

INFORME PROYECTO FINAL IPCOM2 (UNGS)

Introducción

Este trabajo práctico consiste en desarrollar una página web que permite buscar y ver imágenes de los personajes de la serie **Rick & Morty**, utilizando una fuente oficial de internet que nos proporciona la información de los personajes. En la página, cada personaje aparece en una tarjeta que muestra su foto, si está vivo o no, su última ubicación y el primer episodio en el que apareció. Además, hemos incluido una barra de búsqueda para encontrar personajes específicos.

La arquitectura del proyecto está dividida en varias capas, cada una con una responsabilidad específica: **Persistence**, **Services**, **Transport** y **Utilities**. Este diseño asegura que cada componente tenga una función clara y facilita el mantenimiento y la expansión del código.

Modificaciones e Implementaciones Realizadas

Capa de Vista/Presentación (views.py):

Se llama a la función `getAllImages` definida en `services.py` para obtener todas las imágenes, las cuales ya vienen en forma de tarjeta. Estas imágenes se renderizan en la página para que el usuario las vea. Además, la misma función se utiliza en la funcionalidad de búsqueda con la ayuda del `home.html`, para obtener las imágenes que coinciden con el término de búsqueda del usuario.

Capa de Servicio/Lógica de Negocio (services.py):

La función `getAllImages` obtiene los datos crudos desde la API, los convierte en objetos en forma de tarjeta y devuelve una lista de imágenes. Para ello usa `fetchDataFromApi` desde `transport.py` para obtener los datos crudos y convierte cada dato crudo en una tarjeta utilizando la función `fromRequestIntoCard` desde `translator.py`.

Modificaciones en el Template (`home.html`):

Intentamos mostrar el estado de cada personaje de Rick & Morty (si está vivo, muerto o desconocido). Para lo cual implementamos un nuevo parámetro más descriptivo dentro del `home.html` llamado "img status" que compara los estados de los personajes:

- Si el personaje está vivo, el estado será alive (vivo).

- Si el personaje está muerto, el estado será dead (muerto).
- Si no se conoce el estado del personaje, se marcará como unknown (desconocido).

Este cambio nos permite mostrar esta información de manera clara y específica.

Una vez que tenemos los estados de los personajes, queremos que las tarjetas (cards) que contienen la información de los personajes tengan bordes de colores que correspondan a estos estados. Usamos Bootstrap, una herramienta que facilita la creación de diseños web elegantes, para hacer esto.

- Verde para vivo: Usamos la clase “border border-success” de Bootstrap para poner un borde verde alrededor de la tarjeta si el personaje está vivo.
- Rojo para muerto: aplicamos “border border-danger” de Bootstrap para poner un borde rojo alrededor de la tarjeta si el personaje está muerto.
- Naranja para desconocido: la clase “border border-warning” de Bootstrap para poner un borde naranja alrededor de la tarjeta si el estado del personaje es desconocido.

Dificultades

Se intentó desarrollar el código de autenticación del login pero no pudimos concretarlo debido a un desentendimiento o confusión de los apartados de la guía de ayuda del trabajo práctico.