Software
made for you

Construction Project Management API -Training Project

Project Overview

The **Construction Project Management API** is a RESTful backend application built using **ASP.NET Core Web API**.
It helps construction teams manage **projects, tasks, and engineers/supervisors**, while providing **secure access using JWT-based authentication**.

This project is part of a **construction domain capstone**, focusing on clean REST design, validation, authentication, and CRUD operations using a relational database.

Tech Stack

- **Backend:** ASP.NET Core Web API (.NET)
- **Language:** C#
- **Database:** SQL Server
- **ORM:** Entity Framework Core
- **Authentication:** JWT (JSON Web Token)
- **API Documentation:** Swagger
- **Testing:** Postman
- **Version Control:** Git & GitHub

Core Features

- JWT-based **authentication & authorization**
- Secure access to protected endpoints
- CRUD operations for **Projects**
- CRUD operations for **Tasks**
- CRUD operations for **Engineers / Supervisors**
- Assign tasks to projects and engineers
- Track task status (Planned, In Progress, Completed)
- Input validation using **Data Annotations**
- REST-compliant endpoints with correct HTTP status codes
- Swagger UI for API documentation and testing

Authentication (JWT)

- Users authenticate using login credentials
- On successful login, a **JWT token** is generated
- The token must be included in the `Authorization` header for protected endpoints

```
Authorization: Bearer <JWT_TOKEN>
```

- Unauthorized requests return **401 Unauthorized**
- Role-based or claim-based authorization can be extended if required

API Endpoints (Overview)

Authentication

- `POST /api/auth/login` – Authenticate user and generate JWT token
- `POST /api/auth/register` – Register a new user (if enabled)

Projects

- `GET /api/projects`
- `GET /api/projects/{id}`
- `POST /api/projects`
- `PUT /api/projects/{id}`
- `DELETE /api/projects/{id}`

Tasks

- `GET /api/tasks`
- `GET /api/tasks/{id}`
- `GET /api/projects/{projectId}/tasks`
- `POST /api/tasks`
- `PUT /api/tasks/{id}`
- `DELETE /api/tasks/{id}`

Engineers / Supervisors

- `GET /api/engineers`
- `GET /api/engineers/{id}`
- `POST /api/engineers`
- `PUT /api/engineers/{id}`
- `DELETE /api/engineers/{id}`

Setup & Execution Steps

1. **Clone the repository**
2. `git clone https://github.com/eg-sougo/CRUD-App-Construction-Backend.git`
3. **Open the project**
   - Open the solution in **Visual Studio**
4. **Configure JWT & Database**
   - Update `appsettings.json`:
     - Database connection string
     - JWT settings (Issuer, Audience, Secret Key)
5. **Apply migrations**
6. `Update-Database`
7. **Run the application**
   - Press `F5` or run via Visual Studio
8. **Access Swagger**
9. `https://localhost:{port}/swagger`
10. **Authorize in Swagger**
    - Click **Authorize**
    - Enter:
    - `Bearer <JWT_TOKEN>`

Key Assumptions

- Each task belongs to a single project
- Tasks are assigned to one engineer
- JWT token is required to access protected endpoints
- Authentication is stateless
- Role-based authorization is minimal / basic

Software
made for you

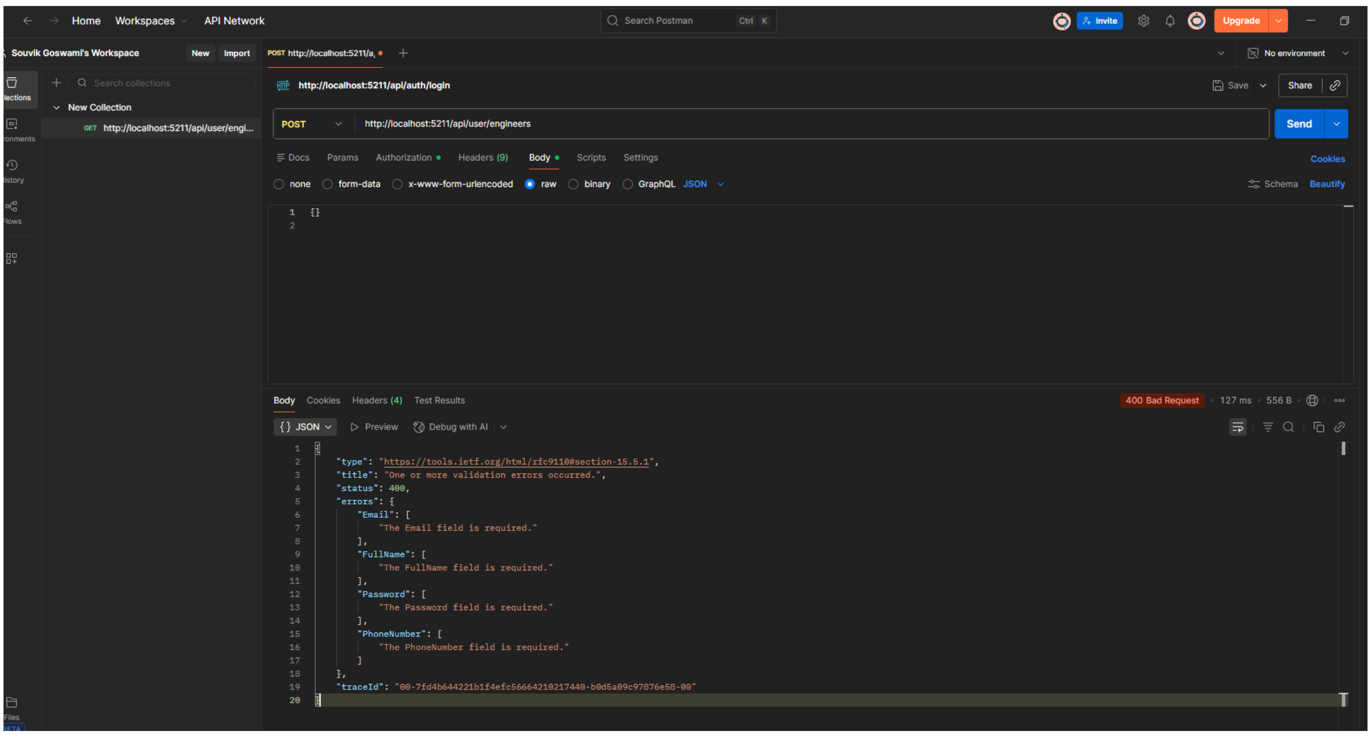
Testcases Screenshots :

Test Case 1: Empty Engineer Creation Request

```
POST /api/user/engineers
Content-Type: application/json

{}
```

Expected Result: 400 Bad Request with errors for FullName, Email, and PhoneNumber



```
{}
```

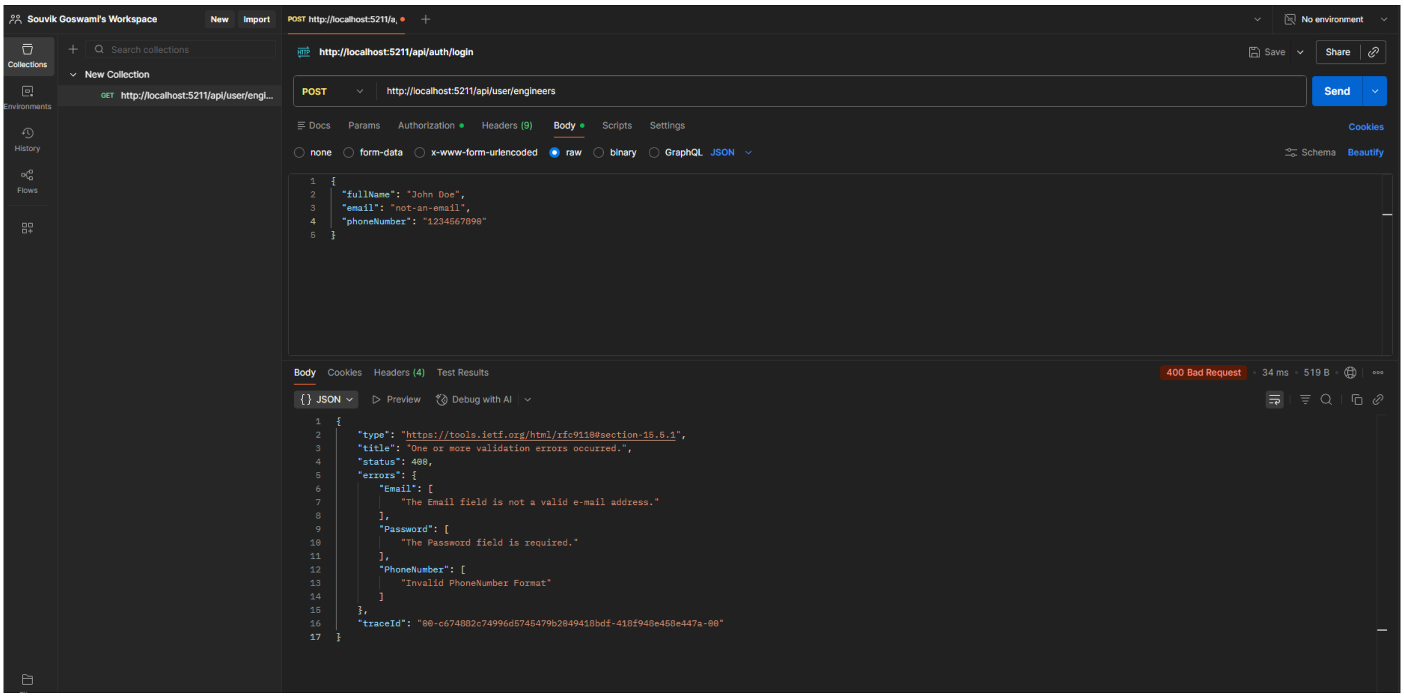
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "Email": [
            "The Email field is required."
        ],
        "FullName": [
            "The FullName field is required."
        ],
        "Password": [
            "The Password field is required."
        ],
        "PhoneNumber": [
            "The PhoneNumber field is required."
        ]
    },
    "traceId": "00-7fd4b644221b1f4efc56664210217440-b0d5a09c97876e58-00"
}
```

Test Case 2: Invalid Email Format

```
POST /api/user/engineers
Content-Type: application/json

{
  "fullName": "John Doe",
  "email": "not-an-email",
  "phoneNumber": "1234567890"
}
```

Expected Result: 400 Bad Request with error: "Invalid email format"



POST http://localhost:5211/api/user/engineers

Body: none / form-data / x-www-form-urlencoded / [✓ raw] / binary / GraphQL — JSON

```
{
  "fullName": "John Doe",
  "email": "not-an-email",
  "phoneNumber": "1234567890"
}
```

Response Body — JSON — 400 Bad Request

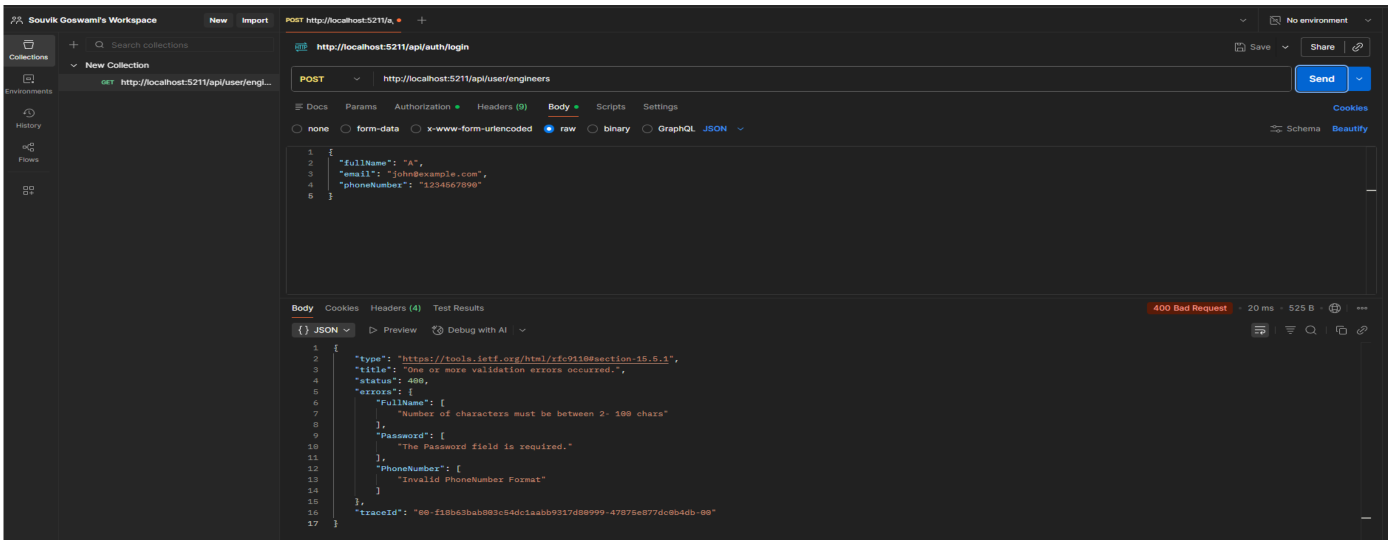
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "Email": [
            "The Email field is not a valid e-mail address."
        ],
        "Password": [
            "The Password field is required."
        ],
        "PhoneNumber": [
            "Invalid PhoneNumber Format"
        ]
    },
    "traceId": "00-c674882c74996d5745479b2049418bdf-418f948e458e447a-00"
}
```

Test Case 3: Name Too Short

```
POST /api/user/engineers
Content-Type: application/json

{
  "fullName": "A",
  "email": "john@example.com",
  "phoneNumber": "1234567890"
```

Expected Result: 400 Bad Request with error: "Full name must be between 2 and 100 characters"



POST http://localhost:5211/api/user/engineers

Body: none / form-data / x-www-form-urlencoded / [✓ raw] / binary / GraphQL — JSON

```
{
  "fullName": "A",
  "email": "john@example.com",
  "phoneNumber": "1234567890"
}
```

Response Body — JSON — 400 Bad Request

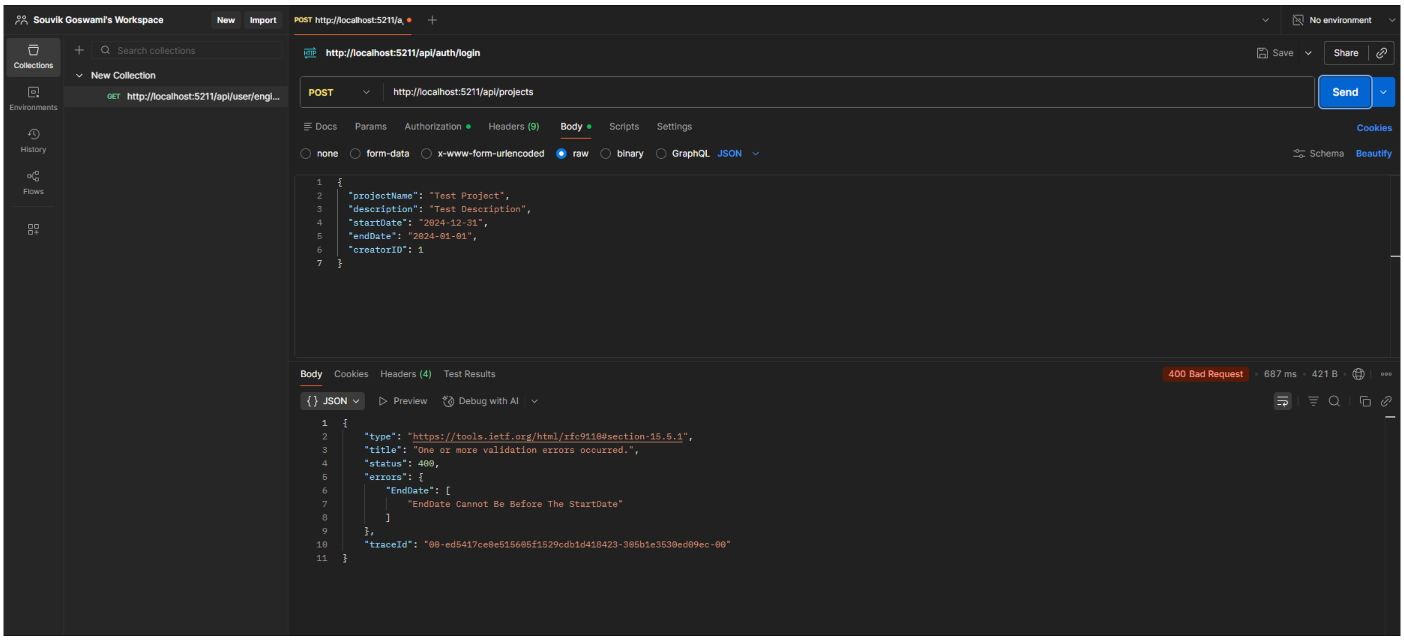
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "FullName": [
            "Number of characters must be between 2- 100 chars"
        ],
        "Password": [
            "The Password field is required."
        ],
        "PhoneNumber": [
            "Invalid PhoneNumber Format"
        ]
    },
    "traceId": "00-f18b63bab803c54dc1aabb9317d80999-47875e877dc0b4db-00"
}
```

Test Case 4: Invalid Date Range (Project)

```
POST /api/projects
Content-Type: application/json

{
  "projectName": "Test Project",
  "description": "Test Description",
  "startDate": "2024-12-31",
  "endDate": "2024-01-01",
  "creatorID": 1
}
```

Expected Result: 400 Bad Request with error: "End date must be after start date"



POST http://localhost:5211/api/projects

```
{
  "projectName": "Test Project",
  "description": "Test Description",
  "startDate": "2024-12-31",
  "endDate": "2024-01-01",
  "creatorID": 1
}
```

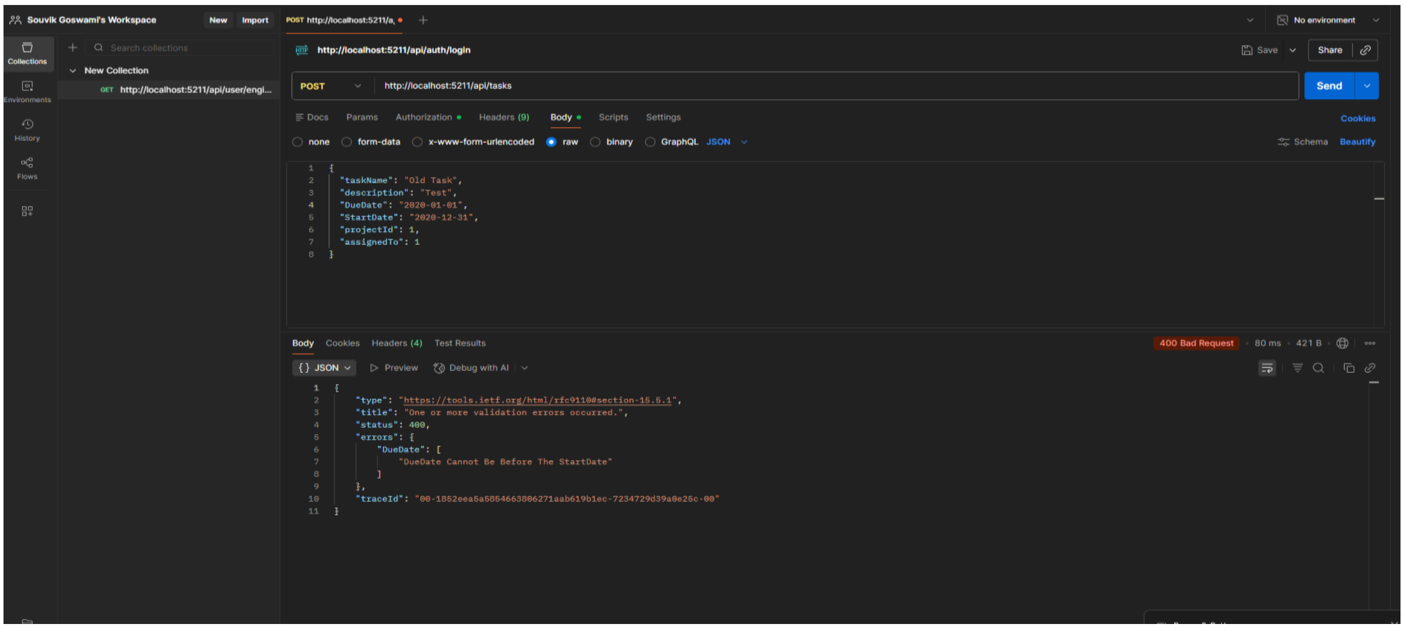
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "EndDate": [
            "EndDate Cannot Be Before The StartDate"
        ]
    },
    "traceId": "00-ed5417ce0e515605f1529cdb1d418423-305b1e3530ed09ec-00"
}
```

Test Case 5: Past Start Date (Task)

```
POST /api/tasks
Content-Type: application/json

{
  "taskName": "Old Task",
  "description": "Test",
  "startDate": "2020-01-01",
  "dueDate": "2020-12-31",
  "projectId": 1,
  "assignedTo": 1
}
```

Expected Result: 400 Bad Request with error: "Start date cannot be in the past"



POST http://localhost:5211/api/tasks

```
{
  "taskName": "Old Task",
  "description": "Test",
  "DueDate": "2020-01-01",
  "StartDate": "2020-12-31",
  "projectId": 1,
  "assignedTo": 1
}
```

400 Bad Request · 80 ms · 421 B

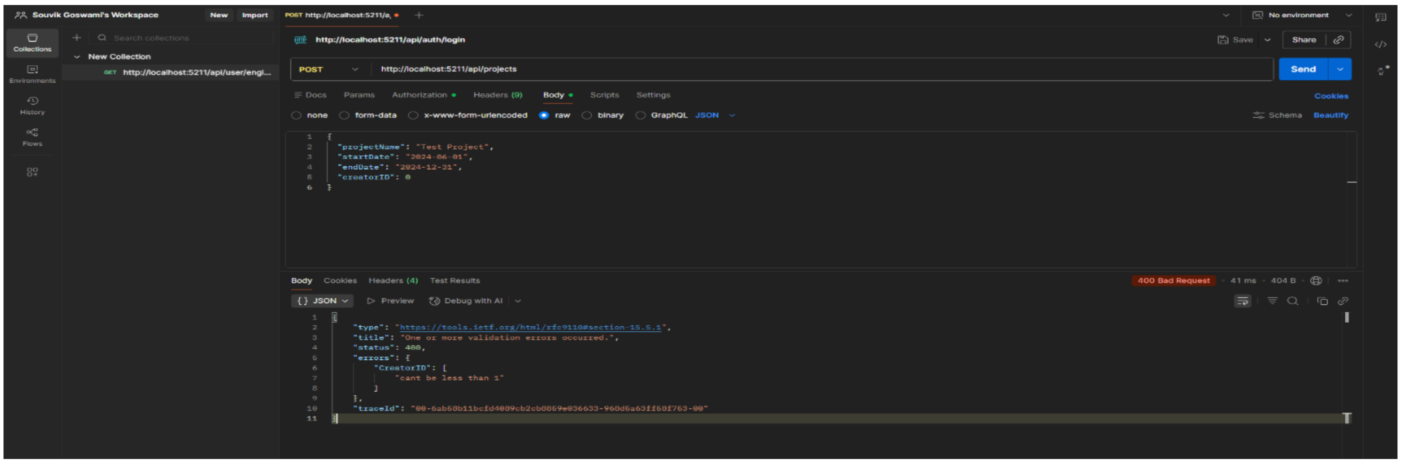
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "DueDate": [
            "DueDate Cannot Be Before The StartDate"
        ]
    },
    "traceId": "00-1852eea5a5854663806271aab619b1ec-7234729d39a0e25c-00"
}
```

Test Case 6: Invalid ID Values

```
POST /api/projects
Content-Type: application/json

{
  "projectName": "Test Project",
  "startDate": "2024-06-01",
  "endDate": "2024-12-31",
  "creatorID": 0
}
```

Expected Result: 400 Bad Request with error: "Creator ID must be greater than 0"



POST http://localhost:5211/api/projects

```
{
  "projectName": "Test Project",
  "startDate": "2024-06-01",
  "endDate": "2024-12-31",
  "creatorID": 0
}
```

400 Bad Request · 41 ms · 404 B

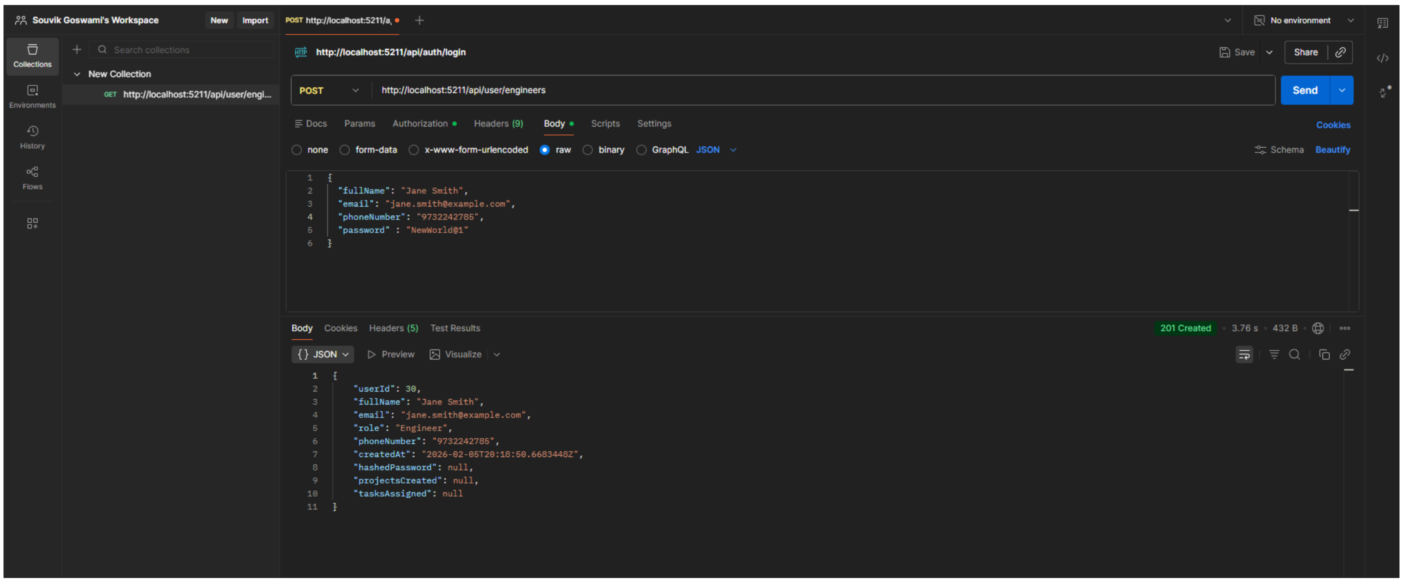
```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "CreatorID": [
            "cant be less than 1"
        ]
    },
    "traceId": "00-6ab58b11bcfd4089cb2cb8859e036633-968d5a63ff58f753-00"
}
```

Test Case 7: Valid Engineer Creation

```
POST /api/user/engineers
Content-Type: application/json

{
  "fullName": "Jane Smith",
  "email": "jane.smith@example.com",
  "phoneNumber": "9876543210"
}
```

Expected Result: 201 Created with engineer details



POST http://localhost:5211/api/user/engineers

none | form-data | x-www-form-urlencoded | [✓ raw] | binary | GraphQL | JSON

```
{
  "fullName": "Jane Smith",
  "email": "jane.smith@example.com",
  "phoneNumber": "9732242785",
  "password" : "NewWorld@1"
}
```

201 Created • 3.76 s • 432 B

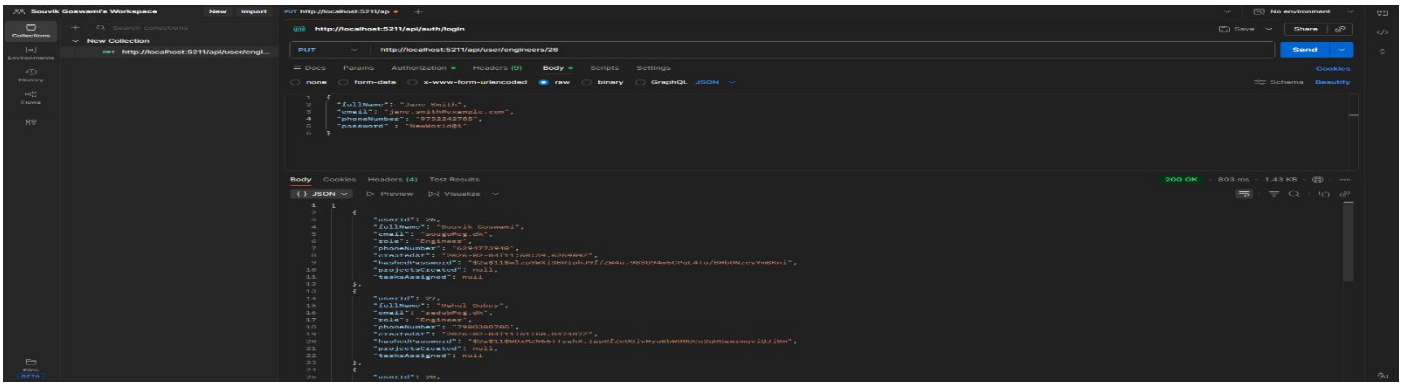
```
{
    "userId": 30,
    "fullName": "Jane Smith",
    "email": "jane.smith@example.com",
    "role": "Engineer",
    "phoneNumber": "9732242785",
    "createdAt": "2026-02-05T20:18:50.6683448Z",
    "hashedPassword": null,
    "projectsCreated": null,
    "tasksAssigned": null
}
```

Test Case 8: Valid Update with Partial Data

```
PUT /api/user/engineers/1
Content-Type: application/json

{
  "fullName": "John Updated"
}
```

Expected Result: 200 OK with updated engineer details (email and phone unchanged)



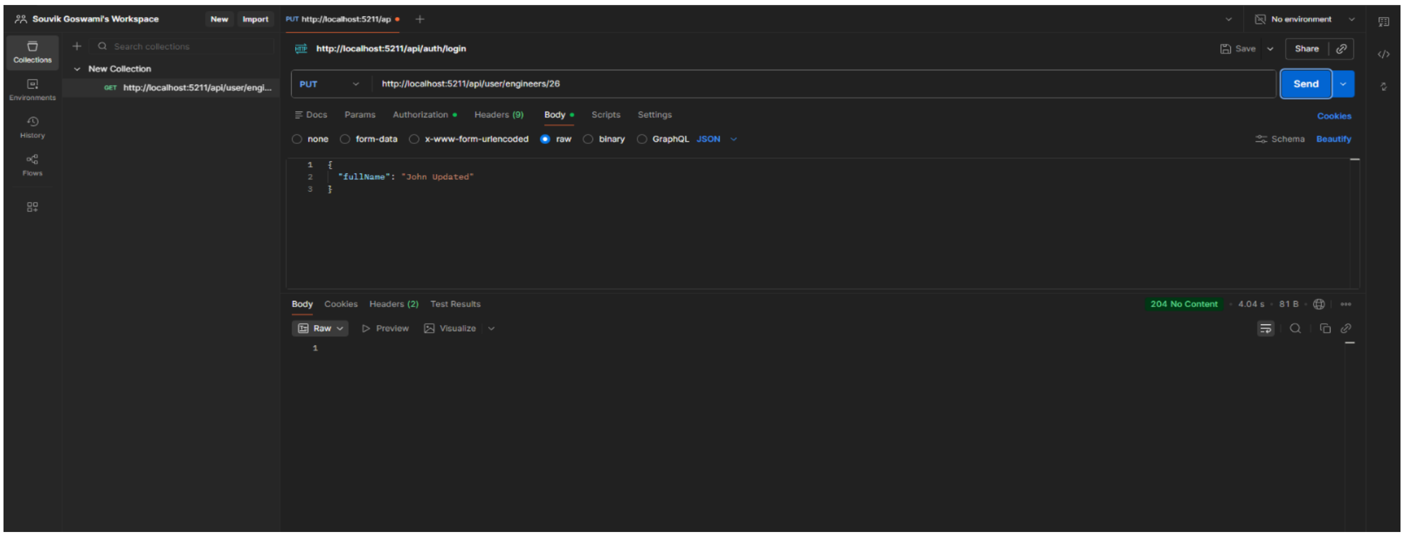
PUT http://localhost:5211/api/user/engineers/28

none | form-data | x-www-form-urlencoded | [✓ raw] | binary | GraphQL | JSON

```
{
  "fullName": "Jane Smith",
  "email": "jane.smith@example.com",
  "phoneNumber": "9732242785",
  "password" : "NewWorld@1"
}
```

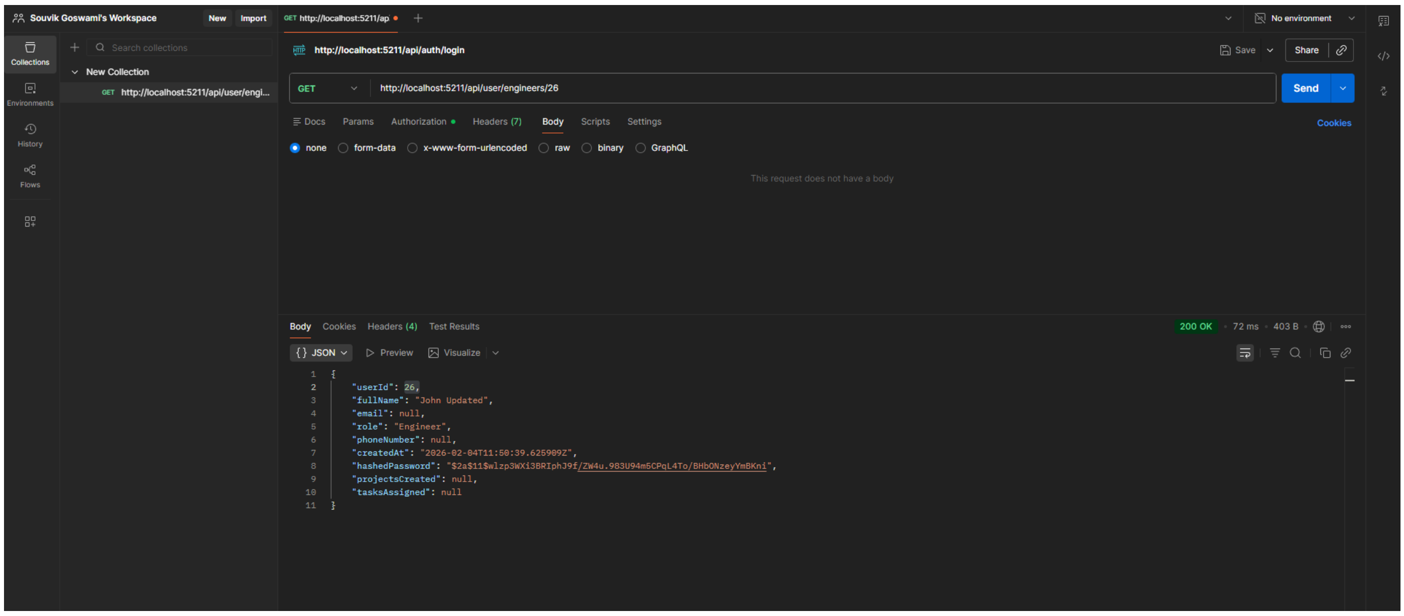
200 OK • 803 ms • 1.43 KB

```
[
    {
        "userId": 26,
        "fullName": "Souvik Goswami",
        "email": "sougo@eg.dk",
        "role": "Engineer",
        "phoneNumber": "6294773948",
        "createdAt": "[illegible]",
        "hashedPassword": "[illegible]",
        "projectsCreated": null,
        "tasksAssigned": null
    },
    {
        "userId": 27,
        "fullName": "Rahul Dubey",
        "email": "radub@eg.dk",
        "role": "Engineer",
        "phoneNumber": "[illegible]",
        "createdAt": "[illegible]",
        "hashedPassword": "[illegible]",
        "projectsCreated": null,
        "tasksAssigned": null
    },
    {
        "userId": 28,
```


PUT http://localhost:5211/api/user/engineers/26

```
{
  "fullName": "John Updated"
}
```

204 No Content · 4.04 s · 81 B


GET http://localhost:5211/api/user/engineers/26

This request does not have a body

200 OK · 72 ms · 403 B

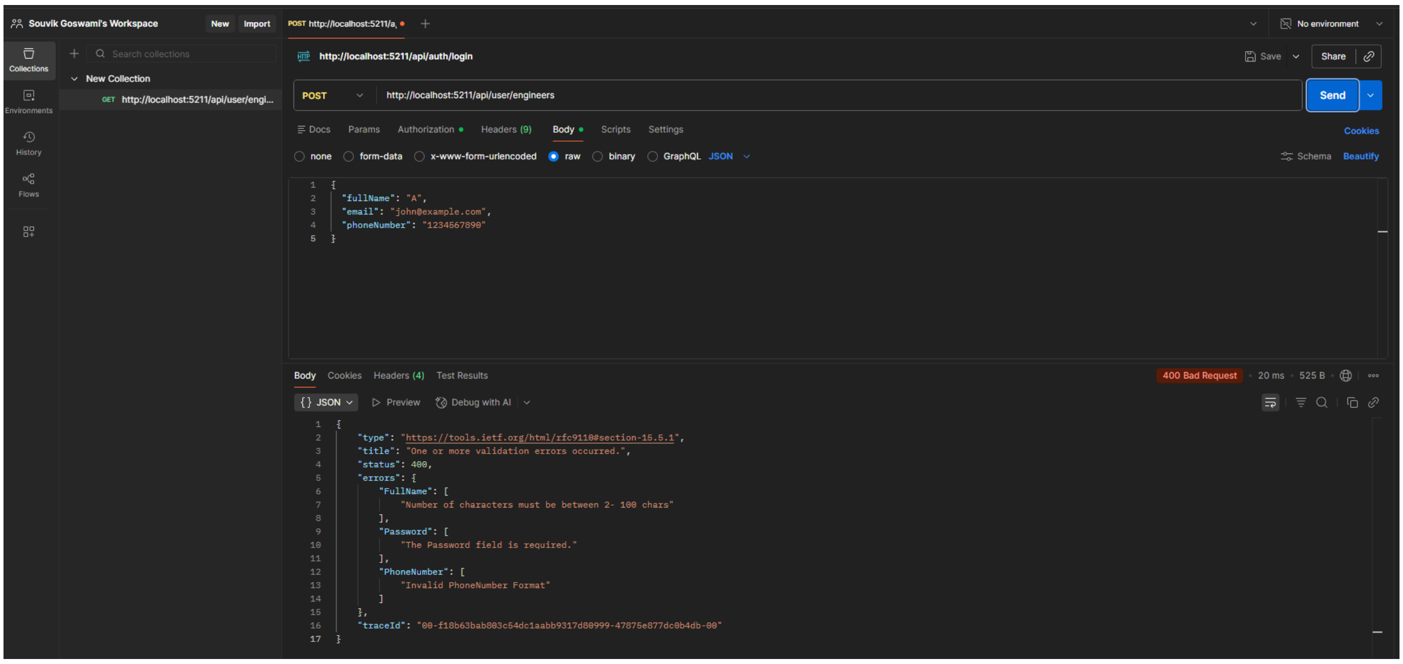
```
{
    "userId": 26,
    "fullName": "John Updated",
    "email": null,
    "role": "Engineer",
    "phoneNumber": null,
    "createdAt": "2026-02-04T11:50:39.625909Z",
    "hashedPassword": "$2a$11$wlzp3WXi3BRIphJ9f/ZW4u.983U94m5CPqL4To/BHbONzeyYmBKni",
    "projectsCreated": null,
    "tasksAssigned": null
}
```

Test Case 9: Multiple Validation Errors

```
POST /api/tasks
Content-Type: application/json

{
  "taskName": "AB",
  "projectId": -1,
  "assignedTo": 0
}
```

Expected Result: 400 Bad Request with errors for TaskName (too short), StartDate (required), DueDate (required), ProjectId (invalid), AssignedTo (invalid)


Souvik Goswami's Workspace

New Collection — GET http://localhost:5211/api/user/engi...

http://localhost:5211/api/auth/login

POST http://localhost:5211/api/user/engineers

Body: raw JSON

```
{
  "fullName": "A",
  "email": "john@example.com",
  "phoneNumber": "1234567890"
}
```

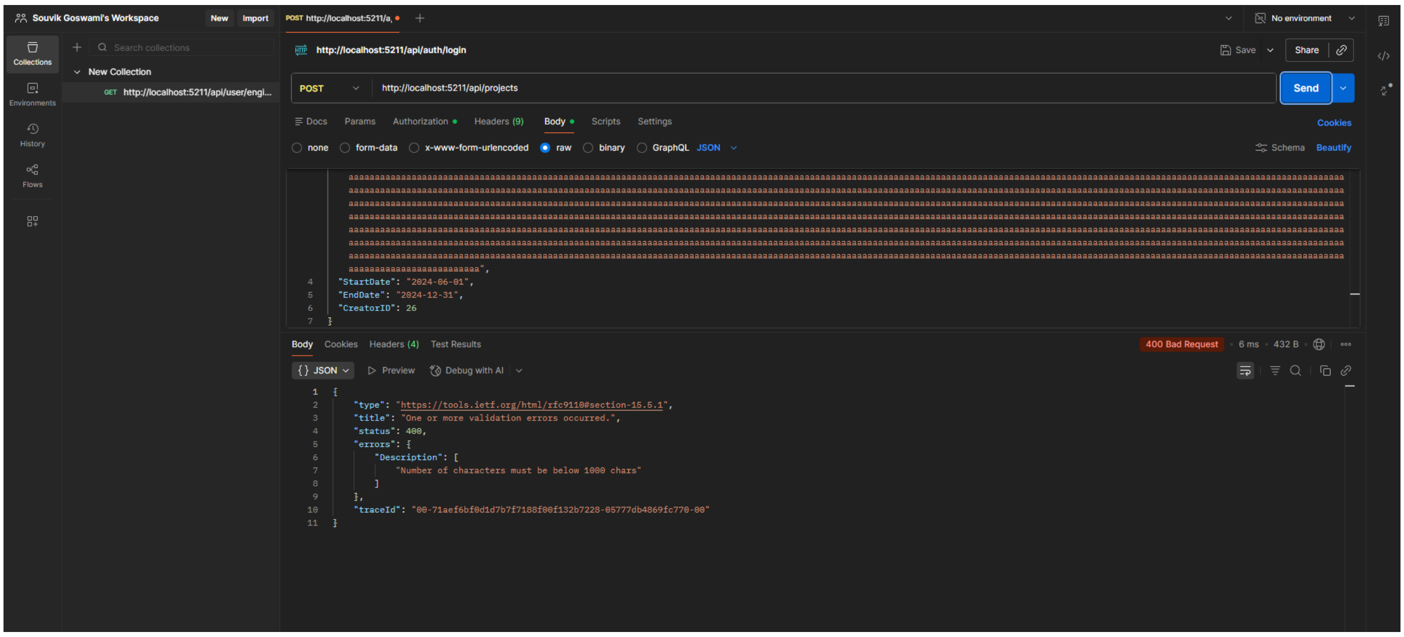
400 Bad Request · 20 ms · 525 B

```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "FullName": [
            "Number of characters must be between 2- 100 chars"
        ],
        "Password": [
            "The Password field is required."
        ],
        "PhoneNumber": [
            "Invalid PhoneNumber Format"
        ]
    },
    "traceId": "00-f18b63bab803c54dc1aabb9317d80999-47875e877dc0b4db-00"
}
```

Test Case 10: Description Length Validation

```
POST /api/projects
Content-Type: application/json

{
  "projectName": "Test",
  "description": "[string with 1001 characters]",
  "startDate": "2024-06-01",
  "endDate": "2024-12-31",
  "creatorID": 1
}
```


Souvik Goswami's Workspace

New Collection — GET http://localhost:5211/api/user/engi...

http://localhost:5211/api/auth/login

POST http://localhost:5211/api/projects

Body: raw JSON

```
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
    aaaaaaaaaaaaaaaaaaaaaaaaa",
  "StartDate": "2024-06-01",
  "EndDate": "2024-12-31",
  "CreatorID": 26
}
```

400 Bad Request · 6 ms · 432 B

```
{
    "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
    "title": "One or more validation errors occurred.",
    "status": 400,
    "errors": {
        "Description": [
            "Number of characters must be below 1000 chars"
        ]
    },
    "traceId": "00-71aef6bf0d1d7b7f7188f00f132b7228-05777db4869fc770-00"
}
```