**Software Requirement Specification for Course Registration**

| **Name** | Ramani R |
| --- | --- |
| **Roll no** | 7376221SE141 |
| **Project ID** | 18 |
| **Problem Statement** | Register their department course and their individual electives and add on courses. |

**1**. **Introduction**

**1.1. Purpose:**

The purpose of this document is to provide a detailed description of the course registration system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate, and how the system will react to external stimuli.

**1.2. Scope of Project:**

●This software system will serve as a portal for students to register for their individual electives, and add-on courses. From an administrative perspective, this system will provide a comprehensive interface for managing course registrations and student enrollment.

**2. System Overview:**

**2.1. Users:**

1. **Students**:

They can register for their individual elective courses based on their departments. They can add any supplementary or add-on courses.

**2. Admins:**

Review and manage student course registrations. Approve or reject course selections. Access analytical dashboards for course enrollment and student data.

**2.2. Features:**

**1. Login and registration:**

Students can register for an account or log in with their existing account.

**2.Department Course Registration:**

Students can select the required courses for their academic

department.

**3. Individual Elective Registration:**

Students can choose their individual elective courses.

**4. Add-on Course Registration:**

Students can register for any additional supplementary or add-on courses.

**5. Admin Course Management:**

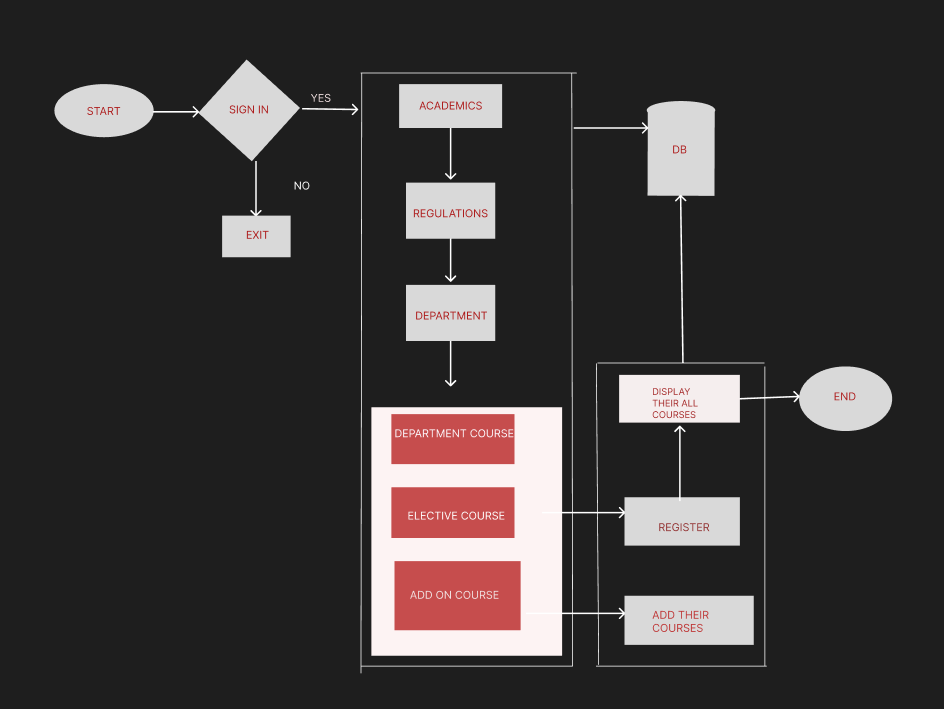
Admins can review and manage all student course registrations. Admins can approve or reject course selections.

**6.Admin Analytical Dashboard:**

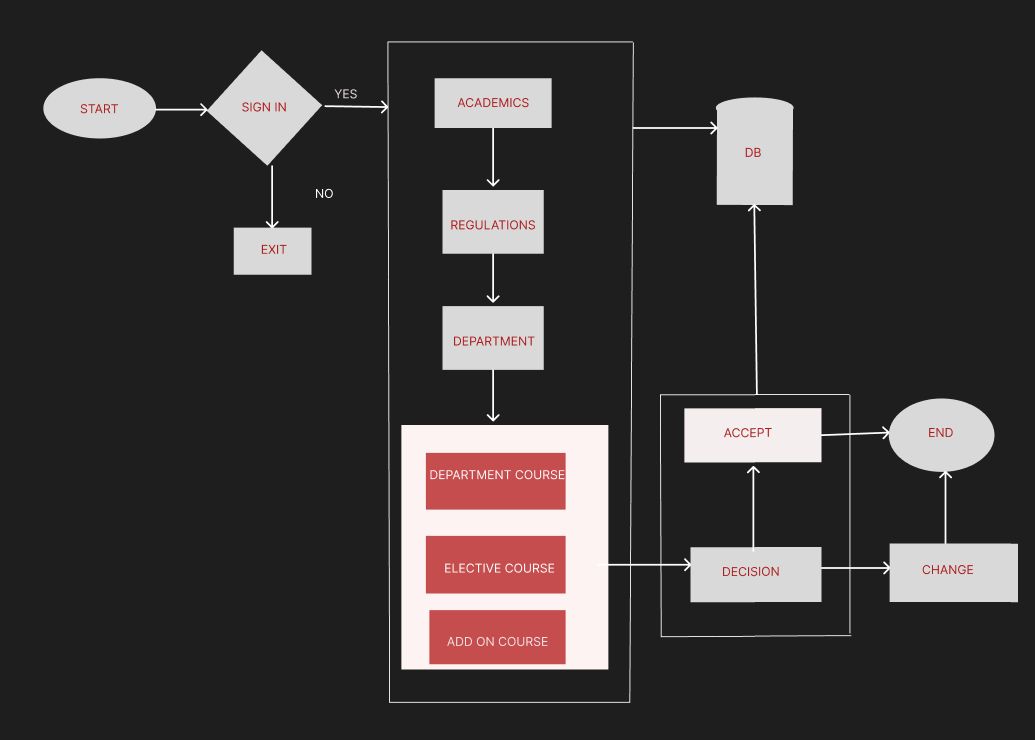
Admins can access analytical data on course enrollment, student demographics, and other relevant metrics.

**FLOW CHART:**

USER’S INTERFACE:



ADMIN INTERFACE:



**3.1. Functional Requirements:**

**User Management:**

Students can register and log in. Admins have access control and dedicated features.

**Course Registration:**

Students can select their department-required courses. Students can choose individual elective courses. Students can register for add-on or supplementary courses.

**Admin Course Management:**

Admins can review and manage all student course registrations. Admins can approve or reject course selections.

**Analytical Dashboard:**

Admins can access comprehensive data on course enrollment and student details.

**3.2. Non-Functional Requirements:**

**Performance:**

The system must respond to user actions within 2 seconds and handle a concurrent user load of at least 100 users without significant performance degradation.

**Security:**

User data must be encrypted during transmission and storage, and access to sensitive functionalities should be restricted to authorised admin users through secure authentication mechanisms.

**Usability:**

The user interface should be intuitive and user-friendly, with clear and concise error messages provided to guide users in case of input errors or system failures.

**Reliability:**

The system should be available 24/7 with minimal downtime and should have a backup and recovery mechanism in place to prevent data loss in case of system failures or crashes.

**Scalability:**

The system should be designed to accommodate an increasing number of users and data volume over time, and it should be scalable to support additional features and functionalities as per future requirements.

**Stack: Mean**

| Front End | Angular(Js Framework) |
| --- | --- |
| Backend | Node Js, Express js |
| Data Base | MongoDB |