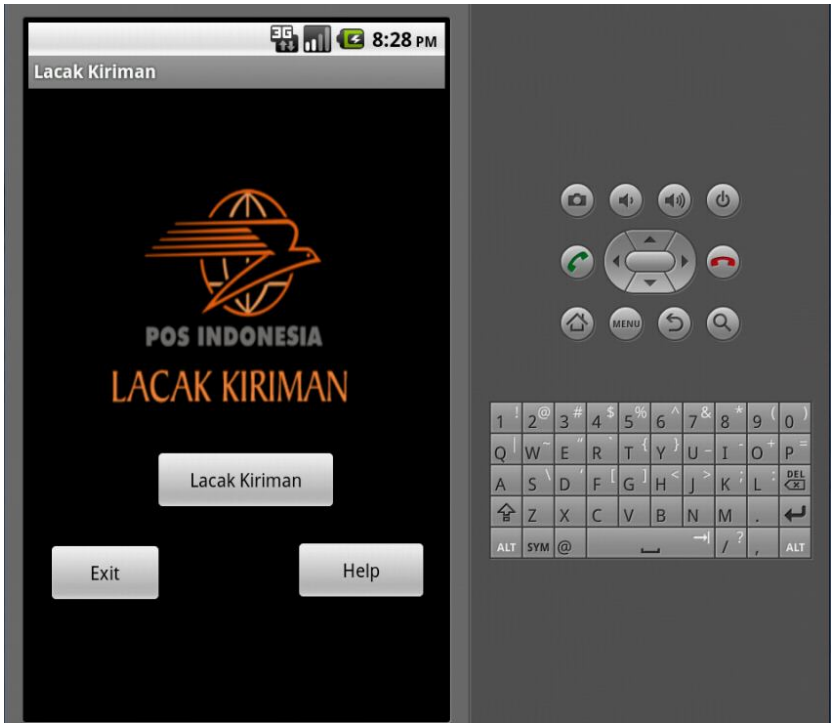
```
CREATE TABLE `pengirim` (
    `id_pengirim` varchar(10) NOT NULL,
    `nama_pengirim` varchar(50) NOT NULL,
     `alamat` text NOT NULL,
      `kota` varchar(30) NOT NULL,
     `telp_fax` varchar(16) NOT NULL,
    PRIMARY KEY  (`id_pengirim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

### 5.1.5. Implementasi AntarMuka

Pada tahap ini akan digambarkan model antar muka untuk pengoperasian sistem yang dapat dilihat langsung oleh pengirim. Rincian dari antar muka ini adalah sebagai berikut.

1. **Tampilan AntarMuka Halaman Utama Lacak Kiriman**

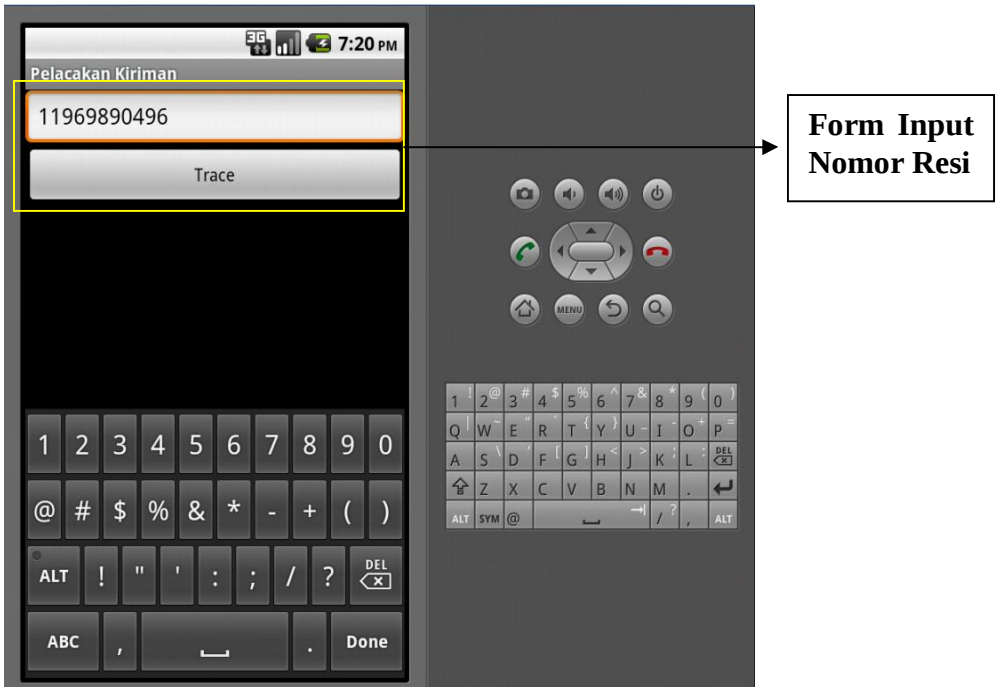
   Tampilan antarmuka halaman utama lacak kiriman merupakan tampilan awal Aplikasi Pelacakan Kiriman untuk melacak kiriman, disini terdapat button lacak kirim untuk masuk ke halaman melacak kiriman, help digunakan untuk mengetahui cara penggunaan aplikasi dan exit untuk keluar dari aplikasi. Nama *file* pada aplikasi ini adalah index.java.




**Gambar 5.1** Tampilan AntarMuka Halaman Utama Lacak Kiriman

## 2. Tampilan AntarMuka Melacak Kiriman

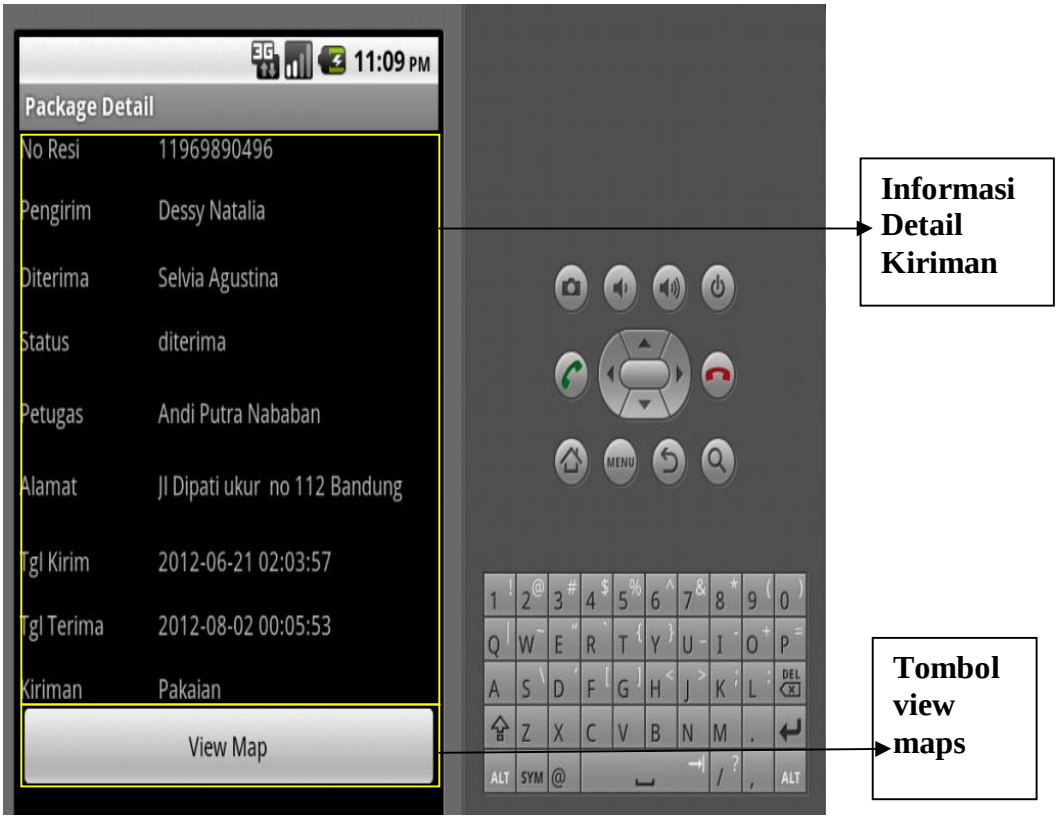
Tampilan antarmuka melacak kiriman merupakan tampilan awal Aplikasi Pelacakan Kiriman untuk melacak kiriman, disini terdapat form *input* data kiriman berupa nomor resi. Nama *file* pada aplikasi ini adalah LacakNomorResi.java.




**Gambar 5.2** Tampilan AntarMuka Melacak Kiriman

### 3. Tampilan AntarMuka Detail Informasi Kiriman

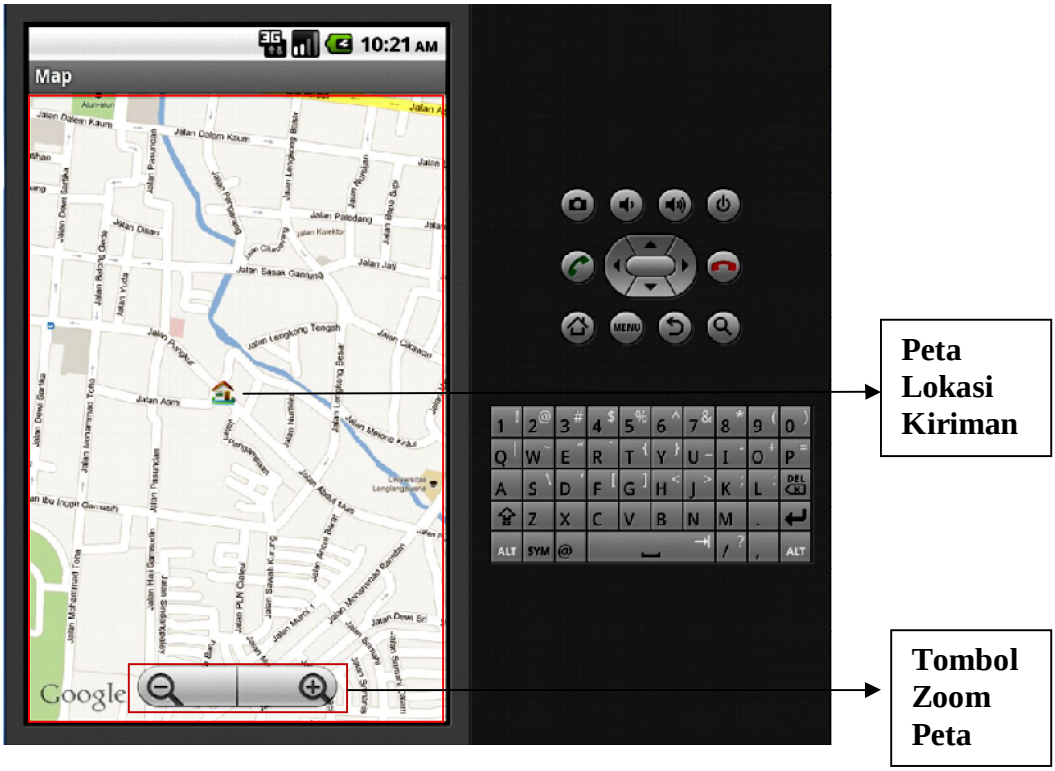
Tampilan antarmuka detail informasi kiriman merupakan detail informasi kiriman dari data kiriman (nomor resi) yang diinputkan di form *tracking* kiriman, disini terdapat detail kiriman dan tombol untuk melihat peta lokasi kiriman. Nama *file* pada aplikasi ini DetailKirimanActivity.java




**Gambar 5.3** Tampilan AntarMuka Detail Informasi Kiriman

## 4. Tampilan AntarMuka Map

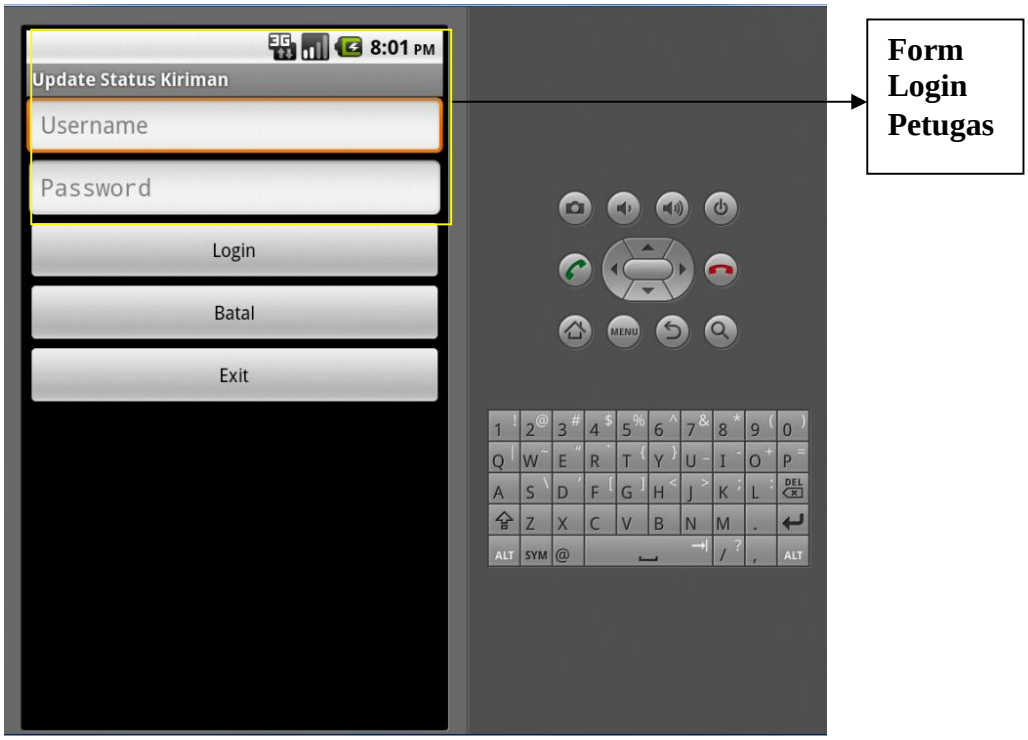
Tampilan antarmuka map merupakan tampilan peta pada lokasi kiriman, tampilan ini memberikan informasi kiriman yang lebih spesifik kepada pengirim. Nama *file* pada aplikasi ini adalah MapActivity.java.




**Gambar 5.4** Tampilan AntarMuka Map

## 5. Tampilan AntarMuka Login

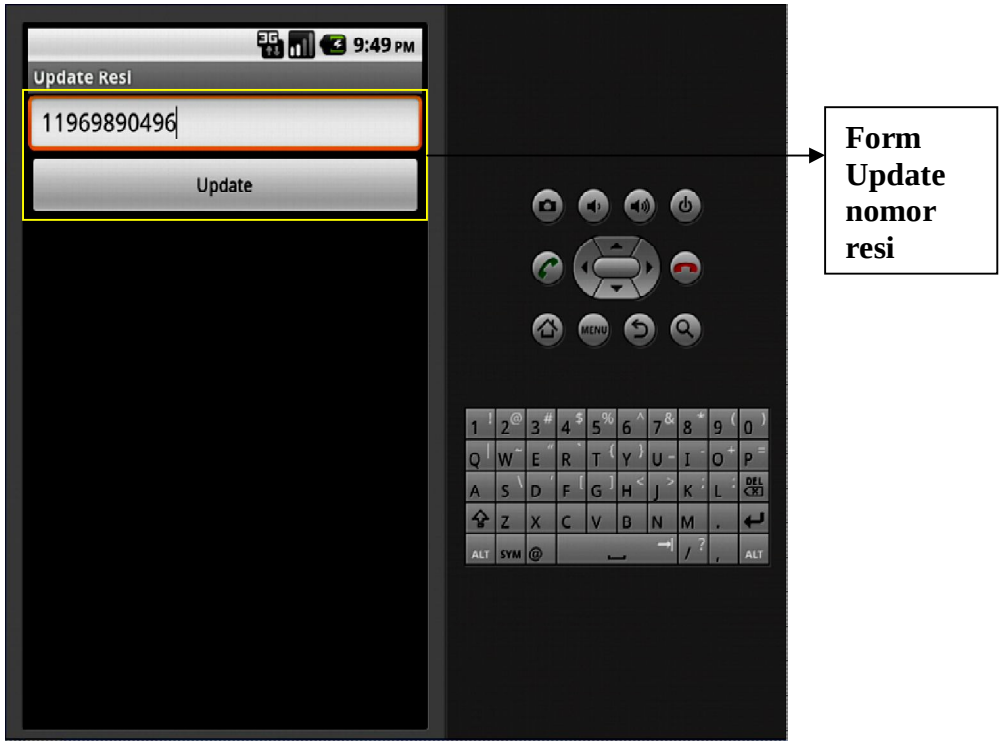
Tampilan antarmuka login merupakan tampilan Form Login Petugas, disini petugas mengisi data petugas berupa *username* dan *password* agar petugas mendapatkan hak akses untuk melakukan peng-*update*-an kiriman. Nama *file* pada aplikasi ini adalah LoginActivity.java.




**Gambar 5.5** Tampilan AntarMuka Login

### 6. Tampilan AntarMuka Update Status Kiriman

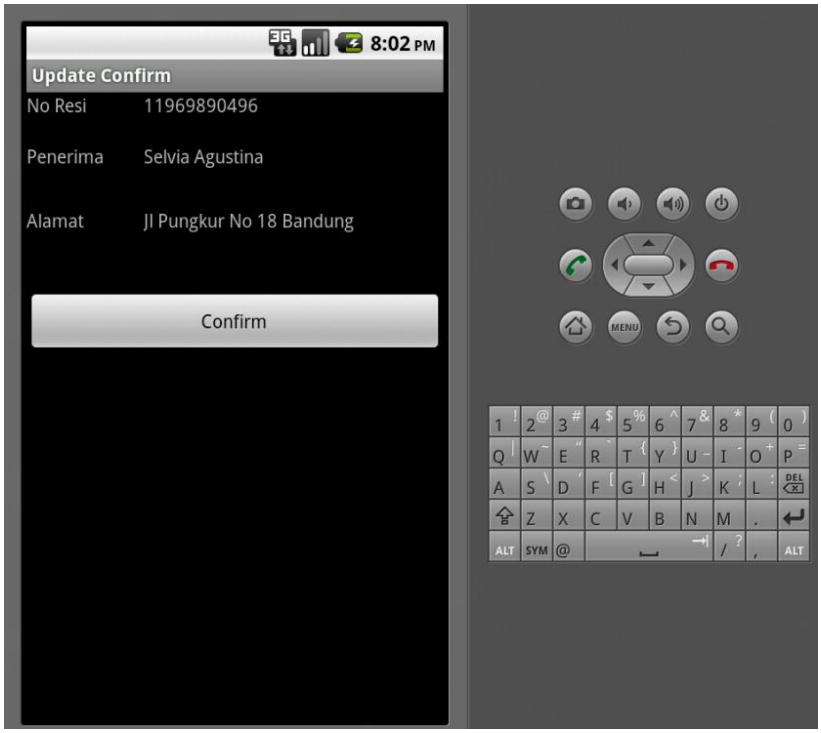
Tampilan antarmuka *update* status kiriman merupakan tampilan form *update* data kiriman berhasil berupa penginputan nomor resi oleh petugas. Nama *file* pada aplikasi ini adalah UpdateResiActivity.java.




**Gambar 5.6** Tampilan AntarMuka *Update* Status Kiriman

### 7. Tampilan AntarMuka Konfirmasi Kiriman

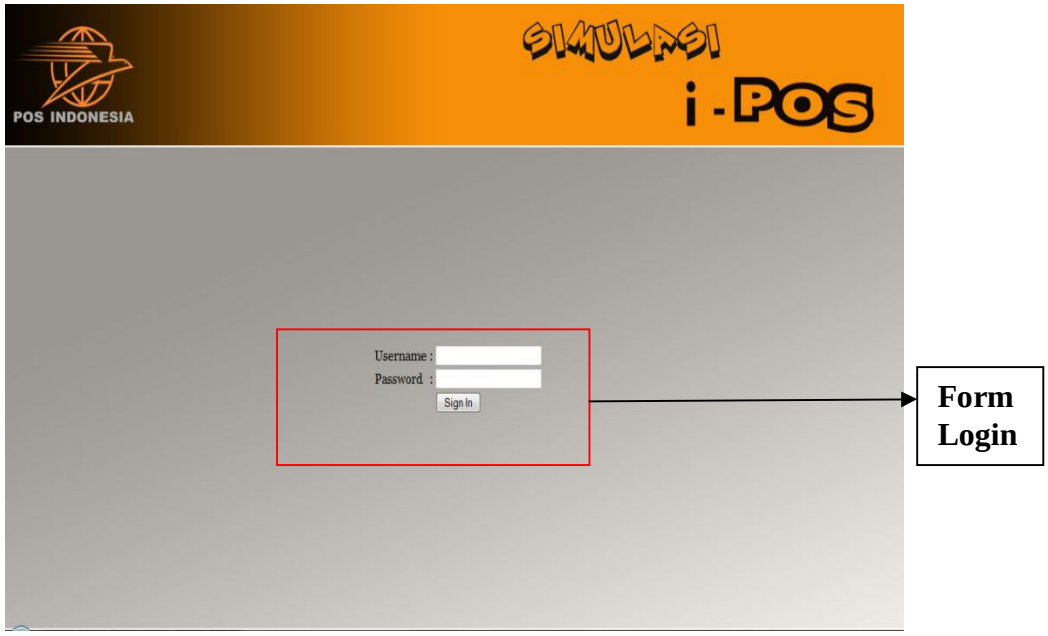
Tampilan antarmuka konfirmasi kiriman merupakan tampilan konfirmasi update nomor resi. Nama *file* pada aplikasi ini adalah ConfirmUpdateResiActivity.java.




**Gambar 5.7** Tampilan AntarMuka Konfirmasi Kiriman

**8. Tampilan AntarMuka Login Simulasi i-POS**

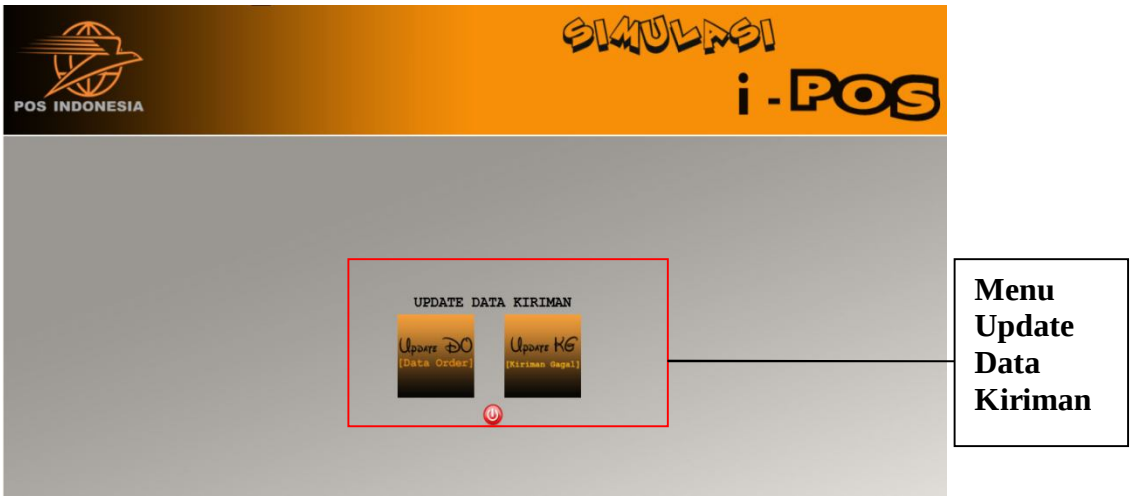
Tampilan antarmuka login Simulasi i-POS merupakan tampilan form login petugas untuk mendapatkan hak akses kedalam sistem. Nama *file* pada aplikasi ini adalah loginForm.php




**Gambar 5.8** Tampilan AntarMuka Login

**9. Tampilan AntarMuka Menu Utama Simulasi i_POS**

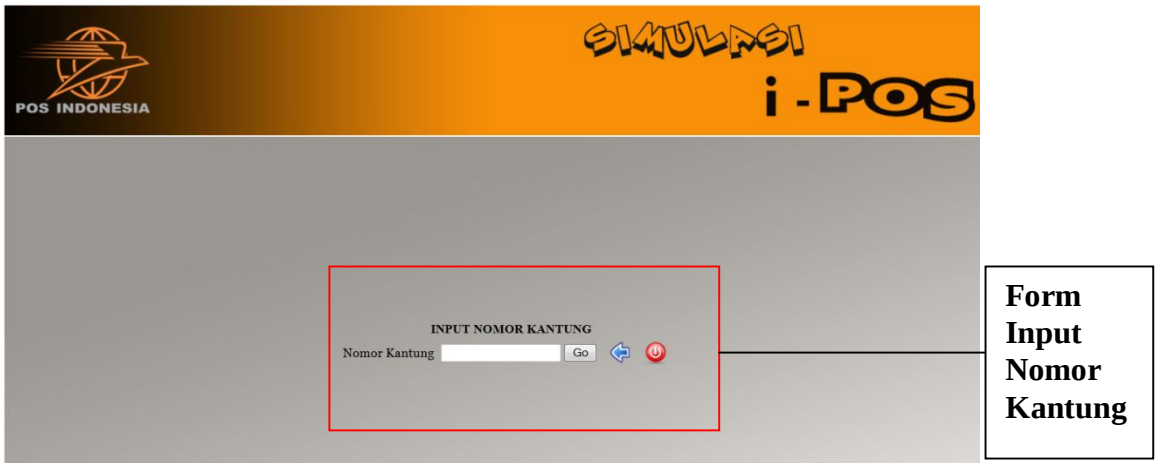
Terdapat 2 menu utama pada Simulasi i-POS yaitu Update Data Order dan Update Kiriman Gagal. Update Data Order digunakan untuk meng-*update* data kantung sedangkan Update Kiriman gagal digunakan untuk meng-*update* nomor resi kiriman gagal. Nama *file* pada aplikasi ini adalah menu_utama.php




**Gambar 5.9** Tampilan AntarMuka Menu Utama

**10. Tampilan AntarMuka Input Data Order Simulasi i_POS**

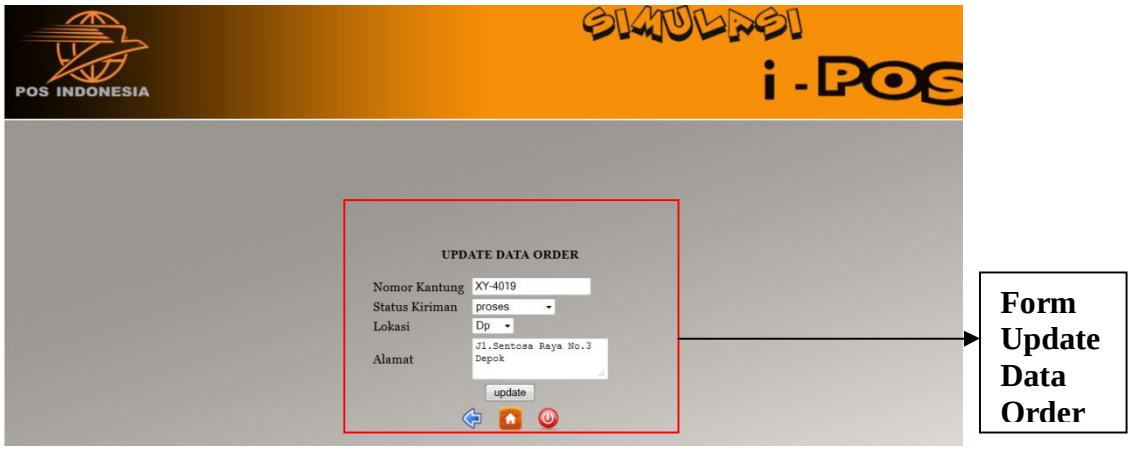
Tampilan Input Data Order merupakan tampilan input nomor kantung yang akan di*update* oleh petugas. Nama *file* pada aplikasi ini adalah kantung.php




**Gambar 5.10** Tampilan AntarMuka Data Order

**11. Tampilan AntarMuka Form Data Order Simulasi i_POS**

Tampilan Form Data Order merupakan tampilan peng-*update*-an status kiriman serta lokasi kiriman berada oleh petugas. Nama *file* pada aplikasi ini adalah DataOrder.php




**Gambar 5.11** Tampilan AntarMuka Data Gagal

**12. Tampilan AntarMuka Input Data Gagal Simulasi i-POS**

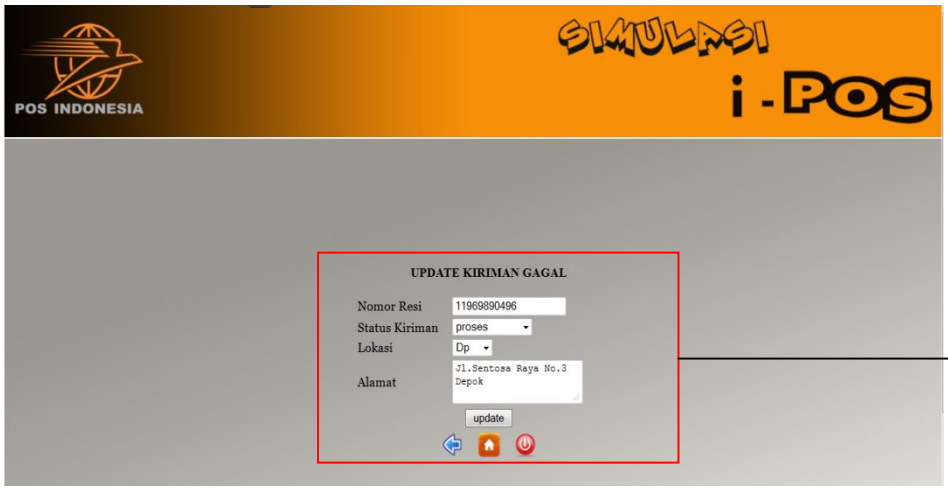
Tampilan Input Data Gagal merupakan tampilan *input* nomor resi gagal kirim, dikembalikan, dan kirim ulang, yang akan di*update* oleh petugas. Nama *file* pada aplikasi ini adalah gagal.php




**Gambar 5.12** Tampilan AntarMuka *Input* Data Gagal

### 13. Tampilan AntarMuka Form Data Gagal Simulasi i-POS

Tampilan Form Data Gagal merupakan tampilan peng-*update*-an status kiriman serta lokasi kiriman berada oleh petugas. Nama *file* pada aplikasi ini adalah datagagal.php

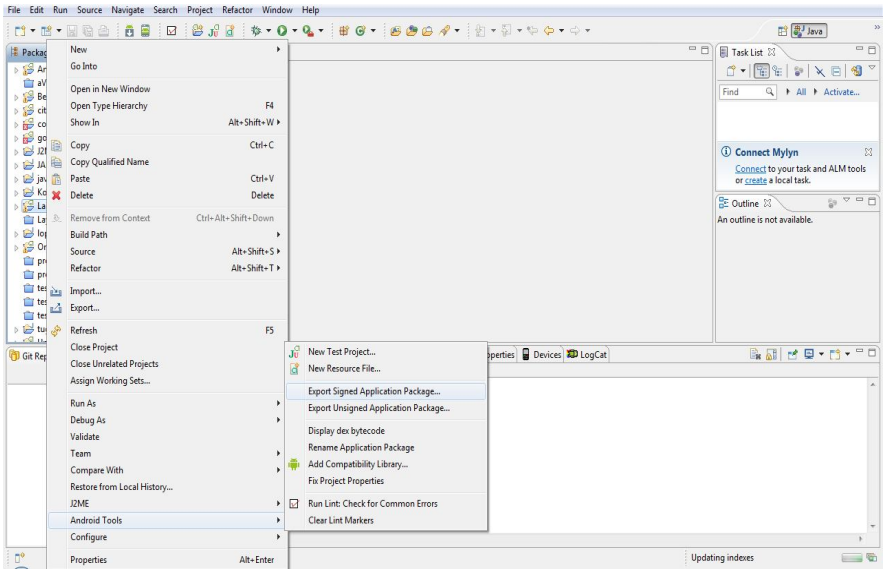



**Gambar 5.13 Tampilan AntarMuka Form Data Gagal**

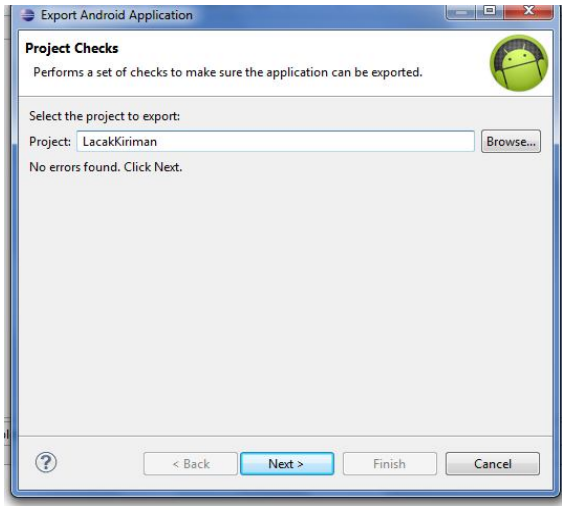
### 5.1.6. Implementasi Instalasi Program

Ada beberapa cara yang dapat dilakukan untuk memasukkan aplikasi Mobile Pelacakan Kiriman ke dalam Mobile phone. Alternatif yang ada antara lain aplikasi dapat dikirimkan ke Mobile phone melalui PC dengan menggunakan kabel data, ataupun Bluetooth jika perangkat yang bersangkutan yang sebelumnya harus instalasi terlebih dahulu driver mobile. Selain itu aplikasi dapat di *convert* ke *file*.apk, seperti langkah-langkah berikut ini :

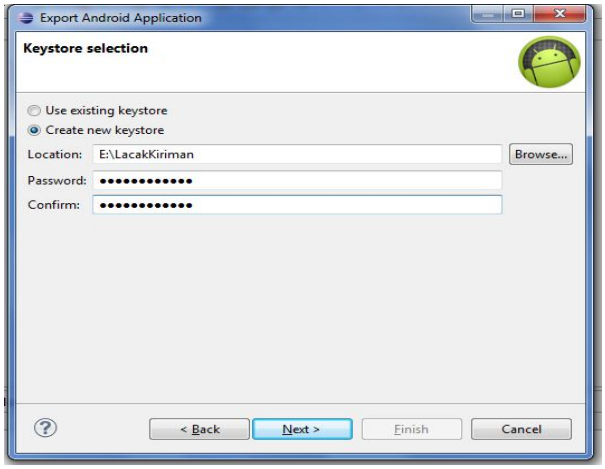
1. Klik kanan Project **LacakKiriman**, lalu pilih **Android Tools**, lalu pilih **Export Signed Application Package**.



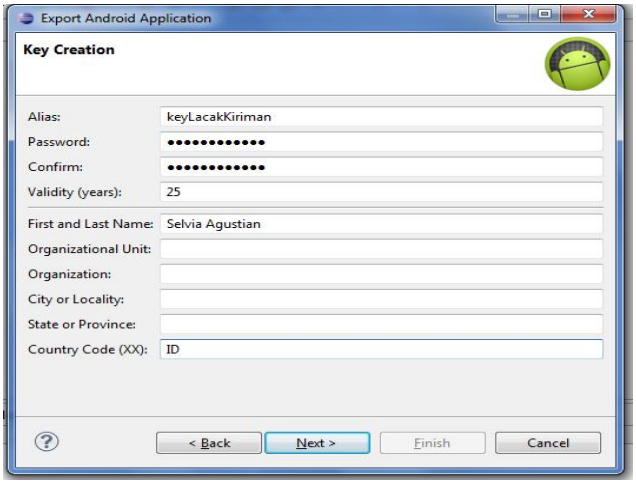

2. Muncul tampilan seperti dibawah ini alalu klik tombol **Next**



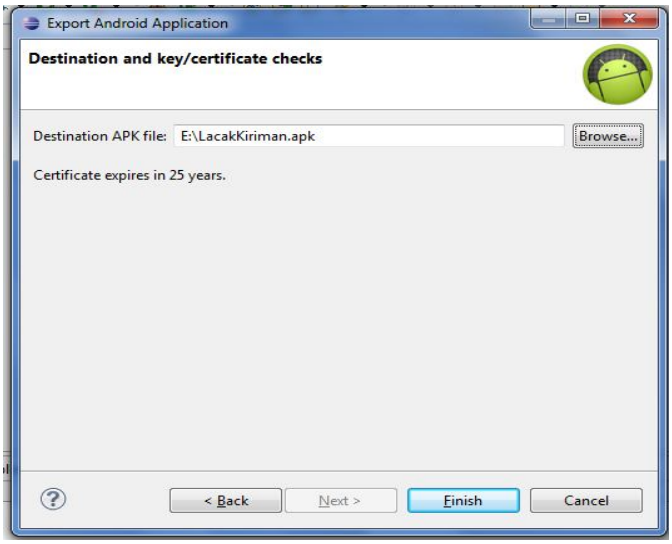
3. Pada Langkah selanjutnya, kita diminta untuk membuat file keystore, isi semua inputan, lalu klik tombol **Next**




4. Langkah Selanjutnya Kita diminta untuk membuat key, disini kita juga diminta untuk maemasukkan identitas gunanya sebagai penanda bahwa aplikasi ini dibuat oleh kita, klik tombol **Next**



5. Langkah selanjutnya kita harus menentukan nama file.apk yang akan dibuat dalam hal ini saya beri nama **LacakKiriman.apk**, Lalu klik Finish.




### 5.1.7. Implementasi Penggunaan Program

Berikut ini merupakan Implementasi penggunaan aplikasi pelacakan kiriman berbasis *mobile* pada layanan Pos *Express* yaitu sebagai berikut;

1. Pengirim membuka aplikasi lacak kiriman di handphone sistem operasi android *versi* 2.2.
2. Tampil halaman lacak kiriman.
3. Kemudian pengirim menginputkan nomor resi kiriman ke form lacak kiriman.
4. klik tombol *Trace.*
5. Tampil halaman informasi detail kiriman.
6. Klik tombol View Maps untuk melihat peta lokasi kiriman berada.
7. Tampil halaman Map.

## 5.2. Pengujian

Pengujian adalah proses untuk memeriksa apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai dengan standar tertentu. Pengujian sistem perlu dilakukan untuk memastikan sistem berjalan dengan semestinya. Ada dua teknik pengujian sistem perangkat lunak, yaitu *white-box testing* dan *black-box testing*. Secara garis besar *white-box testing* digunakan untuk menguji algoritma dari suatu sistem atau dengan kata lain melihat alur proses pada *source code* yang dimiliki suatu sistem. Sedangkan *black-box testing* digunakan untuk menguji fungsionalitas sistem dengan mengabaikan algoritma pada *source code* dan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pada metode ini data uji diinputkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan. Teknik yang akan digunakan untuk menguji Pelacakan kiriman berbasis *mobile* pada layanan Pos *Express* di PT. POS INDONESIA adalah *black-box testing*, lebih rincinya lagi dengan metode *test-case*.

### 5.2.1. Rencana Pengujian

Pengujian yang telah dilakukan selama membuat aplikasi ini, antara lain **:**

1. Pengujian unit, yaitu pengujian ini difokuskan pada suatu unit dari program secara sendiri.
2. Pengujian Penerimaan, yaitu pengujian yang meyakinkan bahwa sistem telah sesuai dengan kebutuhan perusahaan.

Pengujian Aplikasi Pelacakan Kiriman Berbasis *Mobile* pada Layanan Pos *Express* dibawah ini menggunakan data uji berupa sebuah data masukan dari Pengirim dan Petugas Pos.

**Tabel 5.1** Rencana Pengujian Melacak Kiriman

| **Kelas Uji** | **Butir Uji** | **Tingkat Pengujian** | **Jenis Pengujian** |
|---|---|---|---|
| Data Kiriman | Pengisian Data Kiriman | Unit | Black Box |
| Data Detail Kiriman | Tampil Data Detail Kiriman | Unit | Black Box |
| Data Map | Tampil Data Map | Unit | Black Box |

**Tabel 5.2** *Update* Status Kiriman Melalui *Mobile*

| **Kelas Uji** | **Butir Uji** | **Tingkat Pengujian** | **Jenis Pengujian** |
|---|---|---|---|
| Login Petugas Pos | Hak Akses Petugas | Sistem | Black Box |
| Data Kiriman Berhasil | Pengisian Data Kiriman Berhasil | Sistem | Black Box |
| Data Konfirmasi | Pengisian Data Konfirmasi | Sistem | Black Box |

**Tabel 5.3** *Update* Status Kiriman Melalui Simulasi i-POS

| **Kelas Uji** | **Butir Uji** | **Tingkat Pengujian** | **Jenis Pengujian** |
|---|---|---|---|
| Login Petugas | Hak Akses Petugas | Sistem | Black Box |
| Data Order | Pengisian Data Order | Sistem | Black Box |
| Data Gagal | Pengisian Data Gagal | Sistem | Black Box |

### 5.2.2. Kasus dan Hasil Pengujian

Berikut adalah beberapa pengujian yang telah dilakukan yaitu;

1. Pengujian Input Data Kiriman

   Pengujian Input Data kiriman dilakukan untuk meyakinkan bahwa proses penginputan data kiriman berupa nomor resi kiriman oleh pengirim sudah berjalan sesuai dengan kebutuhan sistem secara keseluruhan.

**Tabel 5.4** Pengujian Input Data Kiriman

| **Kasus dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data Diharapkan | Pengamatan | Kesimpulan |
| Klik Tombol *Trace* | Tampil informasi kiriman | Ketika klik tombol *trace* maka akan menampilkan data detail kiriman | [X] DiTerima<br>[ ] DiTolak |
| **Kasus dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamatan | Kesimpulan |
| Data Kiriman (nomor resi) salah atau tidak diisi | muncul *alert* "info not found" | ketika data kiriman diinput salah atau tidak diisi maka muncul *alert* "info not found" | [ ] DiTerima<br>[ ] DiTolak |

2. Data Detail kiriman

   Data Detail Kiriman dilakukan guna untuk meyakinkan bahwa hasil data kiriman yang diinputkan berupa nomor resi sudah menampilkan informasi kiriman berdasarkan data kiriman yang berjalan sesuai dengan kebutuhan sistem secara keseluruhan

**Tabel 5.5** Pengujian Data Detail kiriman

| **Kasus dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data Diharapkan | Pengamatan | Kesimpulan |
| Nomor resi benar | Tampil data detail kiriman sesuai dengan data yang ada di*server* | Ketika nomor resi benar muncul detail kiriman yang ada diserver | [X] DiTerima<br>[ ] DiTolak |
| **Kasus dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamatan | Kesimpulan |
| Nomor Resi Salah atau tidak ada | Tidak tampil data detail kiriman | ketika nomor resi salah atau tidak ada maka tidak akan tampil detail kiriman | [X] DiTerima<br>[ ] DiTolak |

3. Data Map

Data Map dilakukan guna untuk meyakinkan bahwa lokasi kiriman berada tepat di peta oleh pengirim sudah berjalan sesuai dengan kebutuhan sistem

**Tabel 5.6** Pengujian Data Map

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| klik Tombol View Maps | Tampil Peta Lokasi Kiriman | ketika klik tombol View Maps muncul halaman Map | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data Map (longitudes dan lititudes ) salah | Tampil peta tidak seseui dengan lokasi kiriman berada | ketika data map salah maka muncul peta dengan lokasi tidak sesuai | [X] DiTerima<br>[ ] DiTolak |

4. Login Petugas

Pengujian login petugas dilakukan guna meyakinkan bahwa login petugas sudah berjalan sesuai dengan kebutuhan sistem secara keseluruhan.

**Tabel 5.7** Pengujian Login Petugas

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data terdaftar | Tampil halaman utama petugas | Tampil halaman utama petugas | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data tidak terdaftar | Tampil “username & password incorect” | Tampil “username & password incorect” | [X] DiTerima<br>[ ] DiTolak |
| Data tidak diisi atau kosong | Tampil “ info please fill in the text field” | Tampil “ info please fill in the text field” | [X] DiTerima<br>[ ] DiTolak |

5. Penginputan Data Kiriman Berhasil

Penginputan data kiriman berhasil dilakukan guna meyakinkan bahwa proses penginputan data kiriman berhasil berupa nomor resi oleh petugas sudah sesuai dengan kebutuhan sistem secara keseluruhan.

**Tabel 5.8** Penginputan Data Kiriman Berhasil

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Klik tombol update | Mencari data kiriman ( nomor resi) diserver | ketika input nomor resi dan klik tombol update muncul halaman konfirmasi | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data kiriman salah | Tampil “info not found” | ketika data kiriman salah (nomor resi) maka tampil “info not found” | [X] DiTerima<br>[ ] DiTolak |
| Data kiriman tidak diisi atau kosong | Tampil “ info please fill in the text field” | Tampil “ info please fill in the text field” | [X] DiTerima<br>[ ] DiTolak |

6. Penginputan Data Konfirmasi

Penginputan data konfirmasi berhasil dilakukan guna meyakinkan bahwa proses peng-*update*-an data konfirmasi berhasil berupa penginputan nama yang menerima kiriman oleh petugas sudah sesuai dengan kebutuhan sistem secara keseluruhan.

**Tabel 5.9** Penginputan Data Konfirmasi

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Klik tombol konfirmasi | Menyimpan data yang di-*update* | Merubah data kiriman | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data tidak diisi | Tampil " info please fill in the text field" | Tampil " info please fill in the text field" | [X] DiTerima<br>[ ] DiTolak |

7. Login Petugas Pada Simulasi i-POS

Pengujian login petugas dilakukan guna meyakinkan bahwa login petugas sudah berjalan sesuai dengan kebutuhan sistem secara keseluruhan.

**Tabel 5.10** Penginputan Login Petugas

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data terdaftar | Tampil halaman utama petugas | Tampil halaman utama petugas | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data tidak terdaftar | Tampil "Data Not Found" | Tampil "Data Not Found" | [X] DiTerima<br>[ ] DiTolak |
| Data tidak diisi atau kosong | Tampil "Data Not Found" | Tampil "Data Not Found" | [X] DiTerima<br>[ ] DiTolak |

8. Penginputan Data Order

Pengujian data order dilakukan guna meyakinkan bahwa data order sudah berjalan sesuai dengan kebutuhan sistem secara keseluruhan

**Tabel 5.11** Penginputan Data Order

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Klik tombol "GO" | Mencari data order (nomor kantung) di *server* | Ketika tombol GO di klik maka akan menampilkan form *update* data order | [X] DiTerima<br>[ ] DiTolak |
| Klik tombol *Update* | Data *update* simpan ke *server* | ketika form *update* terisi dan klik tombol *update* maka data akan disimpan ke *server* | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data order tidak tersedia | Tampil Alert "Halaman form order kosong" | Ketika data order tidak tersedia maka akan tampil "Halaman form order kosong" | [X] DiTerima<br>[ ] DiTolak |

9. Penginputan Data Gagal

Pengujian data gagal dilakukan guna meyakinkan bahwa data gagal berupa kiriman yang gagal sudah berjalan sesuai dengan kebutuhan sistem secara keseluruhan

**Tabel 5.12** Penginputan Data Gagal

| **Kasus Dan Hasil Pengujian (Data Normal)** | | | |
|---|---|---|---|
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Klik tombol GO | Mencari data kiriman (nomor resi) di *server* | Ketika tombol GO di klik maka akan menampilkan form *update* data kiriman gagal | [X] DiTerima<br>[ ] DiTolak |
| Klik tombol *Update* | Data *update* simpan ke *server* | ketika form *update* terisi dan klik tombol *update* maka data akan disimpan ke *server* | [X] DiTerima<br>[ ] DiTolak |
| **Kasus Dan Hasil Pengujian (Data Salah)** | | | |
| Data Masukan | Data diharapkan | Pengamat | Kesimpulan |
| Data kiriman gagal tidak tersedia | Tampil Alert ”Halaman form gagal kosong” | Ketika data gagal tidak tersedia maka akan tampil ”Halaman form gagal kosong” | [X] DiTerima<br>[ ] DiTolak |

### 5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan metode Black Box yang telah dilakukan, maka dapat ditarik kesimpulan bahwa perangkat lunak Aplikasi Pelacakan Kiriman Berbasis *Mobile* secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.