

ExploreSpace - Documentation

Application Frameworks – SE3040

Assignment 2

IT21345296 – Jayakody T.N.A.

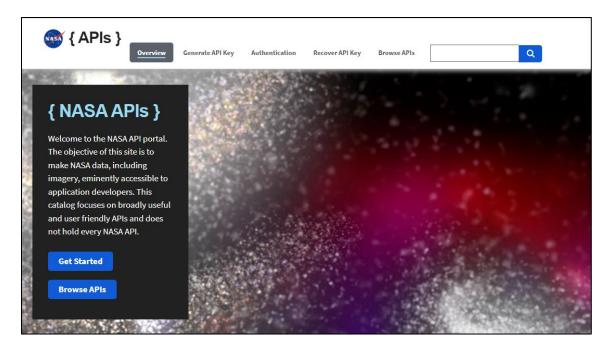
Table of Contents

1.	NASA API's	3
ŀ	How to use the NASA Open API's	3
ſ	NASA Open API's used in Explore Space	4
	APOD – Astronomy Picture Of the Day	4
	Mars Rover Photos	5
2.	Implementation of the system	7
3.	User Interfaces	9
4.	Challenges Faced	13
	Jsage of NASA Open APIs	
	Responsive User Interface Creation	

1. NASA API's

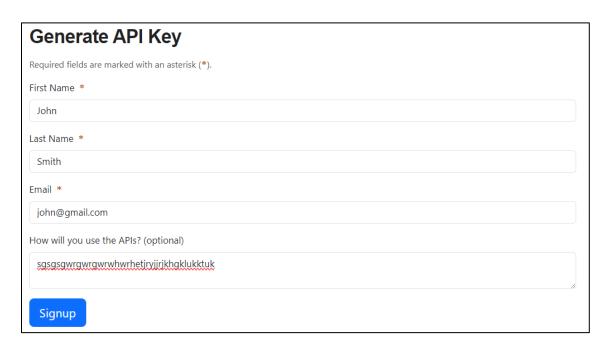
How to use the NASA Open API's

The National Aeronautics and Space Administration which is abbreviated to NASA has made a number of API's available to the public use. They're known as NASA Open API's. They can be found at the website of the NASA API website which is https://api.nasa.gov.



Landing Page of the NASA Open API Website

After arriving at the landing page of the website, people can give their personal information such as the name and the email. Afterwards, an email containing the 'API KEY' for that specific user will be sent to the email provided. That API KEY should be used while fetching information from the specific API listed on the website.



Form to enter details to generate the API KEY

NASA Open API's used in Explore Space

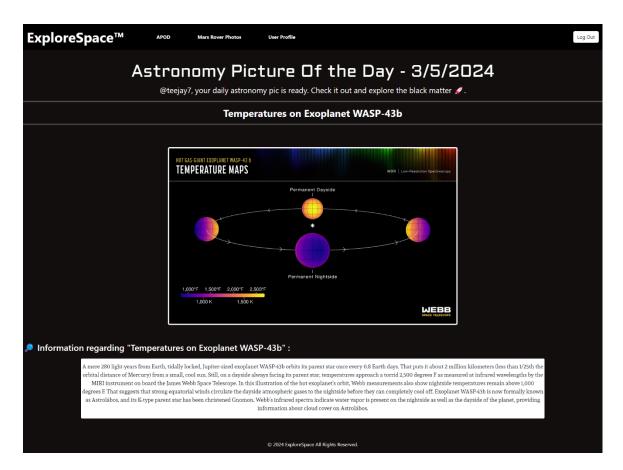
APOD – Astronomy Picture Of the Day

This API is one of the most popular used by developers. Once a request is sent through this API, it returns a picture related to astronomy and it provides a brief explanation regarding the picture in the response. The request needs to be sent with the API KEY that was generated for yourself accordingly.

```
const fetchApodData = async () => {
    try {
        const response = await axios.get('https://api.nasa.gov/planetary/apod?api_key=wC8n5rUiWbcaajhxJRO3GNz14zEGkXWqXfH8E7E0');
        setApodData(response.data);
    } catch (error) {
        console.error("Error fetching APOD data:", error);
    }
}
```

Fetch Request to get the APOD Response

The API implementation done on my project was to load the image in a separate component called APOD, it had details regarding the APOD picture below. For the designing purposes bootstrap components were used and media queries were implemented for the responsive user interface of the APOD component.



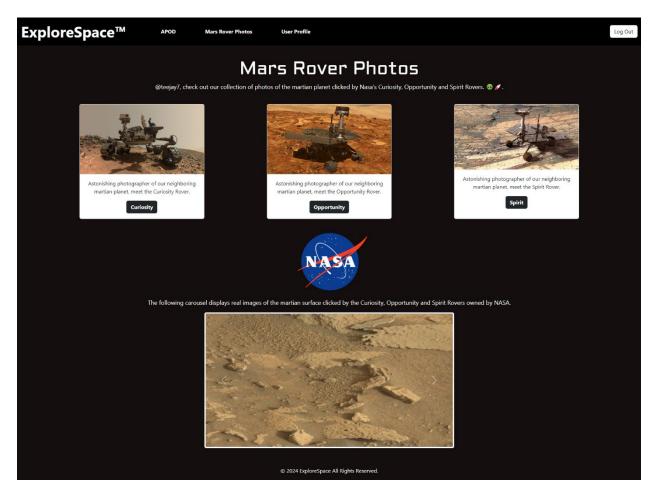
User Interface of the APOD Component

Mars Rover Photos

The Mars Rover Photos API was designed to collect image data gathered by the Rovers on the Martian planet. They are the NASA's Curiosity, Opportunity, and Spirit rovers. This API responds with a collection of images of the surface of the planet Mars when a request is made, and these images are made available to use to the public by the NASA. The request needs to be sent with the API KEY that was generated for yourself accordingly.

Fetch Request to get the Mars Rover Response

The API implementation done on my project was to load the image in a separate component called MarsRover, it had 3 separate cards that redirects you to information regarding the 3 separate rovers that was on Mars for NASA. For the designing purposes bootstrap components were used and media queries were implemented for the responsive user interface of the Mars Rover component. The carousel bootstrap component to use the images that provided in the response in a slideshow, where the user can go back and forth with the images.

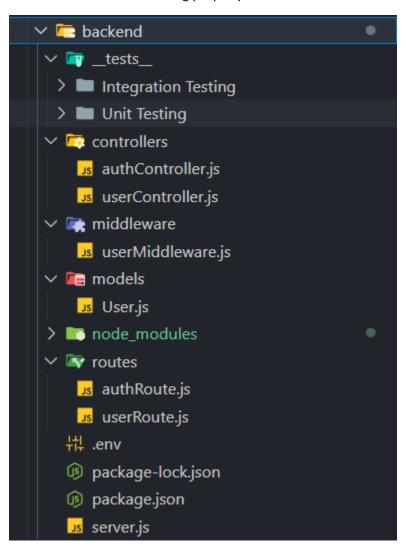


User Interface of the MarsRover Component

2.Implementation of the system

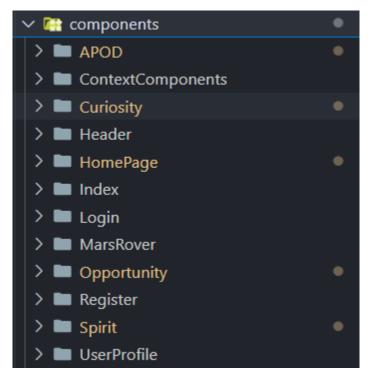
For development of this system various frameworks were used in both frontend and the backend of this system, ExploreSpace. For the frontend development the javascript based framework React was used and for the backend development Node which is also a javascript based framework. The backend development was not a big development for this system as most of the API's were taken from the NASA's Open API portal.

The backend development was done for the user management system of ExploreSpace. I developed a collection of APIs that allowed the users to register themselves and login to the system. Authentication is also needed to login to the system and access the other components for the user. This was done by using jsonwebtoken node package, which gives a random JSON web token everytime the user is logged in, so the user must be logged in to access the components with a valid JSON web token for authentication. The testing of the APIs were done using the Postman API Testing tool and also used Jest node package to see if the controllers are working properly.



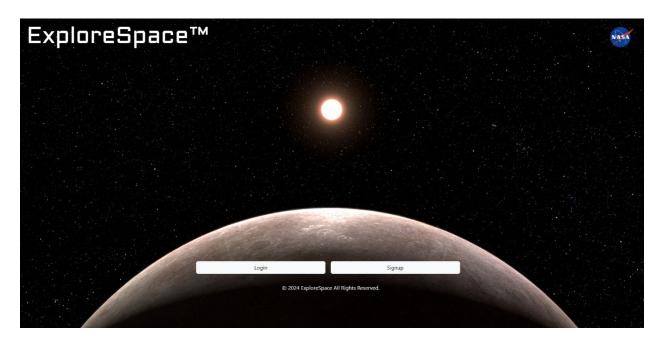
File Structure of the backend system

The frontend development of ExploreSpace was done using the React framework. It is a component-based framework which is used for frontend development. I used the function-based components for the development of the components over class-based components. I also used Bootstrap as a CSS framework which was used to get components such as Cards, Forms, and Carousels.

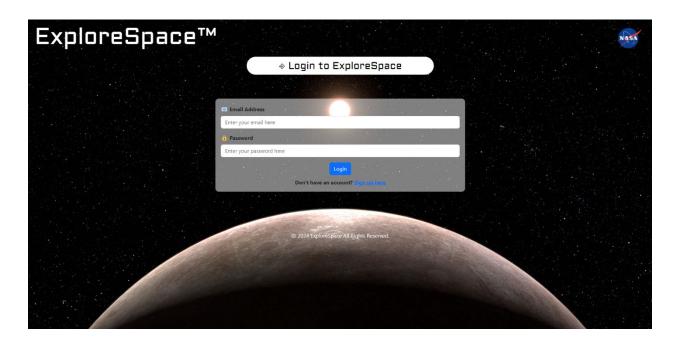


Set of Components that were developed for the FE

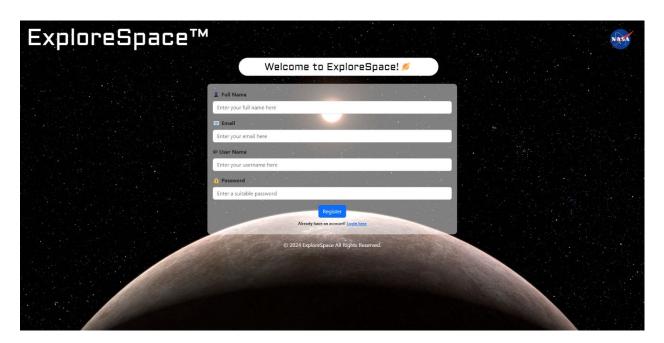
3. User Interfaces



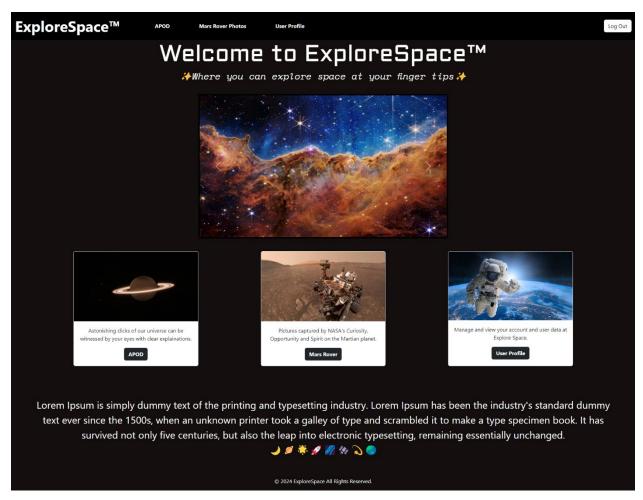
Index Page



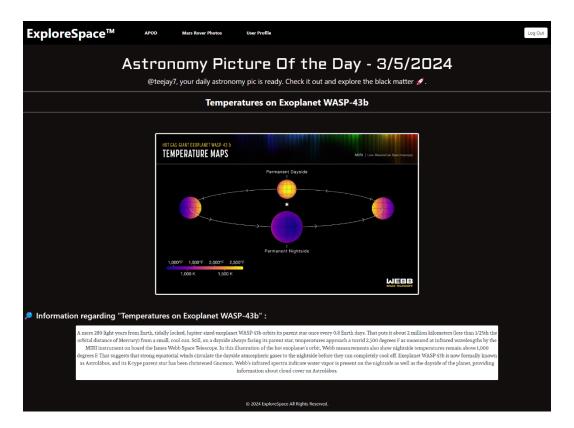
Login Page



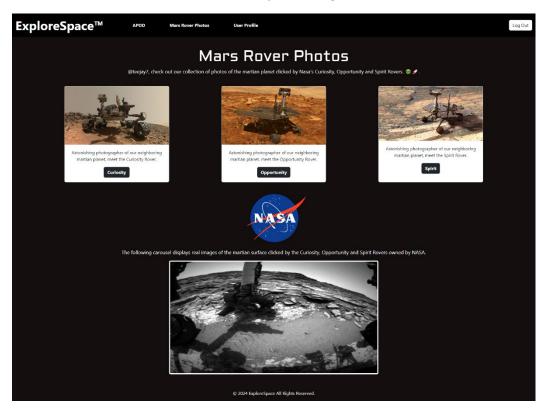
Sign up / Register Page



Main Page / Home Page



APOD Component Page



Mars Rover API Component



Rover Information Page



User Profile Page

4. Challenges Faced

Usage of NASA Open APIs

The main challenge that was faced by me was to get to know how to use the APIs that are available in the NASA Open API Library. I went through the documentation provided by NASA regarding the APIs. Initially I did not get a full understanding regarding these APIs. I followed a YouTube tutorial that was uploaded by the YouTuber "Smoljames", which gave me a proper understanding for the first time.



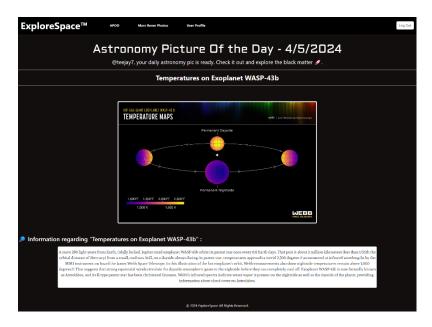
Tutorial that was followed for the understanding of the API

After following this tutorial, I was able to know how to use the APIs and how to generate the API KEY for myself. Afterwards I was able to use the APOD, Mars Rover NASA Open APIs in my project accordingly.

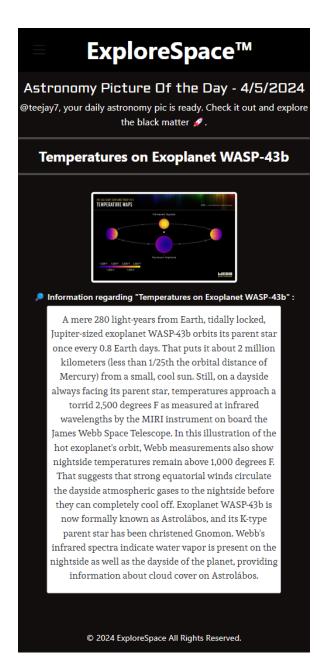
Responsive User Interface Creation

The next challenge I faced was creating the Responsive User Interfaces. Previously, in most of the project I did not care about the responsiveness of the user interfaces. I finally learnt how to make the screens responsive according to the screen size. I was able to master the use of Media queries in CSS which allowed me to make CSS changes depending on the size of the screen.

Usage of Media Queries



Normal User Interface in a Laptop Screen



Normal User Interface in a Smaller (Mobile) Screen