Node.js CRUD Application with MongoDB
Node Task - 01

Task Overview

This task is a simple CRUD (Create, Read, Update, Delete) application built using Node.js, Express.js, and MongoDB. This task aimed to demonstrate backend development skills, including setting up an Express server, connecting to a MongoDB database, defining a data schema, and handling API requests.

Technologies Used

- **Node.js** – JavaScript runtime environment
- **Express.js** – Web framework for Node.js
- **MongoDB** – NoSQL database
- **Mongoose** – ODM (Object Data Modeling) library for MongoDB
- **Postman** – API testing tool
- **Git & GitHub** – Version control and code hosting

Product Schema

A product is represented with the following schema:

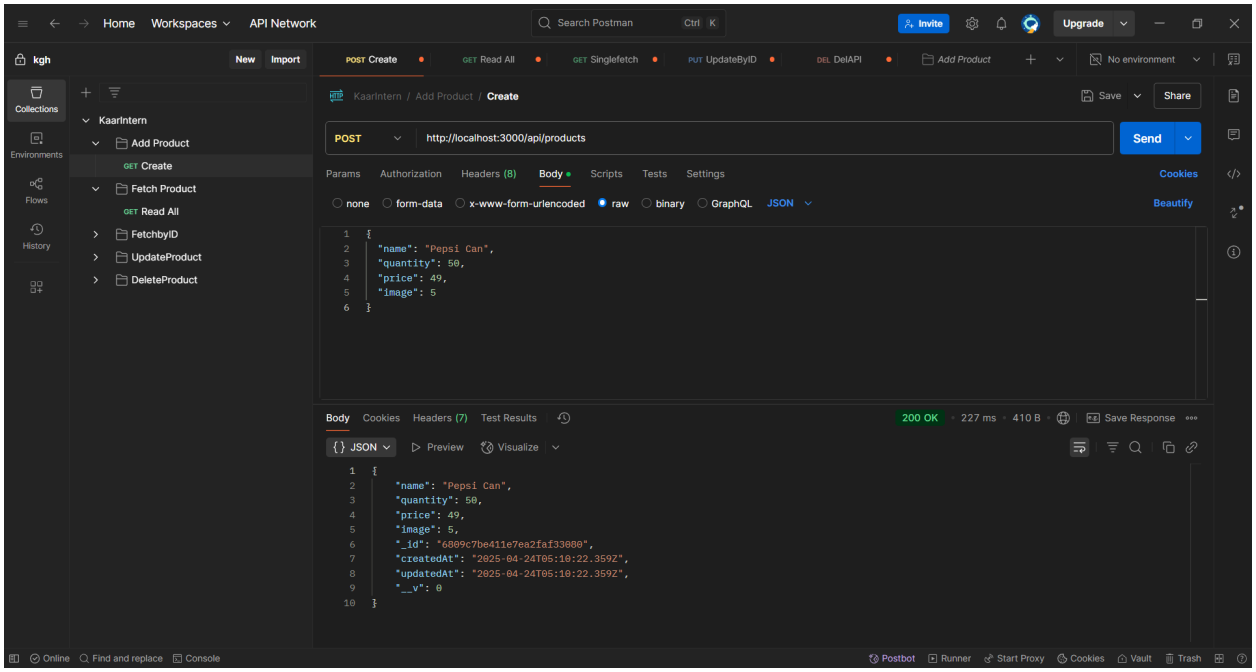
```
{
  name: String,         // Required
  quantity: Number,     // Required, default: 0
  price: Number,        // Required, default: 0
  image: Number         // Optional, default: 0
}
```

API Endpoints

`GET /api/products`

Retrieve all products.

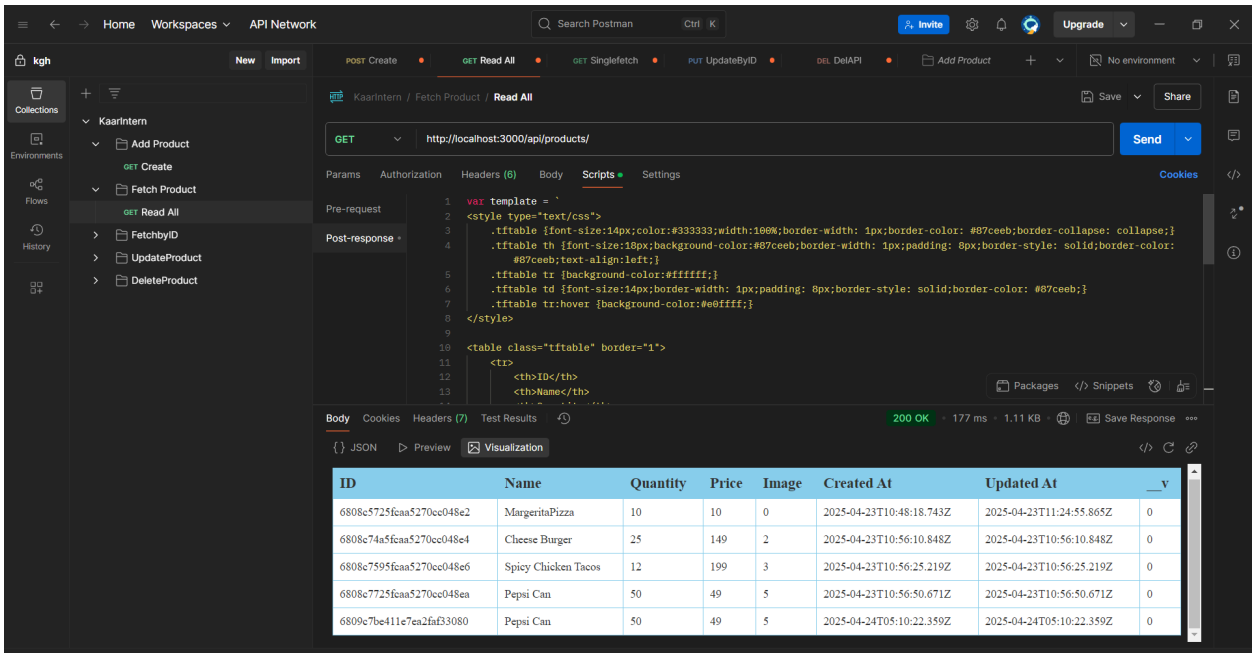
📸 *Screenshot:*



`GET /api/products/:id`

Retrieve a product by ID.

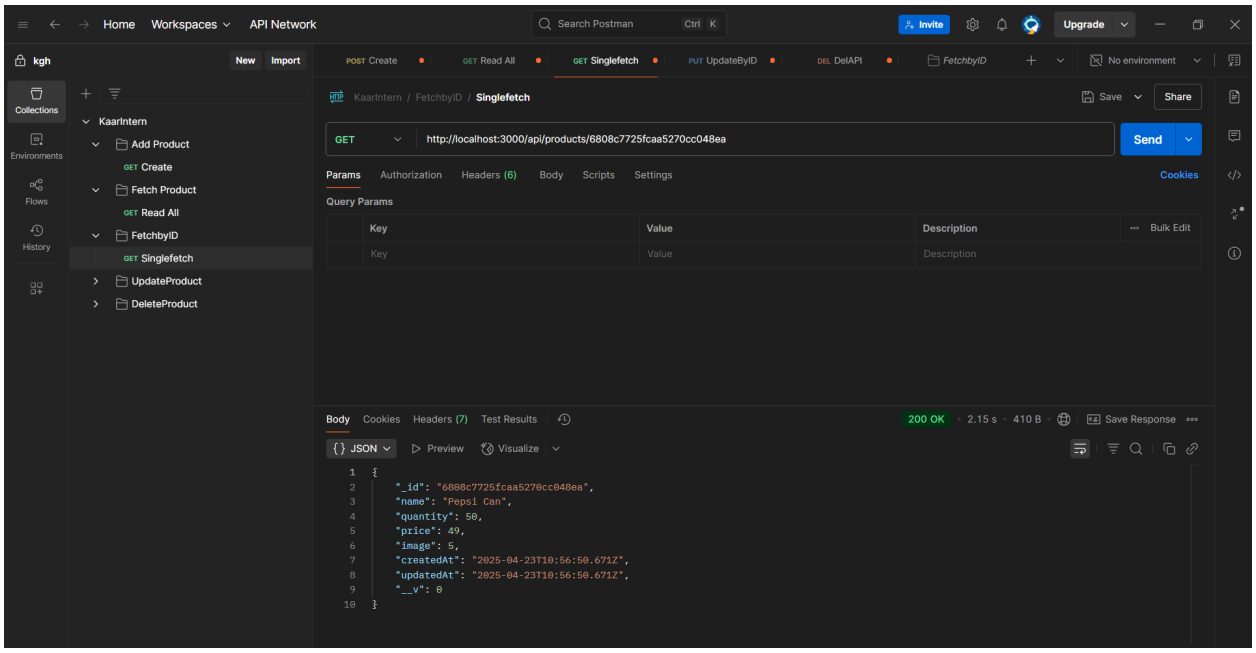
📸 *Screenshot:*



`POST /api/products`

Create a new product.

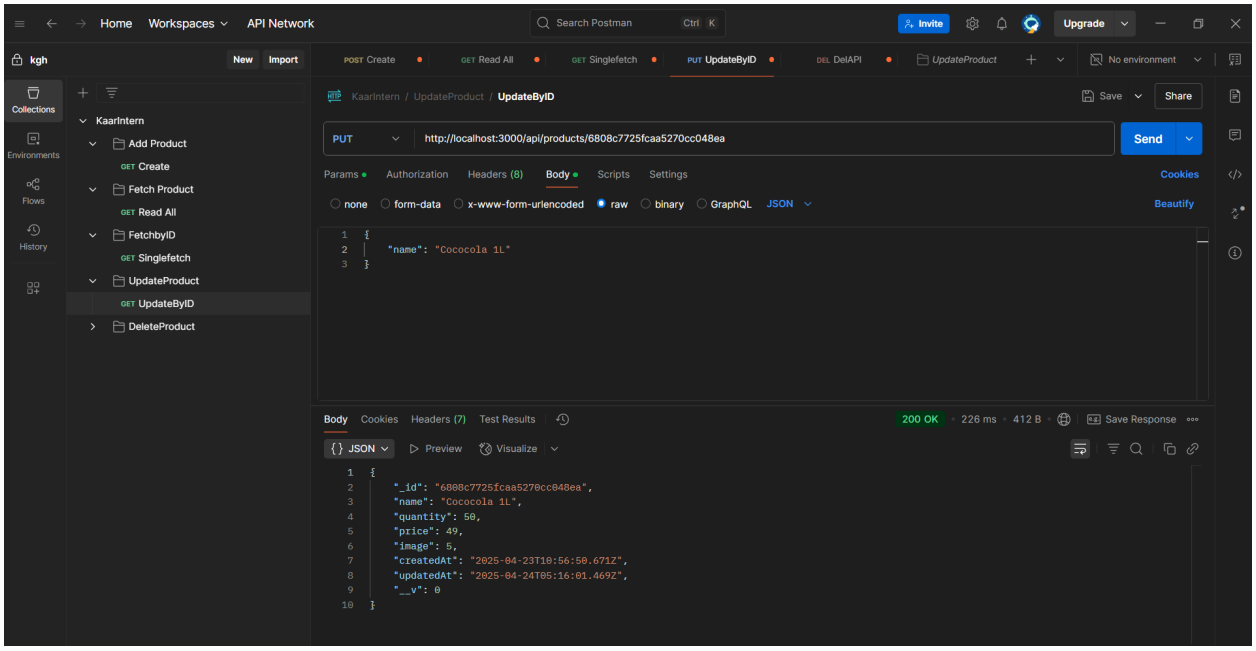
📸 *Screenshot:*



`PUT /api/products/:id`

Update a product by ID.

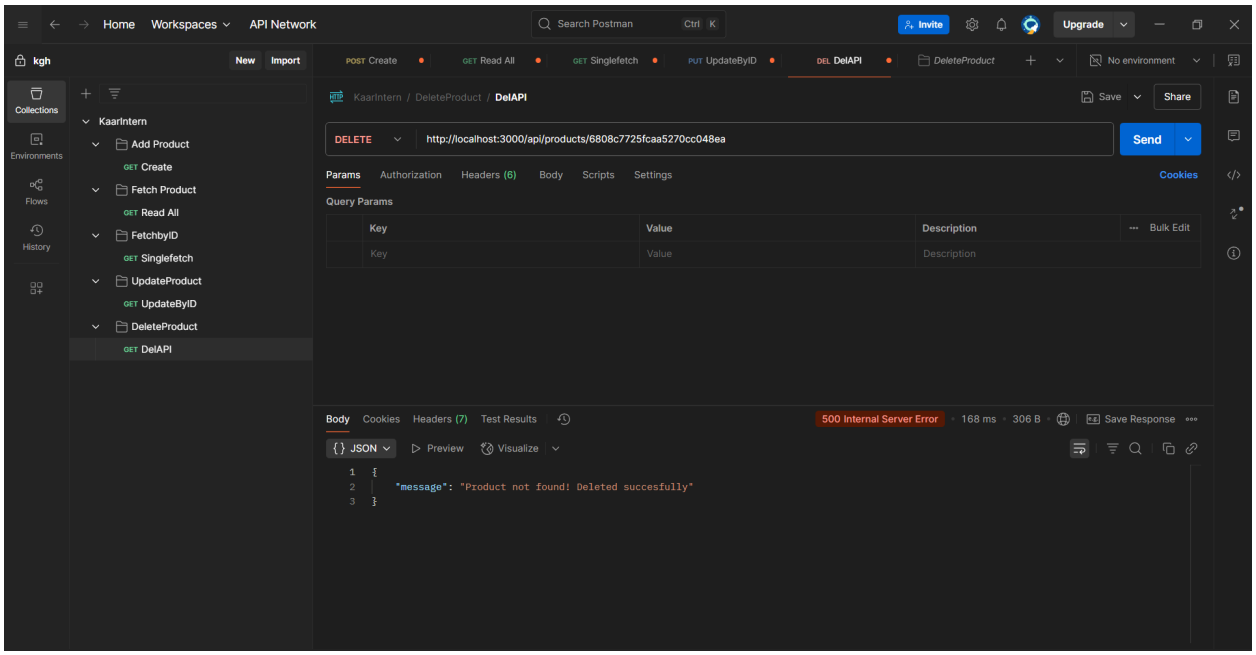
📸 *Screenshot:*



`DELETE /api/products/:id`

Delete a product by ID.

📸 *Screenshot:*



Version Control

- Git was initialized.
- Commits were made at major development milestones.
- Project pushed to GitHub for collaboration and review.

🔗 **GitHub Repository:**
https://github.com/SuryaCreatX/Kaar_NodeLearn.git