

Software System Development (Monsoon 2023) Final Project Report

- Team Name: Code Crafters
 - Team No.: 04
 - Project Title: Website Development for CSG Lab
 - Project No.: 03
 - Instructor/Mentor (mentioned in project details sheet): Soumitra Ghosh
 - Project GitHub Repository URL:
https://github.com/ayush2804/SSD_Team_04_Website_Development_for_CSG_Lab
 - Members Details (with Roll Numbers):
 1. Atharv Doley 2023201061
 2. Atharva Pande 2023201065
 3. Ayush Pandey 2023202009
 4. Ayan Ray Choudhury 2023202010
-
- Requirements:

ID	Requirement Name	Description	Status	Owner
1	Tab Creation	Design and Create HTML/CSS Structure of Home, Faculty, Student	Completed Completed	Atharv Doley
2	Tab creation	Design and Create HTML/CSS Structure of Project, Events and Publications	Completed Completed	Atharva Pande
3	Server	Set up the backend server. Set up authentication and authorization i.e. admin login for content management. Implement JavaScript for any interactive elements pages,	Completed Completed	Ayush Pandey
4	Database	Create API endpoints for retrieving data from the database. Define the data structure and store any dynamic content (e.g., lab history) in a database.	Completed Completed	Ayan Roy Choudhury

Logic

The architecture is designed using the MERN stack as follows :-

Front-end Logic (Vite + React JSX):

- Vite is chosen as the front-end build tool to enhance the development experience and optimize the site's performance. Vite supports React JSX, offering fast and efficient bundling. Through Vite's development server, hot module replacement (HMR) is facilitated, allowing real-time updates during development.
- The front end is built using React JSX, providing a modular and component-based structure. Each tab—Home, Faculty, Students, Projects, Events, and Papers—is represented as a React component. React's state management and component lifecycle ensure a dynamic and responsive user interface. JSX syntax simplifies the creation of complex UI elements.

Middleware Logic (React + Node.js + Express):

- React acts as the view layer, handling the presentation of the website. Each React component corresponds to a specific tab, facilitating the rendering of data retrieved from the backend. React Router is employed for seamless navigation between different tabs.
- Node.js, along with the Express.js framework, serves as the middleware connecting the front end to the backend. Express handles routing, allowing for the creation of APIs that fetch and send data to and from the MongoDB database. Middleware logic includes authentication, error handling, and the establishment of RESTful endpoints for data retrieval and manipulation.

Database Logic (MongoDB):

- MongoDB is chosen as the NoSQL database to store information for each tab—Faculty, Students, Projects, Events, and Papers. The schema-less nature of MongoDB accommodates diverse data types, and its scalability aligns with the potential growth of the research center's information.
- Each tab is represented as a separate collection in MongoDB. For instance, there is a "faculty" collection to store faculty information, a "students" collection for student data, and so on. This logical separation ensures efficient organization and retrieval of data.

Flow for Authentication

- Admins have a dedicated login option. They enter their credentials, and the frontend sends a login request to the backend for authentication.
- The frontend sends an authentication request to the backend, which verifies the user's credentials against the database. If successful, the backend generates a token and sends it back to the frontend for future requests.
- The admin's credentials are authenticated against the database. Upon successful verification, the backend generates a token and sends it to the frontend.
- The frontend stores the token securely, typically in a cookie or local storage, for subsequent requests. This token acts as a key to verify the user's identity and permissions.
- After authentication, the frontend can make requests for data, such as faculty details, student profiles, project information, event schedules, and research papers.
- The backend, developed using Node.js and Express, communicates with MongoDB to fetch the requested data based on the user's permissions. Admins have read and write access, while regular users have read-only access.
- The backend sends the requested data to the frontend. React components dynamically render the information, ensuring a seamless and responsive user interface.

Use Cases

Stakeholders :- Students and faculty of CSG Lab will use it for

1. Getting information on upcoming events, recent activity on social media related to the CSG Lab other details.
2. Other users can use it to get information about faculties, students and events as required.
3. Imperative to enhance communication, collaboration, and information dissemination.

Home: overview of the research center's mission, vision, and recent accomplishments.

Faculty:

- Under the Faculty tab, detailed profiles of faculty members are presented, encompassing personal and professional details, areas of expertise, ongoing research projects, and contact information.
- Admin privileges facilitate the addition or modification of faculty profiles, ensuring real-time updates.

Students:

- The Students tab hosts information about current students, including research interests, ongoing projects, and academic achievements.
- Admin access allows for the addition or removal of student profiles, ensuring accurate and up-to-date information.

Projects:

- The Projects tab offers an overview of ongoing and completed research projects within the research center.
- Each project entry includes pertinent details such as project descriptions, collaborators, funding sources, and project status.
- Admins possess the capability to add, edit, or remove project information as required.

Events:

- The Events tab functions as a dynamic calendar for upcoming and past events, including seminars, conferences, workshops, and other activities.
- Users can access event details, schedules, and registration information.
- Admins are empowered to add new events, update existing ones, and manage event-related content.

Publications:

- The Publications tab showcases the research output of the research center, including publications, journals, and conference papers.
- Each entry comprises publication details, author information,
- and links to the full publication where applicable. Admin access facilitates the addition, modification, or removal of publication entries.

User Authentication:

- Regular users can access information on faculty, students, projects, events, and research publications without any need to login.
- Administrators, through a secure dashboard, have the authority to manage and update data across all tabs.