Phase-2 Development Report

For

An Interactive e-learning Platform for Learners

Version 1.0 approved

Prepared by:

Technotuners

Nikitha Roa Jakati

Sivani Akkem

Haritha Talupula

Umarani Vemula

Jyothirmayee Manne

Sai Rukma Reddy Gade

Lakshmi Dheeraj Oruganti

Sai Krupanand Reddy Bakaram

Date Created: 11/07/2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Nikitha Rao J	07/11/2022	Initial Copy	1

1.Requirements:

Phase 1 Development

The implementation plan for the Learning platform of Phase 1 consists of everything from developing user interfaces homepage, registration development and the setup of the database.

1.1 Homepage of the system:

The Home page includes login and registration sections for both students as well as tutors and it is the same for both whereas, the registration of students differs from tutors. For registration, 2 options are displayed. One is for students and the other is for tutors.

1.2 Account registration for both Tutors and Student:

In order to gain access to this learning platform the tutor or the student should first register by following the steps. Tutors will have to provide their First Name, Last Name, Email, Username and Password for registration whereas, students must provide First Name, Last Name, Roll Number, Student ID, Username, Email Address, and Password. Once registered, they can login to their respective accounts using their credentials.

1.3 Login for both Tutors and Student:

1. Valid username and password should be entered to login to their respective accounts as usual.
2. If a student logs in, the homepage consists of a list of available courses from which the student can enroll to the courses of their choice.
3. If it is a teacher, the homepage must display their profile and dashboard for the course he is tutoring.

1.4 Forgot Password:

We currently include a "forgotten password" feature in this deliverable, but we are unable to complete this submission due to time constraints. We will add this feature in the next deliverable. As this will not impact most of the other functionalities.

1.5 Profile Information :

In our project, we've developed a feature where users can add their profile information such as First name, Last name, Username, Email ID. So, we're planning to incorporate these changes dynamically in the next development phase i.e, phase 2.

1.6 Setting the Project:

First, we need to install a python 3.7 or higher version. Then install a package called pip inside it. Set the path and environment variables and execute the following commands.
`pip -r requirements.txt && python manage.py makemigrations $$ python manage.py migrate && python manage.py runserver`

Phase 2 Development

1.7 User Interface for profile pages:

1. All users can use their student ID /tutor ID and password to login to the system and access the application.
2. The student can login and enroll in courses after completing the registration process.
3. The instructor, on the other hand, can log in and view the courses he or she is teaching.
4. When the instructor/student wants to log in, he or she is prompted for a student/instructor ID, password, and role. Tutors can add assignments, can upload classes files which will be available for download. Tutors can view students' details who are enrolled in the class.

1.8 Phase 2 design and functionalities

In this phase, we will be implementing the student functionalities and instructor functionalities which include the classrooms, assignments , grading students and feed back feature and also this time we include an advanced security system feature that includes a forget password feature which can secure our system in a more advanced way.

1.9 Database data linking:

We've created mongoDB, which was used to create a database and then we've linked to it . Then, we established a connection between the frontend and backend for both the students and teachers which used to access the core functionalities such as classroom, modules, etc.

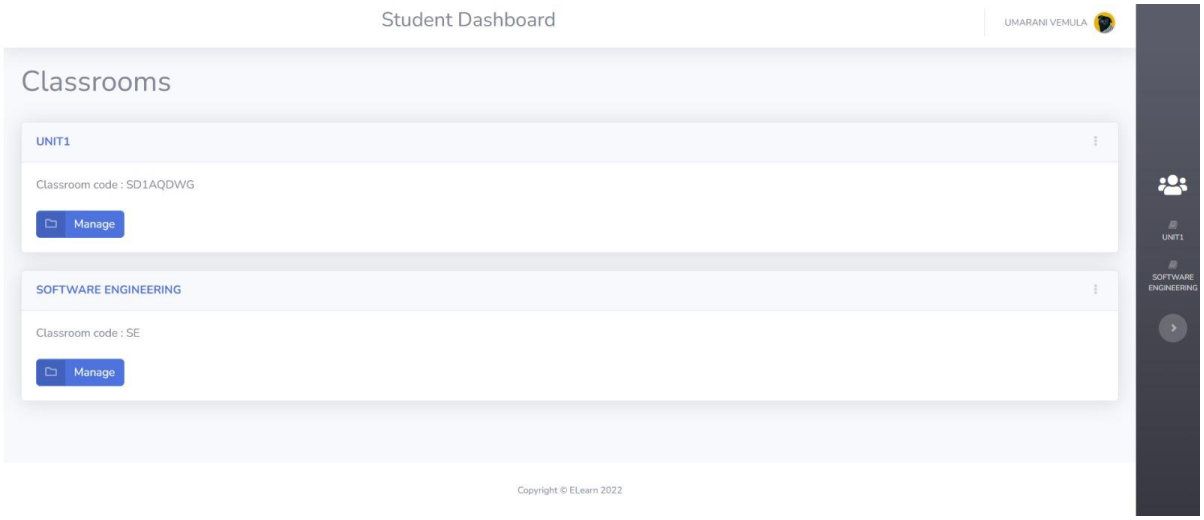
1.10 Challenges for Phase 2 implementation:

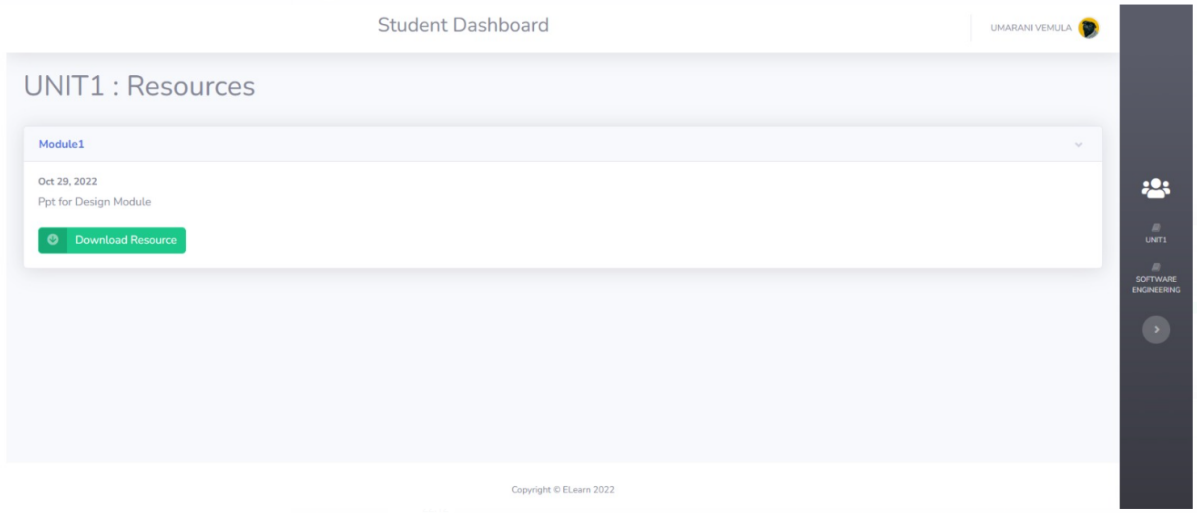
We've actually created and designed the functionalities for the student and implemented them but as the teachers module is interlinked with teachers, it was a challenge for us which made us very difficult for the creation of students classes. So, we've created dummy files in the database and used it dynamically for the student modules.

Project Scope:

Introduction :

In this phase of the project, the implementation of the Student dashboard is done. From this dashboard the student user can join into the available courses and the classes available. By this dashboard all the class functions can be managed by the students. He/She can check their activities in the class in their dashboard.The list of activities are like resources, progress and assignments.The resources posted by the instructor can be downloaded or viewed from the student dashboard. Assignments posted by the instructor get displayed in the student dashboard and the student gets hands-on experience of the course by completing the assignment successfully. The list of classes enrolled are shown in the dashboard immediately after the user gets logged in. Then after each class activity can be viewed by entering into the particular class. By this, the student comes to know the number of classes he/she enrolled and can complete them accordingly.





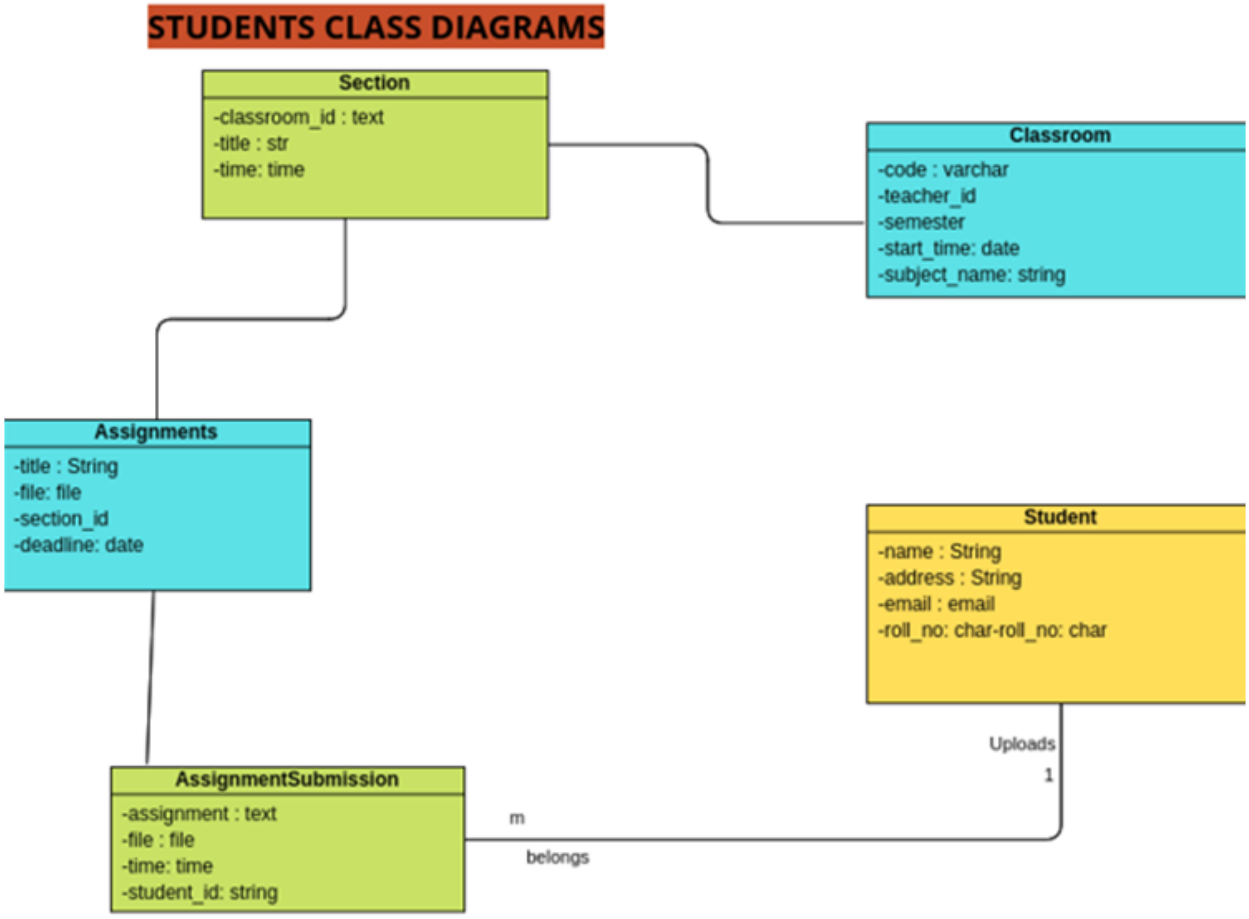
Scope of our Project:

So, ultimately we've created the classes for both teacher and student, but as we've changed our plan according to the one which was agreed upon. We're designed the functionalities for student in a more dynamic way rather than the teacher module. So, in order to check the functionalities of the student, we've created dummy files in the database for the teacher. In teacher we've implemented the dashboard functionality, which can be linked to the student where we've done with all other individual aspects of the students functionalities such as the dashboard, resources, modules, etc.

2 .UML Diagrams:

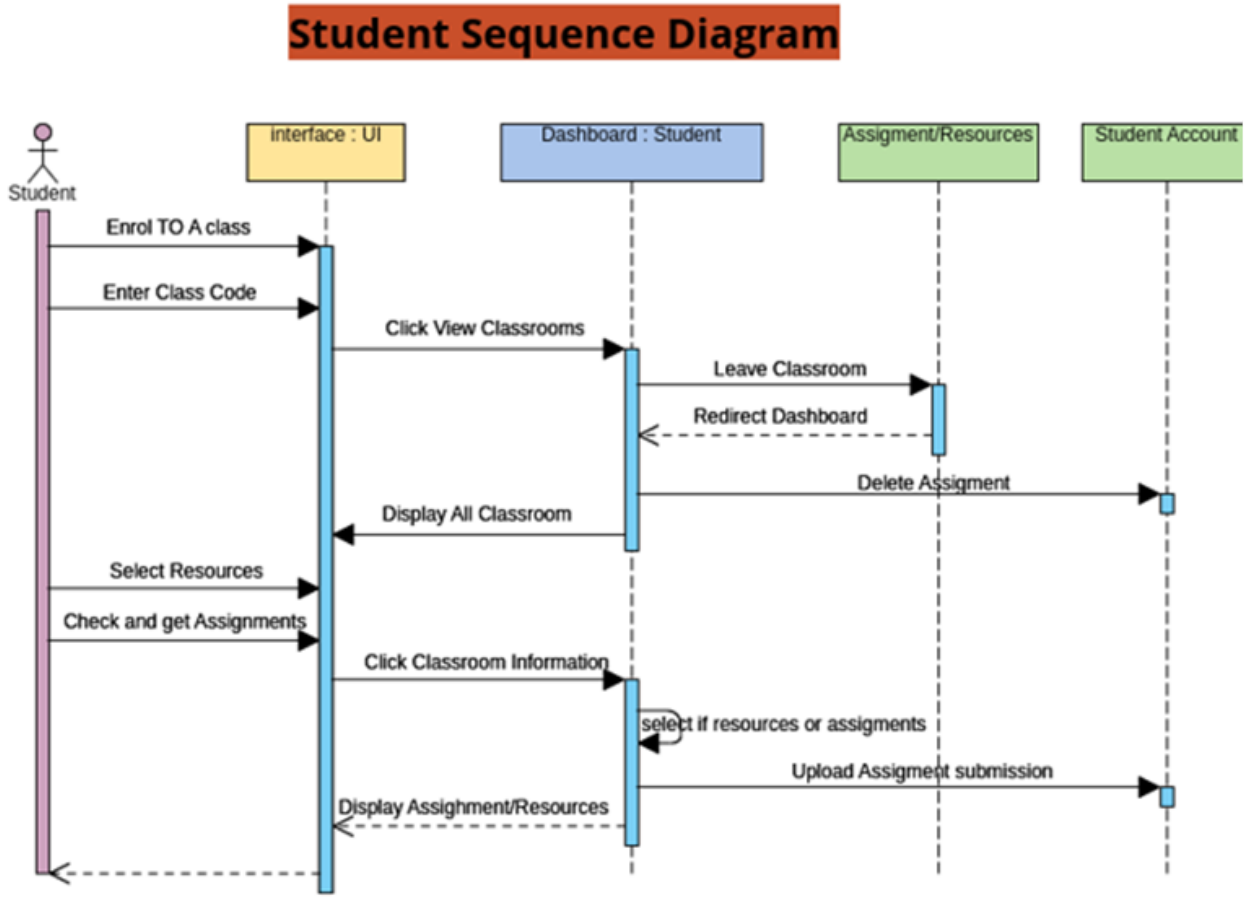
2.1 ER/Class Diagram

There is consensus among industry experts that the ER model is the best example of a conceptual data model at the enterprise level. The concept that underpins this paradigm, as well as its many different iterations, is utilized by a broad range of database design tools in order to simplify the process of developing cutting-edge database applications. The ER (Entity Relationship) Diagram that has been presented here illustrates the idea of an E-Learning Management System Entity. Entity-relationship diagrams can be used to display the interconnections between the various parts of an E-Learning management system, such as students, fees, courses, and training. One way to visualize the database tables that make up an E-Learning management system is in the form of an entity-relationship diagram.



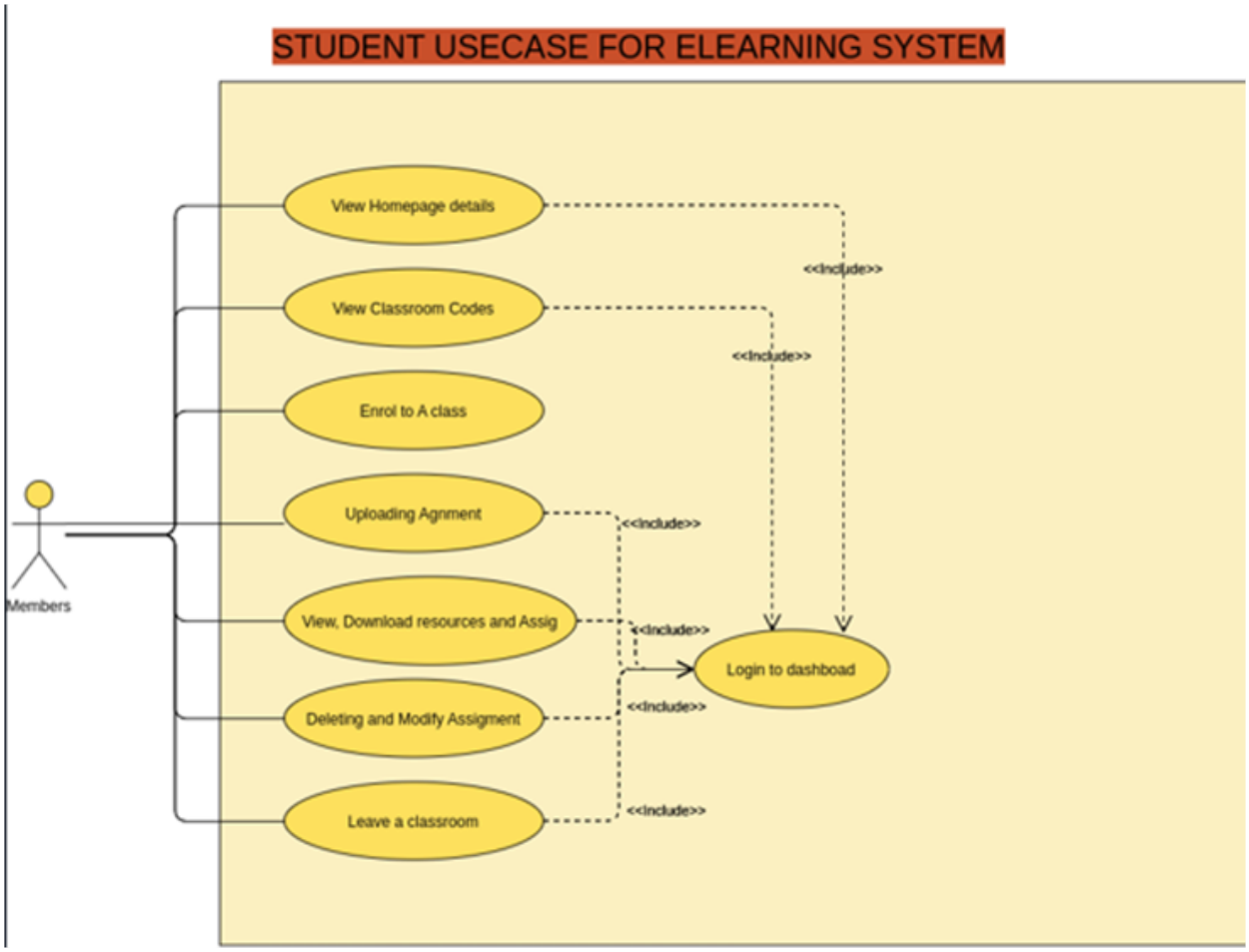
2.2 Sequence Diagram

This is an example of a UML sequence diagram for an e-learning management system. It demonstrates how the objects of Student interact with the system. The diagram may be seen by both students and instructors. The following is a rundown of the chronological order on how the student action from enrolling to a class, selecting resources and checking assignments:

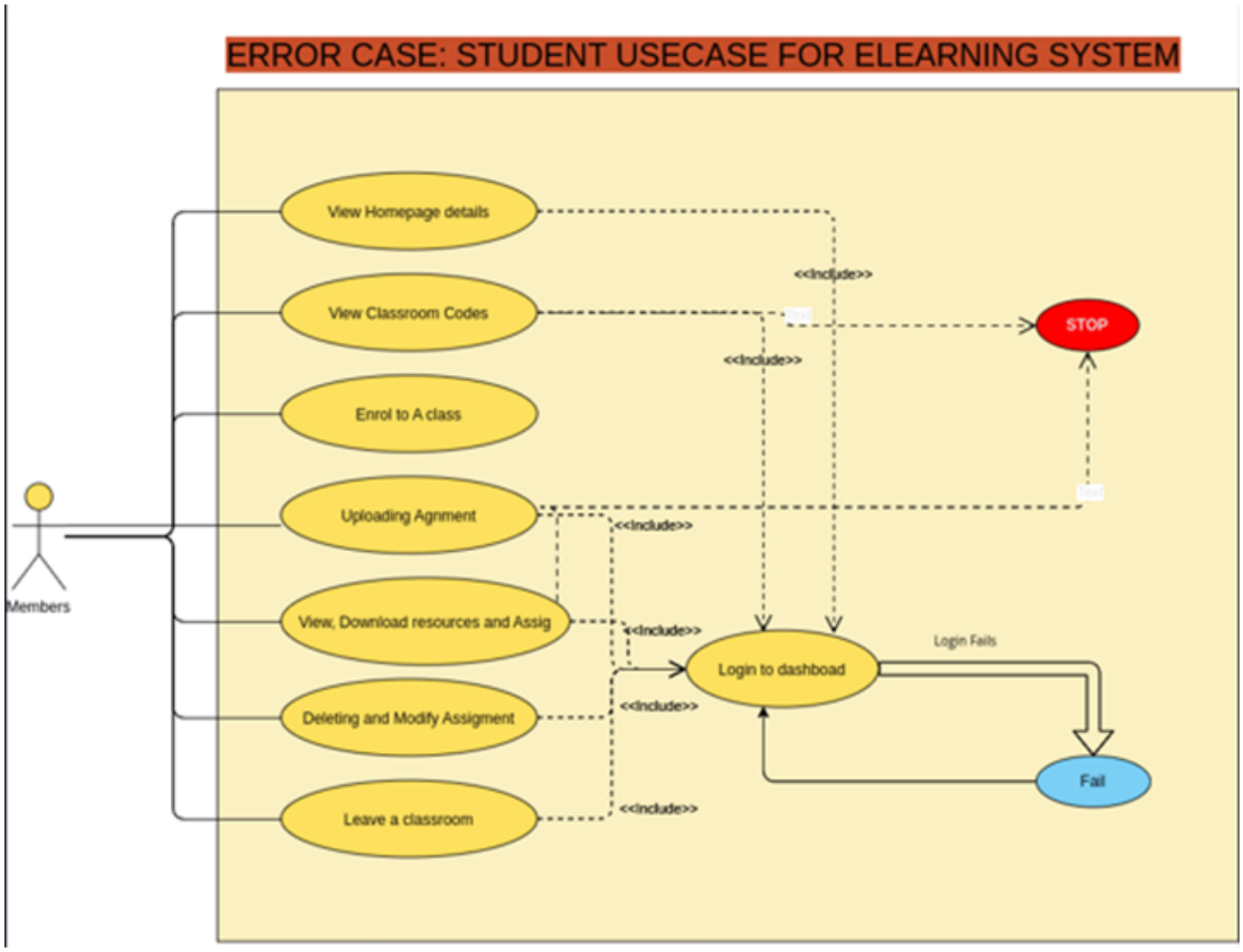


2.3 Use case model of the project

An e-learning management system's use case diagrams are graphical representations of the relationships between the many modules that make up the system. Using a process that is quite similar to the one represented above, the system demands of an e-learning content management system may be uncovered, clarified, and structured. This Use Case Diagram illustrates how the student interacts inside an e-learning management system. Students are executing the many different use cases that make up an e-learning platform. Some of these use cases include: viewing homepage details, viewing classroom codes, enrolling to a class, uploading assignment, viewing, downloading resources and assignment, deleting and modifying assignment, managing the subject, leaving the classroom and managing the full e-learning management system operations.


STUDENT USECASE FOR ELEARNING SYSTEM
View Homepage details
View Classroom Codes
Enrol to A class
Uploading Agnment
View, Download resources and Assig
Deleting and Modify Assigment
Leave a classroom
Login to dashboad
Members
<<Include>>
<<Include>>
<<Include>>
<<Include>>
<<Include>>
<<Include>>

2.4 Student UML Error case



3. Test cases

During the testing phase of an application or piece of software, a "test case" is a specified set of procedures that may be carried out in order to check that the application or program is operating correctly. A test case is nothing more than a set of criteria that need to be evaluated in order to establish whether or not the application or piece of software functions as intended. One of the numerous components of a case is its identification number (ID), followed by its condition, steps, input, expected result, actual result, status, and comments.

Test cases include:

S.No	Test Case	Pre-Condition	Post-Condition	Result
1	Student Dashboard	Users should be able to view the Student dashboard on logging in.	After logging in, he has all read/write access of the dashboard	Pass
2	On Click to the Dashboard	None	Should be able to view all the assignments, courses, and the classrooms	Pass
3	User Profile of Student	Student will be able to see the options of his profile.	On clicking the profile, he should be able to see the dashboard, Class room, and Logout options.	Pass
4	Navigating to Dashboard	The Student should be able to see the dashboard of the class on clicking it.	On clicking, Student was navigated to the dashboard where he could see the classes	Pass
5	Navigation to unit resources	The student should be able to view the units of the class and	On Clicking, the modules, he could be able to see the resources and was	Pass

		download the resources	able to download them	
6	Download Resources	None	On clicking , he must be able to download them	Implementing them as per the teacher module
	Assignments page	None	On clicking them, he could see all the assignments under him.	Implementing them as per the teacher module

4. User Manual

4.1 Installation and Usage:

To run this application, do the following:
- Make sure you have installed python3.8 or higher
- Install pip3
- After installing the above, run the following command.

`pip -r requirements.txt && python manage.py makemigrations $$ python manage.py migrate && python manage.py runserver`
- The above is a 4 commands line which you can split into individuals if you want so.

4.2 Product Features

Features of the project for Phase 1 include:
• Student and Tutor Signup using student/Tutor username, password,
• Student and Tutor Login.

4.3 Screens

The main application screens created as part of Phase 2 are listed below.

4.3.1 Home Page:

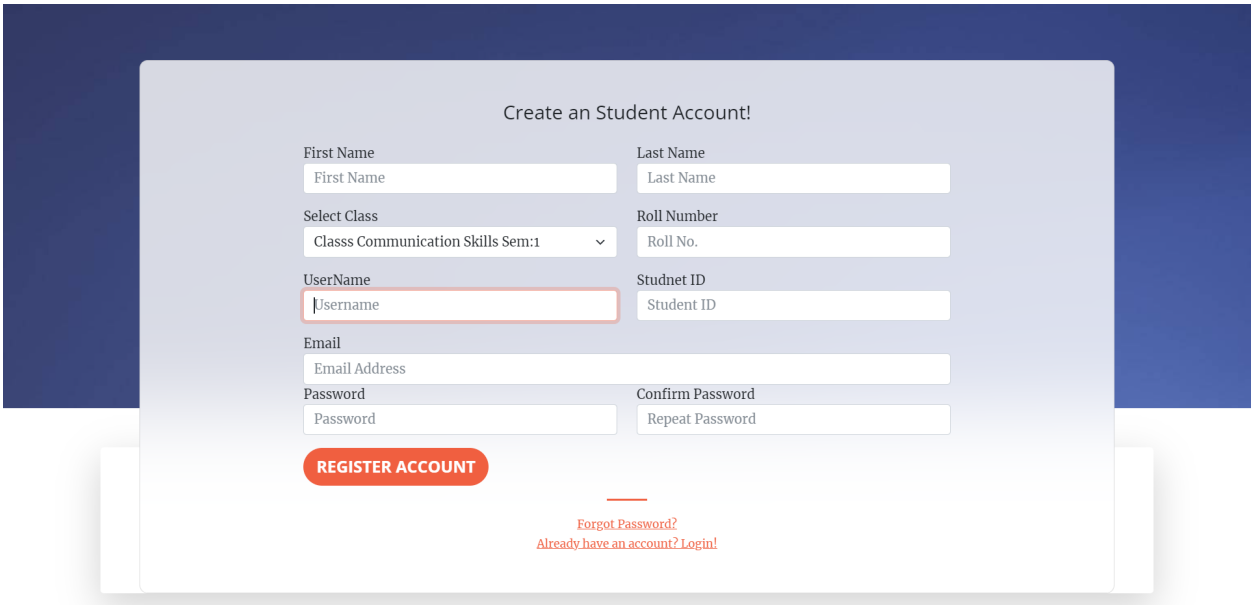
This is the landing page of this application. From this page user can navigate to the login or register page according to their choice.

4.3.2 Register Page:

Registration page for both students and tutors are different.

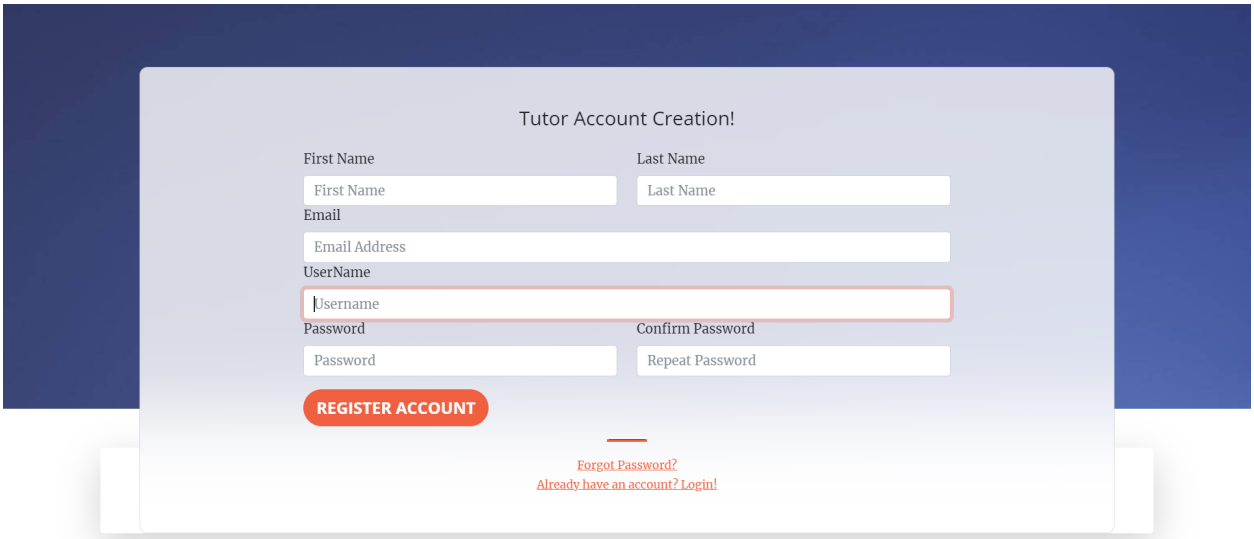
Student Registration page:

Students need to provide First name, Last name, Class, Email, Roll no, Student Id, username, password for registering.



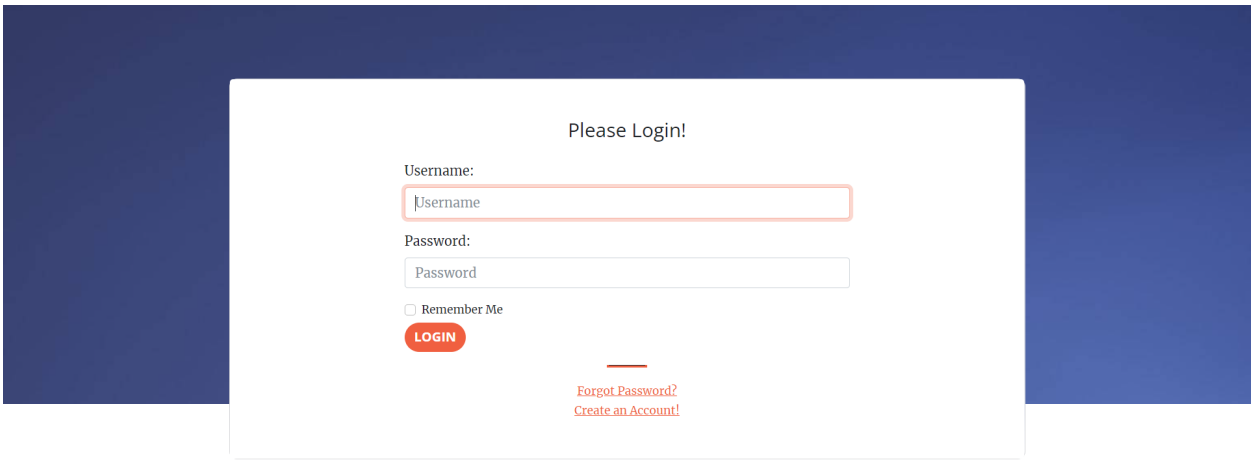
Registration of Tutor:

Tutor need to provide First name, last Name, Email, Username, Password for registering.



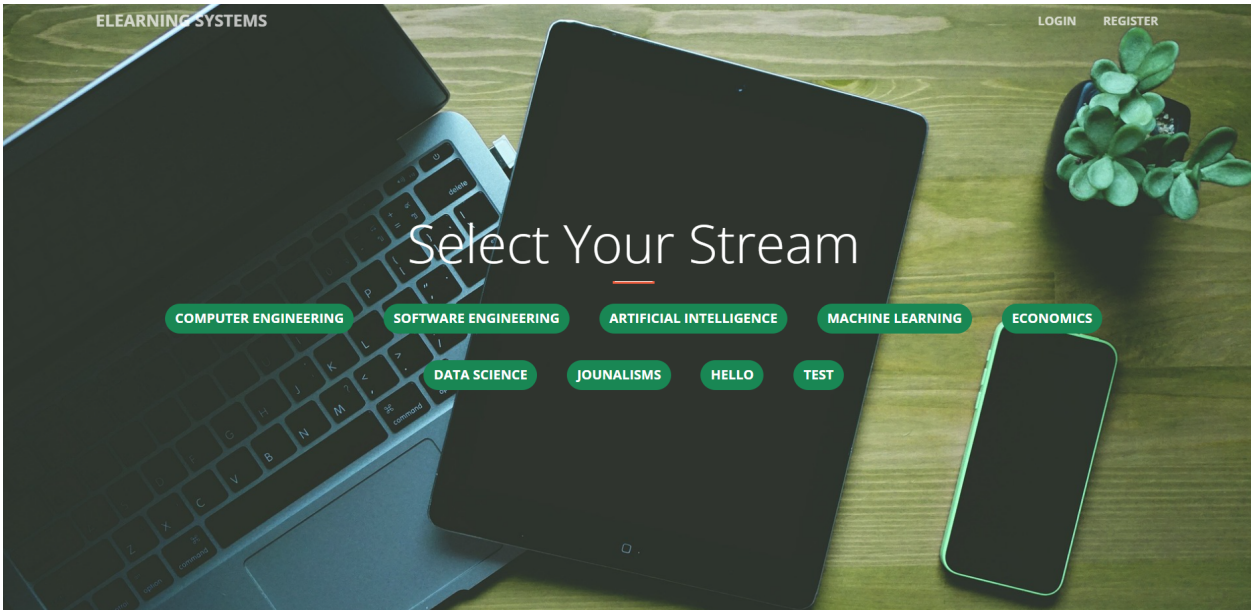
4.3.3 Login Page:

Here the login page for the both students and tutors is same.



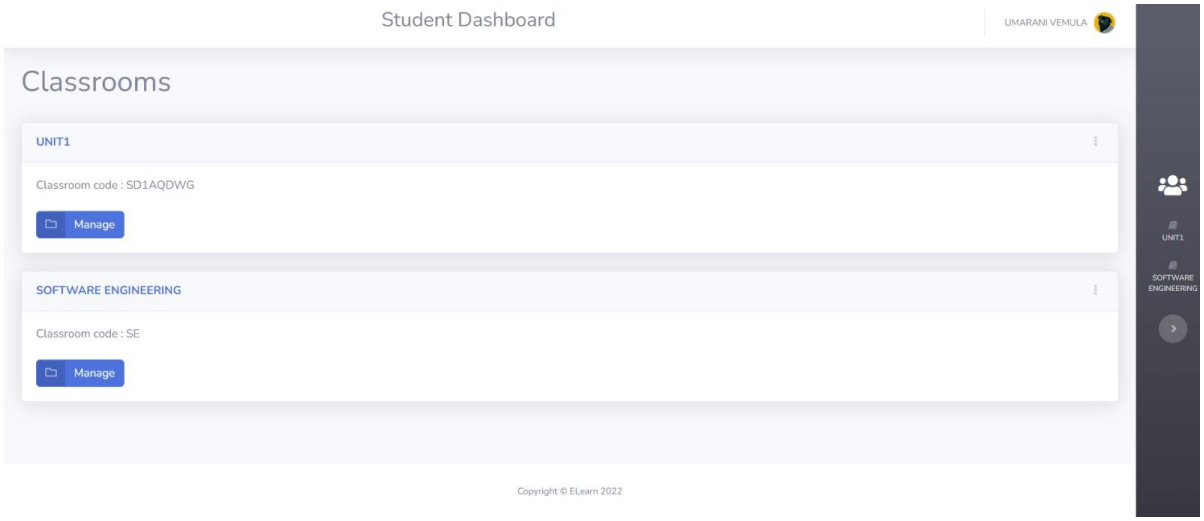
4.3.4 Course Selection before registration

Here the students and tutors need to select the course for registering the account. Means when we click on register page it will navigate to the below screens, after completion of selecting course it will navigate to the registration page for Student/Tutor.



4.3.5 Student Dashboard

Here Student registered courses will display



5 Peer Review feedback

Meeting on Tuesday, 11/01 12:00 PM-1:20PM
Purpose: Peer review with Group12SE

Discussed about the project and explained the features and also done the code inspection.

Suggestions given by the partner group

- Security:

The partner group suggested us about the security of the project, they recommended us to provide forgot password feature so that it will be helpful when the student or professor forget their password it will be easy for them to recover it. They asked us to link email or phone number to the account, so that a otp or link will be sent and it will be easy to the users.

Validating the features of the system:
The partner group were able to test the features of the system and also they said the system is fulfilling all the requirements given by the clients. Due the fact the system is quite user friendly, the vast majority were able to navigate without any problems.

Accepted:
The suggestion about the security of the system given by the partner group is accepted and will be implemented.

Rejected:
The only suggestion given is the security by adding forgot password feature, so there is nothing to reject.

Discussion with Group 12SE:

For the assessment of the tests, will the correct answers provided at the end of the test?
Answer: Yes, at the end of each module a assessment test will be given to the student to test them. And after the test the solutions are provided.

6 A brief reflection

In conclusion, I've realized that developing a completely working system is not only challenging labor, but it also requires an in-depth knowledge of database architecture and several programming languages. This was one of the most valuable lessons I've ever acquired. While carrying out this research, we discovered something that led us to this conclusion. Unfortunately, I do not have the necessary expertise to construct the system that I have in my head, but I wish I did.
If the university takes this approach, it will be able to become a more innovative institution and avoid a stalemate that is produced by the widespread dissemination of erroneous information among the teaching staff and the student body.

7 References

Ülker, D., & Yılmaz, Y. (2016). Learning Management Systems and Comparison of Open Source Learning Management Systems and Proprietary Learning Management Systems. Journal Of Systems Integration, 18-24. https://doi.org/10.20470/jsi.v7i2.255
WARSCHAUER, M. (1997). Computer-Mediated Collaborative Learning: Theory and Practice. The Modern Language Journal, 81(4), 470-481. https://doi.org/10.1111/j.1540-4781.1997.tb05514.x
Syed, T. A., Palade, V., Iqbal, R., & Nair, S. S. (2017). A personalized learning recommendation system architecture for Learning Management System. Proceedings of the 9th International Joint

Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management. https://doi.org/10.5220/0006513202750282

8 Team Member Contribution Table

Name	Contributions	Overall Contribution(%)	Notes
Nikitha Rao Jakati	Backend Developer/ Testing	12.5	Backend code ,HTML (UML diagrams)
Umarani Vemula	Backend Developer	12.5	Backend code, Testing, documentation
Sivani Akkem	Database Administrator	12.5	Backend code, Frontend coding, UML diagrams
Haritha Talupula	UI Developer/ Testing	12.5	Backend code ,CSS, Bootstrap,Frontend coding (UML diagrams)
Jyothirmayee Manne	UI Developer	12.5	Backend code ,JavaScript documentation
Sai Rukma Reddy Gade	Backend Developer/ Testing	12.5	Backend code ,HTML frontend coding, documentation
Lakshmi Dheeraj Oruganti	Database Administrator	12.5	Backend code , frontend code documentation
Sai Krupanand Reddy Bakaram	UI Developer/ Testing	12.5	Backend code, Python, JavaScript, Test cases