**ENTREGA 4: Desarrollo Servidor completo con Django**

En este ejercicio he continuado con el proyecto de la tienda de videojuegos creado en la entrega anterior, añadiendo los nuevos conceptos que se requerían para esta entrega. A continuación, explicaré aquellas partes de código que se han añadido y que diferencian este ejercicio con el ejercicio anterior:

**videojuego/admin.py**

Texto

Descripción generada automáticamente

En el modelo Videojuego, en el archivo admin.py registro Videojuego en nuestro administrador para poder observarlo desde la administración de Django.

**Patrón de fondo

Descripción generada automáticamente**

**tienda\_videojuegos/settings.py**

Texto

Descripción generada automáticamente

En el archivo settings.py añado la base de datos a la que me quiero conectar, en este caso tienda\_videojuegos en MySQL.

**videojuego/urls.py**

Texto

Descripción generada automáticamente

Para el modelo Videojuego se añade el ViewSet a una lista de urls en las que ya estaban las vistas genéricas y la api\_view.

**videojuego/views.py**

**Texto

Descripción generada automáticamente**

Para el modelo VIDEOJUEGO añado las clases de autenticación y de permisos a las vistas genéricas y creo una ViewSet a la que también añado dichas clases y añado una restricción para no poder hacer Update a los videojuegos.

**tienda\_videojuegos/urls.py**

Texto

Descripción generada automáticamente

Añado en el servidor una nueva url para obtener un token de autenticación.

**tienda\_videojuegos/clients/test.py**

**Texto

Descripción generada automáticamente**

En el test de la carpeta Clients he añadido comprobación de usuario y contraseña para poder acceder a nuestros endpoints. Solo aquellos usuarios que estén registrados en nuestra administración de Django y tengan los permisos necesarios podrán acceder correctamente.

**videojuegos/tests.py**

**Captura de pantalla de computadora

Descripción generada automáticamente con confianza media**

Para esta entrega he creado 2 test para reflejar el buen funcionamiento de la API. El primero comprueba que el videojuego indicado se crea correctamente y el segundo comprueba si funciona correctamente la restricción de Update que tenemos creada en nuestro ViewSet.

**Django administration**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente** **Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente** Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En la administración de Django se puede observar como he creado dos grupos de usuarios (administradores y empleados) con distintos permisos y dos usuarios (administrator y empleado). También se puede observar como ha quedado registrado el modelo Videojuego y los tokens demandados por parte de los usuarios.

**Observaciones**

Ha sido una práctica importante ya que ha sido un conjunto de todos los conocimientos que hemos aprendido durante este módulo 6 además de añadirle los nuevos conceptos aprendidos como el ViewSet, la gestión de usuarios y el testing. Con esto ya hemos aprendido suficientes conocimientos para poder crear la parte de backend de nuestro proyecto grupal y nuestro TFM, hecho muy importante ya que son los próximos retos que tenemos en estos meses que quedan de máster.

**Conclusiones**

En general en esta práctica no he tenido grandes problemas que me hayan impedido realizarla, pero sí que tuve un pequeño problema ya que las clases de autorización y las clases de permisos no me dejaba tenerlas en el archivo settings.py del servidor por lo que tuve que ponerlas una a una en los genéricos y ViewSet. Esto no supone un gran problema ya que sigue siendo un camino válido, pero sí que perdí bastante tiempo intentando buscar solución a ello. Por lo demás, todos los conceptos los he podido seguir y entender correctamente de las clases y de los vídeos del módulo.

**Url del repositorio**: https://github.com/bielrc1906/Modulo\_6.git

Biel Roca Coll