ACTIVIDAD 6. Visualización de datos – Aplicaciones Móviles

Inicio

El objetivo de esta actividad es entender los conceptos básicos relacionados con desarrollo de aplicaciones móviles, como mecanismo para la visualización de los datos.

Trabajaremos con Node-RED donde recibimos, almacenamos los datos y se crearon los puntos de acceso usando HTTP. Estos puntos de acceso los usaremos en una aplicación móvil para acceder a los datos y mostrarlos al usuario.

Usaremos como herramienta de desarrollo Android Studio.

Instalación y configuración del ambiente de desarrollo

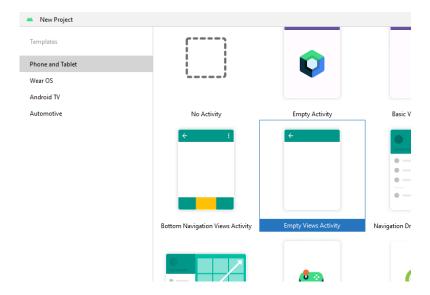
1. Instalación de Android Studio

De la página oficial de Android Studio: https://developer.android.com/studio se descarga la ultima versión y se instala.

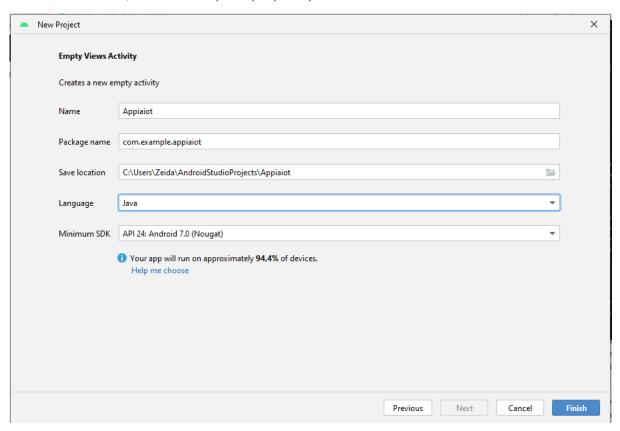
Se recomienda desarrollar el Hello World para que se acabe de configurar y descargar todo el ambiente completo.

Creación del proyecto

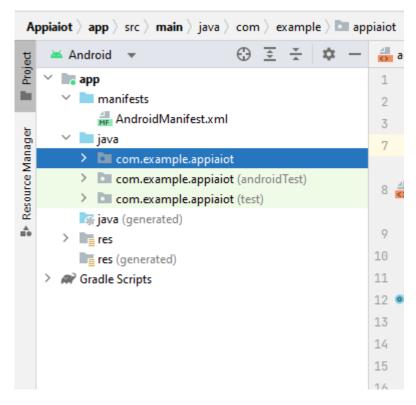
Creamos un nuevo proyecto y seleccionamos la plantilla vacía.



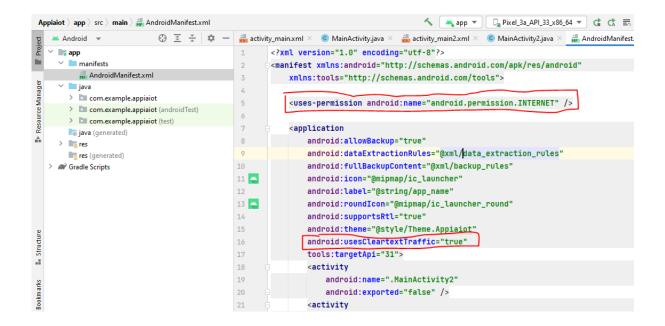
Configuramos los parámetros del proyecto: nombre (appiaiot), lenguaje de desarrollo (java) y versión de Android, usaremos la que el propone por defecto.



Cuando se crea el proyecto automáticamente se crea la primera interfaz y la activity asociada y la estructura del proyecto, con todas las carpetas requeridas:



Configuramos los permisos necesarios para la aplicación, que en este caso será el permiso de acceso a Internet, en el archivo AndroidManifest.xml



Configuramos en el Gradle las librerías que requerimos para nuestro proyecto. En este caso requerimos de la librería Volley que es la que nos permite realizar peticiones HTTP.

```
Appiaiot ⟩ app ⟩ ≈ build.gradle
                            🕲 📱 🛣 - 🕌 activity_main.xml × 🌘 MainActivity.java × 🏭 activity_main2.xml × 🌘 MainActivity2.java × 🟭 AndroidManifest.xm
   You can use the Project Structure dialog to view and edit your project configuration

✓ ■ manifests

         AndroidManifest.xml
                                                                   minifyEnabled false
    🗸 🖿 java
                                                                   proquardFiles getDefaultProguardFile('proguard-android-optimize.txt')
      > com.example.appiaiot
     > 🖿 com.example.appiaiot (androidTest)
      > com.example.appiaiot (test)
                                                          compileOptions {
                                              25
       🔯 java (generated)
    > 📭 res
                                              26
                                                               sourceCompatibility JavaVersion.VERSION_1_8
      res (generated)
                                                               targetCompatibility JavaVersion.VERSION_1_8
  Gradle Scripts
                                              28
       w build.gradle (Project: Appiaiot)
                                                      ₽}-
      m build.gradle (Module :app)
       proguard-rules.pro (ProGuard Rules for ":app")
                                                      dependencies {
       gradle.properties (Project Properties)
                                              32
       gradle-wrapper.properties (Gradle Version)
                                                           implementation 'androidx.appcompat:appcompat:1.4.1'
       | local.properties (SDK Location)
                                                          implementation 'com.google.android.material:material:1.5.0'
                                              34
       ettings.gradle (Project Settings)
                                              35
                                                          implementation 'androidx.constraintlayout:constraintlayout:2.1.3'
                                              36
                                                           testImplementation 'junit:junit:4.13.2'
                                              37
                                                           androidTestImplementation 'androidx.test.ext:junit:1.1.3'
                                                           androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
                                              38
                                              39
                                                           implementation 'com.android.volley:volley:1.2.1
                                              40
```

Layout y Activity para hacer un Login

```
Layout: activity_main.xml:

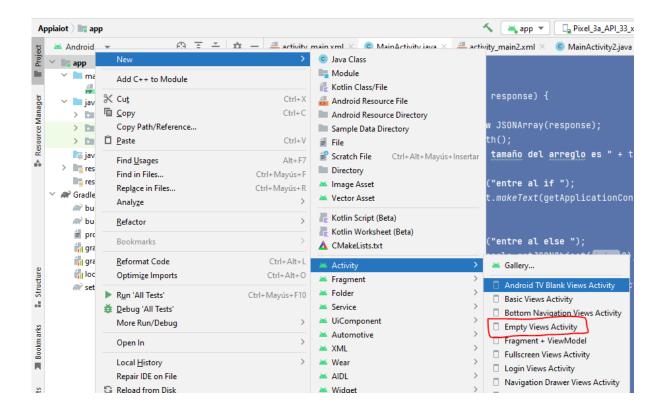
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout android:layout_height="match_parent"

android:layout_width="match_parent"</pre>
```

```
android:orientation="vertical"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <TextView
        android:layout width="match parent"
        android:layout height="wrap content"
        android:text="Ingresa tu usuario:"/>
    <EditText
        android:id="@+id/user"
        android:layout width="match parent"
        android: layout height="wrap content"
        android:inputType="text"/>
    <TextView
        android:layout width="match parent"
        android:layout height="wrap content"
        android:text="Ingresa tu password:"/>
    <EditText
        android:id="@+id/pass"
        android:layout_width="match parent"
        android: layout height="wrap content"
        android:inputType="textPassword"/>
    <Button
        android:id="@+id/bot1"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:text="Ingresar"/>
</LinearLayout>
La activity: MainActivity.java
package com.example.appiaiot;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.concurrent.atomic.AtomicInteger;
public class MainActivity extends AppCompatActivity {
   private EditText user, pass;
   private Button bot1;
   private RequestQueue cola;
   private StringRequest peticion;
```

```
@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main);
        user = findViewById(R.id.user);
        pass = findViewById(R.id.pass);
        bot1 = findViewById(R.id.bot1);
        bot1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String usuario = user.getText().toString();
                String password = pass.getText().toString();
                validarUsuario(usuario, password);
                user.setText("");
                pass.setText("");
        });
    }
    private void validarUsuario(String usuario, String password) {
        String url =
"http://10.0.2.2:1880/usuario?user="+usuario+"&pass="+password;
        System.out.println("la url es "+url);
        cola = Volley.newRequestQueue(this);
        peticion = new StringRequest(Request.Method.GET, url,
this::onResponse,
                (error) -> { error.printStackTrace();}
        );
        cola.add(peticion);
    }
   private void onResponse(String response) {
        try {
            JSONArray arreglo = new JSONArray(response);
            int tam = arreglo.length();
            System.out.println("el tamaño del arreglo es " + tam);
            if (tam == 0) {
                System.out.println("entre al if ");
                Toast toast = Toast.makeText(getApplicationContext(),
"Credenciales invalidas", Toast. LENGTH LONG);
                toast.show();
            } else {
                System.out.println("entre al else ");
                JSONObject js = arreglo.getJSONObject(0);
                String us = js.getString("user");
                Intent intent = new Intent (MainActivity.this,
MainActivity2.class);
                intent.putExtra("user", us);
                startActivity(intent);
            }
        } catch (JSONException e) {
            throw new RuntimeException(e);
    }
}
```

Creamos otra activity que será a la que se acceda cuando las credenciales sean correctas:



Los códigos serían los siguientes:

Layout: activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout android:layout height="match parent"</pre>
    android:layout width="match parent"
    android:orientation="vertical"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <TextView
        android:id="@+id/nameUser"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</LinearLayout>
Activity: MainActivity2.java
package com.example.appiaiot;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
public class MainActivity2 extends AppCompatActivity {
    private TextView nameUser;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main2);
```

Layout y Activity para mostrar los últimos datos

Cambiamos el código de la segunda activity para que contenga los últimos datos. Se está haciendo de manera muy simple, usando un Linear Layout y TextViews

Los códigos serían los siguientes:

Layout: activity main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout android:layout height="match parent"</pre>
    android:layout width="match parent"
    android:orientation="vertical"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android">
   <TextView
        android:id="@+id/nameUser"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:layout gravity="center"
        android: textSize="40sp"/>
    <TextView
        android:id="@+id/accx"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/accy"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/accz"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:textSize="20sp"/>
    <TextView
        android:id="@+id/rotx"
        android: layout width="wrap_content"
        android:layout height="wrap content"
        android: textSize="20sp"/>
    <TextView
        android:id="@+id/roty"
        android:layout width="wrap content"
        android: layout height="wrap content"
        android: textSize="20sp"/>
    <TextView
        android:id="@+id/rotz"
        android:layout width="wrap content"
```

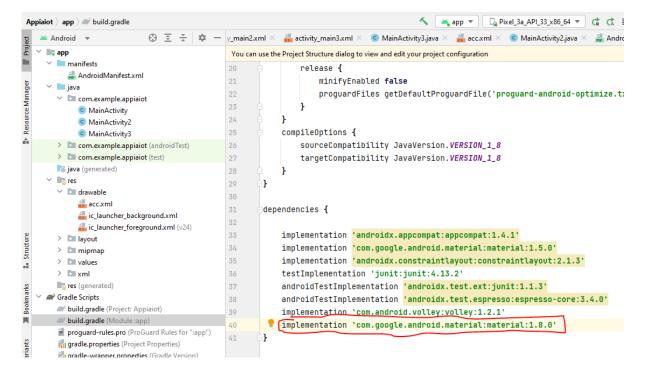
```
android:layout height="wrap content"
        android: textSize="20sp"/>
    <TextView
        android:id="@+id/temp"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android: textSize="20sp"/>
</LinearLayout>
Activity: MainActivity2.java
package com.example.appiaiot;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
public class MainActivity2 extends AppCompatActivity {
    private TextView nameUser, accx, accy, accz, rotx, roty, rotz, temp;
    private RequestQueue cola;
   private StringRequest peticion;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main2);
        nameUser = findViewById(R.id.nameUser);
        String us = getIntent().getExtras().getString("user");
        nameUser.setText("Datos del nodo del usuario: "+us);
        String url = "http://10.0.2.2:1880/datos-ultimo";
        System.out.println("la url es "+url);
        cola = Volley.newRequestQueue(this);
        peticion = new StringRequest(Request.Method.GET, url,
this::onResponse,
                (error) -> { error.printStackTrace();}
        cola.add(peticion);
   private void onResponse(String response) {
        try {
            JSONArray arreglo = new JSONArray(response);
            JSONObject js = arreglo.getJSONObject(0);
            String acc x = js.getString("acc x");
            String acc y = js.getString("acc y");
            String acc z = js.getString("acc_z");
```

```
String rot x = js.getString("rot x");
            String rot y = js.getString("rot y");
            String rot z = js.getString("rot z");
            String temperatura = js.getString("temperatura");
            accx = findViewById(R.id.accx);
            accy = findViewById(R.id.accy);
            accz = findViewById(R.id.accz);
            rotx = findViewById(R.id.rotx);
            roty = findViewById(R.id.roty);
            rotz = findViewById(R.id.rotz);
            temp = findViewById(R.id.temp);
            accx.setText("La aceleracion en x es: "+ acc x);
            accy.setText("La aceleracion en y es: "+ acc y);
            accz.setText("La aceleracion en z es: "+ acc z);
            rotx.setText("La rotacion en x es: "+ rot x);
            roty.setText("La rotacion en y es: "+ rot y);
            rotz.setText("La rotacion en z es: "+ rot z);
            temp.setText("la temperatura es "+ temperatura);
        } catch (JSONException e) {
            throw new RuntimeException(e);
   }
}
```

Layout y Activity para mostrar los últimos datos en Cards

Creamos una tercera activity en donde se mostrarán los últimos datos en Cards, para que se vean de una mejor manera. Se crea tanto el layout activity main3.xml y la activity MainActivity3.java.

Para poder usar los cards hay que agregar una librería, lo cual hacemos en el Gradle:



Los códigos serían los siguientes:

Layout: activity main3.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout</pre>
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android: layout height="match parent"
    android: layout width="match parent"
    android: gravity="center"
    android:orientation="vertical"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <TextView
        android:id="@+id/nameUser"
        android:layout_width="wrap_content"
        android:layout height="wrap content"
        android:layout gravity="center"
        android: textSize="40sp"/>
    <androidx.cardview.widget.CardView</pre>
        android:layout width="340dp"
        android:layout height="200dp"
        app:cardElevation="11dp"
        app:cardCornerRadius="5dp"
        android:layout margin="8dp"
        app:cardBackgroundColor="#FFB6C1"
        app:cardUseCompatPadding="true">
        <LinearLayout</pre>
            android:layout width="match parent"
            android:layout height="wrap content"
            android:orientation="vertical">
            <TextView
                android:layout width="match parent"
                android:layout height="wrap_content"
                android:text="ACELERACION EN X"
                android:textSize="24sp"
                android:gravity="center"
                android:padding="16dp"/>
            <TextView
                android:id="@+id/ax"
                android:layout width="match parent"
                android:layout height="wrap content"
                android:textSize="40sp"
                android:gravity="center"
                android:padding="16dp"/>
        </LinearLayout>
    </androidx.cardview.widget.CardView>
    <androidx.cardview.widget.CardView</pre>
        android:layout width="340dp"
        android:layout height="200dp"
        app:cardElevation="11dp"
        app:cardCornerRadius="5dp"
        android:layout margin="8dp"
        app:cardBackgroundColor="#008B8B"
        app:cardUseCompatPadding="true">
        <LinearLayout</pre>
            android:layout width="match parent"
            android:layout height="wrap content"
            android:orientation="vertical">
                <TextView
```

```
android:layout width="match parent"
                    android:layout height="wrap content"
                    android:text="ACELERACION EN Y"
                    android: textSize="24sp"
                    android:gravity="center"
                    android:padding="16dp"/>
                 <TextView
                    android:id="@+id/ay"
                    android:layout width="match parent"
                    android:layout height="wrap content"
                    android: textSize="40sp"
                    android:gravity="center"
                    android:padding="16dp"/>
        </LinearLayout>
    </androidx.cardview.widget.CardView>
    <androidx.cardview.widget.CardView</pre>
        android:layout width="340dp"
        android:layout height="200dp"
        app:cardElevation="11dp"
        app:cardCornerRadius="5dp"
        android:layout margin="8dp"
        app:cardBackgroundColor="#87cEFA"
        app:cardUseCompatPadding="true">
        <LinearLayout</pre>
            android:layout width="match parent"
            android:layout height="wrap content"
            android:orientation="vertical">
            <TextView
                android:layout width="match parent"
                android:layout height="wrap_content"
                android:text="ACELERACION EN Z"
                android:textSize="24sp"
                android:gravity="center"
                android:textColor="@color/design default color on primary"
                android:padding="16dp"/>
            <TextView
                android:id="@+id/az"
                android:layout width="match parent"
                android:layout height="wrap content"
                android: textSize="40sp"
                android:gravity="center"
                android:textColor="@color/design default color on primary"
                android:padding="16dp"/>
        </LinearLayout>
    </androidx.cardview.widget.CardView>
</LinearLayout>
Activity: MainActivity3.java
package com.example.appiaiot;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;
```

```
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
public class MainActivity3 extends AppCompatActivity {
    private TextView nameUser, accx, accy, accz;
    private RequestQueue cola;
    private StringRequest peticion;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main3);
        nameUser = findViewById(R.id.nameUser);
        String us = getIntent().getExtras().getString("user");
        nameUser.setText(us);
        String url = "http://10.0.2.2:1880/datos-ultimo";
        System.out.println("la url es "+url);
        cola = Volley.newRequestQueue(this);
        peticion = new StringRequest(Request.Method.GET, url,
this::onResponse,
                (error) -> { error.printStackTrace();}
        );
        cola.add(peticion);
    private void onResponse(String response) {
        try {
            JSONArray arreglo = new JSONArray(response);
            JSONObject js = arreglo.getJSONObject(0);
            String acc x = js.getString("acc x");
            String acc_y = js.getString("acc_y");
            String acc z = js.getString("acc z");
            accx = findViewById(R.id.ax);
            accy = findViewById(R.id.ay);
            accz = findViewById(R.id.az);
            accx.setText(acc x);
            accy.setText(acc y);
            accz.setText(acc z);
        } catch (JSONException e) {
            throw new RuntimeException(e);
    }
}
```

En la primera activity cambiamos el código para que llame a esta activity y no a la otra:

```
\textbf{Applaiot} \; \rangle \; \textbf{app} \; \rangle \; \text{src} \; \rangle \; \textbf{main} \; \rangle \; \text{java} \; \rangle \; \text{com} \; \rangle \; \text{example} \; \rangle \; \text{applaiot} \; \rangle \; \textcircled{\textbf{@}} \; \text{MainActivity} \; \rangle \; \textcircled{\textbf{m}} \; \text{onResponse}
                                                                                                          ⊕ 🚡 🛣 - © MainActivity.java × 🚔 activity_main2.xml ×
                                                                                                     🟭 activity_main3.xml × 🏿 © MainActivity3.java × 👼 acc.xml × 🕲 MainActivity2.java × 🚜 Android

Android 

▼
                                                      61
           # AndroidManifest.xml
       ijava java

✓ □ com.example.appiaiot

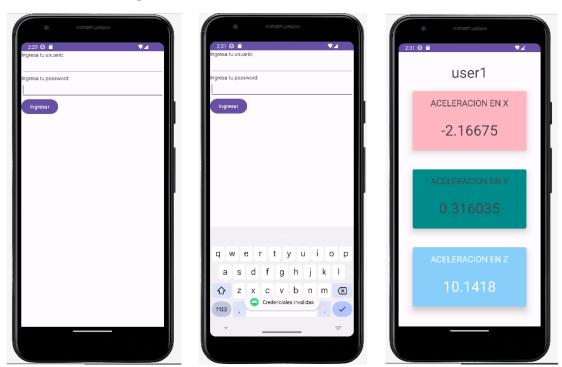
                                                                     private void onResponse(String response) {
      MainActivity
                                                                              JSONArray arreglo = new JSONArray(response);
              MainActivity3
                                                                               int tam = arreglo.length():
        > com.example.appiaiot (androidTest)
                                                                               System.out.println("el tamaño del arreglo es " + tam);
       > com.example.appiaiot (test)
        🖙 java (generated)
                                                                               System.out.println("entre al if ");
                                                                                    Toast toast = Toast.makeText(getApplicationContext(), text "Credenciales invalidas", Toa

✓ ☐ drawable

                                                                                    toast.show();
              🚜 acc.xml
                                                                               } else {
              alic_launcher_background.xml
              ic_launcher_foreground.xml (v24)
                                                                                    System.out.println("entre al else ");
        > 🛅 layout
                                                                                    JSONObject js = arreglo.getJSONObject( index: 0);
                                                                                   String us = js.getString( name: "user");

Intent intent = new Intent( packageContext MainActivity.this, MainActivity3.class);
        > 🖿 mipmap
        > 🖿 xml
                                                                                    intent.putExtra( name: "user", us);
        res (generated)
                                                                                    startActivity(intent);
   Gradle Scripts
        w build.gradle (Project: Appiaiot)
        w build.gradle (Module :app)
                                                                          } catch (JSONException e) {
        proguard-rules.pro (ProGuard Rules for ":app")
        gradle.properties (Project Properties)
                                                                               throw new RuntimeException(e);
        gradle-wrapper.properties (Gradle Version)
        local.properties (SDK Location)
```

El resultado será el siguiente:



Aplicación móvil completa

Añadamos a nuestra aplicación que el usuario pueda ver sus nodos, seleccione uno y ahí si se le muestren los datos asociados a ese nodo, en la pantalla deberá aparecer el nombre del usuario y el nombre del nodo.