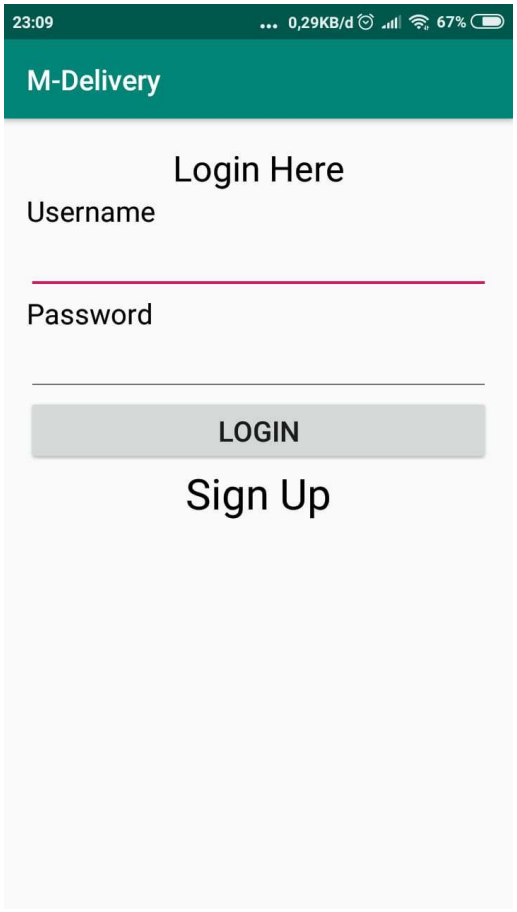
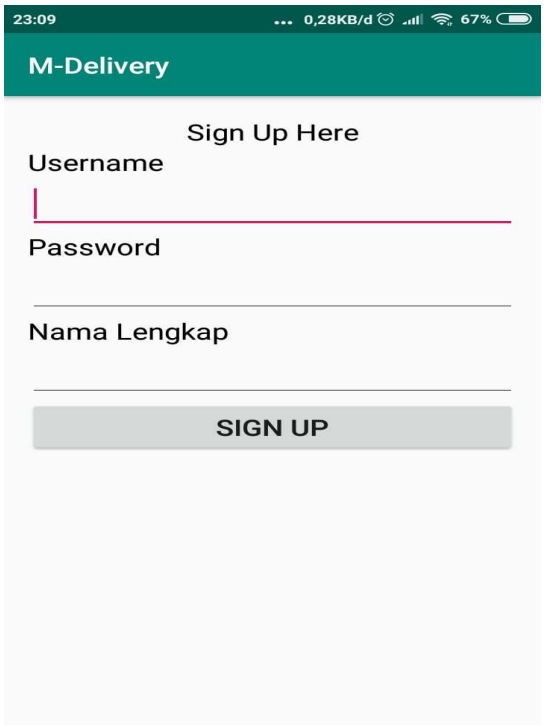
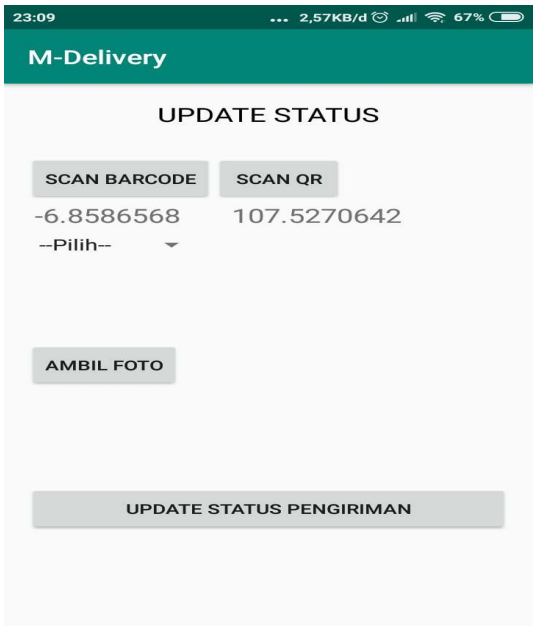
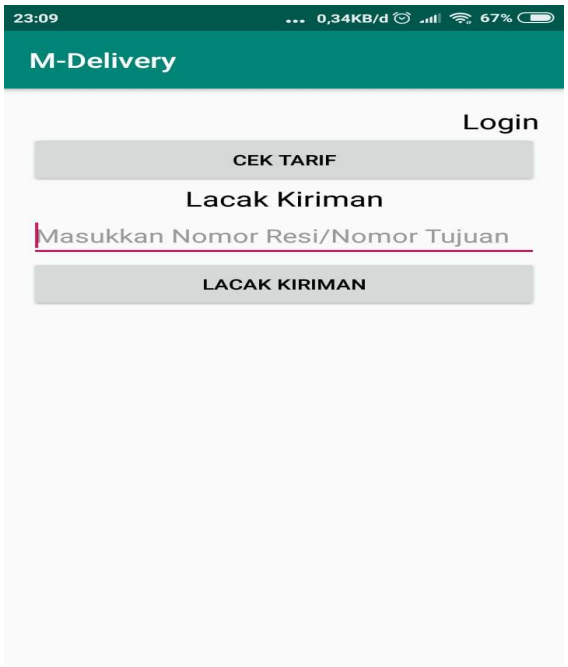
LAMPIRAN A
IMPLEMENTASI ANTARMUKA

LAMPIRAN A
IMPLEMENTASI ANTARMUKA




23:09
0,28KB/d
67%
M-Delivery
Sign Up Here
Username
Password
Nama Lengkap
SIGN UP


23:09
2,57KB/d
67%
M-Delivery
UPDATE STATUS
SCAN BARCODE
SCAN QR
-6.8586568
107.5270642
--Pilih--
AMBIL FOTO
UPDATE STATUS PENGIRIMAN


23:09
0,34KB/d
67%
M-Delivery
Login
CEK TARIF
Lacak Kiriman
Masukkan Nomor Resi/Nomor Tujuan
LACAK KIRIMAN

LAMPIRAN B
LISTING PROGRAM

LAMPIRAN B
LISTING PROGRAM

```
package com.example.mughn.crudlagiah;

import android.Manifest;
import android.app.ProgressDialog;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.provider.MediaStore;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.ActivityCompat;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;

import java.util.HashMap;

public class UpdateStatus extends AppCompatActivity implements
AdapterView.OnItemSelectedListener,GoogleApiClient.ConnectionCallbacks,
GoogleApiClient.OnConnectionFailedListener,
com.google.android.gms.location.LocationListener {

    private TextView editTextNoRestu;
    private Button buttonScanBar;
    private Button buttonScanQR;
    private Button buttonTakePicture;
    private ImageView imageViewHolder;
    //private final int requestCode;
    private Spinner spinnerStatus;
    private Spinner spinnerKetBerhasil;
    private Spinner spinnerKetGagal;
    private Spinner spinnerKetRetur;
    private Spinner spinnerKetDipanggil;
    private Spinner spinnerKetDitahan;
    private Spinner spinnerKetDiteruskan;
    private TextView textViewLat;
    private TextView textViewLong;
    private GoogleApiClient gApi;
```

```
    private Location lokasiAktif;
    LocationRequest mintaLokasi;
    private Button buttonUpdate;

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

        editTextNoRestu = findViewById(R.id.editTextNoRestu);
        buttonScanBar = findViewById(R.id.tmblScanBarcode);
        buttonScanQR = findViewById(R.id.tmblScanQR);
        buttonTakePicture = findViewById(R.id.tmblCapture);
        imageViewHolder = findViewById(R.id.hasilFoto);
        spinnerStatus = findViewById(R.id.spinnerStatus);
        spinnerKetBerhasil = findViewById(R.id.spinnerKetBerhasil);
        spinnerKetGagal = findViewById(R.id.spinnerKetGagal);
        spinnerKetRetur = findViewById(R.id.spinnerKetRetur);
        spinnerKetDipanggil = findViewById(R.id.spinnerKetDipanggil);
        spinnerKetDitahan = findViewById(R.id.spinnerKetDitahan);
        spinnerKetDiteruskan = findViewById(R.id.spinnerKetDiteruskan);
        buttonUpdate = findViewById(R.id.tmblUpdate);


        textViewLat = findViewById(R.id.textViewLatitude);
        textViewLong = findViewById(R.id.textViewLongitude);


        spinnerStatus.setOnItemSelectedListener(this);
        spinnerKetGagal.setOnItemSelectedListener(this);
        spinnerKetRetur.setOnItemSelectedListener(this);
        spinnerKetDipanggil.setOnItemSelectedListener(this);
        spinnerKetDitahan.setOnItemSelectedListener(this);
        spinnerKetDiteruskan.setOnItemSelectedListener(this);

        buttonUpdate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                updateStatus();
            }
        });

        buttonScanBar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                scanBarcode("PRODUCT_MODE");
            }
        });

        buttonScanQR.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                scanBarcode("QR_CODE_MODE");
            }
        });

        buttonTakePicture.setOnClickListener(new View.OnClickListener()
{
```

```
            @Override
            public void onClick(View v) {
                Intent photoCapture = new
Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                startActivityForResult(photoCapture,2);
            }
        });

        if (gApi == null) {
            gApi = new GoogleApiClient.Builder(this)

                    .addConnectionCallbacks(this)
                    .addOnConnectionFailedListener(this)
                    .addApi(LocationServices.API)
                    .build();
        }

        mintaLokasi = new LocationRequest();
        mintaLokasi.setInterval(10000);
        mintaLokasi.setFastestInterval(1000);

mintaLokasi.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    }

    private void updateStatus(){

        final String noresi =
editTextNoRestu.getText().toString().trim();
        final String latitude =
textViewLat.getText().toString().trim();
        final String longitude =
textViewLong.getText().toString().trim();
        final String status =
spinnerStatus.getSelectedItem().toString();
        final String ket_berhasil =
spinnerKetBerhasil.getSelectedItem().toString().trim();
        final String ket_gagal =
spinnerKetGagal.getSelectedItem().toString().trim();
        final String ket_retur =
spinnerKetRetur.getSelectedItem().toString().trim();
        final String ket_dipanggil =
spinnerKetDipanggil.getSelectedItem().toString().trim();
        final String ket_ditahan =
spinnerKetDitahan.getSelectedItem().toString().trim();
        final String ket_diteruskan =
spinnerKetDiteruskan.getSelectedItem().toString().trim();
        final String foto = imageViewHolder.toString().trim();

        class StatusUpdate extends AsyncTask<Void,Void,String>{

            ProgressDialog loading;

            @Override
            protected void onPreExecute(){
                super.onPreExecute();
                loading =
ProgressDialog.show(UpdateStatus.this,"Menambahkan","Please
Wait...",false,false);
```

```
            }

            @Override
            protected void onPostExecute(String s){
                super.onPostExecute(s);
                loading.dismiss();

Toast.makeText(UpdateStatus.this,s,Toast.LENGTH_LONG).show();
            }

            @Override
            protected String doInBackground(Void... v) {
                HashMap<String,String> params = new HashMap<>();
                params.put(konfigurasi.KEY_US_NORESI,noresi);
                params.put(konfigurasi.KEY_US_LATITUDE,latitude);
                params.put(konfigurasi.KEY_US_LONGITUDE,longitude);
                //params.put(konfigurasi.KEY_US_STATUS,status);

//params.put(konfigurasi.KEY_US_KET_BERHASIL,ket_berhasil);
                //params.put(konfigurasi.KEY_US_KET_GAGAL,ket_gagal);
                //params.put(konfigurasi.KEY_US_KET_RETUR,ket_retur);

//params.put(konfigurasi.KEY_US_KET_DIPANGGIL,ket_dipanggil);

//params.put(konfigurasi.KEY_US_KET_DITAHAN,ket_ditahan);

//params.put(konfigurasi.KEY_US_KET_DITERUSKAN,ket_diteruskan);
                //params.put(konfigurasi.KEY_US_FOTO,foto);

                RequestHandler rh = new RequestHandler();
                String result =
rh.sendPostRequest(konfigurasi.URL_UPDATE_STATUS,params);
                return result;
            }
        }
        StatusUpdate su = new StatusUpdate();
        su.execute();
    }

    private void scanBarcode(String mode) {
        try {
            //buat intent untuk memanggil fungsi scan pada aplikasi
zxing
            Intent intent = new
Intent("com.google.zxing.client.android.SCAN");
            intent.putExtra("SCAN_MODE", mode); // "PRODUCT_MODE for
bar codes
            startActivityForResult(intent, 1);

        } catch (Exception e) {

            Uri marketUri =
Uri.parse("market://details?id=com.google.zxing.client.android");
            Intent marketIntent = new
Intent(Intent.ACTION_VIEW,marketUri);
            startActivity(marketIntent);

        }
```

```
    }

    @Override
    public void onActivityResult(int requestCode, int resultCode,
Intent data) {
        super.onActivityResult(requestCode,resultCode,data);

        if(requestCode == 1 && resultCode == RESULT_OK){
            //Bitmap bitmap = (Bitmap)data.getExtras().get("data");
            //imageViewHolder.setImageBitmap(bitmap);
            //editTextNoRestu.setText("");
            String contents = data.getStringExtra("SCAN_RESULT");
            editTextNoRestu.setText(contents);
        } else if(requestCode == 2 && resultCode == RESULT_OK){
            Bitmap bitmap = (Bitmap)data.getExtras().get("data");
            imageViewHolder.setImageBitmap(bitmap);
            //editTextNoRestu.setText("");
            //String contents = data.getStringExtra("SCAN_RESULT");
            //editTextNoRestu.setText(contents);
        }
    }

    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
        if(parent.getId() == R.id.spinnerStatus){
            if(position == 1){
                spinnerKetBerhasil.setVisibility(View.VISIBLE);
                spinnerKetGagal.setVisibility(View.INVISIBLE);
            } else if (position == 2 ){
                spinnerKetGagal.setVisibility(View.VISIBLE);
                spinnerKetBerhasil.setVisibility(View.INVISIBLE);
            }
        } else if (parent.getId() == R.id.spinnerKetGagal){
            if (position == 1){
                spinnerKetRetur.setVisibility(View.VISIBLE);
                spinnerKetDipanggil.setVisibility(View.INVISIBLE);
                spinnerKetDitahan.setVisibility(View.INVISIBLE);
                spinnerKetDiteruskan.setVisibility(View.INVISIBLE);
            } else if (position == 2){
                spinnerKetDipanggil.setVisibility(View.VISIBLE);
                spinnerKetRetur.setVisibility(View.INVISIBLE);
                spinnerKetDitahan.setVisibility(View.INVISIBLE);
                spinnerKetDiteruskan.setVisibility(View.INVISIBLE);
            } else if (position == 3){
                spinnerKetDitahan.setVisibility(View.VISIBLE);
                spinnerKetRetur.setVisibility(View.INVISIBLE);
                spinnerKetDipanggil.setVisibility(View.INVISIBLE);
                spinnerKetDiteruskan.setVisibility(View.INVISIBLE);
            } else if (position == 4){
                spinnerKetDitahan.setVisibility(View.INVISIBLE);
                spinnerKetRetur.setVisibility(View.INVISIBLE);
                spinnerKetDipanggil.setVisibility(View.INVISIBLE);
                spinnerKetDiteruskan.setVisibility(View.VISIBLE);
            }
        }
    }
```

```
    @Override
    public void onNothingSelected(AdapterView<?> parent) {

    }

    @Override
    protected void onStart() {
        gApi.connect();
        super.onStart();
    }

    @Override
    protected void onStop() {
        super.onStop();
        LocationServices.FusedLocationApi.removeLocationUpdates(gApi,
(com.google.android.gms.location.LocationListener) this);
        gApi.disconnect();
    }

    @Override
    public void onConnected(@Nullable Bundle bundle) {
        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            // TODO: Consider calling
            //    ActivityCompat#requestPermissions
            // here to request the missing permissions, and then
overriding
            //   public void onRequestPermissionsResult(int
requestCode, String[] permissions,
            //                                          int[]
grantResults)
            // to handle the case where the user grants the permission.
See the documentation
            // for ActivityCompat#requestPermissions for more details.
            return;
        }
        Location lok =
LocationServices.FusedLocationApi.getLastLocation(gApi);

        if (lok != null){
            textViewLat.setText(String.valueOf(lok.getLatitude()));
            textViewLong.setText(String.valueOf(lok.getLongitude()));
        } else {

LocationServices.FusedLocationApi.requestLocationUpdates(gApi,mintaLoka
si,this);

        }
    }

    @Override
    public void onConnectionSuspended(int i) {
        Log.i("Koneksi","Suspended");
    }
```

```
    @Override
    public void onConnectionFailed(@NonNull ConnectionResult
connectionResult) {
        Toast.makeText(this,"Gagal Koneksi :
"+connectionResult.getErrorMessage(),Toast.LENGTH_LONG).show();
    }

    @Override
    public void onLocationChanged(Location location) {
        lokasiAktif = location;

        textViewLat.setText(String.valueOf(lokasiAktif.getLatitude()));

textViewLong.setText(String.valueOf(lokasiAktif.getLongitude()));
    }
}
```

```
   package com.example.mughn.crudlagiah;

   import android.support.v7.app.AppCompatActivity;
   import android.os.Bundle;
   import android.view.View;
   import android.widget.Button;
   import android.widget.EditText;
   import android.widget.TextView;

   public class CekTarif extends AppCompatActivity {

       private EditText editTextKasal;
       private EditText editTextKtujuan;
       private EditText editTextBeratK;
       private EditText editTextPanjangK;
       private EditText editTextLebarK;
       private EditText editTextTinggiK;

       private TextView textViewRes;

       private Button buttonCekTarif;

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

           editTextKasal = findViewById(R.id.editTextAsal);
           editTextKtujuan = findViewById(R.id.editTextTujuan);
           editTextBeratK = findViewById(R.id.editTextBerat);
           editTextPanjangK = findViewById(R.id.editTextPanjang);
           editTextLebarK = findViewById(R.id.editTextLebar);
           editTextTinggiK = findViewById(R.id.editTextTinggi);

           final String asal =
   editTextKasal.getText().toString().trim();
           final String tujuan =
   editTextKtujuan.getText().toString().trim();
           final String berat =
```

```
editTextBeratK.getText().toString().trim();
        final String panjang =
editTextPanjangK.getText().toString().trim();
        final String lebar =
editTextLebarK.getText().toString().trim();
        final String tinggi =
editTextTinggiK.getText().toString().trim();

        textViewRes = findViewById(R.id.textViewHasil);

        buttonCekTarif = findViewById(R.id.tmblCetar);
        buttonCekTarif.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                textViewRes.setText("Kota Asal :" +asal+" "+"Kota
Tujuan : "+tujuan+" "+"Jenis Kiriman : Paket Kilat Khusus (1 Hari)
Harga : 6930.69 Pajak Harga : 69.31 Total : 7000");
            }
        });
    }
}
```

LAMPIRAN C
HASIL PENGUJIAN

LAMPIRAN C
HASIL PENGUJIAN

1. Apakah Anda (kurir) lebih mudah untuk melakukan tahap update status pengiriman ?

   ☐ Mudah ☐ Cukup Mudah ☐ Tidak Mudah

2. Apakah pengoperasian aplikasi mudah dan dapat dipahami ?

   ☐ Mudah ☐ Cukup Mudah ☐ Tidak Mudah

1. Apakah dengan adanya posisi GPS barang lebih mudah melihat kiriman sudah tepat pada tujuan ?

   ☐ Mudah ☐ Cukup Mudah ☐ Tidak Mudah

2. Apakah dengan adanya bukti nyata digital lebih mudah untuk melakukan penyortiran terhadap pelanggan ?

   ☐ Mudah ☐ Cukup Mudah ☐ Tidak Mudah

3. Apakah dengan adanya aplikasi ini Anda (petugas) dapat dengan mudah melihat berapa status pengiriman yang berhasil atau gagal ?

   ☐ Mudah ☐ Cukup Mudah ☐ Tidak Mudah

LAMPIRAN D
SURAT BALASAN DARI PERUSAHAAN

LAMPIRAN E
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI