Software Design Document

for

(P012)University Management System

Version 1.0

Prepared by :
1. Deepkamal Singh (190001011)
2. Krishanu Saini (190001029)
3. Kuldeep Singh (190001030)
4. Rahul Kumar (190001049)

Submitted to :
Lecturer
May 7, 2021

Contents

1 Introduction

1.1 Purpose

The primary purpose of this document is to describe the implementation of "University management System" project. "UMS" is a all-in-one university manager which aims to make management easier and effective.

1.2 Scope

This document will consist of UML diagrams and data-flow-models along with mockups to describe in detail the functioning of the application. It will give insight to the stakeholders on the goals addressed by the "UMS" project and it will act as design document for the developer.

1.3 Description of Problem

"University Management System" is a website with main goal to inculcate technology in the process of managing University data and bring optimistic changes in this process. The main goal of this website is to streamline the daily activities of an educational institute and make it more interactive and efficient.

Owing to the covid pandemic times and technological advancements, soon most of the education related events will be shifting to more efficient and secure online counter-parts. Improving user accessibility and user involvement in the process is one of the primary reasons for creating this website.

Other goals are -

- To improve the efficiency and provide a good database system to manage large amounts of data.
- Provide online features like:
  - Assignment submission platform
  - Grading platform
  - Notification and daily updates etc.

2 Design Overview

2.1 Technology used

The "University Management System" website should be build on flask (python framework), Bootstrap and material-ui for front-end and mySQL database.
Back-End - python flask.
Front-End - Bootstrap, JavaScript.
Database - mySQL.
Testing - python & javascript.

2.2 System Architecture

"University Management System" has basically 5 types of users interfaces.

- Student interface — UI for student dashboard containing links to related functions.
- Faculty interface — The view shoes a dashboard having a menu with multiple options and links.
- Admin interface — UI for admin dashboard containing links to functions.
- Company Recruiting interface — The view with functions related to placement statistics and interviews scheduling.
- Data Model — The classes needed to organize student academic data, enrollment data, placement statistics, assignments, grades.
- Data Storage — The mySQL database implemented by flask API's. This is the interface for storing and adding raw data in meaningful manner.

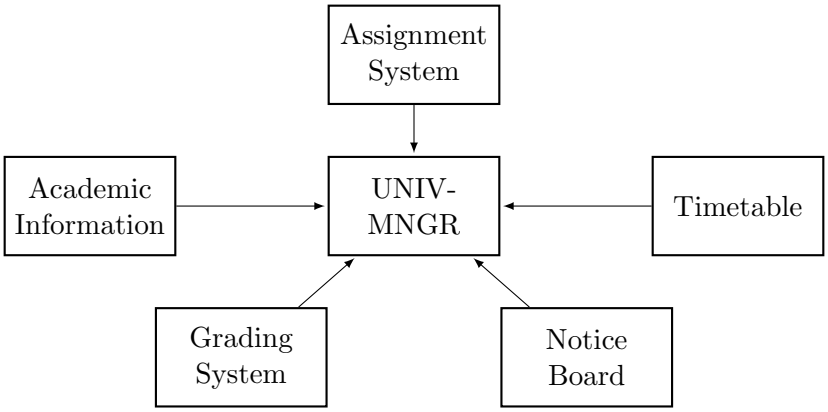


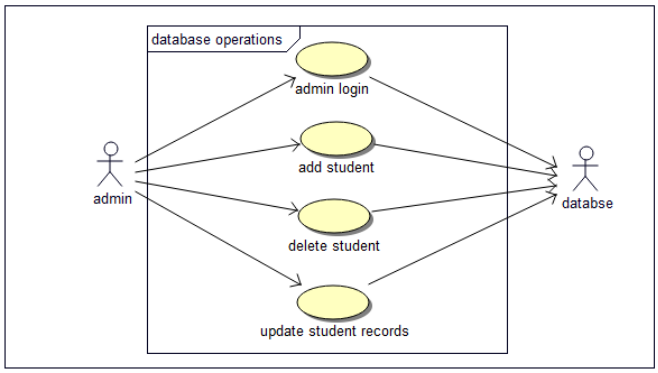
Fig-1: Subsystems of software

2.3 Use Case Diagram

"University Management System" has basically 4 types of users.

- Student
- Faculty
- Academic office (Admin)
  - Director
  - Faculty Representative
  - Student Representative
- Company Recruiting Team

Fig 2.1 Shows a Interaction of stakeholders with the system and its functionalities.



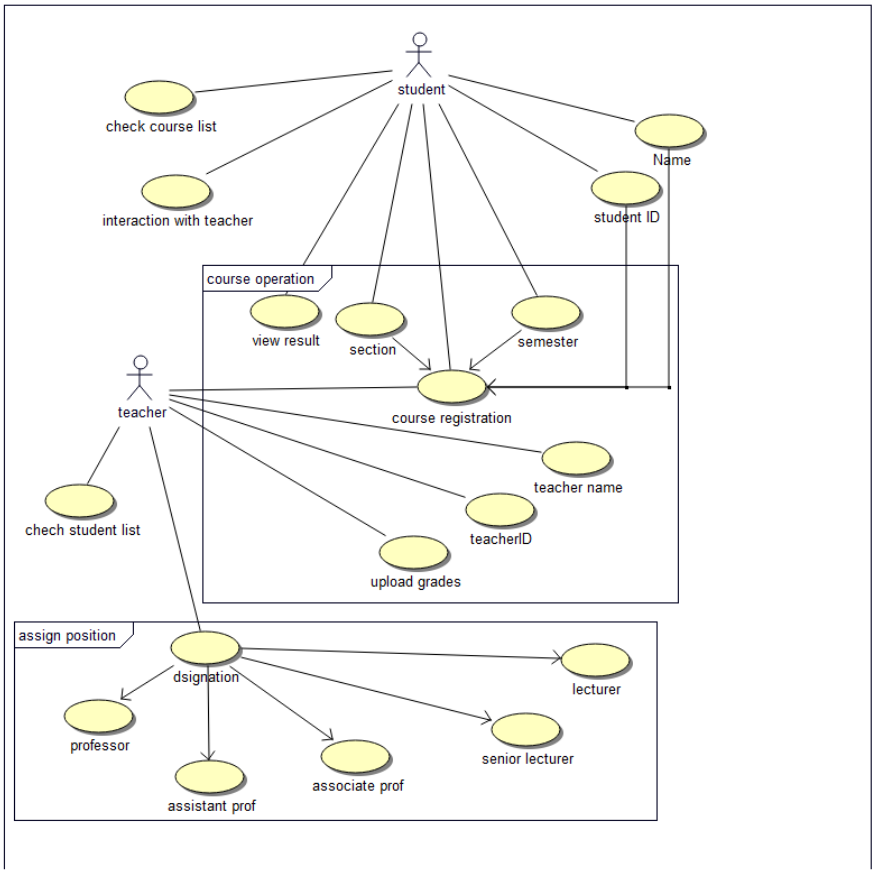


Fig-2.1: Use-Case-Diagram

2.4 Activity Diagram

Fig 3.1 Shows a typical sequence of events that takes place involving the user and the app. The user will first login to the app. then he/she will be able to view his assignment, grades and download them as pdf. Student can also request for courses.

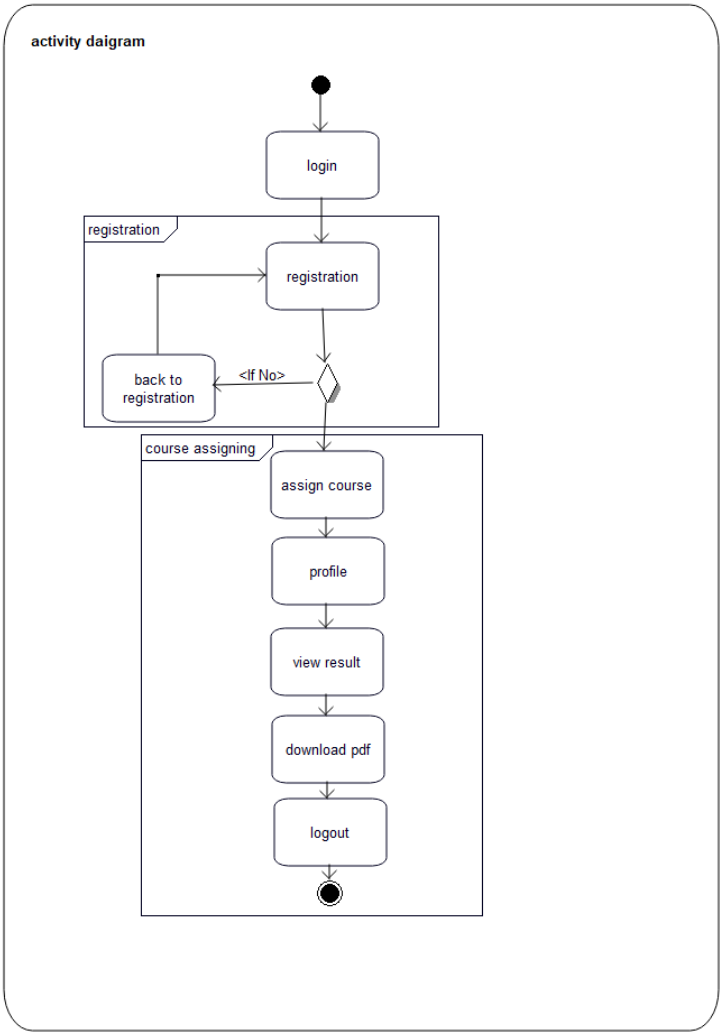


Fig-3.1: Activity-Diagram

2.5 System Operation

Fig 4.1 Shows a typical sequence of events that takes place in the "University Manager System" app when admin is logged in.

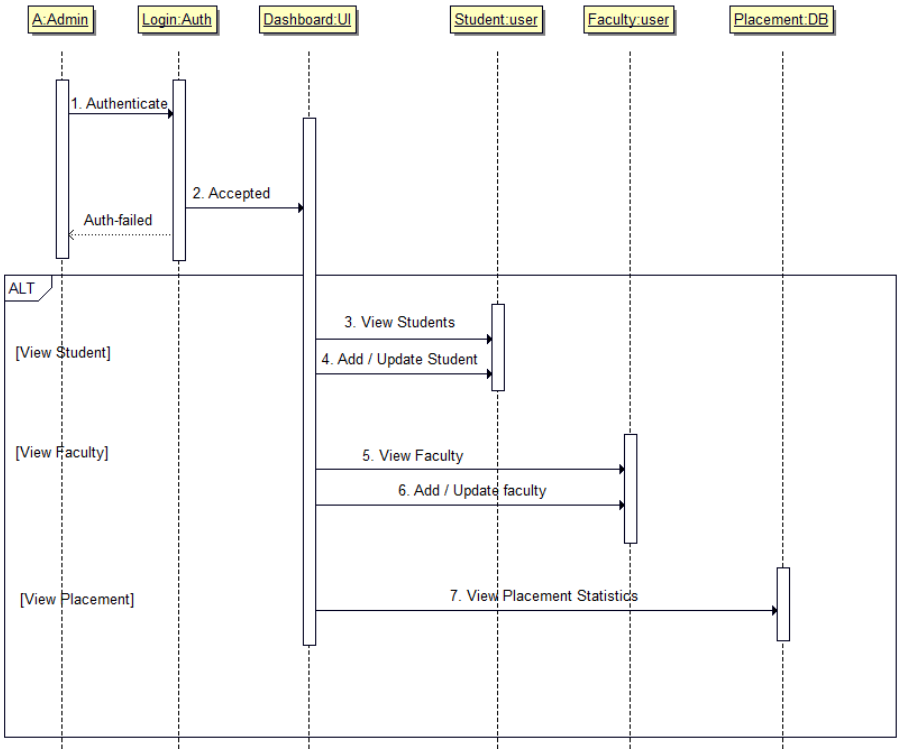


Fig-4.1: Sequence-diagram(Admin)

Fig 4.2 Shows a typical sequence of events that takes place in the "University Manager System" app when faculty and student interaction takes place.

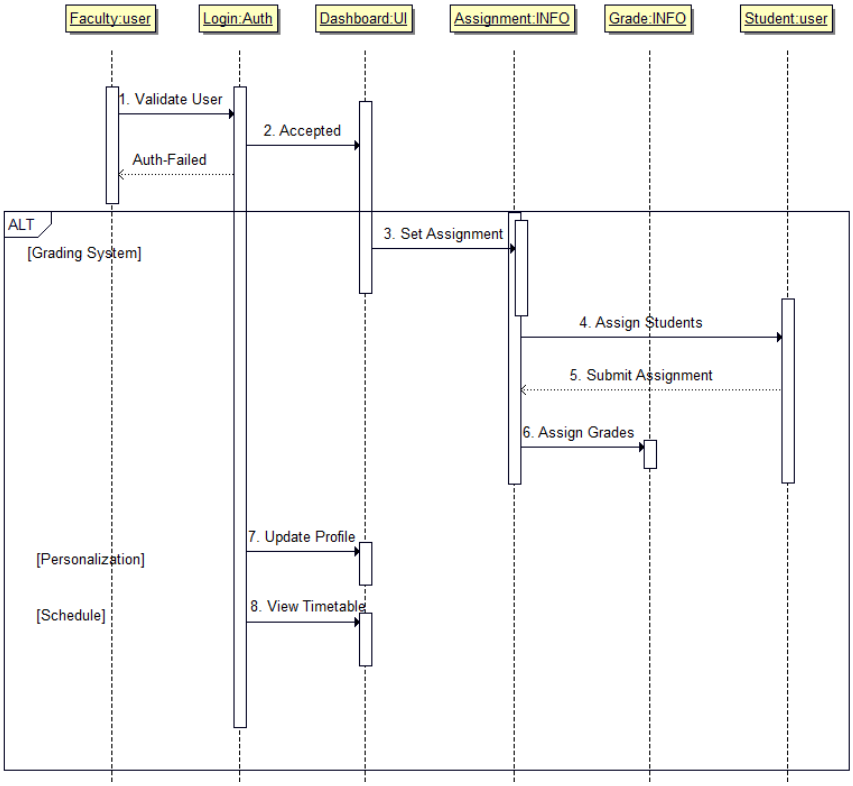


Fig-4.2: Sequence-diagram(Faculty-Student)

2.6 Subsystem Design

Following **Fig 5.1** is the class diagram showing the interface details and the subsystems in the website.

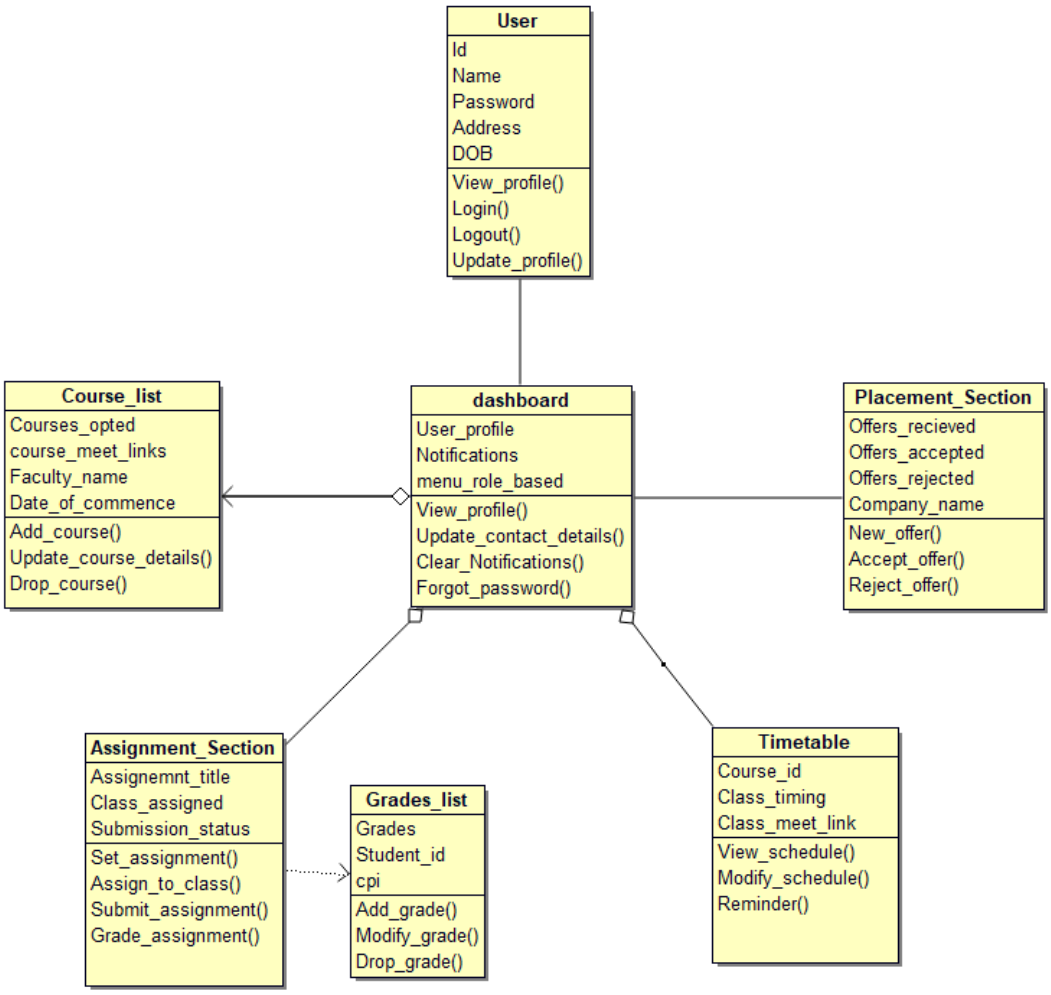


Fig-5.1: class-diagram(Interface)

2.7 State Diagram

Fig 6.1 Shows a state diagram that takes place in the "University Manager System" app when faculty and student interaction takes place. The student first logs in to the website and visits his/her dashboard. Here he can chose different options from the menu as given in the diagram.

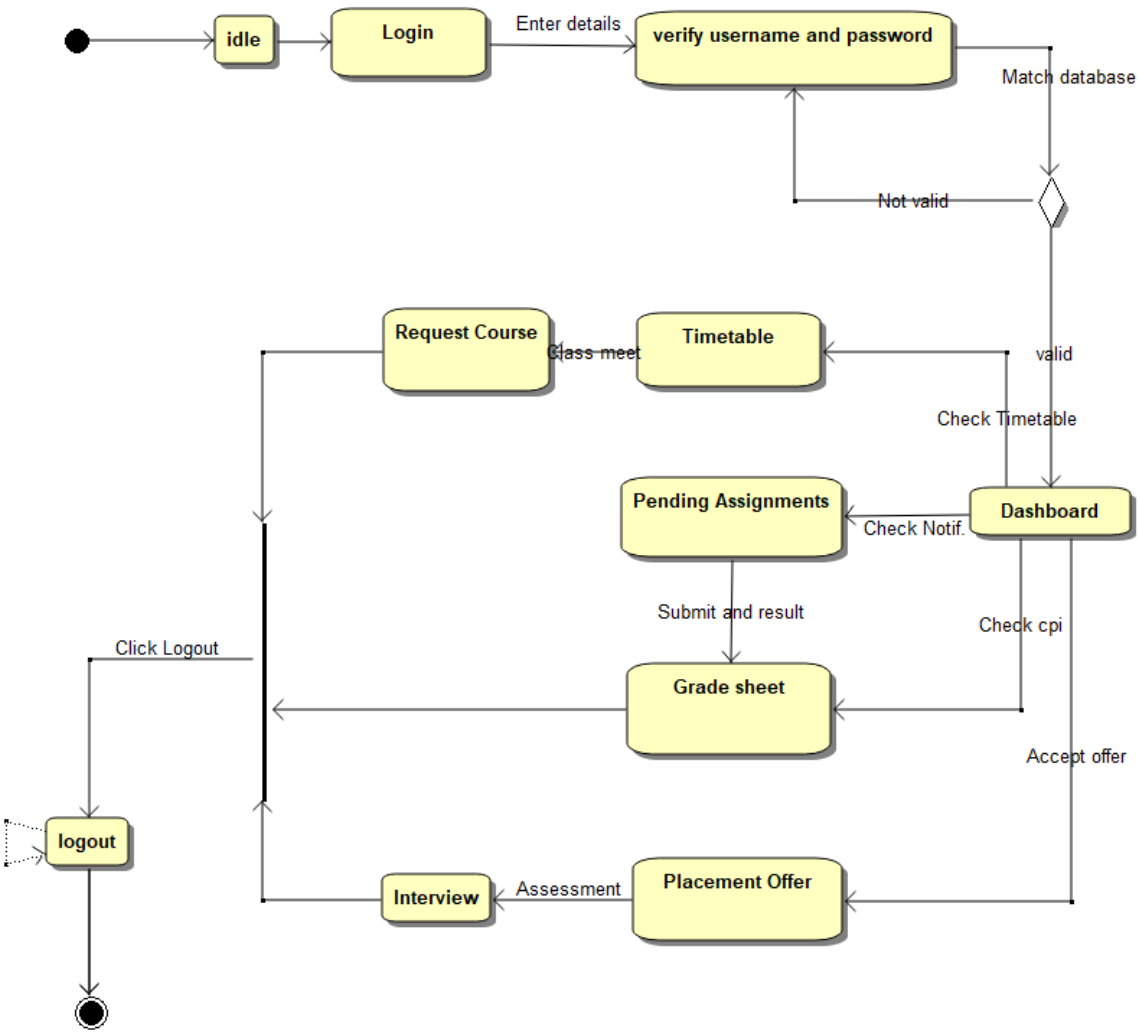


Fig-6.1: State-diagram

3 ER Model

Fig 6.2 Shows a ER diagram for the "University Manager System". The ER diagram shows the entities in the system i.e., Student, Faculty, Course, Department, Program, Assignments etc. and the relationship among them. Main entities and their attributes are -

1. Student - student_id(PK), name, address, phone, emailid, dob, password, cpi, semester, fees_paid
2. Faculty - faculty_id(PK), name, address, phone, emailid, dob, salary, position, research interests
3. Course - course_id(PK), name, description, credits, syllabus
4. Section - section_id(PK), semester, year
5. Classroom - (room_no, building)(PK), capacity
6. Department - department_id(PK), name, building, budget, contactno
7. Program - program_id(PK), name, duration
8. Assignments - assg_id(PK), description, start_date_time, end_date_time, marks, file_url

Some relationships are -

1. student dept - It describes which student is in which department
2. student assg - It describes which student is assigned which assignments and its attributes are marks obtained, date_time submitted.
3. Has assg - It tells about the assignments which belong to a particular section
4. finalgrade - This relation stores the grade obtained by the student in the particular course.
5. TA - This relationship shows which student is teaching assistant for which section
6. manages - manages relation shows which faculty is heading which department
7. department specific,program specific - These tell us about which placement opportunity is restricted for which departments and programs specifically.

8. section room - this relation shows which classroom is used for a section and also stores day and time as attributes.

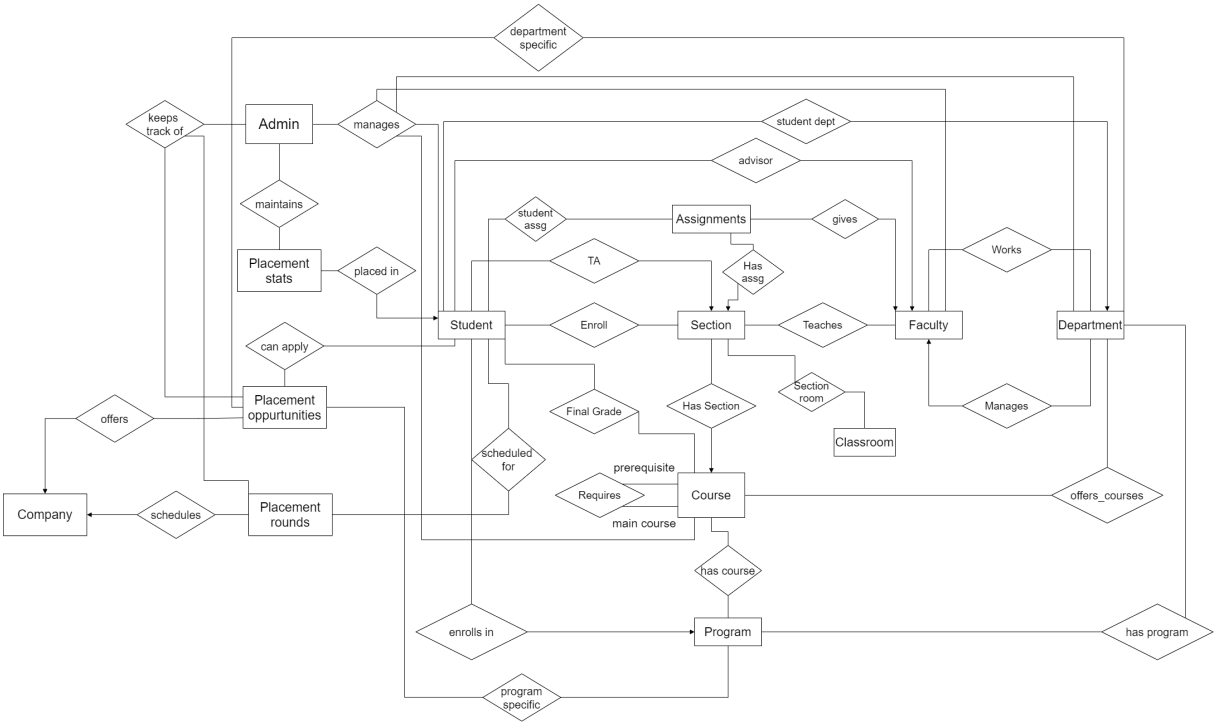


Fig-6.2: ER-diagram(ER-model)

4 Data Models

Following **Fig 8.1** is the class diagram showing how data is modelled in the website.

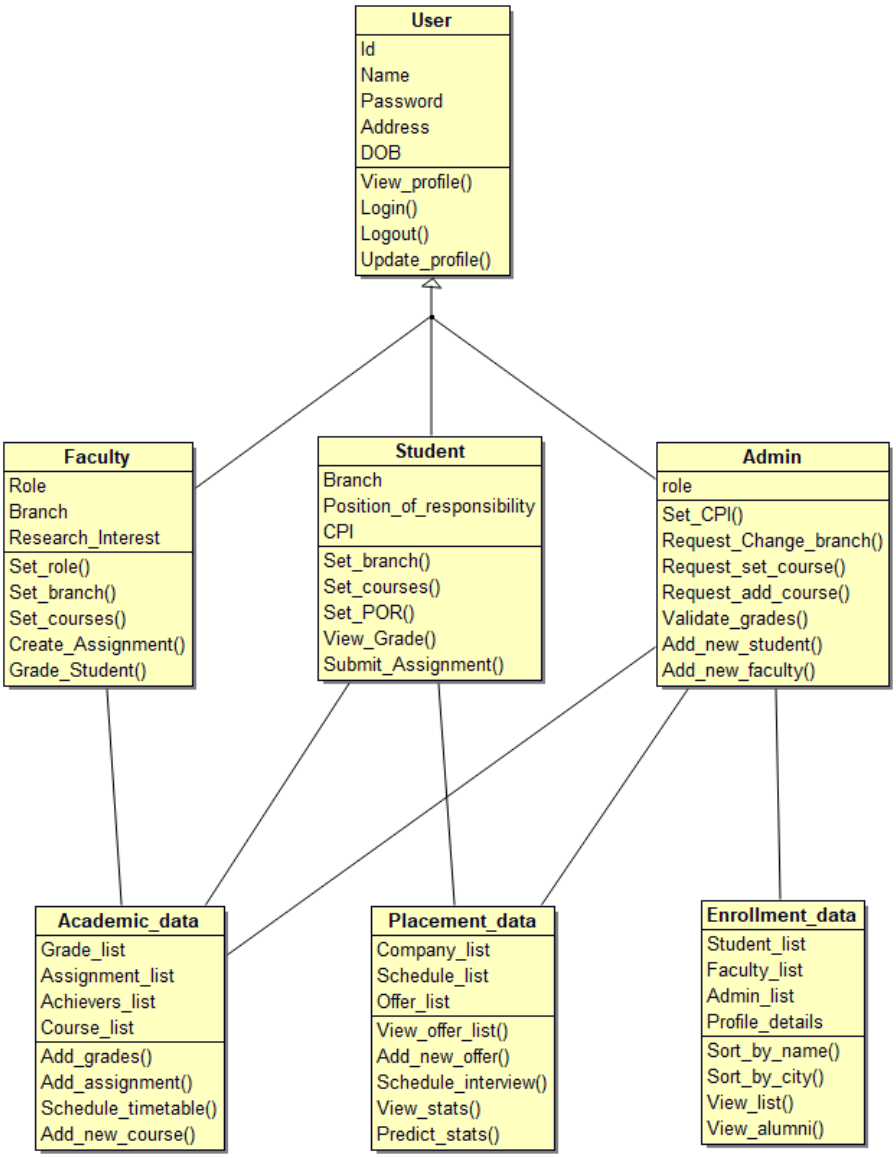


Fig-8.1: class-diagram(Data-model)

Fig 8.2 Shows a typical flow of data from login of faculty to grading assignments and updating course list with timetable. The faculty will utilize the menu from to dashboard to select required option and create assignment, The faculty will choose students from the list. The submissions will be checked and grades will be uploaded to the website and grade-sheet for students will be created.

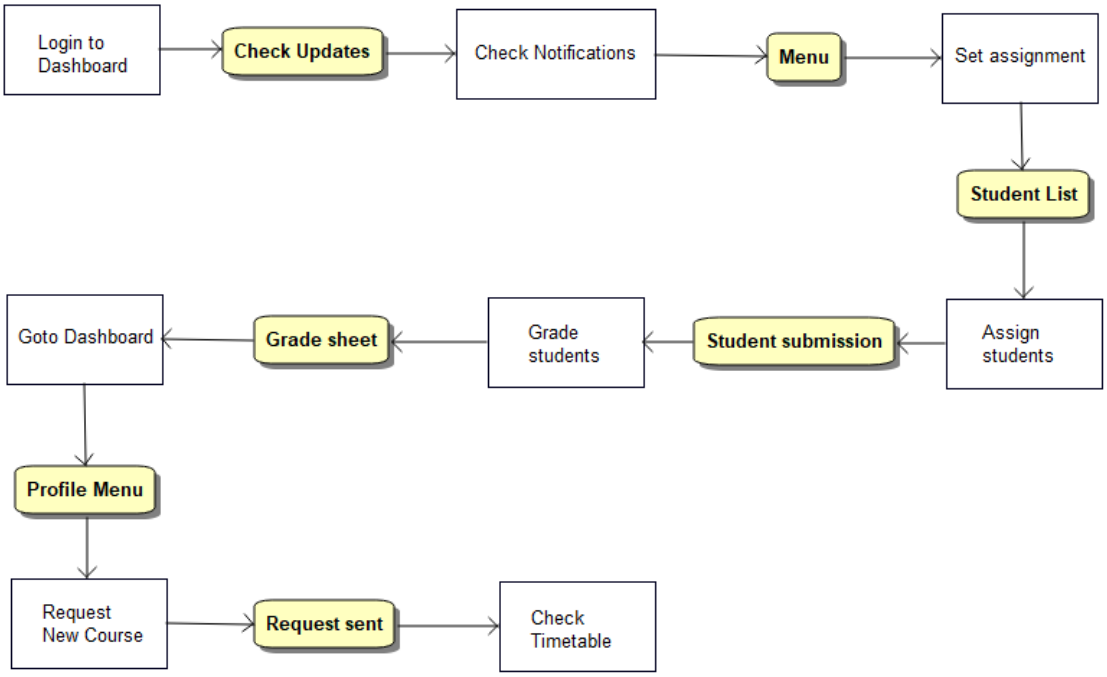
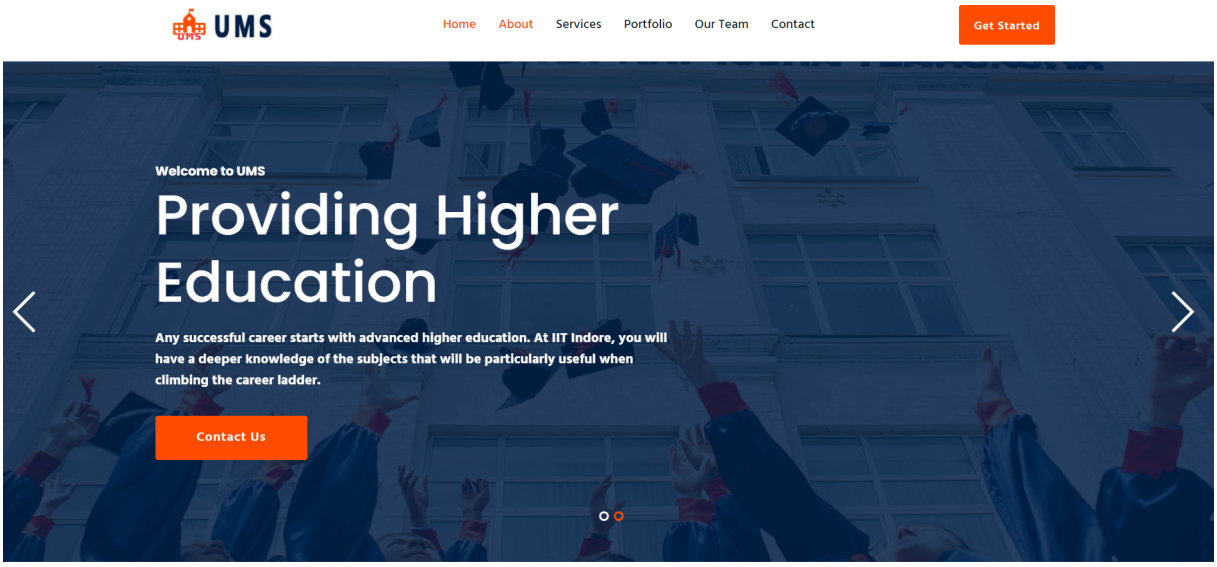
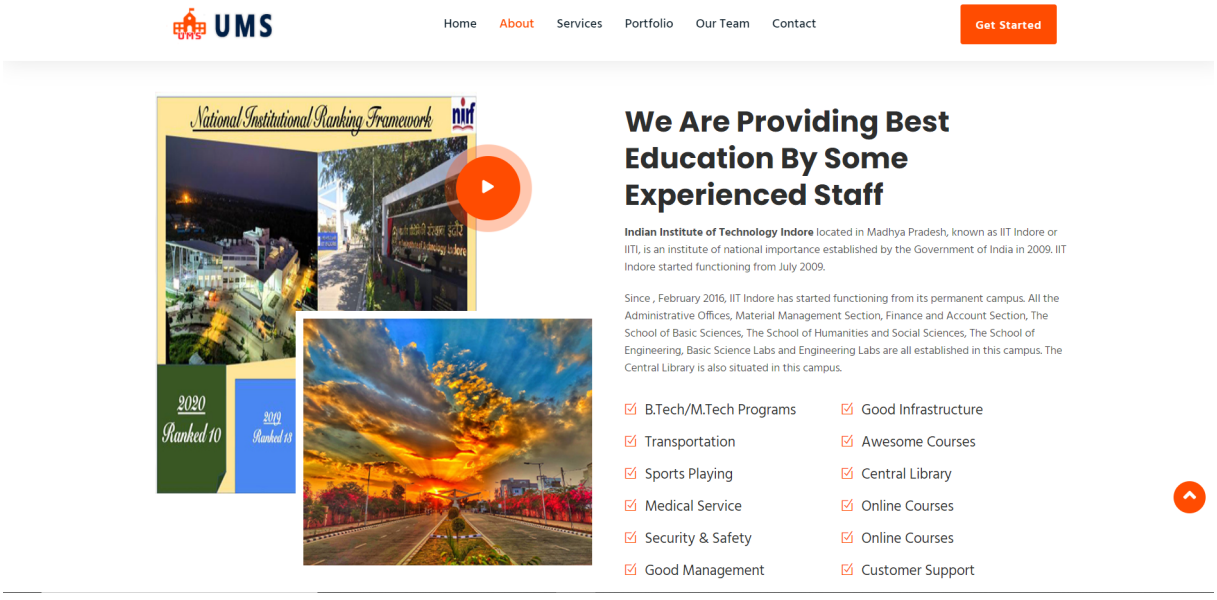


Fig-8.2: Data-Flow-diagram(Data-model)

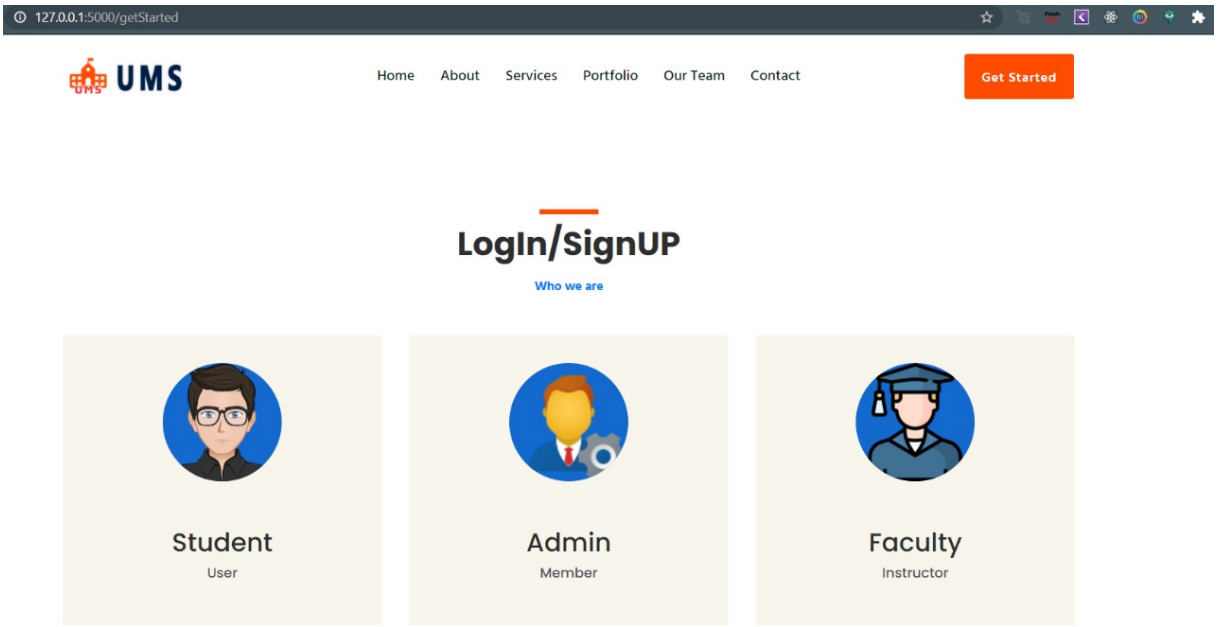
5 Wireframe and Mockups

5.1 Home Page

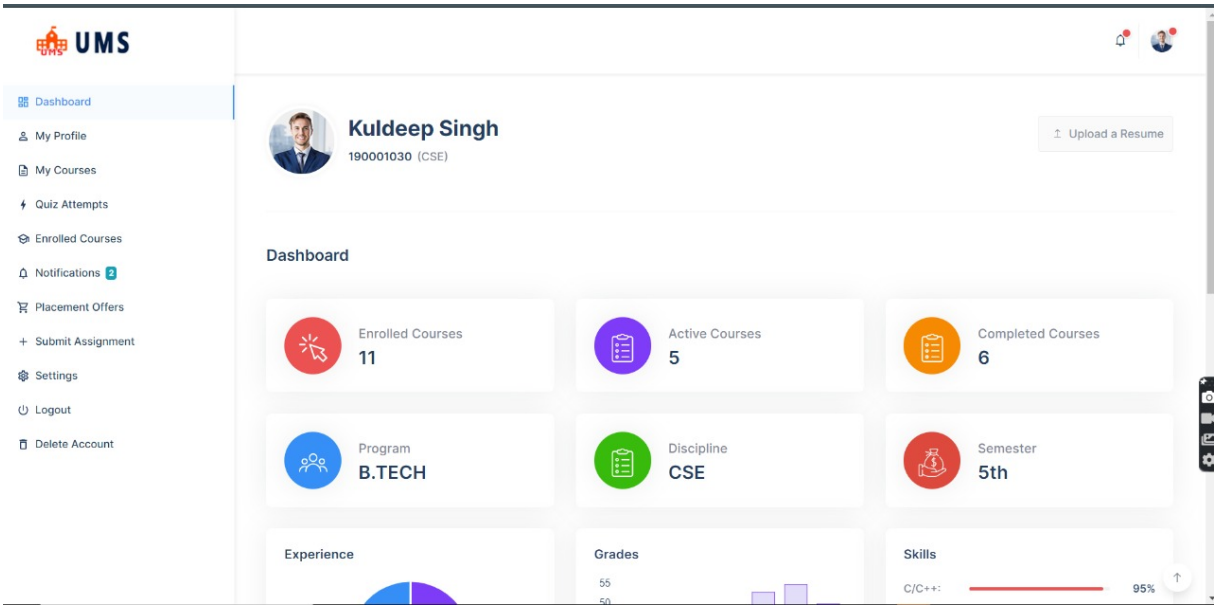




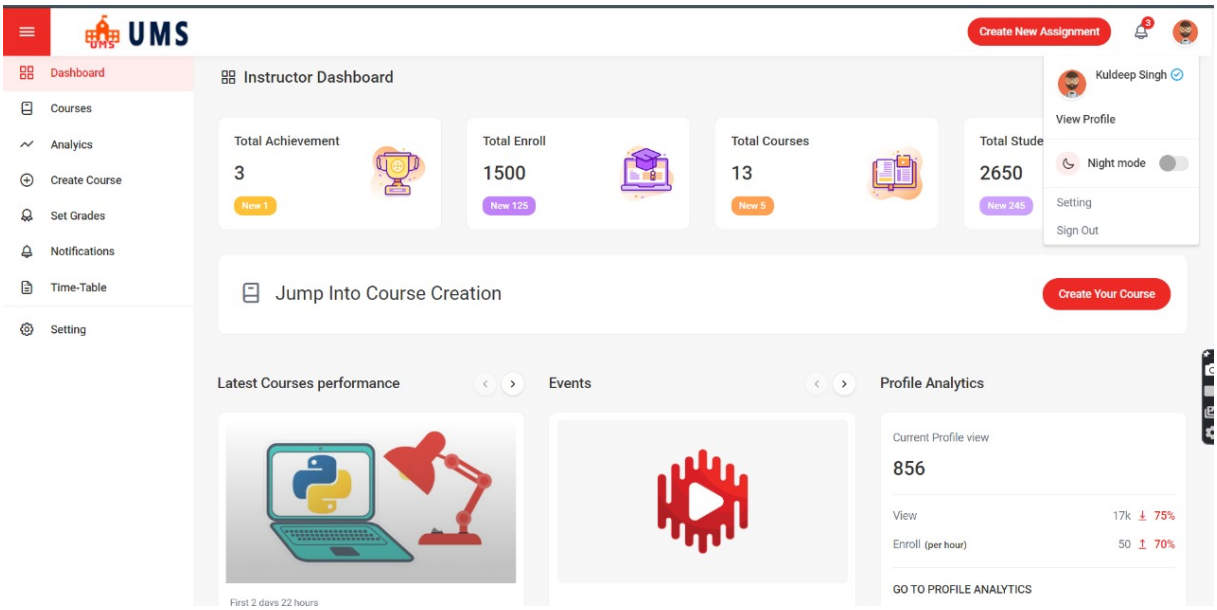
5.2 Login



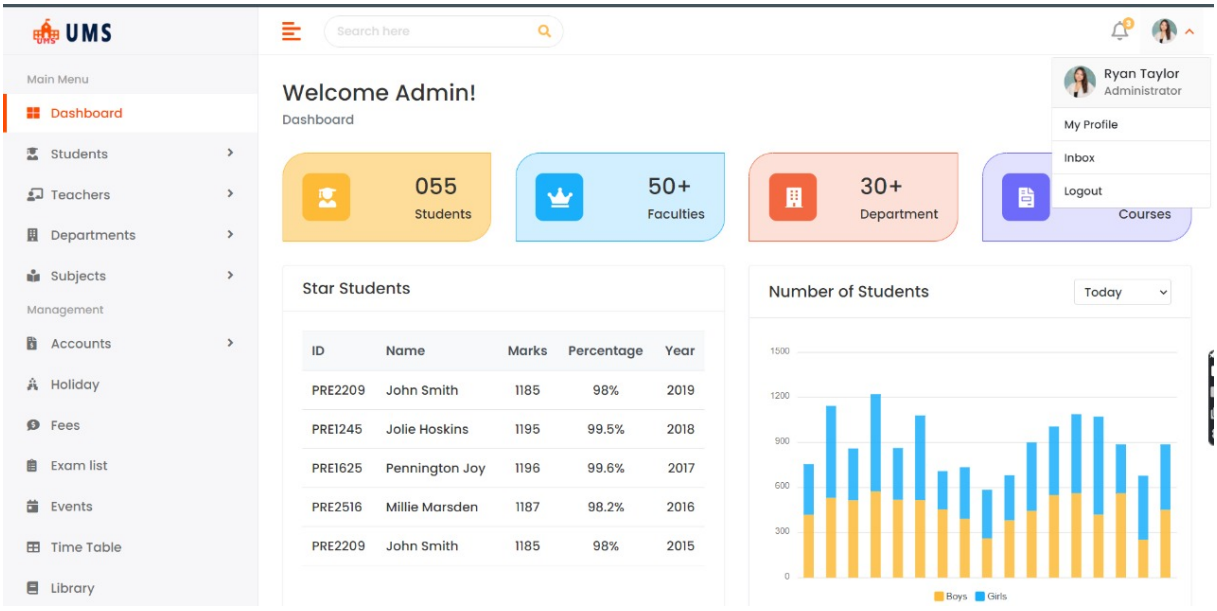
5.3 Student Dashboard



5.4 faculty Dashboard



5.5 Admin Dashboard



6 References

https://www.toptal.com/freelance/why-design-documents-matter.
https://en.wikipedia.org/wiki/Software_design_description