



# UNIVERSIDAD PONTIFICIA DE SALAMANCA

Asignatura: Programación de Aplicaciones

Tema: Android

E-Mail: [acamarara.inf@upsa.es](mailto:acamarara.inf@upsa.es)

Universidad: UPSA

Alumno: Ángel Cámara Ramos

## Resumen:

He realizado una aplicación en Android mediante Android Studio en la que se puede almacenar rutas de senderismo, el programa te permite ver unas rutas ya incluidas en la aplicación (almacenadas en un repositorio), así como añadir una nueva ruta al sistema, además si haces click en una ruta te mostrará la información mas detallada, en un formato diferente.

## Clases del programa:

### 1. MainActivity

La clase MainActivity es una actividad de la aplicación Android que muestra una lista de rutas de senderismo utilizando un RecyclerView. Esta actividad también proporciona la funcionalidad para crear nuevas rutas de senderismo a través de una actividad de NewRutaActivity.

La clase utiliza un MainViewModel para recuperar las rutas de senderismo y un RutasAdapter para mostrarlas en la lista. Al hacer clic en una ruta, se abre una nueva actividad RutaActivity que muestra detalles de la ruta seleccionada.

Además, utiliza un ActivityResultLauncher para iniciar la actividad NewRutaActivity y recuperar los datos de la nueva ruta creada. La clase NewRutaActivityResultContract es un contrato de resultado que especifica cómo se crea la actividad NewRutaActivity y cómo se manejan los resultados devueltos. La clase NewRutaActivityResult es una clase auxiliar que contiene el resultado de la actividad NewRutaActivity.



### 2. MainViewModel

La clase MainViewModel es un ViewModel de la arquitectura de componentes de Android que se utiliza para almacenar y administrar los datos relacionados con la lista de rutas de senderismo en la actividad MainActivity.

Esta clase extiende AndroidViewModel y recibe un objeto Application en el constructor, que se utiliza para inicializar un objeto Repository y obtener la lista de rutas de senderismo mediante el método getRutas().

El objeto Repository se obtiene a través de la clase RutasApplication, que es una subclase de Application que se utiliza para inicializar los componentes de la aplicación, como la base de datos y el objeto Repository.

En resumen, la clase MainViewModel se encarga de proporcionar los datos necesarios para mostrar la lista de rutas de senderismo en la actividad MainActivity, obteniendo dichos datos del objeto Repository.

Además, implementa un sistema de cambio de idioma mediante un ImageButton entre Español e Inglés, de manera que cuando estés viendo el texto en español te mostrará la bandera de España  en la MainView y cuando lo veas en inglés te mostrará la bandera de Inglaterra  (No la de UK).

Sin embargo los títulos de las actividades New Ruta y Ruta Item no se cambian debido a que si quiero traducir también los títulos en las demás vistas, tendría que quitar el label de AndroidManifest, y ponerlo en los layouts, pero la primera vez que entras, en

vez de ese título, sale la ruta (.es.upsa.papps...) y en los casos de Crear y Ver no merece la pena porque no hay un botón ni acción que actualice dichas vistas (siempre verías la ruta).

### 3. NewRutaActivity

Esta actividad es responsable de permitir al usuario crear una nueva ruta y guardarla en la base de datos de la aplicación. Contiene asimismo en la ActionBar para navegar hacia atrás(también incluido en RutaActivity).

### 4. NewRutaViewModel

### 5. Repository

La clase Repository maneja un conjunto de objetos Ruta. Crea un conjunto de rutas predefinidas en el constructor de la clase mediante el uso de un mapa de rutas, donde cada ruta se identifica por una clave única.

Proporciona métodos para recuperar una lista de todas las rutas, insertar nuevas rutas en el conjunto y buscar rutas por su identificador único.

El método getRutas() devuelve una lista de todas las rutas almacenadas en el conjunto, y se realiza mediante la creación de una lista vacía y añadiendo todos los valores del mapa de rutas en la lista.

El método insertRuta() inserta una nueva ruta en el conjunto de rutas. Crea un objeto Ruta con el nombre, explorador, valoración y dificultad proporcionados y asigna un identificador único generado aleatoriamente. Luego, se agrega el objeto de ruta al mapa de rutas y se devuelve el objeto de ruta.

El método findRutaById() encuentra una ruta en el conjunto de rutas según su identificador único. Simplemente busca la ruta en el mapa de rutas usando la clave proporcionada y devuelve el objeto de ruta correspondiente.

### 6. Ruta

### 7. RutaActivity

### 8. RutasAdapter

Define un adaptador personalizado para un RecyclerView que se utiliza para mostrar una lista de objetos de tipo Ruta. La interfaz de callback OnItemClickListener se utiliza para notificar a la actividad principal cuando se hace click en un elemento de la lista.

### 9. RutasApplication

### 10. RutasViewModel

Layouts:

### 1. activity\_main

Este es el layout de la actividad principal de la aplicación que incluye un RecyclerView y un FloatingActionButton. El RecyclerView se utiliza para mostrar una lista de rutas de senderismo y se especifica que los elementos de la lista serán renderizados utilizando el layout "ruta\_item". El FloatingActionButton se utiliza para agregar nuevas rutas a la lista. El layout en sí mismo es un ConstraintLayout que ocupa todo el espacio disponible en la pantalla. Además, contiene el ImageButton que maneja el idioma de la aplicación.

## 2. activity\_new\_ruta

Este layout se utiliza para una actividad en una aplicación de Android llamada NewRutaActivity. Está escrito en XML y utiliza el ConstraintLayout de Android. El diseño consta de varias vistas: dos EditText para ingresar el nombre de una ruta y el nombre del explorador, tres TextView para mostrar diferentes niveles de dificultad, un SeekBar para seleccionar la dificultad y un RatingBar para seleccionar una valoración. También hay un botón para guardar la información ingresada en los EditText. El diseño se ajusta automáticamente a diferentes tamaños de pantalla, ya que utiliza constraints para definir la posición y el tamaño de las vistas.

## 3. activity\_ruta

Este layout es una vista de una pantalla en Android que contiene cinco TextViews, cada uno con un identificador único (@+id/) y restricciones de diseño de ConstraintLayout (@+layout\_constraint...) que establecen la posición de los TextViews en la pantalla. Los TextViews muestran información de una ruta, incluyendo su ID, nombre, explorador, valoración y dificultad. Todos los TextViews tienen una anchura de 0dp y una altura que se ajusta al contenido (wrap\_content).

## 4. activity\_ruta2

## 5. ruta\_item

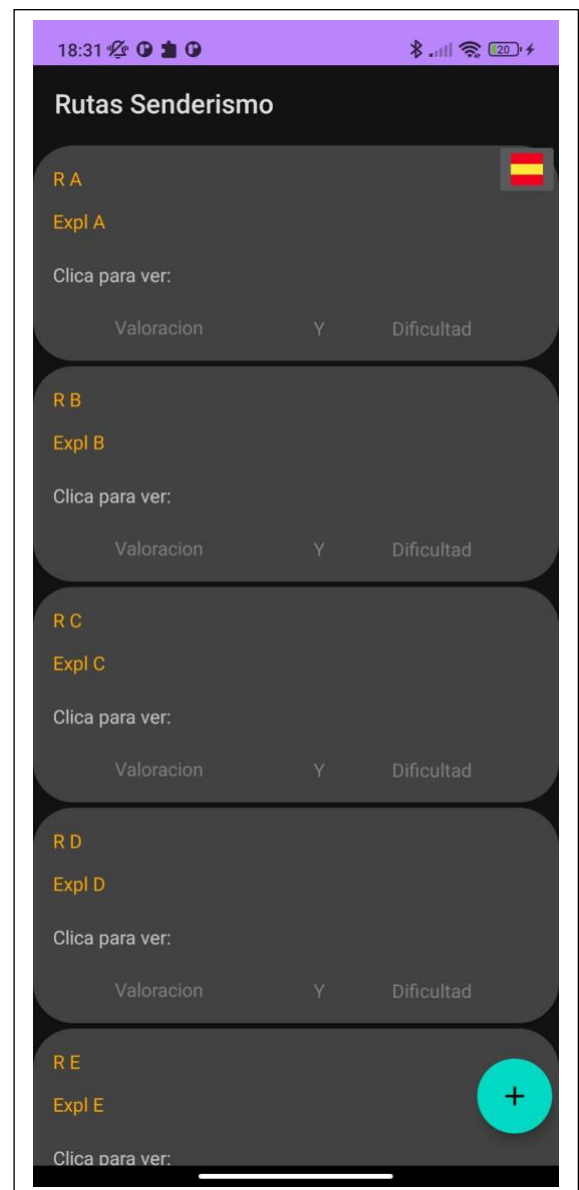
Este layout utiliza un CardView para mostrar información de una ruta en una lista. Dentro del CardView, se encuentra un ConstraintLayout que contiene cuatro TextViews que muestran el nombre de la ruta, el nombre del explorador, la dificultad de la ruta y su valoración. El CardView tiene un radio de borde de 30dp, un relleno de contenido de 15dp y un margen superior de 4dp, y se ajusta a la anchura del match\_parent y la altura del wrap\_content. Los TextViews tienen un ancho de 0dp y una altura de wrap\_content, y se posicionan usando restricciones en relación con otros elementos dentro del ConstraintLayout. Se utilizan atributos de herramientas para mostrar texto de ejemplo en el diseño.

## Ejecución del Programa:

Primero la app nos muestra las rutas que ya venían en la app por defecto de forma vertical, mostrando solo en nombre de la ruta y su explorador, en la parte inferior derecha tenemos un botón “+” con el que podemos añadir otra ruta, mientras que en la parte superior derecha tenemos un botón “Cambiar Idioma” con el que podemos añadir elegir entre Español e Inglés, si vemos la bandera de España veremos el texto en español, si vemos la bandera de Inglaterra, en inglés.

Además, si pulsamos encima de una ruta, nos mostrará en otra pagina la información de forma detallada, además de mostrar la valoración y dificultad.

Podemos ver tanto en modo claro como oscuro.

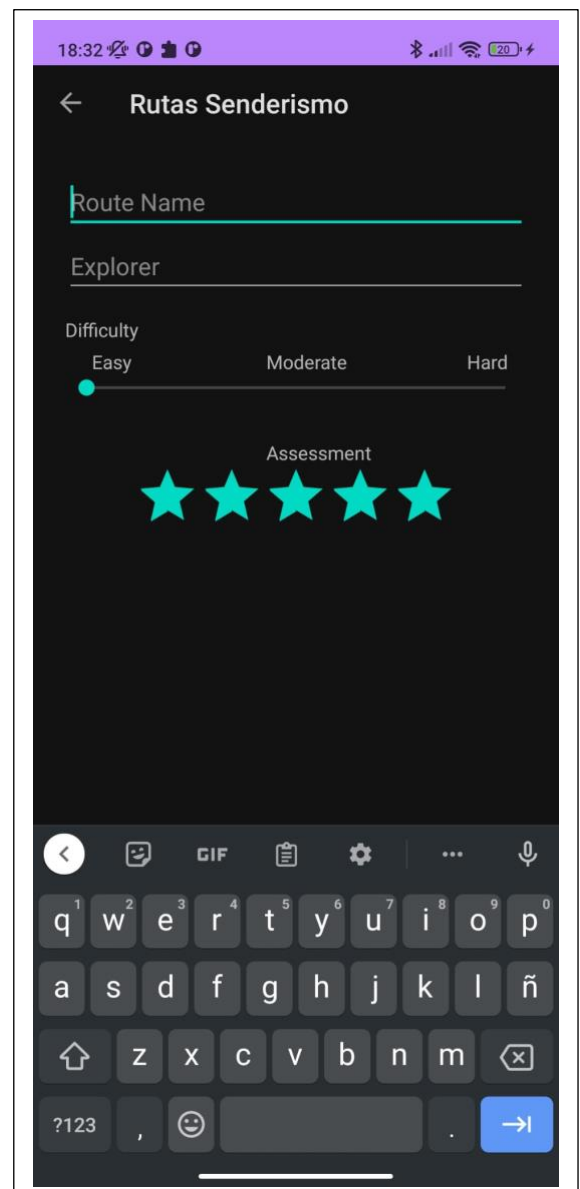
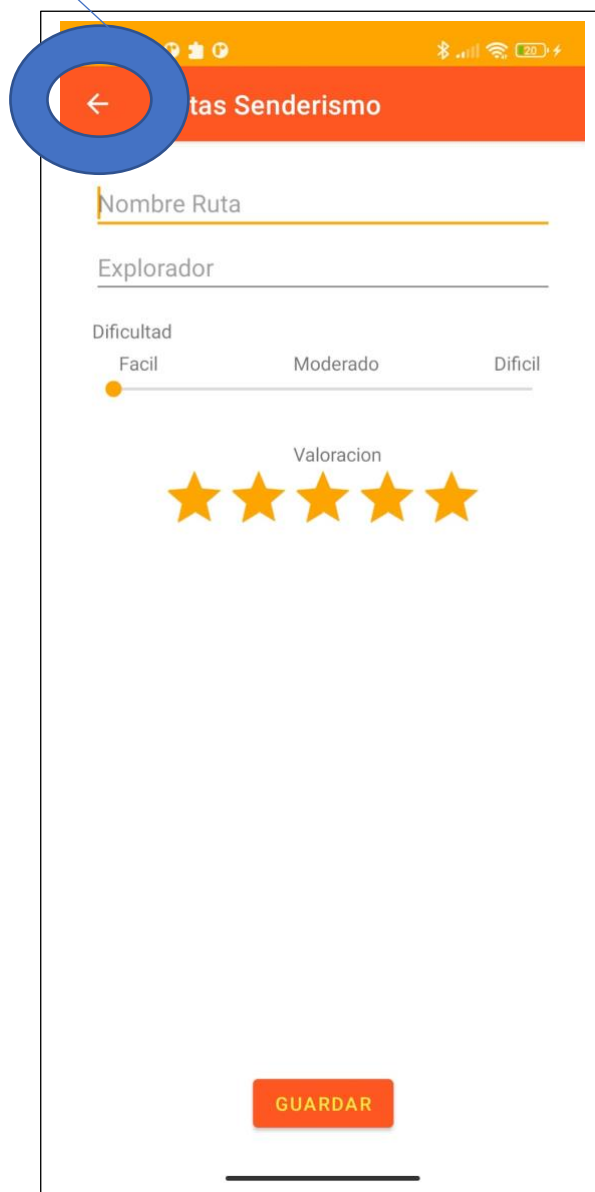


Si hemos optado por pulsar el “+” nos llevará a esta vista, donde nos pedirá el nombre de la ruta, el explorador, la dificultad (mediante una Seek Bar) y la valoración de la ruta (mediante un sistema de rating).

\*Añado un ejemplo para ver su ejecución

\*He dejado los temas (themes) que usamos en la app de películas, al menos para el modo noche

\*Rodeado está el sistema de navegación hacia atrás








En el modo claro esta en español.

En el modo noche en ingles, como ya dije anteriormente, los títulos que no pertenecen a la MainActivity no cambian de idioma, lo demás, si.

Añadimos un ejemplo para ver su ejecución

18:32

 20%

 Rutas Senderismo

Camino de Santiago

Ángel Camara

Dificultad

Facil

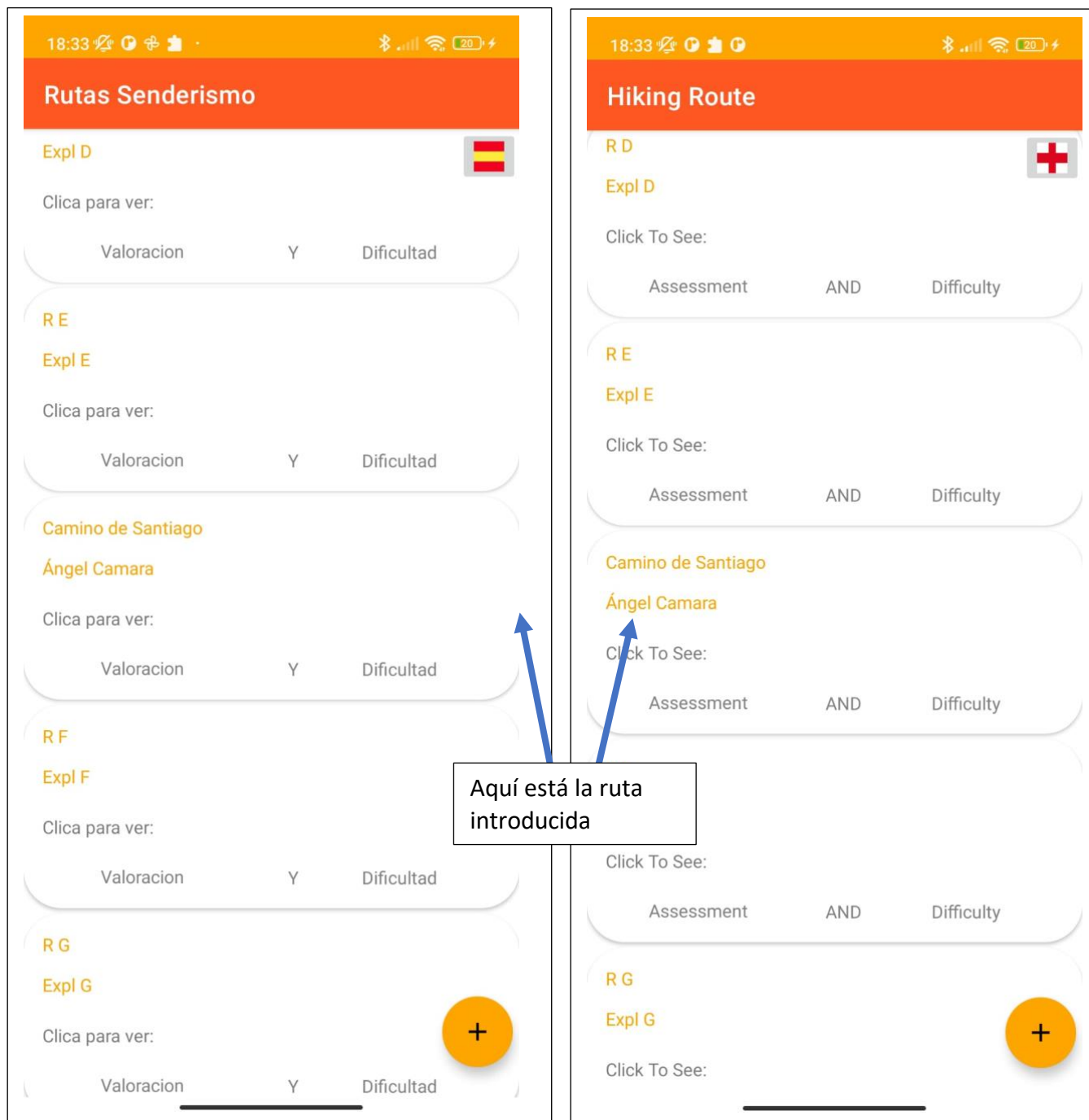
Moderado

Dificil

Valoracion

GUARDAR

Si pulsamos en el botón de “SAVE” guardaremos la ruta correspondiente  
Al volver al listado de rutas aparecerá lo siguiente

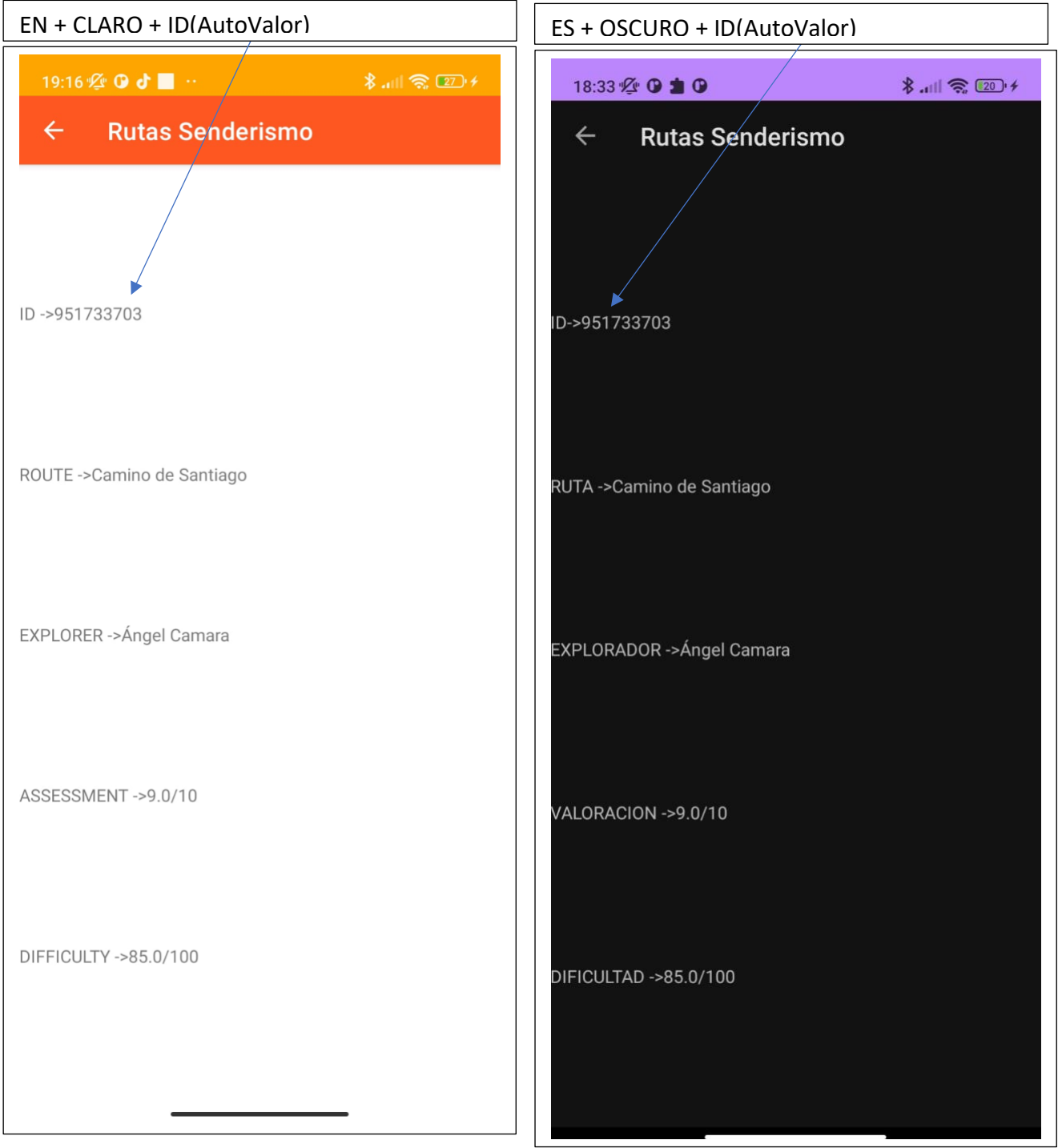


Podemos ver la ejecución en ambos idiomas.

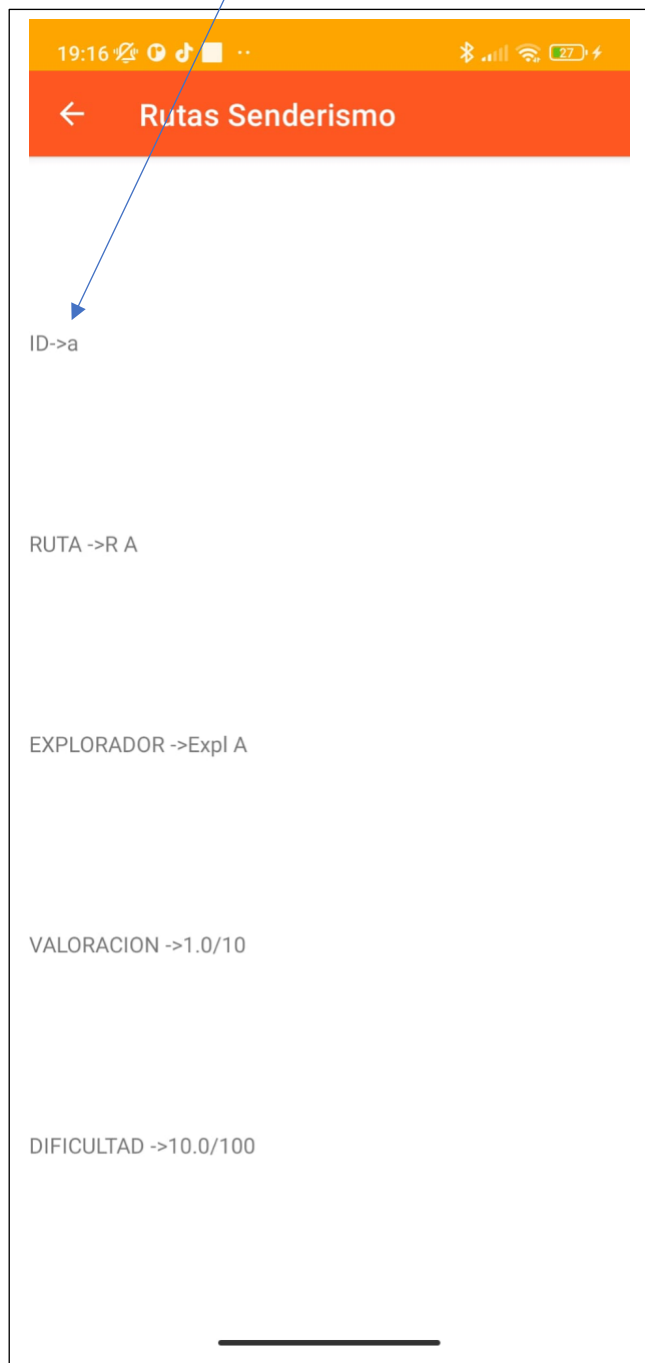


Si pulsamos sobre una de las rutas tendremos lo siguiente, en las rutas por defecto tendremos las claves manuales ('a', 'b'...) mientras que en las claves generadas automáticamente tendremos un auto valor.

\*(Subo capturas de ambos idiomas en ambos temas)



ES + CLARO + ID(Asignada en repositorio)



EN + OSCURO + ID(Asignada en repositorio)

