

Frontend Task

Login Page:

- Create a simple and responsive login form with fields for username and password.
- Implement basic validation for required fields and password strength.
- Display appropriate error messages for invalid inputs.
- Include a "Forgot Password" link for password recovery.

Registration Form / Add Students:

- Design a registration form to add new students to the system.
- Include fields such as name, email, username, password, and any other relevant information.
- Implement validation for the registration form, ensuring data integrity.
- Provide clear error messages for any invalid entries.

Show All Students List:

- Develop a page displaying a list of all registered students.
- Include relevant information like name, email, and any other necessary details.
- Implement pagination or infinite scrolling for enhanced user experience with many students.

Single Student Details Page:

- Create a dedicated page to show detailed information about a single student.
- Include a link or button on the "Show All Students" page to navigate to individual student details.

Edit Students Details:

- Implement a feature allowing users to edit the details of a student.
- Provide a button or link on the "Show All Students" page to access the edit form.
- Include validation to ensure accurate and secure data updates.

Sidebar Header Landing Page (Same header and Sidebar for both Users Roles):

- Design a responsive sidebar and header layout for the application.
- Include navigation links for different sections like "Home," "Login," "Register," "All Students," and "Edit Students."
- Ensure the layout is responsive to various screen sizes for an optimal user experience.

User Roles and Correct Routing:

- Implement a user role system differentiating between "Admin" and "User."
- Ensure that the login page routes users to the appropriate dashboard based on their role.
- Admins should have access to features like adding, editing, and viewing all students.
- Users should have limited access based on their role.

Submission Guidelines:

Provide a clear and concise readme file explaining how to run the application.

Include screenshots or a live demo link showcasing responsiveness and functionality.

Ensure that the application adheres to best practices for web development.

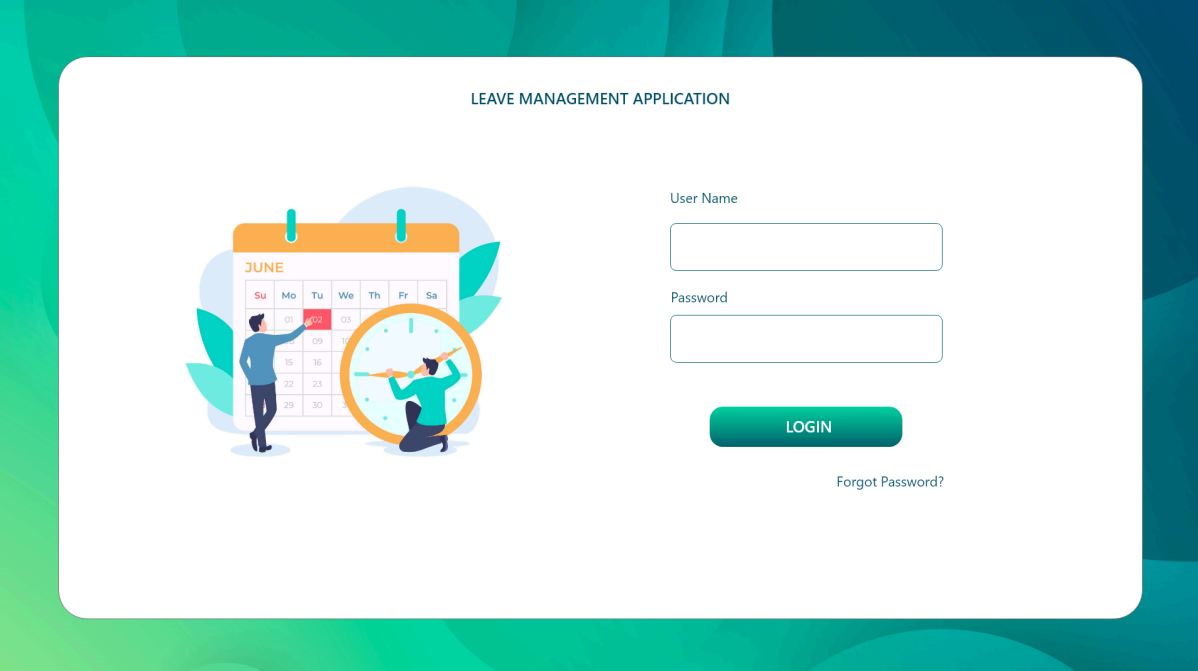
Note: Make use of HTML, CSS, and a web framework of your choice (e.g., React, Angular, Vue.js) for this task.

Frontend Task

Tips:

- You can also use normal HTML CSS and JS
- Use a modern frontend framework like React, Angular, or Vue.js etc. for efficiency and maintainability.
- Implement a consistent and visually appealing design using CSS frameworks like Bootstrap or Tailwind CSS
- You have to design only static pages
- If you are also interested in Backend you can do the possible Basic Backend Code with proper database connection and the Required Fields.
- Ensure good user experience with proper feedback on form submissions and clear navigation.
- Add the Project code on Git

Reference UI for theme and colour combination is



LEAVE MANAGEMENT APPLICATION

User Name

Password

LOGIN

[Forgot Password?](#)