

---

# SOFTWARE REQUIREMENTS SPECIFICATION

for

## Online Classroom Management System (OCMS)

Version 1.0

Prepared by: 1. Sumit Kumar(22CS30056)

2. Aviral Singh (22CS30015)

3. Sagnibha Roy (22CS10064)

March 25, 2024

# Contents

1 Introduction	
1.1 Purpose .....	4
1.2 Document Conventions.....	4
1.3 Intended Audience and Reading Suggestions.....	4
1.4 Product Scope.....	4
1.5 References.....	4
2 Overall Description	
2.1 Product Perspective.....	5
2.2 Product Functions.....	5
2.3 User Classes and Characteristics.....	5
2.4 Operating Environment.....	6
2.5 Design and Implementation Constraints.....	6
2.6 User Documentation.....	7
2.7 Assumptions and Dependencies.....	7
3 External Interface Requirements	
3.1 User Interfaces.....	8
3.2 Hardware Interfaces.....	8
3.3 Software Interfaces.....	8
3.4 Communications Interfaces.....	9
4 System Features	
4.1 User Authentication and Security.....	10
4.2 Virtual Classroom and Messaging Interface.....	10
4.3 File Sharing System for Professors.....	10
4.4 Personalized Calendar for Students and Professors.....	11
4.5 Assignment Uploading and Evaluation.....	11
4.6 Teachers' Functions.....	12
4.7 Students' Functions.....	13
5 Other Nonfunctional Requirements	
5.1 Performance Requirements.....	15
5.1.1 Server Up-time.....	15
5.1.2 Response Time.....	15
5.1.3 System Dependability.....	15
5.2 Security Requirements	
5.2.1 Secure Login.....	15
5.2.2 Verifiable Access.....	15
5.2.3 Student Information Must be Secure.....	15
5.2.4 Secure File Sharing.....	15
5.2.5 Secure Communication Channel.....	15
5.3 Software Quality Attributes	
5.3.1 Reliability.....	16
5.3.2 Information Accessibility.....	16
5.3.3 Maintainability.....	16
6 Other Requirements	

6.1 Appendix A: Glossary.....17

6.2 Appendix B: Analysis Models.....17

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to outline the design and functionality of an Online Class Management System (OCMS), catering to the requirements of educational institutions and to automate administrative tasks related to class scheduling, content sharing, communication, and grading, enhancing efficiency and organization for both students and professors. This is the first version of the application. The application's scope covers an end-to-end system that is self-contained, possessing all the required interfaces and a corresponding backend for interacting with the system to achieve various management tasks.

## 1.2 Document Conventions

Bold text is used for software frameworks and tools. Italicized text is used for specific end-users of the application.

## 1.3 Intended Audience and Reading Suggestions

This SRS is for developers and testers who will be directly working on the implementation of the application. It is recommended that the reader acquaint themselves with common practices in educational institutions' class management systems.

## 1.4 Product Scope

The objective of this application is to streamline class management for professors and Students in an educational setting.

## 1.5 References

- MS Teams
- Google Classroom

## 2 Overall Description

### 2.1 Product Perspective

This application facilitates centralized management of virtual classrooms for students and professors. It enables features such as class scheduling, file sharing, messaging, and grade tracking via a user-friendly web interface. It ensures secure user authentication as well. It operates independently, without reliance on or impact on other software systems within the educational institution. By automating classroom management tasks, OCMS enhances efficiency, reduces errors, and fosters an optimal environment for teaching and learning.

### 2.2 Product Functions

- |                               |   |
|-------------------------------|---|
| 1. User authentication        | 6. Classroom creation                     |
| 2. Classroom Doubt Discussion | 7. File sharing                           |
| 3. Grade tracking             | 8. Class Calendar                         |
| 4. Notification system        | 9. Assignment submission                  |
| 5. Administrative controls    | 10. Live Class Conduction ,Screen Sharing |

### 2.3 User Classes and Characteristics

The following user classes have been differentiated based on the subset of product features used. Each user class possesses a different interface into the application and are exposed to functions that are relevant to them.

Professors: - Create classes, schedule sessions, upload content, assign TAs, evaluate submissions, communicate with students.,conduct live classes

Students: - Attend classes, access uploaded content, submit assignments, participate in discussions, check grades,join online classes

TAs: - Create classes, schedule sessions, upload content, evaluate submissions, communicate with students.

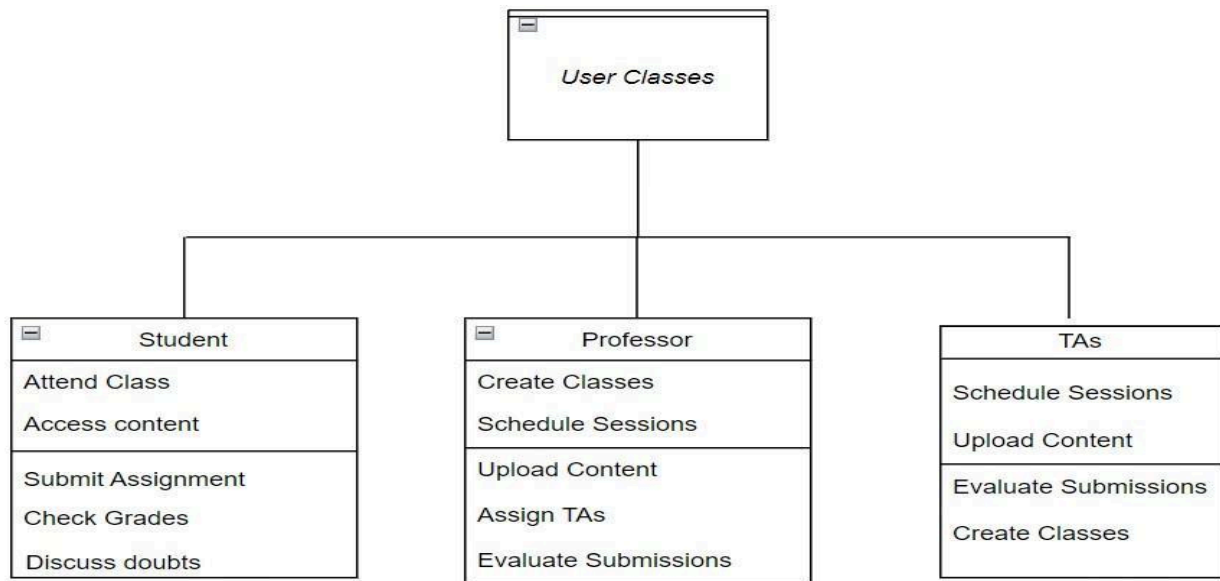


Figure 2.1: User classes

## 2.4 Operating Environment:

The application is designed to operate on platforms that support Node.js and React.js ,both of them are JavaScript frameworks. The application server, which handles incoming requests, can be run on any platform that supports Node.js runtime and has a TCP/IP stack. This includes platforms like Windows, macOS, and various Linux distributions.

The web application can be deployed on servers or cloud platforms that support Node.js runtime and provide the necessary infrastructure for hosting React.js applications. This includes platforms like AWS, Google Cloud Platform, Microsoft Azure, Heroku, and others. This whole application is also containerized so as to make working available without any problem on any system interface.

The server hosting the application should be connected to the internet or an intranet network to allow access to users. Relevant firewall settings should be configured to allow incoming traffic to the server without dropping any requests.

## 2.5 Design and Implementation Constraints:

Developers are constrained to using Node.js for providing the backend services and React.js for building the frontend of the web application. This constraint is imposed due to the preference for modern JavaScript frameworks and libraries for web development.

Using Node.js and React.js allows for rapid development and prototyping of the application within the limited time frame available for the project. Additionally, these frameworks offer a wide range of libraries, tools, and community support, which can expedite the development process.

The Project also includes the use of various libraries of JavaScript so as to increase and enhance the speed of the development time. Some of the libraries used in the project are MongoDB Database for managing the Database, Firebase for the storing of the files uploaded by the teachers and students. Other Frontend Libraries include Tailwind CSS, React, Redux libraries.

The project timeline limits the exploration of alternative technologies or languages for backend development. Therefore, developers must adhere to the chosen stack of Node.js and React.js for consistency and efficiency in development.

Overall, the chosen constraints aim to leverage the strengths of Node.js and React.js to efficiently develop a feature-rich online learning platform similar to Microsoft Teams within the given time frame and resource constraints.

## 2.6 User Documentation

The web application will possess a self-documenting, easy-to-use interface that does not require a specific set of manuals for any of the users. All the relevant information regarding the usage of the web application will be placed at the appropriate locations inside the different webpages and displayed when needed.

## 2.7 Assumptions and Dependencies

1. Internet Access: Users require reliable internet connectivity to access OCMS, impacting accessibility and functionality.
2. Device Compatibility: OCMS must be compatible with various devices and browsers to ensure a seamless user experience.
3. Institutional Support: The educational institution is responsible for supporting OCMS deployment, maintenance, and technical assistance.
4. Data Security Measures: Implementation of robust data security protocols is essential to safeguard sensitive user information within OCMS.

## 3 External Interface Requirements

### 3.1 User Interfaces

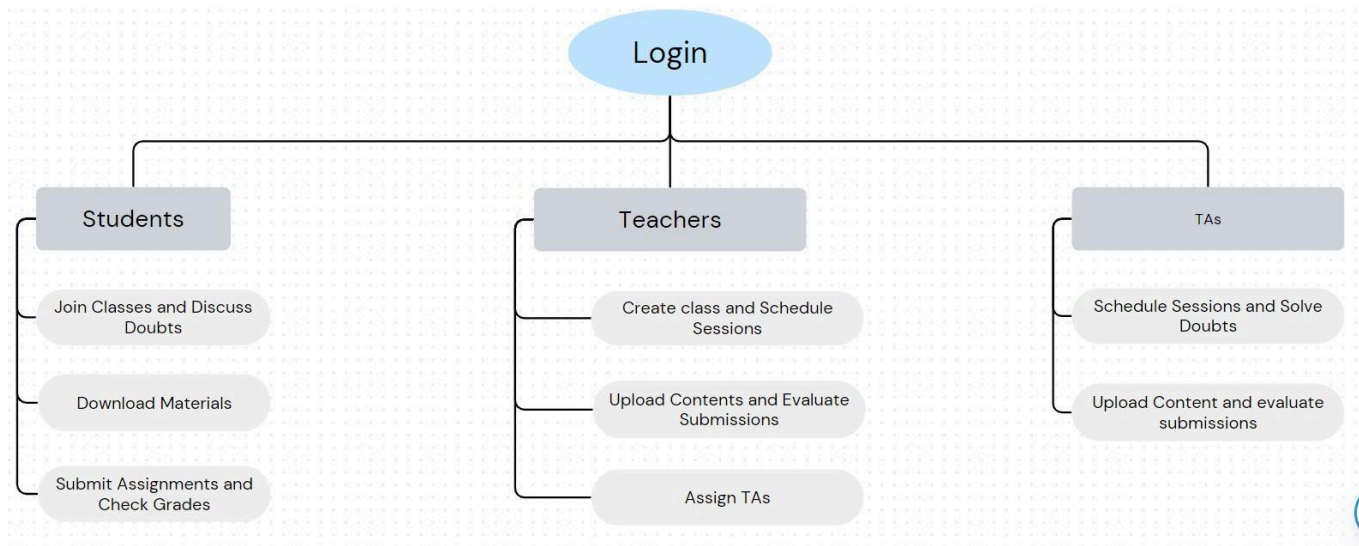


Figure 3.1: User interfaces

### 3.2 Hardware Interfaces

1. The hardware used for this needs to be a server computer with a processor fast enough to handle a large number of requests simultaneously. Preferably a multi-core system with distributed processes across the core will be suitable.
2. Solid state drives instead of hard disks to boost the speed of information retrieval. Enables faster request handling.
3. RAID systems to manage data redundancy. Ensures no user data is lost because of storage device failures by maintaining redundant copies as backup.

### 3.3 Software Interfaces

1. Web browsers needed on the end-users' systems to access the web application



2. Graphical user interfaces combined with the browsers needed. Command-line interface won't be able to meaningfully handle the user interfaces provided in the web application.

### 3.4 Communications Interfaces

1. Will be using a standard TCP/IP stack available on most operating systems.
2. Will use the HTTPS protocol to deliver webpages and handle user login

## 4 System Features

### 4.1 User Authentication and Security

**Secure Login Process:** Users are required to provide valid credentials (username/password) for authentication before accessing the OCMS. The login process is encrypted to prevent unauthorized access. Users visiting for the first time can sign up by entering credentials like name, e-mail, contact number and password in the sign up option. All the data is stored in a database and can be accessed by the application whenever needed.

### 4.2 Virtual Classroom and Messaging Interface

- **Class Creation:** Professors can create virtual classrooms for their courses within OCMS. They provide the link for live classes and specify the course name, description, field name and class level during creation.
- **Scheduled Classes:** Professors schedule classes within the virtual classroom according to the course timetable. These classes are displayed in the calendar view for easy access by students. This online class allows teachers to share their screen, video so as to ensure a seamless communication and avoiding any problems in conducting lectures online.
- **Messaging Interface:** Each virtual classroom is equipped with a messaging interface where students can communicate with the professor and their peers. This facilitates asking questions, discussing topics, and sharing relevant information during classes.
- **Live Classes Option:** The Live Classes feature allows professors to conduct real-time interactive sessions with students on the link provided while class creation.

### 4.3 File-sharing System for Professor

- **Professor Privileges:** Only professors have the authorization to upload files to the virtual classroom. This ensures that educational content is curated and controlled by the course instructor, maintaining consistency and quality.
- **File Types:** Professors can upload a variety of file types, including video lectures, PDFs for assignments, and documents containing class tests. This allows for multimedia-rich content delivery and diverse assessment methods.
- **Accessibility:** Once uploaded, the files are accessible to all students enrolled in the respective virtual classroom. Students can view and download the materials as per their learning needs and preferences.

## 4.4 Personalised Calendar for Students and Professors

- The professors have the options to set the timetable for their classes the time and day they put for the classes is shown on the calendar
- The students who join the class can also view the calendar, this will help the students to know when class is going to happen.

## 4.5 Assignment Uploading and Evaluation

- The teachers and TAs can upload assignments in the form of a pdf through the create assignment option and the reminder will be provided in the reminder section on the right-hand side of the interface.
- While uploading the assignment, the deadline needs to be specified and the status of submission (late submission, submission on time) will be given accordingly.
- The teacher or TA can then specify the grade obtained by a particular student in an assignment which will be displayed to the student in their account.
- The Grades tab serves as a dedicated section for students to access their academic performance in each subject. This feature allows students to view their grades, including marks obtained in assignments, tests, and other evaluations conducted by professors or teaching assistants (TAs)

## 4.6 Teachers' Functions:

- Create Classes: Teachers have the authority to establish virtual classrooms for the subjects or courses they are assigned to teach. They can specify the course name, description, field name and class level during creation. Each class serves as a centralized hub for students to access course materials, interact with peers, and engage in academic activities.
- Schedule Timetable: Teachers meticulously plan and organize the class timetable, outlining the dates, times, and durations of each session. They coordinate with students to establish a consistent schedule that accommodates diverse learning needs and preferences. Timely scheduling enables students to allocate time for attending lectures, completing assignments, and preparing for assessments effectively.

- **Upload Materials:** Teachers curate and upload a variety of educational resources to the class repository, including lecture videos, presentation slides, reading materials, assignments, and assessments. They ensure that the materials are relevant, up-to-date, and aligned with the course curriculum. By centralizing resources in one accessible location, teachers facilitate seamless access to course content for students.
- **Evaluate Submissions:** Teachers assess and evaluate student submissions, including assignments, tests, projects, and other assessments. Through timely feedback, teachers guide students in understanding their strengths and areas for improvement, fostering continuous learning and growth.
- **Interact with Students:** Teachers actively engage with students through various communication channels within the virtual classroom, such as discussion forums, chat rooms, and messaging interfaces. They encourage open dialogue, address student queries, clarify doubts, and facilitate meaningful discussions related to course topics. By fostering a supportive and collaborative learning environment, teachers enhance student engagement and participation.
- **Monitor Attendance:** Teachers monitor student attendance during online classes to track student participation and engagement. They maintain attendance records and use attendance data to identify patterns, trends, and potential areas for intervention. By monitoring attendance, teachers ensure that students actively participate in class activities and meet the required learning outcomes.
- **Manage Grades:** Teachers maintain comprehensive grade records for each student, documenting their performance in assignments, tests, quizzes, and other assessments. They accurately record grades, calculate cumulative scores, and update grade sheets regularly to reflect students' progress and achievements. Transparent grading practices enable students to track their academic performance and make informed decisions about their learning strategies.
- **Assign Teaching Assistants (TAs):** Teachers may delegate certain responsibilities to teaching assistants (TAs) to support course management tasks. TAs assist in uploading course materials, moderating discussions, grading assignments, and providing additional academic support to students. By leveraging TA support, teachers streamline administrative tasks and enhance the overall learning experience for students.
- **Provide Announcements:** Teachers disseminate important announcements, updates, and reminders to students through the class communication channels. They notify students about changes in class schedules, assignment deadlines, upcoming assessments, and other relevant information. Clear and timely communication ensures that students stay informed and engaged in their learning journey within the OCMS platform.

## 4.7 Student's Functions:

### 1. Download Course Materials:

Students can access a repository of course materials such as lecture notes, presentation slides, reading materials, and supplementary resources

- They have the flexibility to download these materials to their devices for offline viewing or reference, allowing them to study at their own pace and convenience.
- Access to comprehensive course materials enhances students' understanding of course concepts and provides additional resources for self-study.

### 2. View Course Schedule:

- The platform provides students with a clear and organized view of their course schedule, detailing the dates and times of lectures.
- Students can easily plan their study routine and allocate time for attending classes, completing assignments, and preparing for assessments.
- A centralized course schedule minimizes confusion and ensures that students stay informed about upcoming events and deadlines, promoting better time management and academic success.

### 3. Participate in Virtual Classrooms:

- Virtual classrooms serve as online spaces where students can engage in discussions, collaborate on projects, and interact with their peers and instructors.
- Through features such as chat rooms, discussion forums, and video conferencing tools, students can actively participate in class activities and contribute to academic discourse.
- Virtual classrooms foster a sense of community among students, despite physical distance, and provide opportunities for collaborative learning and knowledge sharing.

### 4. Submit Assignments:

- The platform facilitates the electronic submission of assignments, eliminating the need for physical submissions or email correspondence.
- Students can upload their completed assignments directly to the platform before the specified deadline, ensuring timely submission and adherence to course requirements.
- Submission portals may include options for attaching files, adding comments, and verifying submission status, streamlining the assignment submission process for students and instructors alike.

5. *View Grades and Feedback:*

- Students have access to a comprehensive gradebook that displays their performance on assignments, tests, quizzes, and other assessments throughout the semester.
- Detailed feedback from instructors accompanies graded assignments, providing insights into strengths, weaknesses, and areas for growth.
- By reviewing their grades and feedback, students can track their progress, gauge their understanding of course material, and make informed decisions about their academic development.

6. *Access Personalized Calendar:*

- Each student receives a personalized calendar within the platform, populated with their course schedule, assignment deadlines, test dates, and other relevant events.
- Calendar entries are dynamically updated based on changes made by instructors or administrators, ensuring accuracy and timeliness.

7. *Communicate with Professors and TAs:*

- Built-in communication tools facilitate direct interaction between students and instructors, fostering a supportive learning environment.
- Students can initiate discussions, ask questions, seek clarification, and request assistance from professors and teaching assistants (TAs) through messaging features or virtual office hours.
- Timely communication ensures that students receive the guidance and support they need to succeed academically, address challenges effectively, and engage meaningfully with course content.

## 5 Other Nonfunctional Requirements

### 5.1 Performance Requirements

#### 5.1.1 Server Up-time

The system needs to keep running for long periods of time for accessibility so the hard-ware needs to be capable of that. Moreover the maintenance frequency needs to be low to have a stable experience for the end users.

#### 5.1.2 Response Time

The system needs to have a good response time as it will be repetitively used by various stakeholders in the Institute and the response time needs to keep up with the incoming requests.

#### 5.1.3 System Dependability

It determines the failure rate of the system, that is, the cases when the system receives an incorrect input or there is a random failure. The end user needs to be notified in such a scenario.

### 5.2 Security Requirements

#### 5.2.1 Secure Login

The system should be protected against forced and malicious login attempts to access the software. The web interface needs to be continuously patched to fix bugs as and when they crop up.

#### 5.2.2 Verifiable Access

The system is designed to serve multiple parties and their levels of access and their permissions are decided by their login IDs. Thus, the login and access to functionality should be robust and extremely secure since the system also deals with the finances of the institute.

#### 5.2.3 Student Information Must be Secure

All the personal student information stored by the system should be safe and secure.

#### 5.2.4 Secure File Sharing:

Implement secure file-sharing mechanisms that allow only authorized users, such as teachers and students enrolled in specific classes, to upload, download, and access educational materials. Employ access controls and file encryption to prevent unauthorized access, data breaches.

#### 5.2.5 Secure Communication Channels:

Ensure the security of communication channels within the OCMS platform, including messaging interfaces, discussion forums, and live chat features. Use encryption protocols and secure messaging standards to protect the confidentiality and integrity of communications exchanged between users.

### 5.3 Software Quality Attributes

#### 5.3.1 Reliability

The system must be dependable, which means it must constantly perform as expected with very low downtime and rate of failure. In absence of this the educational institute will not be able to function smoothly.

#### 5.3.2 Information Accessibility

To save the login credentials and the myriad types of data that the software uses, the application should be able to save, access and alter files on the system so that the information is retained even on successive runs.

#### 5.3.3 Maintainability

The program should be easy to modify and add features to. The code should be clearly commented and the features well documented to allow for easy maintenance.



## 6 Other Requirements

### 6.1 Appendix A: Glossary

Given that we have used fairly intuitive terms throughout the document, it is meant to be self-explanatory. The user classes are defined in terms of the end users themselves and are thus clear on their real-world functionality. Any implementation level detail or term can be looked for in the documentation of the languages used to write this software.

### 6.2 Appendix B: Analysis Models

- Implementation of the TA feature will be done as we proceed .