

Trabajo Final de Máster

**Doble titulación Máster Ingeniería Industrial y Máster
Ingeniería de Automoción**

**Aplicación Android para recogida de datos cinemáticos
de un vehículo**

ANEXOS

Autor: Rubén Camúñez Llanos
Director: M. Antonia de los Santos Lopez
Convocatòria: Julio 2017



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Anexos

1. Código pantalla presentación.....	2
2. Layout pantalla presentación	3
3. Código menú principal	4
4. Layout menú principal.....	6
5. Código modo gráfico.....	9
6. Layout modo gráfico	21
7. Código modo debug	24
8. Layout modo debug	37
9. Android manifest.....	42
10. Código de Matlab	43
11. Fragmento del ciclo realizado.....	47

1. Código pantalla presentación

```
package com.elseib.ruben.proyecto_uni;

import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

import java.util.Timer;
import java.util.TimerTask;

public class PantallaInicio extends AppCompatActivity {

    // Duracion de la pantalla de inicio
    private static final long SPLASH_SCREEN_DELAY = 3000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        //supportRequestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.activity_pantalla_inicio);

        TimerTask task = new TimerTask() {
            @Override
            public void run() {

                // Start the next activity
                Intent mainIntent = new
Intent().setClass(PantallaInicio.this, MainActivity.class);
                startActivity(mainIntent);

                // Close the activity so the user won't able to go back
                // activity pressing Back button
                finish();
            }
        };

        // Simulate a long loading process on application startup.
        Timer timer = new Timer();
        timer.schedule(task, SPLASH_SCREEN_DELAY);
    }
}
```

2. Layout pantalla presentación

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".PantallaInicio"
    android:background="@android:color/background_light">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:contentDescription="@string/app_name"

        android:src="@drawable/logoetseib"
        android:background="@android:color/background_light"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />

</RelativeLayout>
```

3. Código menú principal

```

package com.elseib.ruben.proyecto_uni;

import android.content.Intent;
import android.net.Uri;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.Toast;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.OutputStreamWriter;

public class MainActivity extends AppCompatActivity {

    Spinner spinner_comb;
    Spinner spinner_peso;
    Spinner spinner_pot;

    String[] comb = new String[]{"Combustible", "Gasolina", "Diesel"};
    String[] peso = new String[]{"Peso", "m < 1000kg", "1000kg < m < 1500kg", "m > 1500kg"};
    String[] pot = new String[]{"Potencia", "< 65 kW", "> 65 kW"};

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

        spinner_comb = (Spinner) findViewById(R.id.spincombustible);
        spinner_peso = (Spinner) findViewById(R.id.spinpeso);
        spinner_pot = (Spinner) findViewById(R.id.spinpotencia);

        ArrayAdapter<String> adaptadorcomb = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_item, comb);
        ArrayAdapter<String> adaptadorpeso = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_item, peso);
        ArrayAdapter<String> adaptadorpot = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_item, pot);

        adaptadorcomb.setDropDownViewResource(android.R.layout.simple_spinner_dropd
        own_item);

        adaptadorpeso.setDropDownViewResource(android.R.layout.simple_spinner_dropd
        own_item);

        adaptadorpot.setDropDownViewResource(android.R.layout.simple_spinner_dropdo
        wn_item);

        spinner_comb.setAdapter(adaptadorcomb);
        spinner_peso.setAdapter(adaptadorpeso);
        spinner_pot.setAdapter(adaptadorpot);
    }

    public void cambiar_layout(View v){

```

```
        try {
            OutputStreamWriter datos = new
OutputStreamWriter (openFileOutput ("aceleraciones.txt",
AppCompatActivity.MODE_APPEND));

datos.write (spinner_comb.getSelectedItem().toString()+" "+spinner_peso.getS
electedItem().toString()+" "+spinner_pot.getSelectedItem().toString() +
"\r\n");

            datos.flush();
            datos.close();
        } catch (IOException e) {
        }
        Intent ListSong = new Intent (getApplicationContext(),
Velocimetros.class);
        startActivity(ListSong);
    }

    public void cambiar_layout2 (View v) {
        try {
            OutputStreamWriter datos = new
OutputStreamWriter (openFileOutput ("aceleraciones.txt",
AppCompatActivity.MODE_APPEND));

datos.write (spinner_comb.getSelectedItem().toString()+" "+spinner_peso.getS
electedItem().toString()+" "+spinner_pot.getSelectedItem().toString() +
"\r\n");

            datos.flush();
            datos.close();
        } catch (IOException e) {
        }
        Intent ListSong = new Intent (getApplicationContext(),
ModoDebug.class);
        startActivity(ListSong);
    }

    public void borrar_datos (View v) {
        String archivoTXT = "aceleraciones.txt";
        File txt=new File (getFilesDir()+" "+archivoTXT);
        txt.delete();
        Toast.makeText (getApplicationContext(), "Archivo eliminado",
Toast.LENGTH_SHORT).show();
    }
}
}
```

4. Layout menú principal

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.etsib.ruben.proyecto_uni.MainActivity"
    android:orientation="vertical"
    android:background="@color/azul_fondo">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_gravity="center_horizontal"
        android:paddingTop="16dp"
        android:paddingBottom="16dp">

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:orientation="vertical">

            <Button
                android:id="@+id/button2"
                android:onClick="cambiar_layout"
                android:layout_height="50dp"
                android:layout_width="80dp"
                android:text="MODO GRÁFICO"
                android:textColor="@color/colorPrimary"
                android:background="@drawable/botonnormal"
                android:layout_gravity="center_horizontal" />

        </LinearLayout>

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:orientation="vertical">

            <Button
                android:id="@+id/button3"
                android:onClick="cambiar_layout2"
                android:layout_height="50dp"
                android:layout_width="80dp"
                android:layout_gravity="center_horizontal"
                android:text="MODO DEBUG"
                android:textColor="@color/colorPrimary"
                android:background="@drawable/botonnormal" />

        </LinearLayout>

```



```
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:orientation="vertical">

        <Spinner
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/spincombustible" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:orientation="vertical">

        <Spinner
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/spinpeso" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:orientation="vertical">

        <Spinner
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/spinpotencia" />

    </LinearLayout>

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:layout_gravity="center_horizontal"
```

```
android:paddingTop="16dp"
android:paddingBottom="16dp">

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="0dp"
    android:layout_weight="2"
    android:layout_gravity="center"
    android:gravity="center"
    android:orientation="vertical">

    <ImageView
        android:layout_width="120dp"
        android:layout_height="120dp"
        app:srcCompat="@drawable/logoetseib"
        android:layout_gravity="center"
        android:id="@+id/imageView" />

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:layout_gravity="right"
    android:orientation="vertical">

    <Button
        android:id="@+id/button4"
        android:onClick="borrar_datos"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botondelete" />

</LinearLayout>

</LinearLayout>

</LinearLayout>
```

5. Código modo gráfico

```
package com.elseib.ruben.proyecto_uni;

import android.Manifest;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.content.IntentSender;
import android.content.pm.ActivityInfo;
import android.content.pm.PackageManager;
import android.graphics.Color; //
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
//import android.location.Address;
//import android.location.Geocoder;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Build;
import android.os.PowerManager;
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.TextView;
import android.widget.Toast;
import android.widget.ToggleButton;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.common.api.PendingResult;
import com.google.android.gms.common.api.ResultCallback;
import com.google.android.gms.common.api.Status;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.location.LocationSettingsRequest;
import com.google.android.gms.location.LocationSettingsResult;
import com.google.android.gms.location.LocationSettingsStatusCodes;

import org.apache.commons.net.ftp.FTP;
import org.apache.commons.net.ftp.FTPClient;

import java.io.BufferedInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.net.SocketException;
import java.text.DecimalFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.List;
import java.util.Locale;
import java.util.Timer;
import java.util.TimerTask;
```

```

import static java.lang.Math.acos;
import static java.lang.Math.cos;
import static java.lang.Math.sin;
import static java.lang.Math.sqrt;

public class Velocimetros extends AppCompatActivity implements
SensorEventListener,GoogleApiClient.OnConnectionFailedListener,GoogleApiCli
ent.ConnectionCallbacks, LocationListener {

    private SpeedometerGauge speedometer;
    private CircularGauge circgauge;
    private CircularGauge circgauge2;
    protected PowerManager.WakeLock wakelock;
    private Sensor mAccelerometer;
    long tiempo_actual=0;
    long tiempo_anterior=0;
    long tiempo_inicial;
    String longitud;
    String latitud;
    String velocidad;
    TextView tv1;
    TextView tv4;
    TextView crono;
    boolean detener=false;
    double theta=0;
    double accX;
    double accY;
    double accZ;
    double compX=1;
    double compY=0;
    String longitud_anterior;
    String latitud_anterior;
    int contador_posicion=0;
    DecimalFormat df=new DecimalFormat("0.0000");
    DecimalFormat df2=new DecimalFormat("00");
    Location location_anterior;
    boolean primera=true;
    String distancia;
    float velocidad_num;
    double latitud_num;
    double longitud_num;
    double recorrido=0;
    String precision;
    String proveedor;
    String tiempoGPS;
    String tiempoElapsed;
    boolean primera_escritura=true;
    boolean iguales=false;
    File rutaCompleta;
    BufferedInputStream buffer;
    Boolean logeado=false;
    String nombre_archivo;

    ////////////GPS//////////
    private static final String LOGTAG = "android-localizacion";

    private static final int PETICION_PERMISO_LOCALIZACION = 101;

```

```

private static final int PETICION_CONFIG_UBICACION = 201;

private GoogleApiClient apiClient;

private ToggleButton btnActualizar;

private LocationRequest locRequest;
//////////GPS//////////

TimerTask timerTask = new TimerTask() {
    @Override
    public void run() {
        // Aquí el código que queremos ejecutar.
        tiempo_actual=tiempo_actual+1;

        if (tiempo_actual%10==0) {

            if(longitud_anterior==longitud ){
                contador_posicion=contador_posicion+1;
                iguales=true;
            } else {
                contador_posicion=0;
                iguales=false;
            }
            longitud_anterior=longitud;
            latitud_anterior=latitud;
        }
    }
};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_velocimetros);
    setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

    final PowerManager
pm=(PowerManager) getSystemService(Context.POWER_SERVICE);
    this.wakeLock=pm.newWakeLock(PowerManager.SCREEN_DIM_WAKE_LOCK,
"etiqueta");
    wakeLock.acquire();

    tv1=(TextView) findViewById(R.id.et1);
    tv4=(TextView) findViewById(R.id.et4);
    crono=(TextView) findViewById(R.id.crono);

    tiempo_inicial=System.nanoTime();

    // Aquí se pone en marcha el timer cada segundo.
    Timer timer = new Timer();
    // Dentro de 0 milisegundos avísame cada 100 milisegundos
    timer.scheduleAtFixedRate(timerTask, 0, 100);

    speedometer = (SpeedometerGauge) findViewById(R.id.speedometer);
    speedometer.setMaxSpeed(180);
    speedometer.setLabelConverter(new SpeedometerGauge.LabelConverter()
{
    @Override
    public String getLabelFor(double progress, double maxProgress)
{

```

```

        return String.valueOf((int) Math.round(progress));
    }
});
speedometer.setMaxSpeed(180);
speedometer.setMajorTickStep(20);
speedometer.setMinorTicks(3);
speedometer.addColoredRange(0, 80, Color.GREEN);
speedometer.addColoredRange(80, 120, Color.YELLOW);
speedometer.addColoredRange(120, 180, Color.RED);
speedometer.setSpeed(0, 1000, 300);

circgauge = (CircularGauge) findViewById(R.id.circgauge);
circgauge.setMaxSpeed(180);
circgauge.setLabelConverter(new CircularGauge.LabelConverter() {
    @Override
    public String getLabelFor(double progress, double maxProgress)
{
        return String.valueOf((int) Math.round(progress));
    }
});
circgauge.setMaxSpeed(20);
circgauge.setMajorTickStep(5);
circgauge.setMinorTicks(4);
circgauge.setSpeed(0, 1000, 300);

circgauge2 = (CircularGauge) findViewById(R.id.circgauge2);
circgauge2.setMaxSpeed(180);
circgauge2.setLabelConverter(new CircularGauge.LabelConverter() {
    @Override
    public String getLabelFor(double progress, double maxProgress)
{
        return String.valueOf((int) Math.round(progress));
    }
});
circgauge2.setMaxSpeed(5);
circgauge2.setMajorTickStep(1);
circgauge2.setMinorTicks(4);
circgauge2.setSpeed(0, 1000, 300);

btnActualizar = (ToggleButton) findViewById(R.id.btnActualizar);
btnActualizar.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        toggleLocationUpdates(btnActualizar.isChecked());
    }
});

apiClient = new GoogleApiClient.Builder(this)
    .enableAutoManage(this, this)
    .addConnectionCallbacks(this)
    .addApi(LocationServices.API)
    .build();
}

protected void onDestroy() {
    super.onDestroy();
    this.wakeLock.release();
}

```

```

}

//////////GPS//////////
private void toggleLocationUpdates(boolean enable) {
    if (enable) {
        enableLocationUpdates();
    } else {
        disableLocationUpdates();
    }
}

private void enableLocationUpdates() {
    locRequest = new LocationRequest();
    locRequest.setInterval(1000);
    locRequest.setFastestInterval(1000);
    locRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);

    LocationSettingsRequest locSettingsRequest =
        new LocationSettingsRequest.Builder()
            .addLocationRequest(locRequest)
            .build();

    //...
    PendingResult<LocationSettingsResult> result =
        LocationServices.SettingsApi.checkLocationSettings(
            apiClient, locSettingsRequest);

    result.setResultCallback(new
    ResultCallback<LocationSettingsResult>() {
        @Override
        public void onResult(LocationSettingsResult
    locationSettingsResult) {
            final Status status = locationSettingsResult.getStatus();
            switch (status.getStatusCode()) {
                case LocationSettingsStatusCodes.SUCCESS:

                    Log.i(LOGTAG, "Configuración correcta");
                    startLocationUpdates();
                    break;

                case LocationSettingsStatusCodes.RESOLUTION_REQUIRED:
                    try {
                        Log.i(LOGTAG, "Se requiere actuación del
usuario");

                        status.startResolutionForResult(Velocimetros.this,
PETICION_CONFIG_UBICACION);
                    } catch (IntentSender.SendIntentException e) {
                        btnActualizar.setChecked(false);
                        Log.i(LOGTAG, "Error al intentar solucionar
configuración de ubicación");
                    }
                    break;

                case
LocationSettingsStatusCodes.SETTINGS_CHANGE_UNAVAILABLE:
                    Log.i(LOGTAG, "No se puede cumplir la configuración
de ubicación necesaria");
                    btnActualizar.setChecked(false);
                    break;
            }
        }
    });
}

```

```

    }
    });
}

private void disableLocationUpdates () {
LocationServices.FusedLocationApi.removeLocationUpdates (apiClient, this);
}

private void startLocationUpdates () {
    if (ActivityCompat.checkSelfPermission (Velocimetros.this,
        Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED) {

        Log.i (LOGTAG, "Inicio de recepción de ubicaciones");

LocationServices.FusedLocationApi.requestLocationUpdates (apiClient,
locRequest, Velocimetros.this);
    }

@Override
public void onRequestPermissionsResult (int requestCode, String[]
permissions, int[] grantResults) {
    if (requestCode == PETICION_PERMISO_LOCALIZACION) {
        if (grantResults.length == 1 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            //Permiso concedido
            @SuppressWarnings ("MissingPermission")
            Location lastLocation =
LocationServices.FusedLocationApi.getLastLocation (apiClient);
            updateUI (lastLocation);
        } else {
            //Permiso denegado:

            Log.e (LOGTAG, "Permiso denegado");
        }
    }
}

@Override
protected void onActivityResult (int requestCode, int resultCode, Intent
data) {
    switch (requestCode) {
        case PETICION_CONFIG_UBICACION:
            switch (resultCode) {
                case Activity.RESULT_OK:
                    startLocationUpdates ();
                    break;
                case Activity.RESULT_CANCELED:
                    Log.i (LOGTAG, "El usuario no ha realizado los
cambios de configuración necesarios");
                    btnActualizar.setChecked (false);
                    break;
            }
            break;
    }
}

@Override

```



```

public void onConnectionFailed(ConnectionResult result) {
    Log.e(LOGTAG, "Error grave al conectar con Google Play Services");
}

@Override
public void onConnected(@Nullable Bundle bundle) {
    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, PETICION_PERMISO_LOCALIZ
ACION);
    } else {
        Location lastLocation =
LocationServices.FusedLocationApi.getLastLocation(apiClient);

        updateUI(lastLocation);
    }
}

@Override
public void onConnectionSuspended(int i) {
    Log.e(LOGTAG, "Se ha interrumpido la conexión con Google Play
Services");
}

@Override
public void onLocationChanged(Location location) {
    Log.i(LOGTAG, "Recibida nueva ubicación!");
    //Mostramos la nueva ubicación recibida
    updateUI(location);
}

private void updateUI(Location loc) {
    if (loc != null) {
        if (primera){
            primera=false;
            location_anterior=loc;
        }
        latitud=String.valueOf(df.format(loc.getLatitude()));
        latitud_num=loc.getLatitude();
        longitud=String.valueOf(df.format(loc.getLongitude()));
        longitud_num=loc.getLongitude();
        recorrido=recorrido+loc.distanceTo(location_anterior);
        //distancia=String.valueOf(loc.distanceTo(location_anterior));
        location_anterior=loc;
        velocidad=String.valueOf(loc.getSpeed());
        velocidad_num=loc.getSpeed();
        precision=String.valueOf(loc.getAccuracy());
        proveedor=loc.getProvider();
        tiempoGPS=String.valueOf(loc.getTime());
        if (Build.VERSION.SDK_INT > 16) {

            tiempoElapsed=String.valueOf(loc.getElapsedRealtimeNanos(
));
        }
    } else {
        tv1.setText("Latitud: (desconocida)");
        tv4.setText("Longitud: (desconocida)");
    }
}

```

```

    }

    ////////////////////////////////////GPS//////////////////////////////////////
    ////////////////////////////////////

    protected void onResume() {
        super.onResume();
        wakelock.acquire();
        SensorManager sm = (SensorManager)
        getSystemService(SENSOR_SERVICE);
        List<Sensor> sensors = sm.getSensorList(Sensor.TYPE_ACCELEROMETER);
        if (sensors.size() > 0) { //dispositivo android tiene acelerometro
            sm.registerListener(this, sensors.get(0),
            SensorManager.SENSOR_DELAY_NORMAL);
        }
    }

    public void onSaveInstanceState(Bundle icle) {
        super.onSaveInstanceState(icle);
        this.wakelock.release();
    }

    protected void onPause() {
        SensorManager mSensorManager=(SensorManager)
        getSystemService(SENSOR_SERVICE);
        mSensorManager.unregisterListener(this, mAccelerometer);
        super.onPause();
    }

    protected void onStop() {
        SensorManager mSensorManager=(SensorManager)
        getSystemService(SENSOR_SERVICE);
        mSensorManager.unregisterListener(this, mAccelerometer);
        super.onStop();
    }

    @Override
    public void onSensorChanged(SensorEvent event) {

        /*if (puede_calibrar(event)) {
            calibrar(event);
        }*/

        if (tiempo_actual>tiempo_anterior && !detener) {
            escribir(event);
            tiempo_anterior=tiempo_actual;
        }
    }

    @Override
    public void onAccuracyChanged(Sensor sensor, int accuracy) {
    }

    public void escribir(SensorEvent event) {
        try {

            if (primera_escritura) {

```

```

        OutputStreamWriter datos = new
OutputStreamWriter (openFileOutput ("aceleraciones.txt", AppCompatActivity.MODE
E_APPEND));
        //datos.write("AccX" + ";" + "AccY" + ";" + "AccZ" + ";" +
+ "AccXcorregida" + ";" + "AccYcorregida" + ";" + "AccZcorregida" + ";" +
"latitud" + ";" + "longitud" + ";" + "velocidad" + ";" + "recorrido" + ";" +
+ "timeacc" + ";" + "precision" + ";" + "tiempoGPS" + ";" + "tiempoElapsed"
+ "\r\n");
        datos.write("AccX" + ";" + "AccY" + ";" + "AccZ" + ";" +
"Latitud" + ";" + "Longitud" + ";" + "Velocidad" + ";" + "recorrido" + ";" +
+ "timeacc" + ";" + "precision" + ";" + "tiempoGPS" + ";" + "tiempoElapsed"
+ "\r\n");
        datos.flush();
        datos.close();
        primera_escritura=false;
    }

    tv1.setText("Latitud: "+latitud);
    tv4.setText("Longitud= "+longitud);

    long timeacc=(System.nanoTime()-tiempo_inicial)/1000000;

    crono.setText(df2.format(timeacc/1000/3600)+" :"+df2.format(timeacc/1000%360
0/60)+" :"+df2.format(timeacc/1000%3600%60));
    ///////////////
    ///////////////Modificar aqui lo que escribe en el
txt/////////////////
        OutputStreamWriter datos = new
OutputStreamWriter (openFileOutput ("aceleraciones.txt", AppCompatActivity.MODE
E_APPEND));
        //datos.write(event.values[0] + ";" + event.values[1] + ";" +
event.values[2] + ";" + vector_aceleraciones[0] + ";" +
vector_aceleraciones[1] + ";" + vector_aceleraciones[2]+";"+ latitud +
";" + longitud + ";" + "velocidad"+";"+"recorrido"+";"+ timeacc +
";"+precision+";"+tiempoGPS+";"+tiempoElapsed+ "\r\n");

    datos.write(event.values[0]+";"+event.values[1]+";"+event.values[2]+";"+lat
itud+";"+longitud+";"+velocidad+";"+recorrido+";"+timeacc+";"+precision+";"
+tiempoGPS+";"+tiempoElapsed+ "\r\n");
        datos.flush();
        datos.close();

    //circgauge.setSpeed(sqrt(vector_aceleraciones[0]*vector_aceleraciones[0]+v
ector_aceleraciones[1]*vector_aceleraciones[1]));

    circgauge.setSpeed(sqrt(event.values[0]*event.values[0]+event.values[1]*eve
nt.values[1]));
        circgauge2.setSpeed(recorrido/1000);
        speedometer.setSpeed(velocidad_num*3.6);
        speedometer.setUnitsText(df2.format(recorrido/1000)+"km");
    } catch (IOException e) {
    }
}

public void borrar_datos(View v) {
    String archivoTXT = "aceleraciones.txt";
    File txt=new File(getFilesDir()+"/"+archivoTXT);
    txt.delete();
}

```

```

        Toast.makeText(getApplicationContext(), "Archivo eliminado",
Toast.LENGTH_SHORT).show();
    }

    public void iniciar_registro(View v) {
        detener=false;
    }

    public void detener_registro(View v) {
        detener=true;
    }

    public void compartir(View v) {

////////////////////////////////////
////////////////////////////////////
        detener=true;
        Date fecha=new Date();

//nombre_archivo=fecha.getDay()+"del"+(fecha.getMonth()+1)+"alas"+fecha.get
Hours()+fecha.getMinutes()+fecha.getSeconds()+".txt";
        Calendar rightNow = Calendar.getInstance();
        int dia=rightNow.get(Calendar.DAY_OF_MONTH);
        int mes=rightNow.get(Calendar.MONTH)+1;
        int any=rightNow.get(Calendar.YEAR);
        int hour=rightNow.get(Calendar.HOUR_OF_DAY);
        int minute=rightNow.get(Calendar.MINUTE);

nombre_archivo=dia+"del"+mes+"del"+any+"alas"+hour+"y"+minute+".txt";
        new MyFTP().execute();

        String archivoTXT = "aceleraciones.txt";
        //Crea intent para enviar el email.
        Intent i = new Intent(Intent.ACTION_SEND);
        //File txt = new File(getExternalFilesDir(null)+"/"+archivoTXT);
        File txt=new File(getFilesDir()+"/"+archivoTXT);
        txt.setReadable(true,false);
        txt.setWritable(true,false);
        Uri uri=Uri.fromFile(txt);
        i.setType("text/plain");
        //Agrega email o emails de destinatario.
        i.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
        i.putExtra(Intent.EXTRA_EMAIL, new String[] {
"ruben_c11@hotmail.com" });
        i.putExtra(Intent.EXTRA_SUBJECT, "Envio de archivo TXT.");
        i.putExtra(Intent.EXTRA_TEXT, "Hola te envío un archivo TXT con
aceleraciones.");
        i.putExtra(Intent.EXTRA_STREAM, uri);
        startActivity(Intent.createChooser(i, "Enviar e-mail mediante:"));

////////////////////////////////////
////////////////////////////////////
    }

    private class MyFTP extends AsyncTask<String, Integer, String> {
        @Override
        protected String doInBackground(String... params) {

                //Toast.makeText(getApplicationContext(), "Conectando",
Toast.LENGTH_SHORT).show();
                FTPClient ftpClient=new FTPClient();

```

```

        try {
            //Creamos un objeto Cliente HTTP para manejar la petición
al servidor
            ftpClient.connect("147.83.135.204",113);
        } catch (Exception e) {
            return "Exception happened: " + e.getMessage();
        }
        try {
            logeado=ftpClient.login("RUBEN", "RubenTFM");
        } catch (SocketException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    //////////////////////////////////////
    //////////////////////////////////////

    ftpClient.enterLocalActiveMode();
    try {
        ftpClient.setFileType(FTP.BINARY_FILE_TYPE);
        ftpClient.setFileTransferMode(ftpClient.BINARY_FILE_TYPE);
        ftpClient.enterLocalPassiveMode();
        //Cambia la carpeta Ftp
        //if (ftpClient.changeWorkingDirectory("ftp")){
        //Obtiene la dirección de la ruta
        //Obtiene la ruta completa donde se encuentra el archivo
        //rutaCompleta = new File(rutaSd.getAbsolutePath(),
"prueba.txt");
        rutaCompleta=new
File(getFilesDir()+"/"+"aceleraciones.txt");
        rutaCompleta.setReadable(true,false);
        //Crea un buffer hacia el servidor de subida
        buffer = new BufferedInputStream(new
FileInputStream(rutaCompleta));

        if (ftpClient.storeFile(nombre_archivo, buffer)){
            buffer.close(); //Cierra el bufer
        }
        else{
            buffer.close(); //Cierra el bufer
        }

    } catch (IOException e) {
        e.printStackTrace();
    }
}

    //////////////////////////////////////
    //////////////////////////////////////

    return "Hola";
}

protected void onProgressUpdate(Integer... progress) {
    //Se obtiene el progreso de la petición
    Toast.makeText(getApplicationContext(), "Indicador de progreso
" + progress[0].toString(), Toast.LENGTH_SHORT).show();
}

```

```
protected void onPostExecute(String result) {  
    //Se obtiene el resultado de la petición Asincrona  
    //Log.w(APP_TAG,"Resultado obtenido " + result);  
    //processResult(result);  
}  
  
}
```

6. Layout modo gráfico

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/segundo_layout">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center">

    <com.elseib.ruben.proyecto_uni.SpeedometerGauge
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="8dp"
        android:layout_gravity="center"
        android:id="@+id/speedometer"/>

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center">

    <com.elseib.ruben.proyecto_uni.CircularGauge
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:padding="8dp"
        android:gravity="left"
        android:layout_gravity="left"
        android:id="@+id/circgauge"/>

    <com.elseib.ruben.proyecto_uni.CircularGauge
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:padding="8dp"
        android:gravity="left"
        android:layout_gravity="left"
        android:id="@+id/circgauge2"/>

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:id="@+id/et1"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="AccX" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <TextView
        android:id="@+id/et4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Latitud" />

</LinearLayout>
</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:text="Name"
    android:ems="10"
    android:gravity="center"
    android:id="@+id/crono"
    tools:text="00:00:00"
    android:textSize="24sp"
    android:textColor="@android:color/holo_blue_dark" />

<ToggleButton android:id="@+id/btnActualizar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:textOn="GPS"
    android:textOff="GPS" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <Button
            android:id="@+id/button2"
            android:onClick="iniciar_registro"
```



```
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botonplay" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button3"
        android:onClick="detener_registro"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botonpause" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button4"
        android:onClick="borrar_datos"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botondelete" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button"
        android:onClick="compartir"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botoncompartir" />

</LinearLayout>

</LinearLayout>

</LinearLayout>
```

7. Código modo debug

```
package com.elseib.ruben.proyecto_uni;

import android.Manifest;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.content.IntentSender;
import android.content.pm.ActivityInfo;
import android.content.pm.PackageManager;
import android.content.res.AssetManager;
import android.graphics.Color; //
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Build;
import android.os.Environment;
import android.os.PowerManager;
import android.provider.ContactsContract;
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.TextView;
import android.widget.Toast;
import android.widget.ToggleButton;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.common.api.PendingResult;
import com.google.android.gms.common.api.ResultCallback;
import com.google.android.gms.common.api.Status;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.location.LocationSettingsRequest;
import com.google.android.gms.location.LocationSettingsResult;
import com.google.android.gms.location.LocationSettingsStatusCodes;

import java.io.BufferedInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.net.SocketException;
import java.nio.channels.FileChannel;
import java.text.DecimalFormat;
import java.util.Calendar;
import java.util.Date;
```

```
import java.util.List;
import java.util.Locale;
import java.util.Timer;
import java.util.TimerTask;

import org.apache.commons.net.ftp.FTP;
import org.apache.commons.net.ftp.FTPClient;

import static java.lang.Math.acos;
import static java.lang.Math.cos;
import static java.lang.Math.sin;
import static java.lang.Math.sqrt;

public class ModoDebug extends AppCompatActivity implements
    SensorEventListener, GoogleApiClient.OnConnectionFailedListener, GoogleApiClient.
    ConnectionCallbacks, LocationListener {

    protected PowerManager.WakeLock wakelock;
    private Sensor mAccelerometer;
    long tiempo_actual=0;
    long tiempo_anterior=0;
    long tiempo_inicial;
    long tiempo_inicial_GPS;
    long tiempo_inicial_elapsed;
    String longitud;
    String latitud;
    String velocidad;
    TextView tv1;
    TextView tv2;
    TextView tv3;
    TextView tv4;
    TextView tv5;
    TextView tv6;
    TextView tv7;
    TextView tv8;
    TextView tv9;
    TextView tv10;
    TextView tv11;
    TextView tv12;
    TextView crono;
    boolean detener=false;
    double theta=0;
    double accX;
    double accY;
    double accZ;
    double compX=1;
    double compY=0;
    String longitud_anterior;
    String latitud_anterior;
    int contador_posicion=0;
    DecimalFormat df=new DecimalFormat("0.0000");
    DecimalFormat df2=new DecimalFormat("00");
    Location location_anterior;
    boolean primera=true;
    String distancia;
    float velocidad_num;
    double latitud_num;
    double longitud_num;
```

```

double recorrido=0;
String precision;
String proveedor;
long tiempoGPS;
long tiempoElapsed;
boolean primera_escritura=true;
File rutaCompleta;
BufferedInputStream buffer;
Boolean logeado=false;
String nombre_archivo;

//////////GPS//////////
private static final String LOGTAG = "android-localizacion";

private static final int PETICION_PERMISO_LOCALIZACION = 101;
private static final int PETICION_CONFIG_UBICACION = 201;

private GoogleApiClient apiClient;

private ToggleButton btnActualizar;

private LocationRequest locRequest;
//////////GPS//////////

TimerTask timerTask = new TimerTask() {
    @Override
    public void run() {
        // Aquí el código que queremos ejecutar.
        tiempo_actual=tiempo_actual+1;
        if (tiempo_actual%10==0) {
            if(longitud_anterior==longitud &&
latitud_anterior==latitud){
                contador_posicion=contador_posicion+1;
            } else {
                contador_posicion=0;
            }
            longitud_anterior=longitud;
            latitud_anterior=latitud;
        }
    }
};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_modos_debug);
    setRequestedOrientation (ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

    final PowerManager
pm=(PowerManager) getSystemService (Context.POWER_SERVICE);
    this.wakeLock=pm.newWakeLock (PowerManager.SCREEN_DIM_WAKE_LOCK,
"etiqueta");
    wakeLock.acquire ();

    tv1=(TextView) findViewById (R.id.et1);
    tv2=(TextView) findViewById (R.id.et2);
    tv3=(TextView) findViewById (R.id.et3);
    tv4=(TextView) findViewById (R.id.et4);
    tv5=(TextView) findViewById (R.id.et5);

```

```

tv6=(TextView) findViewById(R.id.et6);
tv7=(TextView) findViewById(R.id.et7);
tv8=(TextView) findViewById(R.id.et8);
tv9=(TextView) findViewById(R.id.et9);
tv10=(TextView) findViewById(R.id.et10);
tv11=(TextView) findViewById(R.id.et11);
tv12=(TextView) findViewById(R.id.et12);
crono=(TextView) findViewById(R.id.crono);

tiempo_inicial=System.nanoTime();

// Aquí se pone en marcha el timer cada segundo.
Timer timer = new Timer();
// Dentro de 0 milisegundos avísame cada 100 milisegundos
timer.scheduleAtFixedRate(timerTask, 0, 100);

btnActualizar = (ToggleButton) findViewById(R.id.btnActualizar);
btnActualizar.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        toggleLocationUpdates(btnActualizar.isChecked());
    }
});

Date fecha=new Date();

//nombre_archivo=fecha.getDay()+"del"+(fecha.getMonth()+1)+"alas"+fecha.get
Hours()+fecha.getMinutes()+fecha.getSeconds()+".txt";
Calendar rightNow = Calendar.getInstance();
int dia=rightNow.get(Calendar.DAY_OF_MONTH);
int mes=rightNow.get(Calendar.MONTH)+1;
int any=rightNow.get(Calendar.YEAR);
int hour=rightNow.get(Calendar.HOUR_OF_DAY);
int minute=rightNow.get(Calendar.MINUTE);

nombre_archivo=dia+"del"+mes+"del"+any+"alas"+hour+"y"+minute+".txt";

ApiClient = new GoogleApiClient.Builder(this)
    .enableAutoManage(this, this)
    .addConnectionCallbacks(this)
    .addApi(LocationServices.API)
    .build();
}

protected void onDestroy(){
    super.onDestroy();
    this.wakeLock.release();
}

//////////GPS//////////
private void toggleLocationUpdates(boolean enable) {
    if (enable) {
        enableLocationUpdates();
    } else {
        disableLocationUpdates();
    }
}

private void enableLocationUpdates() {
    locRequest = new LocationRequest();
    locRequest.setInterval(1000);

```

```

locRequest.setFastestInterval(1000);
locRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);

LocationSettingsRequest locSettingsRequest =
    new LocationSettingsRequest.Builder()
        .addLocationRequest(locRequest)
        .build();

//...
PendingResult<LocationSettingsResult> result =
    LocationServices.SettingsApi.checkLocationSettings(
        apiClient, locSettingsRequest);

result.setResultCallback(new
ResultCallback<LocationSettingsResult>() {
    @Override
    public void onResult(LocationSettingsResult
locationSettingsResult) {
        final Status status = locationSettingsResult.getStatus();
        switch (status.getStatusCode()) {
            case LocationSettingsStatusCodes.SUCCESS:

                Log.i(LOGTAG, "Configuración correcta");
                startLocationUpdates();
                break;

            case LocationSettingsStatusCodes.RESOLUTION_REQUIRED:
                try {
                    Log.i(LOGTAG, "Se requiere actuación del
usuario");
                    status.startResolutionForResult(ModoDebug.this,
PETICION_CONFIG_UBICACION);
                } catch (IntentSender.SendIntentException e) {
                    btnActualizar.setChecked(false);
                    Log.i(LOGTAG, "Error al intentar solucionar
configuración de ubicación");
                }
                break;

            case
LocationSettingsStatusCodes.SETTINGS_CHANGE_UNAVAILABLE:
                Log.i(LOGTAG, "No se puede cumplir la configuración
de ubicación necesaria");
                btnActualizar.setChecked(false);
                break;
        }
    }
});

private void disableLocationUpdates() {

LocationServices.FusedLocationApi.removeLocationUpdates(apiClient, this);
}

private void startLocationUpdates() {
    if (ActivityCompat.checkSelfPermission(ModoDebug.this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED) {
        Log.i(LOGTAG, "Inicio de recepción de ubicaciones");
    }
}

```

```

LocationServices.FusedLocationApi.requestLocationUpdates(apiClient,
locRequest, ModoDebug.this);
    }
}

@Override
public void onRequestPermissionsResult(int requestCode, String[]
permissions, int[] grantResults) {
    if (requestCode == PETICION_PERMISO_LOCALIZACION) {
        if (grantResults.length == 1 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            //Permiso concedido
            @SuppressWarnings("MissingPermission")
            Location lastLocation =
LocationServices.FusedLocationApi.getLastLocation(apiClient);
            updateUI(lastLocation);
        } else {
            //Permiso denegado:

            Log.e(LOGTAG, "Permiso denegado");
        }
    }
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
    switch (requestCode) {
        case PETICION_CONFIG_UBICACION:
            switch (resultCode) {
                case Activity.RESULT_OK:
                    startLocationUpdates();
                    break;
                case Activity.RESULT_CANCELED:
                    Log.i(LOGTAG, "El usuario no ha realizado los
cambios de configuración necesarios");
                    btnActualizar.setChecked(false);
                    break;
            }
            break;
    }
}

@Override
public void onConnectionFailed(ConnectionResult result) {
    Log.e(LOGTAG, "Error grave al conectar con Google Play Services");
}

@Override
public void onConnected(@Nullable Bundle bundle) {
    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, PETICION_PERMISO_LOCALIZ
ACION);
    } else {
        Location lastLocation =
LocationServices.FusedLocationApi.getLastLocation(apiClient);

```

```

        updateUI(lastLocation);
    }
}

@Override
public void onConnectionSuspended(int i) {
    Log.e(LOGTAG, "Se ha interrumpido la conexión con Google Play
Services");
}

@Override
public void onLocationChanged(Location location) {
    Log.i(LOGTAG, "Recibida nueva ubicación!");
    //Mostramos la nueva ubicación recibida
    updateUI(location);
}

private void updateUI(Location loc) {
    if (loc != null) {
        if (primera){
            primera=false;
            location_anterior=loc;
            tiempo_inicial_GPS=loc.getTime();
            if (Build.VERSION.SDK_INT > 16) {
                tiempo_inicial_elapsed=loc.getElapsedRealtimeNanos();
            }
        }
        latitud=String.valueOf(loc.getLatitude());
        latitud_num=loc.getLatitude();
        longitud=String.valueOf(loc.getLongitude());
        longitud_num=loc.getLongitude();
        recorrido=recorrido+loc.distanceTo(location_anterior);
        //distancia=String.valueOf(loc.distanceTo(location_anterior));
        location_anterior=loc;
        velocidad=String.valueOf(loc.getSpeed());
        velocidad_num=loc.getSpeed();
        precision=String.valueOf(loc.getAccuracy());
        proveedor=loc.getProvider();
        tiempoGPS=loc.getTime()-tiempo_inicial_GPS;
        if (Build.VERSION.SDK_INT > 16) {
            tiempoElapsed=(loc.getElapsedRealtimeNanos()-
tiempo_inicial_elapsed)/1000000;
        }
    } else {

    }
}

////////////////////////////////////////GPS////////////////////////////////////////
////////////////////////////////////////

protected void onResume() {
    super.onResume();
    wakelock.acquire();
    SensorManager sm = (SensorManager)
getSystemService(SENSOR_SERVICE);
    List<Sensor> sensors = sm.getSensorList(Sensor.TYPE_ACCELEROMETER);
    if (sensors.size() > 0) { //dispositivo android tiene acelerometro
        sm.registerListener(this, sensors.get(0),
SensorManager.SENSOR_DELAY_NORMAL);
    }
}

```



```

    }

}

public void onSaveInstanceState(Bundle icle) {
    super.onSaveInstanceState(icle);
    this.wakelock.release();
}

protected void onPause() {
    SensorManager mSensorManager=(SensorManager)
getSystemService (SENSOR_SERVICE);
    mSensorManager.unregisterListener(this, mAccelerometer);
    super.onPause();
}

protected void onStop() {
    SensorManager mSensorManager=(SensorManager)
getSystemService (SENSOR_SERVICE);
    mSensorManager.unregisterListener(this, mAccelerometer);
    super.onStop();
}

@Override
public void onSensorChanged(SensorEvent event) {

    /*if (puede_calibrar(event)) {
        calibrar(event);
    }*/

    if (tiempo_actual>tiempo_anterior && !detener) {

        escribir(event);
        tiempo_anterior=tiempo_actual;
    }

}

@Override
public void onAccuracyChanged(Sensor sensor, int accuracy) {
}

public void escribir(SensorEvent event) {
    try {

        if (primera_escritura) {
            OutputStreamWriter datos = new
OutputStreamWriter(openFileOutput("aceleraciones.txt",AppCompatActivity.MOD
E_APPEND));
            datos.write("AccX[m/s2]" + ";" + "AccY[m/s2]" + ";" +
"AccZ[m/s2]" + ";" + "Latitud" + ";" + "Longitud" + ";" +
"Velocidad[m/s]" + ";" + "Recorrido[m]" + ";" + "Crono[ms]" + ";" +
"PrecisionGPS[m]" + ";" + "TiempoGPS[ms]" + ";" + "TiempoElapsed[ms]" +
"\r\n");
            datos.flush();
            datos.close();
            primera_escritura=false;
        }
        //double[] vector_aceleraciones=cambio_base(event);

```

```

        long timeacc=(System.nanoTime()-tiempo_inicial)/1000000;

        tv1.setText("AccX= "+df.format(event.values[0]));
        tv2.setText("AccY= "+df.format(event.values[1]));
        tv3.setText("AccZ= "+df.format(event.values[2]));
        /*tv4.setText("AccX= "+df.format(vector_aceleraciones[0]));
        tv5.setText("AccY= "+df.format(vector_aceleraciones[1]));
        tv6.setText("AccZ= "+df.format(vector_aceleraciones[2]));*/
        tv4.setText("Precision= "+precision);
        tv5.setText("Proveedor= "+proveedor);
        tv6.setText("TiempoGPS= "+tiempoGPS+"ms");
        tv7.setText("Latitud: "+latitud);
        tv8.setText("Longitud= "+longitud);
        tv9.setText("TiempoElapsed= "+tiempoElapsed+"ms");
        tv10.setText("Velocidad= "+velocidad_num +"m/s");
        tv11.setText("Recorrido= "+recorrido+"m");
        tv12.setText("Tiempo= "+timeacc/1000+"s");

        crono.setText(df2.format(timeacc/1000/3600)+":"+df2.format(timeacc/1000%3600/60)+":"+df2.format(timeacc/1000%3600%60));

        ///////////////////////////////////////////////////Modificar aqui lo que escribe en el
        txt////////////////////////////////////
        OutputStreamWriter datos = new
        OutputStreamWriter(openFileOutput("aceleraciones.txt",AppCompatActivity.MODE_APPEND));
        //datos.write(event.values[0] + ";" + event.values[1] + ";" +
        event.values[2] + ";" + vector_aceleraciones[0] + ";" +
        vector_aceleraciones[1] + ";" + vector_aceleraciones[2]+";"+ latitud +
        ";" + longitud + ";"+"velocidad";"+recorrido";"+ timeacc + "\r\n");

        datos.write(event.values[0]+";"+event.values[1]+";"+event.values[2]+";"+latitud+";"+longitud+";"+velocidad+";"+recorrido+";"+timeacc+";"+precision+";"+tiempoGPS+";"+tiempoElapsed+ "\r\n");
        datos.flush();
        datos.close();
    } catch (IOException e) {
    }
}

public void borrar_datos(View v) {
    String archivoTXT = "aceleraciones.txt";
    File txt=new File(getFilesDir()+"/"+archivoTXT);
    txt.delete();
    Toast.makeText(getApplicationContext(), "Archivo eliminado",
    Toast.LENGTH_SHORT).show();
}

public void iniciar_registro(View v) {
    detener=false;
}

public void copyFile(File sourceFile, File destFile)
    throws IOException {

    if (!destFile.exists()) {
        destFile.createNewFile();
    }
}

```

```

FileChannel source = null;
FileChannel destination = null;
FileInputStream is = null;
FileOutputStream os = null;
try {
    is = new FileInputStream(sourceFile);
    os = new FileOutputStream(destFile);
    source = is.getChannel();
    destination = os.getChannel();

    long count = 0;
    long size = source.size();
    while ((count += destination.transferFrom(source, count, size
        - count)) < size)
        ;
} catch (Exception ex) {
} finally {
    if (source != null) {
        source.close();
    }
    if (is != null) {
        is.close();
    }
    if (destination != null) {
        destination.close();
    }
    if (os != null) {
        os.close();
    }
}
}

public void detener_registro(View v) {
    detener=true;
}

public void guardar(View v) {
    detener=true;

    if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.WRITE_EXTERNAL_STORAGE) !=
PackageManager.PERMISSION_GRANTED) {
        Toast.makeText(getApplicationContext(), "faltan permisos",
Toast.LENGTH_SHORT).show();
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, 0);
    } else {
        Toast.makeText(getApplicationContext(), "permiso concedido",
Toast.LENGTH_SHORT).show();
    }

    try {
        File root = new File(Environment.getExternalStorageDirectory(),
"Ciclos");
        if (!root.exists()) {
            root.mkdirs();
        }
        //Toast.makeText(getApplicationContext(), "llega a 1",
Toast.LENGTH_SHORT).show();
        File gpxfile = new File(root, nombre_archivo);

```

```

        //Toast.makeText(getApplicationContext(), "llega a 2",
Toast.LENGTH_SHORT).show();
        rutaCompleta=new File(getFilesDir()+"/"+"aceleraciones.txt");
        rutaCompleta.setReadable(true, false);
        //Toast.makeText(getApplicationContext(), "llega a 3",
Toast.LENGTH_SHORT).show();
        copyFile(rutaCompleta,gpxfile);
        Toast.makeText(getApplicationContext(), "guardado
correctamente", Toast.LENGTH_SHORT).show();
    } catch (IOException e) {
        e.printStackTrace();
        Toast.makeText(getApplicationContext(), "no se ha podido
guardar", Toast.LENGTH_SHORT).show();
    }
}

public void compartir(View v) {

    detener=true;
    /*Date fecha=new Date();

//nombre_archivo=fecha.getDay()+"del"+(fecha.getMonth()+1)+"alas"+fecha.get
Hours()+fecha.getMinutes()+fecha.getSeconds()+".txt";
    Calendar rightNow = Calendar.getInstance();
    int dia=rightNow.get(Calendar.DAY_OF_MONTH);
    int mes=rightNow.get(Calendar.MONTH)+1;
    int any=rightNow.get(Calendar.YEAR);
    int hour=rightNow.get(Calendar.HOUR_OF_DAY);
    int minute=rightNow.get(Calendar.MINUTE);

nombre_archivo=dia+"del"+mes+"del"+any+"alas"+hour+"y"+minute+".txt";*/

    new MyFTP().execute();

////////////////////////////////////
////////////////////////////////////

    String archivoTXT = "aceleraciones.txt";
    //Crea intent para enviar el email.
    Intent i = new Intent(Intent.ACTION_SEND);
    //File txt = new File(getExternalFilesDir(null)+"/"+archivoTXT);
    File txt=new File(getFilesDir()+"/"+archivoTXT);
    txt.setReadable(true, false);
    txt.setWritable(true, false);
    Uri uri=Uri.fromFile(txt);
    i.setType("text/plain");
    //Agrega email o emails de destinatario.
    i.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
    i.putExtra(Intent.EXTRA_EMAIL, new String[] {
"ruben_c11@hotmail.com" });
    i.putExtra(Intent.EXTRA_SUBJECT, "Envio de archivo TXT.");
    i.putExtra(Intent.EXTRA_TEXT, "Hola te envío un archivo TXT con
aceleraciones.");
    i.putExtra(Intent.EXTRA_STREAM, uri);
    startActivity(Intent.createChooser(i, "Enviar e-mail mediante:"));

////////////////////////////////////
////////////////////////////////////
}

private class MyFTP extends AsyncTask<String, Integer, String> {

```

```

@Override
protected String doInBackground(String... params) {

    //Toast.makeText(getApplicationContext(), "Conectando",
    Toast.LENGTH_SHORT).show();
    FTPClient ftpClient=new FTPClient();
    try {
        //Creamos un objeto Cliente HTTP para manejar la petición
al servidor
        ftpClient.connect("147.83.135.204",113);
    } catch (Exception e) {
        return "Exception happened: " + e.getMessage();
    }
    try {
        logeado=ftpClient.login("RUBEN", "RubenTFM");
    } catch (SocketException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

////////////////////////////////////
////////////////////////////////////

ftpClient.enterLocalActiveMode();
try {
    ftpClient.setFileType(FTP.BINARY_FILE_TYPE);
    ftpClient.setFileTransferMode(ftpClient.BINARY_FILE_TYPE);
    ftpClient.enterLocalPassiveMode();
    //Cambia la carpeta Ftp
    //if (ftpClient.changeWorkingDirectory("ftp")){
    //Obtiene la dirección de la ruta
    //Obtiene la ruta completa donde se encuentra el archivo
    //rutaCompleta = new File(rutaSd.getAbsolutePath(),
"prueba.txt");
    rutaCompleta=new
File(getFilesDir()+"/"+"aceleraciones.txt");
    rutaCompleta.setReadable(true,false);
    //Crea un buffer hacia el servidor de subida
    buffer = new BufferedInputStream(new
FileInputStream(rutaCompleta));

    if (ftpClient.storeFile(nombre_archivo, buffer)){
        buffer.close(); //Cierra el bufer
    }
    else{
        buffer.close(); //Cierra el bufer
    }

} catch (IOException e) {
    e.printStackTrace();
}

////////////////////////////////////
////////////////////////////////////

return "Hola";
}

```

```
        protected void onProgressUpdate(Integer... progress) {
            //Se obtiene el progreso de la petición
            Toast.makeText(getApplicationContext(), "Indicador de progreso
" + progress[0].toString(), Toast.LENGTH_SHORT).show();
        }

        protected void onPostExecute(String result) {
            //Se obtiene el resultado de la petición Asincrona
            //Log.w(APP_TAG,"Resultado obtenido " + result);
            //processResult(result);
        }
    }
}
```

8. Layout modo debug

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/segundo_layout">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:paddingBottom="16dp">

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:orientation="vertical">

            <TextView
                android:id="@+id/et1"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="AccX" />

            <TextView
                android:id="@+id/et2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="AccY" />

            <TextView
                android:id="@+id/et3"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="AccZ" />

        </LinearLayout>

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:orientation="vertical">

            <TextView
                android:id="@+id/et4"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Latitud" />

            <TextView
                android:id="@+id/et5"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Longitud" />

            <TextView
                android:id="@+id/et6"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Tiempo" />

</LinearLayout>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:id="@+id/et7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="AccX" />

        <TextView
            android:id="@+id/et8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="AccY" />

        <TextView
            android:id="@+id/et9"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="AccZ" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:id="@+id/et10"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Latitud" />

        <TextView
            android:id="@+id/et11"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Longitud" />

        <TextView
            android:id="@+id/et12"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```



```
        android:text="Tiempo" />

    </LinearLayout>

</LinearLayout>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:text="Name"
    android:ems="10"
    android:gravity="center"
    android:id="@+id/crono"
    tools:text="00:00:00"
    android:textSize="24sp"
    android:textColor="@android:color/holo_blue_dark" />

<ToggleButton android:id="@+id/btnActualizar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:textOn="GPS"
    android:textOff="GPS" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <Button
            android:id="@+id/button2"
            android:onClick="iniciar_registro"
            android:layout_height="50dp"
            android:layout_width="50dp"
            android:layout_gravity="center_horizontal"
            android:background="@drawable/botonplay" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

        <Button
            android:id="@+id/button3"
            android:onClick="detener_registro"
            android:layout_height="50dp"
            android:layout_width="50dp"
            android:layout_gravity="center_horizontal"
```

```
        android:background="@drawable/botonpause" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button4"
        android:onClick="borrar_datos"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botondelete" />

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button"
        android:onClick="compartir"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/botoncompartir" />

</LinearLayout>

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_gravity="center_horizontal"
    android:paddingTop="16dp"
    android:paddingBottom="16dp">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:orientation="vertical">

</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

</LinearLayout>
```

```
<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">
</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <Button
        android:id="@+id/button5"
        android:onClick="guardar"
        android:layout_height="50dp"
        android:layout_width="50dp"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/tarjeta" />

</LinearLayout>

</LinearLayout>

</LinearLayout>
```

9. Android manifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.elseib.ruben.proyecto_uni">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".PantallaInicio"
            android:theme="@style/Theme.AppCompat.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>
        <activity
            android:name=".MainActivity"
            android:label="Menu principal" />
        <activity
            android:name=".Velocimetros"
            android:label="Modo grafico" />
        <activity
            android:name=".ModoDebug"
            android:label="Modo Debug"></activity>
    </application>

</manifest>
```

10. Código de Matlab

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Lectura del fichero
nombre_archivo='ciclo movil fijado';

t=xlsread(nombre_archivo,'H2:H8000');
t=t/1000;
v=xlsread(nombre_archivo,'F2:F8000');
ax=xlsread(nombre_archivo,'A2:A8000');
ax=-ax;
% ax=resample(ax,100,1);
ay=xlsread(nombre_archivo,'B2:B8000');
% ay=resample(ay,100,1);
az=xlsread(nombre_archivo,'C2:C8000');
% az=resample(az,100,1);
n=size(ay)-1;
n=n(1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Filtro sgolay
axf=sgolayfilt(ax,1,17);
ayf=sgolayfilt(ay,1,17);
azf=sgolayfilt(az,1,17);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Aceleracion original y filtrada
figure(1)
hold on
grid on
hk=plot(t,ax,'b-');
hz=plot(t,axf,'r-');
legend([hk hz],'Aceleración sin filtro','Aceleración filtrada',0)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Filtro pasa bajos
gravax=zeros(n+1,1);
for i=2:1:n+1
    gravax(i)=0.999* gravax(i-1)+0.001*axf(i);
end

figure(2)
hold on
grid on
hz=plot(t,axf,'r-');
hk=plot(t,sgolayfilt(ax-gravax,1,17),'b-');
legend([hz hk],'Aceleración total','Aceleración filtro pasa bajos',0)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Cambio de base
theta=zeros(n+1,1);
modulo = zeros(n+1,1);
compX = zeros(n+1,1);
compY = zeros(n+1,1);
diferencias=zeros(n+1,1);

for i=2:1:n+1
    diferencias(i)=abs(axf(i)-axf(i-1));

```

```

end

theta(1)=acos (azf(1)/sqrt (axf(1)^2+ayf(1)^2+azf(1)^2));
modulo(i) = sqrt(axf(1)^2+ayf(1)^2);
compX(i) = ayf(1) / modulo(1);
compY(i) = -axf(1) / modulo(1);

for i=2:1:n-49
    if sum(diferencias(i:i+50)<0.02)==50
        theta(i)=acos (azf(i)/sqrt (axf(i)^2+ayf(i)^2+azf(i)^2));
        modulo(i) = sqrt(axf(i)^2+ayf(i)^2);
        compX(i) = ayf(i) / modulo(i);
        compY(i) = -axf(i) / modulo(i);
    else
        theta(i)=theta(i-1);
        modulo(i) = modulo(i-1);
        compX(i) = compX(i-1);
        compY(i) = compY(i-1);
    end
end

aax=zeros(n+1,1);
aay=zeros(n+1,1);
aaz=zeros(n+1,1);

for i=2:1:n+1
    aax(i)=- (cos(theta(i))+compX(i)^2*(1-cos(theta(i))))*axf(i)-
(compX(i)*compY(i)*(1-cos(theta(i))))*ayf(i)-compY(i)*sin(theta(i))*azf(i);
end

% for i=2:1:n+1
%     aay(i)=- (compX(i)*compY(i)*(1-cos(theta(i))))*axf(i)-
%     (cos(theta(i))+compY(i)^2*(1-
%     cos(theta(i))))*ayf(i)+compX(i)*sin(theta(i))*azf(i);
% end
%
% for i=2:1:n+1
%     aaz(i)=+compY(i)*sin(theta(i))*axf(i)-compX(i)*sin(theta(i))*ayf(i)-
%     cos(theta(i))*azf(i);
% end

figure(3)
hold on
grid on
% plot a-posteriori state estimates:
hk=plot(t,-sgolayfilt(aax,1,17),'b-');
hz=plot(t,axf,'r-');
legend([hk hz], 'Aceleracion cambio de base', 'Aceleracion filtrada
original',0)
hold off

%%%%%%Cálculo de offsets
desplazamientos=zeros(n+1,1);
axd=zeros(n+1,1);

for i=2:1:n-49
    if sum(diferencias(i:i+50)<0.02)==50
        desplazamientos(i)=axf(i);
    else

```

```

        desplazamientos(i)=desplazamientos(i-1);
    end
end

axd=axf-desplazamientos;

figure(4)
hold on
grid on
% plot a-posteriori state estimates:
hk=plot(t,sgolayfilt(axd,1,17),'b-');
hz=plot(t,axf,'r-');
legend([hk hz], 'Aceleración offseteada', 'Aceleración filtrada original', 0)
hold off

%%%%%%%%%%%%Filtro de Kalman
ax2=resample(ax,10,1);

clear s
s.A = 1;
s.Q = 0.001; % variancia, stdev^2
s.H = 1;
s.R = 0.1; % variancia, stdev^2
s.B = 0;
s.u = 0;
% No se define estado inicial
s.x = nan;
s.P = nan;

% Generar las entradas del sistema
for i=1:size(ax)
    s(end).z = ax(i); % crear una medida
    s(end+1)=kalmanf2(s(end)); % hacer iteración de Kalman
end

figure(5)
hold on
grid on
% plot la aceleración original
hz=plot(t,[s(1:end-1).z], 'r');
% plot la aceleración filtrada
hk=plot(t,[s(2:end).x], 'b-');
legend([hz hk], 'Acleración original', 'Aceleración Kalman', 0)
% title('Aceleracion Kalman Q=0.001 R=0.074')
hold off

%%%%%%%%%%%%Velocidad del GPS para Advisor

v=xlsread('ciclo movil fijado','F2:F15175');
plot(v(1:15000)*3.6)
vf=zeros(15000,1);

j=0;
for i=1:1:1500
    vf(i)=sum(v(10*j+1:10*j+10))/10;

```

```

j=j+1;
end
vf=vf(1:1500);
vf=vf*2.23694;

t=zeros(1500,1);
for i=1:1:1500
t(i)=i-1;
end

A=[t vf];

```

```

function s = kalmanf2(s)

% valores por defecto
if ~isfield(s,'x'); s.x=nan*z; end
if ~isfield(s,'P'); s.P=nan; end
if ~isfield(s,'z'); error('Observation vector missing'); end
if ~isfield(s,'u'); s.u=0; end
if ~isfield(s,'A'); s.A=eye(length(x)); end
if ~isfield(s,'B'); s.B=0; end
if ~isfield(s,'Q'); s.Q=zeros(length(x)); end
if ~isfield(s,'R'); error('Observation covariance missing'); end
if ~isfield(s,'H'); s.H=eye(length(x)); end

if isnan(s.x)
% inicializar para primera observacion
if diff(size(s.H))
error('Observation matrix must be square and invertible for state
autoinitialization. ');
end
s.x = inv(s.H)*s.z;
s.P = inv(s.H)*s.R*inv(s.H');
else
% prediccion
s.x = s.A*s.x + s.B*s.u;
s.P = s.A * s.P * s.A' + s.Q;

% ganancia kalman
K = s.P*s.H'*inv(s.H*s.P*s.H'+s.R);

% Correccion
s.x = s.x + K*(s.z-s.H*s.x);
s.P = s.P - K*s.H*s.P;

end

return

```


11. Fragmento del ciclo realizado

Debido a su excesiva extensión a continuación se adjunta solo un fragmento del ciclo analizado en la memoria del presente proyecto.

AccX[m/s ²]	AccY[m/s ²]	AccZ[m/s ²]	Latitud	Longitud	Velocidad[m/s]
0,6380547	2,2661119	9,689334	null	null	0
0,6356605	2,2577322	9,791088	null	null	0
0,61171854	2,2816741	9,733627	null	null	0
0,62728083	2,207454	9,730036	null	null	0
0,62608373	2,267309	9,691729	null	null	0
0,61291564	2,2697031	9,7324295	null	null	0
0,62967503	2,2409728	9,745598	null	null	0
0,641646	2,219425	9,764751	null	null	0
0,6392518	2,2062569	9,713276	null	null	0
0,61411273	2,2505496	9,779117	null	null	0
0,61890113	2,220622	9,787497	null	null	0
0,61530983	2,2409728	9,728839	null	null	0
0,61890113	2,219425	9,781511	null	null	0
0,6452374	2,2780828	9,706094	null	null	0
0,6452374	2,267309	9,776723	null	null	0
0,62129533	2,2409728	9,744401	null	null	0
0,6356605	2,2756886	9,720459	null	null	0
0,62488663	2,1811178	9,753978	null	null	0
0,62847793	2,2697031	9,691729	null	null	0
0,62368953	2,2050598	9,7719345	null	null	0
0,62009823	2,232593	9,731233	null	null	0
0,6344634	2,243367	9,783905	null	null	0
0,61171854	2,2002714	9,742006	null	null	0
0,63206923	2,255338	9,792285	null	null	0
0,62488663	2,208651	9,7863	null	null	0
0,6404489	2,27928	9,727641	null	null	0
0,62728083	2,2110453	9,731233	null	null	0
0,63087213	2,268506	9,701305	null	null	0
0,62368953	2,220622	9,800665	null	null	0
0,62009823	2,2625206	9,726444	null	null	0
0,6440403	2,2278047	9,7719345	null	null	0
0,62488663	2,2110453	9,7324295	null	null	0
0,6356605	2,2625206	9,7803135	null	null	0
0,62488663	2,207454	9,7958765	null	null	0
0,62009823	2,2481554	9,730036	null	null	0
0,6548142	2,2625206	9,733627	null	null	0
0,6428431	2,2732944	9,68694	null	null	0
0,62728083	2,2720973	9,770737	null	null	0
0,61052144	2,2337902	9,755175	null	null	0
0,6356605	2,292448	9,739613	null	null	0

0,61770403	2,2158337	9,713276	null	null	0
0,6392518	2,2589293	9,709685	null	null	0
0,6368576	2,207454	9,733627	null	null	0
0,62488663	2,2242134	9,755175	null	null	0
0,6332663	2,2589293	9,7563715	null	null	0
0,59256494	2,1978772	9,713276	null	null	0
0,6596026	2,280477	9,763555	null	null	0
0,62967503	2,2541409	9,753978	null	null	0
0,6404489	2,2505496	9,745598	null	null	0
0,6655881	2,2493525	9,773131	null	null	0
0,62728083	2,232593	9,734824	null	null	0
0,6560113	2,2876596	9,7563715	null	null	0
0,62967503	2,2278047	9,785102	null	null	0
0,49919105	2,2541409	9,672575	null	null	0
0,5482722	2,2589293	9,7719345	null	null	0
0,5674258	2,2349873	9,740809	null	null	0
0,48362875	2,243367	9,750386	null	null	0
0,59615624	2,1942859	9,765948	null	null	0
0,59735334	2,2744915	9,765948	null	null	0
0,59376204	2,1978772	9,779117	null	null	0
0,6512229	2,2900538	9,728839	null	null	0
0,5506664	2,3415294	9,811439	null	null	0
0,59256494	2,2349873	9,788693	null	null	0
0,39504328	2,3894134	9,874885	null	null	0
0,320823	2,2218192	9,821015	null	null	0
0,17118542	2,4372973	9,625888	null	null	0
0,019153614	2,2517467	9,758766	null	null	0
-0,046686932	2,3116016	9,834184	null	null	0
-0,06703765	2,3199813	9,792285	null	null	0
-0,12569559	2,2493525	9,720459	null	null	0
-0,17836803	2,219425	9,696517	null	null	0
-0,31603462	2,2589293	9,759963	null	null	0
-0,30047232	2,2146366	9,794679	null	null	0
-0,2741361	2,2589293	9,88925	null	null	0
-0,25857377	2,304419	9,818622	null	null	0
-0,31004912	2,2972364	9,692925	null	null	0
-0,23104046	2,2768857	9,787497	null	null	0
-0,30166942	2,220622	9,894038	null	null	0
-0,37828386	2,1380222	9,831789	null	null	0
-0,3196259	2,195483	9,8916445	null	null	0
-0,46208093	2,135628	9,794679	null	null	0
-0,5327099	2,1332338	9,800665	null	null	0
-0,58657944	2,1811178	9,734824	null	null	0
-0,52073884	2,1511903	9,872491	null	null	0
-0,59495914	2,2014685	9,832987	null	null	0
-0,439336	2,1212628	9,939528	null	null	0

-0,21547815	2,3307555	9,76954	null	null	0
-0,37110126	2,0207062	9,774328	null	null	0
-0,36511576	2,098518	9,757569	null	null	0
-0,31364042	2,3127987	9,630676	null	null	0
-0,19512744	2,2050598	9,7958765	null	null	0
-0,23822308	1,9847932	9,813833	null	null	0
-0,2705448	2,337938	9,640253	null	null	0
-0,4333505	2,148796	9,5385	null	null	0
-0,49919105	2,2373815	9,712079	null	null	0
-0,37588966	2,0518308	9,793482	null	null	0
0,17238252	2,231396	9,731233	null	null	0
0,5398925	2,2541409	9,657013	null	null	0
0,6871359	2,1883004	10,135853	null	null	0
0,6979098	2,1260512	10,189722	null	null	0
0,7374141	2,0901382	9,526528	null	null	0
0,7613561	2,2900538	9,483433	null	null	0
0,6560113	2,360683	9,283517	null	null	0
0,82480246	1,8543092	9,504981	null	null	0
1,2509704	2,2290018	9,999383	41,4038	2,1393	0
1,7549498	2,4863784	10,479421	41,4038	2,1393	0
1,9476831	1,9393034	10,230424	41,4038	2,1393	0
1,8303672	2,0111294	9,55047	41,4038	2,1393	0
1,6208745	1,9081788	9,224859	41,4038	2,1393	0
1,5251064	1,9835961	9,544485	41,4038	2,1393	0
1,30484	2,4672248	9,689334	41,4038	2,1393	0
0,9935937	1,6998832	9,068039	41,4038	2,1393	0
0,6512229	1,7310078	9,033323	41,4038	2,1393	0
0,12449849	2,6527755	9,455899	41,4038	2,1393	0
0,1340753	2,4073699	9,870096	41,4038	2,1393	0
0,4237737	2,5019407	10,036493	41,4038	2,1393	0
0,7697359	2,7736826	9,951499	41,4038	2,1393	0
0,65242	2,4528596	9,562442	41,4038	2,1393	0
1,2497733	2,9652188	10,15381	41,4038	2,1393	0
1,5969325	3,1962593	11,071986	41,4038	2,1393	0
1,282095	2,019509	10,59554	41,4038	2,1393	0
0,8523358	2,553416	9,31943	41,4038	2,1393	0
0,6787562	2,9580362	8,735245	41,4038	2,1393	0
0,8451532	3,8067808	8,711303	41,4038	2,1393	0
1,0594343	3,8534677	9,414001	41,4038	2,1393	0
0,35075054	3,7720647	9,950302	41,4038	2,1393	0
0,22864626	3,526659	9,85214	41,4038	2,1393	0
0,38187516	3,8031895	9,400833	41,4038	2,1393	0
0,075417355	4,1096473	9,837775	41,4039	2,1396	1,78
0,36511576	3,764882	9,55047	41,4039	2,1396	1,78
0,23343466	3,4823663	9,838972	41,4039	2,1396	1,78
-0,017956512	2,5151088	10,028113	41,4039	2,1396	1,78

0,06703765	3,3363202	9,90601	41,4039	2,1396	1,78
0,4369418	4,283227	9,233239	41,4039	2,1396	1,78
0,17836803	4,307169	9,305065	41,4039	2,1396	1,78
-0,039504327	3,8774097	9,421184	41,4039	2,1396	1,78
-0,059855044	3,3806129	9,696517	41,4039	2,1396	1,78
-0,24899697	3,1040826	9,455899	41,4039	2,1396	1,78
-0,20350714	3,2944214	9,643845	41,4039	2,1396	1,78
-0,37708676	3,110068	9,605537	41,4039	2,1396	1,78
-0,5734113	2,8634653	9,67856	41,4039	2,1396	2,73
-0,6452374	2,731784	9,740809	41,4039	2,1396	2,73
-0,6380547	2,7006595	9,730036	41,4039	2,1396	2,73
-0,6631939	2,8215668	9,779117	41,4039	2,1396	2,73
-0,6787562	2,6384103	9,949105	41,4039	2,1396	2,73
-0,6979098	2,553416	9,793482	41,4039	2,1396	2,73
-0,92535895	2,3774424	9,56723	41,4039	2,1396	2,73
-0,8523358	2,3187842	9,64145	41,4039	2,1396	2,73
-0,8631097	2,3295584	9,965864	41,4039	2,1396	2,73
-0,91937345	2,1571758	9,829395	41,4039	2,1396	2,73
-0,7314286	2,2649148	9,835381	41,4038	2,1396	3,14
-1,0007763	1,9991584	9,866506	41,4038	2,1396	3,14
-0,7517793	1,8303672	10,013748	41,4038	2,1396	3,14
-0,8834604	1,6340426	9,764751	41,4038	2,1396	3,14
-0,9935937	1,543063	9,791088	41,4038	2,1396	3,14
-0,91697925	1,5646108	9,980229	41,4038	2,1396	3,14
-0,90261406	1,459266	9,919178	41,4038	2,1396	3,14
-1,0067618	1,4520833	10,0664215	41,4038	2,1396	3,14
-0,9864111	1,3862427	9,988609	41,4038	2,1396	3,14
-0,80684596	1,7166426	9,824607	41,4038	2,1396	3,14
-1,0582372	1,6903064	9,886856	41,4038	2,1396	3,14
-0,90141696	1,6555905	9,847352	41,4038	2,1396	3,14
-1,0139444	1,4077905	10,0903635	41,4038	2,1396	3,14
-0,78410107	1,520318	9,922769	41,4038	2,1396	3,14
-0,8619126	1,4987702	10,055647	41,4038	2,1396	3,14
-0,5506664	1,5011644	10,055647	41,4038	2,1396	3,14
-0,80445176	1,5442601	10,160992	41,4038	2,1396	3,14
-0,7493851	1,4999673	10,031705	41,4038	2,1396	3,14
-0,81881696	1,4736311	9,799467	41,4038	2,1396	3,14
-0,78290397	1,5598224	9,797073	41,4038	2,1396	3,14
-0,830788	1,8183962	9,952697	41,4038	2,1394	4,33
-0,79966336	1,388637	10,081984	41,4038	2,1394	4,33
-0,8523358	1,4544775	10,038888	41,4038	2,1394	4,33
-0,8834604	1,8327614	10,09994	41,4038	2,1394	4,33
-0,79248077	1,5574282	10,010158	41,4038	2,1394	4,33
-0,8643068	1,4987702	9,958682	41,4038	2,1394	4,33
-1,0283096	1,4987702	9,921572	41,4038	2,1394	4,33
-0,9516952	0,9923966	10,113108	41,4038	2,1394	4,33

-0,8631097	1,5358804	9,846154	41,4038	2,1394	4,33
-0,91338795	1,328782	9,831789	41,4039	2,1391	6,19
-0,89662856	1,5251064	9,915586	41,4039	2,1391	6,19
-1,0618285	1,7824832	10,049662	41,4039	2,1391	6,19
-0,93493575	1,4987702	10,107122	41,4039	2,1391	6,19
-0,8726865	1,6496049	10,087969	41,4039	2,1391	6,19
-0,8858546	1,6519991	9,989806	41,4039	2,1391	6,19
-0,90620536	1,6496049	9,81503	41,4039	2,1391	6,19
-0,91099375	1,483208	9,896433	41,4039	2,1391	6,19
-0,9744401	1,5801731	10,022128	41,4039	2,1391	6,19
-0,90859956	1,5107412	10,0341	41,4039	2,1391	6,19
-0,8343793	1,6256629	9,85214	41,4039	2,1391	6,19
-0,91458505	1,4185646	9,956287	41,4039	2,1391	6,19
-0,79726917	1,4544775	9,85214	41,4039	2,1391	6,19
-0,830788	1,8291701	10,103531	41,4039	2,1391	6,19
-0,8571242	1,3934253	9,849746	41,4039	2,1391	6,19
-0,79487497	1,2437878	9,799467	41,4039	2,1391	6,19
-0,90859956	1,4041992	9,994595	41,4039	2,1391	6,19
-0,8702923	1,532289	9,690532	41,4039	2,1391	6,19
-0,29209262	1,6089035	9,807847	41,4039	2,1391	6,19
0,17238252	1,4185646	9,879673	41,4038	2,1389	6,85