**Lab Activity 7**

**Objectives**

* To create and design a server to implement MERN Stack Development
* To revise the code relatively

**Materials Needed:**

* A computer with Node.js, MongoDB Compass, and React JS installed.
* IDE: VS Code, Github Codespaces, or Code Sandbox
* Optional: Postman [API Testing]

**Instructions:**

Replicate the code snippet.

Follow and implement the enhancement instructions.

Disregard the **MERNStackDesign.**

Install the following npm packages:

* For backend:
  + bcryptjs
  + cors
  + dotenv
  + express
  + jsonwebtoken
  + mongoose
* For frontend
  + axios
  + dotenv

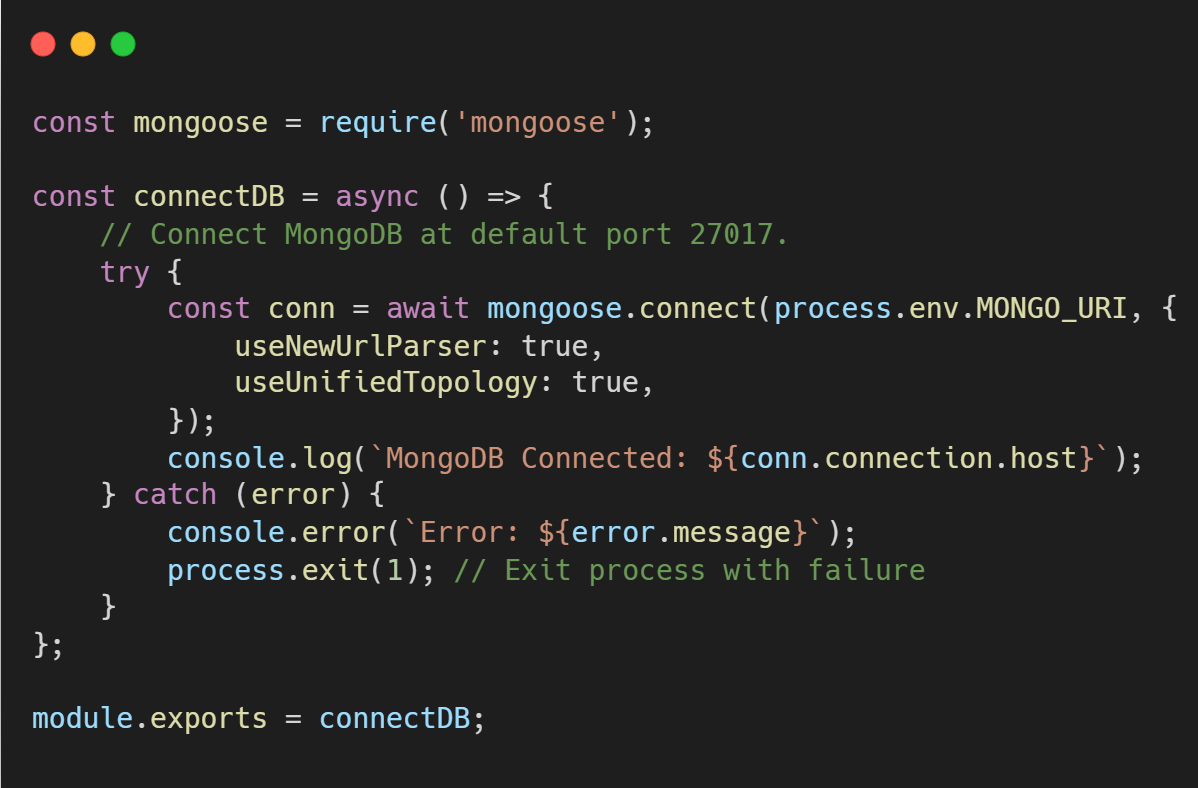
**Web App Initialization:**

|  |  |
| --- | --- |
| Sample design pattern [component based] | |
| Front-end: **surname-front-end** | Back-end: **surname-back-end** |
|  | |

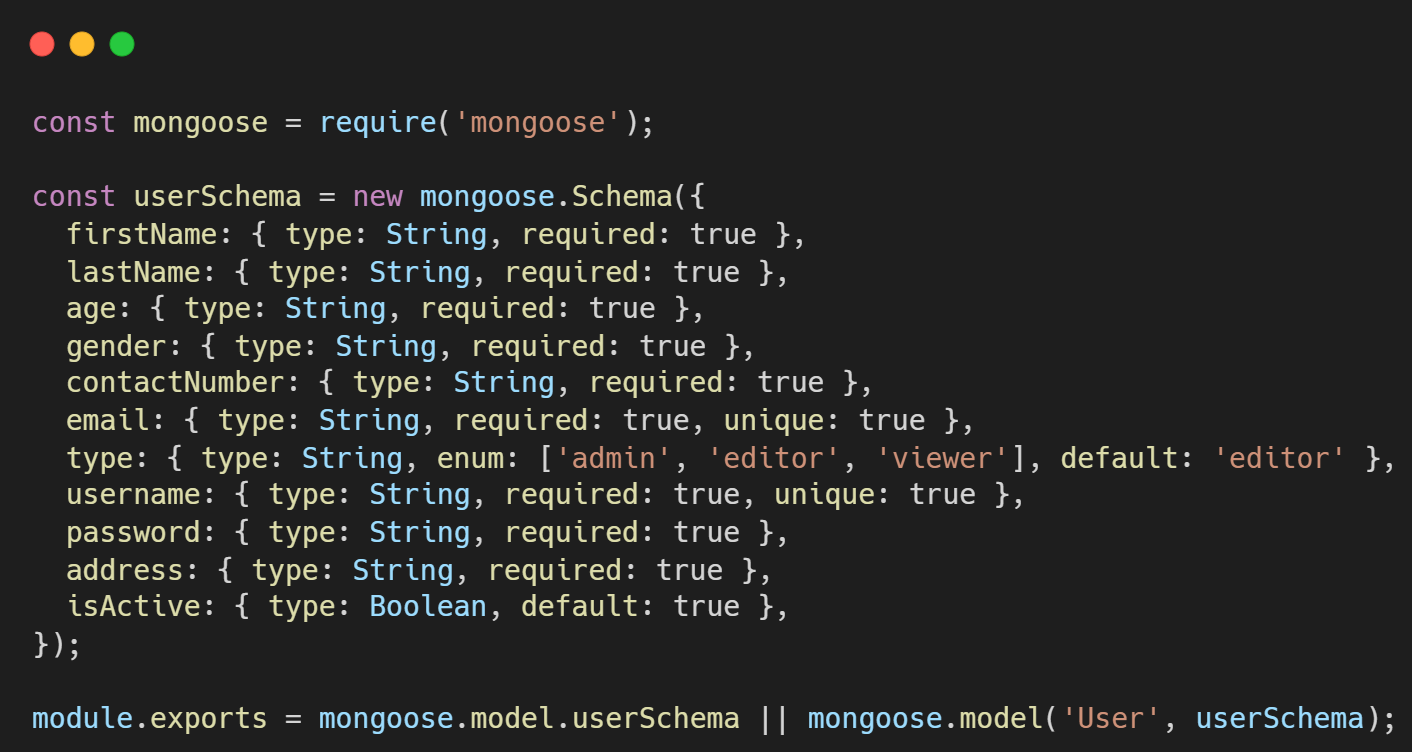
**Code Snippets**

**Back-end**

**db.js**



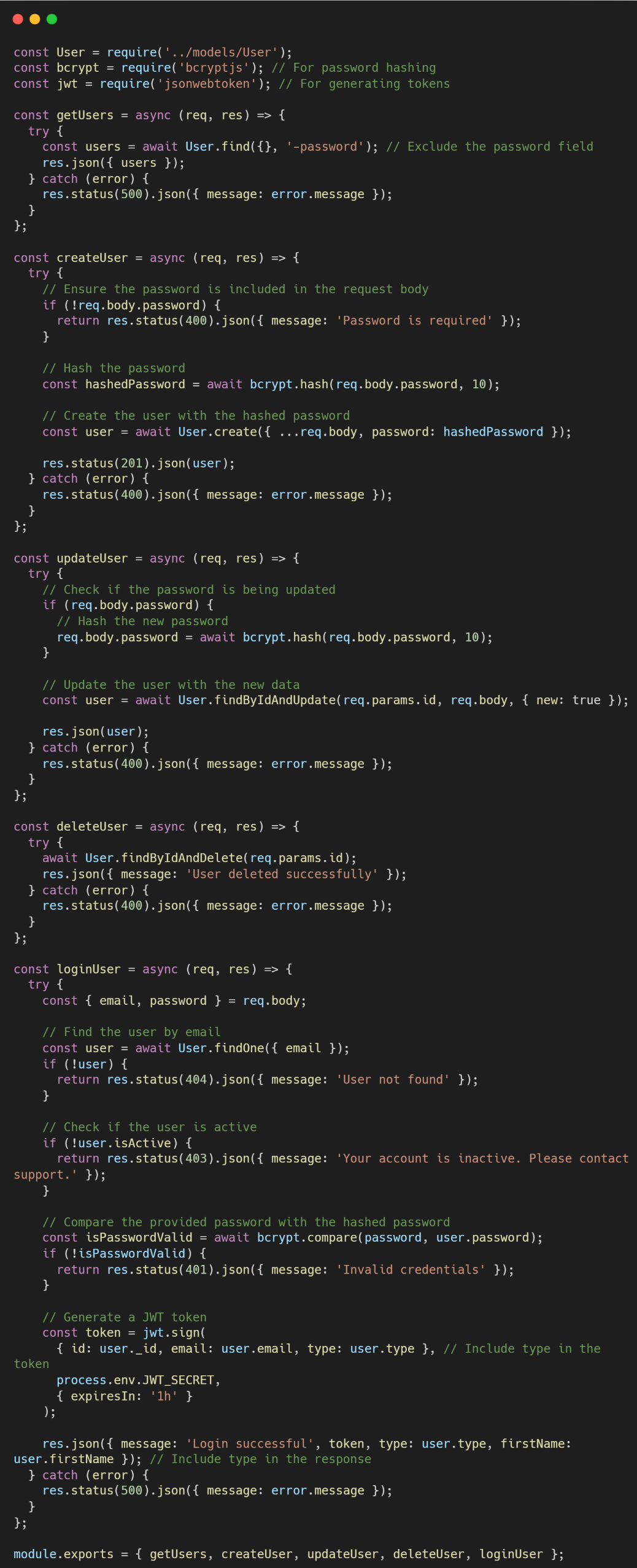
**User.js**



**User.js**



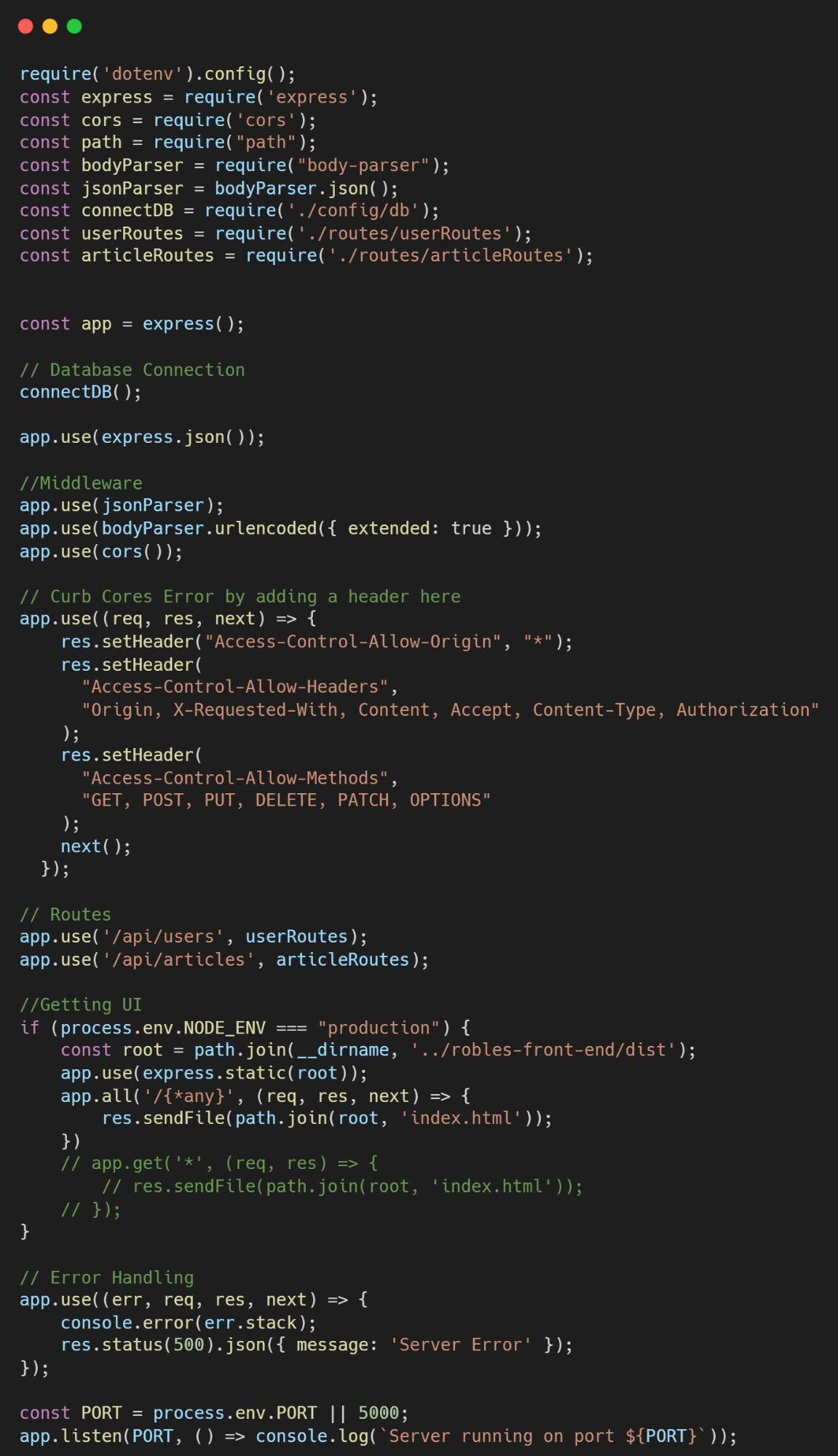
**userController.js**



**.env**

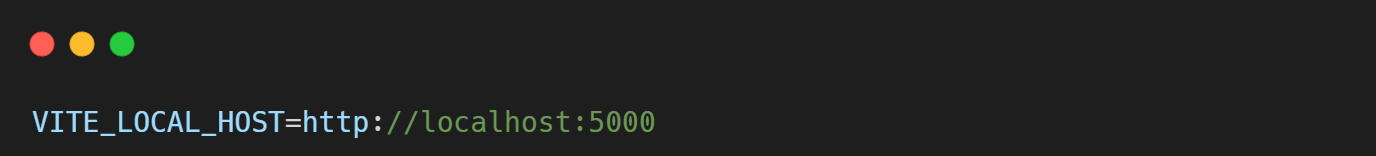


**server.js**



**Front-end**

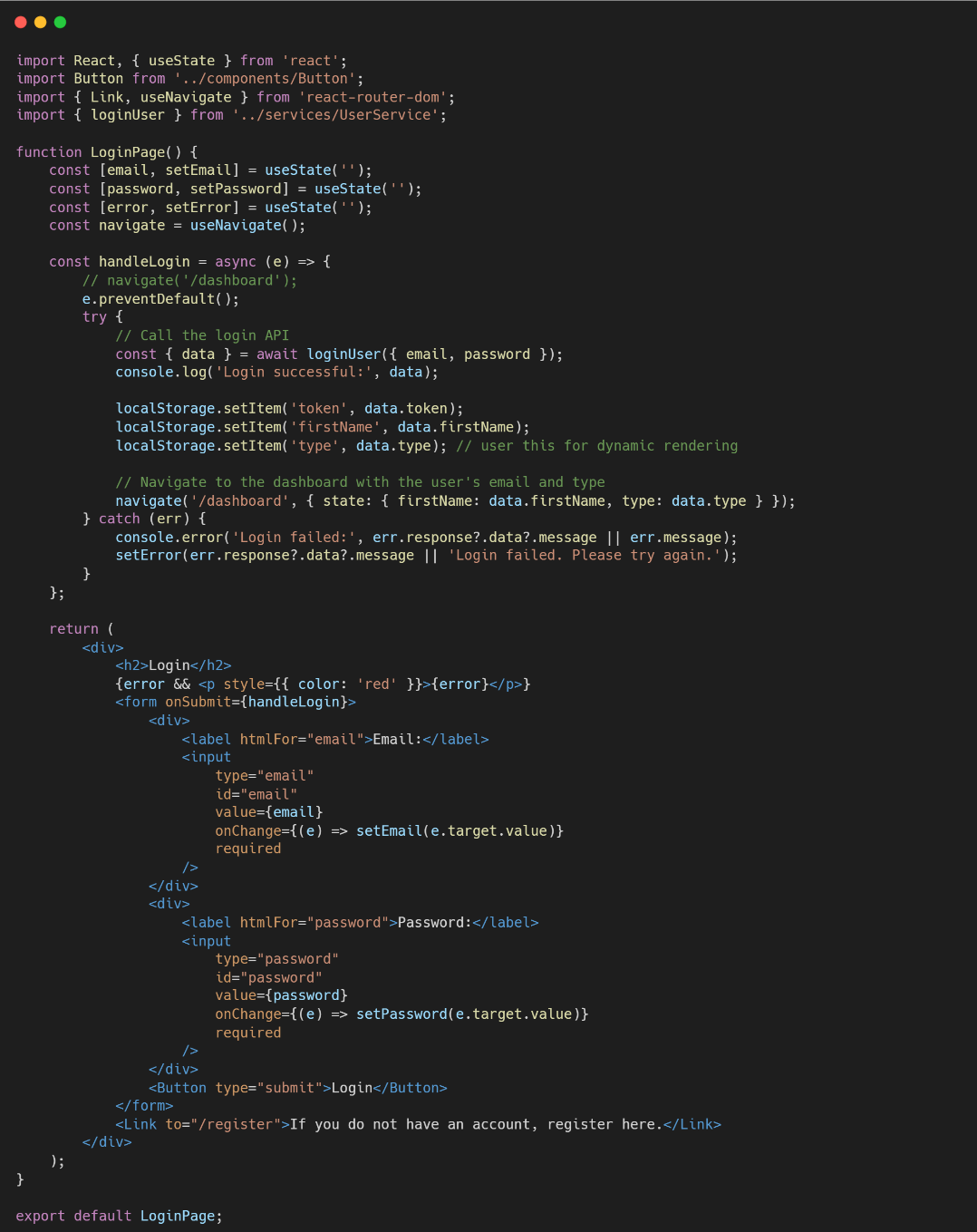
**.env**



**UserService.js**

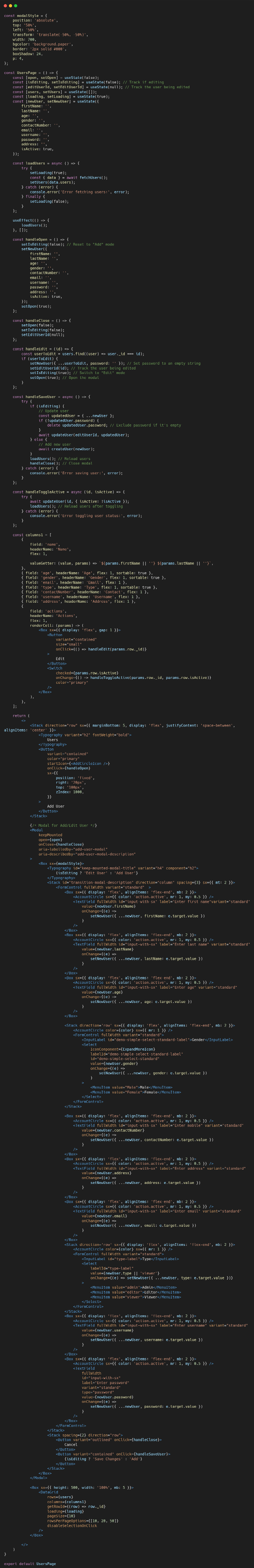


**Login.js [**Please adopt this code with caution. Revise your code relatively**]**



**Sample Revision for**

**UsersPage.jsx [**Please adopt this code with caution. Revise your code relatively**]**

 A blurry image of a city

AI-generated content may be incorrect.

**Sample Output | Enhancement Instructions:** A screenshot of a computer

AI-generated content may be incorrect.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Enhancement 1:** Enhance the UI of Modal User. The editors cannot access the UsersPage.

**Enhancement 2:** Base on UsersPage create a DashArticleListPage with this consideration:

* The articles will be available on ArticleListPage.

|  |  |
| --- | --- |
| **Name** |  |
| **Section** |  |
| **Date** |  |

**Lab Activity Rubric**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Criteria | Excellent (4 pts) | Good (3 pts) | Fair (2 pts) | Poor (1 pt) | Points |
| **Project Setup & Environment** | Project is set up correctly with all dependencies installed. The environment is well-configured and ready for development. | Project is set up with minor issues that do not impact core functionality. | Project setup is attempted but has errors affecting development. | Project setup is incomplete or non-functional. |  |
| **Navigation & Routing** | Navigation is well-structured, intuitive, and includes a responsive design. Routing is properly implemented with dynamic paths if applicable. | Navigation works correctly but may lack responsiveness or design refinements. | Navigation is functional but may have broken links or a confusing structure. | Navigation is missing or not working. |  |
| **Component Structure & Reusability** | Components are well-structured, reusable, and follow best practices (e.g., props, state, hooks). Code is modular and scalable. | Components are properly used but may lack optimization or modularity. | Components are used but with poor structure, making the code difficult to scale. | Components are poorly implemented, leading to excessive redundancy or poor organization. |  |
| **Styling & UI Design** | UI is visually appealing, well-structured, and fully responsive. Consistent styling is applied across the project. | UI is functional and mostly styled well but may have minor inconsistencies or responsiveness issues. | UI is present but lacks proper styling, consistency, or responsiveness. | UI is missing or poorly designed, making navigation difficult. |  |
| **Content & Functionality** | The app displays dynamic content effectively, and all functionalities work as expected without bugs. | Most functionalities work well, but minor issues exist. Content is displayed correctly. | Some functionalities are incomplete, and content may not load properly. | Major functionalities are missing or broken. |  |
| **Code Quality & Best Practices** | Code follows best practices, is clean, well-commented, and easy to maintain. Proper use of hooks, state, and props is evident. | Code is well-written but may have minor inefficiencies or inconsistent formatting. | Code runs but is cluttered, inefficient, or difficult to read. | Code is disorganized, hard to understand, and does not follow best practices. |  |
|  | Other Comments/ Observations: | | | **Total Score** |  |
|  | **Rating: (Total Score/24) \* 100** |  |