

TRABAJO PRÁCTICO INTEGRADOR

INTRODUCCIÓN A LA PROGRAMACIÓN - SEGUNDO CUATRIMESTRE 2024

DOCENTE: Fernando

COMISIÓN: 05

ALUMNO: BRENDA TAMARA DIAZ

DNI: 42.689.978

GMAIL: brendiaz0202@gmail.com

Introducción:

Para la realización de este trabajo, se tuvo que seguir una serie de pasos, con el objetivo de profundizar los temas abordados en clase, se nos encomendó el desafío de desarrollar una aplicación web utilizando Django. La tarea consiste en implementar una galería de imágenes de los personajes de la serie "Rick & Morty" empleando su API oficial.

La aplicación está diseñada para buscar y mostrar imágenes de los personajes de "Rick & Morty", presentándolos en tarjetas que incluyen detalles como la imagen del personaje, su estado, su última ubicación conocida y el episodio en el que apareció por primera vez. Además, de un buscador central para facilitar la exploración de personajes. Este enfoque nos ayudó a entender la importancia de dividir el proyecto en capas: Persistencia, Servicios, Transporte y Utilidades, cada una con responsabilidades claras y definidas.

Pasos que tuvimos que seguir:

1. Instalación de Herramientas:

- **Visual Studio Code:** Editor de código recomendado para el desarrollo del proyecto.
- **Extensiones para VS Code:**
 - After Dark
 - Prettier - Code formatter
 - Pylance
 - Python
 - Python Debugger

2. Instalación de Python:

3. Configuración de GitHub:

4. Instalación de Django y Dependencias:

5. Ejecución del Servidor Django:

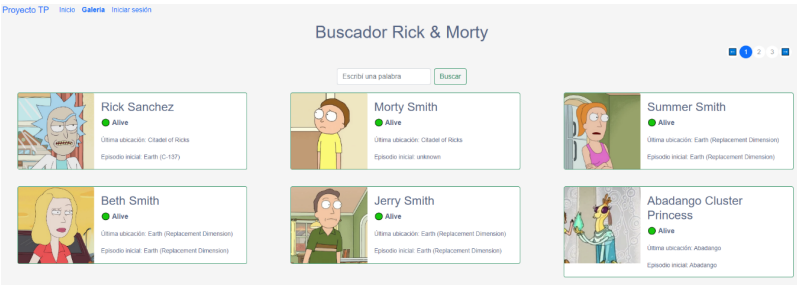
Proyecto:

sigue una serie de multicapas, cada una con una única responsabilidad:

- 1. Persistence:** Manejo de la base de datos para el alta/baja/modificación de objetos.
- 2. Services:** Lógica de negocio de la aplicación.
- 3. Transport:** Consumo de los datos de la API.
- 4. Utilities:** Almacena traductores y otros elementos propios de la aplicación usados en los templates.

Encargadas de hacer que dichas imágenes de la galería se muestren:

- 1. views.py:** La función `home` obtiene las imágenes de los personajes y los favoritos del usuario (si está autenticado). Luego, muestra `home.html` con estos datos.
- 2. services.py:** La función `getAllImages` obtiene una lista de personajes de la API de Rick & Morty. Si hay un `input`, filtra los personajes por nombre.
- 3. home.html:** Este código en `home.html` crea tarjetas para cada personaje, cambiando el color del borde según el estado del personaje (vivo, muerto o desconocido).



Pros y contras de los ejercicios:

En cada uno de estos ejercicios fueron surgiendo inconvenientes /dificultades, por ejemplo dentro de **home.html** , fue todo un desafío llegar a cambiar el color del borde según el estado del personaje.

Dentro de **VIEWS.PY** , fue un desafío obtener los favoritos solo si el usuario está autenticado. Por lo cual, decidimos manejar esta lógica en la capa de servicios para mantener las vistas más limpias.

Pero por otro lado, también hubieron cosas beneficiarias como, la utilización de Json dentro de services.py, lo cual,fue de gran ayuda para que **sea más manejable para el programa, es decir, que pueda ser más manipulado** .

Conclusión:

Aunque este ejercicio resulto ser un gran desafío, sin duda nos ha permitido conocer y afianzar habilidades de suma importancia para el programador. A base de prueba y error se logró comprender la importancia de dedicar un tiempo a la planeación, estudiar y analizar qué cosas son necesarias y en base a esto encarar cada ejercicio de la mejor manera posible.

En retrospectiva, aprendizaje que será indispensable y facilitar nuestro trabajo en el futuro, ya que enriquece la experiencia de los estudiantes al combinar el uso de API, la autenticación de usuarios y la lógica entorno a Django.