BANNARI AMMAN INSTITUTE OF TECHNOLOGY
An Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade
Sathyamangalam - 638401 Erode District, Tamil Nadu, India

Stay Ahead

STUDENT NAME: AAESHA SIDDHEEQA N

ROLL NO: 7376222AD103

SEAT NO: 1

PROJECT ID: 1

PROJECT TITLE: BIT JOB PORTAL

Technical Components

Component	Tech Stack
Backend	Spring Boot
Frontend	React
Database	MySQL
API	RESTful services

Implementation Timeline

Phase	Deadline	Status	Notes
Stage 1	02/05/2024	Completed	Planning and Requirement gathering
Stage 2			Design and Prototyping
Stage 3			DB Designing
Stage 4			Backend Implementation
Stage 5			Testing & Implementation

PROBLEM STATEMENT:

JOB PORTAL DEVELOPMENT:

Develop a user-friendly job portal for our college to centralize job postings and company details. Replace the current method of sharing job information via Excel sheets through email.

ELIGIBILITY CHECK INTEGRATION:

Implement robust eligibility checks to verify that only eligible students can apply for posted job roles. Integrate a system to validate students' qualifications and other eligibility criteria.

STUDENTS DETAILS INTEGRATION:

Integrate a system for students to regularly update their academic details on the portal. Ensure that updated academic information is maintained to retain eligibility for job applications.

CONSTRAINTS FOR PLACED STUDENTS:

Create for placed students from applying for additional job opportunities through the portal. Implement checks to validate the employment status of students before allowing job applications.

ADMINISTRATIVE ACCESS:

Provide administrative access to the college's training and placement team to manage job postings, eligibility criteria, and student profiles efficiently. Enable administrators to track job application statuses.

PROJECT FLOW:

1. PROJECT SETUP AND ENVIRONMENT CONFIGURATION:

Set Up Frontend (React):

Create the React project structure and configure necessary dependencies (e.g., Axios for API calls, React Router for routing).

Set Up Backend (Spring Boot):

Initialize a Spring Boot project and configure dependencies for RESTful API development (e.g., `Spring Web, Spring Data JPA. Spring security).

Database Setup (MySQL):

Set up a MySQL database instance and create tables for storing user profiles, job postings, and related data.

2. GOOGLE OAUTH INTEGRATION:

Implement Google OAuth:

Integrate Google OAuth authentication for user login using OAuth libraries compatible with React (e.g., react-google-login) and Spring Boot.

Secure API Endpoints:

Ensure API endpoints are secure and require authentication for admin access.

3. BACKEND DEVELOPMENT:

User Management:

Implement user management functionalities for admin and regular users, including registration, login, and role-based access control.

Job Posting APIs:

Develop RESTful APIs for admin to post jobs, apply constraints (CGPA, percentages, branch, arrears), and manage job postings.

FUNCTIONAL REQUIREMENTS:

1. **User Authentication:**
    - Users can log in using Google OAuth authentication.
    - Admin users have additional privileges compared to regular users.
2. **Job Posting and Management:**
    - Admin can post job opportunities including job title, description, company details, and eligibility criteria.
    - Admin can edit or remove job postings as needed.
3. **Eligibility Checks:**
    - Admin can set eligibility constraints such as CGPA, percentages, branch, and arrears for job postings.
    - Only eligible students can view and apply for job postings based on the set criteria.
4. **Student Profile Management:**
    - Students can update their details regularly, including CGPA, percentages, branch, and arrears etc.
    - Updated profiles are used for eligibility checks during job application.
5. **Placed Students Constraints:**
    - Admin can apply constraints for placed students regarding their eligibility for second job opportunities.
    - Placed students can view job postings only if they meet the specified constraints.

6. **Job Forum:**

- Users (both admin and students) can access the job forum to view available job postings.
- Job postings display relevant information such as job title, company details, eligibility criteria, and application deadline.

NON-FUNCTIONAL REQUIREMENTS:

1. Performance:

Ensure fast response times for loading job postings and updating user profiles, even under high user traffic conditions.

2. Scalability:

Design the system to handle a growing number of users, job postings, and concurrent interactions without compromising performance.

3. Security:

Implement robust data encryption, secure authentication mechanisms (like OAuth), and access controls to protect sensitive user and job data.

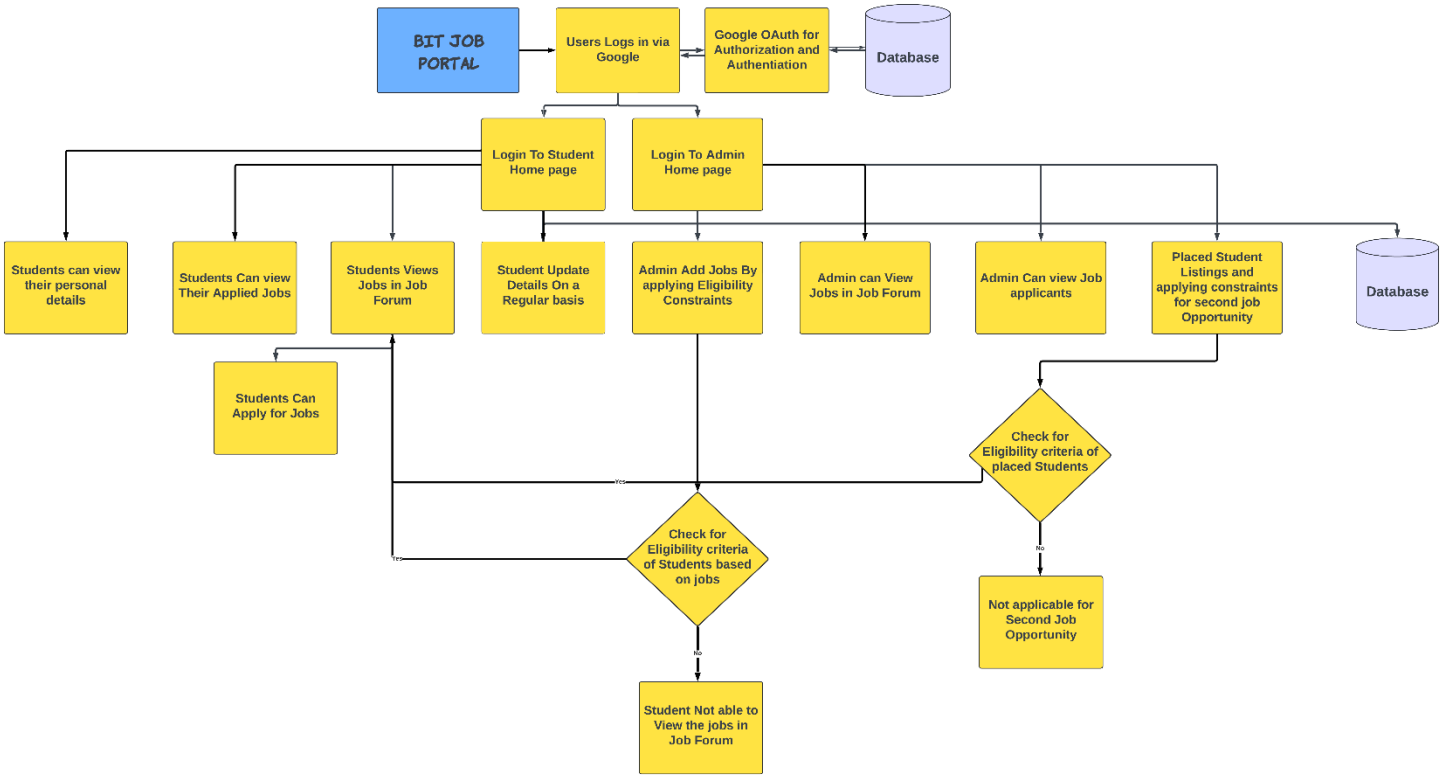
4. Reliability:

Ensure high system availability and minimal downtime through proper error handling, backup mechanisms, and regular maintenance.

5. Compatibility:

Ensure compatibility across different devices, browsers, and screen sizes to provide a seamless experience for all users accessing the job portal.

FLOW CHART:


BIT JOB PORTAL
Users Logs in via Google
Google OAuth for Authorization and Authentiation
Database
Login To Student Home page
Login To Admin Home page
Students can view their personal details
Students Can view Their Applied Jobs
Students Views Jobs in Job Forum
Student Update Details On a Regular basis
Admin Add Jobs By applying Eligibility Constraints
Admin can View Jobs in Job Forum
Admin Can view Job applicants
Placed Student Listings and applying constraints for second job Opportunity
Database
Students Can Apply for Jobs
Check for Eligibility criteria of placed Students
Yes
No
Check for Eligibility criteria of Students based on jobs
Yes
No
Not applicable for Second Job Opportunity
Student Not able to View the jobs in Job Forum