

# Flutter Test

Objetivo: Crear una aplicación simple para evaluar las habilidades de desarrollo.

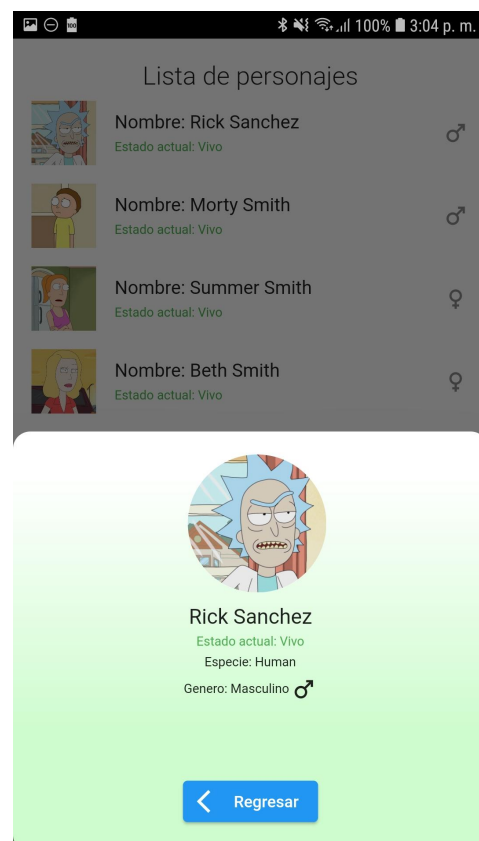
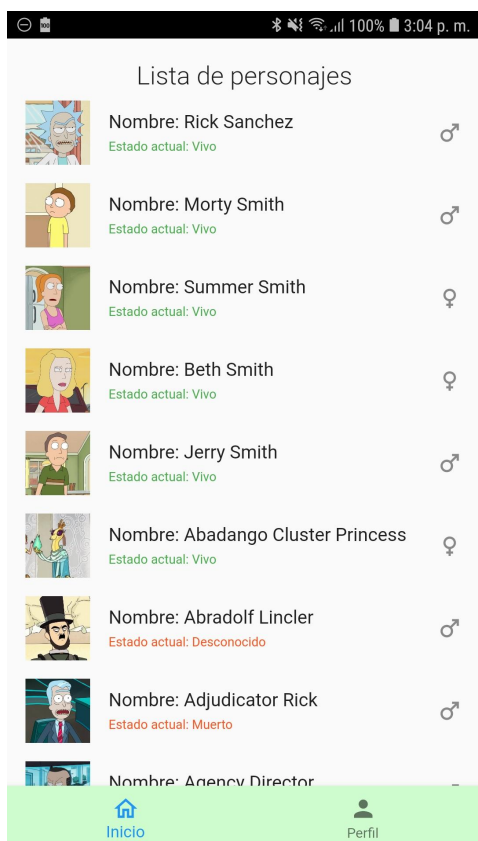
## Pantalla 1

Puntos a evaluar: Replicación de interfaz y consumo de Api Rest.

**Requerimiento:** Desarrolla la interfaz de usuario lo más parecido a las siguientes imágenes, consultando el siguiente endpoint : <https://rickandmortyapi.com/api>

Características:

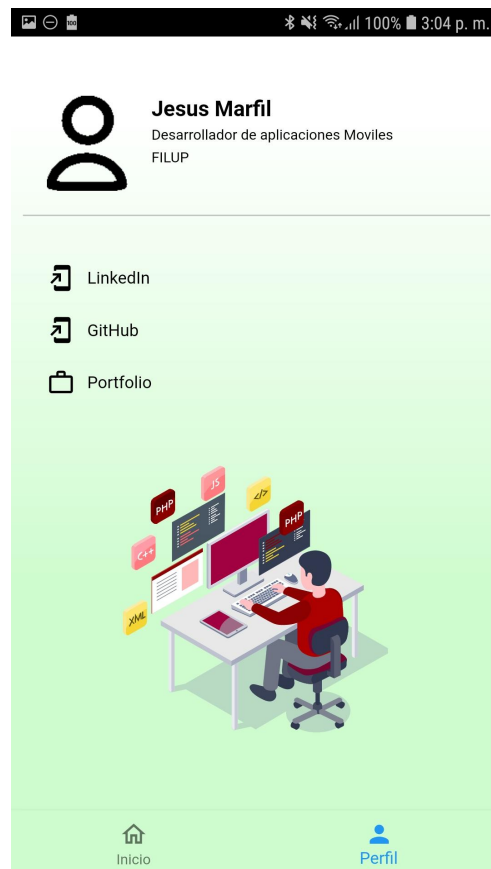
- Listado de personajes
- Cada personaje debe mostrar: imagen, nombre, estado actual (vivo, muerto, desconocido) y un icono que represente el género.
- Al presionar sobre un personaje, se debe mostrar un **ModalBottomSheet** con la información del personal, esta información la debe de consultar primero con el servidor.



## Pantalla 2

Utiliza el diseño que más te guste para crear una página de perfil, el único requisito es mostrar la siguiente Información básica:

- Tu nombre
- Puesto al que aplicas
- Acceso a tu LinkedIn (Opcional)
- Acceso a tu Github (Opcional)
- Página personal (Opcional)



Notas:

- Utiliza un **BottomNavigationBar** para el manejo de pantallas
- Indispensable el uso de gestor de estados, puedes utilizar el que mejor conozcas.
- Puedes guiarte del siguiente video:

<https://drive.google.com/file/d/1aAM5Dp9uzfjVQmjgD2u2u0av3I27TMcD/view?usp=sharing>

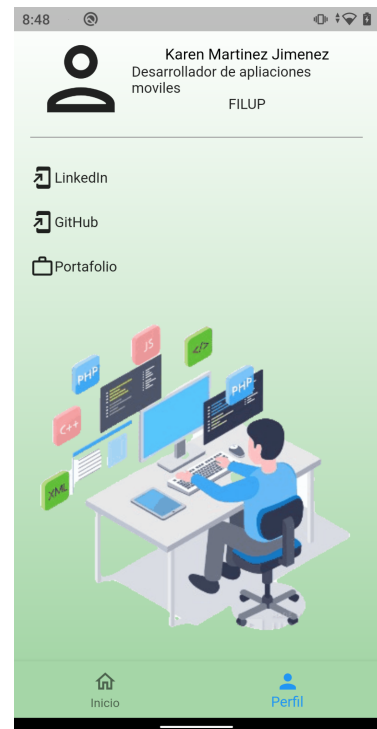
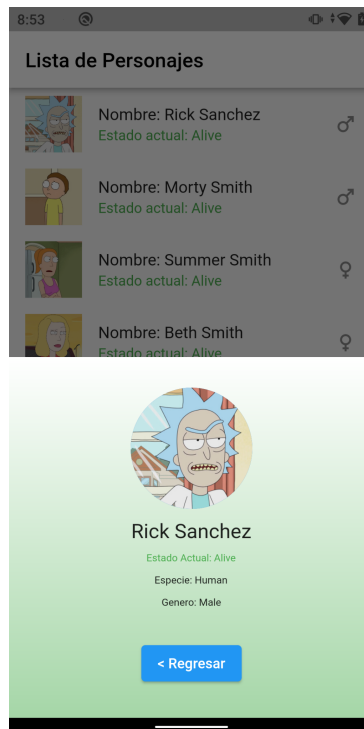
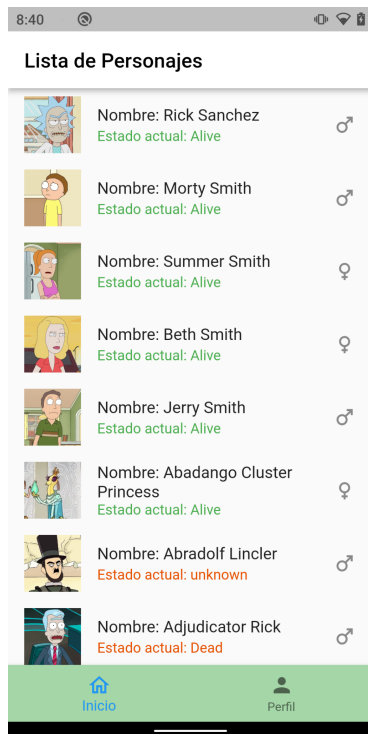
Una vez finalizado el desarrollo, sube el proyecto a un repositorio a GitHub y agrégame para la revisión de código. Mi usuario de GitHub es: **JesusMarlor**

**Documentación:** <https://rickandmortyapi.com/documentation/#rest>

## Resultado de interfaz:

- 1.- La elevación del AppBar es diferente al del diseño
- 2.- El título de la primera pantalla debe ser centrado y con el estilo lo más cercano al siguiente: `TextStyle( fontSize: 22, fontWeight: FontWeight.w300)`.
- 3.- Falto la traducción del status de cada personaje (Vivo, Muerto o Desconocido).
- 4.- El icono que representa el género debió cambiar entre 4 opciones (female, male, genderless or unknown).
- 5.- En la información individual del **ModalBottomSheet**, solo faltaron los bordes redondeados, el icono del género, la traducción del género y estado actual.
- 6.- El botón del **ModalBottomSheet** debió ser un **ElevatedButton.icon** con icono **Icons.arrow\_back\_ios** en lugar de un **ElevatedButton** con el texto "< regresar".
- 7.- La página de perfil cumple con la información solicitada, solo falto darle un diseño propio.

En general la interfaz quedó muy parecida a la solicitada.



### Resultado de funcionalidad:

- 1.- Buen manejo del gestor de estados.
- 2.- Pese a no ser un requisito explícito, hubiera sido una buena forma de demostrar el manejo de peticiones http, paginando el resultado (Como se muestra en el [video](#))
- 3.- Al mostrar el **ModalBottomSheet** se produce el siguiente error **Incorrect use of ParentDataWidget.** debido al uso del widget Expanded
- 4.- La información que se muestra en el **ModalBottomSheet** se obtiene desde el listado principal y no desde el endpoint individual del carácter (<https://rickandmortyapi.com/api/character/1>).
- 5.-La pantalla de perfil si cumple con la redirección a la página web de cada elemento ( GitHub, LinkedIn).
- 6.- Buena decisión al utilizar wantKeepAlive para mantener el estado de la interfaz ya construida.
- 7.-Archivos bien ordenados, código limpio y fácil de entender.

El proyecto realizado en kotlin es un excelente agregado para poder demostrar que también puede programar en nativo.

**El resultado final fue Sobresaliente.**