

During the development of my website, I faced several technical challenges, but I managed to overcome them through research, debugging, and implementing best practices. Here are some of the key problems and how I resolved them:

1. File Upload Handling:

Problem: Implementing a robust file upload feature was challenging. I had to ensure that files were correctly processed and saved to the server.

Solution: I used `multer`, a middleware for handling `multipart/form-data`, which is primarily used for uploading files. Integrating this with my backend, ensuring correct parsing of file data, and saving it to the appropriate directory were critical steps.

2. User Authentication and Authorization:

Problem: Implementing a secure and reliable user authentication system was essential to ensure that only authenticated users could upload and access notes.

Solution: I integrated an authentication middleware that populates the `req.user` object with user information. This allowed me to associate uploaded files with the correct user and ensure that only authenticated users could perform certain actions.

3. Setting Up Authentication Middleware:

Problem: I faced issues with token extraction in the `auth.js` middleware. The token was not being correctly retrieved from the `Authorization` header due to case sensitivity issues.

Solution: I modified the token extraction logic to handle the `Authorization` header correctly by checking `req.headers['authorization']` in lowercase. This ensured that the token was correctly extracted and processed.

4. Configuring File Uploads:

Problem: Implementing file uploads with `multer` and ensuring the uploaded files were saved to the correct directory was challenging. Additionally, ensuring the middleware authenticated the requests correctly before allowing file uploads was tricky.

Solution: I created a `multer` storage configuration to save files to a specific directory and included the authentication middleware in the upload route. I also added logic to create the destination folder if it didn't exist.

5. Frontend-Backend Integration:

Problem: Integrating the frontend with the backend, especially handling CORS issues and ensuring secure data transmission.

Solution: I used the `cors` middleware in Express to handle cross-origin requests and ensured that the frontend correctly sent authentication tokens in the request headers.

6. Responsive Design:

- **Problem:** Ensuring that the website was responsive across various devices and screen sizes was crucial.
- **Solution:** I used a combination of CSS Grid and Flexbox for layout management, ensuring that components adjusted gracefully across different screen sizes.

7. Error Handling and Debugging:

Problem: Throughout the development process, I faced various bugs and errors that needed to be identified and resolved.

Solution: I implemented comprehensive error handling to catch and log errors at different stages of the file upload process. This helped in diagnosing issues quickly and providing meaningful error messages to the users.

By addressing these challenges systematically and implementing robust solutions, I was able to successfully complete the project and deliver a functional and user-friendly website for college students to share and access study notes.