ANEXO C. CÓDIGO DE LAS CLASES DE LA APLICACIÓN ANDROID

Para el proyecto "Automatización de un campo de tiro configurado y controlado mediante un dispositivo móvil", se utilizó Android Studio 2.0 para el desarrollo de la aplicación Android. A continuación se incluye, la arquitectura de programación, el código de cada clase .java y el código de la interfaz grafica de cada activity en .XML, y en diseño.

1. ARQUITECTURA DE PROGRAMACIÓN

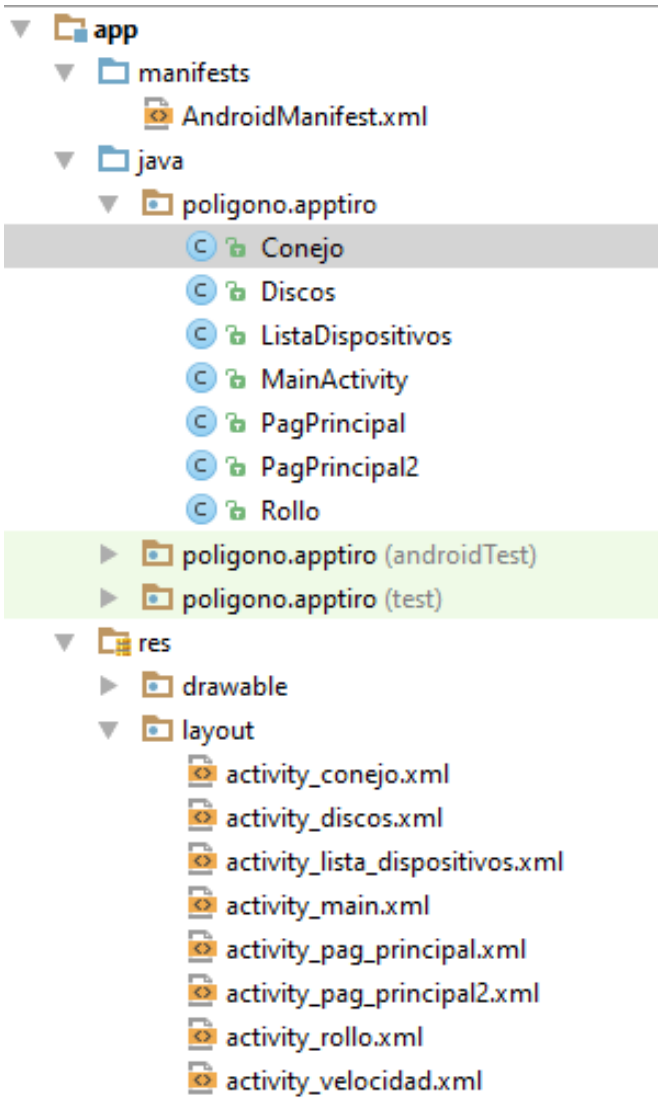


Figura 1. Arquitectura de programación de la aplicación AppTiro.

2. CÓDIGO DE CADA CLASE .java.

2.1. Conejo.java

```
package poligono.apptiro;

import android.app.ProgressDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.util.UUID;

public class Conejo extends AppCompatActivity {

    Button startc,puntuacionc,mejorpuntuacionc;
    PagPrincipal2 conej = new PagPrincipal2();
    String y=null;
    TextView ejem,puntaje;
    Spinner lista;
    String [] datos = {"","1 m/s","2 m/s","3 m/s","4 m/s","5 m/s"};

    String address = "null";
    private ProgressDialog progrso;
    BluetoothAdapter myBluetooth = null;
    BluetoothSocket btsocket = null;
    private boolean isbtconectado = false;
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-
00805F9B34FB");

    Handler bluetoothIn;
    final int handlerState = 0;
    private StringBuilder recDataString = new StringBuilder();
    String dataInPrint;
    private ConnectedThread mConnectedThread;

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

        Intent newint = getIntent();
        address = newint.getStringExtra(PagPrincipal2.ADDRESS);

        startc = (Button)findViewById(R.id.button11);
        puntuacionc = (Button)findViewById(R.id.button12);
        mejorpuntuacionc = (Button)findViewById(R.id.button13);
```

```
        ejem = (TextView) findViewById(R.id.textView7);
        puntaje = (TextView) findViewById(R.id.textView10);
        lista= (Spinner) findViewById(R.id.spinner);

       new ConnectBT().execute();

        bluetoothIn = new Handler() {


            public void handleMessage(android.os.Message msg) {
                if (msg.what == handlerState) {          //if message is what we
want
                    String readMessage = (String) msg.obj;
// msg.arg1 = bytes from connect thread
                    recDataString.append(readMessage);              //keep
appending to string until ~
                    int endOfLineIndex = recDataString.indexOf("~");
// determine the end-of-line
                    if (endOfLineIndex > 0) {
// make sure there data before ~
                        dataInPrint = recDataString.substring(0, endOfLineIndex);
// extract string
                        ejem.setText("Datos recibidos = " + dataInPrint);
                        int dataLength = dataInPrint.length();       //get length
of data received
                        //txtStringLength.setText("Tamaño del String = " +
String.valueOf(dataLength));

                    }
                    recDataString.delete(0, recDataString.length());
//clear all string data
                    // strIncom =" ";
                }


            }
        };

        ArrayAdapter<String> adapt = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_item,datos);
        lista.setAdapter(adapt);

        lista.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener()
{
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {

                switch (position) {
                    case 1:
                        y = "3";
                        ejem.setText(y);

                        if (btsocket!=null)
                        {
                            try
                            {

btsocket.getOutputStream().write(y.toString().getBytes());

                            }
                            catch (IOException e)
                            {
```

```
                                msg("Error");
                            }
                        }

                        Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                        conej.EnviarDato(y, ejem);

                        break;

                    case 2:
                        y = "4";
                        ejem.setText(y);

                        if (btsocket!=null)
                        {
                            try
                            {

btsocket.getOutputStream().write(y.toString().getBytes());

                            }
                            catch (IOException e)
                            {
                                msg("Error");
                            }
                        }
                        Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                        conej.EnviarDato(y, ejem);
                        break;

                    case 3:
                        y = "5";
                        ejem.setText(y);

                        if (btsocket!=null)
                        {
                            try
                            {

btsocket.getOutputStream().write(y.toString().getBytes());

                            }
                            catch (IOException e)
                            {
                                msg("Error");
                            }
                        }
                        Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                        conej.EnviarDato(y, ejem);
                        break;

                    case 4:
                        y = "6";
                        ejem.setText(y);

                        if (btsocket!=null)
                        {
                            try
                            {
```

```
btsocket.getOutputStream().write(y.toString().getBytes());

                            }
                            catch (IOException e)
                            {
                                msg("Error");
                            }
                        }
                        Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                        conej.EnviarDato(y, ejem);
                        break;

                    case 5:
                        y = "7";
                        ejem.setText(y);

                        if (btsocket!=null)
                        {
                            try
                            {

btsocket.getOutputStream().write(y.toString().getBytes());

                            }
                            catch (IOException e)
                            {
                                msg("Error");
                            }
                        }
                        Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                        conej.EnviarDato(y, ejem);
                        break;

                    default:
                        break;

                }
            }

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

            }
        });

        startc.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                y = "8";
                ejem.setText(y);

                if (btsocket!=null)
                {
                    try
                    {

btsocket.getOutputStream().write(y.toString().getBytes());

                    }
                    catch (IOException e)
                    {
```

```
                        msg("Error");
                    }
                }
                Toast.makeText(getApplicationContext(), y,
Toast.LENGTH_SHORT).show();
                conej.EnviarDato(y, ejem);
            }
        });

        puntuacionc.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                puntaje.setText(dataInPrint);
            }
        });

        mejorpuntuacionc.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                puntaje.setText(dataInPrint);
            }
        });


    }

    private void msg(String s)
    {
        Toast.makeText(getApplicationContext(), s, Toast.LENGTH_LONG).show();
    }


    @Override
    public boolean onKeyDown(int keyCode, KeyEvent event) {
        // TODO Auto-generated method stub
        if (keyCode == event.KEYCODE_BACK) {
            if (btsocket!=null)
            {
                try
                {
                    btsocket.close();
                }
                catch (IOException e)
                { msg("Error");}
            }
        }
        return super.onKeyDown(keyCode, event);
    }
   public class ConnectBT extends AsyncTask<Void, Void, Void>  // UI thread
    {
        private boolean ConnectSuccess = true;

        @Override
        protected void onPreExecute()
        {
            progrso = ProgressDialog.show(Conejo.this, "Connecting...", "Please
wait!!!");
        }

        @Override
        protected Void doInBackground(Void... devices)
```

```
        {
            try
            {
                if (btsocket == null || !isbtconectado)
                {
                    myBluetooth = BluetoothAdapter.getDefaultAdapter();
                    BluetoothDevice dispositivo =
myBluetooth.getRemoteDevice(address);//conectamos al dispositivo y chequeamos si
esta disponible
                    btsocket =
dispositivo.createInsecureRfcommSocketToServiceRecord(myUUID);
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    btsocket.connect();
                    mConnectedThread = new ConnectedThread(btsocket);
                    mConnectedThread.start();
                }
            }
            catch (IOException e)
            {
                ConnectSuccess = false;
            }
            return null;
        }
        @Override
        protected void onPostExecute(Void result)
        {
            super.onPostExecute(result);

            if (!ConnectSuccess)
            {
               // msg("Conexión Fallida");
                finish();
            }
            else
            {
                //  msg("Conectado");
                isbtconectado = true;
            }
            progrso.dismiss();
        }
    }

    private class ConnectedThread extends Thread {
        private final InputStream mmInStream;
        private final OutputStream mmOutStream;

        //creation of the connect thread
        public ConnectedThread(BluetoothSocket socket) {
            InputStream tmpIn = null;
            OutputStream tmpOut = null;

            try {
                //Create I/O streams for connection
                tmpIn = socket.getInputStream();
                tmpOut = socket.getOutputStream();
            } catch (IOException e) { }

            mmInStream = tmpIn;
            mmOutStream = tmpOut;
        }


        public void run() {
```

```
            byte[] buffer = new byte[256];
            int bytes;

            // Keep looping to listen for received messages
            while (true) {
                try {
                    bytes = mmInStream.read(buffer);         //read bytes from
input buffer
                    String readMessage = new String(buffer, 0, bytes);
                    // Send the obtained bytes to the UI Activity via handler
                    bluetoothIn.obtainMessage(handlerState, bytes, -1,
readMessage).sendToTarget();
                } catch (IOException e) {
                    break;
                }
            }
        }

    }
}
```

2.2. Discos.java

```
package poligono.apptiro;

import android.app.Activity;
import android.app.ProgressDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.TextView;
import android.widget.Toast;
import android.os.Handler;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.UnsupportedEncodingException;
import java.util.UUID;

import java.util.logging.LogRecord;

public class Discos extends Activity {

    String x = null;
    Button Start,Restart,puntuacion,mejorpuntuacion;
    PagPrincipal2 disco = new PagPrincipal2();
    TextView ejemplo,puntaje;
    String address = "null";
    private ProgressDialog progrso;

    BluetoothAdapter myBluetooth = null;
    BluetoothSocket btsocket = null;
```

```
    private boolean isbtconectado = false;
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-
00805F9B34FB");

    Handler bluetoothIn;
    final int handlerState = 0;
    private StringBuilder recDataString = new StringBuilder();
    String dataInPrint;
    private ConnectedThread mConnectedThread;

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

        Intent newint = getIntent();
        address = newint.getStringExtra(PagPrincipal2.ADDRESS);


        Start= (Button) findViewById(R.id.button6);
        Restart= (Button) findViewById(R.id.button7);
        puntuacion= (Button)findViewById(R.id.button8);
        mejorpuntuacion= (Button)findViewById(R.id.button9);
        ejemplo = (TextView) findViewById(R.id.textView6);
        puntaje = (TextView)findViewById(R.id.textView9);

        new ConnectBT().execute();

        bluetoothIn = new Handler() {


            public void handleMessage(android.os.Message msg) {
                if (msg.what == handlerState) {          //if message is what we
want
                    String readMessage = (String) msg.obj;
// msg.arg1 = bytes from connect thread
                    recDataString.append(readMessage);              //keep
appending to string until ~
                    int endOfLineIndex = recDataString.indexOf("~");
// determine the end-of-line
                    if (endOfLineIndex > 0) {
// make sure there data before ~
                        dataInPrint = recDataString.substring(0, endOfLineIndex);
// extract string
                        ejemplo.setText("Datos recibidos = " + dataInPrint);
                        int dataLength = dataInPrint.length();       //get length
of data received
                        //txtStringLength.setText("Tamaño del String = " +
String.valueOf(dataLength));

                    }
                    recDataString.delete(0, recDataString.length());
//clear all string data
                    // strIncom =" ";
                }


            }
        };
```

```
        Start.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
               x = "1";
                ejemplo.setText(x);

                if (btsocket!=null)
                {
                    try
                    {

btsocket.getOutputStream().write(x.toString().getBytes());

                    }
                    catch (IOException e)
                    {
                        msg("Error");
                    }
                    Toast.makeText(getApplicationContext(), x,
Toast.LENGTH_SHORT).show();
                    disco.EnviarDato(x, ejemplo);
                }
            }

        });
        Restart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                x = "2";
                ejemplo.setText(x);

                if (btsocket!=null)
                {
                    try
                    {

btsocket.getOutputStream().write(x.toString().getBytes());

                    }
                    catch (IOException e)
                    {
                        msg("Error");
                    }
                }

                Toast.makeText(getApplicationContext(), x,
Toast.LENGTH_SHORT).show();
                disco.EnviarDato(x, ejemplo);
            }
        });


        puntuacion.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                puntaje.setText(dataInPrint);

            }
        });

        mejorpuntuacion.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
                puntaje.setText(dataInPrint);
            }
        });
    }

    @Override
    public boolean onKeyDown(int keyCode, KeyEvent event) {
        // TODO Auto-generated method stub
        if (keyCode == event.KEYCODE_BACK) {
            if (btsocket!=null)
            {
                try
                {
                    btsocket.close();
                }
                catch (IOException e)
                { msg("Error");}
            }
        }
        return super.onKeyDown(keyCode, event);
    }


    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_pagprincipal2, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);

    }

    private void msg(String s)
    {
        Toast.makeText(getApplicationContext(), s, Toast.LENGTH_LONG).show();
    }


    public class ConnectBT extends AsyncTask<Void, Void, Void>  // UI thread
    {
        private boolean ConnectSuccess = true;

        @Override
        protected void onPreExecute()
        {
            progrso = ProgressDialog.show(Discos.this, "Connecting...", "Please
wait!!!");
        }

        @Override
        protected Void doInBackground(Void... devices)
        {
            try
            {
```

```
                if (btsocket == null || !isbtconectado)
                {
                    myBluetooth = BluetoothAdapter.getDefaultAdapter();
                    BluetoothDevice dispositivo =
myBluetooth.getRemoteDevice(address);//conectamos al dispositivo y chequeamos si
esta disponible
                    btsocket =
dispositivo.createInsecureRfcommSocketToServiceRecord(myUUID);
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    btsocket.connect();
                    mConnectedThread = new ConnectedThread(btsocket);
                    mConnectedThread.start();

                }
            }
            catch (IOException e)
            {
                ConnectSuccess = false;
            }
            return null;
        }
        @Override
        protected void onPostExecute(Void result)
        {
            super.onPostExecute(result);

            if (!ConnectSuccess)
            {
               // msg("Conexión Fallida");
                finish();
            }
            else
            {
              //  msg("Conectado");
                isbtconectado = true;
            }
            progrso.dismiss();
        }
    }


    private class ConnectedThread extends Thread {
        private final InputStream mmInStream;
        private final OutputStream mmOutStream;
        //creation of the connect thread
        public ConnectedThread(BluetoothSocket socket) {
            InputStream tmpIn = null;
            OutputStream tmpOut = null;

            try {
                //Create I/O streams for connection
                tmpIn = socket.getInputStream();
                tmpOut = socket.getOutputStream();
            } catch (IOException e) { }

            mmInStream = tmpIn;
            mmOutStream = tmpOut;
        }
        public void run() {
            byte[] buffer = new byte[256];
            int bytes;

            // Keep looping to listen for received messages
```

```
            while (true) {
                try {
                    bytes = mmInStream.read(buffer);         //read bytes from
input buffer
                    String readMessage = new String(buffer, 0, bytes);
                    // Send the obtained bytes to the UI Activity via handler
                    bluetoothIn.obtainMessage(handlerState, bytes, -1,
readMessage).sendToTarget();
                } catch (IOException e) {
                    break;
                }
            }
        }

    }
}
```

2.3. ListaDispositivos.java

```
package poligono.apptiro;

import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.Set;

public class ListaDispositivos extends AppCompatActivity {

    Button dispovinculados;
    ListView listadispositivos;
    private BluetoothAdapter myBluetooth = null;
    private Set<BluetoothDevice> listVinculados;
    public static String EXTRA_ADDRESS = "device_address";


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


        dispovinculados = (Button) findViewById(R.id.button5);
        listadispositivos = (ListView) findViewById(R.id.listView);

        myBluetooth = BluetoothAdapter.getDefaultAdapter();

        if(myBluetooth == null)
        {
            //Mostramos un mensaje, indicando al usuario que no tiene conexión
```

```
bluetooth disponible
            Toast.makeText(getApplicationContext(), "Bluetooth no disponible",
Toast.LENGTH_LONG).show();

            //finalizamos la aplicación
            finish();
        }
        else if(!myBluetooth.isEnabled())
        {
            //Preguntamos al usuario si desea encender el bluetooth
            Intent turnBTon = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            startActivityForResult(turnBTon,1);
        }

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

    }
    private  void  listadispositivosvinculados()
    {
        listVinculados = myBluetooth.getBondedDevices();
        ArrayList list = new ArrayList();

        if (listVinculados.size()>0)
        {
            for (BluetoothDevice bt : listVinculados)
            {
                list.add(bt.getName() + "\n"+ bt.getAddress()); // se obtienen
nombres y direcciones de los dispositivos vinculados
            }
        }
        else
        {
            Toast.makeText(getApplicationContext(), "No se han encontrado
dispositivos vinculados", Toast.LENGTH_LONG).show();
        }
        final ArrayAdapter adapter = new
ArrayAdapter(this,android.R.layout.simple_list_item_1, list);
        listadispositivos.setAdapter(adapter);
        listadispositivos.setOnItemClickListener(MyListListener);
    }

    private AdapterView.OnItemClickListener MyListListener = new
AdapterView.OnItemClickListener(){

        @Override
        public void onItemClick(AdapterView<?> parent, View v, int position, long
id) {
            String info = ((TextView) v).getText().toString();
            String address = info.substring(info.length() - 17);

            Intent i = new Intent(ListaDispositivos.this,PagPrincipal2.class);
            i.putExtra(EXTRA_ADDRESS,address);


            startActivity(i);

        }
    };
```

```
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_listadispositivos, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
}
```

2.4. MainActivity.java

```
package poligono.apptiro;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

2.5. PagPrincipal.java

```
package poligono.apptiro;

import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class PagPrincipal extends AppCompatActivity {
    BluetoothAdapter newbluetooth = null;
    private static final int onbluetooth = 1;

    Button Conectar,Discos,Conejo,Rollo;

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

```
        //Verificar que el bluetooth este encendido:
            newbluetooth = BluetoothAdapter.getDefaultAdapter();

        if (newbluetooth == null){
            //se muestra un mensaje de no es posible conectar
            Toast.makeText(getApplicationContext(), "No es posible conectar el
bluetooth", Toast.LENGTH_LONG).show();
        }
        else if (!newbluetooth.isEnabled()) {
            //Se activa el bluetooth por medio de un Intent que es como decirle
ejecute
            Intent activarbluetooth = new
Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            startActivityForResult(activarbluetooth, onbluetooth);
        }

        //Se obtiene la respuesta de la clase broadcastreceiver
        IntentFilter filter = new
IntentFilter(BluetoothAdapter.ACTION_STATE_CHANGED);
        registerReceiver(broadcastReceiver,filter);

        //Intanciación de los botones
        Conectar = (Button)findViewById(R.id.button);
        Discos = (Button)findViewById(R.id.button2);
        Conejo = (Button)findViewById(R.id.button3);
        Rollo = (Button)findViewById(R.id.button4);

        //Discos(es un boton). Aqui se le agrega la accion que deber realizar
cuando sea presionado
        Discos.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(), "Por favor, conecte el
dispositivo", Toast.LENGTH_LONG).show();
            }
        });

        Conejo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(), "Por favor, conecte el
dispositivo", Toast.LENGTH_LONG).show();
            }
        });

        Rollo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(), "Por favor, conecte el
dispositivo", Toast.LENGTH_LONG).show();
            }
        });


    }
    //Esta clase lo que hace es verificar todo el tiempo la conexion con
bluetooth
    private final BroadcastReceiver broadcastReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            final String action = intent.getAction();

            if (action.equals(BluetoothAdapter.ACTION_STATE_CHANGED)){
```

```
                final int estado =
intent.getIntExtra(BluetoothAdapter.EXTRA_STATE,BluetoothAdapter.ERROR);
                switch (estado){
                    case  BluetoothAdapter.STATE_OFF:
                        Toast.makeText(getApplicationContext(), "Bluetooth
desactivado. Aplicación deshabilitada.", Toast.LENGTH_LONG).show();

                        break;

                    case BluetoothAdapter.STATE_ON:
                        Toast.makeText(getApplicationContext(), "Se ha activado
el bluetooth ", Toast.LENGTH_LONG).show();
                        break;

                    default:
                        break;

                }
            }
        }
    };

    protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
        //super.onActivityResult(requestCode, resultCode, data);
        switch (requestCode) {
            case onbluetooth:
                if (resultCode == Activity.RESULT_OK) {


                } else {
                    Toast.makeText(getApplicationContext(), "La aplicación será
cerrada", Toast.LENGTH_LONG).show();

                    finish();
                }
                break;


        }
    }

    public void irconexion (View view){
        startActivity(new Intent(this, ListaDispositivos.class));

    }
}
```

2.6. PagPrincipal2.java

```
package poligono.apptiro;

import android.app.ProgressDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.ActionBarActivity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
```

```
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.io.InputStream;
import java.util.UUID;

public class PagPrincipal2 extends ActionBarActivity {
    Button salir, discos,conejo,rollo;
    String address = null;

    private ProgressDialog progrso;
    public static String ADDRESS = "device_address";


    BluetoothAdapter myBluetooth = null;
    BluetoothSocket btsocket = null;
    private boolean isbtconectado = false;
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-
00805F9B34FB");
    TextView textView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pag_principal2);
        new ConnectBT().execute();
        //recibimos la MAC obtenida
        Intent newint = getIntent();
        address = newint.getStringExtra(ListaDispositivos.EXTRA_ADDRESS);
        //textView = (TextView)findViewById(R.id.textView);

        salir = (Button) findViewById(R.id.button);
        discos= (Button) findViewById(R.id.button2);
        conejo= (Button) findViewById(R.id.button3);
        rollo = (Button) findViewById(R.id.button4);



        //Desconectarse del dispositivo:
        salir.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(), "dispositivo
Desconectado", Toast.LENGTH_SHORT).show();
                Disconnect();

            }
        });

        discos.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

               // textView.setText(address);
                if (btsocket!=null)
                {
                    try
                    {
                        btsocket.close();
```

```
                }
                catch (IOException e)
                { msg("Error");}
            }
            Intent i = new Intent(PagPrincipal2.this,Discos.class);
            i.putExtra(ADDRESS, address);
            startActivity(i);
        }
    });

    conejo.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            //textView.setText(address);
            if (btsocket!=null)
            {
                try
                {
                    btsocket.close();

                }
                catch (IOException e)
                { msg("Error");}
            }
            Intent i = new Intent(PagPrincipal2.this,Conejo.class);
            i.putExtra(ADDRESS, address);
            startActivity(i);

        }
    });

    rollo.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            if (btsocket != null) {
              //  textView.setText(address);
                if (btsocket!=null)
                {
                    try
                    {
                        btsocket.close();

                    }
                    catch (IOException e)
                    { msg("Error");}
                }
                Intent i = new Intent(PagPrincipal2.this,Rollo.class);
                i.putExtra(ADDRESS, address);
                startActivity(i);
            }
        }
    });



}

private void Disconnect()
{
    if (btsocket!=null)
    {
```

```
            try
            {
                btsocket.close();
                finish();
            }
            catch (IOException e)
            { msg("Error");}
        }

    }

    @Override
    public boolean onKeyDown(int keyCode, KeyEvent event) {
        // TODO Auto-generated method stub
        if (keyCode == event.KEYCODE_BACK) {
            if (btsocket!=null)
            {
                try
                {
                    btsocket.close();
                }
                catch (IOException e)
                { msg("Error");}
            }
        }
        return super.onKeyDown(keyCode, event);
    }

    public void EnviarDato(String x,TextView textView)
    {

        textView.setText(x);
        //Envia dato al arduino:
        if (btsocket!=null)
        {
            try
            {
                btsocket.getOutputStream().write(x.toString().getBytes());

            }
            catch (IOException e)
            {
                msg("Error");
            }
        }

    }

    public int RecibirDato () {

        byte[] buffer = new byte[8];
       // int bytes=20;
        //Buscara continuamente mensajes recibidos
        while (true) {
            try {
                InputStream mmInstream = btsocket.getInputStream();
                int readmessage = mmInstream.read(buffer);
               /* btsocket.getInputStream().read(buffer); //lee bytes desde el
buffer de entrada
                String readmessage = new String(buffer, 0, bytes);*/

                return readmessage;
```

```
            } catch (IOException e) {
                msg("Error");
            }

        }

    }

    private void msg(String s)
    {
        Toast.makeText(getApplicationContext(), s, Toast.LENGTH_LONG).show();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_pagprincipal2, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);

    }

    public class ConnectBT extends AsyncTask<Void, Void, Void>  // UI thread
    {
        private boolean ConnectSuccess = true;

        @Override
        protected void onPreExecute()
        {
            progrso = ProgressDialog.show(PagPrincipal2.this, "Connecting...",
"Please wait!!!");
        }

        @Override
        protected Void doInBackground(Void... devices)
        {
            try
            {
                if (btsocket == null || !isbtconectado)
                {
                    myBluetooth = BluetoothAdapter.getDefaultAdapter();
                    BluetoothDevice dispositivo =
myBluetooth.getRemoteDevice(address);//conectamos al dispositivo y chequeamos si
esta disponible
                    btsocket =
dispositivo.createInsecureRfcommSocketToServiceRecord(myUUID);
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    btsocket.connect();
                }
            }
            catch (IOException e)
            {
                ConnectSuccess = false;
            }
```

```
            return null;
        }
        @Override
        protected void onPostExecute(Void result)
        {
            super.onPostExecute(result);

            if (!ConnectSuccess)
            {
                msg("Conexión Fallida");
                finish();
            }
            else
            {
                msg("Conectado");
                isbtconectado = true;
            }
            progrso.dismiss();
        }
    }
}
```

2.7. Rollo.java

```
package poligono.apptiro;

import android.app.ProgressDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.util.UUID;

public class Rollo extends AppCompatActivity {

    Button cambiar,puntuacionr,mejorpuntuacionr;
    PagPrincipal2 objrollo = new PagPrincipal2();
    String z = null;
    TextView ejem3,puntaje;
    String address = "null";
    private ProgressDialog progrso;
    BluetoothAdapter myBluetooth = null;
    BluetoothSocket btsocket = null;
    private boolean isbtconectado = false;
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-
00805F9B34FB");

    Handler bluetoothIn;
    final int handlerState = 0;
    private StringBuilder recDataString = new StringBuilder();
    String dataInPrint;
```

```
   private ConnectedThread mConnectedThread;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_rollo);
        Intent newint = getIntent();
        address = newint.getStringExtra(PagPrincipal2.ADDRESS);

        cambiar= (Button)findViewById(R.id.button14);
        puntuacionr = (Button) findViewById(R.id.button15);
        mejorpuntuacionr =(Button) findViewById(R.id.button16);
        ejem3 = (TextView)findViewById(R.id.textView8);
        puntaje = (TextView)findViewById(R.id.textView11);

        new ConnectBT().execute();

        bluetoothIn = new Handler() {


            public void handleMessage(android.os.Message msg) {
                if (msg.what == handlerState) {          //if message is what we
want
                    String readMessage = (String) msg.obj;
// msg.arg1 = bytes from connect thread
                    recDataString.append(readMessage);              //keep
appending to string until ~
                    int endOfLineIndex = recDataString.indexOf("~");
// determine the end-of-line
                    if (endOfLineIndex > 0) {
// make sure there data before ~
                        dataInPrint = recDataString.substring(0, endOfLineIndex);
// extract string
                        ejem3.setText("Datos recibidos = " + dataInPrint);
                        int dataLength = dataInPrint.length();       //get length
of data received
                        //txtStringLength.setText("Tamaño del String = " +
String.valueOf(dataLength));

                    }
                    recDataString.delete(0, recDataString.length());
//clear all string data
                    // strIncom =" ";
                }


            }
        };


        cambiar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                 z = "9";
                ejem3.setText(z);

                if (btsocket!=null)
                {
                    try
                    {

btsocket.getOutputStream().write(z.toString().getBytes());
```

```
                    }
                    catch (IOException e)
                    {
                        msg("Error");
                    }
                }
                Toast.makeText(getApplicationContext(), z,
Toast.LENGTH_SHORT).show();
                objrollo.EnviarDato(z, ejem3);
            }
        });

        puntuacionr.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
               puntaje.setText(dataInPrint);

            }
        });

        mejorpuntuacionr.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                puntaje.setText(dataInPrint);

            }
        });
    }



    private void msg(String s)
    {
        Toast.makeText(getApplicationContext(), s, Toast.LENGTH_LONG).show();
    }

    @Override
    public boolean onKeyDown(int keyCode, KeyEvent event) {
        // TODO Auto-generated method stub
        if (keyCode == event.KEYCODE_BACK) {
            if (btsocket!=null)
            {
                try
                {
                    btsocket.close();
                }
                catch (IOException e)
                { msg("Error");}
            }
        }
        return super.onKeyDown(keyCode, event);
    }

    public class ConnectBT extends AsyncTask<Void, Void, Void>  // UI thread
    {
        private boolean ConnectSuccess = true;

        @Override
        protected void onPreExecute()
        {
            progrso = ProgressDialog.show(Rollo.this, "Connecting...", "Please
wait!!!");
        }
```

```
        @Override
        protected Void doInBackground(Void... devices)
        {
            try
            {
                if (btsocket == null || !isbtconectado)
                {
                    myBluetooth = BluetoothAdapter.getDefaultAdapter();
                    BluetoothDevice dispositivo =
myBluetooth.getRemoteDevice(address);//conectamos al dispositivo y chequeamos si
esta disponible
                    btsocket =
dispositivo.createInsecureRfcommSocketToServiceRecord(myUUID);
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    btsocket.connect();
                    mConnectedThread = new ConnectedThread(btsocket);
                    mConnectedThread.start();
                }
            }
            catch (IOException e)
            {
                ConnectSuccess = false;
            }
            return null;
        }
        @Override
        protected void onPostExecute(Void result)
        {
            super.onPostExecute(result);

            if (!ConnectSuccess)
            {
              //  msg("Conexión Fallida");
                finish();
            }
            else
            {
                //  msg("Conectado");
                isbtconectado = true;
            }
            progrso.dismiss();
        }
    }
    private class ConnectedThread extends Thread {
        private final InputStream mmInStream;
        private final OutputStream mmOutStream;

        //creation of the connect thread
        public ConnectedThread(BluetoothSocket socket) {
            InputStream tmpIn = null;
            OutputStream tmpOut = null;

            try {
                //Create I/O streams for connection
                tmpIn = socket.getInputStream();
                tmpOut = socket.getOutputStream();
            } catch (IOException e) { }

            mmInStream = tmpIn;
            mmOutStream = tmpOut;
        }
```

```
        public void run() {
            byte[] buffer = new byte[256];
            int bytes;

            // Keep looping to listen for received messages
            while (true) {
                try {
                    bytes = mmInStream.read(buffer);         //read bytes from
input buffer
                    String readMessage = new String(buffer, 0, bytes);
                    // Send the obtained bytes to the UI Activity via handler
                    bluetoothIn.obtainMessage(handlerState, bytes, -1,
readMessage).sendToTarget();
                } catch (IOException e) {
                    break;
                }
            }
        }

    }
}
```

3. CÓDIGO DE CADA CLASE .XML.

3.1. activity_conejo.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.Conejo"
    android:background="#091e0e">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView2"
        android:src="@drawable/con_fondo"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true" />

    <ImageView
        android:layout_width="270dp"
        android:layout_height="45dp"
        android:id="@+id/imageView6"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="150dp"
        android:background="@drawable/con_btn1" />

    <Spinner
        android:layout_width="150dp"
```

```
        android:layout_height="50dp"
        android:id="@+id/spinner"
        android:layout_marginTop="220dp"
        android:layout_centerHorizontal="true"
        android:background="@drawable/con_btn2"
        android:spinnerMode="dialog" />

    <Button
        android:layout_width="150dp"
        android:layout_height="30dp"
        android:id="@+id/button11"
        android:layout_below="@+id/spinner"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:background="@drawable/con_btn3" />

    <Button
        android:layout_width="150dp"
        android:layout_height="30dp"
        android:id="@+id/button12"
        android:layout_marginTop="330dp"
        android:layout_centerHorizontal="true"
        android:background="@drawable/con_btn4" />

    <Button
        android:layout_width="150dp"
        android:layout_height="60dp"
        android:id="@+id/button13"
        android:layout_marginTop="370dp"
        android:layout_centerHorizontal="true"
        android:background="@drawable/con_btn5" />

    <TextView
        android:layout_width="60dp"
        android:layout_height="20dp"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Tx"
        android:id="@+id/textView7"
        android:layout_marginTop="200dp"
        android:layout_marginBottom="30dp"
        android:layout_marginStart="90dp"
        android:layout_marginEnd="10dp" />

    <TextView
        android:layout_width="70dp"
        android:layout_height="20dp"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Rx"
        android:id="@+id/textView10"
        android:layout_marginTop="200dp"
        android:layout_marginBottom="30dp"
        android:layout_marginStart="200dp"
        android:layout_marginEnd="10dp" />
</RelativeLayout>
```

3.2. activity_discos.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context="poligono.apptiro.Discos"
    android:background="#091e0e">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:src="@drawable/fondo_discos"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true" />

    <ImageView
        android:layout_width="270dp"
        android:layout_height="100dp"
        android:id="@+id/imageView5"
        android:layout_marginTop="150dp"
        android:src="@drawable/disc_btn1"
        android:layout_centerHorizontal="true"/>

    <Button
        android:layout_width="140dp"
        android:layout_height="30dp"
        android:id="@+id/button6"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="230dp"
        android:background="@drawable/disc_btn2" />

    <Button
        android:layout_width="140dp"
        android:layout_height="30dp"
        android:id="@+id/button7"
        android:layout_below="@+id/button6"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="30dp"
        android:background="@drawable/disc_btn3" />

    <Button
        android:layout_width="140dp"
        android:layout_height="30dp"
        android:id="@+id/button8"
        android:layout_below="@+id/button7"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="30dp"
        android:background="@drawable/disc_btn4" />
```

```
    <Button
        android:layout_width="140dp"
        android:layout_height="60dp"
        android:id="@+id/button9"
        android:layout_below="@+id/button8"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="30dp"
        android:background="@drawable/disc_btn5" />

    <TextView
        android:layout_width="60dp"
        android:layout_height="20dp"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Tx"
        android:id="@+id/textView6"
        android:layout_marginTop="380dp"
        android:layout_marginBottom="30dp"
        android:layout_marginStart="250dp"
        android:layout_marginEnd="10dp" />

    <TextView
        android:layout_width="60dp"
        android:layout_height="20dp"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Rx"
        android:id="@+id/textView9"
        android:layout_marginTop="420dp"
        android:layout_marginBottom="30dp"
        android:layout_marginStart="250dp"
        android:layout_marginEnd="10dp" />

</RelativeLayout>
```

3.2. activity_lista_dispositivos.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.ListaDispositivos">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mostrar Dispositivos vinculados"
        android:id="@+id/button5"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:background="#000000"
        android:textColor="#ffffff" />
```

```
    <ListView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/listView"
        android:layout_centerHorizontal="true"
        android:layout_below="@+id/button5" />

</RelativeLayout>
```

3.3. activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.MainActivity">

</RelativeLayout>
```

3.4. activity_pag_principal.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.PagPrincipal"
    android:background="#091e0e">

    <ImageView
        android:layout_width="400dp"
        android:layout_height="500dp"
        android:id="@+id/imageView4"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true"
        android:src="@drawable/ini_fondo" />

    <Button
        android:layout_width="200dp"
        android:layout_height="30dp"
        android:id="@+id/button"
        android:layout_marginTop="30dp"
        android:background="@drawable/ini_conec_disp"
```

```
        android:layout_marginStart="110dp"
        android:onClick="irconexion"/>

    <Button
        android:layout_width="130dp"
        android:layout_height="30dp"
        android:id="@+id/button2"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="185dp"
        android:background="@drawable/ini_btn2" />

    <Button
        android:layout_width="130dp"
        android:layout_height="30dp"
        android:id="@+id/button3"
        android:layout_alignTop="@+id/button2"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="60dp"
        android:background="@drawable/ini_btn3" />

    <Button
        android:layout_width="130dp"
        android:layout_height="60dp"
        android:id="@+id/button4"
        android:layout_marginTop="31dp"
        android:background="@drawable/ini_btn4"
        android:layout_below="@+id/button3"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

3.5. activity_pag_principal2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.PagPrincipal2"
    android:background="#091e0e">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView4"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true"
        android:src="@drawable/ini_fondo" />

    <Button
```

```
        android:layout_width="200dp"
        android:layout_height="30dp"
        android:id="@+id/button"
        android:layout_marginTop="30dp"
        android:background="@drawable/ini_desc_salir"
        android:layout_marginStart="110dp" />

    <Button
        android:layout_width="130dp"
        android:layout_height="30dp"
        android:id="@+id/button2"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="185dp"
        android:background="@drawable/ini_btn2" />

    <Button
        android:layout_width="130dp"
        android:layout_height="30dp"
        android:id="@+id/button3"
        android:layout_alignTop="@+id/button2"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="60dp"
        android:background="@drawable/ini_btn3" />

    <Button
        android:layout_width="130dp"
        android:layout_height="60dp"
        android:id="@+id/button4"
        android:layout_marginTop="31dp"
        android:background="@drawable/ini_btn4"
        android:layout_below="@+id/button3"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

3.6. activity_rollo.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.Rollo"
    android:background="#091e0e">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView3"
        android:src="@drawable/fondo_rollo"
```

```
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true" />

    <ImageView
        android:layout_width="270dp"
        android:layout_height="100dp"
        android:id="@+id/imageView7"
        android:layout_marginTop="145dp"
        android:background="@drawable/roll_btn1"
        android:layout_centerHorizontal="true"/>

    <Button
        android:layout_width="150dp"
        android:layout_height="60dp"
        android:id="@+id/button14"
        android:layout_below="@+id/imageView7"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="15dp"
        android:background="@drawable/roll_btn2" />

    <Button
        android:layout_width="150dp"
        android:layout_height="30dp"
        android:id="@+id/button15"
        android:layout_below="@+id/button14"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:background="@drawable/roll_btn3" />

    <Button
        android:layout_width="150dp"
        android:layout_height="60dp"
        android:id="@+id/button16"
        android:layout_below="@+id/button15"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true"
        android:background="@drawable/roll_btn4" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Tx"
        android:id="@+id/textView8"
        android:layout_alignTop="@+id/button14"
        android:layout_alignParentEnd="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:text="Dato Rx"
        android:id="@+id/textView11"
        android:layout_below="@+id/button14"
        android:layout_alignParentEnd="true" />

</RelativeLayout>
```

3.7. activity_velocidad.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    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="poligono.apptiro.Velocidad">

</RelativeLayout>
```

4. Diseño de cada clase.

4.1. activity_conejo



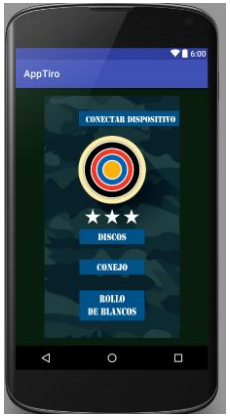
4.2. activity_discos



4.3. activity_lista_dispositivos



4.4. activity_pag_principal



4.5. activity_pag_principal2



4.6. activity_rollo

