



EXTRA | Challenge

Flutter Position

En éste Challenge buscamos que nos muestres todo tu talento desarrollando apps!

Recordá la importancia de la legibilidad, las buenas prácticas, la calidad del código y la experiencia de usuario.

TECH STACK

- > Flutter
- > Dart
- > State Management (Optional)
 - BLoC
 - Provider
 - Riverpod
- > Hive/Shared Preferences
- > Git
- > Todas las librerías y sorpresas que quieras agregar!

EJERCICIO

El planeta tierra esta siendo invadido!
Te necesitamos para generar una herramienta de consulta y reporte, de los personajes de Star Wars en la Tierra.

El ejercicio consta de realizar una app en donde se pueda:

- > Ver una lista de personajes
- > Ver el detalle de un persona
- > Reportar un personaje
- > Menu para activar/desactivar conexión

La app deberá constar de las siguientes pantallas:

- > Inicio
 - Detalle
- > Menu

Inicio

Se requiere de un listado de personajes provistos por la API de Star Wars swapi.dev/api/

Dicho listado debe respetar las siguientes reglas

- > Visualización de forma paginada
- > Cada Item de personaje debera visualizar el nombre y el genero del mismo.
- > Al presionar sobre un personaje, navegar al Detalle. También deberá contar con botón de Atrás.

Opciones de paginación

- > Al llegar al final del listado obtener la siguiente página
- > Botones indicadores de Next y Previous con número de Página.

El servicio a utilizar para realizar esta pantalla de SWAPI.

Lista de personajes
GET <https://swapi.dev/api/people/>

Mas usos en la documentación: <https://swapi.dev/documentation>

Detalle

Se presentará el detalle del personaje seleccionado anteriormente.

Esta pantalla se compone de lo siguiente:

- > Datos adjuntos al modelo del personaje
 - Nacimiento
 - Color de ojos
 - Género
 - Color de pelo
 - Altura
 - Mundo natal
 - Peso
 - Nombre
 - Naves
 - Vehiculos

- Un botón para reportar el avistamiento de un personaje.
 - Dicho botón estará habilitado si la conexión del Menu se encuentra activa.
 - El reporte se realizará al siguiente endpoint con el formato adecuado
 - Una vez enviado notificar en pantalla cual fue el estado del request. (Snackback, Material-Banner, etc).

url: <https://jsonplaceholder.typicode.com/posts>
httpMethod: POST

body:

- userId: int, character id
- dateTime: String, character report DateTime
- character_name: String, character name

example:

```
{
  "userId": 1,
  "dateTime": "2022-05-12 11:00:32.334679",
  "character_name": "Luke Skywalker"
}
```

GET <https://swapi.dev/api/people/>

Mas usos en la documentación: <https://swapi.dev/documentation>

Menu

La app debe tener un menú con lo siguiente:

- Switch para activar/desactivar la “conexión” de la aplicación (dicha conexión es independiente de datos / wifi del telefono)

CONSIDERACIONES

El estilo que le quiera dar a la App es libre mientras cumpla con los puntos mencionados.

Se valorarán buenas prácticas aplicadas a: Disposición / arquitectura de la App. Manejo y Validaciones de errores, Clean code, Principios SOLID.

ENTREGA

Una vez finalizado el desarrollo de la aplicación, subirla a un repositorio de Github privado que se pueda compartir con los siguientes usuarios @NataliaSuarez @glofiego10.