Final Documents FYP

by Ahmad Cs

Submission date: 12-Dec-2022 04:45PM (UTC+0500)
Submission ID: 1979032505
File name: final_doc_fyp_10-12-22_1.docx (4.67M)
Word count: 9499
Character count: 50192

BSCS F22-009

03-134191-005 Areef Imran

03-134191-033 Muhammad Ahmad Mumtaz

Learners Arena

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Nadeem Sarwar

Department of Computer Sciences

Bahria University, Lahore Campus

December 2022

Certificate

We accept the work contained in the report titled

"Learners Arena"

Written by

Areef Imran

Muhammad Ahmad Mumtaz

As a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Nadeem Sarwar

_____ (Signature)

December 12, 2022

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134191-005	Areef Imran	
03-134191-033	Muhammad Ahmad Mumtaz	

Date : December 12, 2022

Specially dedicated to

My beloved Grandparents, Father, and Mother

(Areef Imran)

My beloved Grandparents, Father, and Mother

(Muhammad Ahmad Mumtaz)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Mr Nadeem Sarwar for his invaluable advice, guidance, and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Areef Imran

Muhammad Ahmad Mumtaz

Learners Arena

ABSTRACT

As a result of the COVID-19 issue, teaching has become extremely tough for Instructors. It's difficult for students and instructors to leave their homes in that pandemic situation and students face a lot of problems regarding their studies. As a result, we decided to create an automated system that would allow teachers and students to engage more efficiently. There is no time restriction left to learn

You can learn whenever you want to learn. Teachers will be able to offer courses to students and add their lecture notes and videos using our system. Students, on the other hand, can easily learn the course content with just the internet connection. There is no need to go anywhere for learning. The application will be consisting numerous features such as course creation and selection, online payment of course, certificate after completing the course, discounted coupons for paid courses, student support service to assist students, take their feedback, comments and complains, grading of assignments by instructor, quiz uploading and automatic grading, text editor for assignments and code editor for programming assignments, notification on email of registration and new course addition, report generation regarding grades by student and instructor as well of enrolled Students, performance and abilities of student, profile management and grade analytics for the student

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

We are developing a platform that will be adopted by learning institutes to improve their revenue in a circumstance like Covid-19. Our system's primary responsibilities are instructor and student, and we provide user registration for both. The instructor can design a course, and the student can easily register during his or her choice. Instructors can provide lecture information in the form of videos and notes to our system, which is easily accessible via the student site. Instructors post video content in the form of a lecture that students can download. Before course publishing, for the purpose of assessing student abilities, the teacher may also have the right to post quiz questions, which will be shown to the student. The student will be provided his grades and percentage of marks, and he or she be able to compile a report of attempted Quizzes and Assignments. When the student registers itself or a new course added by the teacher then the student will be notified by email. Some courses in our system are paid, and we also provide an online payment portal for enrolling in those courses. Students receive a validated certificate upon completion of the course. We also give hands-on practice for our courses on our platform. When a student completes a course, he or she generates a performance report. Furthermore, we offer a student support service to ask his/her queries, feedback or complain. In our system, profile management is customizable, and we also provide various types of offers in the form of discounts.

1.2 Problem Statements

The project is structured in such a manner that anyone from all over the world may access it and learn whatever they want assuming it is available on the site without being restricted to a four-wall room. The information would be created in such a way that every one of any gender, race, culture, nationality, religion, or age might learn something from it. The only thing required is an internet-connected Smartphone or Personal Computer of any specification. A university student will receive a course, will learn something new that is not taught at his or her institution, and will receive a completion certificate at the conclusion of the course. A professional will receive a course that will improve his or her abilities, resulting in a promotion. A person who is just curious about something will undoubtedly increase his or her knowledge level. An effective online learning platform that allows administrators to register new students, appoint teachers, and learn at any time. This method enables students to get online lectures and connect easily with their lecturers. It also saves student progress, payment history, and allows teachers to follow their students' development

1.3 Aims and Objectives

The objectives of the thesis are shown as following:

i) To make a web platform where students can easily gain knowledge anytime.

ii) To eradicate the restrictions of place, age, time, experience, prerequisite

1.4 Scope of Project

The major purpose of our system is for those who have difficulty learning and want to learn from home to utilise our system to learn and interact with various types of courses. There are numerous issues to learn about in college these days, when covid -19 situations are widespread. Our system provides a variety of courses, as well as tools based on the course. Students may readily consult with the help of our system's Student Support System, or you can talk to its teacher at any moment. The instructor provides course information in the form of recorded lectures, and students can view and download videos. The instructor can do online evaluations by taking the online quizzes and assignments. Students may examine their quiz results at that time, and after completing the course, a certified completion certificate is delivered, as well as a course assessment report. The reason behind making this kind of platform is to assist people getting jobs by learning new skills.

.

CHAPTER 2

SOFTWARE REQUIREMENTS SPECIFICATION

2.1 Product Perspective

The project is structured in such a manner that anyone from all over the world may access it and learn whatever they want assuming it is available on the site without being restricted to a four-wall room. The information would be created in such a way that every one of any gender, race, culture, nationality, religion, or age might learn something from it. The only thing required is an internet-connected Smartphone or Personal Computer of any specification. A university student will receive a course, will learn something new that is not taught at his or her institution, and will receive a completion certificate at the conclusion of the course. A professional will receive a course that will improve his or her abilities, resulting in a promotion. A person who is just curious about something will undoubtedly increase his or her knowledge level. An effective online learning platform that allows administrators to register new students, appoint teachers, and learn at any time. This method enables students to get online lectures and connect easily with their lecturers. It also saves student progress, invoicing history, and allows teachers to follow their students' development.

2.2 User Classes and Characteristics

Learners Arena consist of four major panels

i. Admin Panel
ii. Teacher Panel
iii. Student Panel

2.2.1 Admin Panel

Admin panel is main panel that can control everything. It can

i. Manage Users (Teachers, students)
ii. Manage payment method
iii. Manage Student Responses
iv. Manage Discounted Coupons
v. Manage Courses

2.2.2 Teacher Panel

Following are the major activities in teacher panel:

i. Create Course
ii. Upload Learning material
iii. View enrolled Students
iv. View Student Progress
v. Assessment System
vi. Delete Course

2.2.3 Student Panel

In this Panel, student can:

i. Make an Account
ii. Select Course
iii. Use text and code editor for assignments
iv. Give online quiz
v. View Analytics
vi. Generate e-certificate
vii. Manage Profile
viii. View and download learning material

2.3 Operating Environment

This is a web-based application that requires a laptop, an Android smartphone, and tablets. The software components that we will require to develop this proposed project are visual studio code and the Code Igniter framework

2.4 Design and Implementation Constraints

In the client side, we will use HTML, CSS, JavaScript, and React.js. Depending on our needs, we chose PostgreSQL. For the backend of our application, we use ROR itself on making the development of web applications faster and more efficient by using a compact software design. Moreover, we developed using extreme programming methodology. As with continuousintegration, all the testing will be done in development phase by J Meter. This purpose project use stripe API because of the payment of paid courses and Google API for emailing.

2.5 Assumptions and Dependencies

Resource-level assumptions play heavily into the success or failure of a project because alarge portion of project management is based on available and allocated resources.

Time-based assumptions are impacted by resource availability, technology, quality factors,financial limitations, and other assumptions.

Quality-specific assumptions play a vital role because all the activity of Learners Arena will be done by this purposed system. In this situation there is usually very little or even no roomfor error.

2.6 Other Non-functional Requirements

2.6.1 Performance Requirements:

The project goal is to provide online education for skill development. Provide a User-friendly application interface on the updated web version so that students may Attend lectures anywhere and at any time with a stable internet connection

2.6.2 Security Requirements

It is made sure that proper authentication, authorization, and validation is applied so that no one other than a valid user should be able to visit the site. Unauthorized Person is not able to access the application.

2.6.3 Software Quality Attributes

Quality Attributes being simple in the project are written bellow:

Availability: As the system is an online accessible and web application, it is easily Accessible. Only thing required to run the application is a web browser having Capability

Adaptability: The system is on any kind of device whether it's a Linux system or Windows or MACOS /IOS or Android. Every OS has a web browser, and it is the only

Thing needed to run the application.

Maintainability: The application can be maintained afterwards as it implemented in

Micro services.

Reliability: The application is certainly reliable. It does not crash occasionally.

Reusability: The code is reusable, and it can be used further for the addition of new Features.

Testability: The application has gone through a phase in which thorough testing was Done.

Usability: The application serves its purpose as described in software requirements.

CHAPTER 3

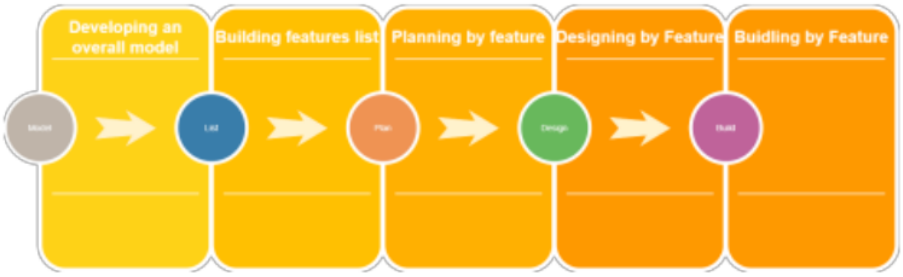
DESIGN AND METHODOLOGY

3.1 Methodology

Feature-driven development (FDD) is a software development methodology that focuses on delivering small, incremental pieces of functionality (called "features") to the customer on a regular basis. It is an agile approach that emphasizes collaboration between developers, customers, and other stakeholders, and it emphasizes the importance of understanding and prioritizing the customer's requirements.

In FDD, the development process is divided into a series of short, iterative cycles, each of which focuses on delivering a specific set of features to the customer. At the beginning of each cycle, the development team works with the customer to identify the most important features that need to be delivered, and they use this information to create a detailed plan for the cycle. The development team then works together to implement the features, using a combination of coding, testing, and refactoring to ensure that the features are delivered on time and with high quality.

FDD is designed to be a flexible and adaptable approach that can be applied to a wide range of software development projects. It is often used in conjunction with other agile methodologies, such as Scrum, to provide a more comprehensive and effective approach to software development.



Following are the features which are going to be the part of application:

i. The user can register himself by signing up into the application.
ii. The learner can view various courses and can choose the course best suited for himself or herself.
iii. To access the course content learner will be needed to pay for the course and this will take him/her to payment module where he/she can pay online.
iv. Teachers will have a dashboard to upload course content, modify or update the content, delete content. Course content will contain videos, handouts, assessments.
v. Teachers will have a dashboard to analyze the results of students, analyze the percentage of people acquiring the course, check the age, gender, race of people joining the course.
vi. This analytic system will be having charts and graphs to show all the stuff mentioned.
vii. Learner will be notified by email, private on the site about course registration, and addition of new course.
viii. Student Support Service will provide automated system to handle queries related to courses and content, give feedback, and submit complaints.
ix. Learners will have analytics system to make them analyze the progress related to handouts reading, video watching, assignment and quizzes submission and grades.
x. Learner will have prediction of grades for upcoming quizzes.
xi. Quizzes will MCQ based, and text based. MCQs will be checked automatically while text based will be checked by instructor himself.
xii. Teacher will check the assignment of the learner.
xiii. A learner will get a report including grades, stats, performance report after the completion of the course.
xiv. Learner will get a completion certificate after the completion of course.
xv. A student can also get promotions and discounts on some specific course for limited time.

3.2 Feasibility Plan

3.2.1 Resource Requirement

i. IDEs
ii. Hardware minimum is core i3, 8GB Ram
iii. Server-side scripting
iv. Programming tools

The resources mentioned above may vary according to our needs

3.2.2 Tools / Technology

i. HTML
ii. CSS
iii. Material UI
iv. JavaScript
v. React.js
vi. ROR
vii. PostgreSQL

3.2.3 JavaScript:

JavaScript is a high-level, interpreted programming language. It is widely used for building web and mobile applications, and it is one of the core technologies of the World Wide Web. JavaScript is a client-side language, which means that it is executed by the user's web browser, rather than on the server. This allows JavaScript to provide a rich and interactive user experience, by allowing web pages to respond to user input in real-time. JavaScript is an object-oriented language, which means that it allows developers to define and manipulate objects that have attributes and behaviours. It is also a dynamically typed language, which means that you don't need to specify the data type of a variable when you declare it.

3.2.4 React:

React is a JavaScript library used to create user interfaces. It is maintained by Facebook, Instagram, and a community of individual developers and corporations. React allows developers to create reusable UI components, which can be used in applications to create a user interface that is simple, fast, and scalable.

3.2.5 Redux:

Redux is a predictable state container for JavaScript apps. It is often used with React to manage application state in a scalable and maintainable way. Redux allows you to manage your application state in a single, immutable store, making it easier to debug and reason about your code. It also provides several tools for managing asynchronous actions, like network requests, in a consistent and predictable way.

3.2.6 Ruby:

Ruby is a general-purpose, interpreted programming language. It was created in 1995 by Yukihiro "Matz" Matsumoto in Japan. Ruby is dynamically typed, meaning you don't need to specify the data type of a variable when you declare it. It is also object-oriented, meaning it allows you to define and manipulate objects that have attributes and behaviours. Ruby is known for its simplicity and elegance, and it is often used for web development, data analysis, and system automation.

3.2.7 Ruby on rails:

Ruby on Rails, or Rails, is a web application framework written in the Ruby programming language. It was created in 2003 by David Heine Meier Hansson while working on the project management tool Basecamp. Rails is designed to make it easier to develop web applications by providing a set of pre-defined structures for commonly needed tasks, such as routing requests, handling database interactions, and rendering views. This allows developers to focus on building the unique features of their applications, rather than having to write the same boilerplate code for every project. Rails is known for its "convention over configuration" philosophy, which emphasizes the use of well-established best practices and sensible defaults, rather than requiring developers to specify every detail of the application's behaviour.

3.2.8 PostgreSQL:

PostgreSQL is an open-source object-relational database management system (ORDBMS).PostgreSQL was created in 1996 as a fork of the Ingres database, and it has a long history of stability and reliability. It is known for its support for advanced data types and features, such as arrays, custom data types, and full-text search. PostgreSQL also has strong support for transactional integrity, and it can be used to store and manage data for a wide variety of applications.

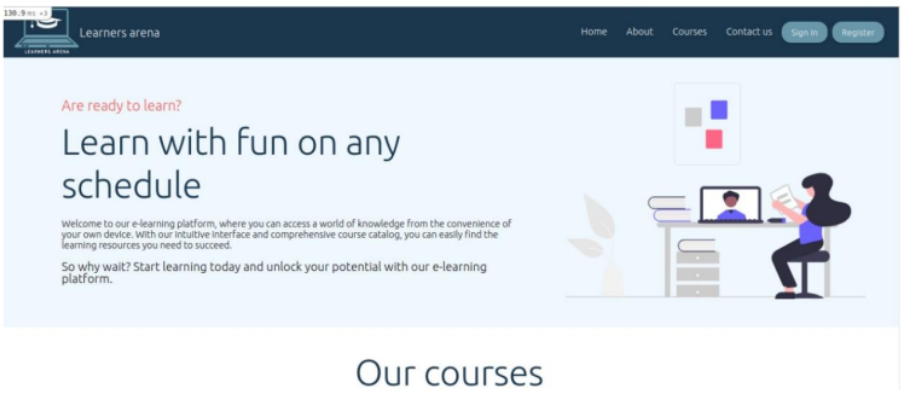
3.2.9 Stripe API:

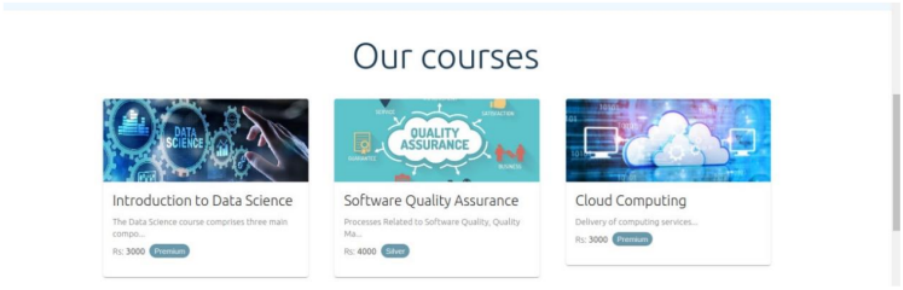
Stripe is a suite of payment APIs that allows developers to easily integrate online payment processing into their applications. The Stripe API provides a set of services that can be used to process payments, manage customers, and handle subscription billing. It is a popular choice for developers because it is easy to use, well documented, and has a robust set of features. The Stripe API is available in multiple languages, including Ruby, Python, and JavaScript.

3.2.10 Material UI:

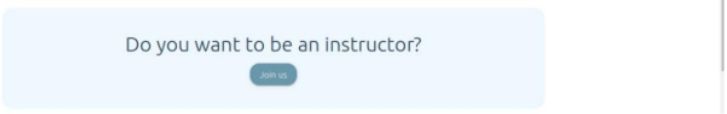
Material-UI is a popular React-based UI framework for building responsive, accessible, and consistent user interfaces. It is built on top of Google's Material Design guidelines, which provide a visual language for design that is based on user research and incorporates the latest advances in technology. Material-UI provides a set of pre-built components that implement the Material Design guidelines and can be easily integrated into a React application. These components are designed to be customizable and adaptable, so that developers can create unique and intuitive interfaces that fit the needs of their users.

3.3 User Interface




Our courses
DATA SCIENCE
Introduction to Data Science
The Data Science course comprises three main compo...
Rs: 3000
Premium
QUALITY ASSURANCE
Software Quality Assurance
Processes Related to Software Quality, Quality Ma...
Rs: 4000
Silver
Cloud Computing
Delivery of computing services...
Rs: 3000
Premium


Helping people to grow their careers, everyday!
In addition to our extensive library of courses, our platform also offers a range of tools and resources to help you stay on track and achieve your goals. You can track your progress, participate in discussions with other learners, and receive personalized recommendations based on your interests and learning style.


Do you want to be an instructor?
Join us

3.4 System Use Case:

3.4.1 Use case Diagrams

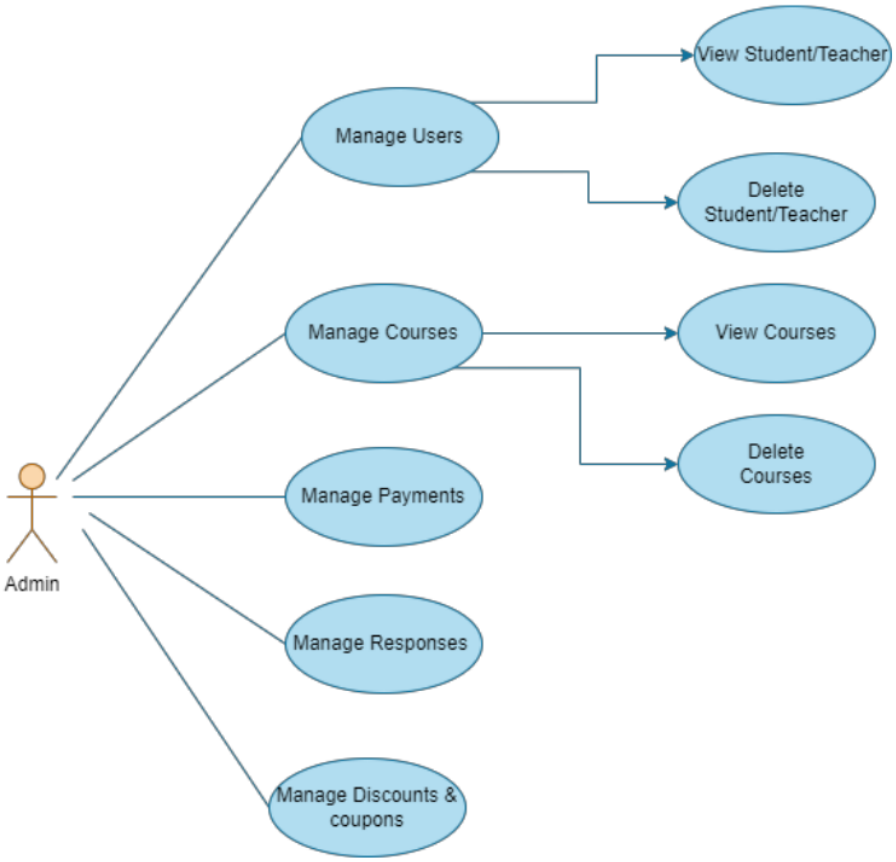


Figure 3. 1: Admin panel use case Diagram

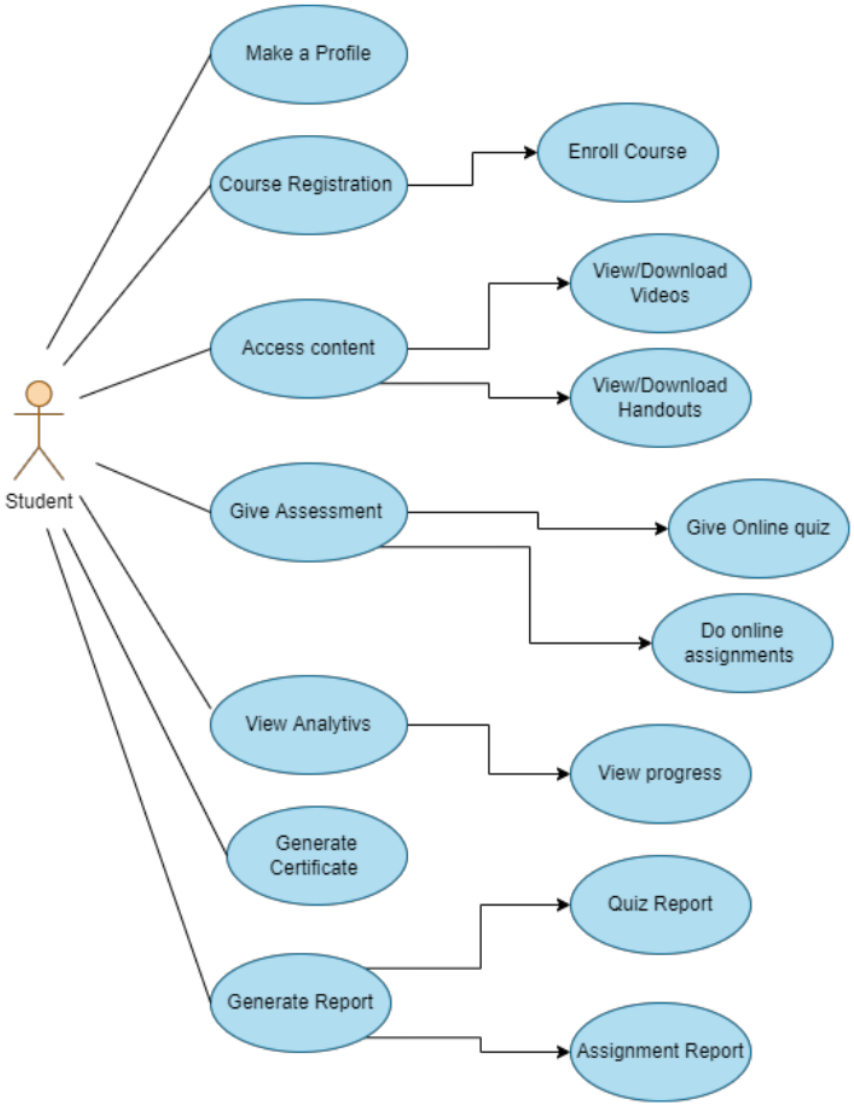


Figure 3. 2: Student panel use case Diagram

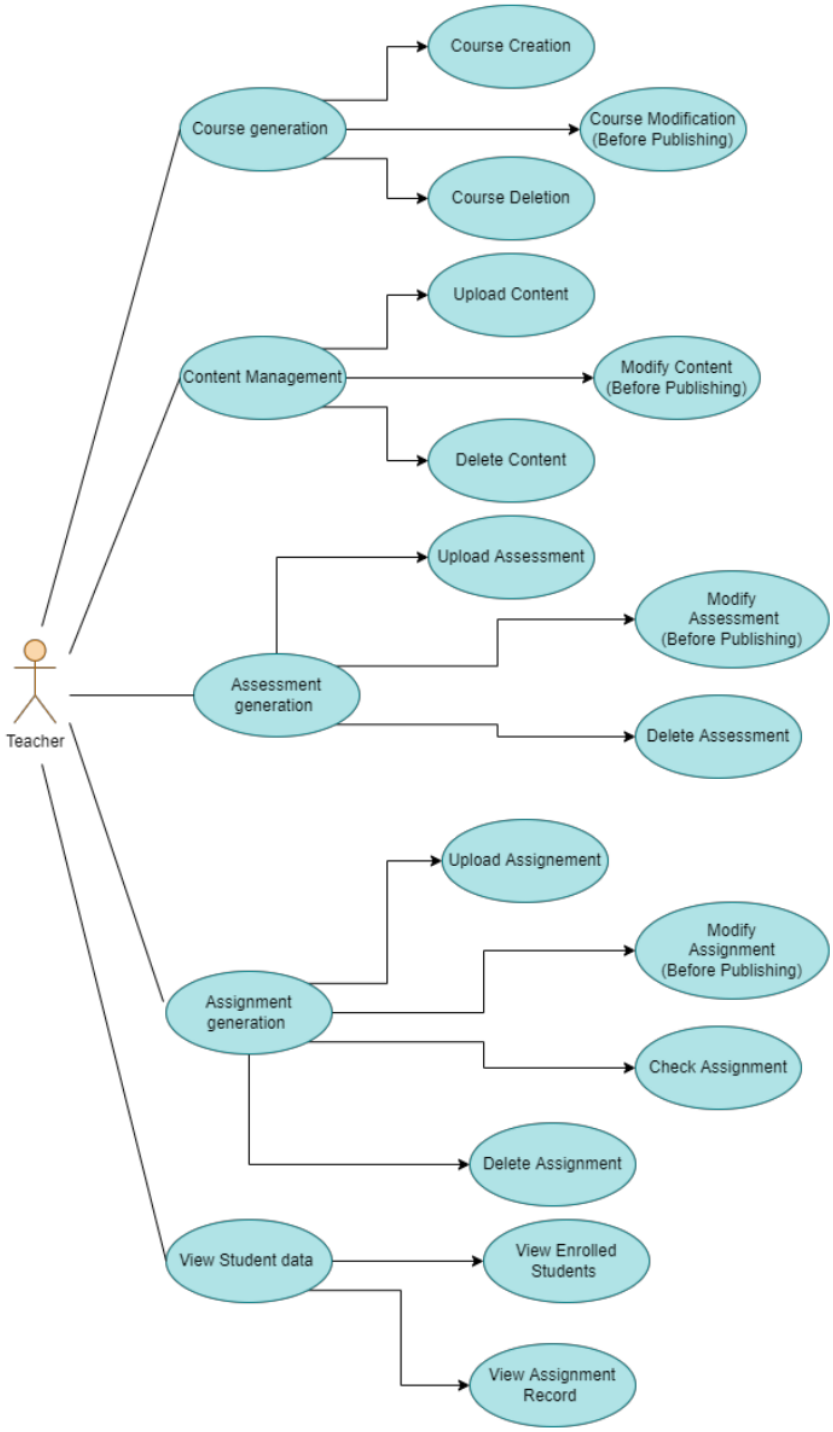


Figure 3. 3: Teacher panel use case Diagram

3.4.2 Manage Users (LA-01-001)

This use case is about the admin. He login to the application to manage the users

Name	Manage Users
Unique Identifier	LA-01-001
Objective	Managing the users
Priority	High
Actors	Admin
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Enter admin credentials. 3. Press sign in to enter in admin panel. 4. View Students & Teachers 5. Click on delete for deletion of users
Alternative Flow	1. Visit website http://127.0.0.1:3000/ 2. Change the URL to change it in admin panel 3. Fill fields and manage users
Preconditions	User must open the website
Post Conditions	Admins manage the users accordingly.
Notes/Issues	This use case describes the event of manage users, admin is already registered. Internet may not work

Table 3. 1: Manage Users Use Case

3.4.3 Manage Courses (LA-01-002).

This use case is about the admin he login to the application to manage the courses

Name	Manage Courses
Unique Identifier	LA-01-002
Objective	Managing the Courses
Priority	High
Actors	Admin
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Enter admin credentials. 3. Press sign in to enter in admin panel. 4. View Courses 5. Click on delete for deletion of course
Alternative Flow	1. Visit website http://127.0.0.1:3000/ 2. Change the URL to change it in admin panel 3. Fill fields and manage courses
Preconditions	Admin must open the website
Post Conditions	Admin is logged in and manage courses
Notes/Issues	Internet may not work

Table 3. 2: Admin Use Case

3.4.4 Manage Payments (LA-01-003)

In this admin can manage payments of paid courses

Name	Manage Payments
Unique Identifier	LA-01-003
Objective	This use case initiates when Admin is going to manage payment
Priority	Medium
Actors	Admin
Basic Flow	**1.** Visit website http://127.0.0.1:3000/ **2.** Enter admin credentials. **3.** Press sign in to enter in admin panel. **4.** Add amount and categories of paid courses. **5.** Press enters to finalize.
Alternative Flow	1. Visit website http://127.0.0.1:3000/ 2. Change the URL to change it in admin panel 3. Fill fields and manage payment of paid courses.
Preconditions	Admin must be login
Post Conditions	Admin add the criteria of paid courses
Notes/Issues	Internet may not work

Table 3. 3: Manage Payment Use Case

3.4.5 Manage Responses (LA-01-004)

This is use case is for the admin to manage the responses given by students.

Name	Manage Responses
Unique Identifier	LA-01-004
Objective	This use case is to see the responses given by student
Priority	Medium
Actors	Admin
Basic Flow	• Visit website http://127.0.0.1:3000/ • Enter admin credentials. • Press sign in to enter in admin panel. • Click on side menu • Select Reponses section • View the responses along with categories and time of response
Alternative Flow	1. Visit website http://127.0.0.1:3000/ 2. Change the URL to change it in admin panel 3. Fill fields and see the Reponses.
Preconditions	Admin must be logged in.
Post Conditions	Admin view the Reponses
Notes/Issues	Internet may not work

Table 3. 4: Manage Responses Use Case

3.4.6 Manage discounts and coupons (LA-01-005)

The admin manage the discounts and coupons for paid courses

Name	Manage classes
Unique Identifier	LA-01-005
Objective	To give the discounts on paid courses
Priority	Medium
Actors	Admin
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Enter admin credentials. 3. Press sign in to enter in admin panel. 4. Click on side menu 5. Select coupons section 6. Click on generate coupons 7. Discounted coupon generated
Alternative Flow	1. Visit website http://127.0.0.1:3000/ 2. Change the URL to change it in admin panel 3. Fill fields and generate discounted coupons.
Preconditions	Admin must be logged in.
Post Conditions	Discounted coupons generated
Notes/Issues	Internet may not work

Table 3. 5: Manage Classes Use Case

3.4.7 Course Generation (LA-01-006)

In this use case, Teacher generate course and make change in it before publishing

Name	Generate Course
Unique Identifier	LA-01-006
Objective	To create courses
Priority	High
Actors	Teacher
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. New Teacher Click on sign up 3. Make a profile 4. Sign into teacher panel by giving credentials 5. Click on side menu 6. Select add course 7. Fill details of courses 8. Click on add 9. Courses added
Alternative Flow	Alt-1. Teacher is not logged in due to wrong username or password Alt-2. Admin couldn't add course
Preconditions	Teacher must be login
Post Conditions	Courses added from teacher panel
Notes/Issues	Internet may not work

Table 3. 6: Generate Course Use Case

3.4.8 Manage Content (LA-01-007)

In this use case teacher will manage the learning content of course

Name	Manage Content
Unique Identifier	LA-01-007
Objective	To upload and manage the content of course
Priority	High
Actors	Teacher
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into teacher panel by giving credentials 3. Click on side menu 4. Select what kind of content he/she wants to upload 5. Fill details of relevant content 6. Click on add 7. Content added to respective week 8. After Completing the course content 9. Click on publish button
Alternative Flow	Alt-1. Teacher is not logged in due to wrong username or password Alt-2. Teacher clicks on another button rather than adding content.
Preconditions	Teacher must be login Course must be created
Post Conditions	Learning Content added
Notes/Issues	Internet may not work

Table 3. 7: Manage Content Use Case

3.4.9 Assessment generation (LA-01-008)

The use case is about the creation of assessment by teacher panel

Name	Assessment Generation
Unique Identifier	LA-01-008
Objective	This use case initiates when teacher is going to create assessment.
Priority	Medium
Actors	Teacher
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into teacher panel by giving credentials 3. Click on side menu 4. Click on Add Quiz 5. Fill the details 6. Add Questions in it may be multiple choice or short answer 7. Click on Add to publish it
Alternative Flow	Alt-1. Teacher is not logged in due to wrong username or password Alt-2. Teacher clicks on another button rather than adding quiz
Preconditions	Teacher must be login Course must be created
Post Conditions	Quiz Created
Notes/Issues	Internet may not work

Table 3. 8: Assessment Generation Use Case

3.4.10 Assignment Generation (LA-01-009)

This use case is for teacher to create Assignment generation

Name	Assignment Generation
Unique Identifier	LA-01-009
Objective	To create Assignment for the student
Priority	Medium
Actors	Teacher
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into teacher panel by giving credentials 3. Click on side menu 4. Click on Add Assignment 5. Fill the details 6. Add Questions in it 7. Click on Add to publish it
Alternative Flow	Alt-1. Teacher clicks on another button rather to add assignment
Preconditions	Teacher must be login Course must be created
Post Conditions	Assignment Added
Notes/Issues	Internet not working

Table 3. 9: Assignment Generation

3.4.11 View Student (LA-01-010)

Teacher use case to view enrolled students and their assignments record

Name	View Student
Unique Identifier	LA-01-010
Objective	To view enrolled students and their assignment record
Priority	High
Actors	Teacher
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into teacher panel by giving credentials 3. Click on the course card 4. View the enrolled students 5. Click on view to see assignment record
Alternative Flow	Alt-1. Teacher clicks on another button clicks on another button
Preconditions	Teacher must be login into his portal Course must be created
Post Conditions	Student viewed and assignment checked
Notes/Issues	Internet not working

Table 3. 10: View Student Use Case

3.4.12 Make Profile (LA-01-011)

Student use case in which he/she can create its profile

Name	Make Profile
Unique Identifier	LA-01-011
Objective	Make profile of student
Priority	High
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. New Student Click on sign up 3. Make a profile 4. Sign into Student panel by giving credentials 5. Student Profile created
Alternative Flow	Alt-1. New Student click on sign in instead of signing up Alt-2 Wrong username or password entered by student
Preconditions	Student must be sign up
Post Conditions	Student profile created
Notes/Issues	Internet not working

Table 3. 11: Make Profile Use Case

3.4.13 Course Registration (LA-01-012)

Student use case to register the course

Name	Course Registration
Unique Identifier	LA-01-012
Objective	To register the course by student
Priority	High
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on all courses 4. Select what course want to view by click view button 5. Click on enrol button for enrolling in particular course 6. Click on buy button to purchase paid course
Alternative Flow	Alt-1. Student click on another button instead of enrol, buy or view
Preconditions	Student must be login Course must be shown
Post Conditions	Enrolled in that course
Notes/Issues	Internet not working

Table 3. 12: Course Registration Use Case

3.4.14 Access Content (LA-01-013)

Student use case to access the learning content if the course

Name	Access Content
Unique Identifier	LA-01-013
Objective	To access the learning content of the course
Priority	Medium
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on all courses 4. Select what course want to view by click view button 5. Click on enrol button for enrolling in particular course 6. Click on that week which you want to access 7. Click on videos to watch video of that week 8. Click on hand out to see hand-outs of that week
Alternative Flow	Alt-1. Click on another button instead of watching content
Preconditions	Student must be login
Post Conditions	Student access the content of course
Notes/Issues	Internet not working

Table 3. 13: Access Content Use Case

3.4.15 Give Assessment (LA-01-014)

Student Use case to give the assessment of the particular course

Name	Give Assessment
Unique Identifier	LA-01-014
Objective	To evaluate the performance of student in Course
Priority	Medium
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on the Dashboard 4. Select the course 5. Click on that specific week which you want access 6. Click on quiz and start attempting 7. Click on Assignment to attempt it
Alternative Flow	Alt-1. Click on another button instead of selecting week Alt-2. Assignment or quiz is not created on that week from teacher end
Preconditions	Teacher must be login
Post Conditions	Assignment or quiz attempted successful
Notes/Issues	Internet not working

Table 3. 14: Give Assessment Use Case

3.4.16 View Analytics (RTA-01-015)

Student use case to view analytics of quizzes and assignments

Name	View Analytics
Unique Identifier	LA-01-015
Objective	To view Analytics of Quiz and Assignments
Priority	low
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on the Dashboard 4. Select the course 5. Click on analytics 6. View the analytical view of your progress
Alternative Flow	Alt-1. Click on another button instead of view analytics Alt-2. Wrong course selection
Preconditions	Student must be login Must be enrol in student
Post Conditions	Analytics viewed
Notes/Issues	Internet not working

Table 3. 15: View Analytics Use Case

3.4.17 Generate Certificate (LA-01-016)

Student Use case for generation of certificate

Name	Generate Certificate
Unique Identifier	LA-01-016
Objective	To get the certificate after completion
Priority	Medium
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on the Dashboard 4. Select the course 5. Click on certificate 6. Click on generate certificate
Alternative Flow	Alt-1. Click on another button instead of Certificate generation
Preconditions	Student must be login Student complete the criteria of Certificate completion
Post Conditions	Certificate generated
Notes/Issues	Internet not working

Table 3. 16: Generate Certificate Use Case

3.4.18 Generate Report (LA-01-017)

Student Use case for generation report

Name	Generate Report
Unique Identifier	LA-01-017
Objective	To get the report of quiz and assignment
Priority	Medium
Actors	Student
Basic Flow	1. Visit website http://127.0.0.1:3000/ 2. Sign into student panel by giving credentials 3. Click on the Dashboard 4. the course 5. Click on certificate 6. Click on generate certificate
Alternative Flow	Alt-1. Click on another button instead of Certificate generation
Preconditions	Student must be login Student complete the criteria of Certificate completion
Post Conditions	Certificate generated
Notes/Issues	Internet not working

Table 3. 17: Generate Report Use Case

3.5 Sequence Diagram

3.5.1 Role of admin

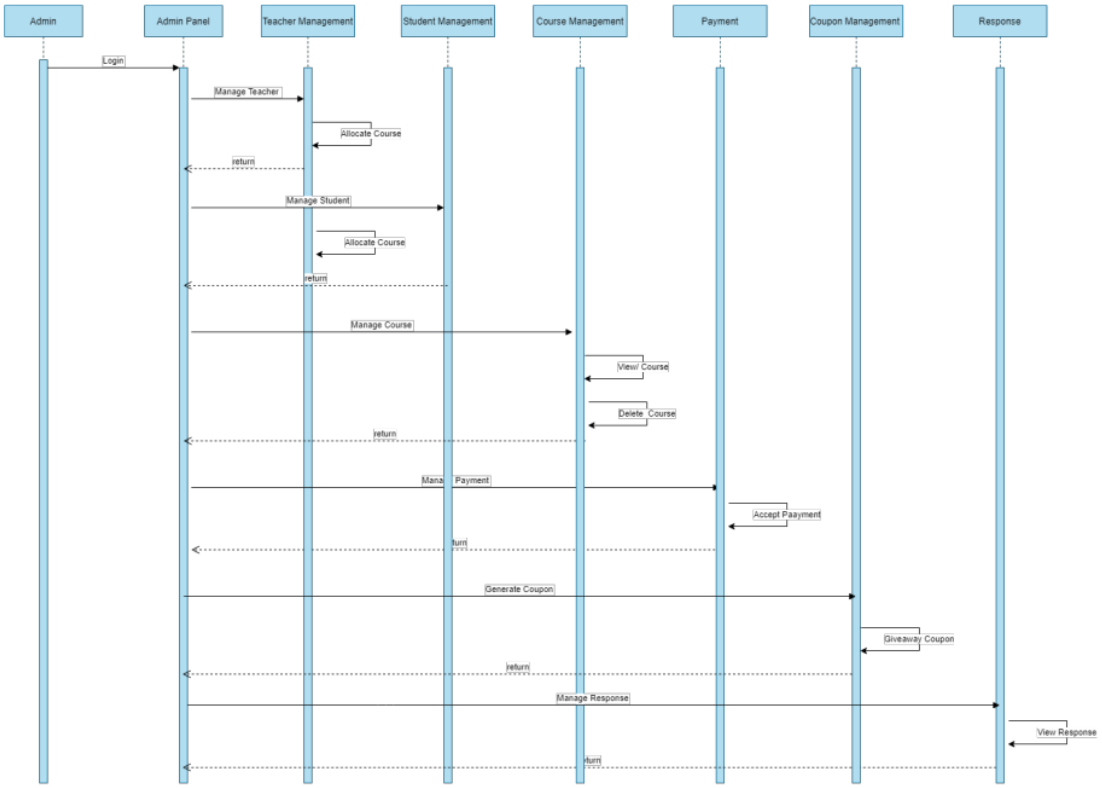


Figure 3. 4: Admin sequence diagram

3.5.2 Role of teacher

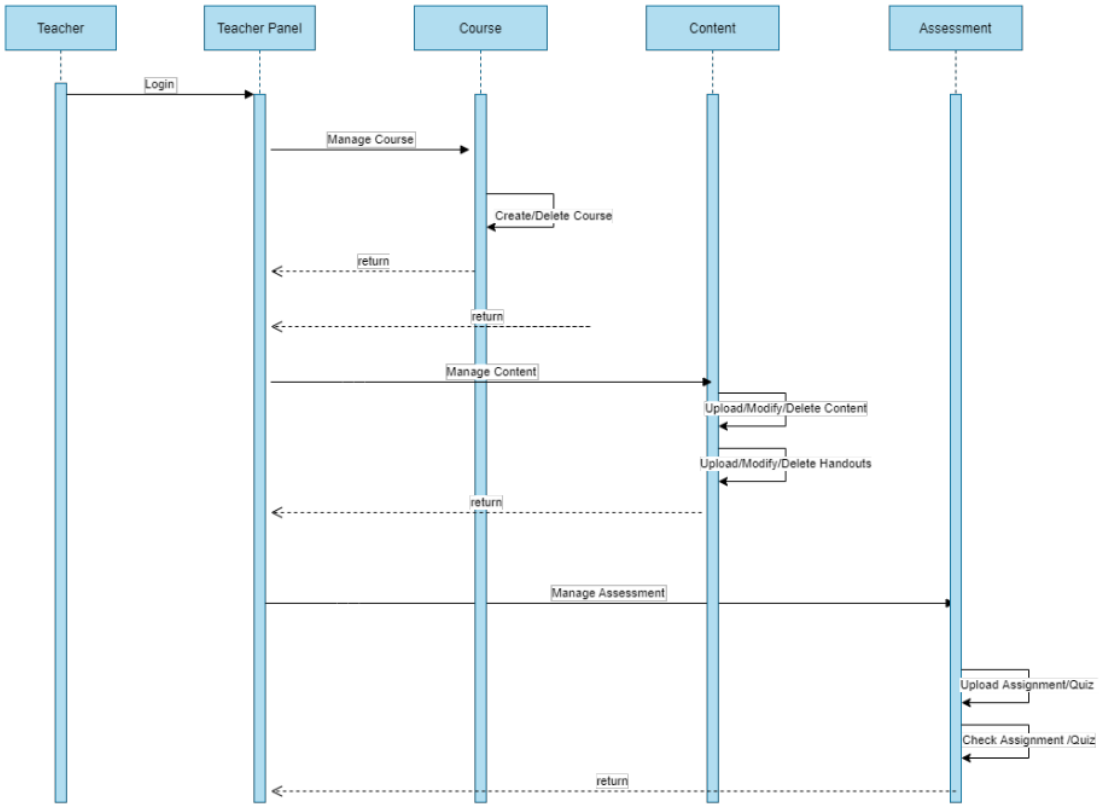


Figure 3. 5: Teacher sequence diagram

3.5.3 Role of Student

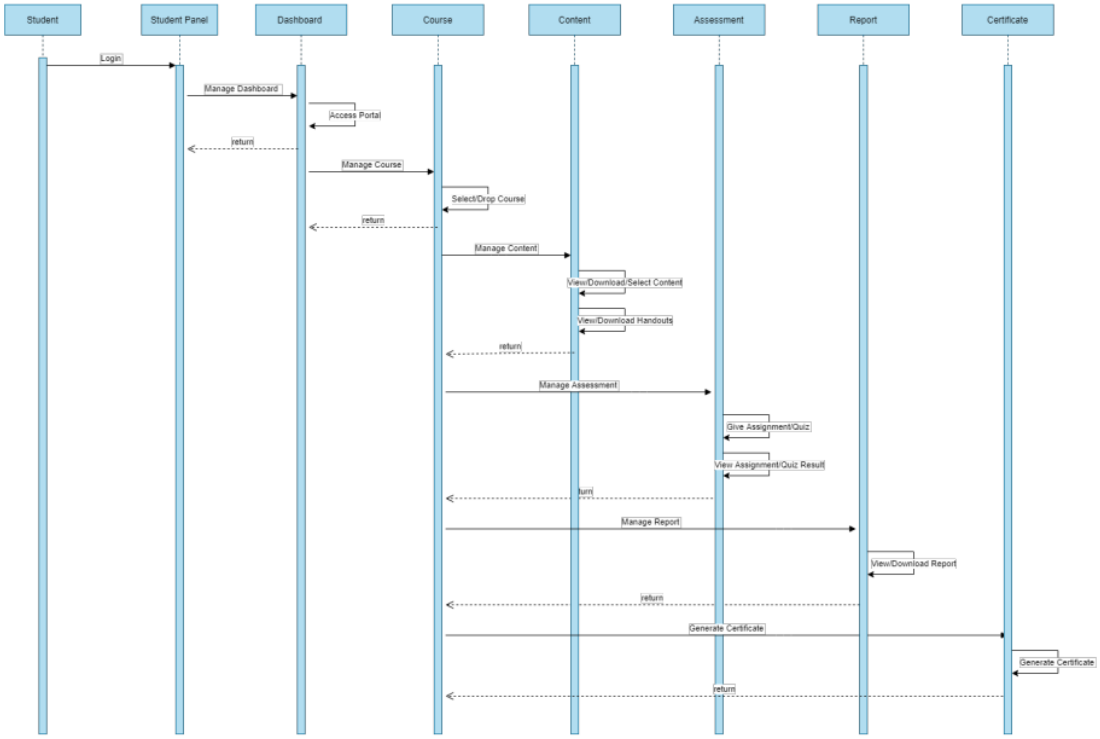


Figure 3. 6: Student sequence diagram

3.6 Collaboration Diagram

3.6.1 Admin Panel

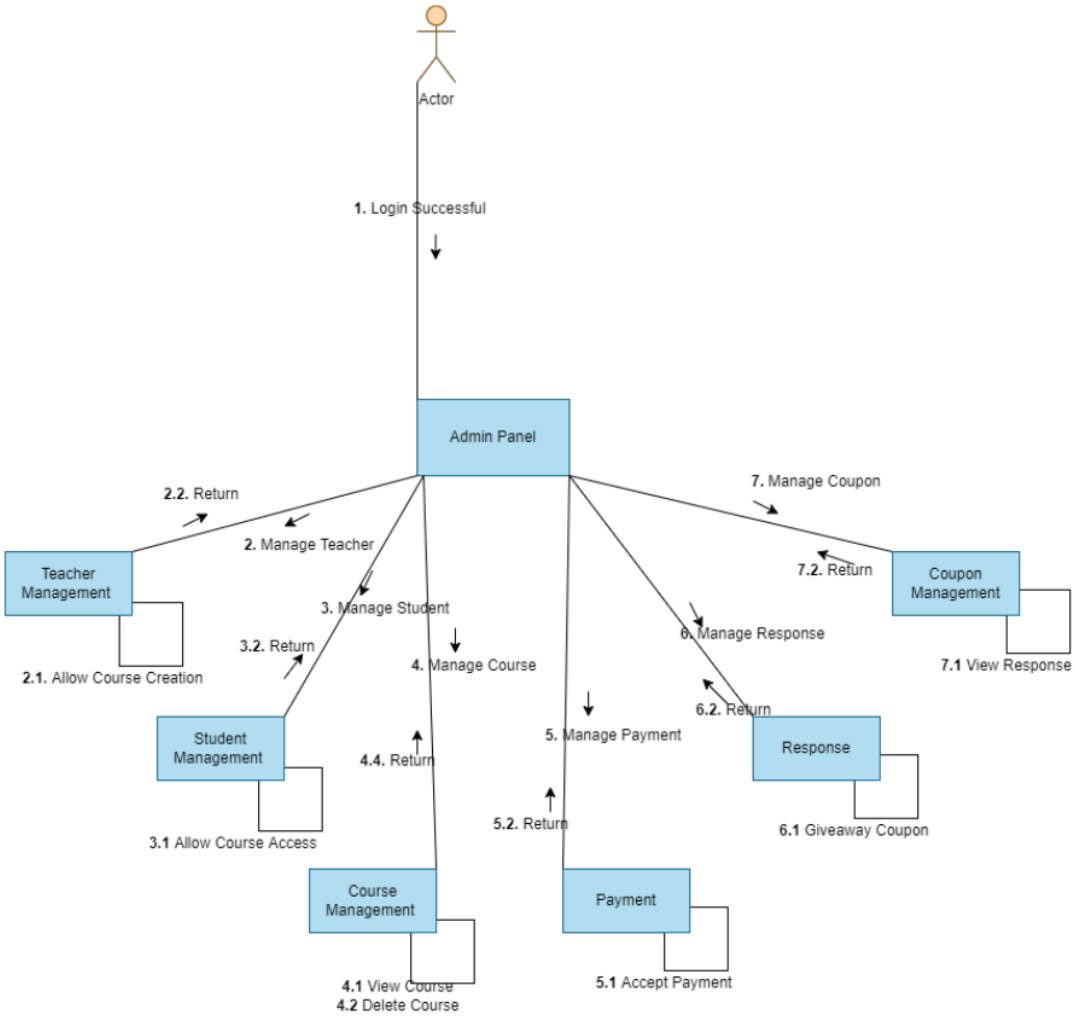


Figure 3. 7: Admin Panel Collaboration Diagram

3.6.2 Student Panel

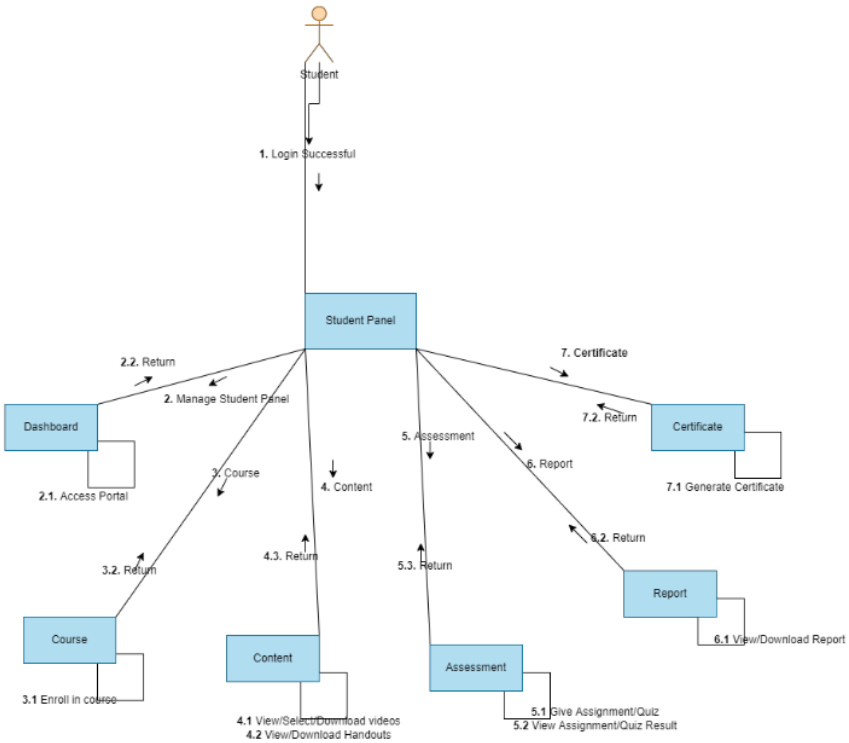


Figure 3. 8: Student Panel Collaboration Diagram

3.6.3 Teacher Panel

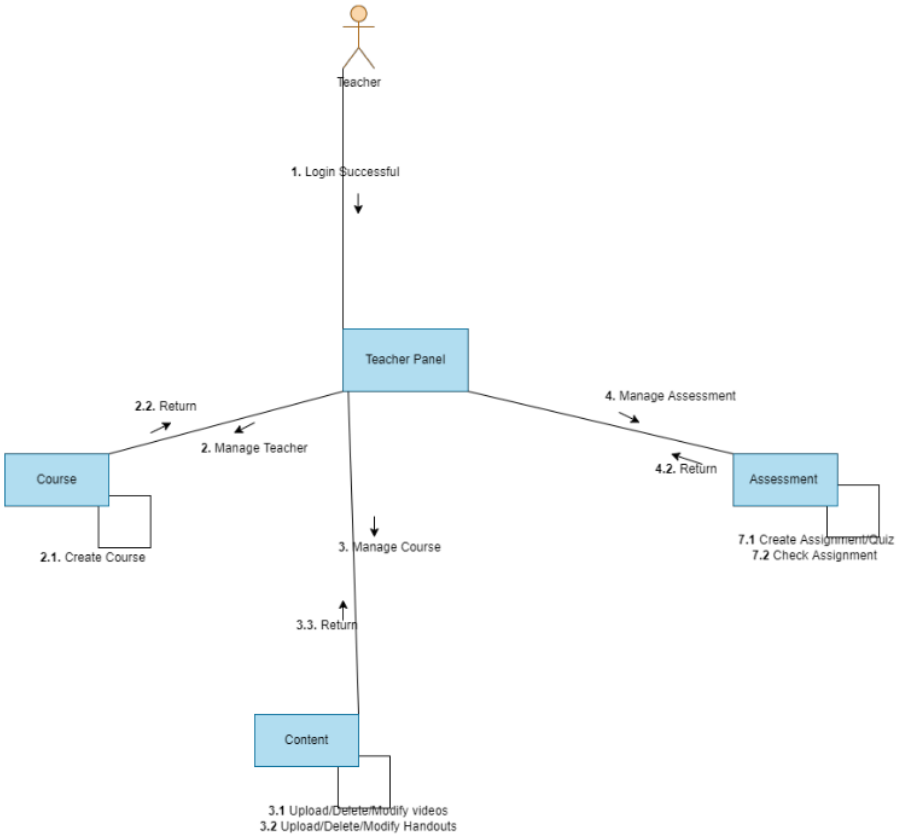


Figure 3. 9: Teacher Panel Collaboration Diagram

3.7 Entity Relationship Diagram

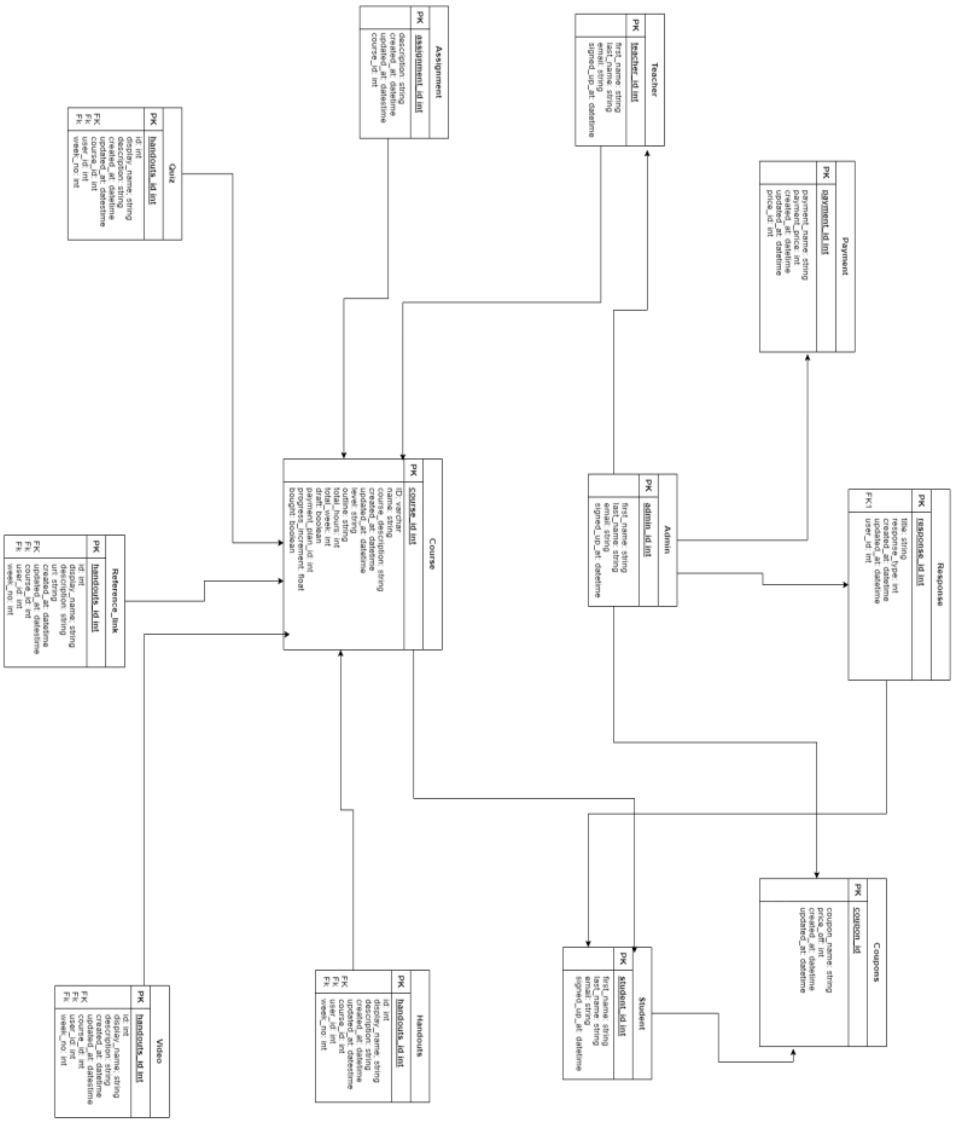


Figure 3. 10: Entity Relationship Diagram

3.8 Class Diagram

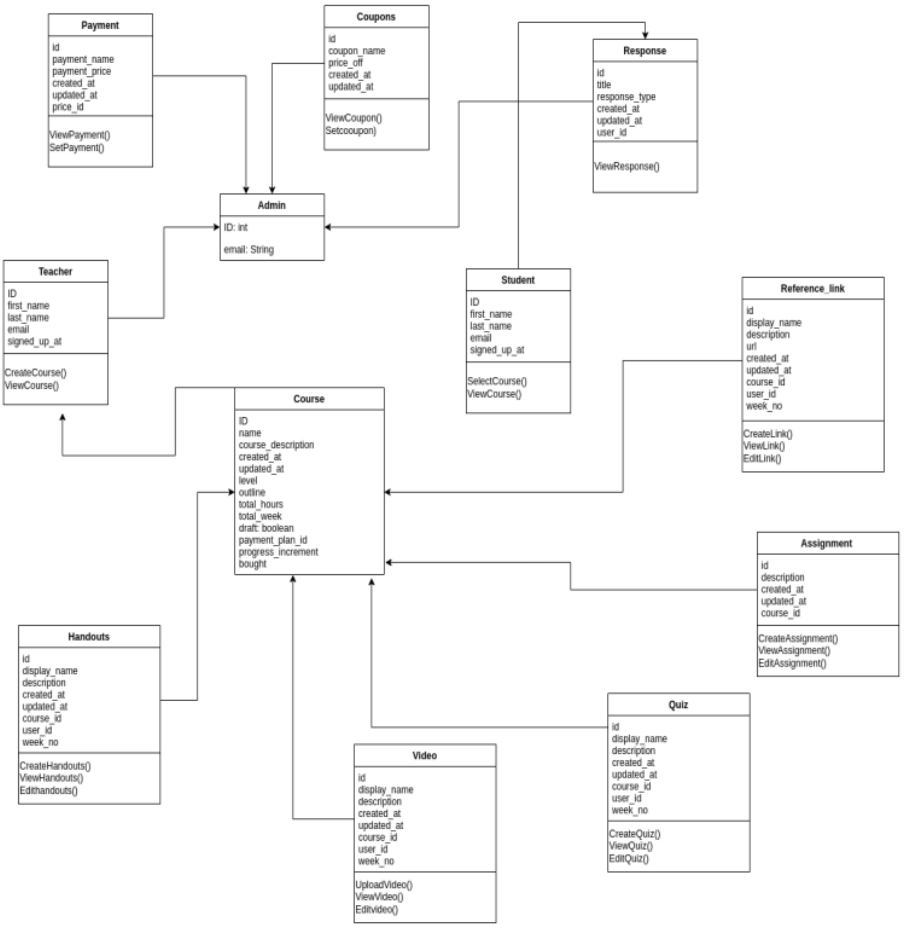


Figure 3. 11: Class Diagram

3.9 Domain Model

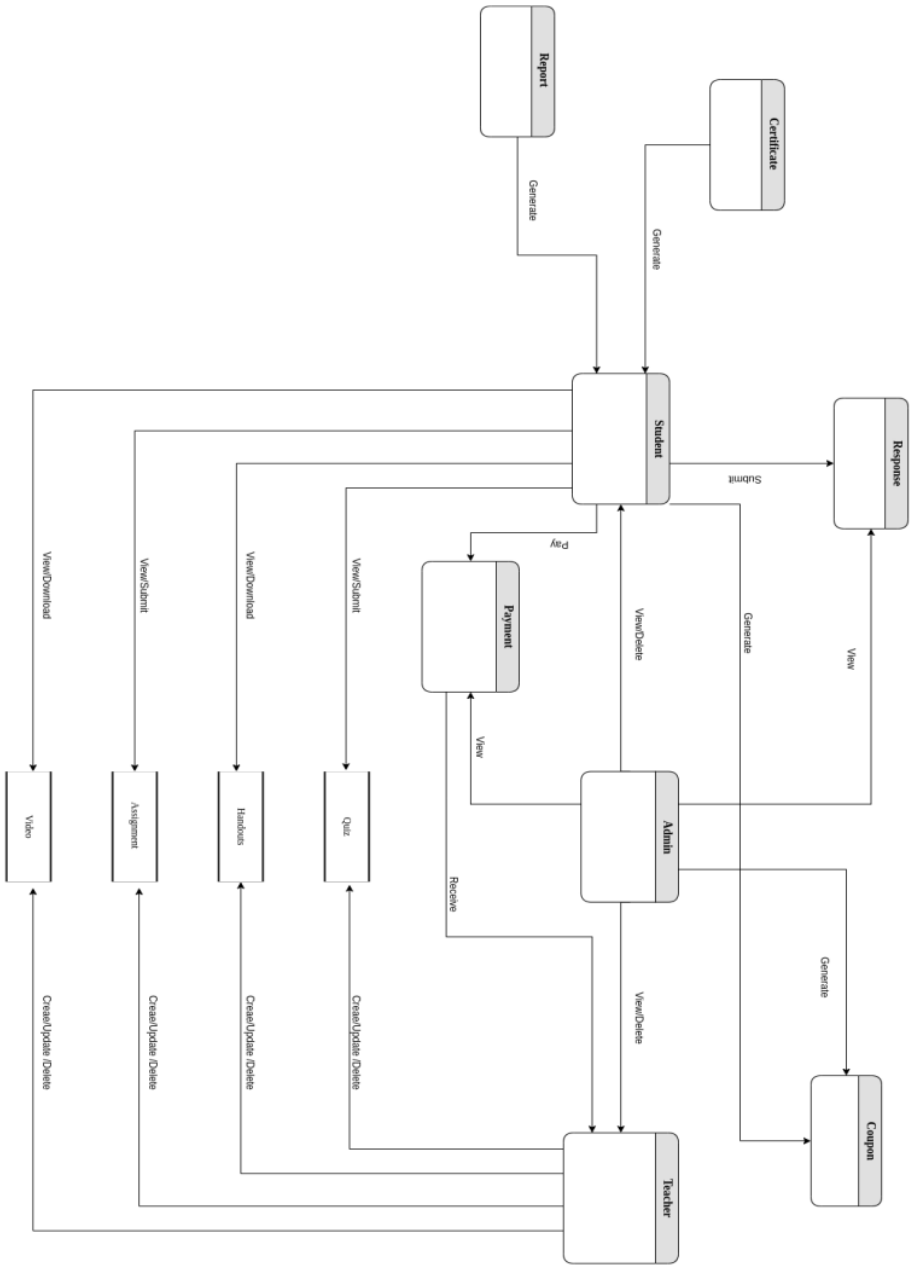


Figure 3. 12: Domain Model

3.10 Operation Contracts

When an operation occurs, the system state is changed according to a UML Operation contract. It will essentially specify what each system operation accomplishes. An operation is extracted from Learners Arena's system flow diagram. It is just one instance of that diagram's event.

3.10.1 Operation Contracts Sign in

Name: Sign in
Responsibilities: To sign in Users
Cross References: Sign in
Exceptions: User entered wrong credentials or not provide full info
Preconditions: Go on http://127.0.0.1:3000/
Post conditions: User Sign in

3.10.2 Operation Contracts Sign up

Name: Sign up
Responsibilities: To sign up Users
Cross References: Sign up
Exceptions: User does not provide full information
Preconditions: Go on http://127.0.0.1:3000/
Post conditions: User Sign up

3.10.3 Operation Contracts Manage Users

Name: Manage Users
Responsibilities: To manage users of the application
Cross References: Manage Users
Exceptions: Wrong credentials provided by admin
Preconditions: Go on http://127.0.0.1:3000/ and login as admin
Post conditions: User Managed by admin

3.10.4 Operation Contracts Manage Payments

Name: Manage Payments

Responsibilities: To make the categories of paid courses

Cross References: Manage Payment

Exceptions: Wrong credentials provided by admin

Preconditions: Go on http://127.0.0.1:3000/ and login as admin

Post conditions: Paid Courses categories added by admin

3.10.5 Operation Contracts Manage Responses

Name: Manage Responses

Responsibilities: Seeing the responses given by student

Cross References: Manage Responses

Exceptions: User entered wrong credentials or wrong option selected

Preconditions: Go on http://127.0.0.1:3000/ and login as admin

Post conditions: Responses seen by admin

3.10.6 Operation Contracts Manage Discounted Coupons

Name: Manage Discounted Coupons

Responsibilities: To Manage the discounted coupons for paid courses

Cross References: Manage Discounted Coupons

Exceptions: User entered wrong credentials or wrong option selected

Preconditions: Go on http://127.0.0.1:3000/ and login as admin

Post conditions: Discounted Coupon Added by admin

3.10.7 Operation Contracts Make Profile

Name: Make Profile

Responsibilities: Make a student Profile

Cross References: Make Profile

Exceptions: Already Student Profile exists

Preconditions: Go on http://127.0.0.1:3000/ and sign up as student

Post conditions: Student Profile build

3.10.8 Operation Contracts Course Registration

Name: Course Registration

Responsibilities: To Register the course

Cross References: Course Registration

Exceptions: Student select the wrong course

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Course enrolled by student

3.10.9 Operation Contracts Access Content

Name: Access Content

Responsibilities: To Access course content

Cross References: Access Content

Exceptions: Student select wrong course or the wrong content week

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Course content accessed by the student

3.10.10 Operation Contracts Give Assessment

Name: Give Assessment

Responsibilities: To give the assessment of enrolled course

Cross References: Give Assessment

Exceptions: Wrong course selected by student or the week of course

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Assessment given by student

3.10.11 Operation Contracts View Analytics

Name: View Analytics

Responsibilities: To view the progress of course

Cross References: View Analytics

Exceptions: Student Select wrong option

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Analytics viewed by student

3.10.12 Operation Contracts Generate Certificate

Name: Generate Certificate

Responsibilities: After the completion of course by student

Cross References: Generate Certificate

Exceptions: Student trying to generate without complete completion criteria

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Course completion certificate generated

3.10.13 Operation Contracts Generate Reports

Name: Generate Reports

Responsibilities: To view the results of quizzes and assignments

Cross References: Generate Reports

Exceptions: Student do not attempting even a single quiz or assignment

Preconditions: Go on http://127.0.0.1:3000/ and login as student

Post conditions: Course assessments report generated

3.10.14 Operation Contracts Course generation

Name: Course generation

Responsibilities: To create the course

Cross References: Course generation

Exceptions: Teacher do not exist in application or select any other option

Preconditions: Go on http://127.0.0.1:3000/ and login as teacher

Post conditions: Course generated

3.10.15 Operation Contracts Content Management

Name: Content Management

Responsibilities: To manage the content of course

Cross References: Content Management

Exceptions: Teacher selected the wrong course

Preconditions: Go on http://127.0.0.1:3000/ and login as teacher

Post conditions: Course managed and published

3.10.16 Operation Contracts Assessment Generation

Name: Assessment Generation

Responsibilities: For evaluation of students create assessment

Cross References: Assessment Generation

Exceptions: Teacher selected wrong or wrong week to create assessment

Preconditions: Go on http://127.0.0.1:3000/ and login as teacher

Post conditions: Assessment generated by teacher

3.10.17 Operation Contracts Assignment Generation

Name: Assignment Generation

Responsibilities: To generate the assignment for course

Cross References: Assignment Generation

Exceptions: Trying selected wrong course or wrong week

Preconditions: Go on http://127.0.0.1:3000/ and login as teacher

Post conditions: Assignment generated by teacher panel

3.10.18 Operation Contracts View Student Data

Name: View Student Data

Responsibilities: To view the enrolled students in course and their progress

Cross References: View Student Data

Exceptions: No student enroll in the course or no course created by teacher

Preconditions: Go on http://127.0.0.1:3000/ and login as teacher

Post conditions: Student data viewed

CHAPTER 4

IMPLMENTATION

4.1 Implementation and Testing

4.1.1 Frontend Compatibility:

We have tested our user interface on multiple screens types and different browsers and there is no bug or defect in our user interface.

4.1.2 Resource Loading Time:

We have compressed the images used in our website so that the resource loading time can be reduced.

4.1.3 Load Testing:

We have used J Meter to test different pages sign up, sign in, home with multiple users and result are as following:

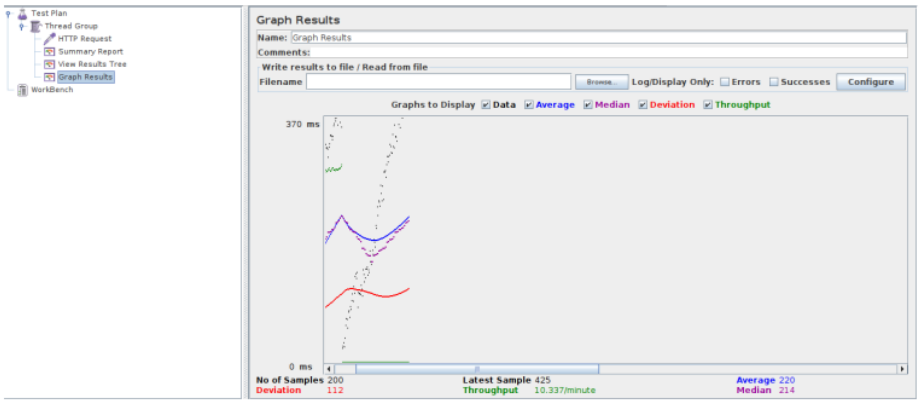


Figure 4. 1: Home

We have tested sign in with 200 samples and received number of deviation is 112 and throughput time is 10.337/minute with the average of 220 and median of 214

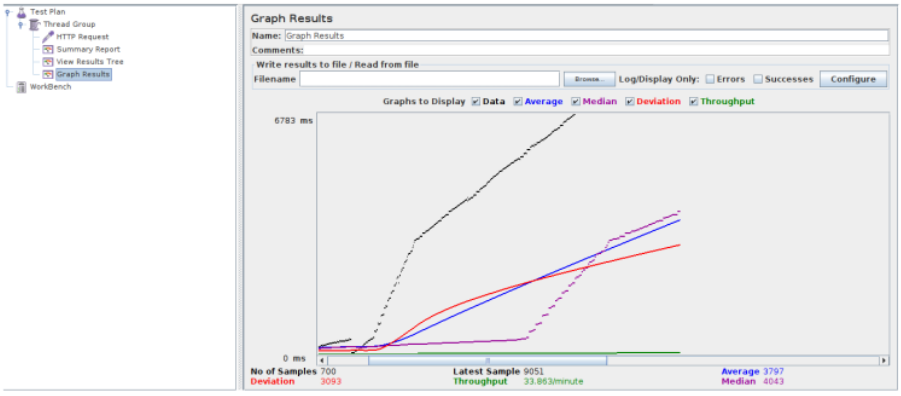


Figure 4. 2: Sign in

We have tested sign in with 700 samples and received number of deviation is 3093 and throughput time is 33.863/minute with the average of 3797 and median of 4043

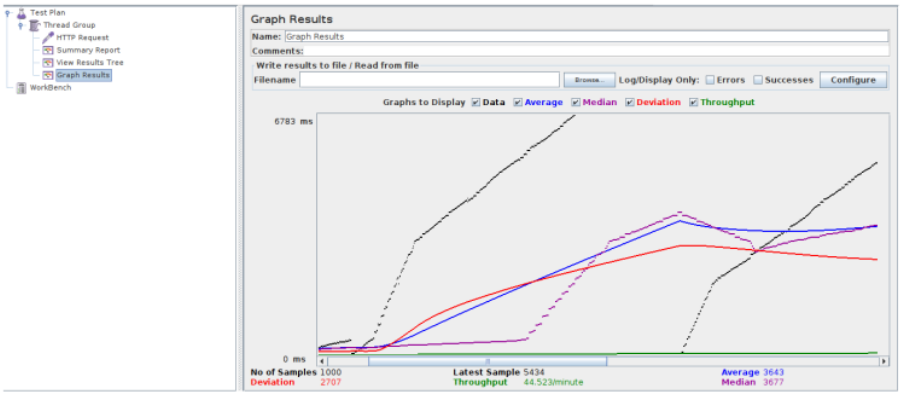


Figure 4. 3: Sign up

We have tested sign in with 1000 samples and received number of deviation is 2707 and throughput time is 44.523/minute with the average of 3643 and median of 3677

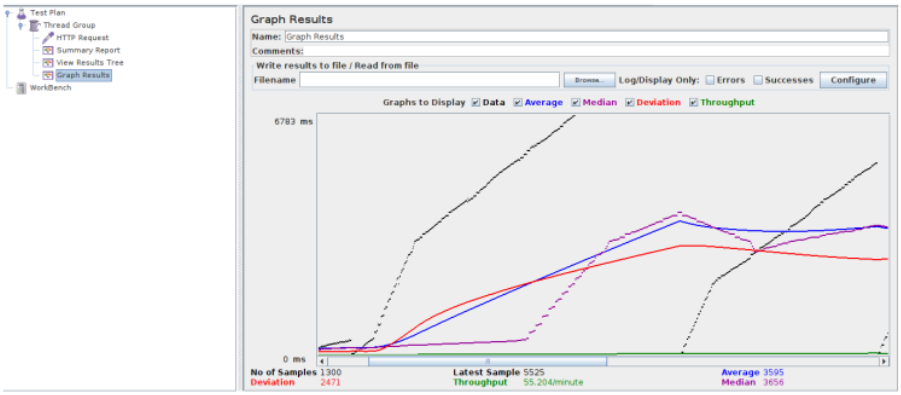


Figure 4. 4: Sign up (1)

We have tested sign in with 1300 samples and received number of deviation is 2471 and throughput time is 55.204/minute with the average of 3595 and median of 3656

CHAPTER 5

USER MANUAL

5.1 User Manual

This is the user manual for making this application easy to use for a layman by guiding each step in it. This application can be used on different operating systems like Linux and windows. A person who knows English language and knows who to use a web-based application can use this application easily without any kind of difficulty. This application is designed for those students who wants to enhance their skills with just a stable internet connection.

5.2 Main Functionality

The basic function of this application is to provide a system for different worldwide instructors and different kinds of students without any kind of classification to polish their skills through this learning system. Instructors manage the courses, and all that content and student can get enrol in it after giving the requirement whereas the admin can manage the courses, users, payment method and responses given students

5.3 Demonstration about the application

There are some screenshots that make it clear and easy to use application

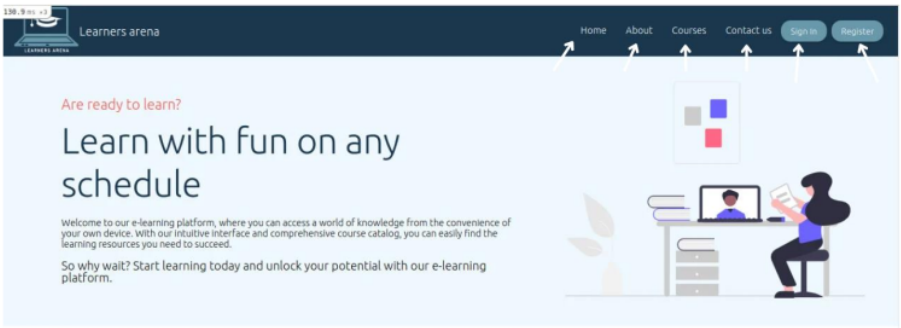


Figure 5. 1: Home Page

1. Click on Home to see the home page
2. Click on the About of Learners Arena
3. Click on Course to see all Courses
4. Click on Contact us for any kind of guidance
5. Click sign in if you already have an account in Learners Arena
6. Click on Register if you are new to Learners Arena

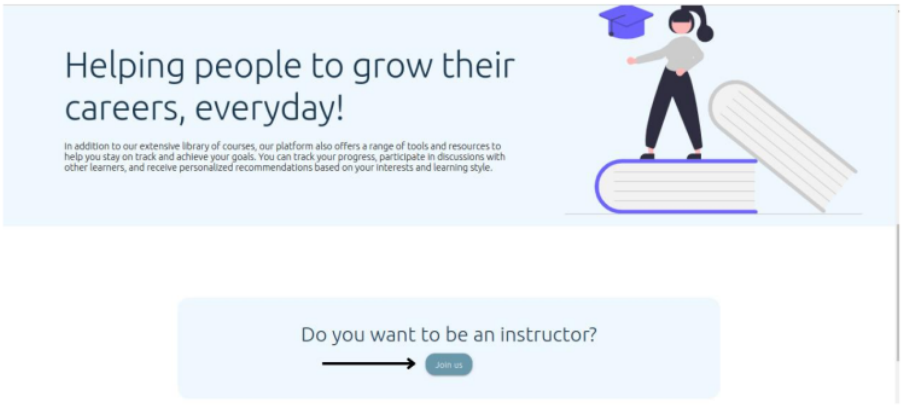


Figure 5. 2: Home Page 1

1. If you want to Join Learners Arena as an instructor click on join us

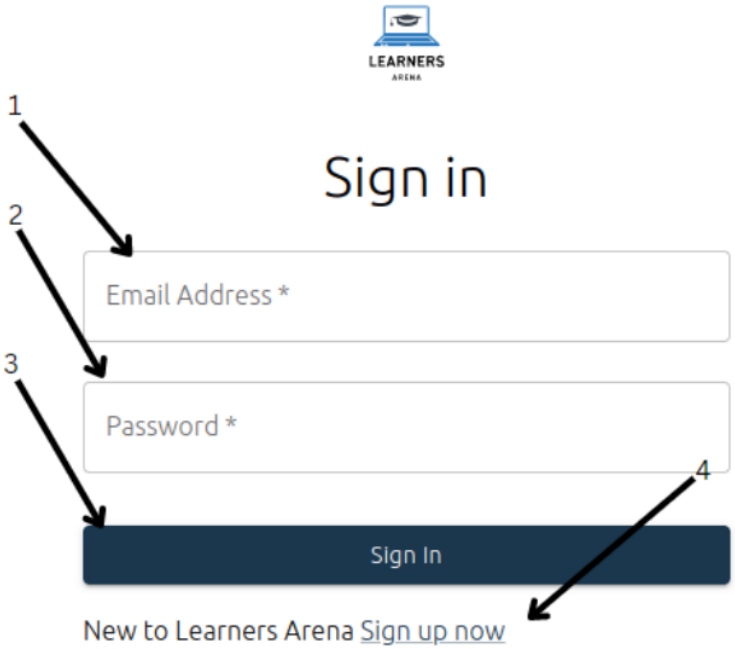


Figure 5. 3: Sign in

1. Enter your email
2. Enter your password
3. Click Sign in to enter in your relevant Panel
4. If you are new at learner's arena, then click on sign up now

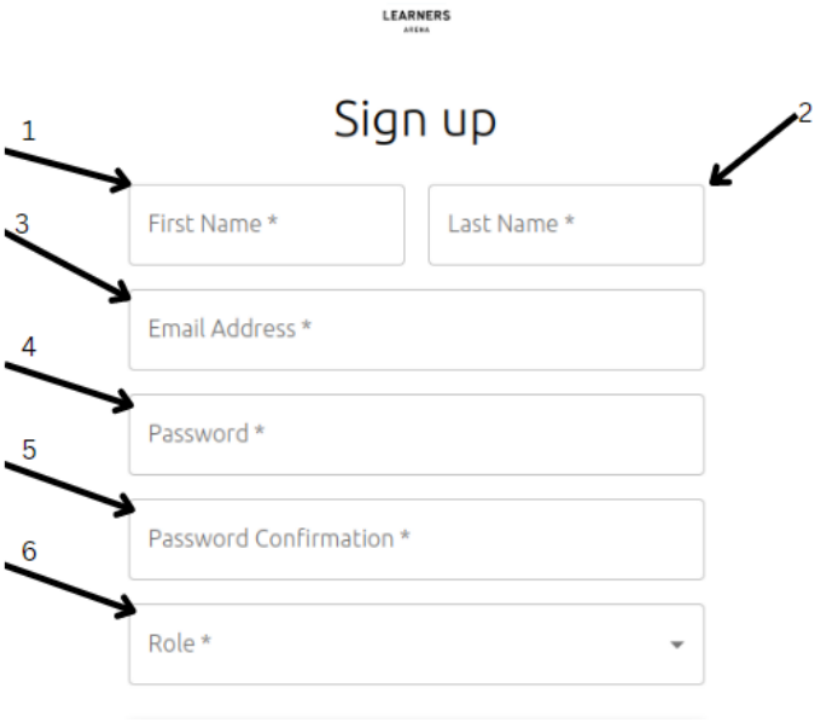


Figure 5. 4: Sign up

1. Enter your first name
2. Enter your second name
3. Enter your email
4. Enter your password
5. Again, confirmation of password
6. Select your role between student and teacher

5.3.1 Admin Panel

Figure 5. 5: Admin Dashboard

1. Total number students in application
2. Total number of teachers in application
3. Total number of courses in application

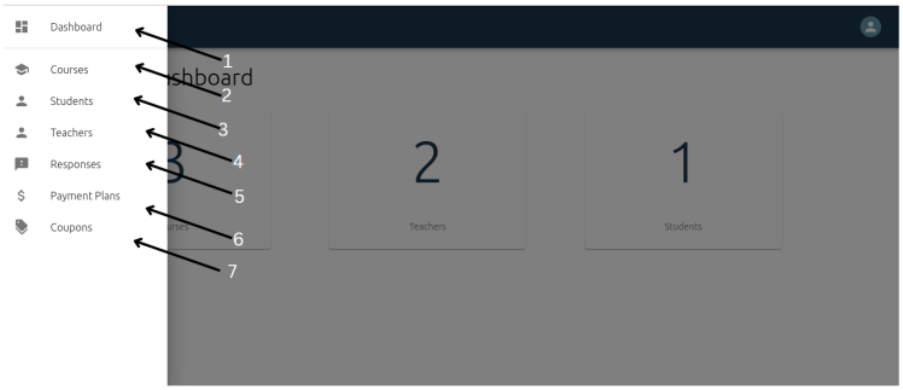


Figure 5. 6: Views of Admin Dashboard

1. Dashboard of Admin panel
2. All courses seen by click on Courses
3. All students seen by click on Students
4. All students seen by click on teachers
5. All Student Responses seen by click on responses

6. All Payments plan seen by click on payment plan
7. All discounted coupons seen by click on coupons

Figure 5.7: All courses

1. Course Id
2. Course Title
3. Course description
4. Status of course
5. Instructor name
6. View Course
7. Deletion of course

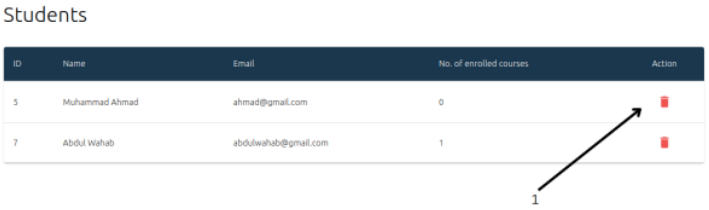


Figure 5.8: All students

1. Deletion of any Student from application
2. Details of registered students

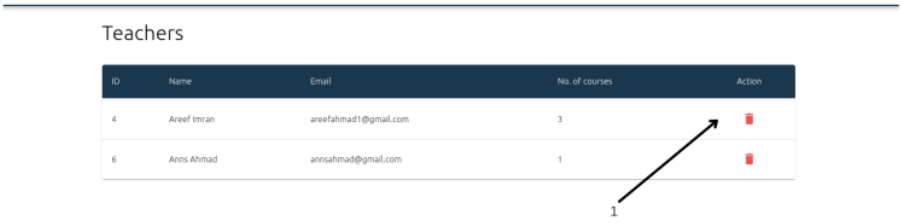


Figure 5. 9: All Teachers

1. Deletion of any teacher anytime
2. Record of registered teachers

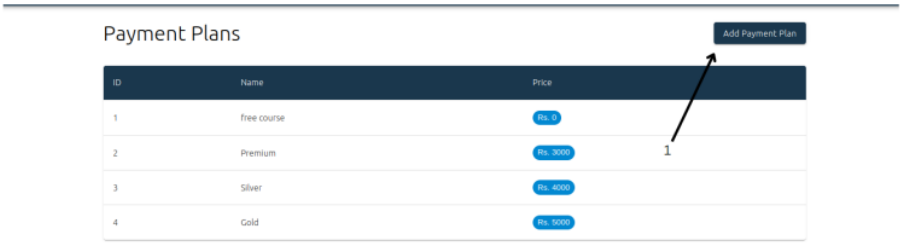


Figure 5. 10: Payment Plan

1. Click on Add Payment plan by clicking on it Payment plan for paid courses added

Figure 5. 11: Discounted Coupons

1. Click on Add coupons to generate discount coupon

5.3.2 Student Panel

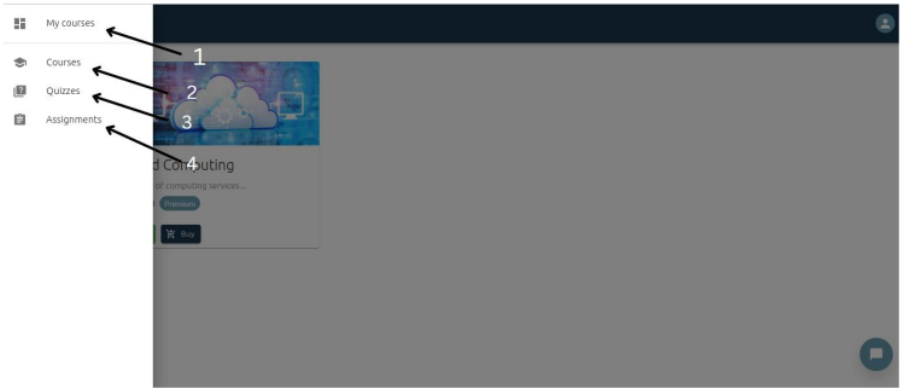


Figure 5. 12: Student Dashboard

1. Click on My courses to see Enrolled Courses
2. Courses show all the offered Courses
3. Quizzes show all the quizzes
4. Assignments show all the assignments

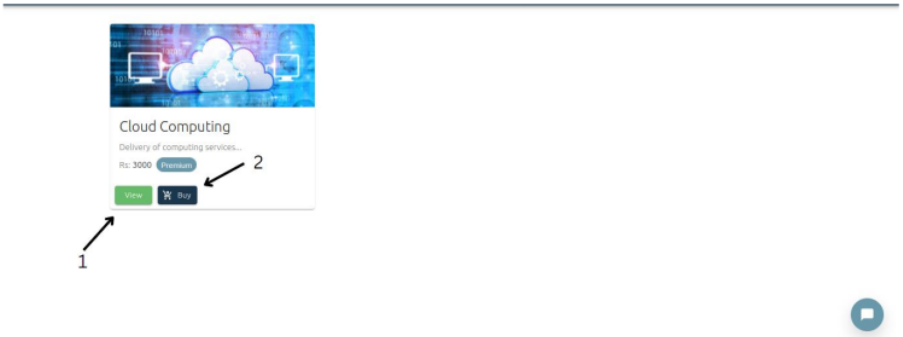


Figure 5. 13: Course Card

1. View Description of Course by clicking view
2. Interested in it then to enrol first buy it

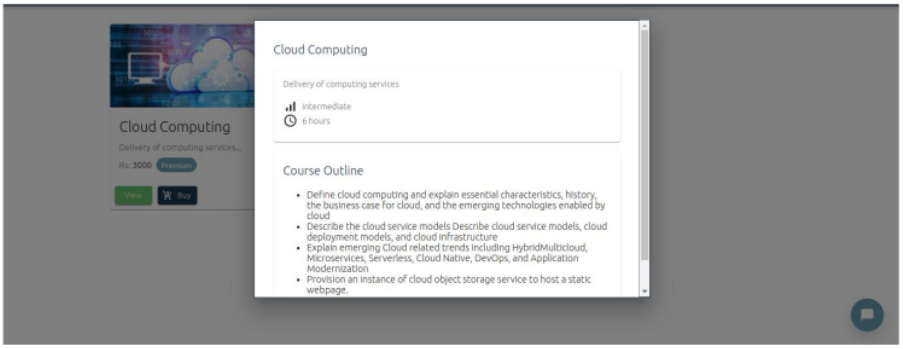


Figure 5. 14: Course Outline

1. After clicking on view this pop up appears

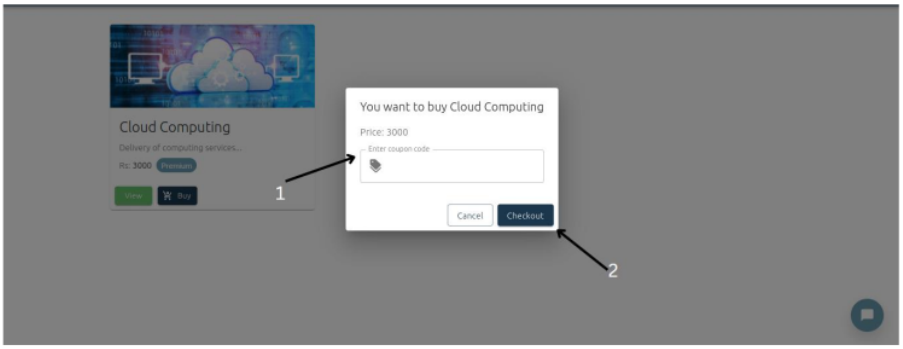


Figure 5. 15: Buying course

1. If having coupon then enter it
2. Click on checkout to proceed your payment

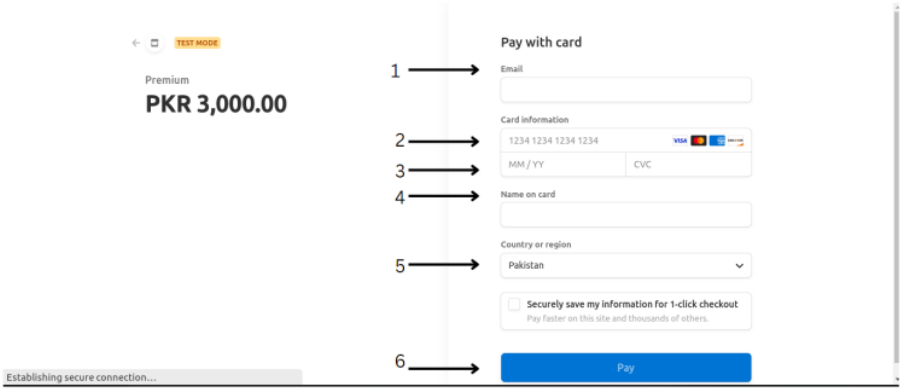


Figure 5. 16: Card Details

1. Enter your email
2. Card Number
3. Date on Card
4. Name on card
5. Select Country
6. Click on pay

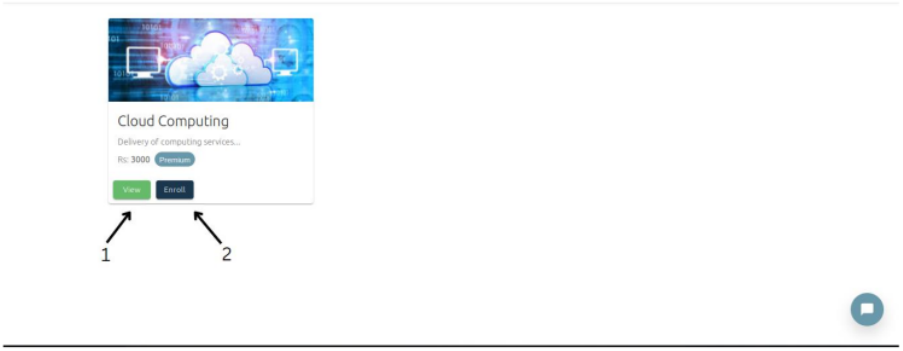


Figure 5. 17: Enrolment in course

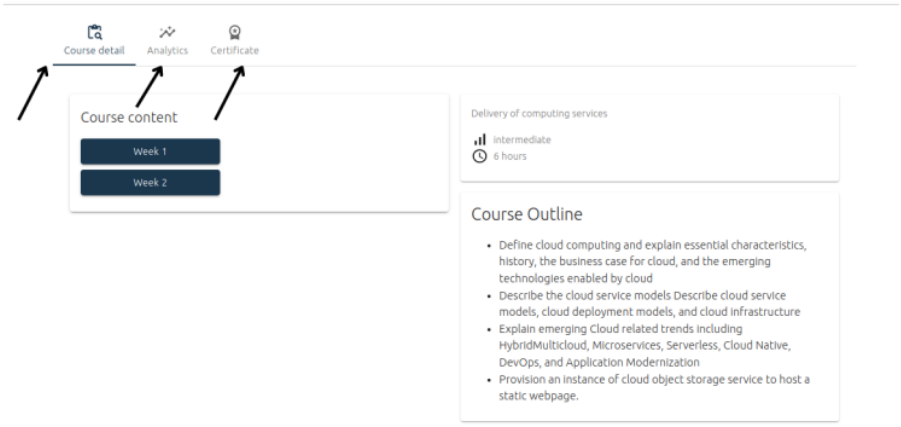


Figure 5. 18: Course Content

1. Click on Course detail to view Course material
2. Click on Analytics to view progress
3. Click on certificate after completion of course to generate e-certificate

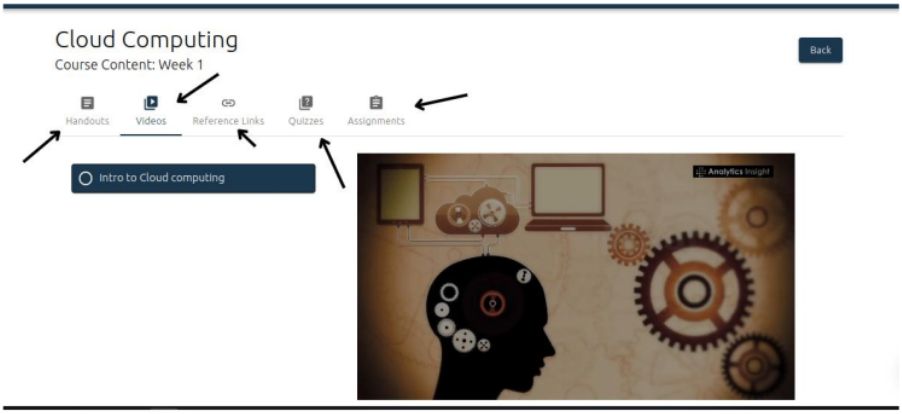


Figure 5. 19: Course Content View

1. Click on hand-outs for viewing hand-outs
2. Click on videos for watching videos
3. Click on reference links view reference links
4. Click on quizzes to view the quizzes to courses
5. Click on assignments for attempting assignments

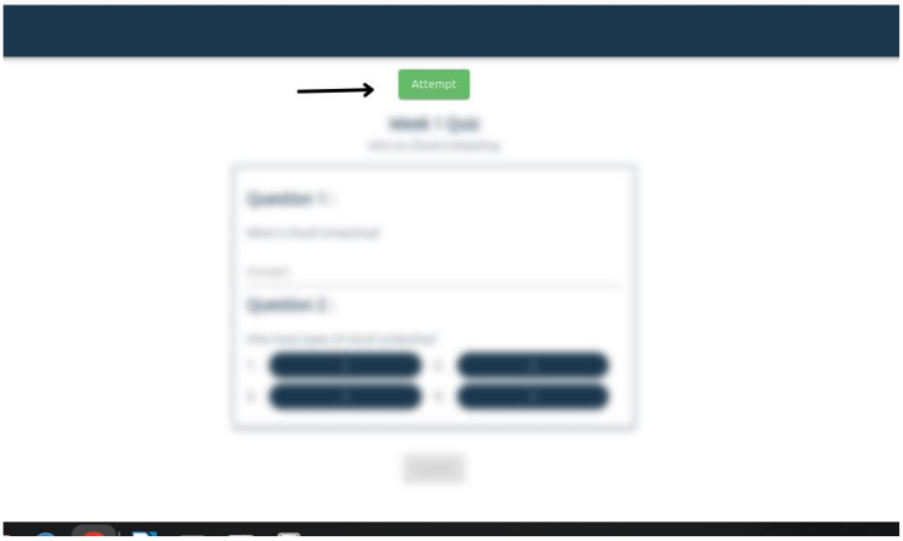


Figure 5. 20: Quiz Attempt

1. Click on Attempt to attempt quiz

Figure 5. 21: Certificate Generation

1. Click on generate certificate to generate certificate after completion of course

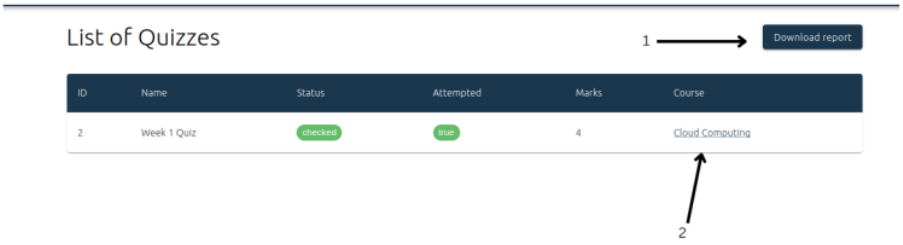


Figure 5. 22: List of Quizzes

1. Download the Quiz progress report
2. View the course

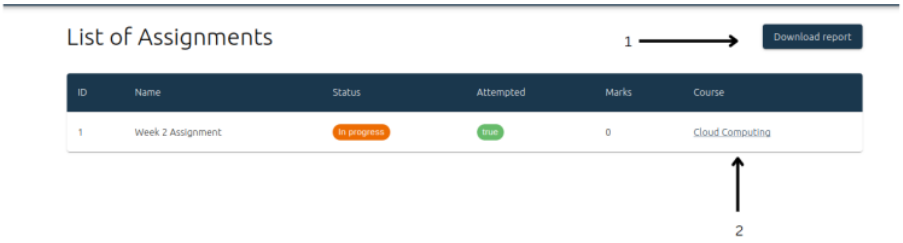


Figure 5. 23: List of Assignments

1. Click on download report to download the progress report of assignments
2. Click on course to view it

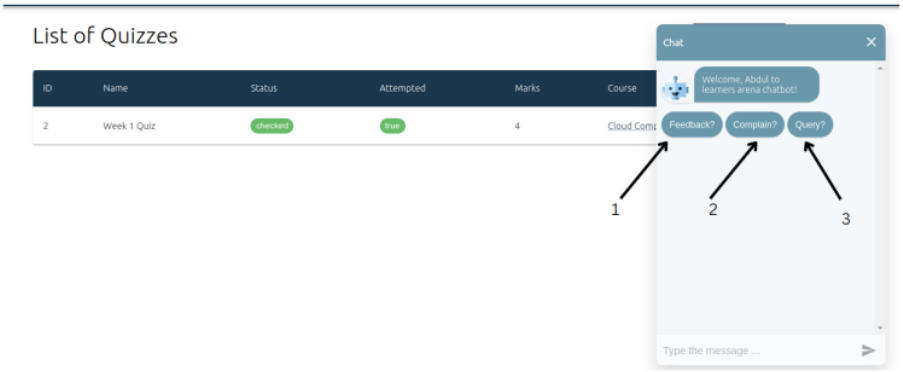


Figure 5. 24: Student Feedback

1. Click on feedback for our feedback
2. Click on complain for your complain
3. Click on query if you have any query

5.3.3 Teacher Panel

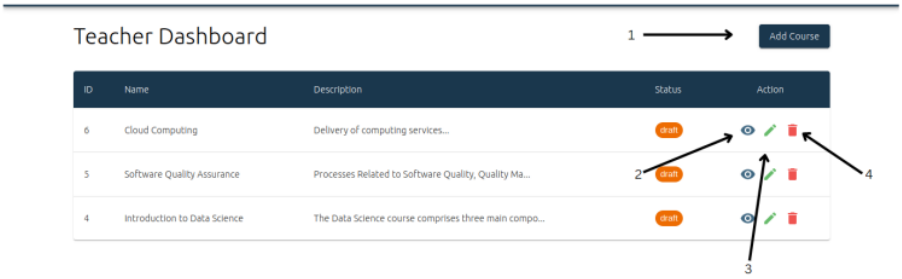


Figure 5. 25: Teacher Dashboard

1. Click on Add course to add a new course in
2. Click on view to view course
3. Click on pen icon to edit the course
4. Click on delete icon to delete the course

Figure 5. 26: Course Card

1. Number of enrolled students
2. Difficulty level
3. Total hours
4. Total teacher income by a course
5. Enrolled students' data
6. Course outline

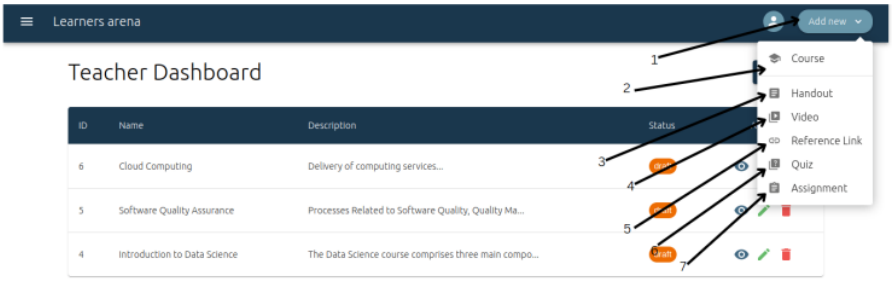


Figure 5. 27: Add Content

1. Click on add new to add new thing in teacher panel
2. Click in course to add new course
3. Click on hand out to add new hand out in any course
4. Click on video to add a new video in any course

5. Click on Reference links to add a new reference link in any course
6. Click on quiz to add a new quiz in any course
7. Click on assignment to add a new assignment in any course

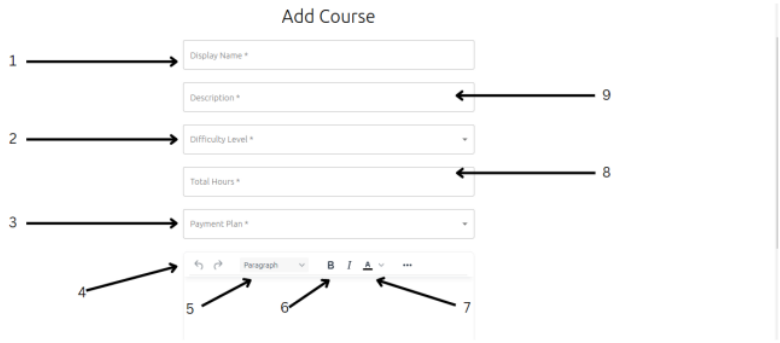


Figure 5. 28: Add course

1. Title of course
2. Difficulty level of course
3. Course paid or free
4. Editor text formatting
5. Size of text
6. Bold Text
7. Underline text
8. Total number of hours of course
9. Description of course

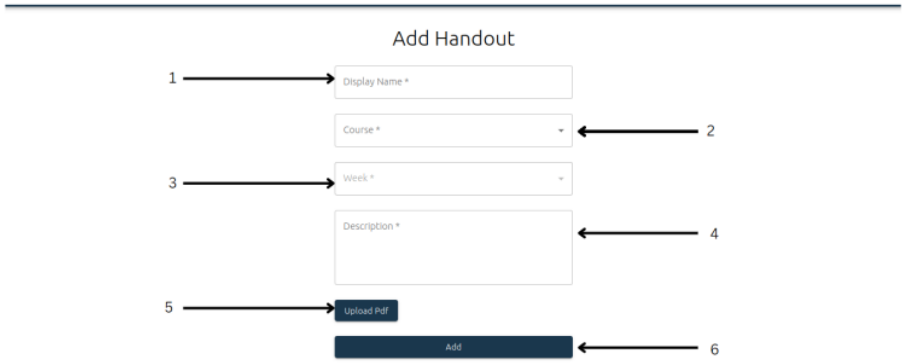


Figure 5. 29: Add hand out

1. Add hand out name
2. Select course
3. Select week
4. Description of hand out
5. Upload file
6. Click on add to add a handout

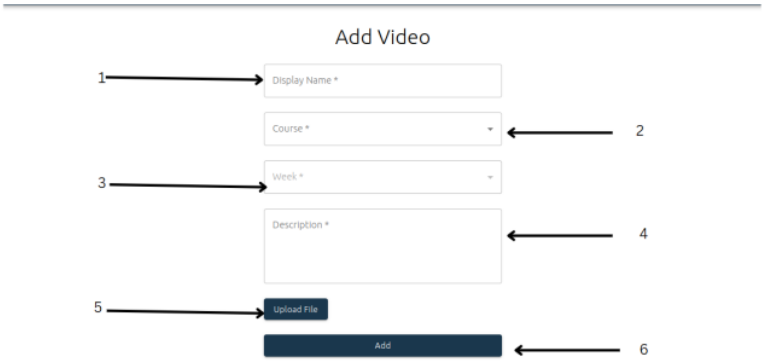


Figure 5. 30: Add video

1. Add display name
2. Select the course
3. Select the week
4. Add description about it

5. Upload video
6. Click on Add to add it

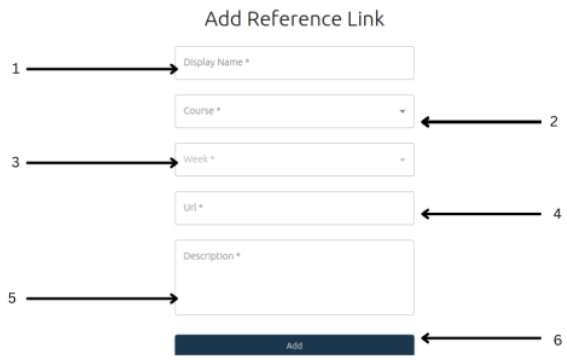


Figure 5. 31: Add Reference Link

1. Add a display name
2. Select course
3. Select week
4. Add URL
5. Add description of it
6. Click on add to add it

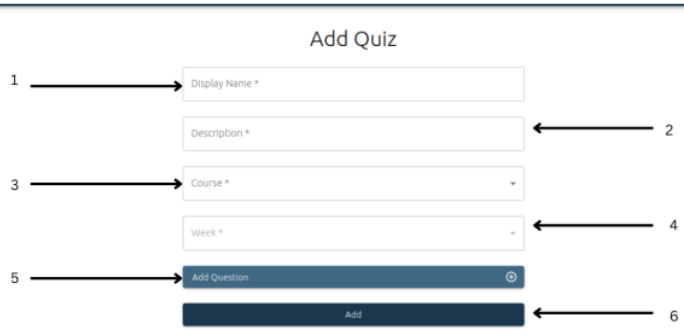


Figure 5. 32: Add Quiz

1. Add a display name
2. Description about it

3. Select the course
4. Select the week
5. Adding the question of both multiple choice or textual
6. Click on add to add it

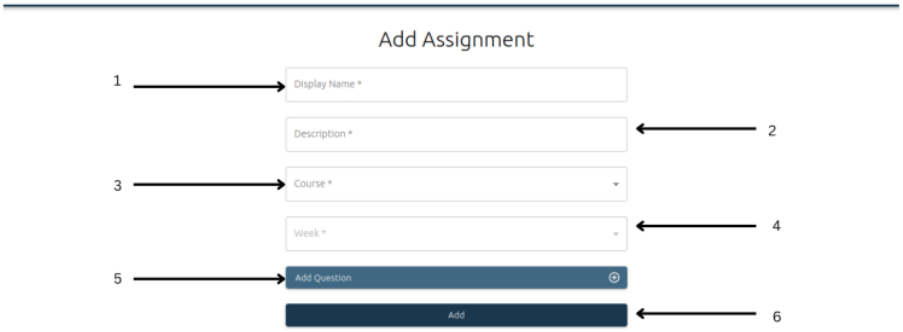


Figure 5. 33: Add Assignment

1. Add a display name
2. Description about it
3. Select the course
4. Select the week
5. Adding the question of both coding or text base
6. Click on add to add it

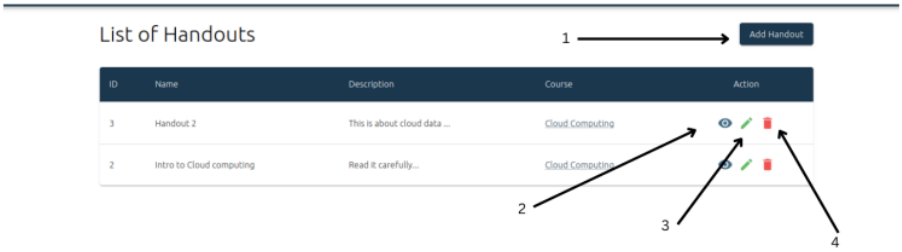


Figure 5. 34: List of hand outs

1. Click on add hand out to add a new hand out
2. Click on view to view it
3. Click in edit to edit it
4. Click on delete to delete it

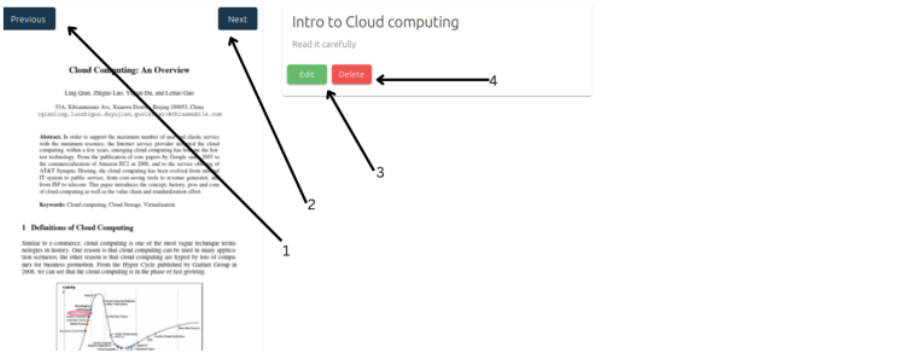


Figure 5. 35: Hand out View

1. Click on previous to see previous page of hand out
2. Click on next to see next page
3. Click on edit to edit it
4. Click on delete to delete it

Figure 5. 36: Edit Hand out

1. Change the display name
2. Change the description
3. Add new file
4. Click on update to update it

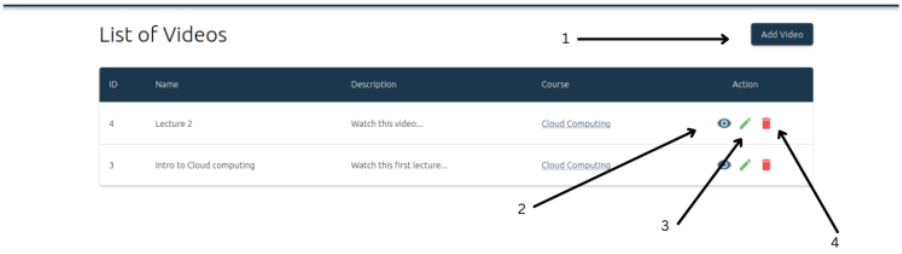


Figure 5. 37: List of videos

1. Add a video by click on add video
2. Click on view to view it
3. Click on edit to edit it
4. Click on delete to delete it

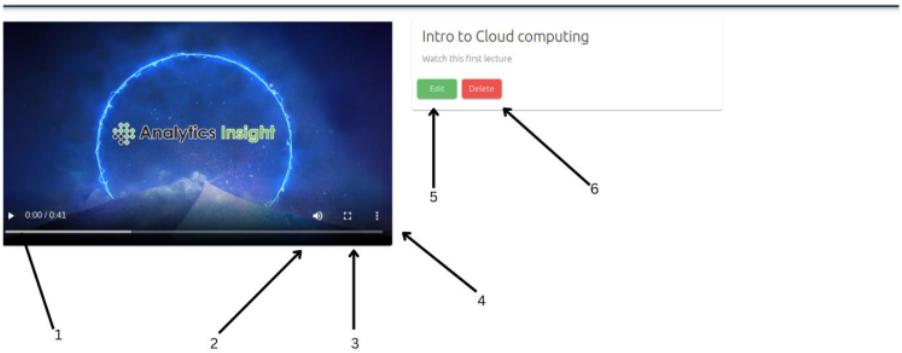


Figure 5. 38: Video view

1. Start the video
2. Volume off
3. Full screen mood
4. Download it
5. Edit it
6. Delete it

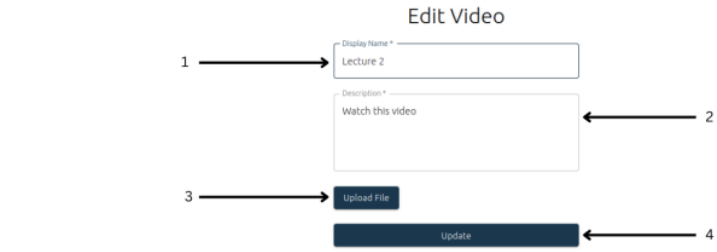


Figure 5. 39: Edit Video

1. Change the display name
2. Change the description of video
3. Upload new file
4. Click on Update to update it

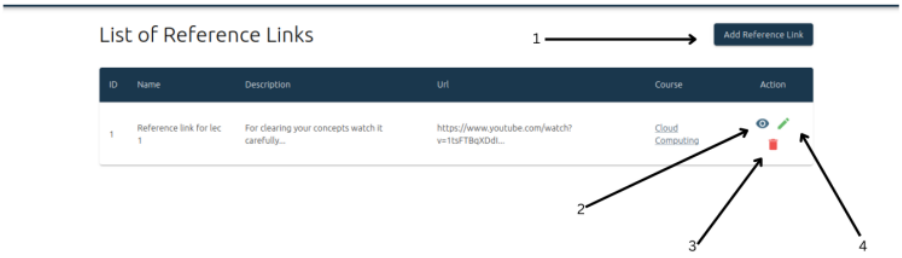


Figure 5. 40: List of Reference Links

1. Click on reference link to add a new link
2. Click on view to view it
3. Click on delete to delete it
4. Click on pen to edit it

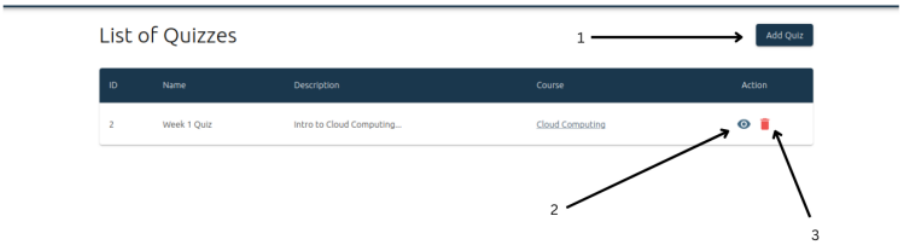


Figure 5. 41: List of Quizzes

1. Click on Add quiz to add
2. Click on view to view it
3. Click on delete for deletion of quiz

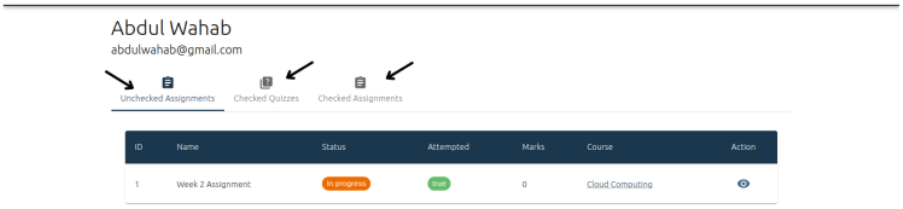


Figure 5. 42: Student Progress

1. Click on Unchecked Assignments to see the assignments that are not checked
2. Click on checked quizzes to see the result of quizzes
3. Click on checked assignments to see result of assignments

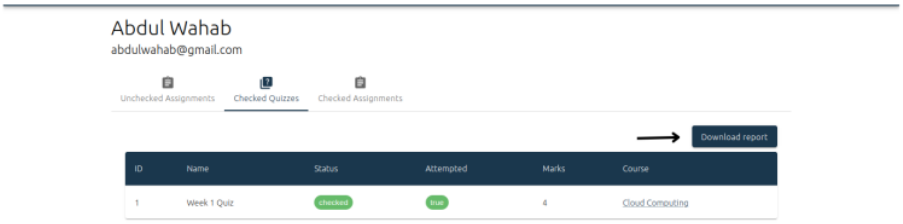


Figure 5. 43: Student Report

1. Download the report by clicking Download report

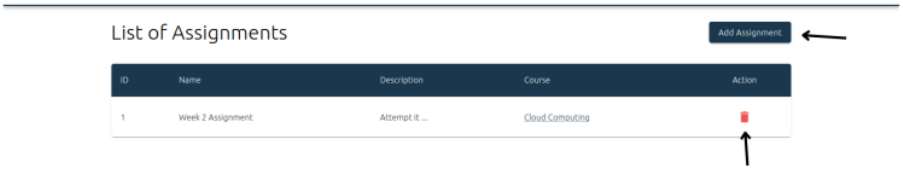


Figure 5. 44: List of Assignments

1. Add Assignment by clicking on add assignment
2. Click on delete to delete it

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

In conclusion, the project has successfully addressed the problem of online learning by successfully developing a user-friendly and efficient platform for digital evangelists, technopreneurs, entrepreneurs, tech enthusiasts, researchers, job holders, and more. The key features of our solution include online ready-to-watch lectures, learning materials, an assessment platform to evaluate yourself, prediction of future grades, recommendation of newer courses, and finally, the certificate of completion and the report of grades. Through thorough research, design, and development, we have created user-friendly and effective software that meets the needs of our target users. The system has been tested extensively and has demonstrated its ability to improve user satisfaction. The system has demonstrated its capabilities in meeting the project objectives and ensuring compliance with industry standards. We believe that our solution will greatly benefit learners of any kind, mentors which want to share their knowledge and experience and contribute to the advancement of learning in the 4th generation of the industrial revolution.

6.2 Recommendation

As the software continues to evolve and improve, there are several recommendations for future updates that can enhance the user experience and expand the capabilities of the Learners Arena. Some potential recommendations include adding new features such as implementing cyber security algorithms to protect user data of teachers and students to prevent external unauthorized access and applying AI algorithms to improve user data analysis of registered users of Learners Arena. Additionally, ongoing maintenance and support are essential to ensure that the software continues to operate smoothly and effectively. By implementing these updates and enhancements, the software can continue to provide value and support the needs of its users.

REFERENCES

Journal Papers:

[1] Anaraki, Firouz. "Developing an effective and efficient eLearning platform." *International Journal of the computer, the internet and management* 12.2 (2004): 57-63.

[2] Oproiu, Gabriela Carmen. "A study about using e-learning platform (Moodle) in university teaching process." *Procedia-Social and Behavioral Sciences* 180 (2015): 426-432.

[3] Violante, Maria Grazia, and Enrico Vezzetti. "Implementing a new approach for the design of an e-learning platform in engineering education." *Computer Applications in Engineering Education* 22.4 (2014): 708-727.

APPENDICES

JavaScript: JavaScript is a high-level, interpreted programming language. It is widely used for building web and mobile applications, and it is one of the core technologies of the World Wide Web.

ReactJS: React is a JavaScript library for building user interfaces.

Ruby: Ruby is a general-purpose, interpreted programming language.

Ruby on Rails: Ruby on Rails, or Rails, is a web application framework written in the Ruby programming language.

Redux: Redux is a predictable state container for JavaScript apps.

Material UI: Material-UI is a popular React-based UI framework for building responsive, accessible, and consistent user interfaces.

PostgreSQL: PostgreSQL is an open-source object-relational database management system (ORDBMS).

Stripe API: Stripe is a suite of payment APIs that allows developers to easily integrate online payment processing into their applications.

Gmail API: The Gmail API is a RESTful API that allows developers to access Gmail mailbox data in order to integrate Gmail features into their own applications.

Use Case Diagram:

A use case diagram is a visual representation of the interactions between a system and its users, known as actors.

Sequence Diagram:

A sequence diagram is a type of interaction diagram that shows the interactions between objects or entities in a system in the form of a sequence of messages.

Collaboration Diagram:

A collaboration diagram is a type of interaction diagram that shows the interactions between objects or entities in a system in the form of a diagram.

Entity Relationship Diagram:

An entity relationship diagram (ERD) is a visual representation of the relationships between entities in a database.

Class Diagram:

A class diagram is a type of static structure diagram that represents the structure of a system by showing the system's classes, their attributes, and the relationships between the classes.

Domain Model Diagram:

A domain model diagram is a visual representation of the classes, attributes, and relationships in a system.

Feature Driven Development:

Feature-driven development (FDD) is a software development methodology that focuses on delivering small, incremental pieces of functionality (called "features") to the customer on a regular basis.

Final Documents FYP

ORIGINALITY REPORT

4%	3%	0%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	2%
2	Submitted to Informatics Education Limited Student Paper	1%
3	www.techrepublic.com Internet Source	1%
4	Submitted to Monash University Student Paper	1%
5	hellotech633.wordpress.com Internet Source	1%

Exclude quotes	On	Exclude matches	< 1%
Exclude bibliography	On		