|  |  |
| --- | --- |
|  |  |
| Backend Note Taking App CRUD API |  |
|  |  |
|  | Parthraj Panchal DATE :21st March,2024 |
|  |  |

# Index

**Project Structure 3**  
Controllers **3**

Middleware **3**

Repository **3**

Routes **3**

.env **4**

Utility **4**

API Collection.json **5**

DatabaseScript.sql **6**

**APIs 7**

Get all notes **7**

Get note by id **8**

Create Note **9**

Update note **10**

Delete note **11**

**Special note 12**

### Project Structure

**For developing backend API project, I have followed a particular folder structure to keep it clean and organized.**

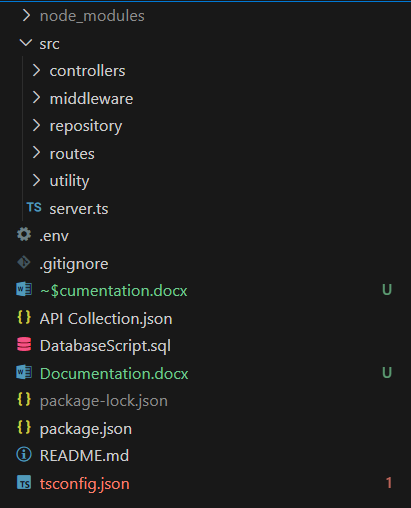


Figure 1: Project folder structure

1. **Controllers  
    Controllers in Express.js projects are modules containing encapsulating request processing logic for better code organization and maintainability.  
    In this project, we only have ‘NoteController.ts’ which handles logic for note’s CRUD operations.**
2. **Middleware  
    Middleware in Express.js intercepts and modifies incoming requests or outgoing responses, enabling functionalities like logging, authentication, and error handling, enhancing the flexibility and extensibility of web applications.  
    In this project, we have implemented ‘RequestLoggerMiddleware.ts’ which logs all incoming requests with data.**
3. **Repository  
    In Express.js projects, repositories typically encapsulate data access logic, providing an abstraction layer for interacting with databases or other data sources, promoting separation of concerns and maintainability in applications.  
    In this project, we have created ‘NoteRepository.ts’ which consist helper methods for to interact with database.**
4. **Routes  
    In an Express.js project, routes define the URL endpoints that clients can request, and they map these endpoints to specific controller functions, facilitating the organization and handling of incoming requests in the application.  
    In this project, we have used ‘NoteRoute.ts’ which handles and redirect the incoming HTTP requests to specific controller method.**
5. **Utility  
   This folder contains specific utility items such as** 
   * **ResponseModel (for ensuring specific response format to user)  
     This model contains three fields**
     + **isSuccess -> shows status of operation**
     + **message -> specific message from backend**
     + **responseData-> data of interest**
   * **Logger (for logging important logs to file and console)**
     + **Example:**

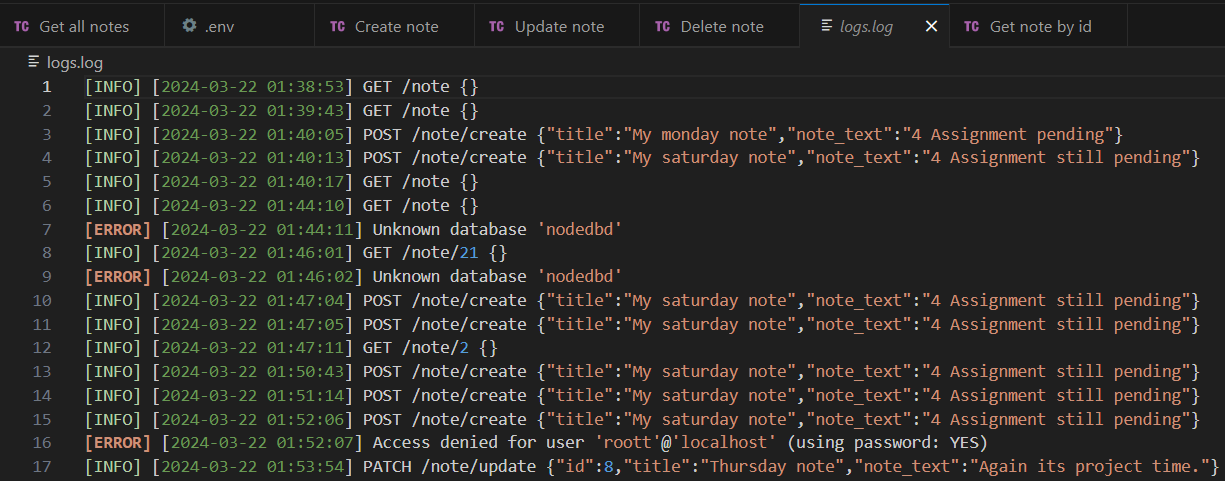


Figure 2: Application logs

* + **DatabaseConstants (for storing all the SQL query string)**
    - **Query to select all notes**
    - **Query to select note based on note id**
    - **Query to create new note**
    - **Query to update existing note**
    - **Query to delete note**

1. **.env  
   This file contains the configurations elements of project. In our project it is specifically used for database credentials and information.**

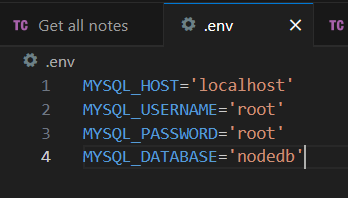


Figure 3: .env file content

1. **API Collection.json  
    This file is an import file for API client such as postman, thunderclient, etc. It contains all the API request of this project to test.**

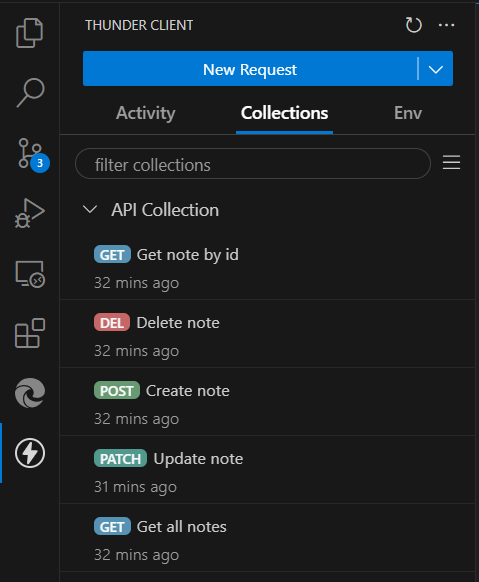


Figure 4: API collection

1. **DatabaseScript.sql  
    This SQL script file contains the SQL commands to setup database for this project. It contains database creation and table creation command with exist check.**

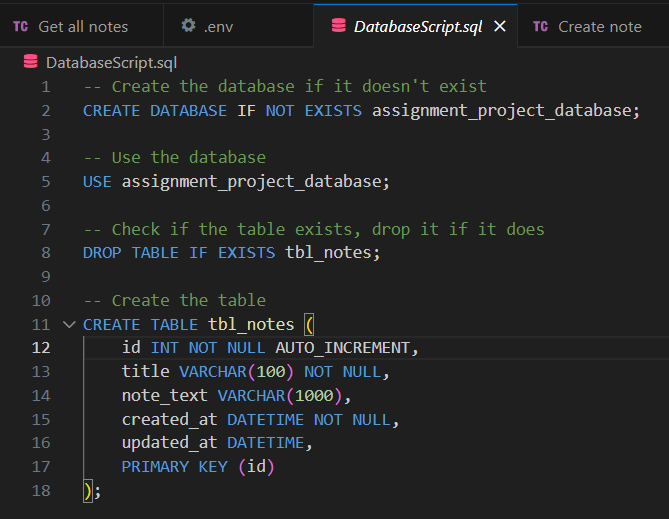


Figure 5: Database script

### APIs

1. **Get all notes**

|  |  |
| --- | --- |
| **Get all notes** | |
| **URL** | **/note** |
| **Request parameter** | **-** |
| **Request method** | **GET** |
| **Request body** | **-** |
| **Response body** | **{**  **"isSuccess": true,**  **"responseData": [**  **{**  **"id": 1,**  **"title": "My monday note",**  **"note\_text": "4 Assignment pending",**  **"created\_at": "2024-03-22T01:40:05.000Z",**  **"updated\_at": "2024-03-22T01:40:05.000Z"**  **}**  **],**  **"message": "All notes fetched."**  **}** |
| **Success test** |  |
| **Fail test** |  |

1. **Get note by id**

|  |  |
| --- | --- |
| **Get note by id** | |
| **URL** | **/note/21** |
| **Request parameter** | **Note id** |
| **Request method** | **GET** |
| **Request body** | **-** |
| **Response body** | **{**  **"isSuccess": true,**  **"responseData": [**  **{**  **"id": 2,**  **"title": "My saturday note",**  **"note\_text": "4 Assignment still pending",**  **"created\_at": "2024-03-22T01:47:05.000Z",**  **"updated\_at": "2024-03-22T01:47:05.000Z"**  **}**  **],**  **"message": "Note fetched successfully."**  **}** |
| **Success test** |  |
| **Fail test** |  |

1. **Create Note**

|  |  |
| --- | --- |
| **Create note** | |
| **URL** | **/note/create** |
| **Request parameter** | **-** |
| **Request method** | **POST** |
| **Request body** | **{**  **"title":"My saturday note",**  **"note\_text":"4 Assignment still pending"**  **}** |
| **Response body** | **{**  **"isSuccess": true,**  **"responseData": [**  **{**  **"id": 4,**  **"title": "My saturday note",**  **"note\_text": "4 Assignment still pending",**  **"created\_at": "2024-03-22T01:51:14.000Z",**  **"updated\_at": "2024-03-22T01:51:14.000Z"**  **}**  **],**  **"message": "Note created successfully."**  **}** |
| **Success test** |  |
| **Fail test** |  |

1. **Update note**

|  |  |
| --- | --- |
| **Update note** | |
| **URL** | **/note/update** |
| **Request parameter** | **-** |
| **Request method** | **PATCH** |
| **Request body** | **{**  **"id":8,**  **"title":"title 1 update 4",**  **"note\_text":"note text it is updated"**  **}** |
| **Response body** | **{**  **"isSuccess": true,**  **"responseData": [**  **{**  **"id": 4,**  **"title": "My saturday note",**  **"note\_text": "4 Assignment still pending",**  **"created\_at": "2024-03-22T01:51:14.000Z",**  **"updated\_at": "2024-03-22T01:51:14.000Z"**  **}**  **],**  **"message": "Note created successfully."**  **}** |
| **Success test** |  |
| **Fail test** |  |

1. **Delete note**

|  |  |
| --- | --- |
| **Delete note** | |
| **URL** | **/note/delete/2** |
| **Request parameter** | **Note id** |
| **Request method** | **DELETE** |
| **Request body** | **-** |
| **Response body** | **{**  **"isSuccess": true,**  **"message": "Note 3 deleted."**  **}** |
| **Success test** |  |
| **Fail test 1** |  |
| **Fail test 2** |  |

### Special note

I have 4.5 years of experience in software development in technology such as .NET, RabbitMQ, Spring boot (started in Dalhousie University course project). Though I did not touched Node.js in past, I tried my best to learn it and apply software development concept which I was aware of (for developing application using .Net and Spring boot). I can still develop the frontend part for this application, but due to time constraints and work load of assignment plus project (specially next week), I will not be able to do that.

Thank you SHIFTKEY !