

**Introducción a la programación**

**Trabajo práctico grupal**

**Estudiantes**: Sorroche Agustina y Greco Lucas

**Comisión**: 03

**Docentes:** Daniel Bressky, Esteban Fassio y Santiago Velázquez

**Año**: 2024

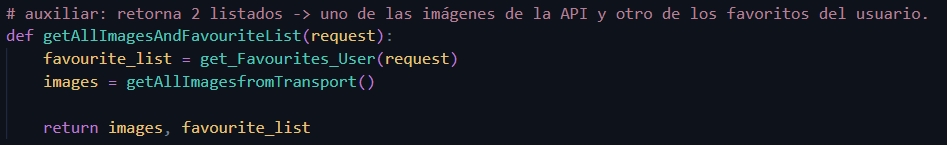
**Carrera**: Lic. Sistemas

**Nota:**

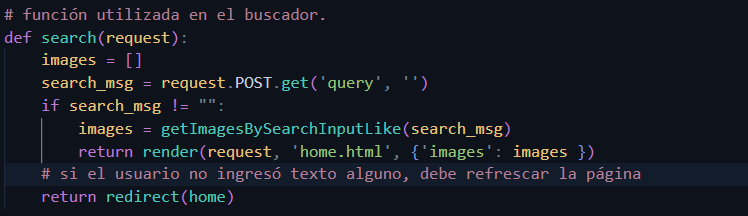
**Introducción**

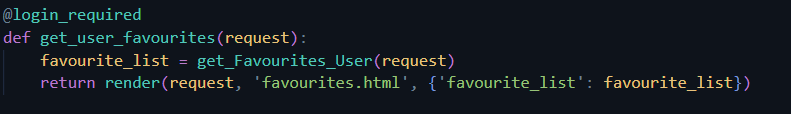
El presente informe detalla el desarrollo de un proyecto práctico. El proyecto se centra en la creación de una galería de imágenes utilizando el framework Django, diseñada para mostrar imágenes provenientes de la API de la NASA y hacer un login de usuario. Este informe proporciona una visión general de las funcionalidades implementadas, las dificultades que se tuvieron durante el proceso de desarrollo.

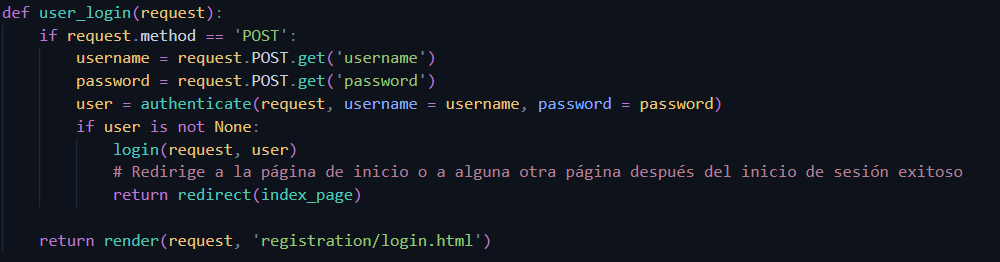
**Views.py**

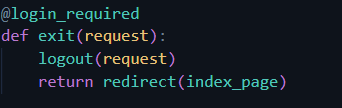
****La función getAllImagesAndFavouriteList(request) es una función auxiliar que obtiene una lista de imágenes de la API y la lista de los favoritos del usuario y estos datos se devuelven en una tupla para luego ser utilizada en la función home(request).

**** La función home(request) actúa como el controlador principal para la página de inicio de la galería. Utiliza una función auxiliar para obtenerlistas de imágenes y favoritos, luegorenderiza el contenido de 'home.html' pasando estos datos como contexto. Esto permite que 'home.html' muestre las imágenes recuperadas y los favoritos del usuario

****la función search(request) obtiene un mensaje de búsqueda ingresado por el usuario, si encuentra un mensaje de búsqueda valido, realiza la búsqueda de imágenes correspondientes para luego renderizar home.html con los resultados de la misma. Si no encuentra ningún mensaje de búsqueda refresca la página.

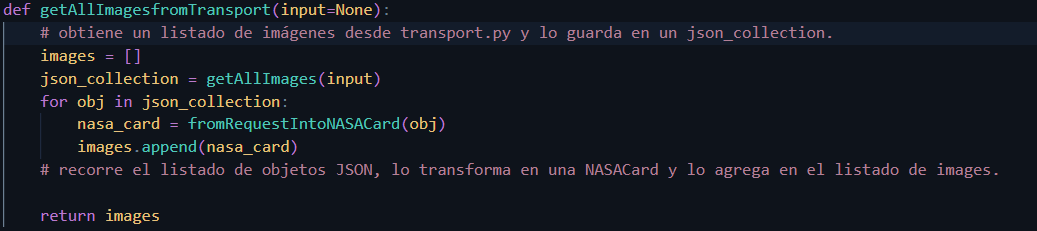
****La función get\_user\_favourites(request) obtiene la lista de favoritos del usuario llamando a la función que se encuentra services\_nasa\_image\_gallery.py y renderiza el contenido de favourites.html con el contenido de la lista de favoritos.

****La función user\_login(request) obtiene los datos del usuario y contraseña del formulario de inicio de sesión, luego verifica las credenciales del usuario, si las credenciales son correctas y el usuario existe en la base de datos la función inicia sesión y redirige al usuario a la pagina de inicio, si no existe el usuario refresca la pagina para que el usuario vuelva a intentar

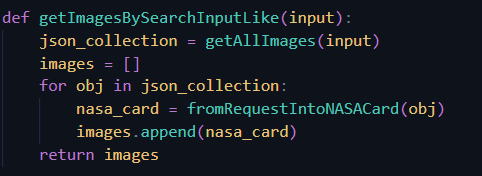
****La función exit(request) se encarga de cerrar la sesión del usuario autenticado, para eso llama a la función logout(request) para eliminar la sesión actual del usuario y luego lo redirige a la pagina de inicio

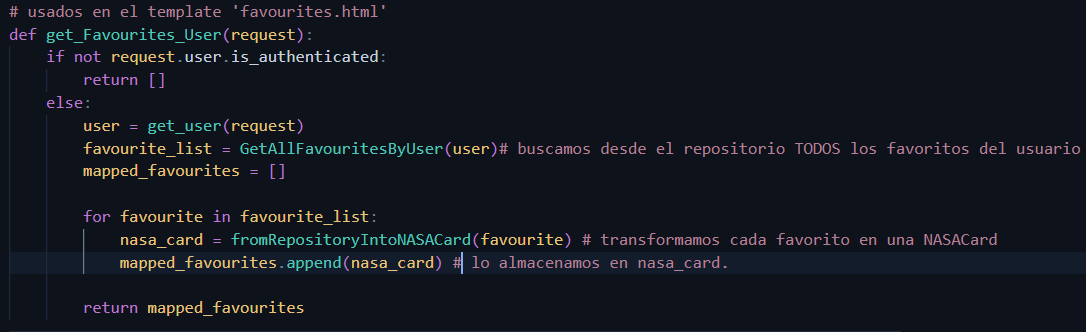
**Services\_nasa\_image\_gallery.py**

La función getALLImagesfromTransport(input=None) obtiene las imágenes de la API y los guarda en json\_collection, luego se recorre cada imagen para transformarla en una NASA Card y lo agrega al listado de images. La función devuelve el listado de imágenes transformadas en NASA Card

****

La función getImagesBySearchInputLike(input) es la función que utiliza el buscador para conseguir las imágenes a partir del input del usuario y luego transformarlas en NASA Card para después mostrarlas



La función get\_Favourites\_User(request) verifica si el usuario esta autenticado, si no esta autenticado devuelve una lista de favoritos vacía, si el usuario si esta identificado obtiene todos los favoritos del mismo, luego convierte cada favorito en una NASA Card para luego devolver una lista de NASA Cards que representan los favoritos del usuario y que se muestren cuando se ejecuta la función de favoritos en el archivo views.

Luego de estas funciones se hicieron pequeños ajustes a los archivos urls.py y header.html, para poder desarrollar el inicio de sesión, ya que en urls.py se llamaba a la función index(request) en lugar de llamar a la función user\_login(request) y en el header.html los botones de inicio de sesión y salir no redirigían a ningún lado

En el presente trabajo se desarrollaron los condicionales “barra de busqueda” y “inicio de sesión”, se desarrollo una parte del condicional de favoritos, pero no se pudo llevar a cabo, se pueden mostrar los favoritos del usuario, pero no se pueden agregar o borrar.