Activity 2 Document

Title of Activity

○ Notes API + UI

○ Short description (what the app does)

-The Notes App allows users to register with their full name, email, and password, then log in securely using a unique email and strong password. On the frontend, users can create, edit, update, and delete notes through a simple and user-friendly dashboard. On the backend API, the app handles the same CRUD operations with added security, ensuring all data is properly stored in the database.

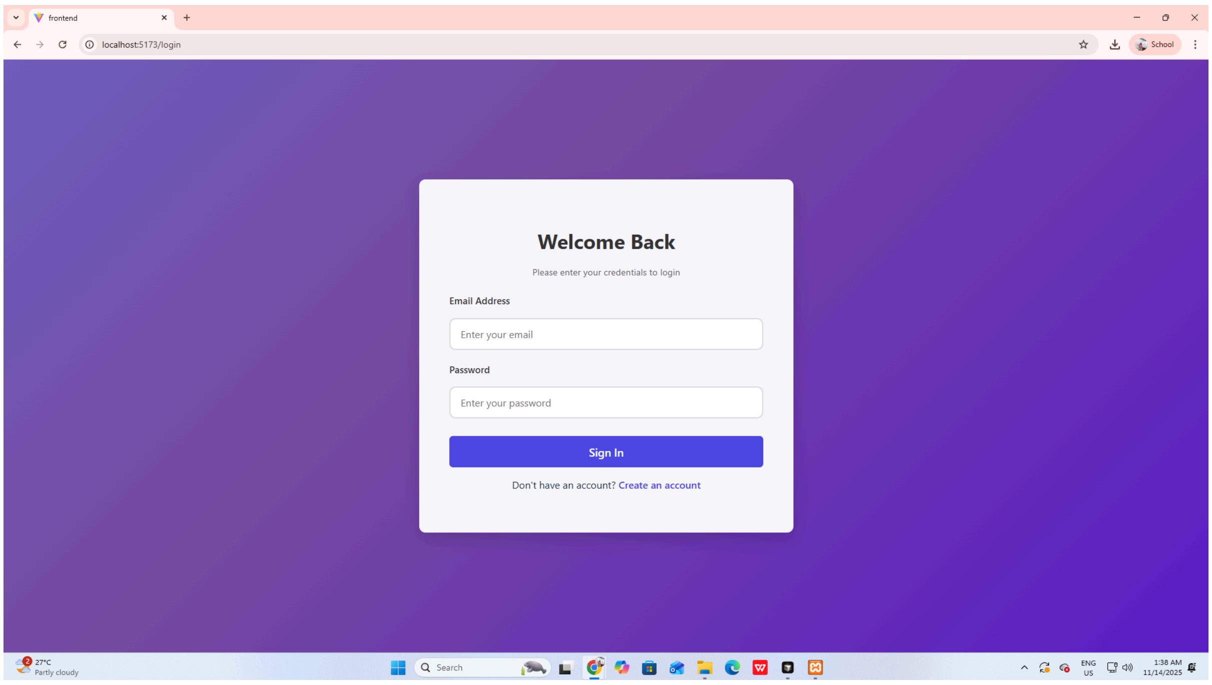
Features:

- Login that you can add your email and password
- Register that you can add your email ,password and confirm password
- Dashboard that you can add,edit,update and delete your notes and save directly to database and also have a CRUD API, that can you also test in swagger docs user cursor ai

○ Screenshot(s) of working system (UI + API example)

Woking System UI
Frontend /Backend

- **Front End**



frontend

localhost:5173/login

Welcome Back

Please enter your credentials to login

Email Address

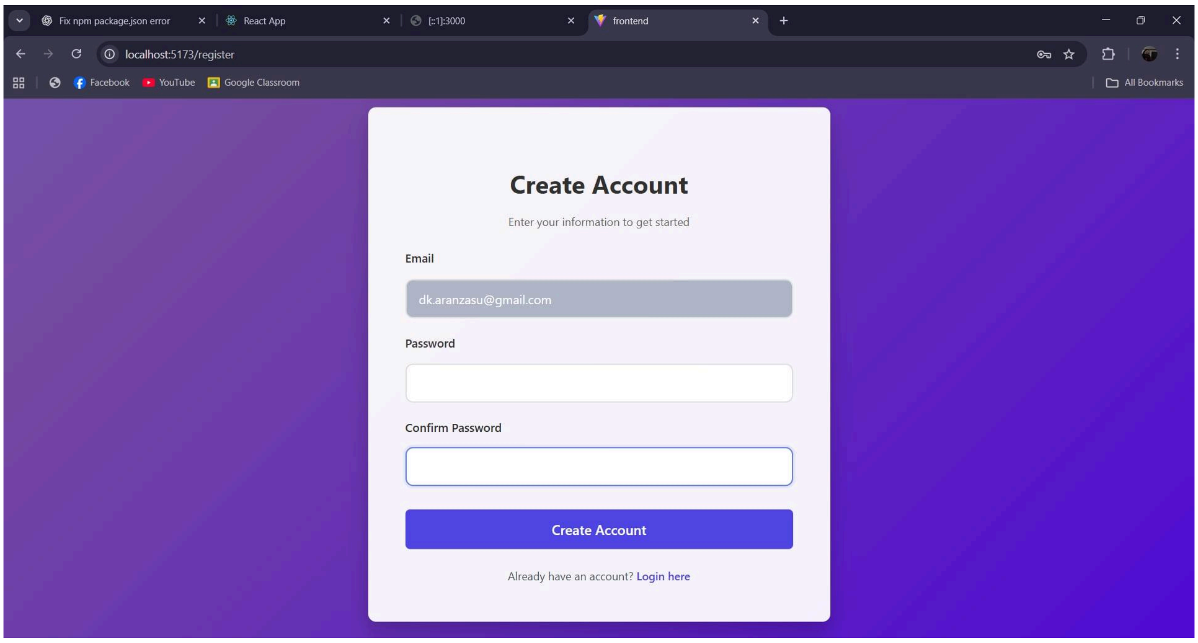
Enter your email

Password

Enter your password

Sign In

Don't have an account? Create an account

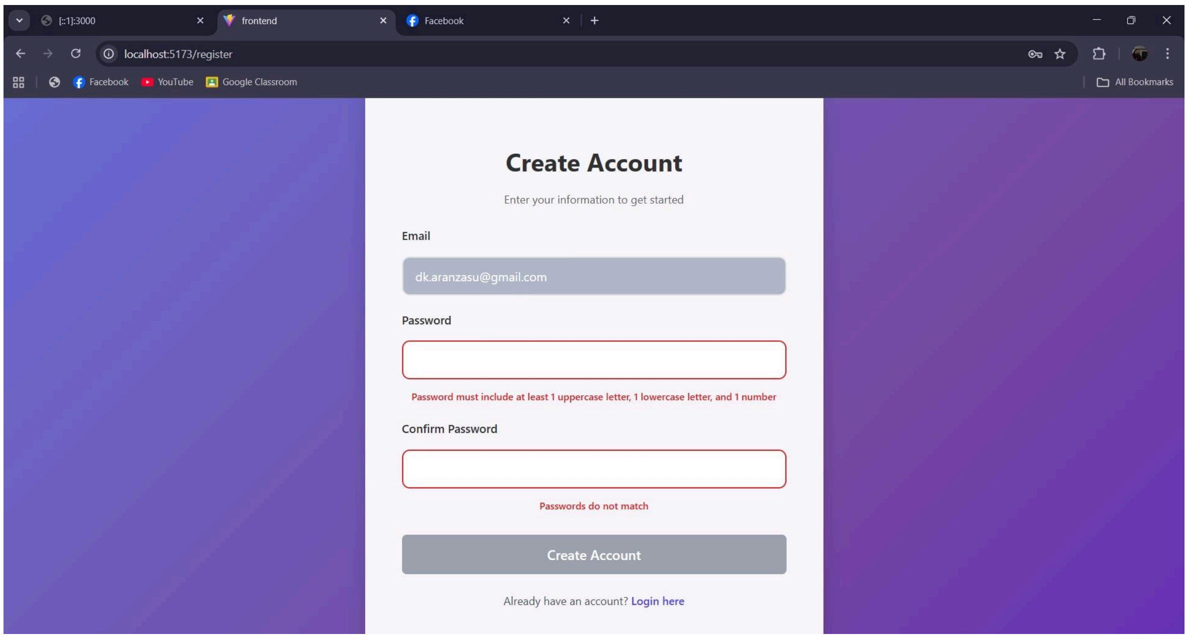

Create Account

Enter your information to get started

Email: dk.aranzasu@gmail.com

Password:

Confirm Password:


Create Account

Enter your information to get started

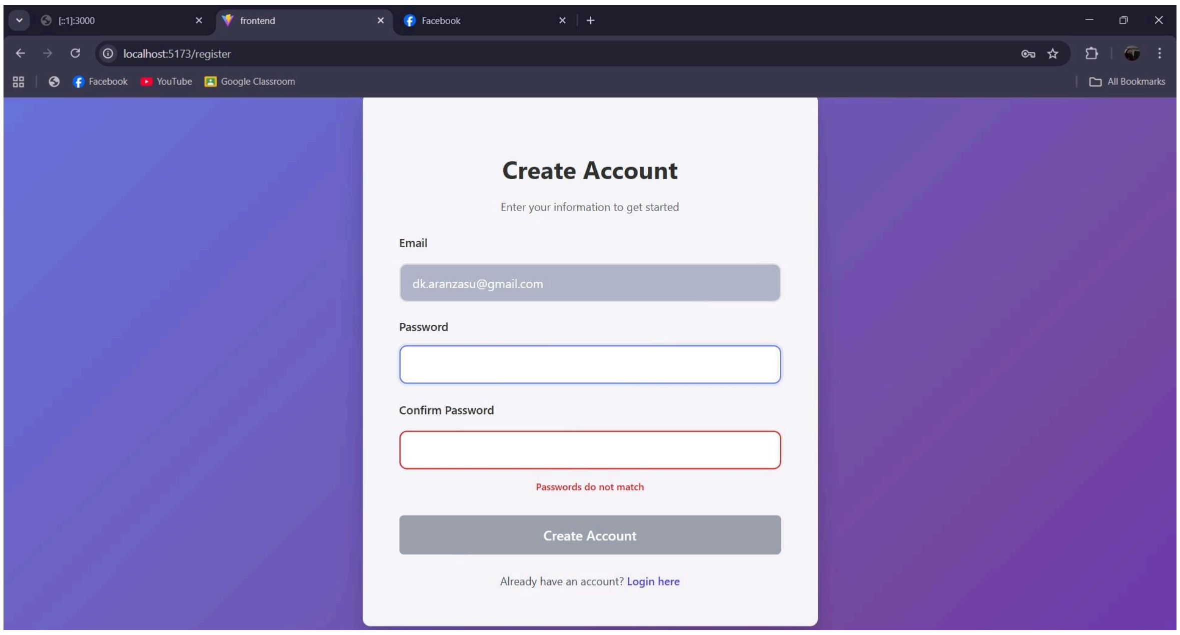
Email: dk.aranzasu@gmail.com

Password:

Password must include at least 1 uppercase letter, 1 lowercase letter, and 1 number

Confirm Password:

Passwords do not match



[::1]:3000 | frontend | Facebook

localhost:5173/register

Facebook | YouTube | Google Classroom | All Bookmarks

Create Account

Enter your information to get started

Email

dk.aranzasu@gmail.com

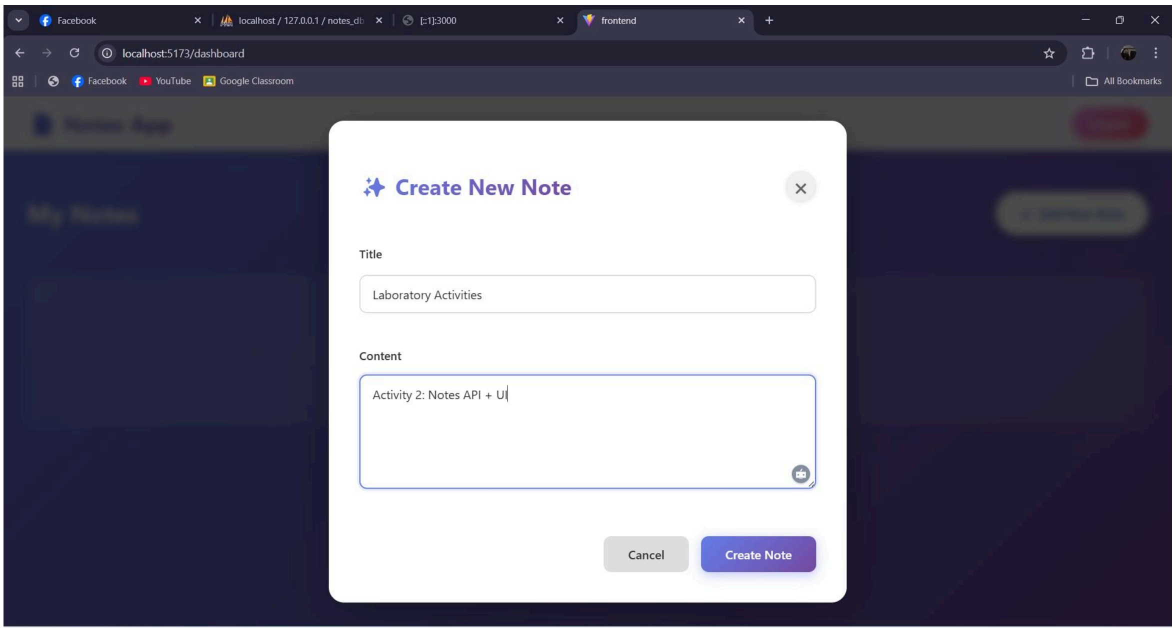
Password

Confirm Password

Passwords do not match

Create Account

Already have an account? Login here



Facebook | localhost / 127.0.0.1 / notes_db | [::1]:3000 | frontend

localhost:5173/dashboard

Facebook | YouTube | Google Classroom | All Bookmarks

Create New Note

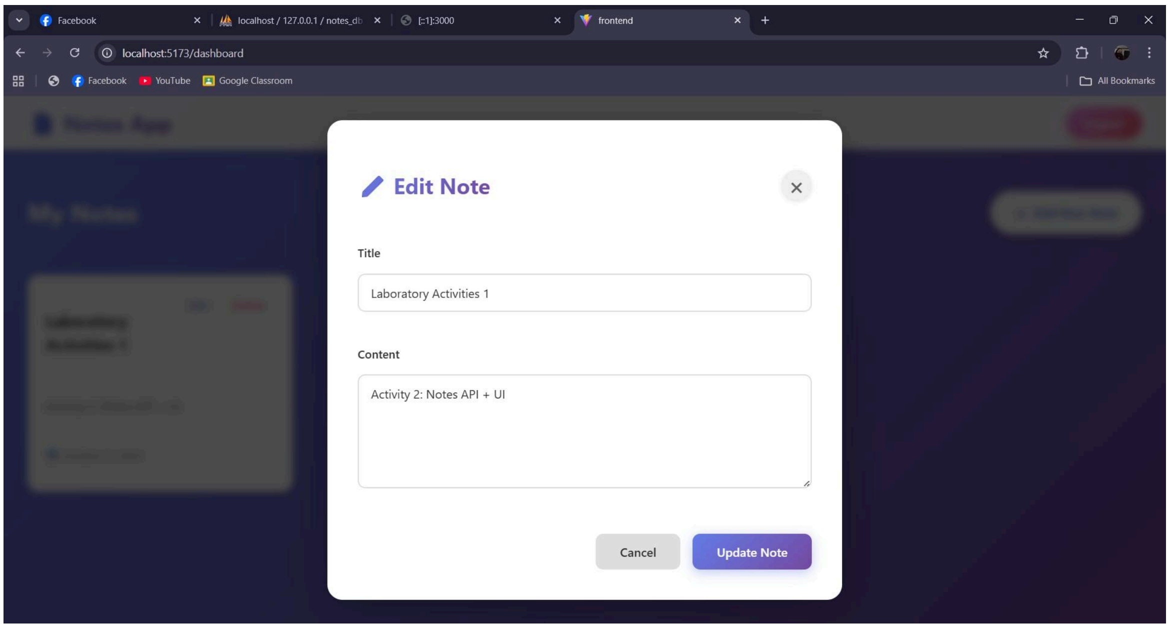
Title

Laboratory Activities

Content

Activity 2: Notes API + UI

Cancel | Create Note



Facebook | localhost / 127.0.0.1 / notes_db | [::1]:3000 | frontend

localhost:5173/dashboard

Facebook | YouTube | Google Classroom | All Bookmarks

Edit Note

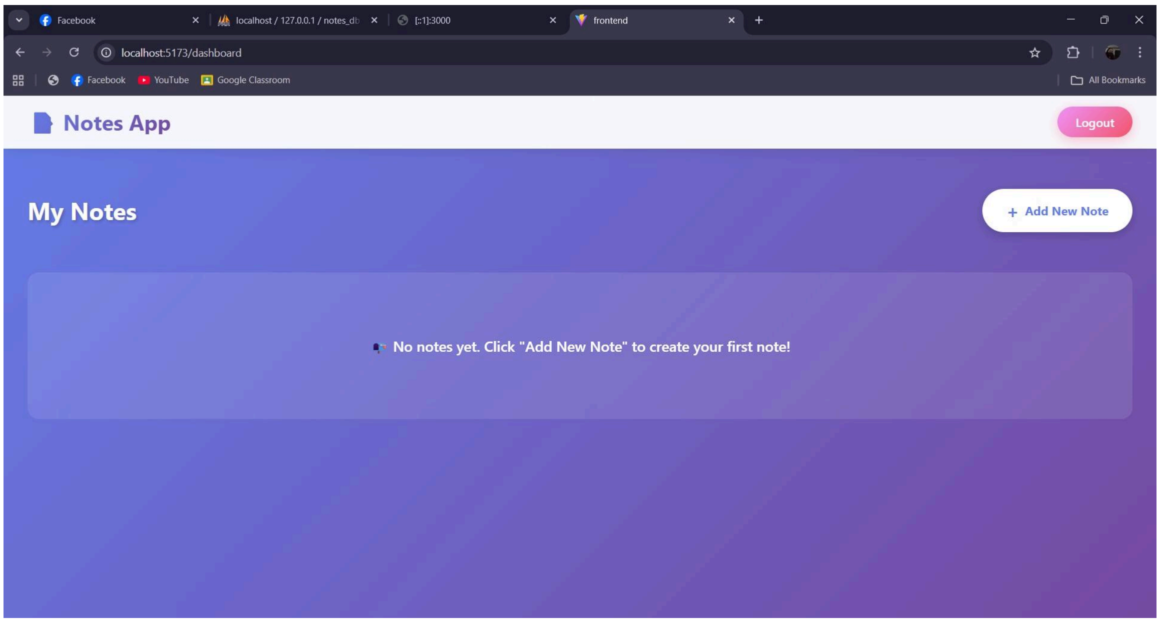
Title

Laboratory Activities 1

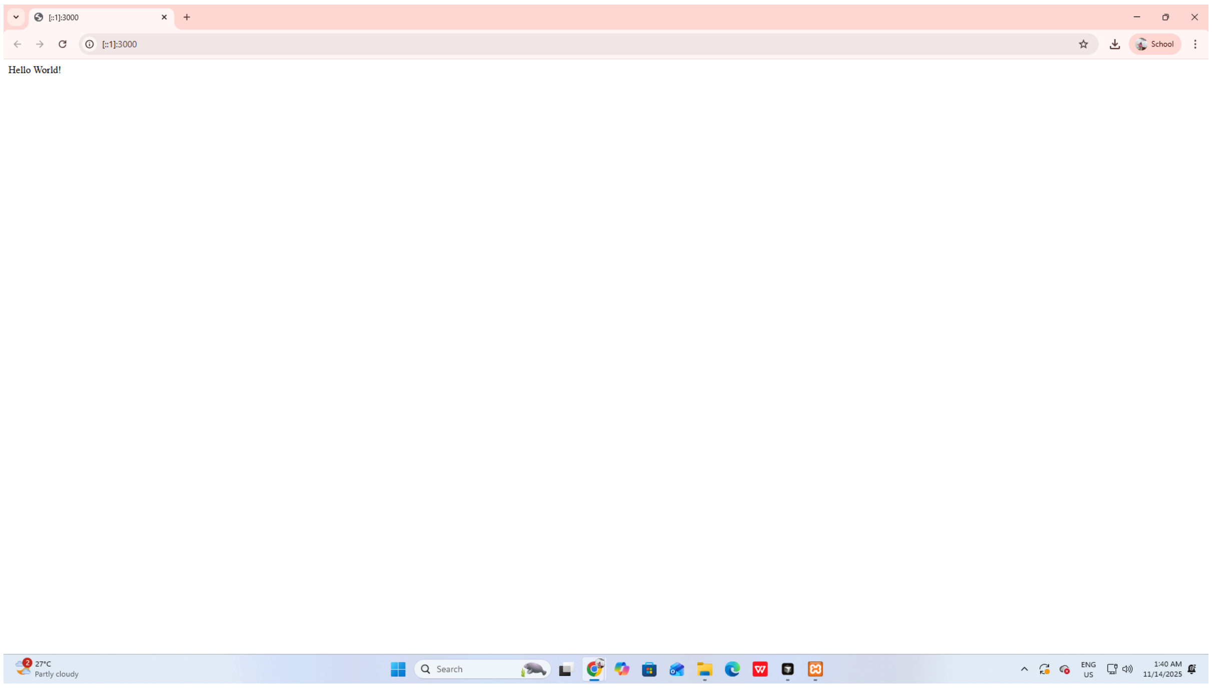
Content

Activity 2: Notes API + UI

Cancel | Update Note

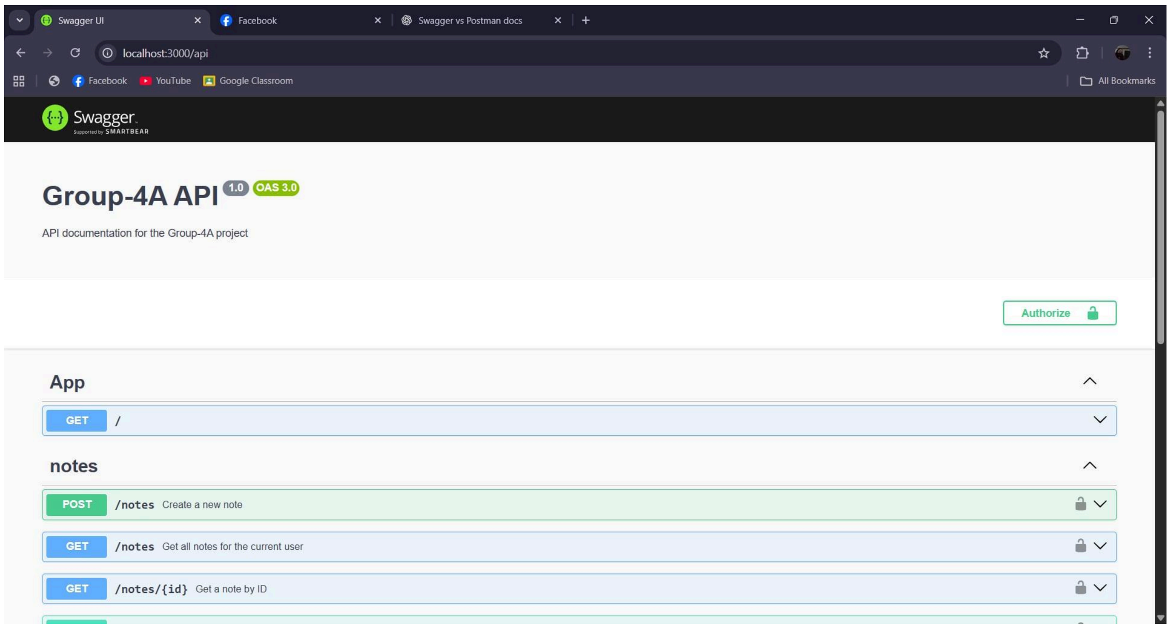


Backend



Hello World!

API



Group-4A API 1.0 OAS 3.0

API documentation for the Group-4A project

App

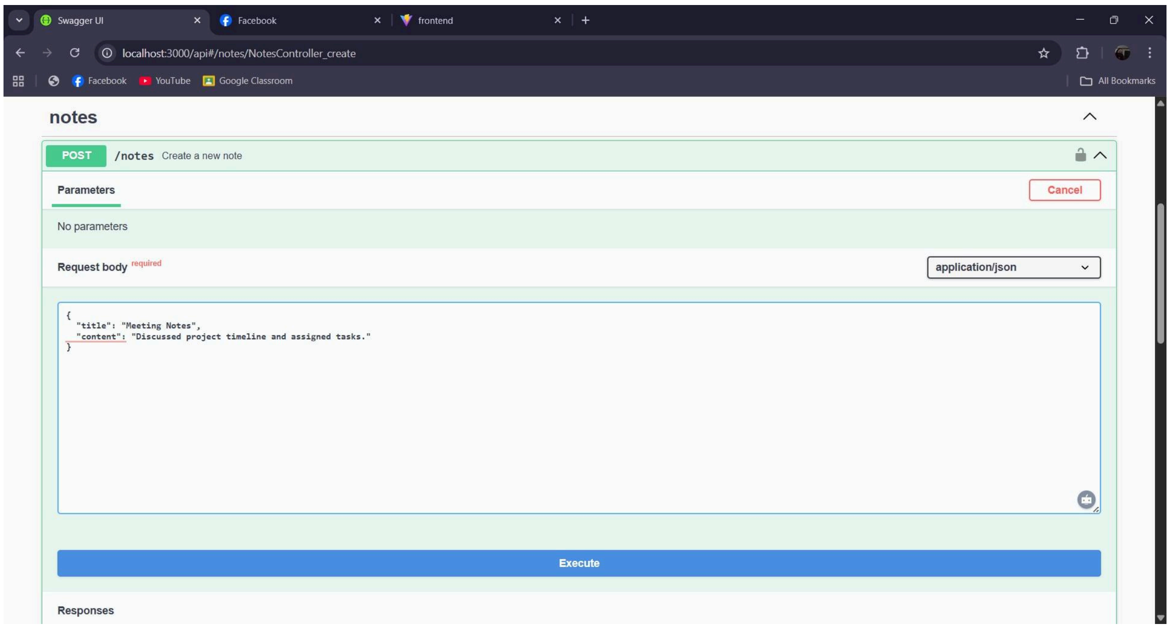
GET /

notes

POST `/notes` Create a new note

GET `/notes` Get all notes for the current user

GET `/notes/{id}` Get a note by ID



notes

POST `/notes` Create a new note

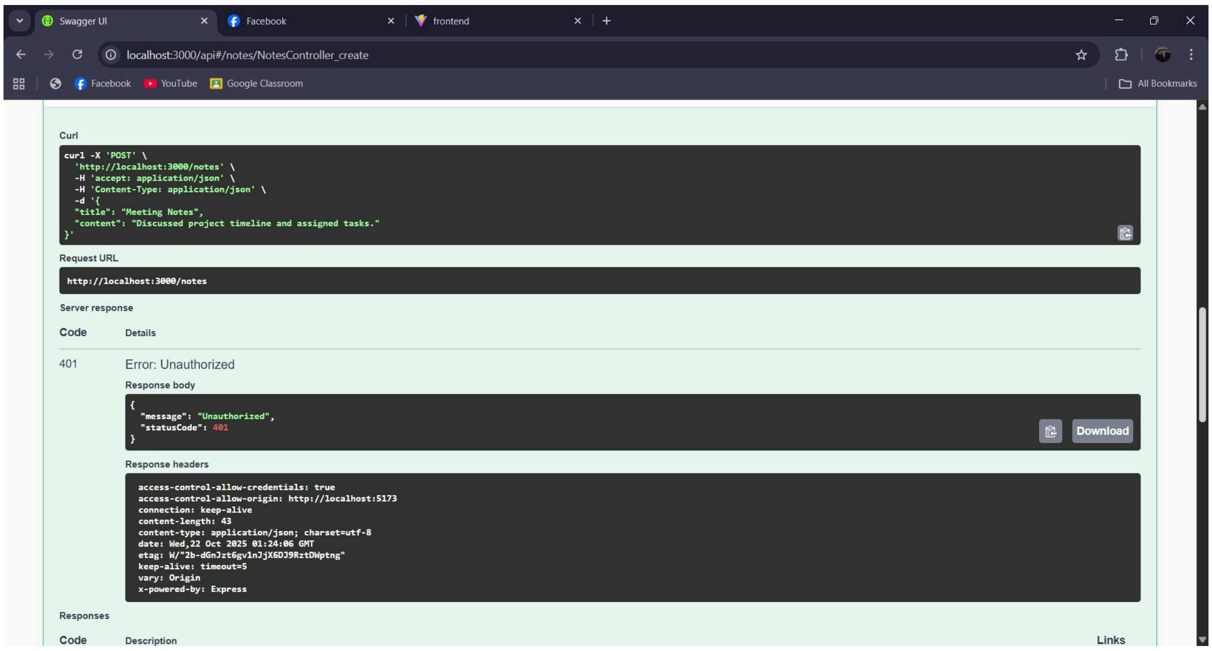
Parameters

No parameters

Request body required — application/json

```
{
  "title": "Meeting Notes",
  "content": "Discussed project timeline and assigned tasks."
}
```

Responses



Curl

```
curl -X 'POST' \
  'http://localhost:3000/notes' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Meeting Notes",
  "content": "Discussed project timeline and assigned tasks."
}'
```

Request URL

```
http://localhost:3000/notes
```

Server response

Code Details

401 Error: Unauthorized

Response body

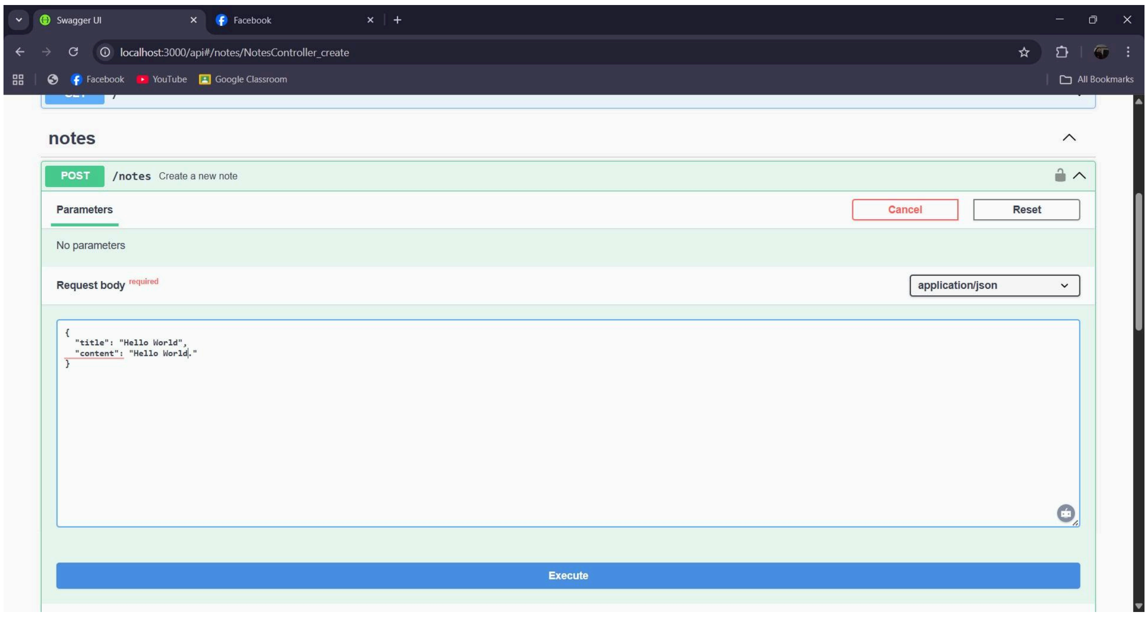
```
{
  "message": "Unauthorized",
  "statusCode": 401
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: http://localhost:5173
connection: keep-alive
content-length: 43
content-type: application/json; charset=utf-8
date: Wed,22 Oct 2025 01:24:06 GMT
etag: W/"2b-dGnJzt6gv1nJjX6DJ9RztDWptng"
keep-alive: timeout=5
vary: Origin
x-powered-by: Express
```

Responses

Code Description Links



notes

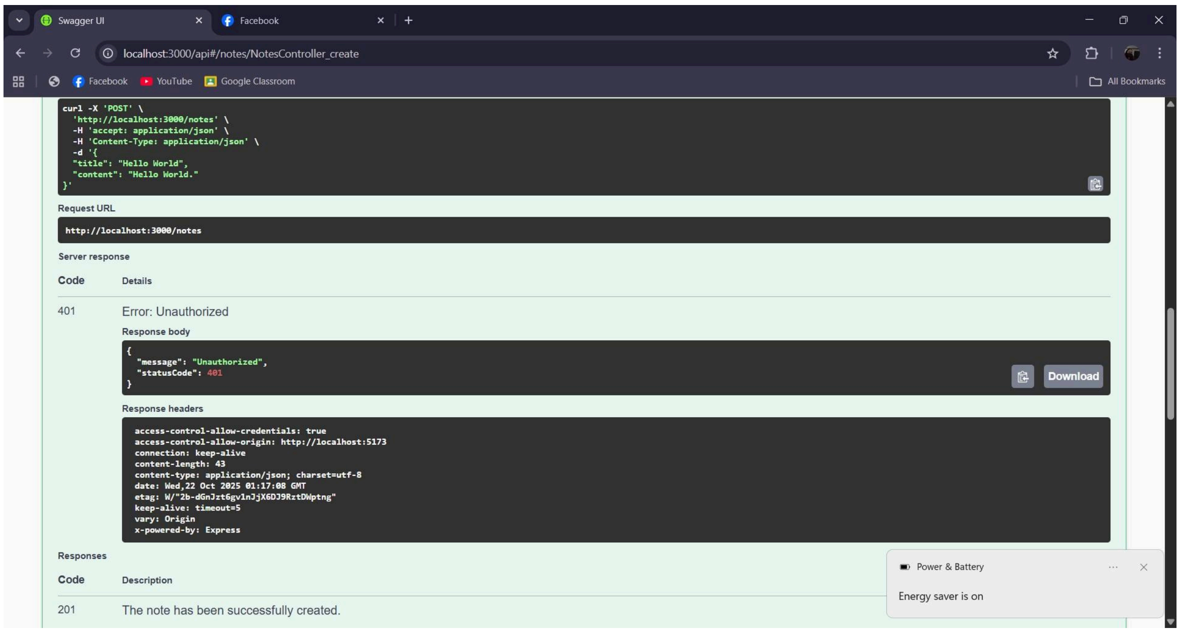
POST /notes Create a new note

Parameters

No parameters

Request body required

```
{
  "title": "Hello World",
  "content": "Hello World."
}
```



```
curl -X 'POST' \
  'http://localhost:3000/notes' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Hello World",
  "content": "Hello World."
}'
```

Request URL

```
http://localhost:3000/notes
```

Server response

Code Details

401 Error: Unauthorized

Response body

```
{
  "message": "Unauthorized",
  "statusCode": 401
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: http://localhost:5173
connection: keep-alive
content-length: 43
content-type: application/json; charset=utf-8
date: Wed,22 Oct 2025 01:17:08 GMT
etag: W/"2b-dGnJzt6gv1nJjX6DJ9RztDWptng"
keep-alive: timeout=5
vary: Origin
x-powered-by: Express
```

Responses

Code Description

201 The note has been successfully created.

○

Instructions on how to run the project

Activity 2 Requirements

- Node.js (version 16 or higher)

-React (Typescripts & Javascripts)

- npm (Node Package Manager)

- Git (for cloning the repository)

Step-by-Step Instructions

1. Clone the Repository

- Open Visual Studio Code.

- Open the terminal

- Run the following command to clone the GitHub repository:

```
git clone https://github.com/CVSU-IMUS-BSIT-4A/Group-4A-Repository.git
```

- Navigate to the project folder:

```
cd C:\Users\demro\OneDrive\Desktop\Group-4A-Repository-1
```

2. Install Backend Dependencies

- Go to the backend directory:

- cd C:\Users\demro\OneDrive\Desktop\Group-4A-Repository-1\ACTIVITY-2\backend

Install all required dependencies:

```
npm install
```

3. Set Up the Database

Make sure XAMPP is running (Apache and MySQL).

Open phpMyAdmin at:
 http://localhost/phpmyadmin

Create a new database (e.g., notes_db).

Once configured properly, the backend will automatically create the necessary tables (for email, password, etc.) in your database.

4. Start the Backend Server

If your package.json includes a start script (e.g., start:dev), run the backend server using

```
npm run start:dev
```

After it starts successfully, the backend should be running at:
http://localhost:3000

You can check the API documentation or routes by visiting:
http://localhost:3000/api

5. Install Frontend Dependencies

Open a new terminal in Visual Studio Code.

Navigate to the frontend directory:

```
cd C:\Users\demro\OneDrive\Desktop\Group-4A-Repository-1\ACTIVITY-2\frontend
```

Install the required dependencies:

```
npm install
```

6. Start the Frontend Server

In the same terminal, run:

```
npm run dev
```

After it starts, open your browser and go to:

http://localhost:5173

7. Access the Application

Frontend Application: http://localhost:5173

Backend API: http://localhost:3000

Swagger or API Docs (if available): http://localhost:3000/api