ONLINE PLACEMENT INFORMATION GATHERING SYSTEM

TEAM : THREE AWESOME

CHANDRANSH SINGH (22CS30017)

VISTRINNA GOGOI (22CS10080)

GAURAV ROY (22CS10082)

PROBLEM DESCRIPTION

- The need for an efficient user-friendly software to digitalise the placement process for the college and ease the process experience for the students, companies and the institute personnel.
- The main aim is to develop a secure online software to fasten the tiresome manual process and also makes the process error-free and reliable. The dependency on large teams can be reduced to a huge extent with the software taking care of the needs 24x7 as and when needed.
- The online software should make it easier for companies to look into the current talent base of the college by looking at the portfolios of all students and facilitates the students to gain insights into the company by contacting the alumni of the institute and chatting with them.
- The task of company validation should be easily done by the admin before directly allowing the companies to engage with the college and its students which makes a safer and better placement system.

SOLUTION

The solution to the above problem has been solved efficiently using an online web based online placement information gathering system that allows the college to carry out the placement procedure smoothly.

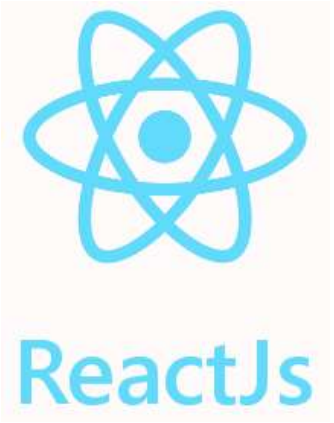
A full stack user friendly website with engaging UI and easy to use features has been developed using the **MERN** stack.

TECHNICAL DETAILS

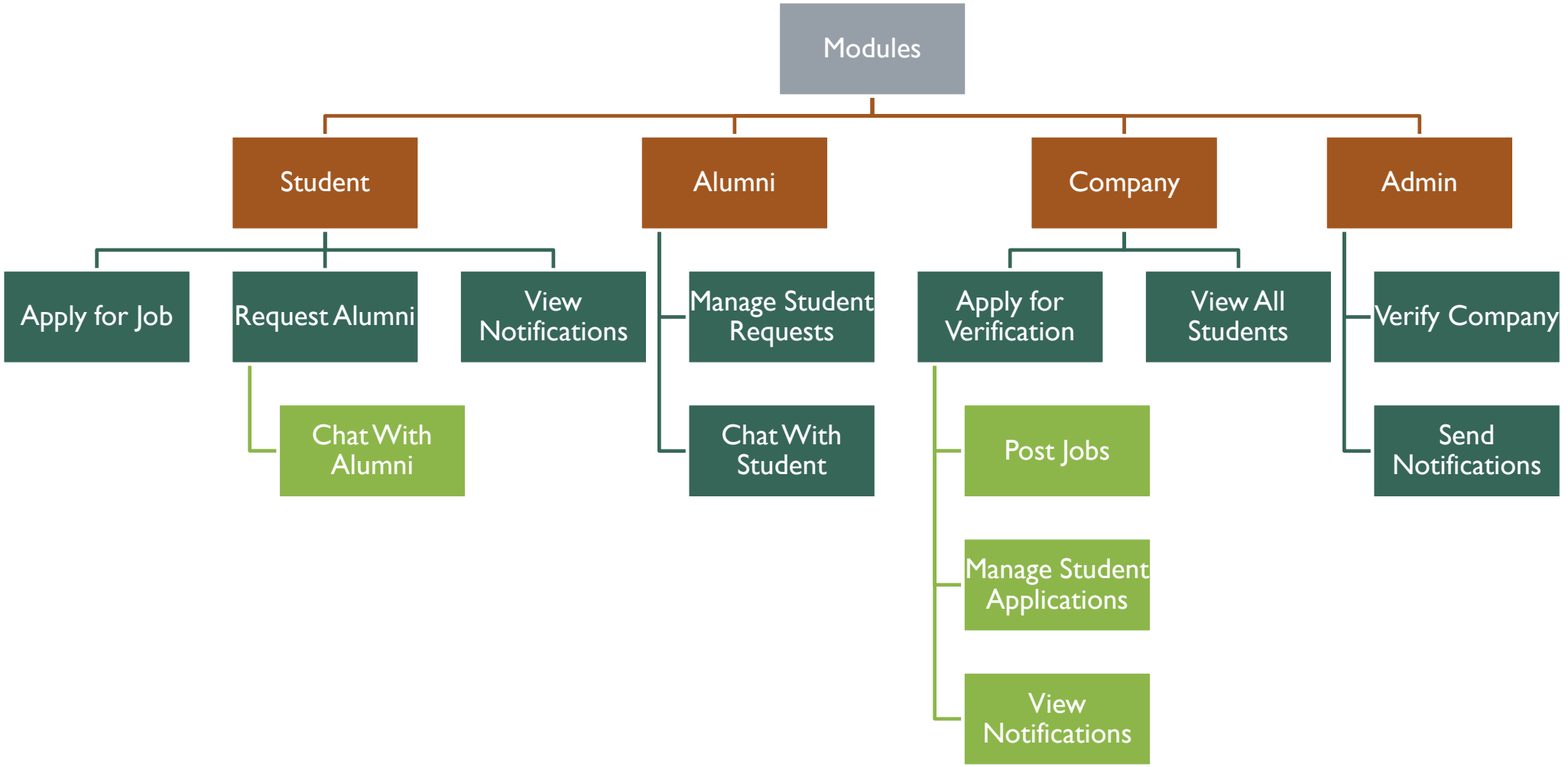
- **Architecture** : The web-based online placement portal consists of a frontend for user interaction, a backend handling data processing and logic, and a database storing all data. It ensures security with authentication, integrates external services, and scales efficiently.
- **Programming Languages and Frameworks** : The software is built in JavaScript. The frontend is designed using HTML and CSS and JS based frameworks like ReactJS, NodeJS, ExpressJS and Socket.IO.
- **Database Management** : For database management, MongoDB has been used. It allows efficient data storage, retrieval and manipulation.

TECHNICAL DETAILS

- **APIs and Integrations** : External libraries such as Axios, Toast, React-Router, JSONWebToken, Bcrypt and APIs like Cloudinary have been integrated into the project.
- **Scalability and Performance** : The software runs smoothly on various desktop based web browsers. The responsiveness of the software is not satisfactory on mobile and tablet devices as of now.



USE CASES


Modules
Student
Alumni
Company
Admin
Apply for Job
Request Alumni
View Notifications
Chat With Alumni
Manage Student Requests
Chat With Student
Apply for Verification
View All Students
Post Jobs
Manage Student Applications
View Notifications
Verify Company
Send Notifications

CHALLENGES FACED

- Merging code changes made by different team members on GitHub initially posed a problem but was overcome later with practise and familiarity with the process.
- Implementing real time chat between student and alumni brought challenges in implementation due to lacking knowledge of Socket.IO framework and was overcome with the help of online documentations.
- Integrating the backend and frontend development together to retrieve and post data was initially challenging.
- Ensuring that the user does not get logged out on refreshing using saved token posed a challenge in frontend implementation.

SCOPE OF IMPROVEMENT

- The website can be made more responsive to mobile and tablet devices which could not be implemented due to time constraints.
- File upload system can be improved with better reading into the documentations of cloudinary and efficient implementation with Multer.
- Option to download the resume of a student in PDF format can be implemented.
- Stronger security measures for user authentication can be implemented with OTP or security questions.
- The unread or new messages can be indicated with the help of notifications or chat bubbles.

THANK YOU