

**INFORME DE TRABAJO PRÁCTICO DE INTRODUCCIÓN A LA
PROGRAMACIÓN: GALERÍA DE IMÁGENES DE LA NASA**

Universidad Nacional de General Sarmiento
Introducción a la programación
Primer Semestre 2024

Giagnorio Maximiliano
Polcaro Bianca
Maidana Zoe

Introducción

Como proyecto de la materia Introducción a la Programación en la UNGS los alumnos deberán realizar una galería de imágenes de la NASA, la cual es una página web que cuenta con una pantalla de inicio, un apartado de inicio de sesión, una galería en la cual se encuentran imágenes tomadas por la NASA, traídas a la app mediante el uso de su API. En la misma se pueden buscar imágenes de cualquier tema relacionado a la galaxia, como: el sol, luna, estrellas, etc, y también posee sección de favoritos donde el usuario registrado puede almacenar las imágenes que haya guardado. La página cuenta con un diseño interactivo para el usuario, para que el mismo pueda utilizar las funciones de la app de una manera sencilla.

El propósito de este informe es el de explicar las decisiones tomadas para el trabajo práctico, en donde se solicita la implementación de ciertas funciones y a modo opcional algunos atributos casi definidos y otros que requieren desarrollo desde cero.

Desarrollo

A modo de inicio, se comenzó por realizar lo pedido de forma obligatorio para la aprobación del trabajo, filtrar las imágenes a la hora de buscar algo en la galería, que aparezcan las predeterminadas sólo yendo a la galería, y hacer funcionar la sección de “favoritos”.

Comenzamos con la función GetAllImages la cual contiene todas las imágenes a mostrar, recorriendo json_collection, transformando cada objeto JSON en un objeto NASACard utilizando mapper.fromRequestIntoNASACard y añadiendo el objeto NASACard a la lista images.

```
def GetAllImages(input=None):
    # obtiene un listado de imágenes desde transport.py y lo guarda en un json_collection.
    # ¡OJO! el parámetro 'input' indica si se debe buscar por un valor introducido en el buscador.
    json_collection = transport.getAllImages(input)

    images = []

    # recorre el listado de objetos JSON, lo transforma en una NASACard y lo agrega en el listado de images. Ayuda: ver mapper.py.
    for json_object in json_collection:
        nasa_card = mapper.fromRequestIntoNASACard(json_object)
        images.append(nasa_card)

    return images
```

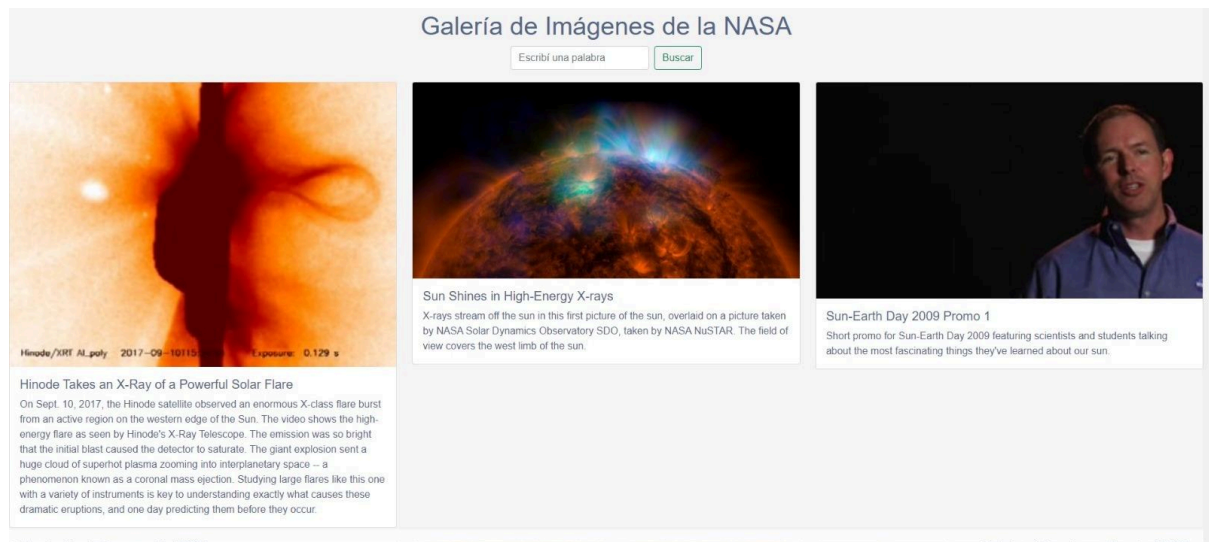
Luego, a la variable con la lista vacía json_collection se le asignó la función GetAllImages proveniente del archivo transport.py. Se recorre cada objeto en json_collection.

Para cada objeto JSON, llama a la función mapper.fromRequestIntoNASACard(data) (del archivo mapper.py), la cual transforma el objeto JSON en una instancia de NASACard.

Agrega cada instancia de NASACard a la lista images.

Para lograr definitivamente el renderizado se tiene que ir al apartado views.py y asignar en la función home a la variable images la función GetAllImages que se encuentra en el archivo services_nasa_images_gallery.py

Posteriormente de lograr los puntos obligatorios, se siguió con la primera opción semi desarrollada opcional, la barra de búsqueda. Cuya función es que se pueda buscar las imágenes pedidas por el usuario.



Para redireccionar se encontró que desde `django.shortcuts` se importaba `render` y `redirect`, pero esta última aparece sin uso. En un principio se intentó con `home.html` como parámetro pero arrojaba error. Se intentó sólo con `home` y funcionaba.

Para filtrar las imágenes de búsqueda se realizó una lista vacía para que se acumulen las imágenes de la api con el query ya implementado en la variable `search_msg`.

Y para renderizar se realizó igual que como estaba hecha en la función `home`. Sólo se copió y se cambiaron los datos necesarios.

También se implementó la función de Login. Para esto se tomó como referencia el tutorial brindado en el repositorio del trabajo práctico, tomando en cuenta las funciones ya definidas, se agregaron las acciones de login y logout en `header.html` y `login.html`:

```
<li class="nav-item">
  {% if request.user.is_authenticated %}
  <a class="nav-link" href="{% url 'logout' %}">Salir</a> {% else %}
  <a class="nav-link" href="{% url 'login' %}">Iniciar sesión</a> {% endif %}
</li>
```

```
<form action="{% url 'login' %}" method="POST" style="display: inline-block;">
```

[Proyecto TP](#) [Inicio](#) [Galería](#) [Favoritos](#) [Salir](#)

Bienvenido ADMIN!

Mirá las imágenes desde [este link](#).

Una vez implementado el login, se continuó con la función de añadir y borrar favoritos, los cuales se pueden ver en su apartado correspondiente. Para ello, también se utilizaron

funciones ya definidas en mapper.py, llamandolas en las funciones correspondientes para crear las variables en services_nasa_image_gallery.py:

```
# añadir favoritos (usado desde el template 'home.html')
def saveFavourite(request):
    fav = mapper.fromTemplateIntoNASACard(request) # transformamos un request del template en una NASACard.
    fav.user = get_user(request) # Le seteamos el usuario correspondiente.

    return repositories.saveFavourite(fav) # Lo guardamos en la base.
```

```
# usados en el template 'favourites.html'
def getAllFavouritesByUser(request):
    if not request.user.is_authenticated:
        return []
    else:
        user = get_user(request)

        favourite_list = repositories.getAllFavouritesByUser(user) # buscamos desde el repositorio TODOS los favoritos
        mapped_favourites = []

        for favourite in favourite_list:
            nasa_card = mapper.fromRepositoryIntoNASACard(favourite) # transformamos cada favorito en una NASACard
            mapped_favourites.append(nasa_card)

        return mapped_favourites
```

Luego en views.py se crearon las funciones getAllFavouritesByUser, saveFavourite y deleteFavourite, para ver, añadir y borrar favoritos. Se utilizaron métodos post y las funciones modificadas anteriormente, se les añadió la función de messages de django mediante su importación, para que más adelante, modificando los archivos html, el usuario pueda ver un mensaje informativo de si la imagen se guardó o se eliminó de favoritos correctamente.

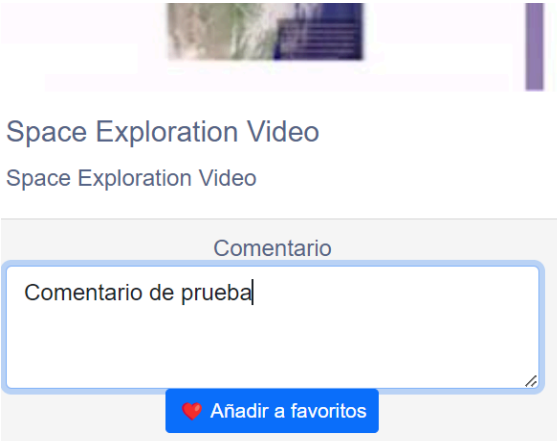
```
@login_required
def getAllFavouritesByUser(request):
    favourite_list = services_nasa_image_gallery.getAllFavouritesByUser(request)
    return render(request, 'favourites.html', {'favourite_list': favourite_list})

@login_required
def saveFavourite(request):
    if request.method == 'POST':
        fav = services_nasa_image_gallery.saveFavourite(request)
        if fav:
            messages.success(request, '¡La imagen se ha guardado en favoritos correctamente!')
            return redirect('home')
        else:
            messages.info(request, '¡La imagen ya está guardada en favoritos!')
            return redirect('home')
```

```
@login_required
def deleteFavourite(request):
    if request.method == 'POST':
        success = services_nasa_image_gallery.deleteFavourite(request)
        if success:
            messages.success(request, '¡Se ha eliminado correctamente!')
            return redirect('favoritos')
        else:
            return redirect('favoritos')
```

Después, se creó un loading spinner o spinner de carga, una de las primeras funciones opcionales para realizar en la página. Para que mientras las imágenes están cargando en la página, el usuario entienda que todavía no están todas las solicitadas para que no se muestre la página estática, sino sólo el spinner hasta que está acción sea completada (traer todas las imágenes). Se aplicó simplemente creando con CSS el círculo animado en el documento de home.html y utilizando una lógica con script para que sólo aparezca antes que las imágenes aparezcan tanto en la galería, como luego de realizarse una búsqueda.



Como siguiente función opcional, se creó un apartado donde se puede agregar comentarios a una imagen antes de guardarla en favoritos y luego visualizarlo en el apartado de favoritos, para ello se modificó el modelo de favorito ya definido y la clase nasa_card, agregando un apartado de comentario. Se realizaron cambios agregando la nueva sección de comentario en todas las funciones que gestionan guardar un favorito en los archivos mapper.py, repositories.py, services_nasa_image_gallery.py y views.py para una correcta comunicación entre la app y la base de datos. Utilizando la consola, se hizo una migración a la base de datos para que se cree una nueva tabla y se modificaron los archivos home.html para que se muestre el formulario de comentarios y la función SweetAlert2 para que se vean en pantalla los messages que confirman que se realizó con éxito la operación. Y favourites.html para agregar una nueva sección que muestre el comentario.




SUCCESS

¡La imagen se ha guardado en favoritos correctamente!

Listado de FAVORITOS

| # | Imagen | Título | Descripción | Fecha | Comentario | Acciones |
|---|---|-------------------------|-------------------------|--------------|----------------------|---|
| - |  | Space Exploration Video | Space Exploration Video | Oct. 1, 2018 | Comentario de prueba |  |

Se creó un paginado para administrar la cantidad de imágenes que se muestran tanto en el home como al realizar una búsqueda. Inicialmente se muestran 5 fotos, pero el usuario puede elegir en mostrar 10 o 20, además de poder saltar directamente a la última página o ir pasando de una en una. Utilizando “Django Paginator”.

Proyecto TP
Inicio
Galería
Iniciar sesión

Galería de Imágenes de la NASA

Imágenes por página:
5
5
10
20

EXPLORAMOONtoMARS



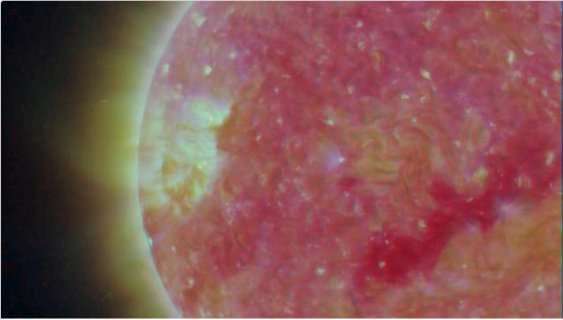
Space Exploration Video
Space Exploration Video



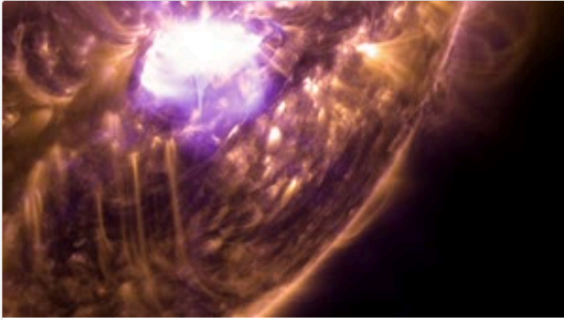
Space 24/7/365: Space Station Benefits
As a home to humans and a lab for scientific research for more than 20 years, the International Space Station is a one-of-a-kind platform for advancing technologies such as robots, computers, health monitors, life support systems, and more for both space and ground applications. The station also has made it possible for numerous experiments to delve into how space travel affects people physically and psychologically – knowledge that often improves the quality of life for people on Earth, too. Learn more: <https://go.nasa.gov/3ATQOq6> Explore other station benefits: <https://www.nasa.gov/stationbenefits> Image Credit: Canadian Space Agency 0.17



Space-to-Ground_171_170407
NASA's Space to Ground is your weekly update on what's happening aboard the International Space Station. Got a question or comment? Use #spacetoground to talk to us. FOLLOW THE SPACE STATION! Twitter: https://twitter.com/Space_Station Facebook: <https://www.facebook.com/ISS> Instagram: <https://instagram.com/iss/>



Close-up View of an Active Region of the Sun, March 23, 2007
 NASA Solar TERrestrial RELations Observatory STEREO satellites have provided the first three-dimensional images of the Sun. The structure of the corona shows well in this image.



Holiday Lights on the Sun
 The sun emitted a significant solar flare, peaking at 7:24 p.m. EST on Dec. 19, 2014. NASA's Solar Dynamics Observatory, which watches the sun constantly, captured an image of the event. Solar flares are powerful bursts of radiation. Harmful radiation from a flare cannot pass through Earth's atmosphere to physically affect humans on the ground, however -- when intense enough -- they can disturb the atmosphere in the layer where GPS and communications signals travel. To see how this event may affect Earth, please visit NOAA's Space Weather Prediction Center at <http://spaceweather.gov>, the U.S. government's official source for space weather forecasts, alerts, watches and warnings. This flare is classified as an X1.8-class flare. X-class denotes the most intense flares, while the number provides more information about its strength. An X2 is twice as intense as an X1, an X3 is three times as intense, etc.

1 < 3 > Última

Introducción a la Programación | UNGS Trabajo práctico - 1er cuatrimestre del 2024.

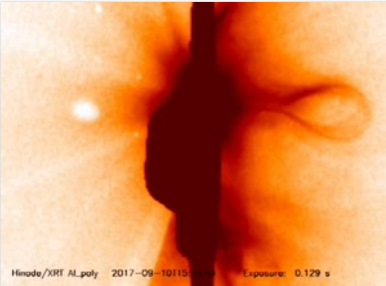
También se elaboró una traducción automática de palabras ingresadas en español, a inglés, para que el admin pueda buscar si es que desconoce la palabra en inglés, reciba resultados en la búsqueda con la palabra en español. Por ejemplo, si busca “sol” en lugar de “sun” aparecen de igual forma imágenes del sol, gracias a la instalación de Google Translate.

Proyecto TP Inicio **Galería** Iniciar sesión

Galería de Imágenes de la NASA

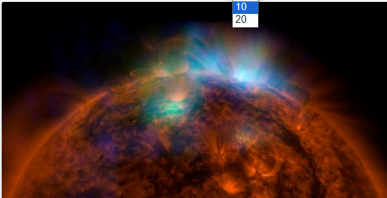
Imágenes por página:

5 10 20




Hinode/XRT Al_poly 2017-09-10T15:00:00 Exposure: 0.129 s

Hinode Takes an X-Ray of a Powerful Solar Flare



Sun Shines in High-Energy X-rays
 X-rays stream off the sun in this first picture of the sun, overlaid on a picture taken by NASA Solar Dynamics Observatory SDO, taken by NASA NuSTAR. The field of view covers the west limb of the sun.



Sun-Earth Day 2009 Promo 1
 Short promo for Sun-Earth Day 2009 featuring scientists and students talking about the most fascinating things they've learned about our sun.

Se encontró dificultad con buscar en español pero al hacer la consulta directamente en inglés no funcionaba porque el programa intentaba traducirlo y no lo reconocía.

Para eso se hizo uso de la biblioteca langdetect y se pudo resolver.

Por último, se modificó la interfaz gráfica para que quede más agradable a la vista utilizando estilos en Bootstrap y cambiando los archivos html de los respectivos apartados.

Conclusión

El trabajo fue positivo en lo que respecta a desafiar nuestras habilidades y capacidad de resolución de conflictos, ya que en el trayecto se presentaron varios, algunos con resolución inmediata después de mucha búsqueda y otros que requerían pruebas seguidas en todos los archivos y también en la plataforma donde se trabajó con todos los archivos, GitHub. Se dejaron opciones como “eliminar resultados de búsqueda no interesantes” o “alta de nuevos usuarios” por una cuestión de practicidad y tiempo, elegimos las opciones no obligatorias que consideramos necesarias para el funcionamiento de una página básica y priorizamos la calidad de funciones y su mantenimiento que su cantidad. Fue un proyecto muy interesante y ayudó a conocer más sobre las tareas que parecen sencillas en una página viéndolas desde el lado del usuario, como usualmente estamos acostumbrados, en realidad toman mucho tiempo y comprensión para que puedan ser útiles y eficaces.