



Graduation Project, Part-I (SWE 496)
Software Engineering Department
CCIS, KSU

Project Advisor:
Zohair Chentouf

Submitted by

Mohammed Al-Tamimi, 437100922
Abdulrahman Aldakkan, 437100413
Yousef Alkahtani, 435101137
Faisal alhumayyd 437102787

Date submitted

ABSTRACT

This report describes the devolvment of our project which is an educational website aimed for those who're looking to improve their skills in a certain subject by taking courses in that specific field, And for any person that want to conduct a course for free or for a certain amount of fees.

Table of Contents

1. INTRODUCTION	1
2. DOMAIN ANALYSIS	2
3. RISK/CONSTRAINTS.....	2
4. PROJECT PLAN.....	6
5. QUALITY ASSURANCE PLAN	7
6. REQUIREMENTS	8
6.1 FUNCTIONAL REQUIREMENTS.....	8
6.2 NON-FUNCTIONAL REQUIREMENTS.....	9
7. PROBLEM COMPLEXITY.....	9
8. SYSTEM USE-CASES	10
8.1 USE CASE1	ERROR! BOOKMARK NOT DEFINED.
8.2 USE CASE2	ERROR! BOOKMARK NOT DEFINED.
9. ANALYSIS CLASS	26
10. INTERACTION DIAGRAM.....	27
11. DESIGN CLASS	29
12. SYSTEM ARCHITECTURE	30
13. USER INTERFACE MOCKUP	31
14. DATABASE SCHEMA	35
15. ALGORITHMS	35
16. EXPECTED DEPLOYMENT	36
17. TEST SCENARIO	36
18. PROJECT STATUS	39
19. CONCLUSION	39
20. REFERENCE.....	39

List of Figures

Figure 1. use cases	Error! Bookmark not defined.
Figure 2. Class Diagram	Error! Bookmark not defined.
Figure 3. System architecture	Error! Bookmark not defined.
Figure 4. System deployment diagram	36

List of Tables

TABLE 1. FEATURES COMPARESON	4
TABLE 2. RISK/CONSTRAINS	4

1. Introduction

Online education in the current situation has become a place of interest due to COVID 19 pandemic, The COVID-19 has resulted in schools shut all across the world. Globally, over 1.2 billion children are out of the classroom.

As a result, education has changed dramatically, with the distinctive rise of e-learning, whereby teaching is undertaken remotely and on digital platforms.

Research suggests that online learning has been shown to increase retention of information, and take less time, meaning the changes coronavirus have caused might be here to stay.

In this project we will implement a platform that combine both graduate and undergraduate students and give them the ability to improve their knowledge in certain topics of their choosing.

Even more for teachers who wants to lecture a course online paid or free hopefully our project will help them do the deed.

1.Domain Analysis

1.1.WEA



Description

WEA is a charity website focuses on adult education for social purpose to increase social knowledge In the United Kingdom, the goal of the website is aspirations and develop educational opportunities for the most disadvantaged. This includes providing basic math, English, and IT skills for employment.

1.2. Udemy



Udemy, Inc. is an American massive open online course provider aimed at professional adults and students. the platform has more than 35 million students and 57,000 instructors teaching courses in over 65 languages. It supports wide range of courses.

1.3.Coursera



Coursera is an American massive open online course provider, offers massive open online courses, specializations, degrees, professional and master track courses.

Coursera works with universities and other organizations to offer online courses, certifications, and degrees in a variety of subjects, such as engineering, data science, machine learning, mathematics, business, financing, computer science, digital marketing, humanities, medicine, biology, social sciences, 3000 plus variety of courses giving students a very broad range of information & experience in different fields.


















1.4.Shroo7



Is an educational platform that works to create an interactive learning environment by offering Special Services that help student to improving their knowledge and students who have difficulty studying with qualified and trusted teachers and provide online courses.

1.5. Summary

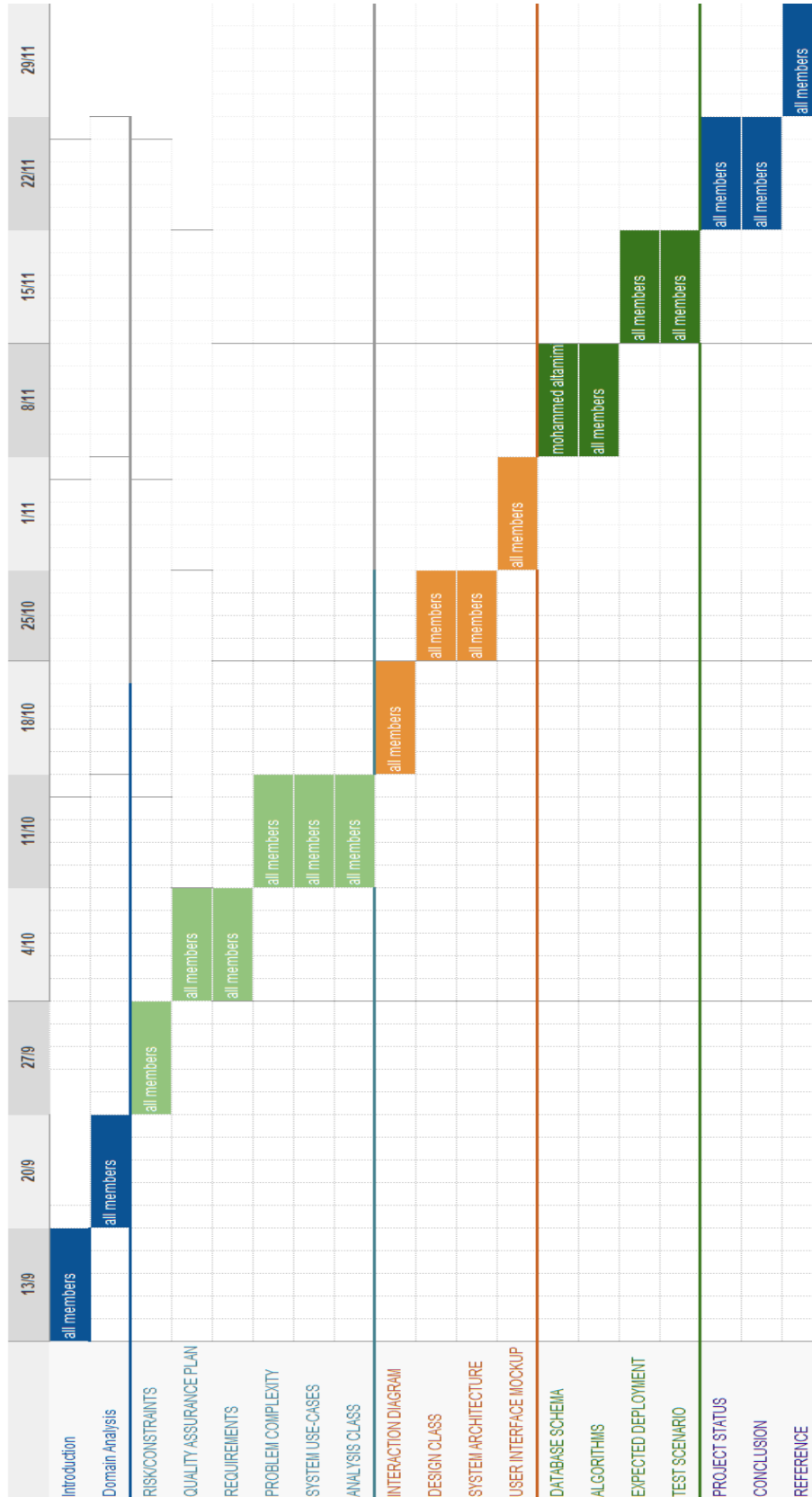
Table 1 is a brief comparison of the main features of our website and the website listed above.

Features	EIM++	WEA	Shroo7	Coursera	Udemy
live lecturing					
course modification					
course rating					
offline mode					
automate exam grading					
Provide course material.					
Print course certificate					

2. Risk/Constraints

No	RISK	Type	Severity	Likelihood	Risk Management Strategy
1	Limited knowledge of required technologies.	Product	Medium	Low	Take courses that teach these technologies.
3	Lack of courses at the beginning.	Product	Medium	High	Provide special offers for lecturer.
4	Sharing video for unauthorized access.	Business	Medium	Medium	Blue mark on the video

3. Project Plan



4. Quality Assurance Plan

- **Reviews:**

- Software Requirement Specification Review in requirement phase.
- Design review during various stages of the design process and at the end of the design phase.
- Software Test Procedure Review in testing phase.
- Implementation and code review once developers start writing the code, at the end of the implementation phase and at the end of the project.
- Maintenance document review to make sure the maintenance team successfully fix the problems that was occurred

- **Testing:**

The system will go through different kinds of testing such as: reliability testing, integrity testing and deployment testing.

- **Verification:**

Can be assured the developers built the specified design when the actual design output is the same as the expected design output that meets the product specification. It can be assured if developers followed standards by creating specific guidelines, developers training, and using templates and checklists.

- **Walkthroughs:**

Walkthrough is a form of software peer review. Walkthroughs are considered as informal way of reviewing each artifact of the software development lifecycle. Every team member will go through the artifacts of the project individually to gather enhancement point for the final system.

- **Training Team members:**

There will be training to each of the group member to increase the overall skills in web development.

5. Requirements

5.1 Functional Requirements

Administrator-related requirements:

1. Admin shall be able to edit his account password.
2. Admin shall be able to view his account information (Username, email).
3. Admin shall be able to delete lecturer account.
4. Admin shall be able to approve/deny lecturer account.
5. Admin shall be able view Students username and email.
6. Admin shall be able view lecturer username, email, national Id, name, phone.
7. Admin shall be able to login using his username and password.
8. Admin shall be able to view the following statistics:
 - Number of users.
 - Number of registered courses in each month.
9. Admin shall be able to view courses.
10. Admin shall be able to delete courses.
11. Admin shall be able to review course materials.

Student-related requirements:

1. Student shall be able to edit his account information except username.
2. Student shall be able to view his account information (Username, email).
3. Student shall be able to login using his username and password.
4. Student shall be able to Sign-up using his (Email, Username, password,).
5. Student shall be able to search for a course.
6. Student shall be able to subscribe in a course.
7. Student shall be able to unsubscribe from a course.
8. Student shall be able to report a course.
9. Student shall be able to rate a course.
10. Student shall be able to view course rating.
11. Student shall be able to download the course.
12. Student shall be able to take course exam.
13. Student shall be able to view exam results.
14. Student shall be able to print course certificate.
15. Student shall be able to choose his payment method for paid courses.

lecturer -related requirements:

1. lecturer shall be able to edit his account information except username.
2. lecturer shall be able to view his account information (Username, email, phone).
3. lecturer shall be able to delete his account.
4. lecturer shall be able to login using his username and password.
5. lecturer shall be able to Sign-up using his (Email, Username, password,).
6. lecturer shall be able to delete course.
7. lecturer shall be able to add course.

8. lecturer shall be able to edit the name of the course.
9. lecturer shall be able to delete course materials.
10. lecturer shall be able to add course materials.!!!
11. lecturer shall be able to edit the name of the course material.
12. lecturer shall be able to upload course materials.
13. lecturer shall be able to conduct an exam.
14. lecturer shall be able to view the following statistics:
 - Course rating
 - Number of participants per month.

Visitor-related requirements:

1. Visitor shall be able to Sign-up as lecturer or student.
2. Visitor shall be able to view search course.
3. Visitor shall be able to view top rated courses in given category.

5.2 Non-Functional Requirements

Supportability:

1. The system shall support Arabic and English languages.

Usability:

1. The user shall be able to learn the interface in no longer than 30 minutes.

Security:

1. The system shall store user's personal and payment info in an encrypted form.

6. Problem Complexity

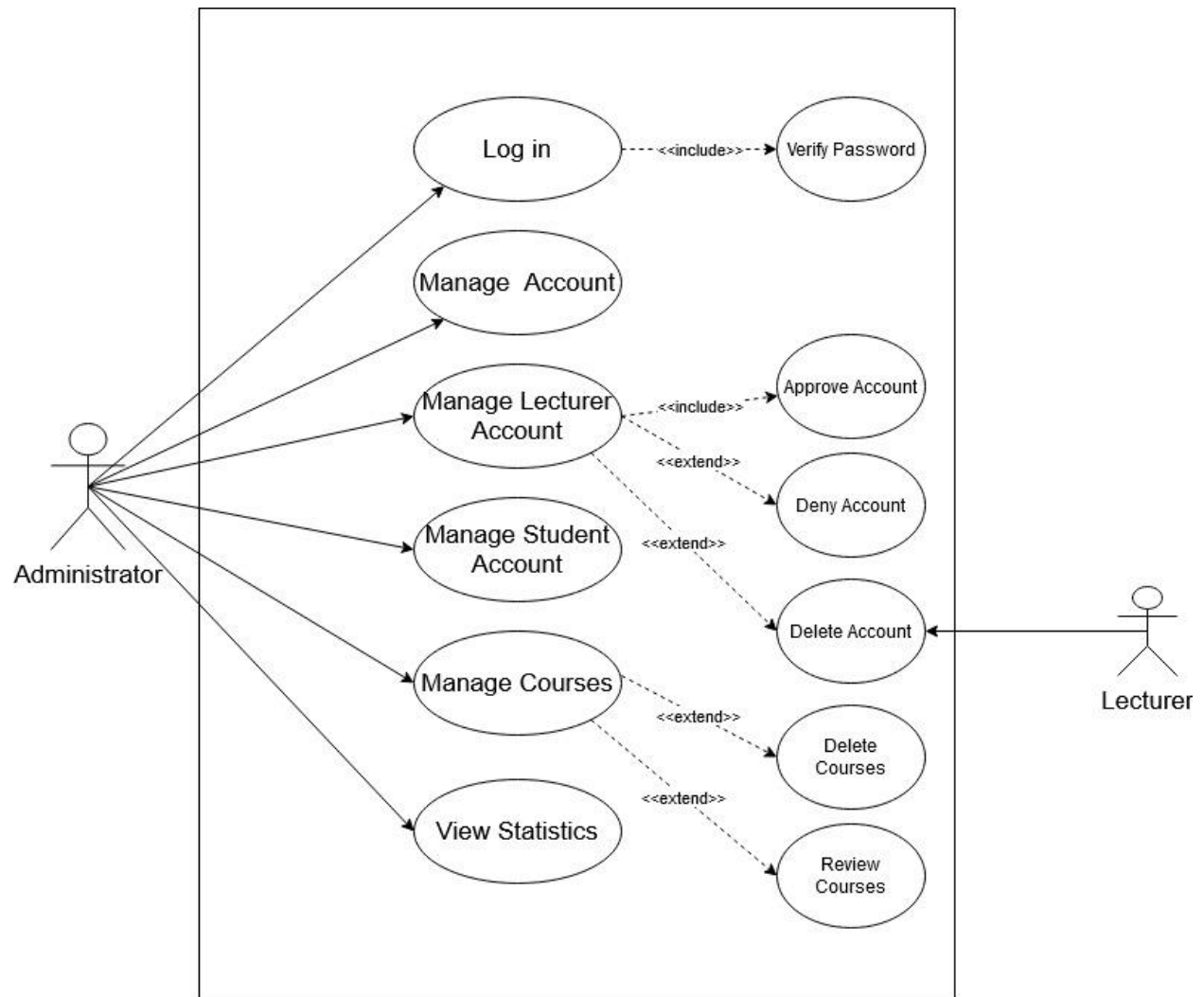
-Algorithmic complexity we might encounter this in some of the features as it may take long time to be developed.

-Development complexity as the project grows the processes of the development may become slower due to some Algorithmic problems.

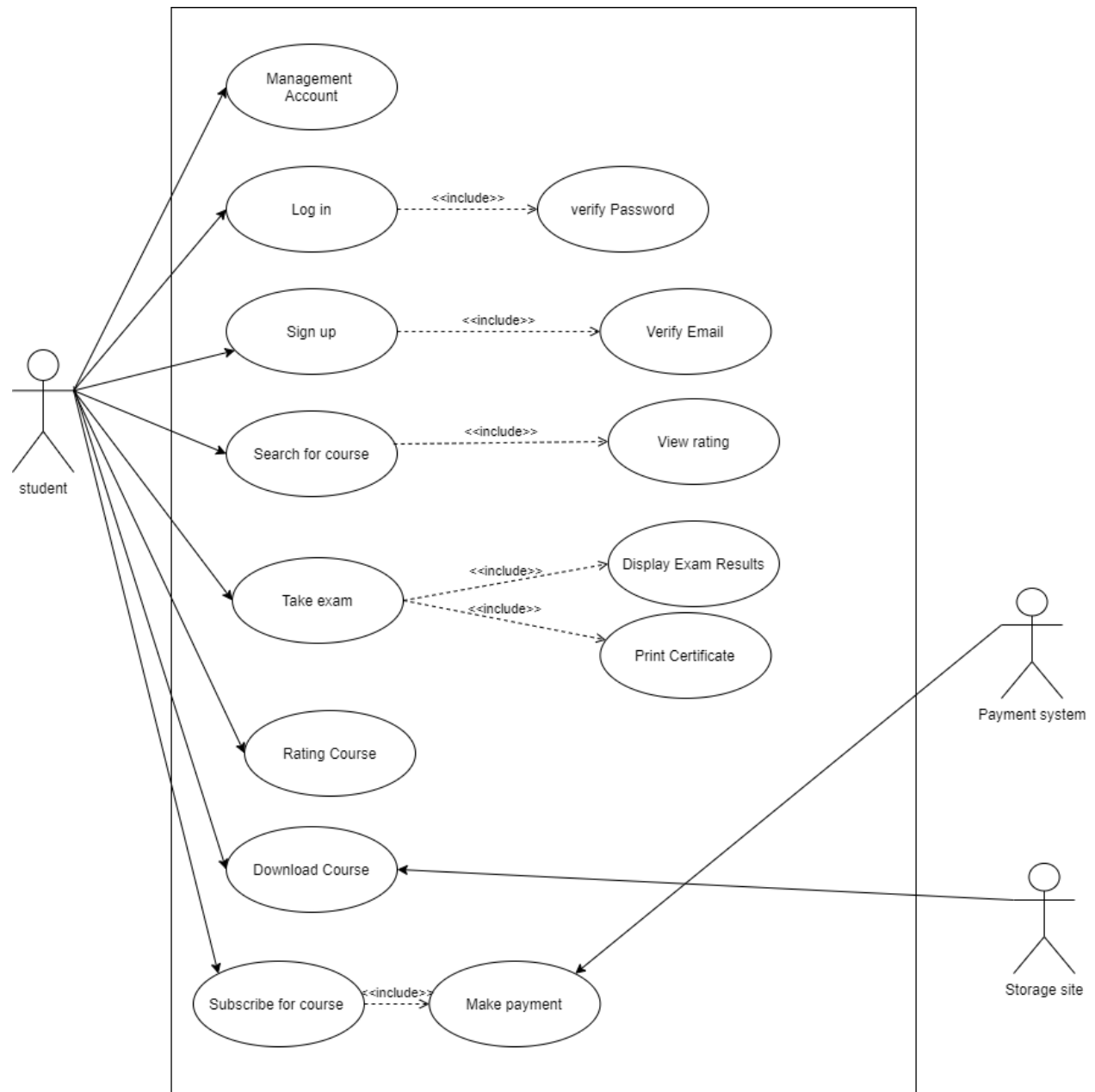
- Information complexity the number of characters or the size of the code could be a problem.

7. System Use-Cases

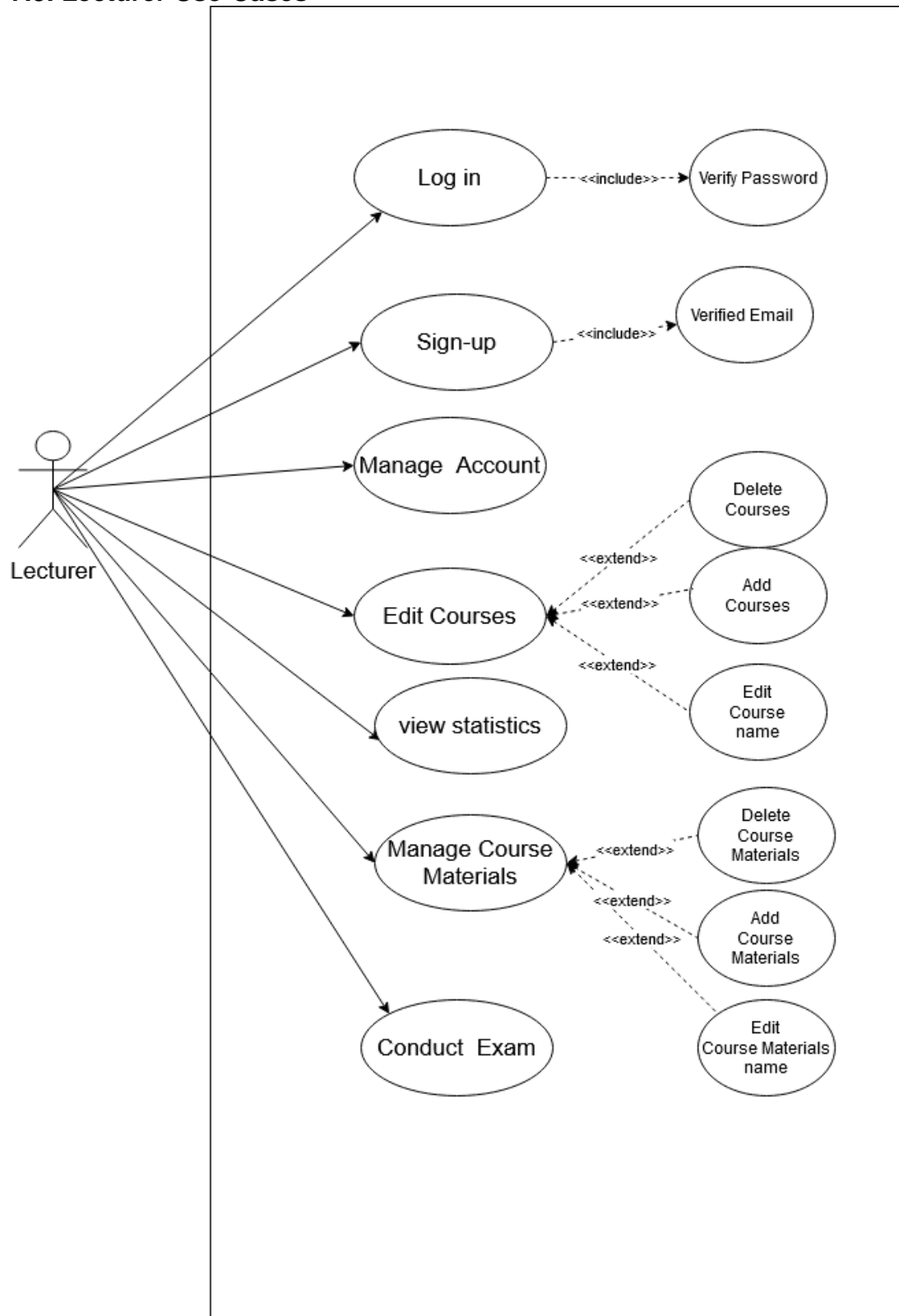
Administrator Use-Cases



7.1 Student Use-Cases



7.3. Lecturer Use-Cases



7.4 Use case description

Use Case Description	
Use Case Name: Log in	
Actor's: lecturer, student, Administrator	
Description: let the actor get access into his account	
Relationship's: Include: Verify Password properties Extends: None	Covered requirements: (R1.2),(R2.3),(R3.4)
Pre-condition: the actor is not log in	
Basic Flow	
Actor: 2. Enter his username and Password 3.Choose the type of account 4. select login	System: 1. Display login Page 5. Validate the username and password 6. login successfully. 7. Display home page.
Alternative Flow: 4.1 the username and Password do not match the record 4.2 continue with step 1	

Use Case Description	
Use Case Name: Sign Up	
Actor's: lecturer, student	
Description: The actor creates an account in the system	
Relationship's: Include: none properties Extends: None	Covered requirements: (R2.3) , (R3.5)
Pre-condition: the actor is not log in	
Basic Flow	
Actor: 2. Select Sign Up. 3. Enter Username, Email, Password, Re-Password. 4.Select Create.	System: 1. Display home page. 5. Check if the username available and all data correct. 6. Create the account.
Alternative Flow: 6.1. If username not available or data do not correct. 6.2. The system clears the rejected field. 6.3. The user enters the rejected field again. 6.4. Continue with Step 1.	

Use Case Description	
Use Case Name: Rating course	
Actor's: student	
Description: the student rate curses from 1 to 5	
Relationship's: Include: none properties Extends: None	Covered requirements: (R2.9)
Pre-condition: none	
Basic Flow	
Actor: 1. Select the course. 3. Select the rating icon 5. choose the rate 6. select the rate	System: 2. Go to the course page. 4. Display rating page
Alternative Flow: none	

Use Case Description	
Use Case Name: Manage account	
Actor's: Administrator, Student, Lecturer	
Description: Let the actor view email, username and edit his email	
Relationship's: Include: none properties Extends: None	Covered requirements: (R1.1) (R2.1) (R3.1) (R1.2) (R2.2) (R3.2)
Pre-condition: The user should be already signed in	
Basic Flow	
Actor: 1. Click the profile icon	System: 2. Opening profile web page
Alternative Flow: none	

Use Case Description	
Use Case Name: Manage Lecturer Account	
Actor's: Administrator Secondary actor: lecturer	
Description: Let the Administrator approve and deny and delete lecturer account and view accounts information.	
Relationship's: Include: none properties Extends: 1. Approve account 2. Deny account 3. Delete account	Covered requirements: (R1.4) (R1.3) (R1.6)
Pre-condition: The actor should be already signed in	
Basic Flow	
Actor: 1. Click the lecturer management icon 4. choose the lecturer 6. save the profile	System: 2. Opening lecturer management web page. 3. List lectures accounts 5. Open the lecturer profile.
Alternative Flow: 6.1.1 Select delete icon 6.1.2 go back to step 4	

Use Case Description	
Use Case Name: Manage student Account	
Actor's: Administrator	
Description: Let the Administrator view student accounts information.	
Relationship's: Include: none properties Extends: none	Covered requirements: (R1.5)
Pre-condition: none	
Basic Flow	
Actor: 1. Click the student management icon 4. choose the student	System: 2. Opening student management web page. 3. List student accounts 5. Open the student profile.
Alternative Flow: none	

Use Case Description	
Use Case Name: Manage courses	
Actor's: Administrator	
Description: Let the Administrator delete and review and approve courses	
Relationship's: Include: none properties Extends: 1.Delete courses 2. Review courses	Covered requirements: (R1.9) (R1.10) (R1.11)
Pre-condition: none	
Basic Flow	
Actor: 1. Click the manage courses icon 4.choose the courses	System: 2. Opening manage courses web page. 3. List the courses 5.Open course web page.
Alternative Flow: 5.1.1 Select delete icon 5.1.2 go back to step 3	

Use Case Description	
Use Case Name: Edit courses	
Actor's: Lecturer	
Description: Let the Lecturer delete and add and edit courses name	
Relationship's: Include: none properties Extends: 1.Delete courses 2. add courses 3. edit courses	Covered requirements: (R3.6) (R3.7) (R3.8)
Pre-condition: none	
Basic Flow	
Actor: 1. Click the edit courses icon 4.choose the courses 6. edit name or delete courses	System: 2. Opening edit courses web page. 3. List the courses 5.Open course web page.
Alternative Flow: 4.1.1 Select add icon 4.1.2 go to add course page 4.1.3 choose the file and course name 4.1.4 add the course to the list	

Use Case Description	
Use Case Name: View statistic	
Actor's: Administrator. Lecturer	
Description: Let the actor view the statistic of number of users and Number of registered courses in each month	
Relationship's: Include: none properties Extends: none	Covered requirements: (R1.4) (R1.3) (R1.6)
Pre-condition:	
Basic Flow	
Actor: 1. Click the View statistic icon	System: 2. Opening view statistic web page.
Alternative Flow:	

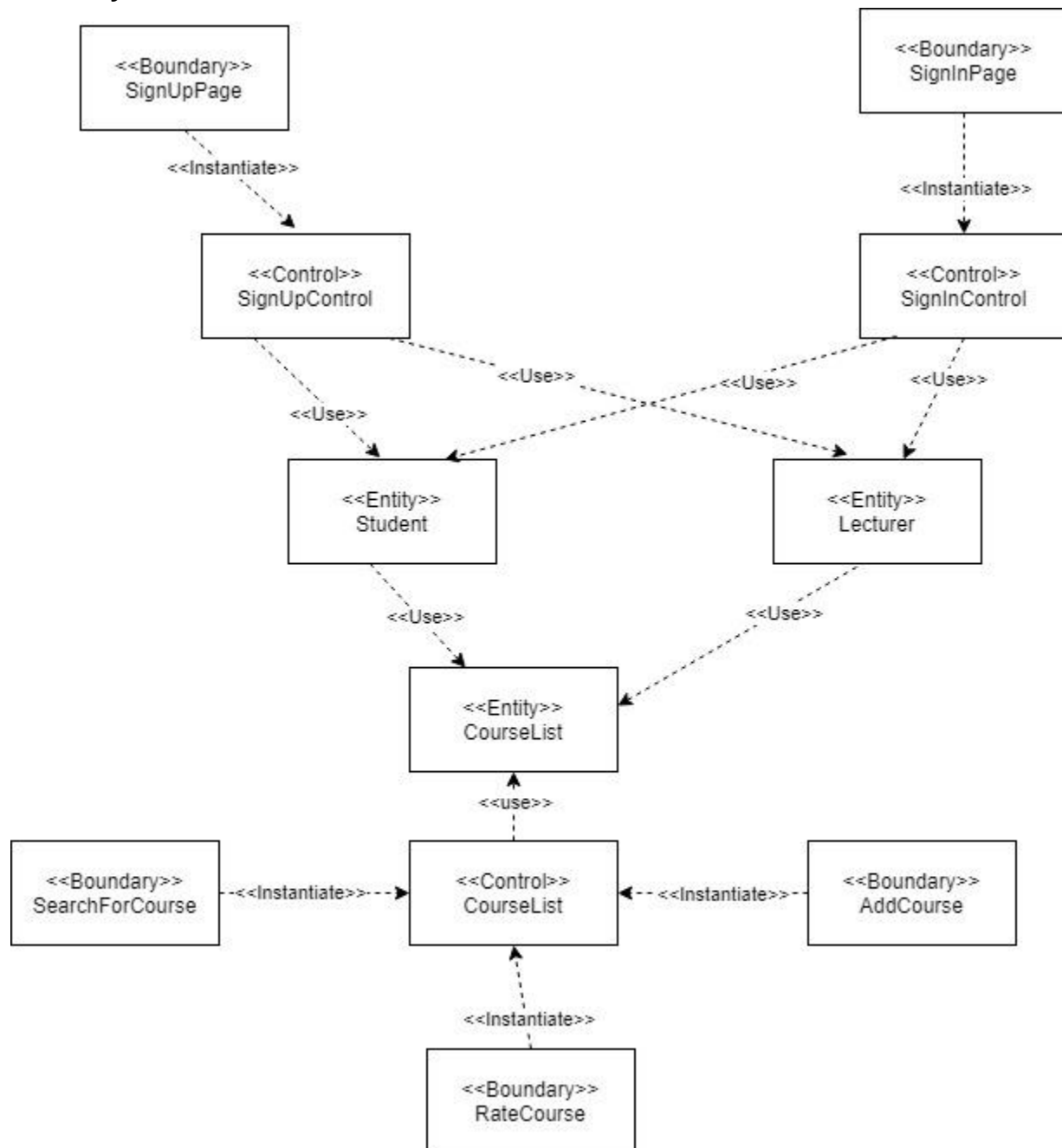
Use Case Description	
Use Case Name: Search for a course	
Actor: Student	
Description: let the student search for a course and view the course's rating.	
Relationship's: Include: non properties Extends: View rating	Covered requirements: (R2.5),(R2.10)
Pre-condition: none	
Basic Flow	
Actor: 1. Select search icon. 3.Type required course 4.Click search.	System: 2.Display search bar. 5.Display search result.
Alternative Flow: 5.1.If there is no result display 'not found'.	

Use Case Description	
Use Case Name: Take exam	
Actor: Student	
Description: let the student take the exam after finishing the course the view the exam result.	
Relationship's: Include: Display Exam result. properties Extends: none.	Covered requirements: (R2.12),(R2.13)
Pre-condition: finished the course	
Basic Flow	
Actor: 1.Click on exam icon. 4.Start solving the exam 5.Click submit.	System: 2.Display exam page. 3.Star exam time 6.Display the grade
Alternative Flow: 5.1.1 If submitted or lost connection before finishing the exam. 5.1.2 go to step 2	

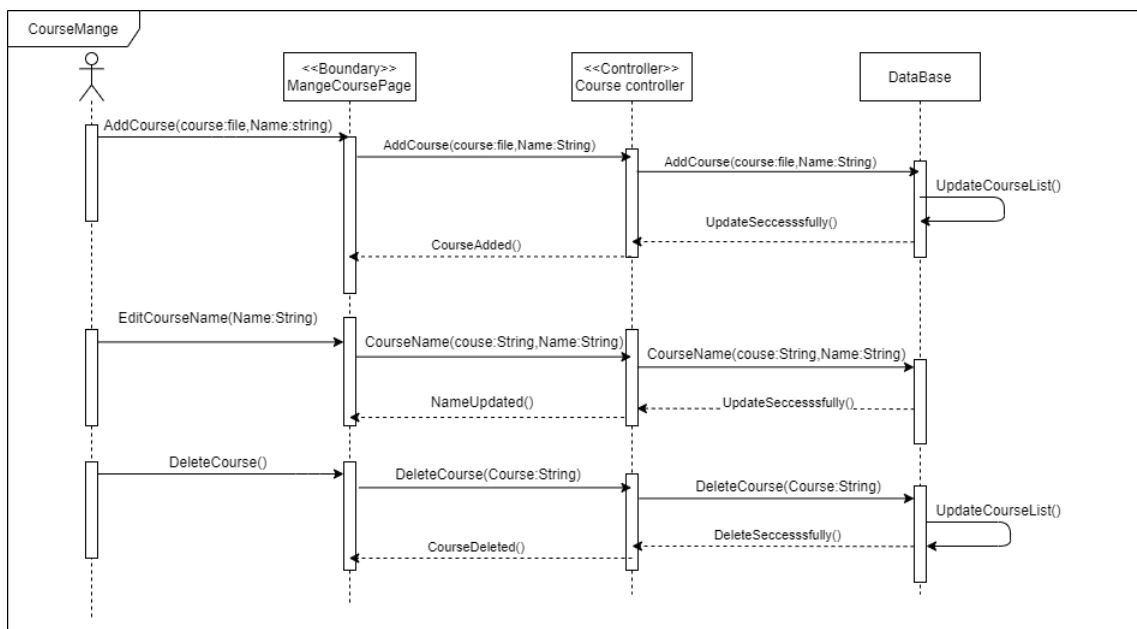
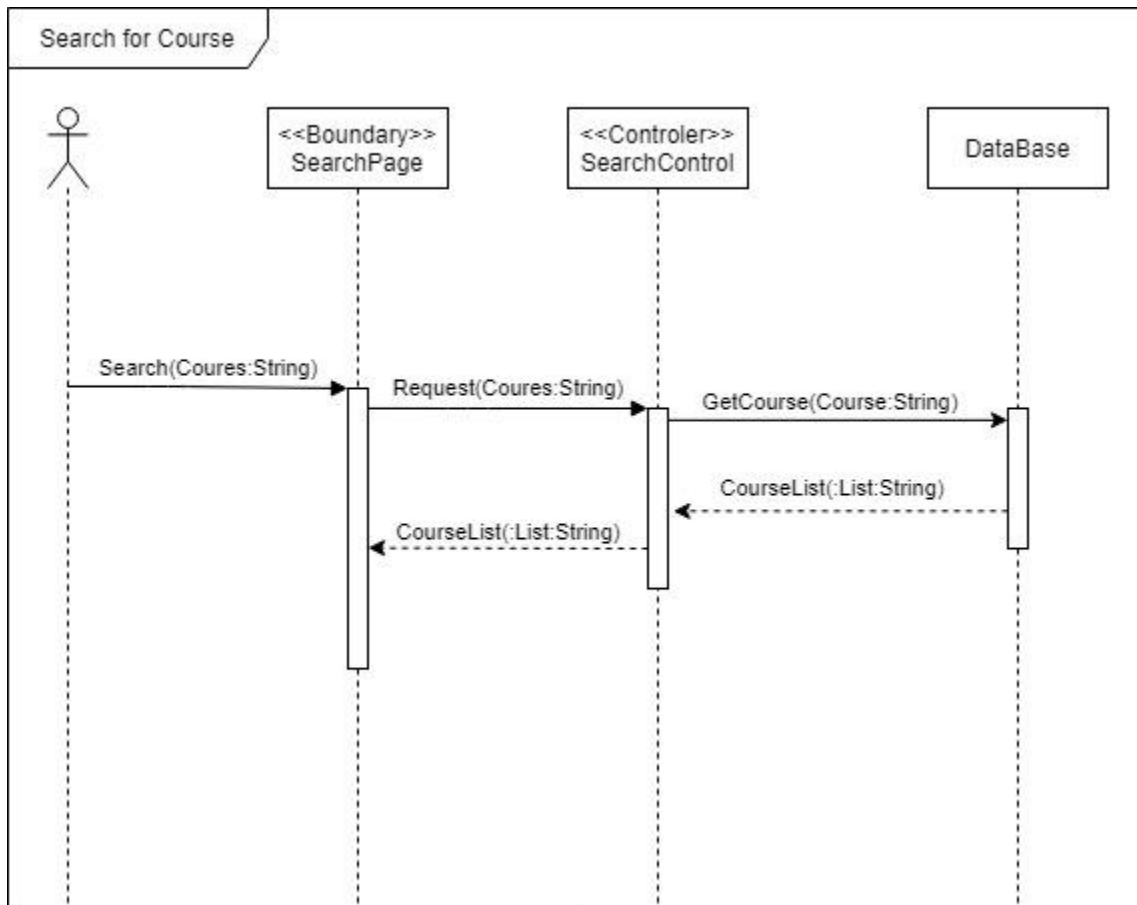
Use Case Description	
Use Case Name: Manage Course Material	
Actor: Lecturer	
Description: Let the lecturer add and edit and delete the course material	
Relationship's: Include: none. properties Extends: 1.Delete Course Material. 2.Add Course Material. 3.Edit Course Material.	Covered requirements: (R2.12), (R2.13)
Pre-condition: The lecturer already in the course page	
Basic Flow	
Actor: 1.Click on Course Material icon. 3.choose add material 5. select the type of the material. 6.select save icon.	System: 2.Display Course Material page. 4.Display the type of material. 7.Display added successfully
Alternative Flow: 3.1.1 Choose edit material name. 3.1.2 Change material name. 3.1.3 Select save icon. 3.1.4 Go Back to step 2 3.2.1 Choose delete material. 3.2.2 Click the delete confirmation. 3.2.3 Go Back to step 2	

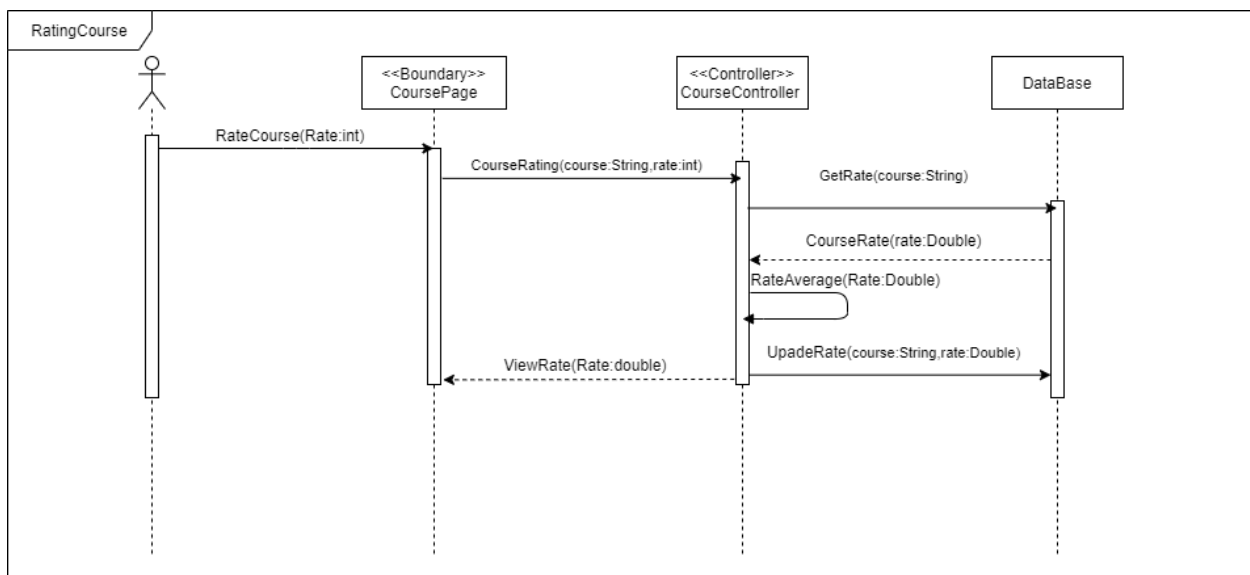
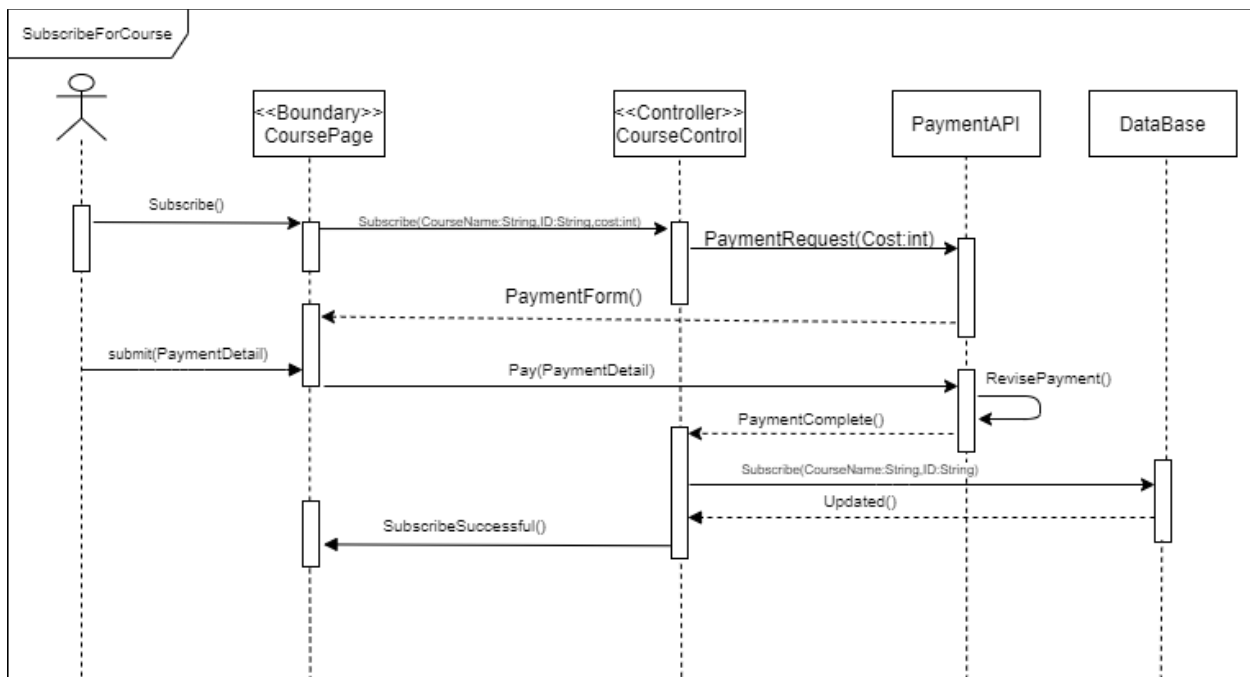
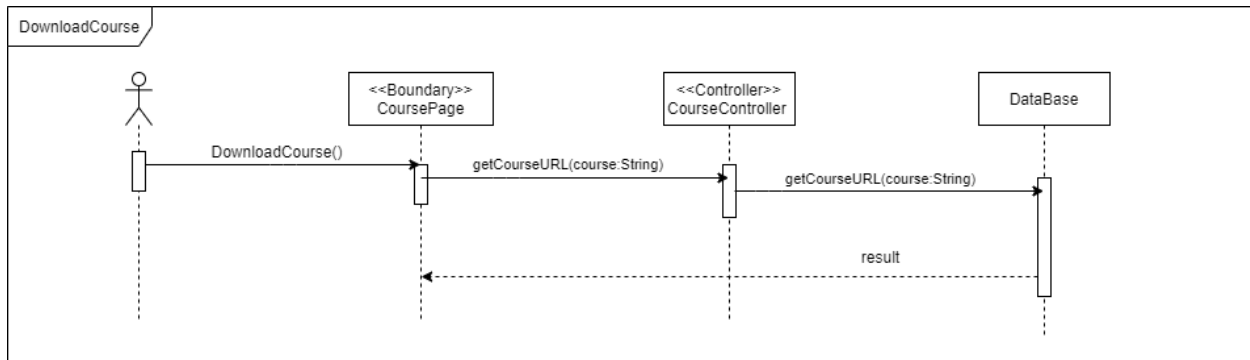
Use Case Description	
Use Case Name: Download Course	
Actor: Student	
Description: let the student to download the subscribed courses.	
Relationship's: Include: none. properties Extends: none.	Covered requirements: (R2.11)
Pre-condition: the actor is should be subscribed in the course	
Basic Flow	
Actor: 1.Click on subscribe in course. 3.click in download icon.	System: 2.Display course page. 4. Open download page
Alternative Flow: none	

8. Analysis Class

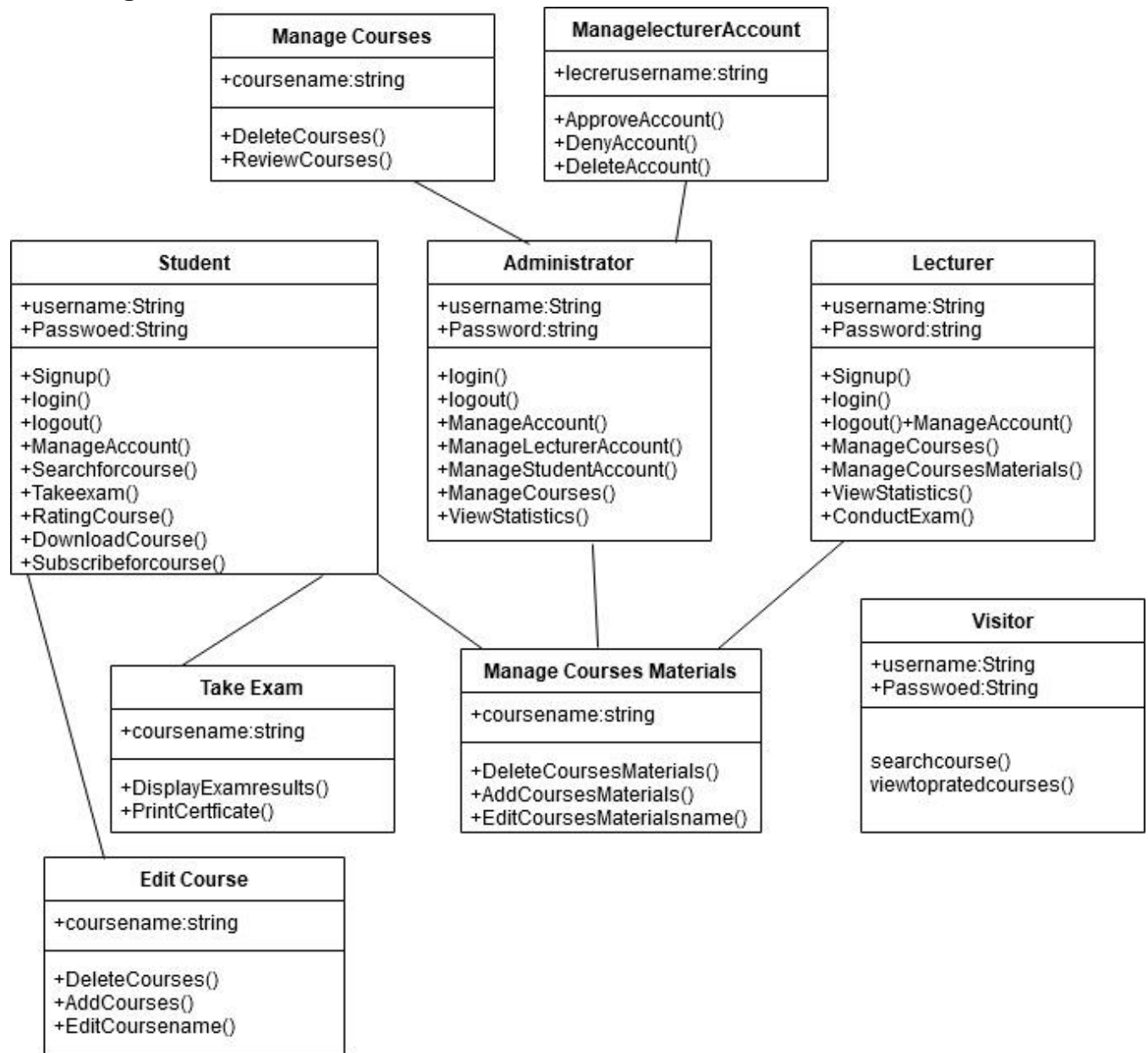


9. Interaction Diagram





10. Design Class



11. System Architecture

This part of the project we're going to talk about what and why and how we choose our system architecture.

We went for the MVC architecture, MVC patterns separate the input, processing, and output of an application. This model divided into three interconnected parts called the model, the view, and the controller.

As of why we choose it, simply because it's popular and well-suited architecture for web application, also it supports rapid and parallel development. Therefore one team member can work on the frontend while the other can work on the backend or the controller to create the business logic of the web application.

Another important features that the MVC architecture can also integrate with the JavaScript Framework. This means that MVC applications can be made to work even with PDF files, site-specific browsers, also with desktop widgets. And MVC support of Asynchronous Technique which helps for faster load time.

The most important feature is separation of concerns, it is obvious that you make frequent changes in your web application like changing colors, fonts, screen layouts, and adding new device support for mobile phones or tablets. Moreover, adding a new type of view are very easy in the MVC pattern because the Model part does not depend on the views part. Therefore, any changes in the Model will not affect the entire architecture.

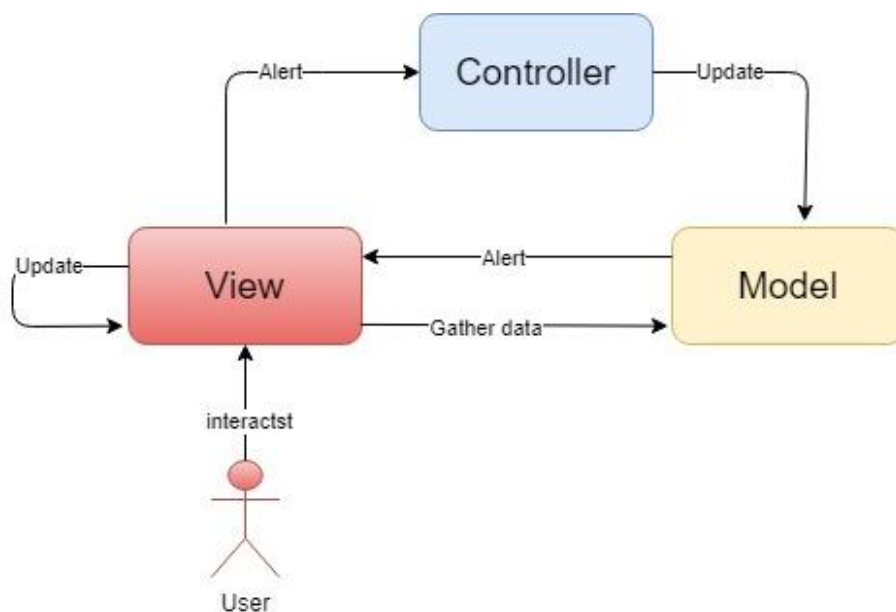
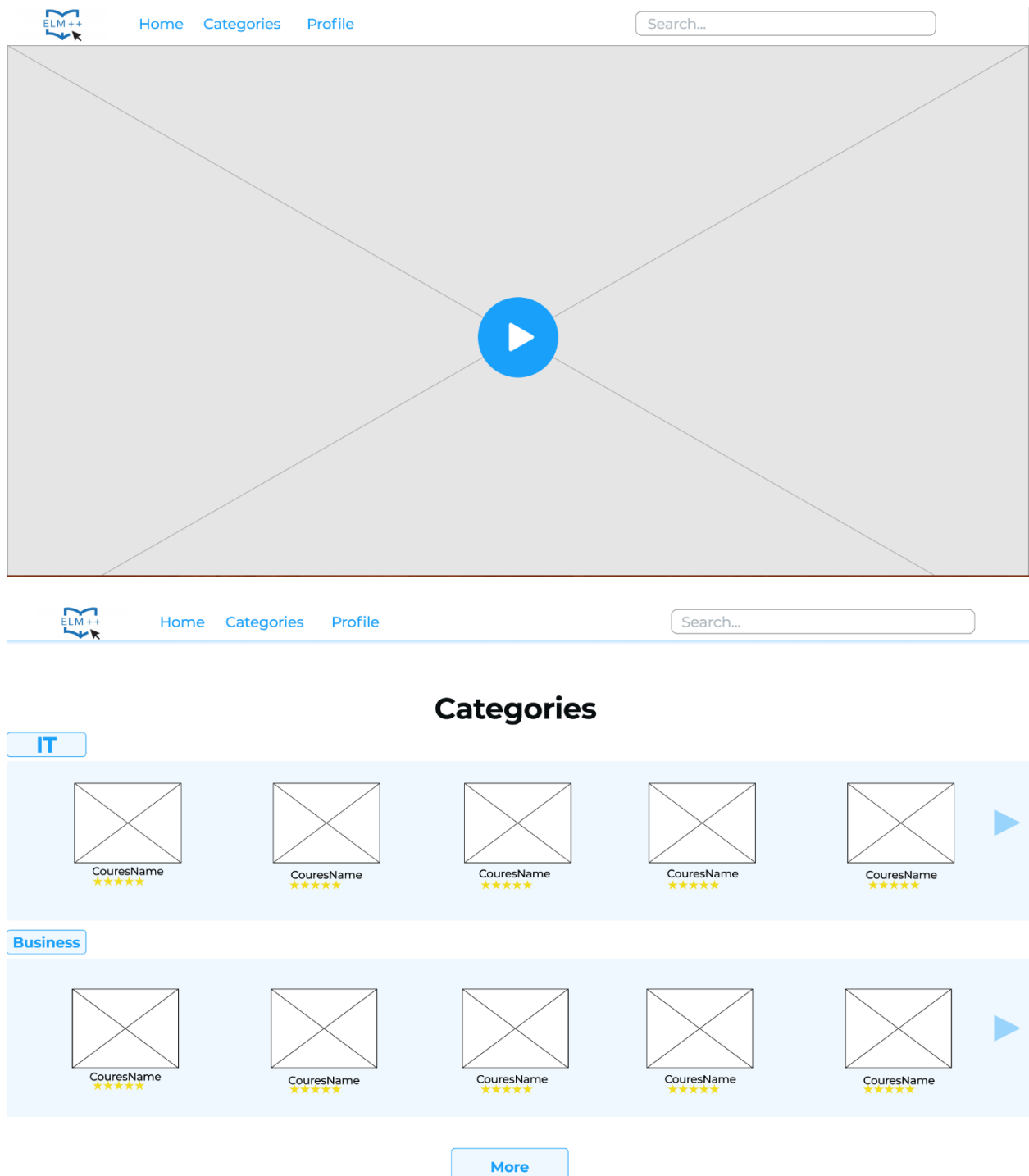


Figure 3. System Architecture

12. User Interface Mockup





Categories

Business



CouresName
★★★★★ Free

[Subscribe](#)

CouresName
★★★★★ 40 SR

[Subscribe](#)

CouresName
★★★★★ 50 SR

[Subscribe](#)

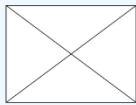
CouresName
★★★★★ Free

[Subscribe](#)

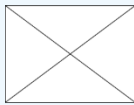
CouresName
★★★★★ Free

[Subscribe](#)

CouresName
★★★★★ 20 SR

[Subscribe](#)

CouresName
★★★★★ 30 SR

[Subscribe](#)

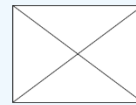
CouresName
★★★★★ Free

[Subscribe](#)

CouresName
★★★★★ 70 SR

[Subscribe](#)

CouresName
★★★★★ Free

[Subscribe](#)

CouresName
★★★★★ 50 SR

[Subscribe](#)

CouresName
★★★★★ Free

[Subscribe](#)

Payment

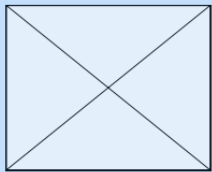
[Pay Now](#)



Edit

Edit

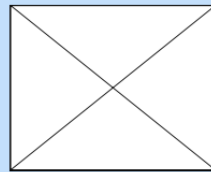
My Courses



CouresName

Delete

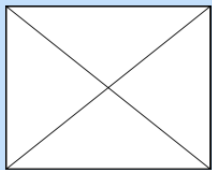
EditName



CouresName

Delete

EditName



CouresName

Delete

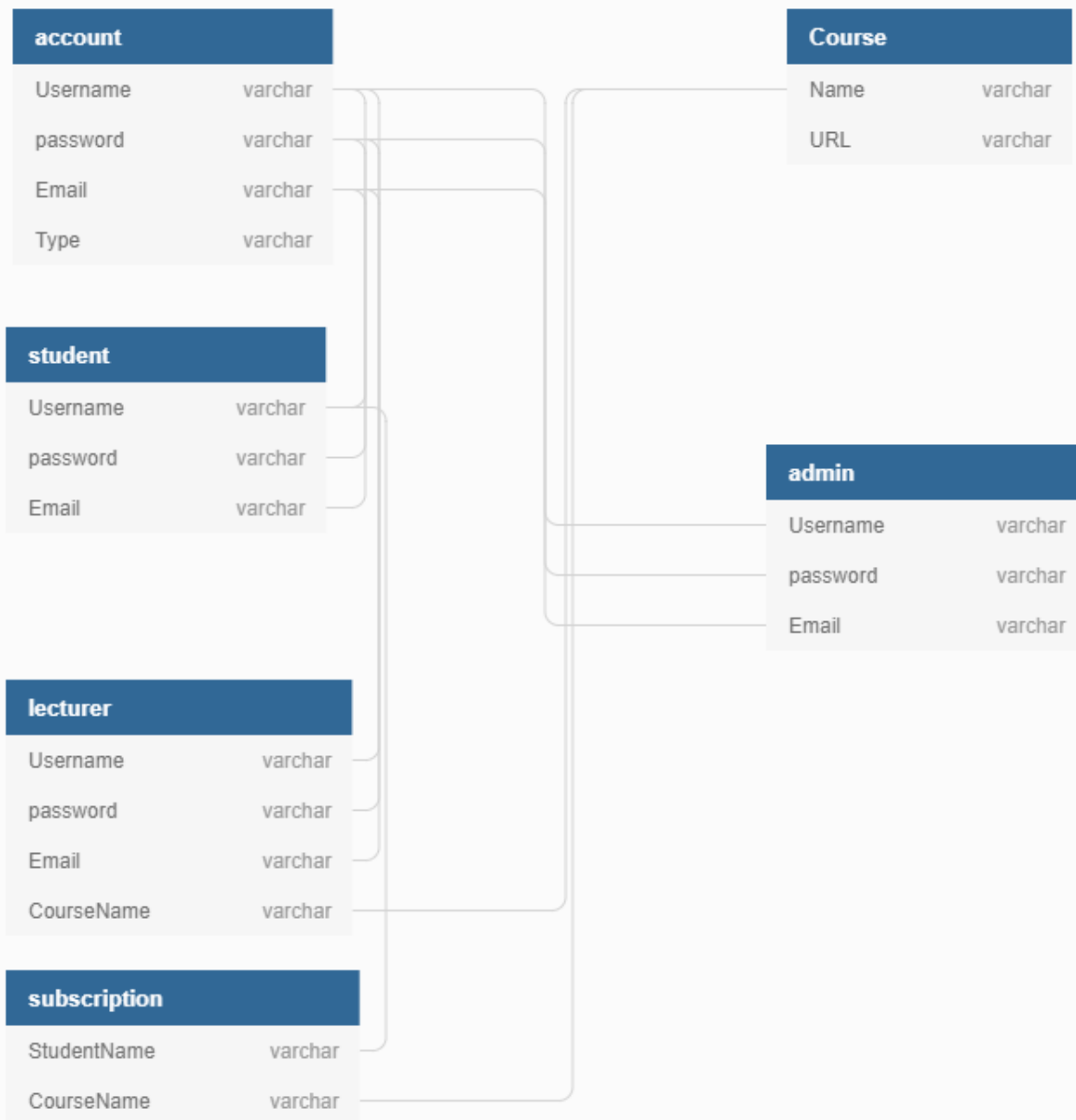
EditName



Log in

Sign up

13. Database Schema



14. Algorithms

Our project main idea is courses and how the student can find the required topic, therefore we will develop a filtering and searching algorithm for the student to be able to find the course as fast as possible, from the home page there will be the main filter witch is category filter, and then higher rating first filter. After choosing the category the student needs to search for a particular key

word for the required course.

15. Expected Deployment

Global Overview

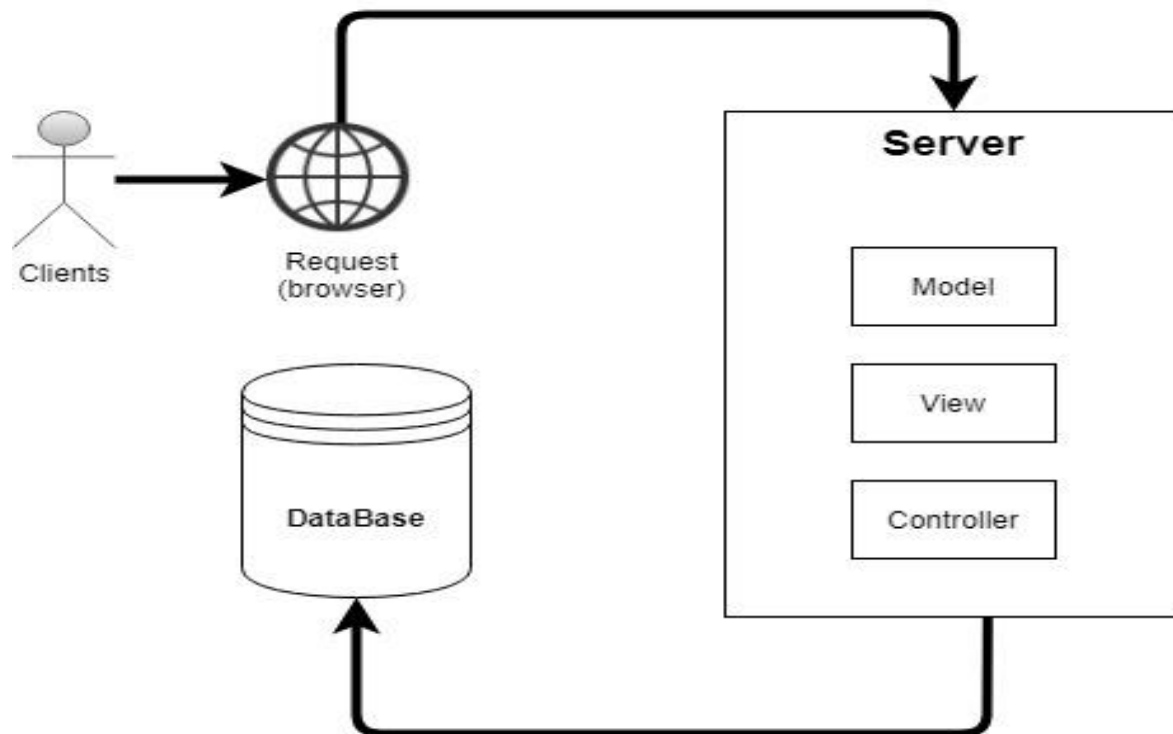


Figure 4. System deployment diagram

16. Test Scenario

Table 1. Test Table

16.1 Sign up

Test case number	input	output
1.1	email = "mohammed@gmail.com " username = "Mohammed" Password = "MoHaMmed12" Re password = " MoHaMmed12" Account type = "Student"	Account created successfully.
1.2	email= "mohammed@gmail.com " username = " Mohammed" Password = " MoHaMmed" Re password = " MoHaMmed" Account type = "Lecturer"	error message. Password did not have numbers.
1.3	email = "nasser@gmail.com " username = "Mohammed" Password = " MoHaMmed12" Re password = " " Account type = "Student"	error message. All fields should be filled.

16.2 Sign in

Test case number	input	output
1.1	email="mohammed@gmail.com" Password =" MoHaMmed12"	signed in Successfully

1.2	email = "Mohammed" Password = " MoHaMmed12"	Error message Wrong Email or password
1.3	email="mohammed@gmail.com" Password = ""	Error message All fields should be filled
1.4	email="Mohammed@gmail.com" Password = "@(892#010"	Error message Wrong Email or password

16.3 Search course

Test case number	input	output
1.1	Search keyword = "math "	The result should appears successfully.
1.2	Search keyword = ""	Error massage. Please enter a key word to search.

16.4 Create a Course

Test case number	input	output
1.1	Title = "Math 106" Description = "any text " Video = "Video 1"	Course created successfully.
1.2	Title = "Math 106" Description = "" Video = "Video 1"	Error message. All fields should be filled

17. Project Status

[Give the current status of the project following the status report template—mention the different issues you have worked on and the issues that you plan to work on in the subsequent iterations]

18. Conclusion

Write the conclusion here.

19. Reference

<https://www.coursera.org/>
<https://www.udemy.com/>
<https://www.wea.org.uk/>
<https://www.shroo7.com/>
<https://www.wikipedia.org/>