SOFTWARE REQUIREMENTS

SPECIFICATION

FOR

ONLINE CLASS MANAGEMENT SYSTEM

VERSION 1.0 approved

Prepared by:
HARSH SHARMA(21CS30023)
HUNNY CHANDRA(21CS30024)
SAKET JHA(21CS30044)

IIT Kharagpur

Prof Sourangshu Bhattacharya, Prof Abir Das

Contents

9 Appendix:C To Be Determined List

1 INTRODUCTION

1.1 **PURPOSE**

Geographically dispersed students and employees can receive web-based training from instructors thanks to the virtual classroom. In the virtual classroom the instructor posts assignments,recorded lectures,and gives out study material to students. The instructor can also put up tests,set up deadlines for the same and evaluate them on their performance.

Students can view recorded lectures,can message in the chat box.

In this Online or virtual learning makes the education convenient and easy for both the instructor and learner.

It saves time and money both.

It gives students a way to learn collaboratively.

The purpose of this document is to give a thorough explanation of the OCMS system's goals, features, user interface, and application. An extensive profile of the external interfaces, performance factors, and design limitations that have been imposed on the implementation are all included.

1.2 Document Conventions

The standard font used is Arial with font size 15.The titles have been represented in bold with font size 19 and font used is Arial.Important parts have been highlighted in bold.

1.3 Target Audience and Recommended Reading

This SRS is intended for programmers, project managers, writers of documentation, testers, and users.

All of the interior, exterior, functional, and nonfunctional requirements for the online class management system will be covered in the following discussion (OCMS).

As previously indicated, the following audience members include:

Project managers are responsible for overseeing the whole project.

Developers—

This team of experts uses programming languages to carry out the developer's idea.

They are in charge of all the graphical user interfaces and application components.

Testers:

 Members of this group run specific test cases against the developed system to gauge its effectiveness and predict its performance.

Documentation Writers:

Documentation writers produce user guides and other documents necessary for a system's proper configuration in a given operating environment.

OCMS USERS-those who favour the system's adoption in educational institutions. By offering insightful feedback on the initial requirement papers, they are responsible for the calibre of the software requirement specification document.

There are five subsections in this document of software requirements specifications.
Introduction in Section 1
Section 2 provides a general overview of the web application, along with details on its features, user-defined classes, operating systems, restrictions, and documentation.
Giving a brief introduction to user, hardware, software, and communications interfaces is Section 3: External Interface Requirements.
Section 4: Outlines the various features' functional needs.
A list of non-functional needs is provided in Section 5.

Section 6: Other requirements

1.4 Product Scope

OCMS is an online learning environment that serves as a replacement for traditional classroom settings and allows students who live in remote locations to participate and interact in these classes. Students benefit from the tutors' distribution of recorded lectures and study

materials. The tutor acts as a moderator in this type of interaction, directing the learning process.

Students have access to recorded videos at their convenience.

Exam results can be sent to students after instructors have graded test papers.
Management will be able to efficiently manage all resources thanks to the system's user-friendly web-based service.

1.5 References

IEEE.IEEE Std 830-1998 IEEE Recommended Practice for software Requirements specifications.IEEE Computer Society,1998

2 Overall Description

2.1 Product Perspective:

This is meant to replace an existing system (Like Microsoft Teams). Through this platform, professors and students can communicate and collaborate on any academic task. Additionally, the frontend will be used by the users to access/submit/upload assignments , join classes ,add video lectures , provide grades or any other purpose. Our software engine will enable us to aggregate the data, store the data, and generate results that the users will find useful. Django + Python will be used to build the web application, and it will interface with the SQLite3 server to store and retrieve pertinent data. The SQLite3 server, which will store and push the data used by the application, will function as the backend. Professors will be the main users of the backend.

Students	Teachers
Can see notice on common chat	Write notice for students on common chat
Submit assignments	Upload assignments
Can access student list	Can access student list
See recorded lectures	Record lectures for students
Join classes by code	Create classroom and generating class code
Change personal details	Change personal details
Timetable access	Timetable access
Communication through common chat	Communication through common chat
Access to study materials	Providing study materials

Students	Teachers
Give tests	Conduct test

2.2 Product Functions

• Maintaining Database for

1. Details of students and professors
2. semester lectures, study materials and assignments
3. marks distribution enumerate

• Professor's End
1. upload assignments/study material/recorded lectures
 2. access student list
3. evaluate assignments and assignment grades
4. write a class notice on common chat
5. can organize tests
6.Provide Recorded lectures

 • Student's list
1. submit assignments
 2. access study material/recorded lectures
3.Access student's list
 4. see assignment's grades
 5. access class notice through common chat
 6. can give tests
 7. View recorded lectures

 • Common Interface/Website
1. can create account

2. change personal details
3. change passwords

2.3 User Classes and characteristics

The System has the following user classes:

 • Professor
 • Student

The Student can create an account and then attend classes, submit assignments, access recorded lectures. The Professor sees the assignments and evaluates them. The student has a certain deadline to give his or her tests and assignments. The professor sets a deadline and ensures that each student has submitted or not.

2.4 Operating Environment
 • Windows 7
 • Windows 8
 • Windows 10
 • Linux
 • Mac OS X

 This will be a web based system.We will actually have a server to run. Stores all functions and custom user data. Custom ends will just be a GUI. Every one of the modules referenced in requirements.txt are expected to be introduced for application to work.

2.5 Design and Implementation Constraints

• The framework has reliance on the information base .
 • Web association is a requirement for this framework ,hence client necessities to have great organization association with associate with our web interface
 • High level security highlights haven't been presented.
 • This product was created in Django(python framework)+ HTML+CSS+JavaScript.
 • This product requires explicit modules to be introduced before hand.
• Django rendition ≥ 2.2 is required.
• For the Windows working framework , Microsoft Visual c++ fabricate devices ≥ 14 are required.
 •Live talks are not set in stone.

2.6 User Documentation
License and User Manual will be provided along with the software.

2.7 Assumption and Dependencies
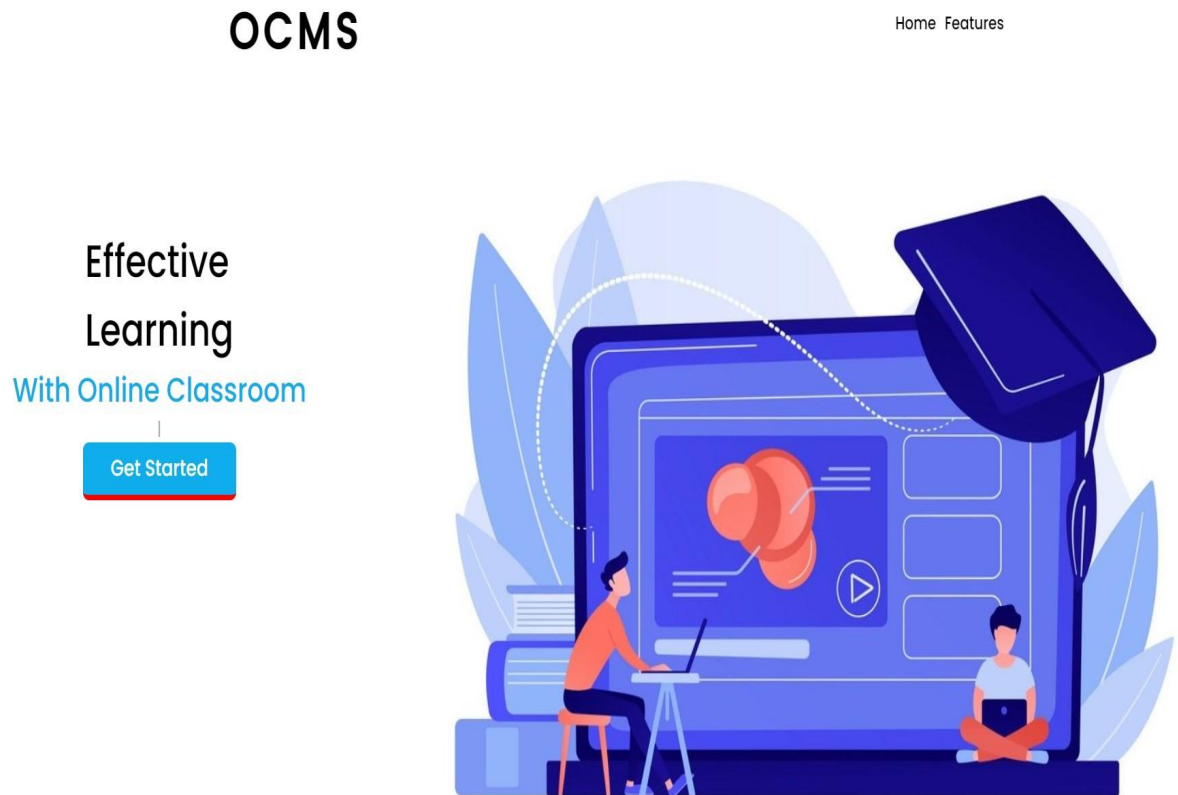 Assumption: The user will have the basic resources required for this web application :-
• Active Internet Connection
• A browser to access internet
 • A desktop system.

Other than these, the web application software depends on various APIs, which is required.

3 External Interface Requirements

3.1 User Interfaces

1. Starting Page: Brief introduction about the platform

2.

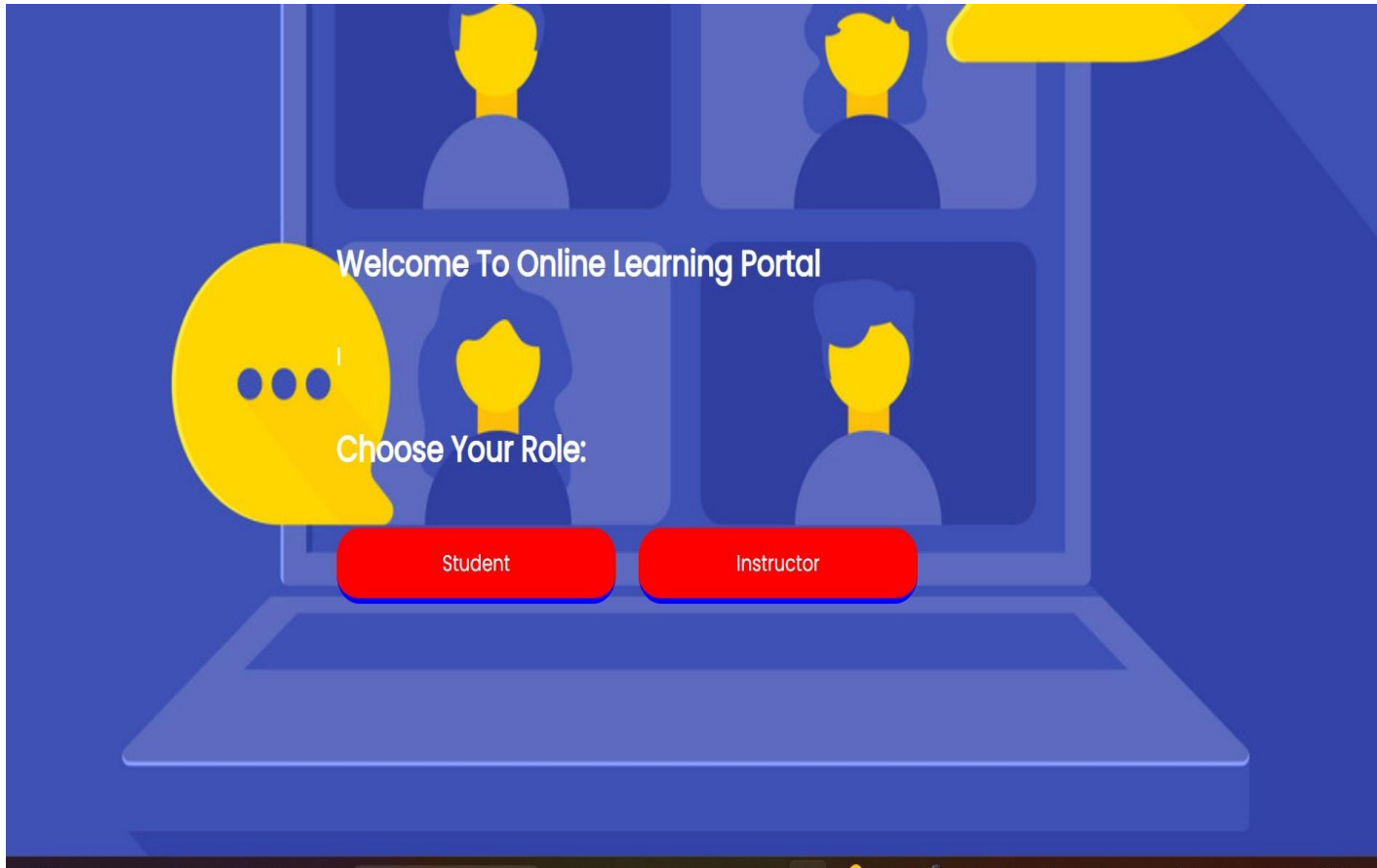2.You will be asked to choose between instructor and Student and will have to login into the respective field:
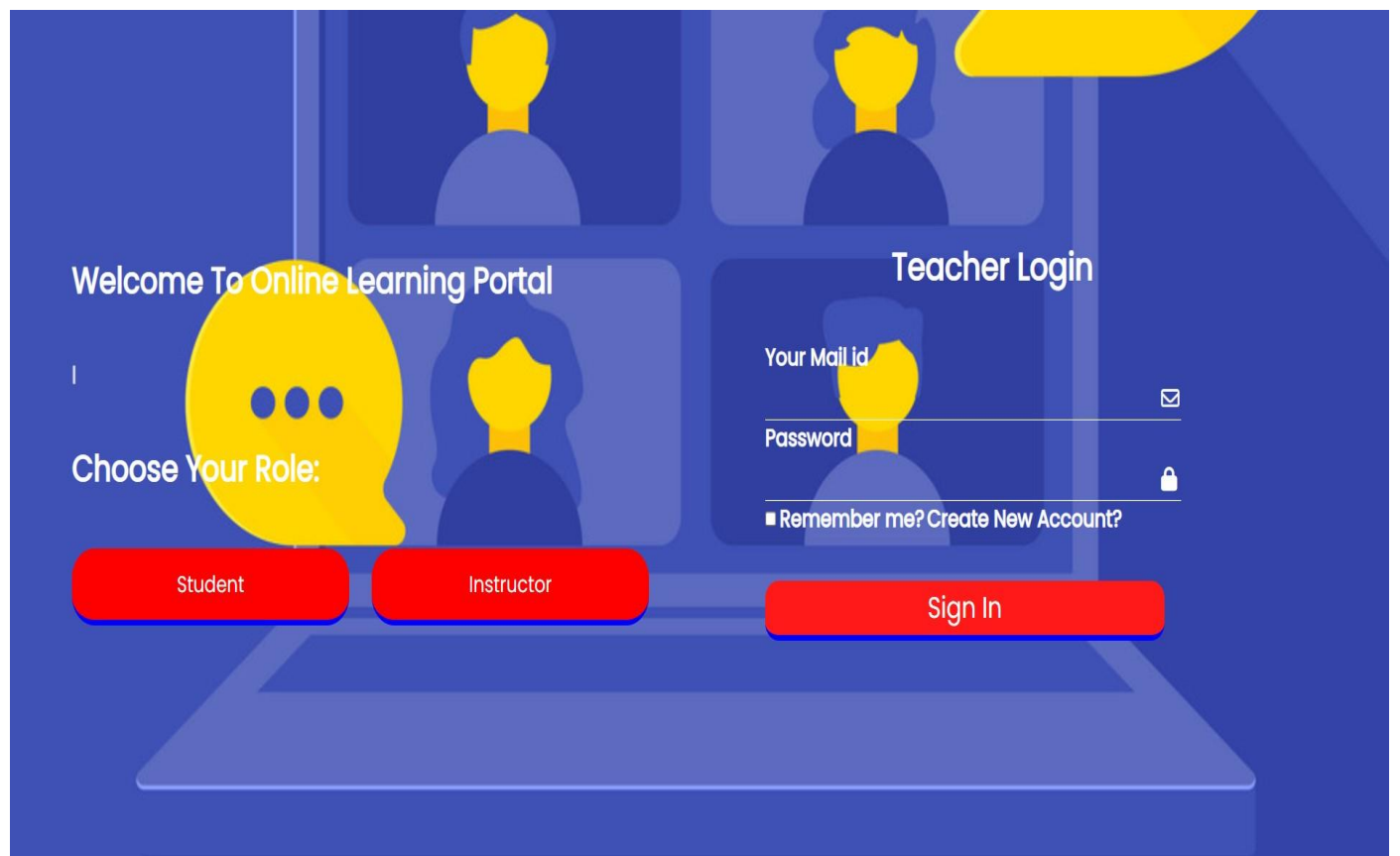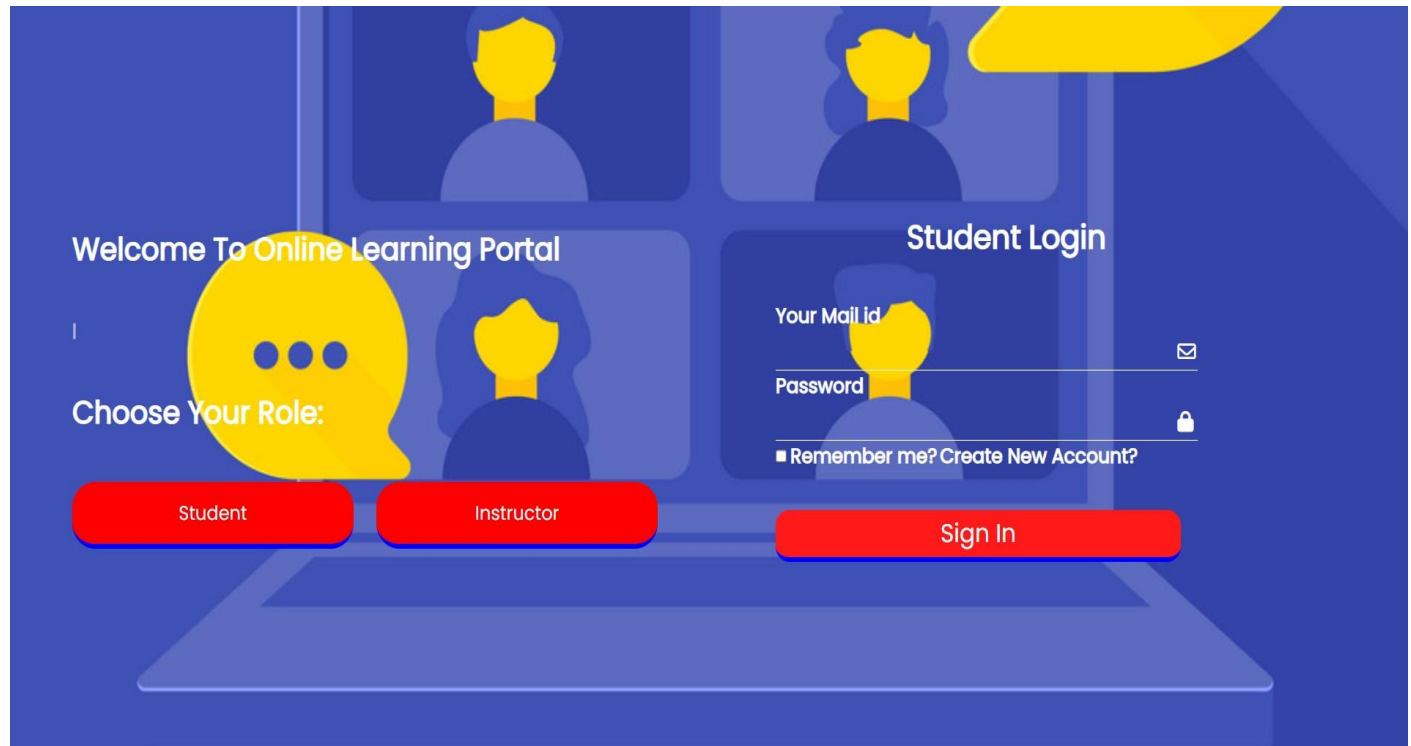
Welcome To Online Learning Portal

Choose Your Role:

Student   Instructor

3.Login:

Welcome To Online Learning Portal

Choose Your Role:

Student

Instructor

Student Login

Your Mail id

Password

✉

🔒

▪ Remember me? Create New Account?

Sign In

Welcome To Online Learning Portal

Choose Your Role:

Student

Instructor

Teacher Login

Your Mail id

Password

✉

🔒

▪ Remember me? Create New Account?

Sign In

4.Sign up : for new users

Sign Up

Name

Mail Id

Create a password

Confirm Password

Create Account To Start Learning

Mon Mar 27 2023 04:19:08

Let's Get Started

5. Available features:

The features for students and professors are different.

For professor

OCMS

Search for.....

Assignment

Test

Class Room

Time Table

Create Room

Students

Update Profile
Activity
Profile
Logout
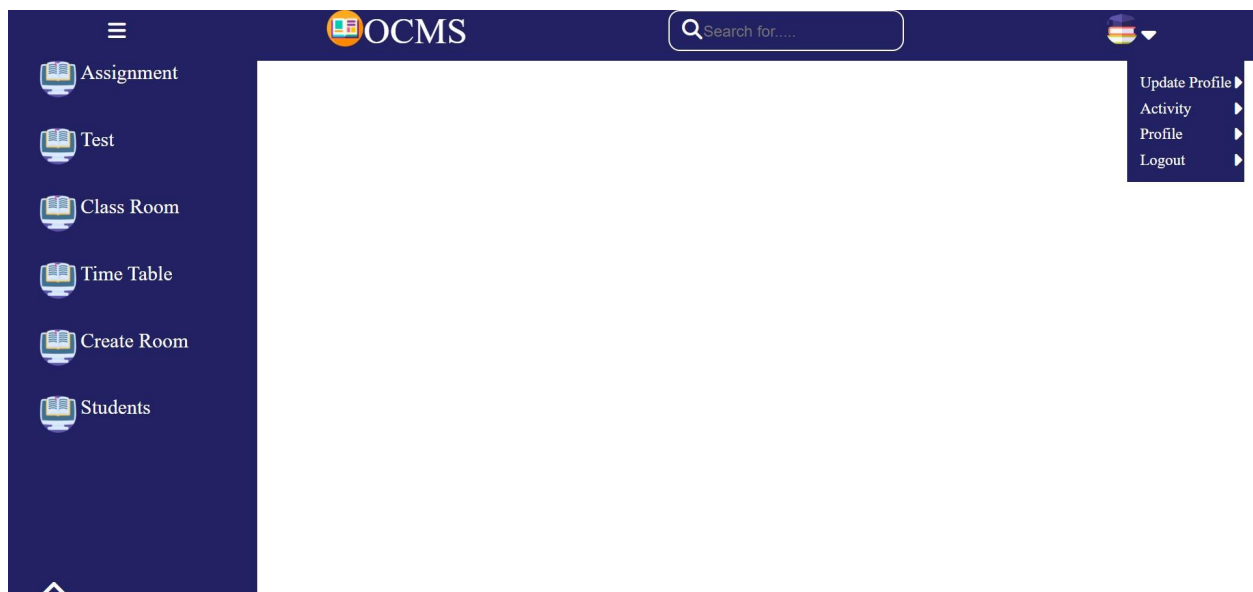
For students
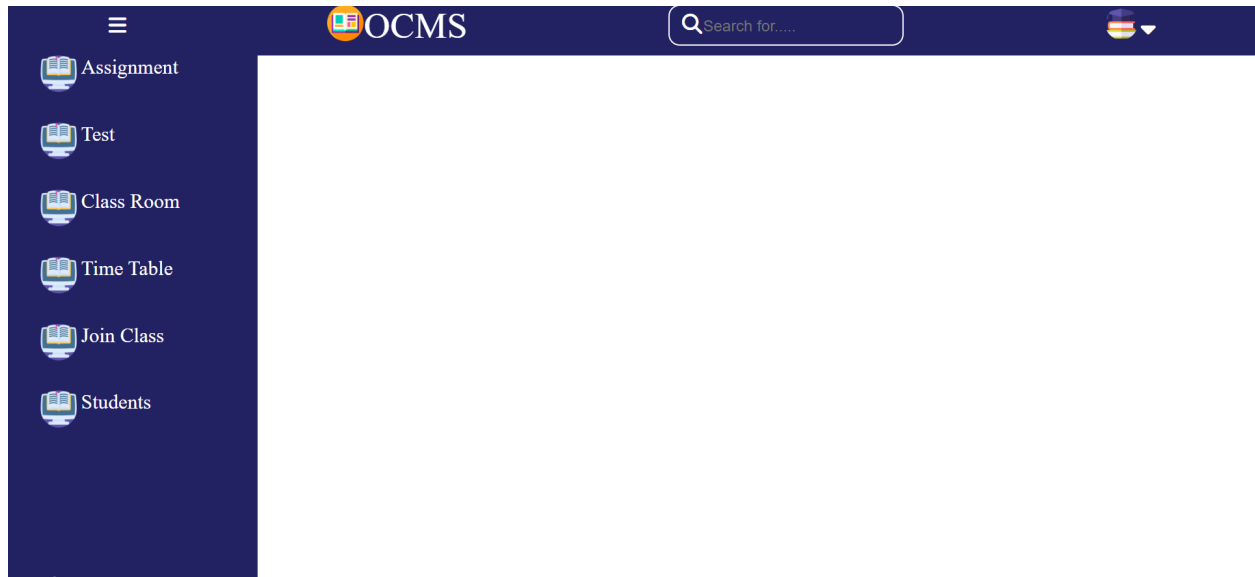
3.2 Hardware Interfaces

Our web application will be gotten to by means of an internet browser on a work area or PC. Expecting the client is working in a standard PC climate, the client can connect with the framework through mouse snaps and console inputs. There could be no other extraordinary equipment interface necessities.

3.3 Software Interfaces

The GUI will be made by HTML, CSS,Javascript. Python would be utilized for the handling on the server. Information base will be made utilizing SQLite3. Class Director utilizes the multi-client idea of Django where understudies and teachers are various kinds of client and have various functionalities.

3.4 Communication Interfaces

The correspondence will occur across an organization in a scrambled configuration. While getting to our web application by means of an internet browser on a PC/work area, the client will utilize HTTPS to get to the web interface. Our framework will be gotten to by the client through our site.

4 System Features

This subsection presents the identified functional requirements for the online class management system.

4.1 SignUp & Login

4.1.1 Description and Priority

This is the initial step for each user to do in order to begin using these services. By joining up for these services, the user must create an account.

4.1.2 Stimulus/Response Sequences

The user must provide his personal, educational, and professional information. The system will establish an area for the user and his data.

4.1.3 Functional Requirements

• Requirement 1: Request user information.
 • Requirement 2: Allocate memory space on the server.
 • Requirement 3: A function that acts as an extension to keep the user logged in.
 • Requirement 4: A function for creating a user's profile.

4.2 Uploading Assignments/Video Lectures/class notes

4.2.1 Description and Priority

This will be the most used features of our web application. Because after every class lecture,notes and probably assignments need to be uploaded.This is the most important feature after login and sign up.

4.2.2 Stimulus/Response Sequences

Required files need to be submitted by the professor.The files will be retrieved by the students.

4.2.3 Functional Requirements

 • Requirement 1: Request server information.
 • Requirement 2: Allocate memory space on the server.
 • Requirement 3: A function that provides the user with an option to upload any file.
 • Requirement 4: A function for creating a user's resume.

4.3 Maintenance

4.3.1 Description and Priority

The maintenance of web applications is just as important as their development. That's because online apps that are malware-infected might not function as intended, making the entire development

process useless. Any company will find it too expensive and inconvenient to delay this important action. Obsolete technology and incompatible data can significantly hurt web traffic and earnings.

4.3.2 Stimulus/Response Sequences

Corrective actions like technological updates, debugging, patch updates, and functional optimisation are necessary according to current web trends.

4.3.3 Functional Requirements

To Be Determined

5 Other Nonfunctional Requirements

5.1 Performance Requirements

1. A local server will be used to build the software.
2. Depending on the processor, the product will be extremely quick.
3. Performance will be influenced by hardware elements.
4. When combined with high speed internet, the server response time should be quick (within 2 seconds).

5.2 Safety Requirements

1. No threat the system may issue poses a threat to anyone. There will undoubtedly be risks associated with the internet because it is a web-based application. These should be reduced as much as is practical.

5.3 Security Requirements

1.Only authorised Institute workers are able to access the system because of its login features. Depending on who the knowledge is meant for, different permissions are given for reading, writing, and changing.
2. Some of the data can be read after authenticating the user's credentials.

5.4 Software Quality Attributes

1. Viability
The item's numerous renditions ought to be easy to keep up with. It ought to be easy to add code to a current framework and to redesign for new elements and innovations as they become accessible. Upkeep ought to be reasonable and straightforward.
2. Adaptability
It ought to be versatile to different items with which it should lock in. It ought to be easy to coordinate with other standard outsider components(APIs). Well known programs, for example, Chrome and Firefox ought to execute the web application without a hitch.This web application offers both fundamental and high level usefulness to its clients.
3. Convenience
This might be decided concerning ease of use. The application ought to be easy to utilize. It ought to be easy to learn. The route ought to be clear. It could be used by the two trained professionals and normal individuals in light of its very much planned and easy to use interface.

5.5 Business Rules

This web application ought to possibly be utilized on the off chance that a reasonable agreement understanding has been reached with the firm that made it. In case of an issue, the client ought to tell the undertaking supervisor immediately. Without earlier assent, the web application ought not be moved to an outsider.

6 Other Requirements

1. Copyright and licensing are necessary.

2. An internet platform will be required to host the database

7 Appendix A:Glossary

HTML - HyperText Markup Language

CSS - Cascading Style Sheets

SQLite3 - SQLite is atomicity, consistency, isolation, durability (ACID) compliant. This embedded relational database management system is contained in a small C programming library and is an integral part of client-based applications.

OCMS - Online Class Management System

IEEE - Institute of Electrical and Electronics Engineers

HTTPS - Hypertext Transfer Protocol Secure
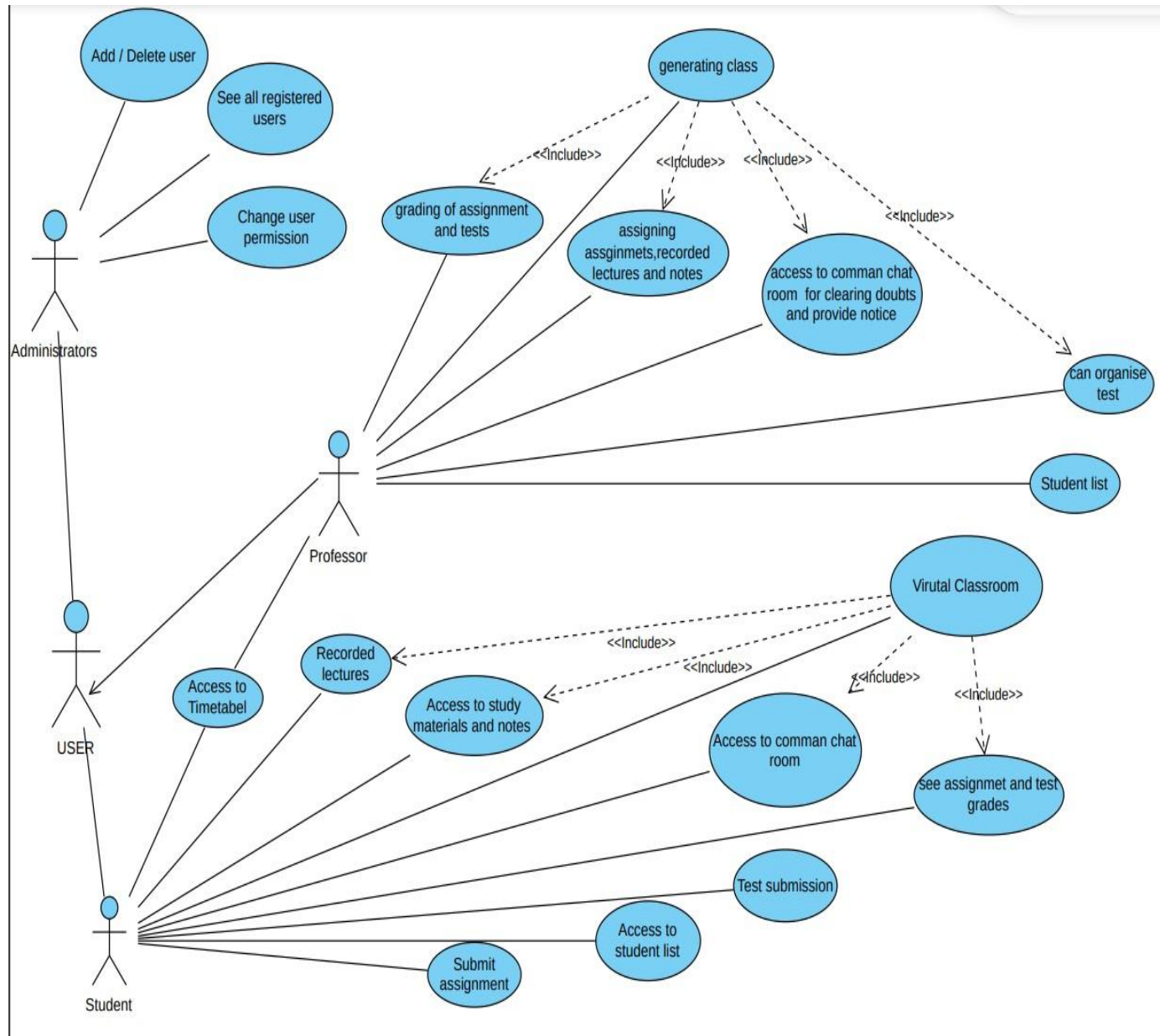
API - Application Programming Interface

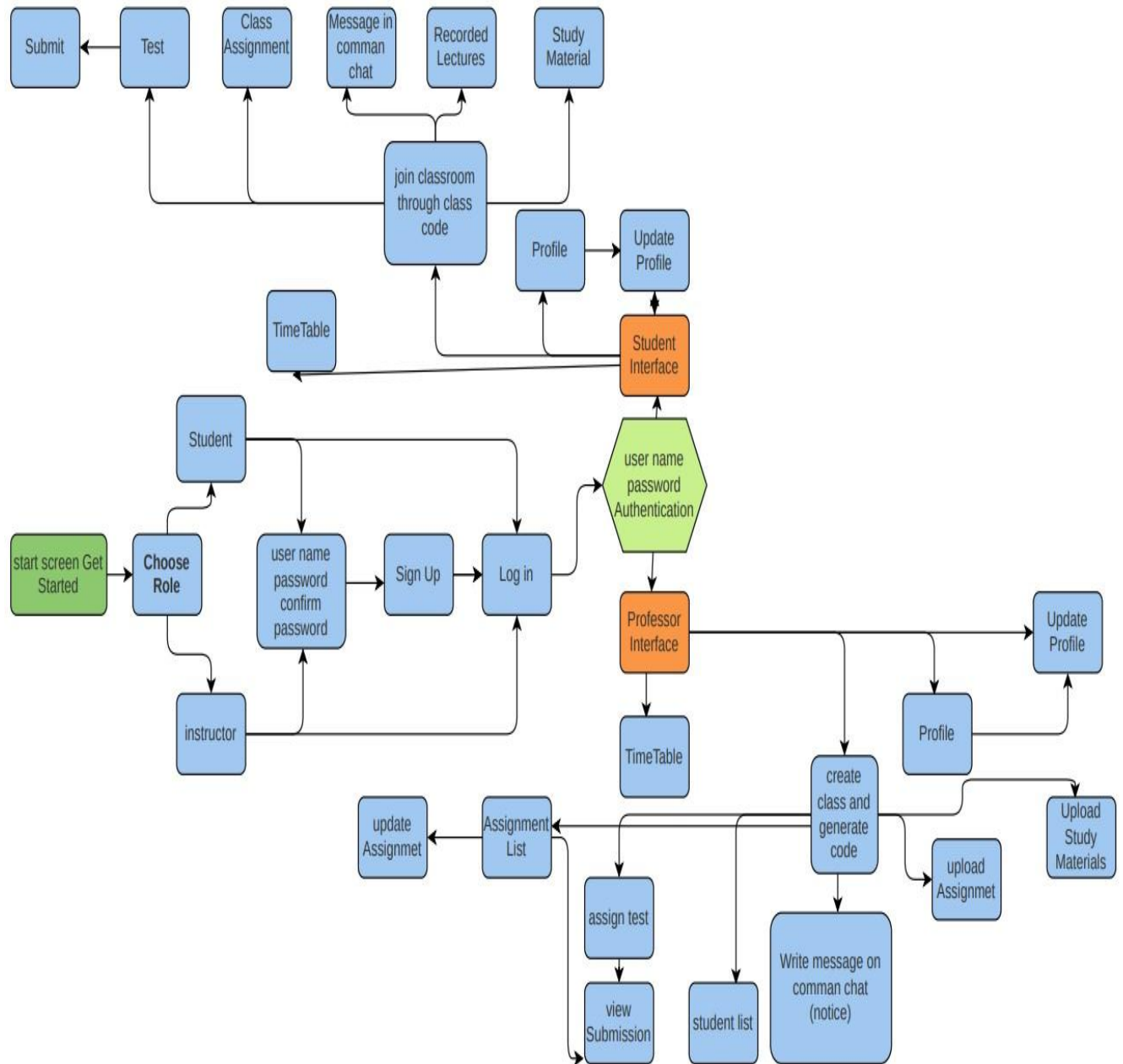UML - Unified Modelling Language

TBD - To Be Determined

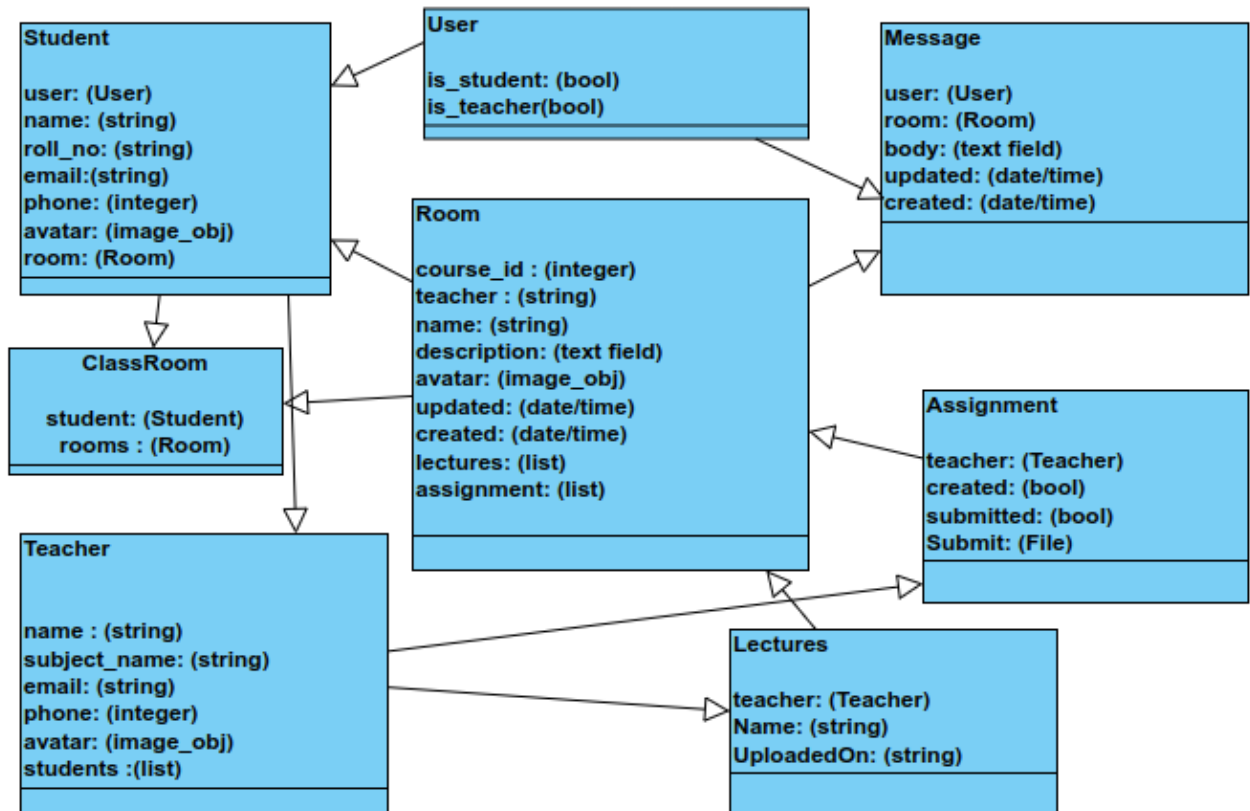# 8 Appendix B:Analysis Models

8.1 UML Diagram

8.1.1 use case diagram

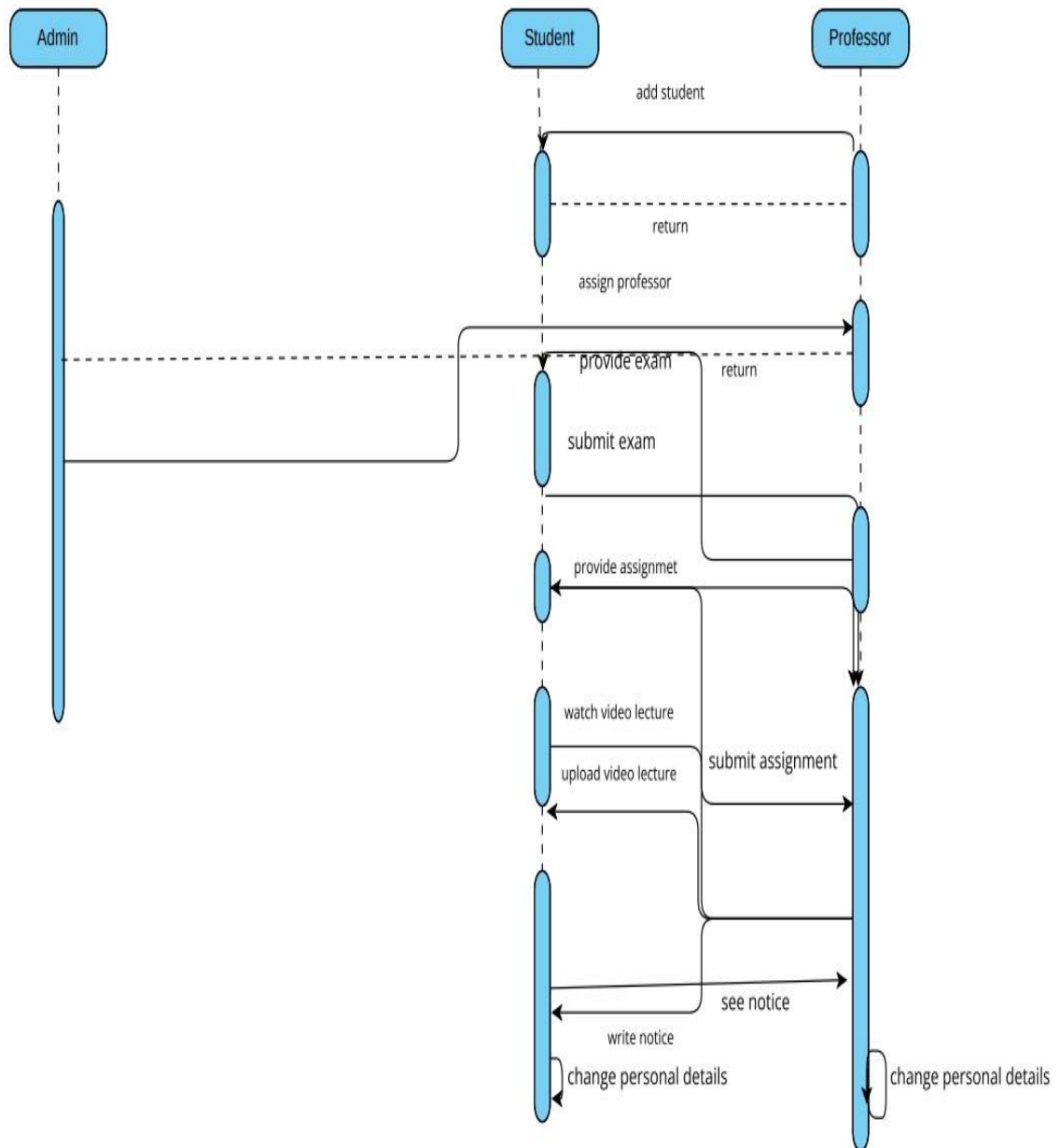# 8.1.2 Interface diagram

8.1.3 Class Diagram

Student

user: (User)
name: (string)
roll_no: (string)
email:(string)
phone: (integer)
avatar: (image_obj)
room: (Room)

User

is_student: (bool)
is_teacher(bool)

Message

user: (User)
room: (Room)
body: (text field)
updated: (date/time)
created: (date/time)

Room

course_id : (integer)
teacher : (string)
name: (string)
description: (text field)
avatar: (image_obj)
updated: (date/time)
created: (date/time)
lectures: (list)
assignment: (list)

ClassRoom

student: (Student)
rooms : (Room)

Assignment

teacher: (Teacher)
created: (bool)
submitted: (bool)
Submit: (File)

Teacher

name : (string)
subject_name: (string)
email: (string)
phone: (integer)
avatar: (image_obj)
students :(list)

Lectures

teacher: (Teacher)
Name: (string)
UploadedOn: (string)

8.1.4 Sequence Diagram

9 Appendix C:To Be Determined List