Space Explorer Project Report

Pakinam Khaled

May 2, 2025

Contents

1 Introduction

The **Space Explorer** project is a Node.js application that integrates with various space-related APIs to provide users with up-to-date information about astronomical phenomena, the International Space Station (ISS), and upcoming space launches. This report details the implementation steps, code structure, and testing procedures undertaken to develop the application.

2 Project Setup

2.1 Initializing the Project

1. Initialize a new Node.js project:

```
npm init -y
```

2. Install necessary dependencies:

```
npm install express axios
npm install --save-dev nodemon jest
```

2.2 Project Structure

The project directory is organized as follows:

```
space-explorer/
          services/
                apodService.js
                issService.js
                launchService.js
          tests/
                apodService.test.js
                issService.test.js
                launchService.test.js
          server.js
          package.json
          README.md
```

3 API Integrations

3.1 Astronomy Picture of the Day (APOD)

3.1.1 Service Implementation

```
const axios = require('axios');

const getAPOD = async () => {
  const response = await axios.get(
    'https://api.nasa.gov/planetary/apod?api_key=DEMO_KEY'
  );
  return response.data;
};

module.exports = { getAPOD };
```

Listing 1: services/apodService.js

3.1.2 Testing the APOD Service

```
const { getAPOD } = require('../services/apodService');
const axios = require('axios');

jest.mock('axios');

describe('NASA APOD API', () => {
  it('should return image data', async () => {
    const mockData = {
      data: {
        title: "Mock Title",
        explanation: "Mock Explanation",
        date: "2024-12-12",
        url: "http://example.com/image.jpg"
      }
    };
    axios.get.mockResolvedValue(mockData);

    const result = await getAPOD();
    expect(result.title).toBe("Mock Title");
    expect(result.explanation).toBe("Mock Explanation");
    expect(result.url).toBe("http://example.com/image.jpg");
  });
});
```

Listing 2: tests/apodService.test.js

3.1.3 Test Result

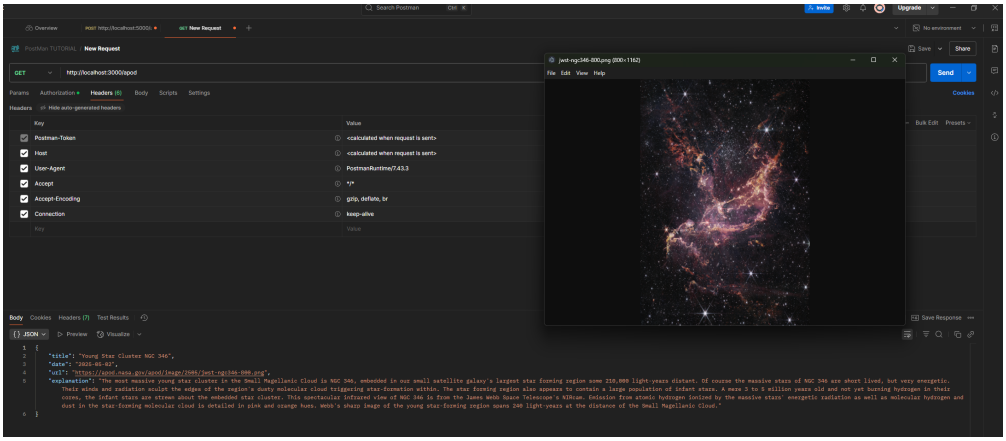

Figure 1: APOD Service Test Result

3.2 International Space Station (ISS) Location Tracker

3.2.1 Service Implementation

```
const axios = require('axios');

const getISSLocation = async () => {
  const response = await axios.get(
    'http://api.open-notify.org/iss-now.json'
  );
  return response.data;
};

module.exports = { getISSLocation };
```

Listing 3: services/issService.js

3.2.2 Testing the ISS Service

```
const { getISSLocation } = require('../services/issService');
const axios = require('axios');

jest.mock('axios');

describe('ISS Location API', () => {
  it('should return ISS location data', async () => {
    const mockData = {
      data: {
        iss_position: {
          latitude: "45.0",
          longitude: "-122.3"
        },
        timestamp: 1596567890,
```

```
        message: "success"
      }
    };
    axios.get.mockResolvedValue(mockData);

    const result = await getISSLocation();
    expect(result.iss_position.latitude).toBe("45.0");
    expect(result.iss_position.longitude).toBe("-122.3");
    expect(result.message).toBe("success");
  });
});
```

Listing 4: tests/issService.test.js

3.2.3 Test Result

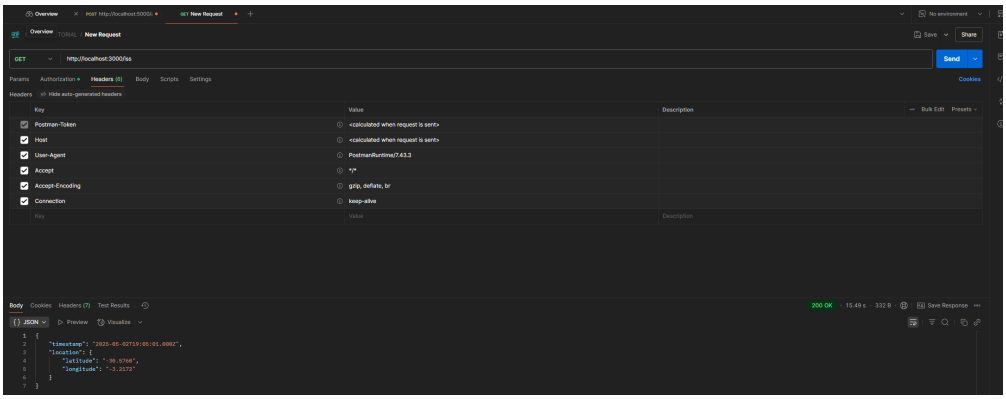

Figure 2: ISS Service Test Result

3.3 Upcoming Launches

3.3.1 Service Implementation

```
const axios = require('axios');

const getUpcomingLaunches = async () => {
  const response = await axios.get(
    'https://ll.thespacedevs.com/2.2.0/launch/upcoming/'
  );
  return response.data;
};

module.exports = { getUpcomingLaunches };
```

Listing 5: services/launchService.js

3.3.2 Testing the Launch Service

```
const { getUpcomingLaunches } = require('../services/launchService');
const axios = require('axios');

jest.mock('axios');

describe('Launch Library API', () => {
  it('should return upcoming launches', async () => {
    const mockData = {
      data: {
        results: [
          {
            name: "Mission Alpha",
            net: "2025-05-10T14:00:00Z",
            rocket: {
              configuration: {
                name: "Falcon 9"
              }
            }
          }
        ]
      }
    };
    axios.get.mockResolvedValue(mockData);

    const result = await getUpcomingLaunches();
    expect(result.results[0].name).toBe("Mission Alpha");
    expect(result.results[0].rocket.configuration.name).toBe("Falcon 9");
  });
});
```

Listing 6: tests/launchService.test.js

3.3.3 Test Result

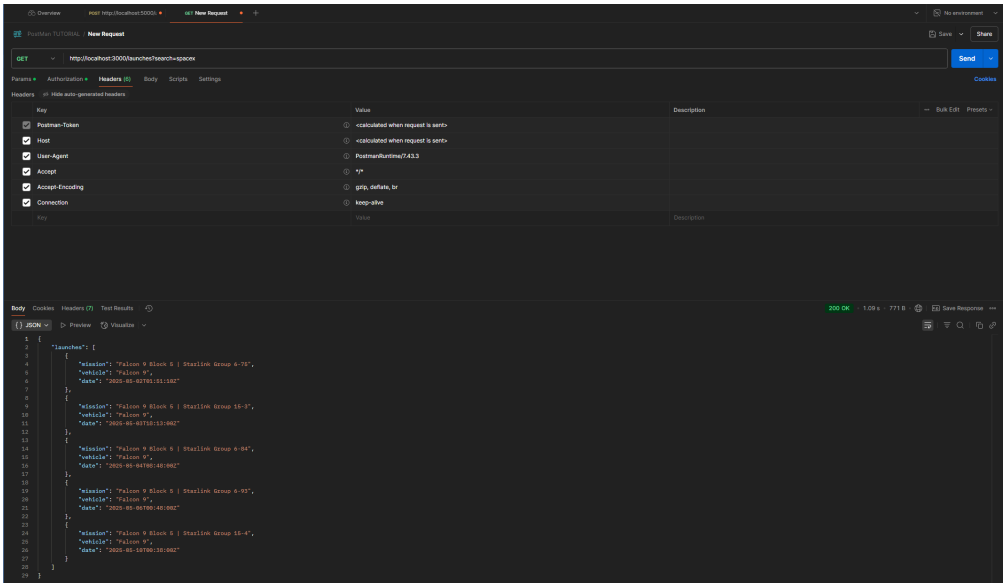

Figure 3: Launch Service Test Result

4 Final Testing

After implementing and testing all services individually, a comprehensive test was conducted to ensure the integration of all components works seamlessly.

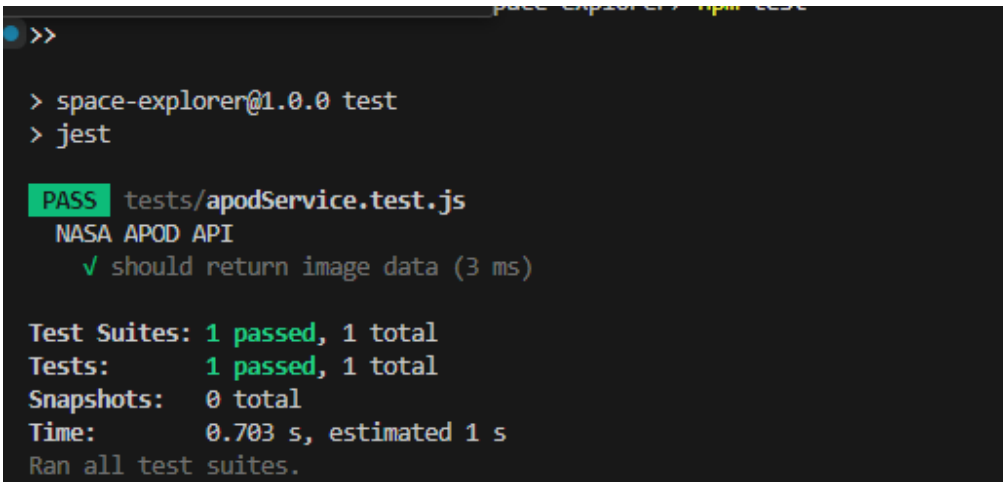

Figure 4: Final Integration Test Result

5 Conclusion

The **Space Explorer** project successfully integrates multiple space-related APIs to provide users with real-time information. Through modular code design and thorough testing using

Jest, the application demonstrates reliability and scalability for future enhancements.