IT314: Software Engineering
Project Detail Document
Group No: 29

Title: **Academic Management System**

Team Members:

Khushbu Patel	201901140
Mahima Rohit	201901091
Nirav Damor	201901156
Priyanshu	201901468
Tarun Parashar	201901221
Varun Motwani	201901044

Reporting TA - Kalgi Gandhi

Date - 22 March 2022

Version number - 1.0 v

Start Date:20-03-2022
End Date: 13-05-2022 (8 weeks)

Objectives of the Project -

The primary objective of this project is to develop a system that facilitates the management of academic work in a university. This primarily helps in large-scale data management in a university. When the amount of data is huge, the paperwork seems infeasible. This system provides several functionalities on a web-based interface to its users. The end-users of this academic management system include faculty members, students, and the admin in a university.

The purpose of the project is to build an application that can reduce the manual work for managing the Students, Faculties, and admin. It allows them to stay up to date regarding all the aspects of the academic institution. It also allows the students to have access to the profiles, getting updated with the upcoming announcements from the faculty and administration. The purpose of this application also covers a few advantages for the faculty members which are student course feedback, quick material uploads/announcements, easy quiz hosting with the question randomizer.

Functionalities -

The system functionalities can be broadly categorized into three subparts according to three user classes. These are -

❖ Students

1. After students get admission into the university, they are provided with user credentials (which can be later modified).
2. Once registered students can log in with their respective credentials.
3. Students can edit their profile (contact details, address, profile picture, etc.).

4. Students can register for different courses for a given semester, which is then to be approved by the course instructor.
5. Within some specified time duration after course registration, students can adjust their courses.
6. Students can view the course material, announcements, and attendance for the registered courses.
7. Students can submit the assignments on the course page.
8. Students can view their semester-wise results.
9. Students can view the schedule and announcements made by the admin.
10. Students can view the fees structure and can check the status of the fees (paid, due).
11. Students will receive email notifications if any announcements, assignments, and results are posted for the course.
12. On the Helpdesk page, students can raise an issue and ask queries
13. Students can give feedback on the courses at the end of the semester.

❖ Faculty

1. Faculty can register themselves and the admin approves the registration.
2. Faculties are assigned to courses of their field by the admin.
3. Make announcements, post assignments, and study materials, declare the results of their courses and mark the attendance of the students.
4. Faculty will get the course feedback from the students.
5. Faculty members will have their profile where they can add/update their details (area of interest, experience, profile picture, and other contact details).
6. Course instructors can take quizzes and the system will randomize all questions for the students in order to minimize plagiarism.

❖ Admin

1. Admin can manage the courses(create new courses, edit attributes of course, delete course).
2. Admin can assign instructors to the courses.
3. Admin can approve the user registrations.

4. Admin can query all the data of the system regarding students, faculties, or courses.
5. Admin can post notices on the dashboard of the website.
6. Admin can set and edit the fees structure.

9. Project Deliverables

(a) Milestones (with intermediate milestones at two weeks)

After 2 weeks	- Sign Up/login functionality for the users with Authentication. - Developing database Models for all Entities - DFD Modeling
After 4 Weeks	- Implementation of Courses, grades(for students) and DashBoard Functionalities For the users. - UML Design
After 6 Weeks	- Course Feedback and Help Desk Functionality - Final Design of database Models and all backend APIs and routes. - Use Cases Design, Data-flow Diagram
After 8 Weeks	- Final Documentation/Design - Functionality for querying data for admin + previous functionalities.

(b) List of final deliverables

1. Complete documentation (internal included)
2. Report for audit
3. User guide (includes deployment guide, end user guide etc.)

10. Estimated total time (unit: hours) (Please specify a breakdown of different project activities)

1. Front-end Design & implementation - 25 hrs
2. Back-end Design & implementation - 20 hrs
3. Database models and schema design - 12 hrs
4. Research & Data collection - 15 hrs
5. DFD & UML Modeling - 12 hrs
6. Documentation - 10hrs
7. Testing - 5 hrs

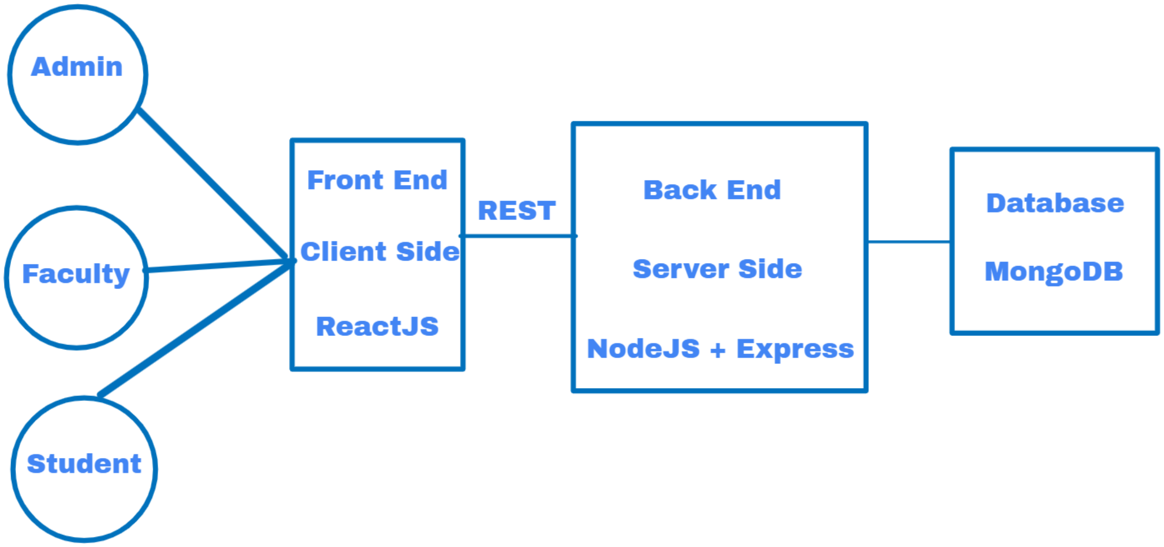
11. Hardware requirements

- Computer, laptop, Smartphone (any)
- Network connectivity
- Hard Drive: Minimum 10 GB
- Memory (RAM): Minimum 2 GB

12. Software requirements

- Operating System (Windows, Linux, MacOs) on PC
- Web browser Latest version; Recommended - Chrome, firefox, edge, safari
- Operating System on smartPhone (Android 7+, iOS 12.1 +)

12. Technology / Architecture (attach a separate document, if necessary)



13. Standard to be followed throughout the project (team need to fix a standard and follow the conventions recommended).

This project heavily relies on the core principles of the software development process. One core methodology of software development that we have decided to follow in this project is the concept of **Agile Development Methodology**. With agile, we can make sure that any requirements, even if it comes during the SDLC process, is implemented without affecting the rest of the functionality.

All the design and functionalities are well documented so that each team member can easily understand what is being implemented and how.
Scrum meetings are held two-three times a week, to discuss the flow of work and the problems which are occurring in the path of development.

Apart from that we need to maintain some notion of reliability, efficiency and also the source code is well commented and the names of functions and variables are relevant and self- explanatory.