## **Note API Documentation**

### **MongoDB Setup:**

* The MongoDB database is used to store notes.
* Each note consists of a title, content, and timestamps for creation and last modification.

### **Data Validation:**

* Validation is implemented for note creation and updating.
* The title and content fields must be provided and within appropriate lengths.

### **Error Handling:**

* Comprehensive error handling covers scenarios like invalid input, attempting to access or modify non-existent notes, etc.

.

### **Documentation for Authentication:**

#### 1. Google Authentication:

* Endpoint:
  + GET /auth/google
  + Initiates the Google OAuth2 authentication process.
* Callback Endpoint:
  + GET /auth/google/callback
  + Handles the Google OAuth2 callback.
* Logout Endpoint:
  + GET /auth/logout
  + Logs the user out.

### **Dashboard Routes:**

#### 1. Dashboard:

* Endpoint:
  + GET /dashboard
  + Renders the dashboard page with a list of notes.

#### 2. View Note:

* Endpoint:
  + GET /dashboard/item/:id
  + Renders a specific note for viewing.

#### 3. Update Note:

* Endpoint:
  + PUT /dashboard/item/:id
  + Updates the content of an existing note.

#### 4. Delete Note:

* Endpoint:
  + DELETE /dashboard/item-delete/:id
  + Deletes a note from the database.

#### 5. Add Note:

* Endpoint:
  + GET /dashboard/add
  + Renders the add note page.
* Endpoint:
  + POST /dashboard/add
  + Submits a new note for addition.

#### 6. Search Notes:

* Endpoint:
  + GET /dashboard/search
  + Renders the search page.
* Endpoint:
  + POST /dashboard/search
  + Submits a search query and renders the search results.