

Re-Design: Service Layer(s)

Elysia Burton

SWDV 691 - Software Development Capstone

Professor Joseph Gradecki

11/12/2023

Added:

Considering the feedback, I included endpoints for posting new events and missions.

POST '/api/events' (Admins only)

Purpose: Allows admins to create new space events.

Sample Request: { "eventName": "New Comet Sighting", "eventDate": "2024-12-05",
"description": "Visible from Northern Hemisphere" }

Sample Response: { "status": "success", "message": "Event created successfully" }

POST /api/missions (Admin only)

Purpose: Allows admin to add new missions to the dashboard.

Sample Request: { "missionName": "Mars Rover 2025", "launchDate": "2025-08-14",
"objectives": ["Explore Mars surface", "Collect soil samples"] }

Sample Response: { "status": "success", "message": "Mission added successfully" }

For stretch feature endpoints:

GET /api/archive

Purpose: Retrieve the historical archive of past space missions for research purposes.

Sample Response: [{ "archiveID": "arch123", "missionName": "Apollo 11", "historicalData": {
... } }]

POST /api/forum/posts (For community forum)

Purpose: Allows users to create new discussion posts.

Sample Request: { "userID": "user123", "content": "Discussing the potential of life on Mars." }

Sample Response: { "status": "success", "message": "Post created successfully" }

POST /api/forum/replies (For community forum)

Purpose: Allows users to reply to discussion posts.

Sample Request: { "postID": "post123", "userID": "user124", "content": "I think we will find
microbial life first." }

Sample Response: { "status": "success", "message": "Reply added successfully" }

Updated diagram of service endpoints:

Login/Register Page --(POST)--> /api/users/login or /api/users/register
Admin Dashboard Page --(POST)--> /api/events, /api/missions (Admin only)
Dashboard Overview Page --(GET)--> /api/missions, /api/events
Mission Details Page --(GET)--> /api/missions/{missionID}
[Stretch] Historical Archive Page --(GET)--> /api/archive
[Stretch] Community Forum Page --(POST)--> /api/forum/posts, /api/forum/replies

//Below is the original submission

Space Exploration Dashboard: Service Layer Design

Service Endpoints:

a) User Management

Endpoint: /api/users/register

HTTP Verb: POST

Purpose: Register a new user.

Sample Request:

```
{
  "username": "astronautJohn",
  "password": "securePass123",
  "name": "John Doe",
  "email": "john.doe@space.com"
}
```

Sample Response:

```
{
  "status": "success",
  "message": "User registered successfully!"
}
```

Endpoint: /api/users/login

HTTP Verb: POST

Purpose: Authenticate and login a user.

Sample Request:

```
{
  "username": "astronautJohn",
  "password": "securePass123"
}
```

Sample Response:

```
{
  "status": "success",
  "token": "authToken123456"
}
```

b) Missions

Endpoint: /api/missions

HTTP Verb: GET

Purpose: Retrieve a list of all space missions.

Sample Response:

```
[
  {
    "missionID": "1",
    "missionName": "Apollo 11",
    "launchDate": "1969-07-16",
    "missionStatus": "Completed"
  },
  // ... I would include more missions here
]
Endpoint: /api/missions/{missionID}
HTTP Verb: GET
Purpose: Retrieve details of a specific mission.
Sample Response:
{
  "missionID": "1",
  "missionName": "Apollo 11",
  "launchDate": "1969-07-16",
  "crewMembers": ["Neil Armstrong", "Buzz Aldrin", "Michael Collins"],
  "objectives": ["Land on the moon", "Perform moonwalks", "Safely return to Earth"],
  "missionStatus": "Completed"
}
```

c) Events

Endpoint: /api/events
 HTTP Verb: GET
 Purpose: Fetch all space exploration events.
 Sample Response:

```
[
  {
    "eventID": "e001",
    "eventName": "First Moonwalk",
    "eventDate": "1969-07-20",
    "description": "Neil Armstrong becomes the first human to walk on the moon.",
    "relatedMission": "1"
  },
]
```

2. Communication Diagram:

1. Login/Register Page --POST--> /api/users/login or /api/users/register
2. Dashboard --GET--> /api/missions and /api/events
3. Mission Details Page --GET--> /api/missions/{missionID}