SOFTWARE REQUIREMENTS

PLACEMENT REGISTRATION FORM

Name	**VISHAL V R**
Roll number	**7376222CT161**
Seat number	**281**
Project number	**01**
Problem statement	**PLACEMENT REGISTRATION FORM**

PROBLEM STATEMENT : PLACEMENT REGISTRATION FORM

In college placement processes, managing job registrations and facilitating communication between students and potential employers can be time-consuming and inefficient. Manual handling of registration forms, lack of transparency in job offers, and difficulties in matching students with relevant job opportunities are common challenges faced by both students and administrators.

STACK:

Component	Tech Stack
Frontend	**Anguar(Js Framework)**
Backend	**Express.js(Web framework for Node.js)** **Node.js(Javascript runtime environment)**
Database	**MongoDB(NOSQL Database)**
API	**REST Ful API / GraphQL APIs**

PROGRESS - TIMELINE:

Phase	Deadline	Status	Notes
Stage 1	03/05/2024	In Progress	Planning and Requirement Gathering
Stage 2		Not Started	Design and Prototyping
Stage 3		Not Started	DB Designing
Stage 4		Not Started	Backend Implementation
Stage 5		Not Started	Testing & Implementation
Stage 6		Not Started	Deployment

1. INTRODUCTION

1.1 PURPOSE :

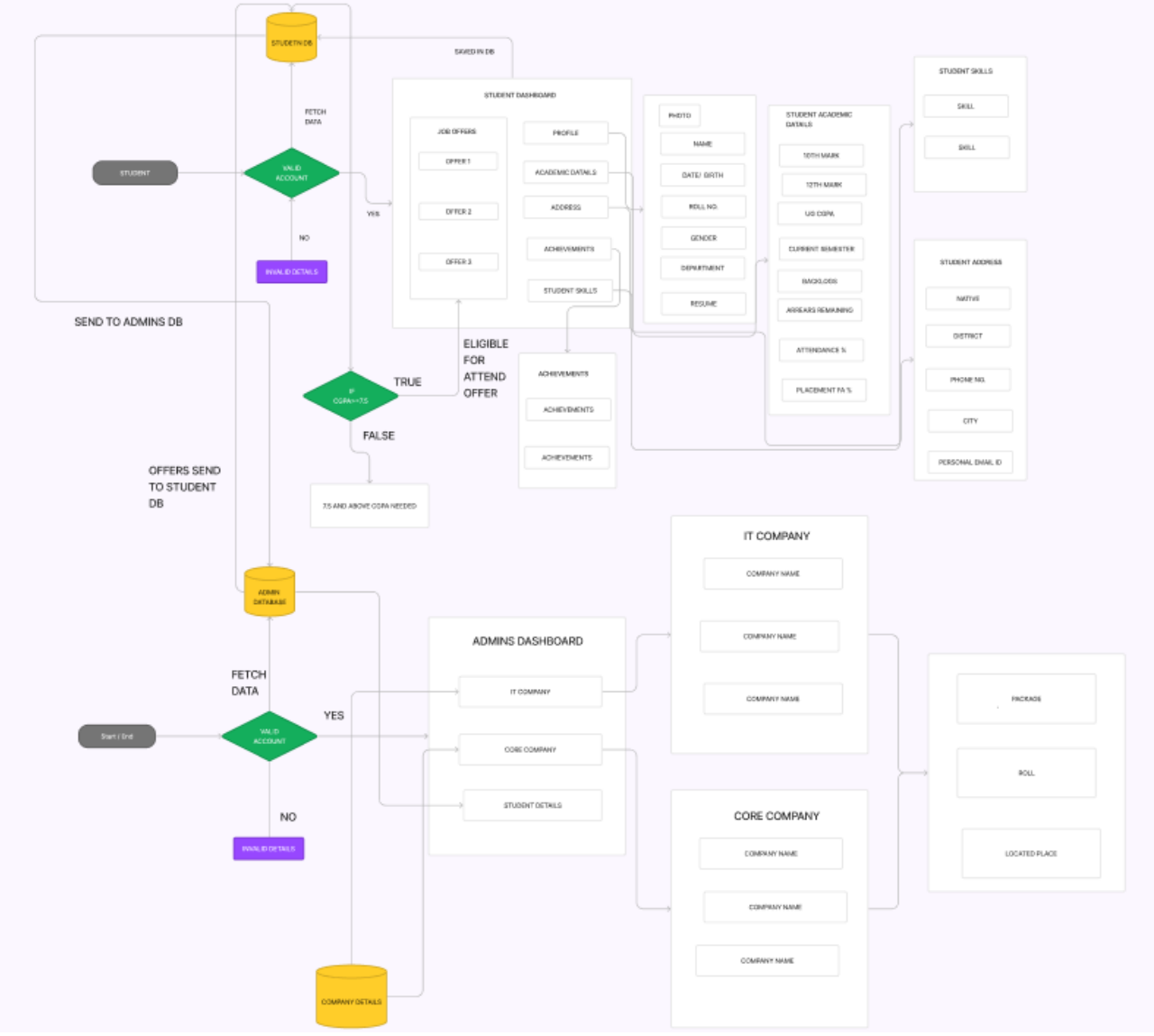
The purpose of this document is to present a detailed description of the placement registration form ,this form helps students to increase transparency in job offers , and easy to make a communication between students and job providers.

1.2 SCOPE OF THE PROJECT :

- **User Profiles:** The system will have two types of users: students and administrators.
- **Key Working Features:** Registration of student profiles, display of job offers and packages, eligibility criteria based on CGPA and SKILLS known, application for job offers, placement status updates.
- **Messaging and Communication:** Allow communication between students and administrators regarding job offers and placements.

- **Purpose of the System:** To automate and simplify the process of job registration and placement for students and administrators.
- **Dependencies:** Angular, Express.js, Node.js, MongoDB, RESTful API or GraphQL APIs.
- **Authentication:** Implement authentication mechanisms to ensure that only authorized users can access the system and perform actions.

FLOW CHART :



FUNCTIONAL REQUIREMENTS :

1. **Student Registration:**
    - Students should be able to sign up by filling out a form with their details like name, student ID, etc.
2. **Job Listings:**
    - Admins can post job opportunities with details like company name, job title, salary, and criteria like minimum GPA required.
3. **Viewing Jobs:**
    - Students can see all available job listings with their details.
4. **Applying for Jobs:**
    - Students can apply for jobs they're interested in by clicking a button.
5. **Checking Application Status:**
    - Students and admins can see where each student stands in the application process.
6. **Messaging:**
    - Students and admins can send messages to each other regarding job offers and applications.
7. **Security:**
    - Only authorized users can access the system, ensuring privacy and safety.

STACK FUNCTIONALITY:

1. **Frontend: Angular (JavaScript Framework)**

    - **Functionality:**
      Angular is used to build the user interface (UI) of the placement registration form system.

    - **Specific Features:**
        - Creating user-friendly forms for student registration and job application.
        -

2. **Backend: Express.js (Web Framework for Node.js) and Node.js (JavaScript Runtime Environment)**

    - **Functionality:**
      Express.js and Node.js handle the server-side logic and communication between the frontend and the database.

    - **Specific Features:**
        - Handling HTTP requests from the frontend, such as submitting registration forms and retrieving job listing

3. **Database: MongoDB (NoSQL Database)**

    - **Functionality:**
      MongoDB stores the data used by the placement registration form system.

    - **Specific Features:**

        - Storing student information, including names, roll numbers, and application statuses.
        - Storing job listings with details like company names, job titles, eligibility criteria, and application statuses.

4. **API: RESTful API / GraphQL APIs**

    - **Functionality:**
      The API serves as the interface between the frontend, backend, and database, allowing them to communicate and exchange data.

- **Specific Features:**
  - Exposing endpoints for frontend components to interact with backend services, such as submitting registration forms and fetching job listings.

LOGIN PAGE :