



جامعة الهرم الكندية  
AHRAM CANADIAN UNIVERSITY

# E-learning Platform

## Graduation Project Documentation

### Faculty Of Engineering Department of Computer Engineering



#### Prepared By:

Mansour Ashraf Sobh	Ahmed Mohamed Mansour
Essam Mohamed Rezk	Nada Ibrahim Mohamed
Abdullah Mohammed Shalabi	Omar Emad-Eldeen Abo-Deaf
Kareem Magdy Loulah	Amr Ahmed Megahed

#### Instructors

Dr. Mohamed Elaskily      Dr. Soha Mohamed

## CONTENTS

CHAPTER1: Introduction .....	7
ABSTRACT .....	7
1.1 The history of e-learning .....	7
1.2 Brief History of Machine Learning.....	8
1.3 RESULTS .....	10
CHAPTER2: System modeling .....	11
2.1 ROAD MAP .....	11
2.1.1 Structure.....	11
2.1.2 MAPPING .....	12
2.2 MODELING .....	13
2.2.1 Context Diagram.....	13
2.2.2 Flow Chart Diagram.....	13
2.2.3 Use Case Diagram .....	17
2.2.4 Sequence Diagram.....	22
2.2.5 ER- Diagram .....	26
CHAPTER3: System Services .....	30
3.1 Services .....	30
3.1.1 The Instructor.....	30
3.1.2 The Students .....	31
3.1.3 The Admin Only .....	31
3.2 METHODOLOGY.....	32
3.2.1 Software Development Tools .....	32
3.2.2 Machine Learning.....	34
CHAPTER4: Demo .....	37
Student.....	41
Main page .....	41
Courses page.....	42
About page.....	43

Be instructor page.....	43
User Dashboard.....	44
Course Page.....	44
Admin Dashboard Page.....	45
Instructor Dashboard.....	47
Add New Course .....	49
CHAPTER 5: Recommendation system .....	51
5.1 Description.....	51
5.1.1 Context .....	51
5.1.2 Content .....	51
5.1.3 Dataset.....	51
5.2Project Structure.....	52
5.3Structure model.....	53
5.4Data Analysis .....	54
5.5Data Pre-processing .....	59
5.6Features engineering .....	62
5.7get_recommendations.....	66
5.8What is New Feature ? .....	68
refrences.....	70

## **LIST OF FIGURES:**

Figure 2.1: Diagram of the Components of the Educational site .....	14
Figure 2.2: A diagram that explains the contents of the site's parts and the features of each function .....	15
Figure 2.3: A diagram showing how to think .....	16
Figure 2.4: Context Diagram of platform .....	17
Figure 2.5: A diagram flow chart with homepage and dashboard .....	18
Figure 2.6: A diagram flow chart with start courses .....	19
Figure 2.7: A diagram flow chart with homepage and dashboard .....	20
Figure 2.8: Flow chart diagram to show dashboard .....	21
Figure 2.9: Flow chart diagram to show home page .....	22
Figure 2.10: Use Case diagram to show interaction between guest and system .....	23
Figure 2.11: Use Case diagram to show interaction between student and system .....	24
Figure 2.12: Use Case diagram to show interaction between instructor and system .....	26
Figure 2.13: Use Case diagram to show interaction between admin and system .....	27
Figure 2.14: Use Case diagram to show interactions between all users and system .....	29
Figure 2.15: Sequence diagram to show User and system .....	31
Figure 2.16: Sequence diagram to show admin and system .....	32
Figure 2.17: Sequence diagram to show student and system .....	33
Figure 2.18: Sequence diagram to show student and system .....	35
Figure 2.19: ER diagram .....	36
Figure 2.20: ER diagram to user .....	37
Figure 2.21: ER diagram to course .....	37
Figure 2.22: ER diagram to lesson .....	38
Figure 2.23: ER diagram to question .....	38
Figure 2.24: ER diagram to completed .....	39
Figure 2.25: ER diagram to rated .....	40
Figure 4.1: Index page .....	48

Figure 4.2: Courses in index page .....	49
Figure 4.3: Paid and Free Courses in index page .....	49
Figure 4.4: Login page .....	50
Figure 4.5: Register page .....	50
Figure 4.6: Register page with existing accounts .....	51
Figure 4.7: Publish course by instructor .....	52
Figure 4.8: instructor privilege .....	53
Figure 4.9: Menu (navbar) .....	54
Figure 4.10: Footer .....	54
Figure 4.11: Slider .....	54
Figure 4.12: Discover courses .....	55
Figure 4.13: Filter courses .....	55
Figure 4.14: Information about the platform .....	56
Figure 4.15: Information need to fill to be a constructor .....	57
Figure 4.16: User dashboard .....	58
Figure 4.17: Course page .....	59
Figure 4.18: Admin dashboard .....	60
Figure 4.19: Admin dashboard (courses manager) .....	61
Figure 4.20: Admin dashboard (students' manager) .....	62
Figure 4.21: Instructor dashboard .....	63
Figure 4.22: Course dashboard .....	64
Figure 4.23: Update course .....	65
Figure 4.24: Add new courses .....	66
Figure 4.25: Choose free or paid course .....	67
Figure 5.1: Display the data .....	69
Figure 5.2: Road map .....	71
Figure 5.3: Shape of the data .....	71
Figure 5.4: Represent top 5 machine learning courses respect to rating .....	72

Figure 5.5: Represent top 5 courses respect to reviews .....	73
Figure 5.6: Represent top 5 courses respect to price .....	73
Figure 5.7: Represent top 5 data science/big data courses respect to rating .....	74
Figure 5.8: Represent top 5 python courses respect to rating .....	74
Figure 5.9: Represent top 5 books respect to rating .....	75
Figure 5.10: Represent top 5 data science/big data books respect to rating .....	75
Figure 5.11: Represent top 5 python books respect to rating .....	76
Figure 5.12: Represent top 5 machine learning books respect to rating .....	76
Figure 5.13: Explanation of data processing .....	77
Figure 5.14: Explanation of data processing II .....	78
Figure 5.15: Result after apply processing .....	78
Figure 5.16: create variable and store sample of data .....	80
Figure 5.17: import TF-IDF process .....	81
Figure 5.18: Result of TF-IDF process .....	81
Figure 5.19: Similarity between two documents .....	83
Figure 5.20: Example Simple Data .....	84
Figure 5.21: Result of similarity between (ex1) and (ex2) .....	84
Figure 5.22: Result of similarity between (ex1) and (ex4) .....	84
Figure 5.23: Get recommendations .....	85
Figure 5.24: Example data from recommendation system .....	86
Figure 5.25: Recommend books based on courses .....	86
Figure 5.26: Using Stream Lit with recommendation system .....	87

# CHAPTER 1: INTRODUCTION

## ABSTRACT

We want to make an E-Learning platform for people of a variety of ages containing a lot of categories & fields such as development, designing, finance, etc. we want to reach the greatest number of people possible and help them reach their goals.

We are planning to make an easy-to-use website for people of different ages. Courses are going to be divided into different parts (weeks). Tasks are going to be implemented between videos in the week's section, Quizzes will be made between weeks if applicable by the professor.

The first main goal of this website is that every course will contain a lot of questions. The questions are divided into three main categories: Tasks, quizzes, Final exams. These questions are linked with the content of the course so the question will be linked with the name of the course, chapter, topic.

We are also working on a recommendation system for a series of courses in a specified category. For example, a person who took an HTML and CSS course and finished the recommendation system should recommend Java to complete front-end web development.

we also want to make a chat-bot contain a recommendation system to help the user to join new courses depending on the courses he enrolled before

### 1.1 The history of e-learning

The term “E-learning” has only been in existence since 1999, when the word was first utilized at a CBT systems seminar. Other words also began to spring up in search of an accurate description such as “online learning” and “virtual learning”. However, the principles behind E-learning have been well documented throughout history, and there is even evidence which suggests that early forms of E-learning existed as far back as the 19th century.

An E-learning timeline Long before the internet was launched, distance courses were being offered to provide students with education on particular subjects or skills. In the 1840’s Isaac Pitman taught his pupils shorthand via correspondence. This form of symbolic writing was designed to improve writing speed and was popular amongst secretaries, journalists, and other individuals who did a great deal of note taking or writing. Pitman, who was a qualified teacher, was sent completed assignments by his students via the mail system and he would then send them more work to be finished.

In 1924, the first testing machine was invented. This device allowed students to tests themselves. Then, in 1954, BF Skinner, a Harvard Professor, invented the “teaching machine”, which enabled schools to administer programmed instruction to their students. It wasn’t until 1960 however that the first computer-based training program was introduced to the world.

This computer-based training program (or CBT program) was known as PLATO-Programmed Logic for Automated Teaching Operations. It was originally designed for students attending the University of Illinois, but ended up being used in schools throughout the area.

With the introduction of the computer and internet in the late 20th century, E-learning tools and delivery methods expanded. The first MAC in the 1980's enabled individuals to have computers in their homes, making it easier for them to learn about particular subjects and develop certain skill sets. Then, in the following decade, virtual learning environments began to truly thrive, with people gaining access to a wealth of online information and E-learning opportunities.

In the 2000's, businesses began using E-learning to train their employees. New and experienced workers alike now had the opportunity to improve upon their industry knowledge base and expand their skill sets. At home individuals were granted access to programs that offered them the ability to earn online degrees and enrich their lives through expanded knowledge. Today, E-learning is more popular than ever, with countless individuals realizing the benefits that online learning can offer.

2010Udemy, one of the world's largest and most popular online course platforms was founded in 2010 by a trio of Turkish professionals. Although it started out small, it ushered in a novel E-learning business model that has catapulted it to great heights of success.

As of January 2020, over 57,000 teachers and giving online courses to more than 50 million global students in more than 65 languages.

2020The COVID-19 outbreak was the first event in a long time that dealt a real blow to traditional classroom-based training and openly exposed the limitations of this form of learning.

With lockdowns getting implemented all over the world to curb the spread of the virus, students of all ages found themselves stuck at home with education having been brought to a complete halt.

In this testing time, parents and educational institutions alike were forced to consider and adopt the alternative, i.e., of E-learning. In a short span of time, students starting attending live online classrooms where a teacher would be teaching from her home using teaching software.

Because of the pandemic, the world has come to see the versatility and benefits of E-learning, and this has already convinced a lot of people that this is now the way forward.

Below is a quick snapshot of the progression and history of E-learning, plus key milestones.

[1] [A brief history of E-learning \(infographic\) - eFront Blog \(efrontlearning.com\)](#)

## 1.2 Brief History of Machine Learning

Just a few decades ago, machine learning was nowhere near reality and seemed like the subject of a science fiction novel. Today, it's an essential technology in the field of artificial intelligence that helps us to carry out tasks ranging from driving cars to finding the best products.

Thanks to countless mathematicians, philosophers, and computer scientists, we have gone a very long way from the dream of self-learning machines to the actual field of machine learning. The future is only going to expand this trend as the machine learning market is expected to grow from \$1.03 billion in 2016 to a smashing \$8.81 billion by 2022, registering a Compound Annual Growth Rate of 44.1% during that period.

But how did machine learning come about? Who was the first person to create the concept of a self-learning machine? And how did that concept develop over time? In this article, we take a closer look at

the history of machine learning to show you how its origins, methods, the current trends and key use cases, and its future.

It's always good to start with the basics, so let's answer this question. Machine learning is a particular application of artificial intelligence (AI) that provides machines with the ability to automatically learn and improve from experience – but without being explicitly programmed to do so.

This is the essence of machine learning. It focuses on building software that can access data and use it to train itself in order to provide better results, without software developers having to train it themselves.

The primary goal of machine learning is to allow machines such as computers to learn automatically without any human intervention or assistance and then adjust their actions accordingly based on the insights they have uncovered.

It all starts with observations or data such as direct experience, instruction, or examples. A machine learning algorithm looks for patterns in data and analyzes the examples we feed it. On the basis of these examples, it generates insights that allow making smarter decisions.

### The origins and history of machine learning

1952 – Arthur Samuels, the American pioneer in the field of artificial intelligence and computer gaming, wrote the very first computer learning program. That program was actually the game of checkers. The IBM computer would first study which moves lead to winning and then put them into its program.

1967 – The nearest neighbor algorithm was written for the first time this year. It allows computers to start using basic pattern recognition. This algorithm can be used to map a route for a traveling salesman that starts in a random city and ensures that the salesman passes by all the required cities in the shortest time. Today, the nearest neighbor algorithm called KNN is mostly used to classify a data point on the basis of how their neighbors are classified. KNN is used in retail applications that recognize patterns in credit card usage or for theft prevention when implemented in CCTV image recognition in retail stores.

The 1990s – during the 1990s, the work in machine learning shifted from the knowledge-driven approach to the data-driven approach. Scientists and researchers created programs for computers that could analyze large amounts of data and draw conclusions from the results. This led to the development of the IBM Deep Blue computer, which won against the world's chess champion Garry Kasparov in 1997.

2010 – this year saw the introduction of Microsoft Kinect that could track even 20 human features at the rate of 30 times per second. Microsoft Kinect allowed users to interact with machines via gestures and movements.

2011 – this was an interesting year for machine learning. For starters, IBM's Watson managed to beat human competitors at Jeopardy. Moreover, Google developed Google Brain equipped with a deep neural network that could learn to discover and categorize objects (in particular, cats).

2012 – Google X lab developed a machine learning algorithm able to autonomously browse YouTube videos and identify those that contained cats.

2014 – Facebook introduced Deep Face, a special software algorithm able to recognize and verify individuals on photos at the same level as humans.

2015 – this is the year when Amazon launched its own machine learning platform, making machine learning more accessible and bringing it to the forefront of software development. Moreover, Microsoft created the Distributed Machine Learning Toolkit, which enables developers to efficiently distribute machine learning problems across multiple machines. During the same year, however, more than three thousand AI and robotics researchers endorsed by figures like Elon Musk, Stephen Hawking, and Steve Wozniak signed an open letter warning about the dangers of autonomous weapons that could select targets without any human intervention.

2016 – this was the year when Google's artificial intelligence algorithms managed to beat a professional player at the Chinese board game Go. Go is considered the world's most complex board game. The AlphaGo algorithm developed by Google won five out of five games in the competition, bringing AI to the front page.

2020 – Open AI announced a groundbreaking natural language processing algorithm GPT-3 with a remarkable ability to generate human-like text when given a prompt. Today, GPT-3 is considered the largest and most advanced language model in the world, using 175 billion parameters and Microsoft Azure's AI supercomputer for training.

[2] [A Brief History of Machine Learning - DATAVERSITY](#)

## 1.3 RESULTS

An easy-to-use e-learning platform for different people of different ages covering many categories such as development, business, finance, information technology and software, design, and marketing. The video by referring to it and a display screen showing the information of the participants in this course and the provision of some services to the trainers with a monthly subscription, such as analyzing the students' data to assist the trainer in discovering defects and working on them to improve the service and reach the largest amount of scientific benefit in a simplified way.

Additional features in the educational site, which are the proposal of some educational books based on machine learning because there is a group of students who are always looking for additional information, the aim of which is to gain experience and review information faster, in addition to making the platform work with universities in a simplified way that makes it easier for students to access the contents of the educational course With the knowledge of the grades obtained electronically after the exams automatically, and finally the addition of a chat bot that facilitates access to information that the student cannot access and provide assistance between the student and the teacher in the event of a problem for the student such as the lack of content or the inability to enter an exam or the presence of an error.

# CHAPTER2: SYSTEM MODELING

In this chapter we will describing system in 2 fields as first field is Road map that describe structure, mapping, brainstorming for system, the second field is modeling that describe context diagram, flowchart diagram, use case diagram, sequence diagram, ER diagram

## 2.1 ROAD MAP

Show what is the structure and what is the component in the system and how its connected together

### 2.1.1 Structure

Diagram describing the contents of the educational site, which contains some important and indispensable characteristics in a simplified presentation of the access to the site, the content of the courses, site users and purchases as shown in figure 2.1

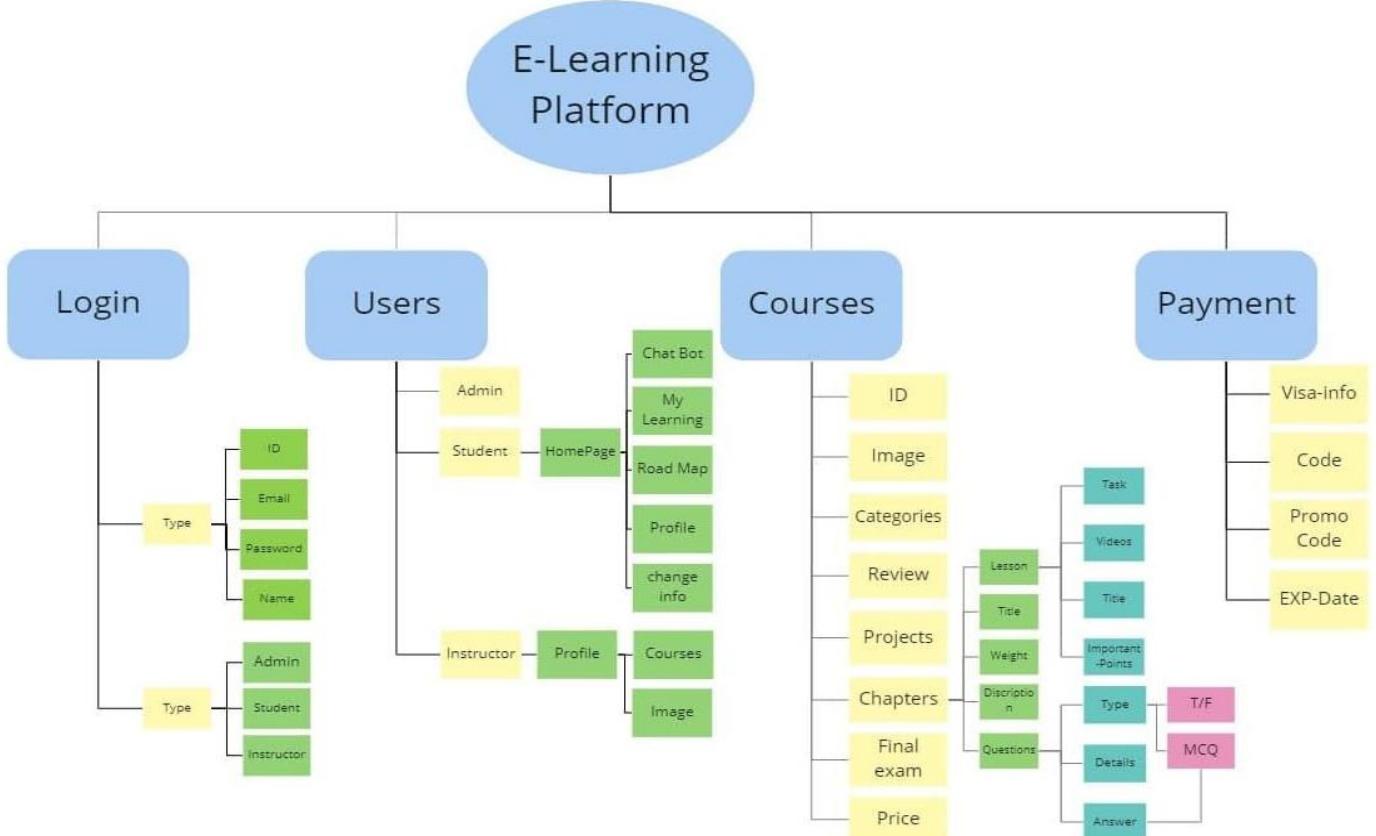


Figure 2.1: Diagram of the Components of the Educational site

## 2.1.2 MAPPING

A diagram that describes the educational site more accurately, the way to connect the parts to each other, and explain the entry paths that will be relied upon in building the structure of the site as shown in figure 2.2

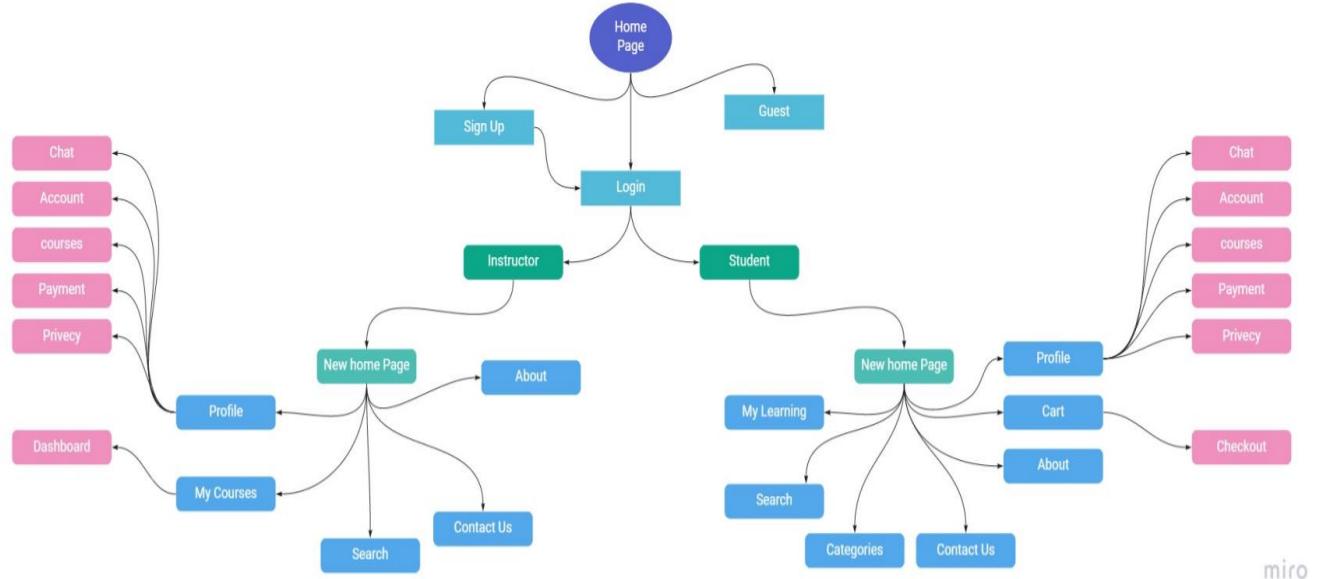


Figure 2.2: A diagram that explains the contents of the site's parts and the features of each function

## 2.1.3 BRAINSTORMING

collecting the necessary information to build an educational website by thinking about the necessary contents of the site and putting the features and collecting ideas from several different sources and putting the ideas in the form of an illustration as shown in figure 2.3

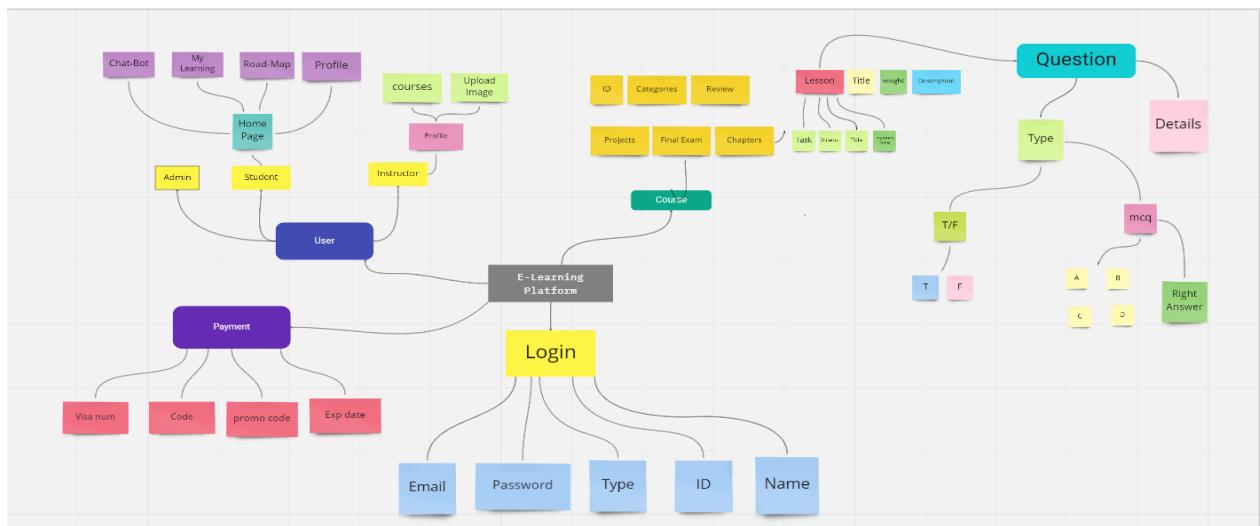


Figure 2.3: A diagram showing how to think

## 2.2 MODELING

Show the models in our system

### 2.2.1 Context Diagram

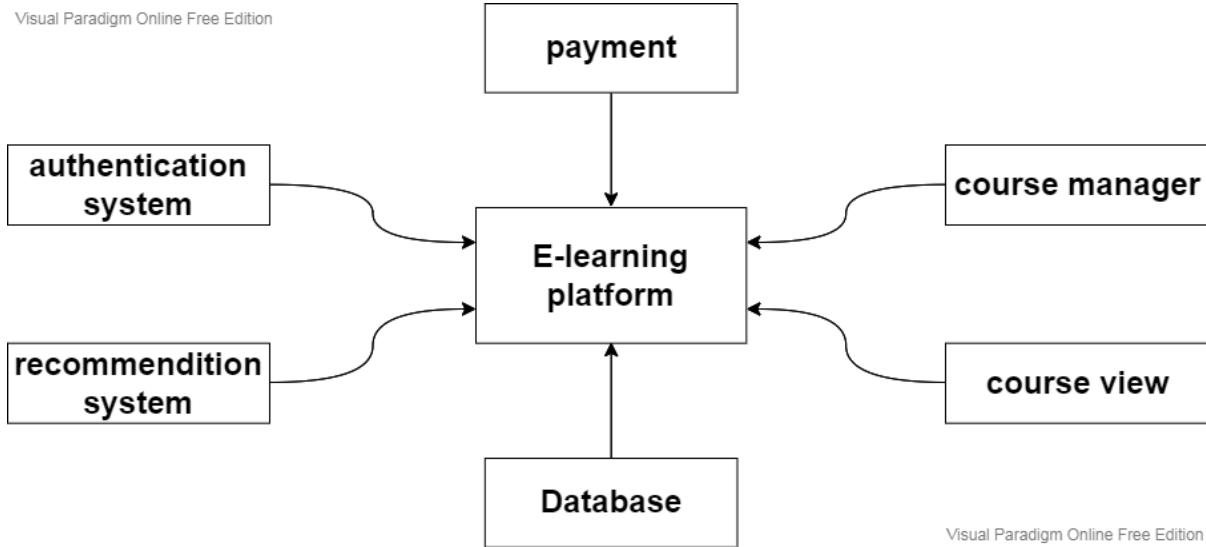


Figure 2.4: Context Diagram of platform

The following diagram represent the whole system in outside look

that show platform communicate with first system is authentication system to join in platform that Verify that the registration is correct, the second system is course management that manage all feature in course by instructor ,the third system is course view that manage all feature has to student , the fourth system is recommendation system that recommend courses to student, the fifth system is payment system that instructor took it ,the sixth system is database has all data stored as shown in figure 2.4

### 2.2.2 Flow Chart Diagram

At the Homepage, the user can either log-in the site or browse the site. If he is going to log-in and he was an old user that means that he is going to use his credentials, if he is new to the site he is going to sign up as a new user the data-base check the typed credentials for duplicates and checks other security and finishes with logging in the user. The user can view and search the courses, and select a specific course to view or buy or to add to a wish list, if he buys the course the course will be added to his owned courses. The user can also go to his dashboard the site checks for his type (student, instructor) if he is a student he can view and manage his owned courses, if he was an instructor, he can manage his courses add, edit and do other instructor services as shown in figure 2.5

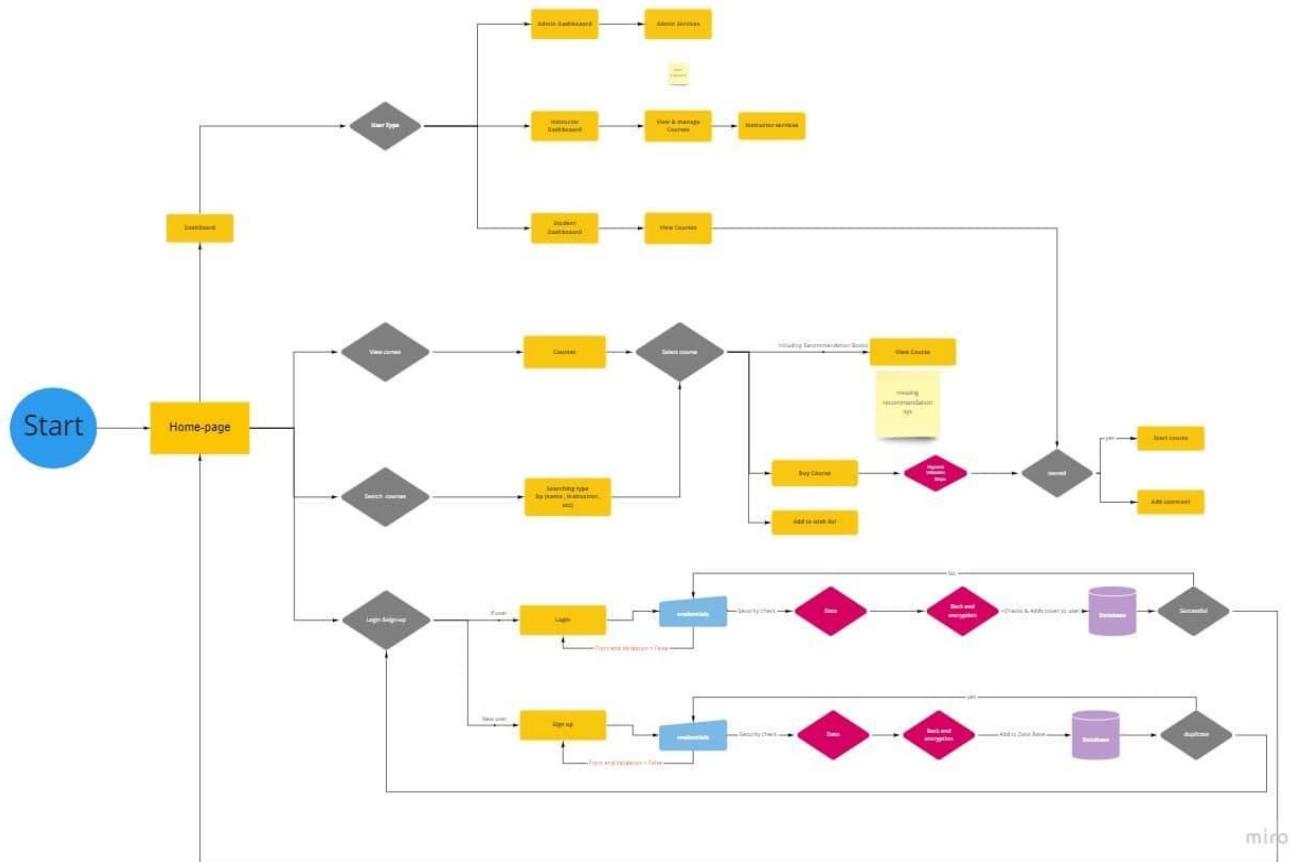


Figure 2.5: A diagram flow chart with homepage and dashboard

Diagram showing if the student starts a new course, he will pass through a set of paths that lead to making sure that he completed the course successfully as shown in figure 2.6 , 2.7

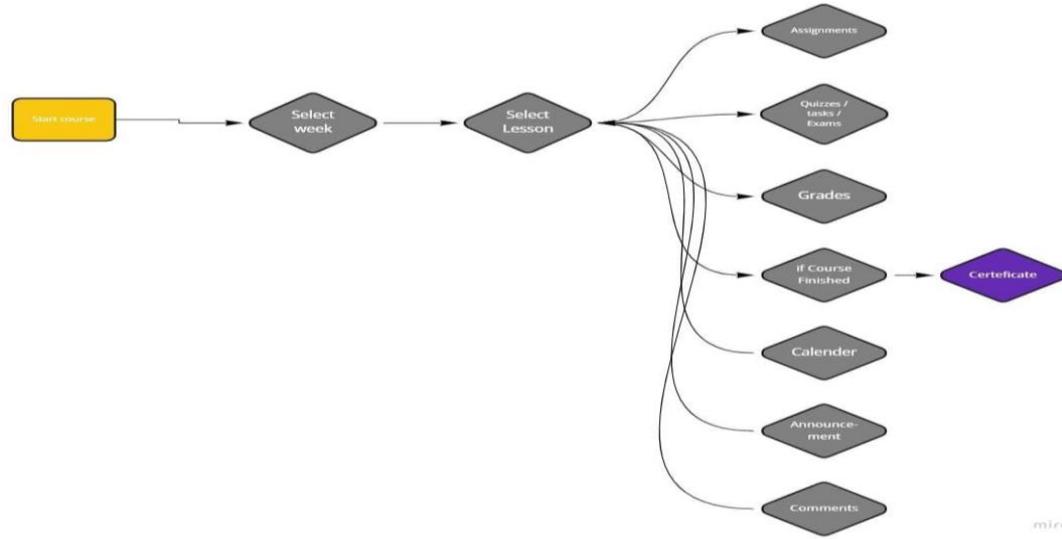


Figure 2.6: A diagram flow chart with start courses

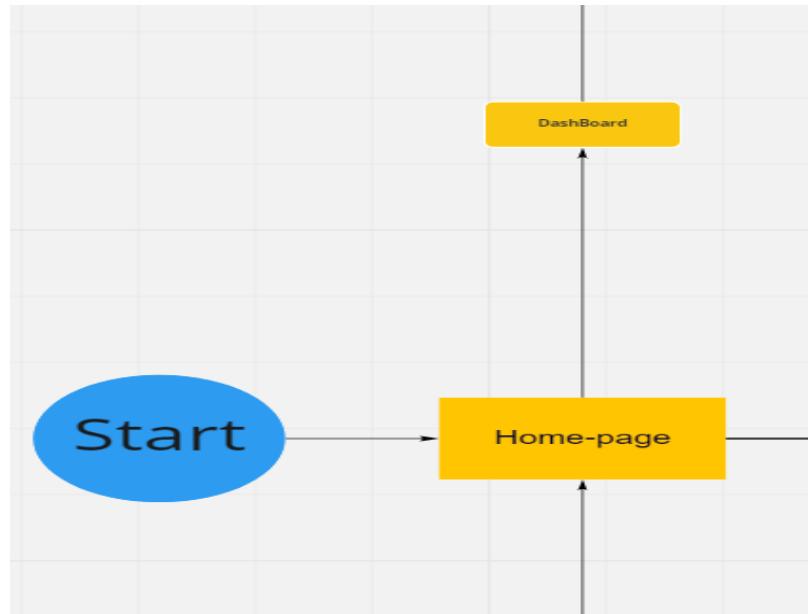


Figure 2.7: A diagram flow chart with homepage and dashboard

A diagram showing the features available in the dashboard, such as the types of users available (student, instructor, and admin), so a student has characteristics that differ from the instructor. Provided that it is his own, he has all the powers and finally the admin has all the powers of the site, he is responsible for managing the site, coordinating and reviewing the content before accepting the courses as shown in figure 2.8

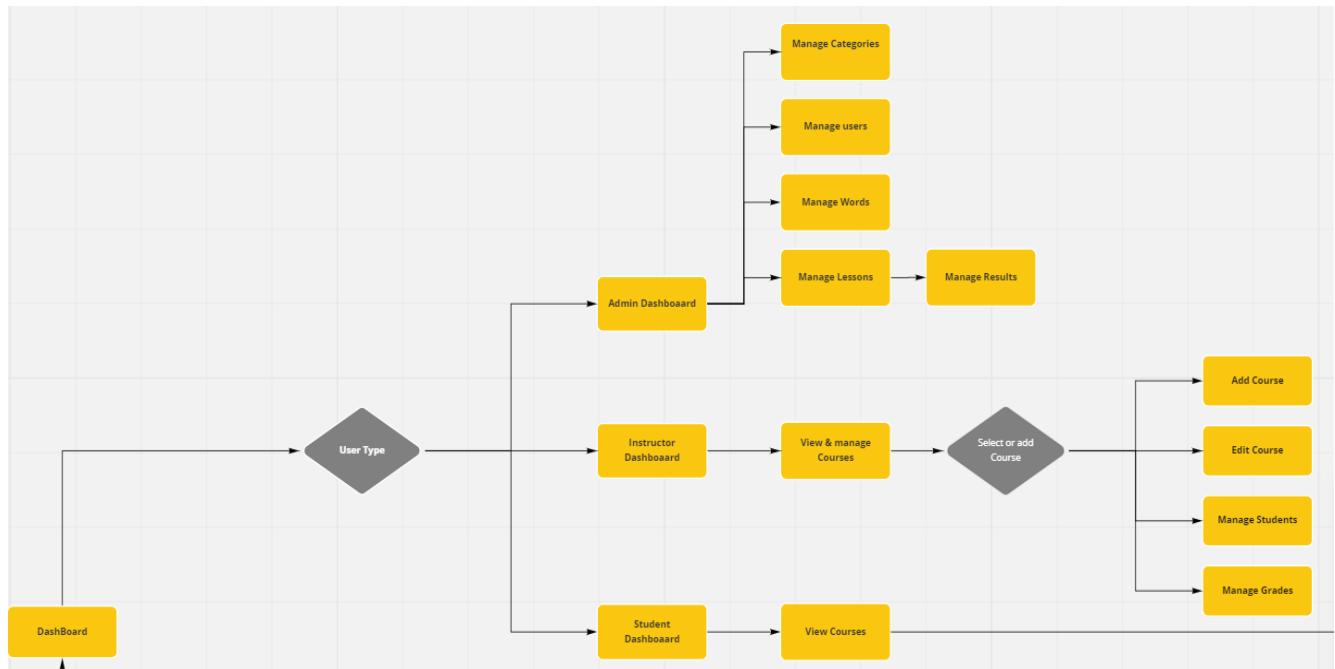


Figure 2.8: Flow chart diagram to show dashboard

A diagram describing the contents of the home page of the educational site, which contains registration processes, course views, purchasing courses, knowing the most viewed courses, and providing the best suggestions from books based on the available courses, with an explanation of how to store important information in the database and the processes that take place in case of entry as shown in figure 2.9

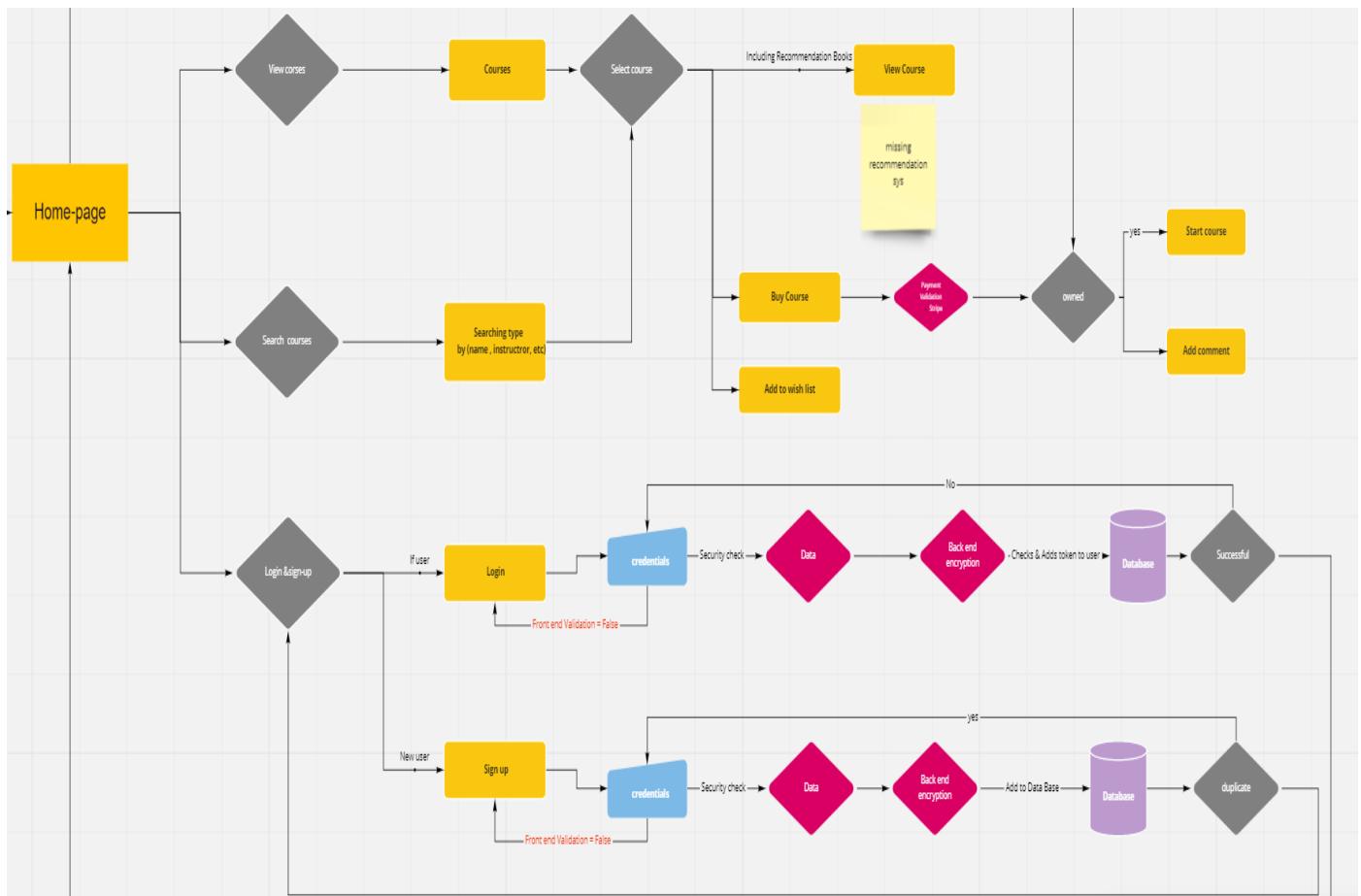


Figure 2.9: Flow chart diagram to show home page

### 2.2.3 Use Case Diagram

In this diagram show guest's function that he can does in platform he can choose track to see courses in this track or see all courses or search about specific course with name of course or instructor then can view details of course and calendar of it and can view reviews and rating of course and view instructor info to know about him and see his courses then if he wants this course can add to cart and continue search. as shown in figure 2.10

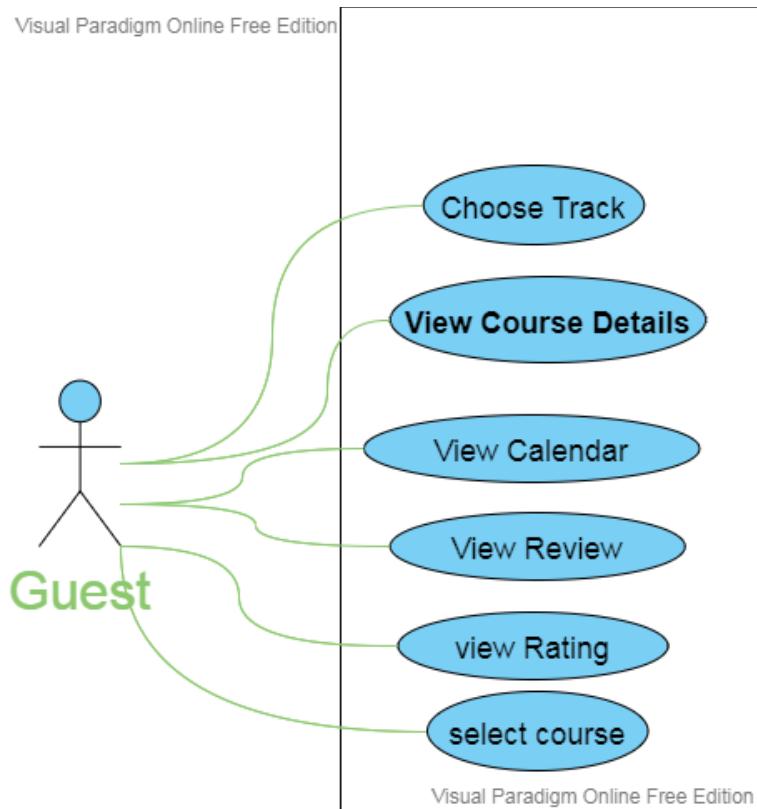


Figure 2.10: Use Case diagram to show interaction between guest and system

In this diagram show student's function that he can does in platform first he should register as student then transfer to dashboard that makes him can see his courses, calendar, contact platform and use chatbot to solve his problem and change in account setting .Platform chance for a student to choose his track to take a course that is recommended by platform or choose by himself then when finding the target course he can view the info about instructor and course reviews and rating then can enroll the course then pay for it that makes the course add to his dashboard to study. He can download course material and send messages to instructors and can view his course, videos, lessons, description, chapters, title, and others. He can view exams, assignment, quiz, tasks and submit answers of these. He can add comments and add rating to the course, view announcements, view answers of quiz, exams and tasks and put the marks and it gives him the opportunity to view FAQ and notes of instructors and add review about course and instructor as shown in figure 2.11

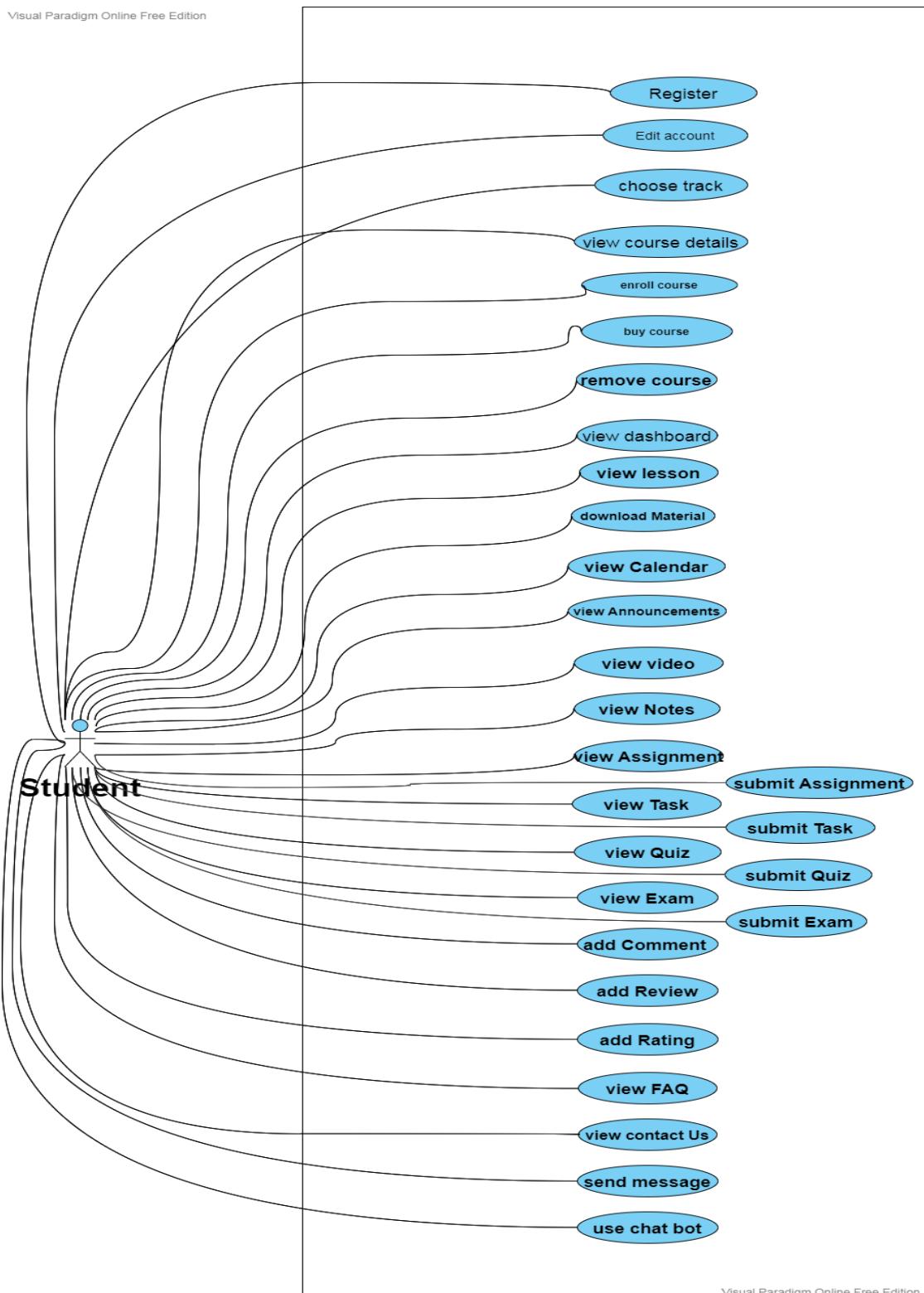


Figure 2.11: Use Case diagram to show interaction between student and system

In this diagram show instructor's function that he can does in platform first he should register as instructor then transfer to dashboard that makes him can see his courses, calendar, contact platform and use chatbot to solve his problem. Platform chance for instructor to add information about himself and his courses and change in account setting. He can add, edit in his calendar, or respond to messages that get to him from students and can upload, delete, edit in his course, videos, lessons, info, title, and others. He can add, remove course, exam, assignment, quiz, and he can review comments and reply to comments, view rating of course, view answers of quiz, exams and tasks and put the marks and it gives him the opportunity to put FAQ and notes to show to the students and takes the course's money that the student bought. as shown in figure 2.12

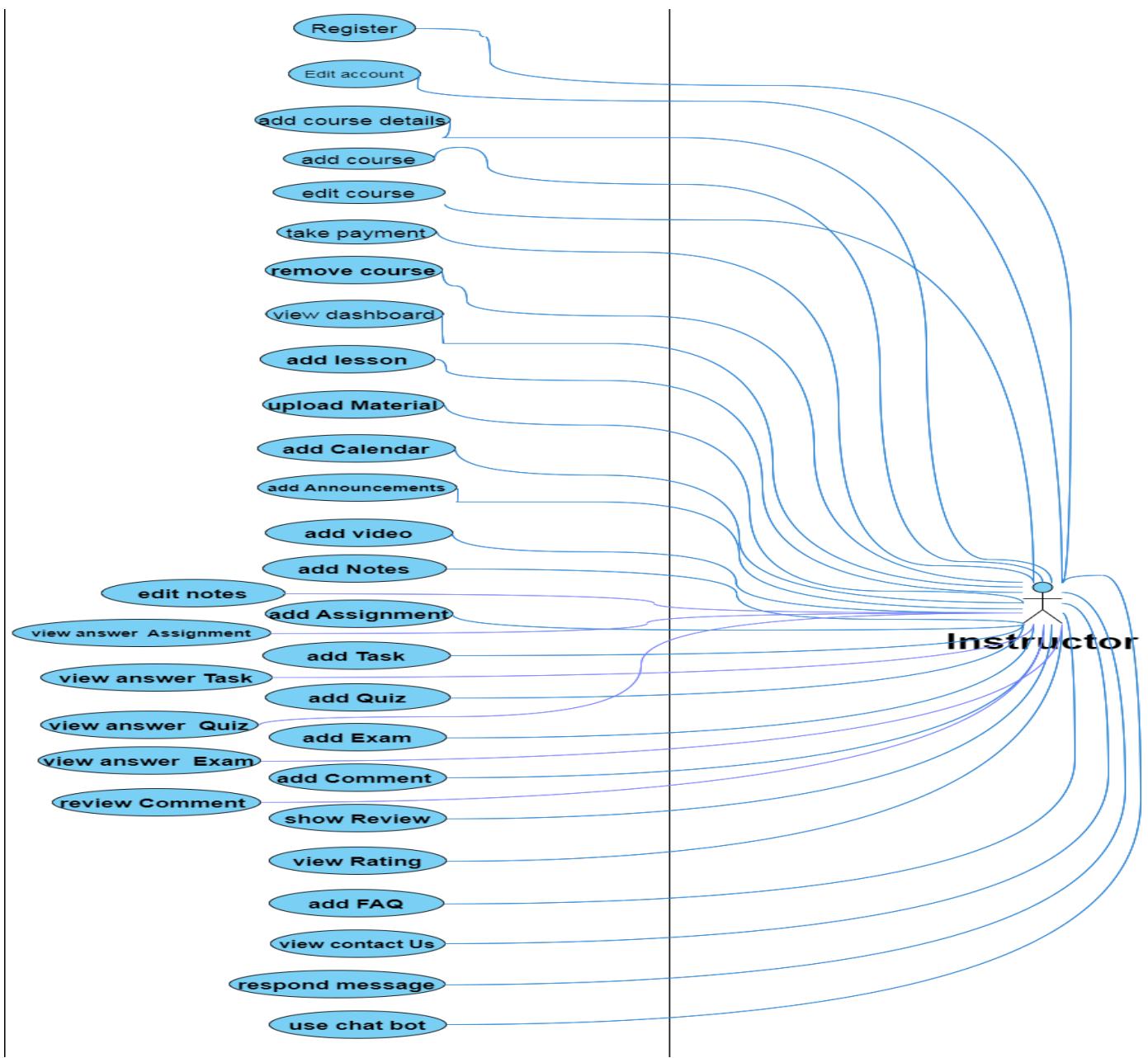


Figure 2.12: Use Case diagram to show interaction between instructor and system

In this diagram show admin's function that he can does in platform first he should register as admin then transfer to dashboard that makes him can add , archive or remove instructor's courses that has bad rating and reviews and can manage calendar and manage categories that arrange it and put the recommended courses in first that has the best priority or put the specific track to study for student can choose from it or leave the freedom to the student to choose courses. He can manage student by add student or remove from platform. he can manage instructor by add instructor or remove from platform and manage calendar that accept to the event put for student or remove and set each student his personal calendar according to the date of taking the course.as shown in figure 2.13

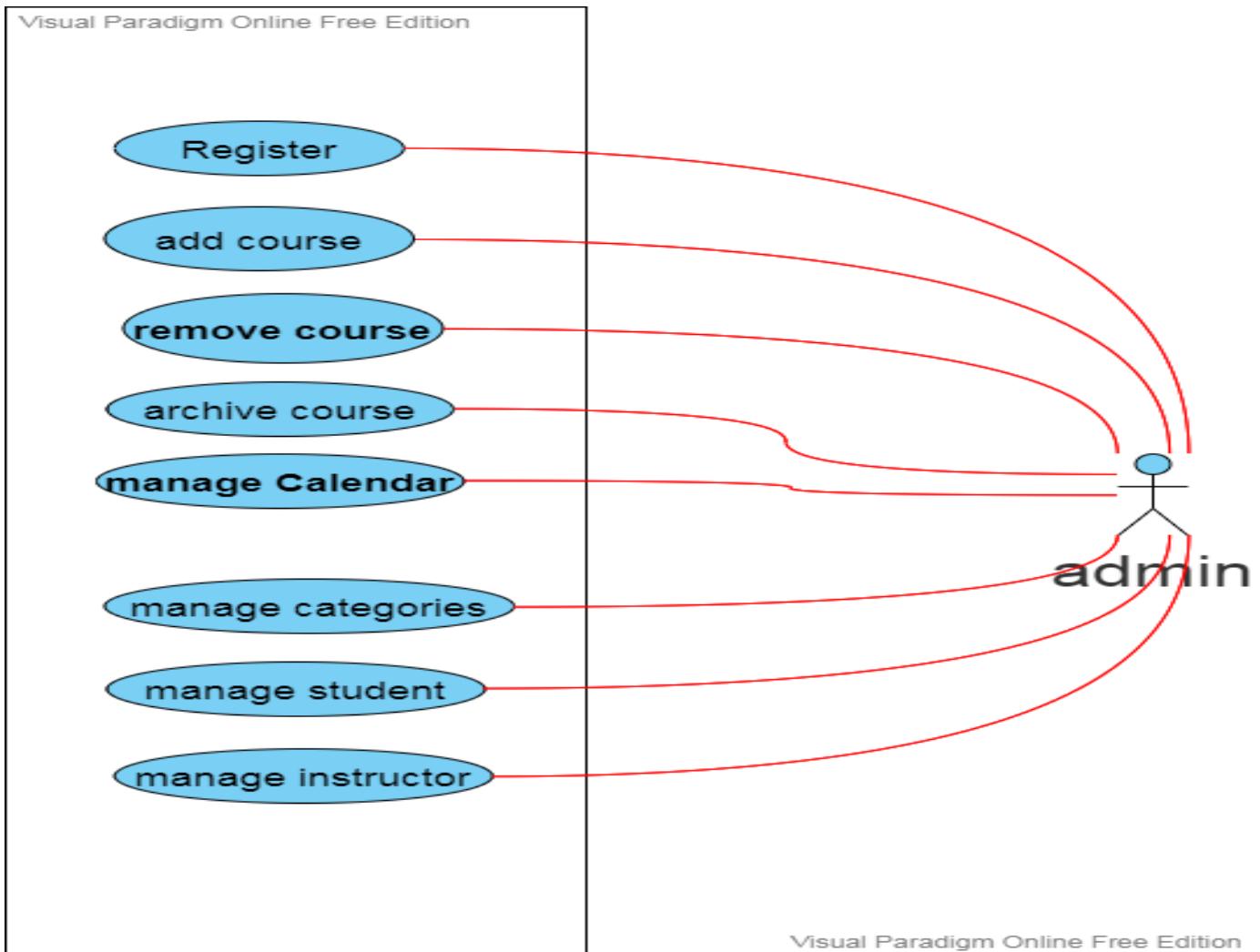


Figure 2.13: Use Case diagram to show interaction between admin and system

This diagram shows how users interact with each other and with the system. In the past, we knew each system and its work with the user, and now we will know the difference in a simplified way between them in dealing with them and with the system. First A guest can see the courses and information about the course, the instructor, the rating, and the reviews left about the courses and add course to cart, but he cannot buy it or see specific details ,and he can submit the deliverables required of him and

he can put rating and comments and ask for help from the system or send messages to the instructor, while the instructor is the one who puts the course and changes it or deletes it or changes in the calendar and puts what is required to be delivered from the student and then correct the deliveries and he can respond to comments and messages While the admin is the one who controls everything, who agrees to put the course on the platform or deletes the course and who allows the student to be in the platform or delete it and allows the instructor to be in the platform or delete it .He controls the calendar for each student and controls the instructor and adds it or deletes it or can pause the course until it is modified by instructor and he who puts the categories that the student can choose from it as shown in figure 2.14

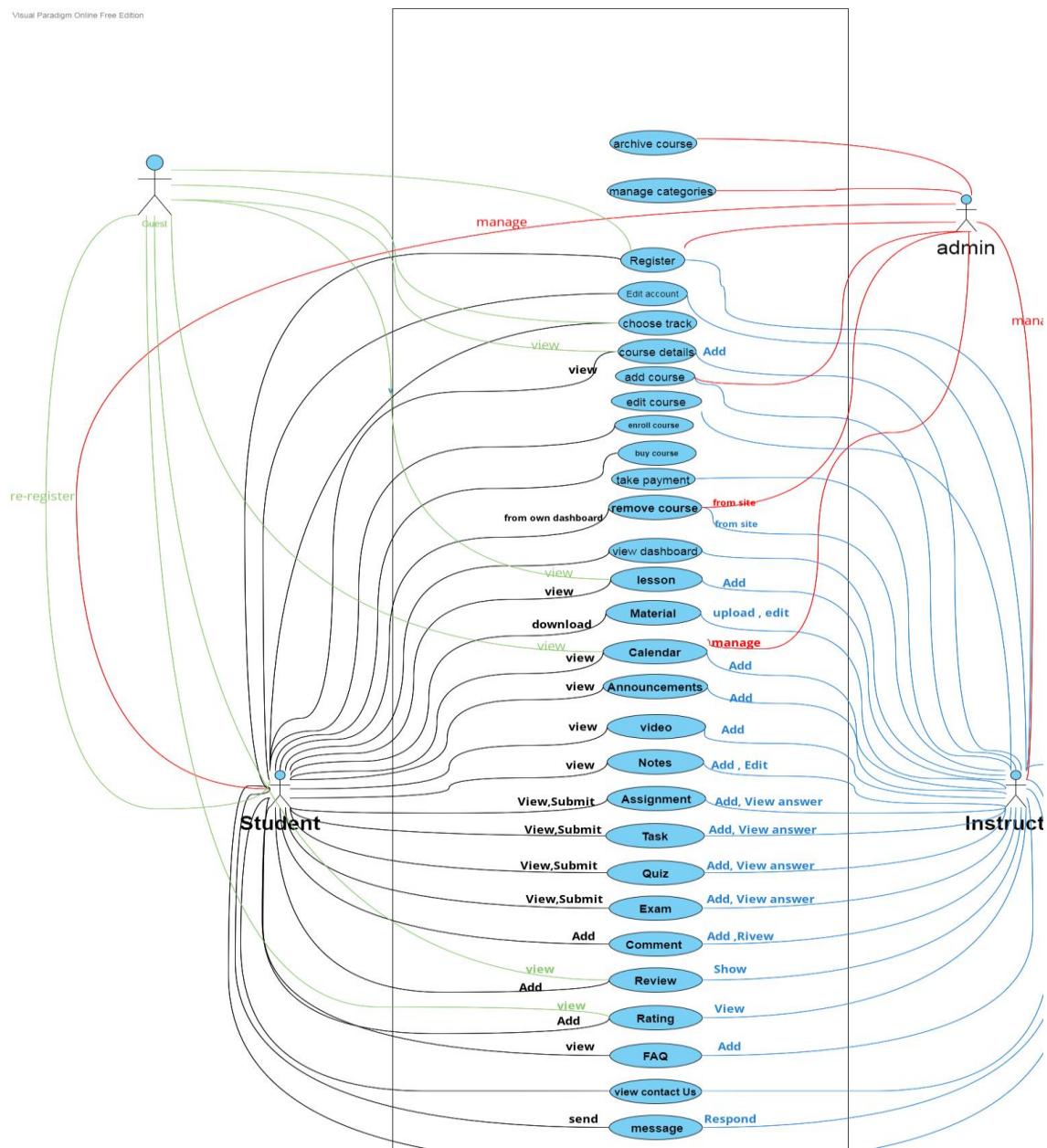


Figure 2.14: UseCase diagram to show interactions between all users and system

## 2.2.4 Sequence Diagram

When the user of any type (student, instructor, admin) press the login button, he/she goes to the login page, then the user enters their email and password, if it is correct, it goes to the database, If the information matches one of the stored data, a message appears that the login has been successful, then displays the home page with all services that match the type of user, if that the input data was incorrect, the error message appears (please use the formal and show the formal) , or in case if the information wasn't match any data in database error message will appears (email or password is not matched)as shown in figure 2.15

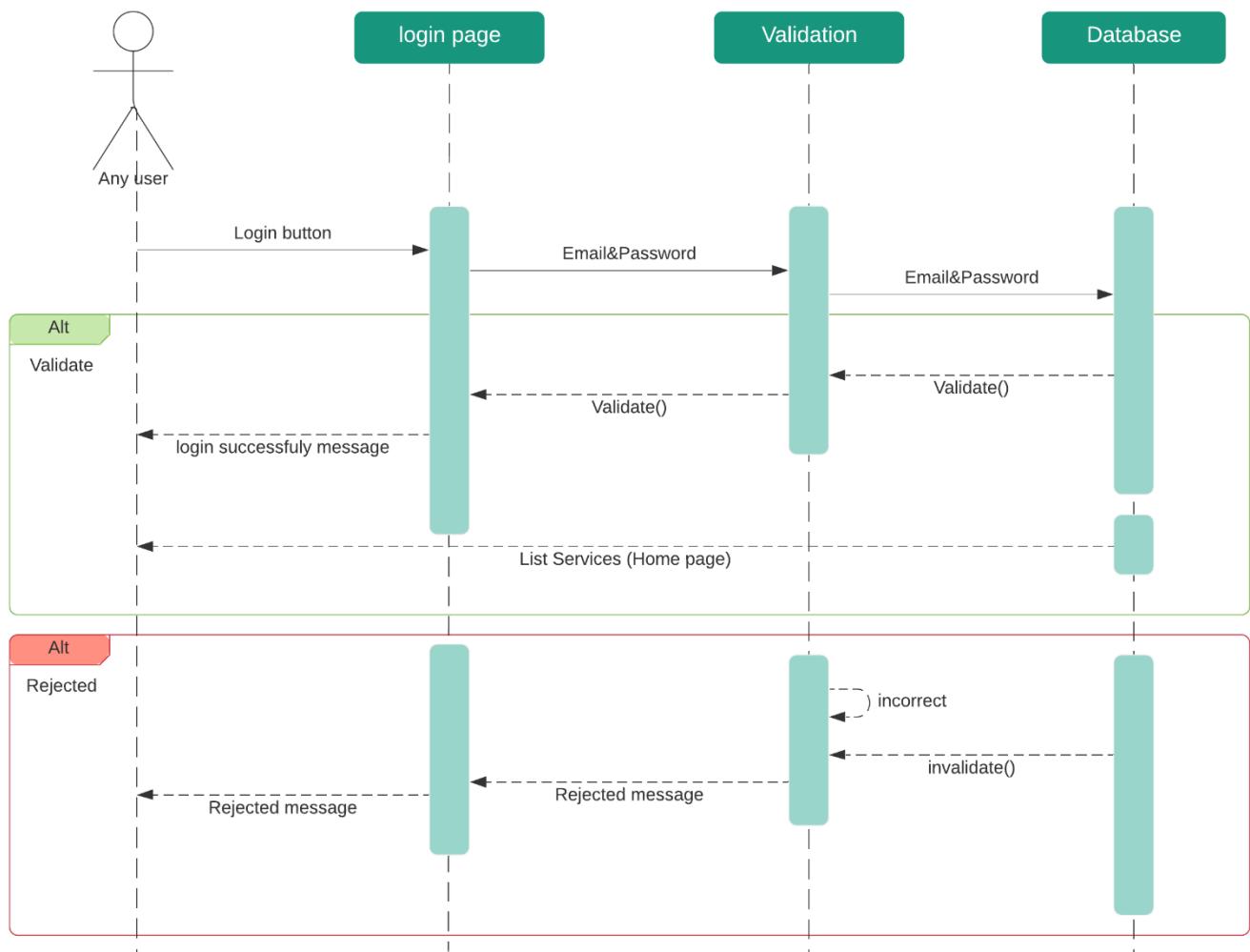


Figure 2.15: Sequence diagram to show User and system

It is a diagram that represents the work of the admin on the site After the login process Admin can access more than one page and control the site When Admin moves to a course management, Admin can add and delete any course Admin can also share or hide any course and if Admin goes to the schedule management through this page, can add and delete a list, and he can also modify the existing list. As for the student management part, through which Admin can add and delete any student And

through the payment page, the admin can follow up on the payments for the courses and also withdraw money.as shown in figure 2.16

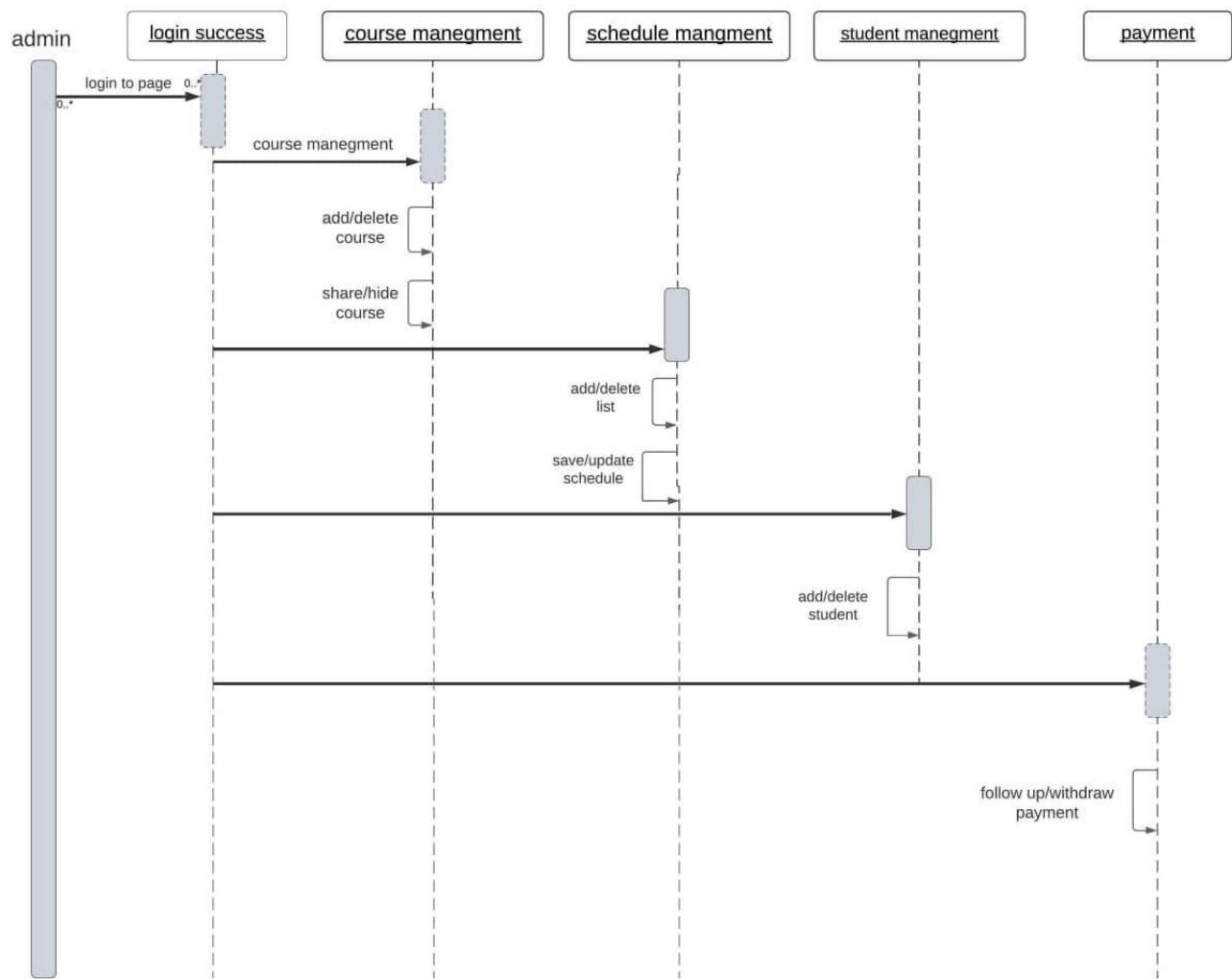


Figure 2.16: Sequence diagram to show admin and system

When the student logs in, he goes directly to the home page and from there he can go to all services (dashboard, courses page, cart, Wishlist page and his profile) If he goes to dashboard he can check his completed courses, announcements, calendar, and tasks. In case he goes to courses page he can view the courses he paid and show his progress for each course, and if he chose course even if he chose it from home page, courses page, result search page or dashboard he will go to the selected course page and from there he can view the course brief, description, content, review and rating and if he has the course he can see his progress, quizzes, tasks and exams. And he can go from there to instructor page to show his profile and see his rating and feedbacks. And if he goes to his profile page he can add, remove, or update his picture, and he can change his Email or password, and he can add a new payment information as shown in figure 2.17

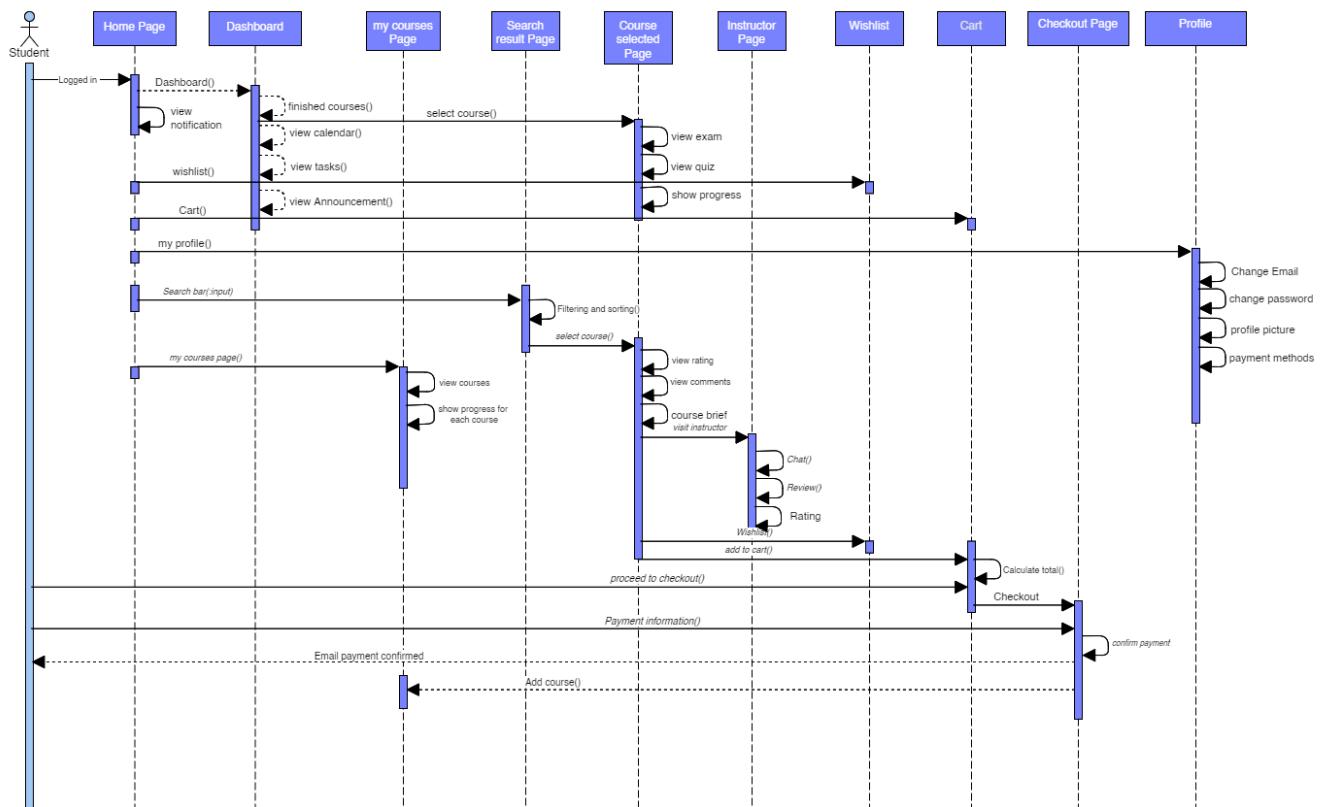


Figure 2.17: Sequence diagram to show student and system

When instructor logs in he goes directly to dashboard from there he can manage his courses he gives. He can add course, edit course, view the reviews and rating, calendar, and take payment. If he wants to add course he will translate into page and from there he can add the videos, course title, lesson, and description then it will create a new course page and he can make exam, assignments, quizzes and add announcements and view answers, comments, and upload materials as shown in figure 2.18

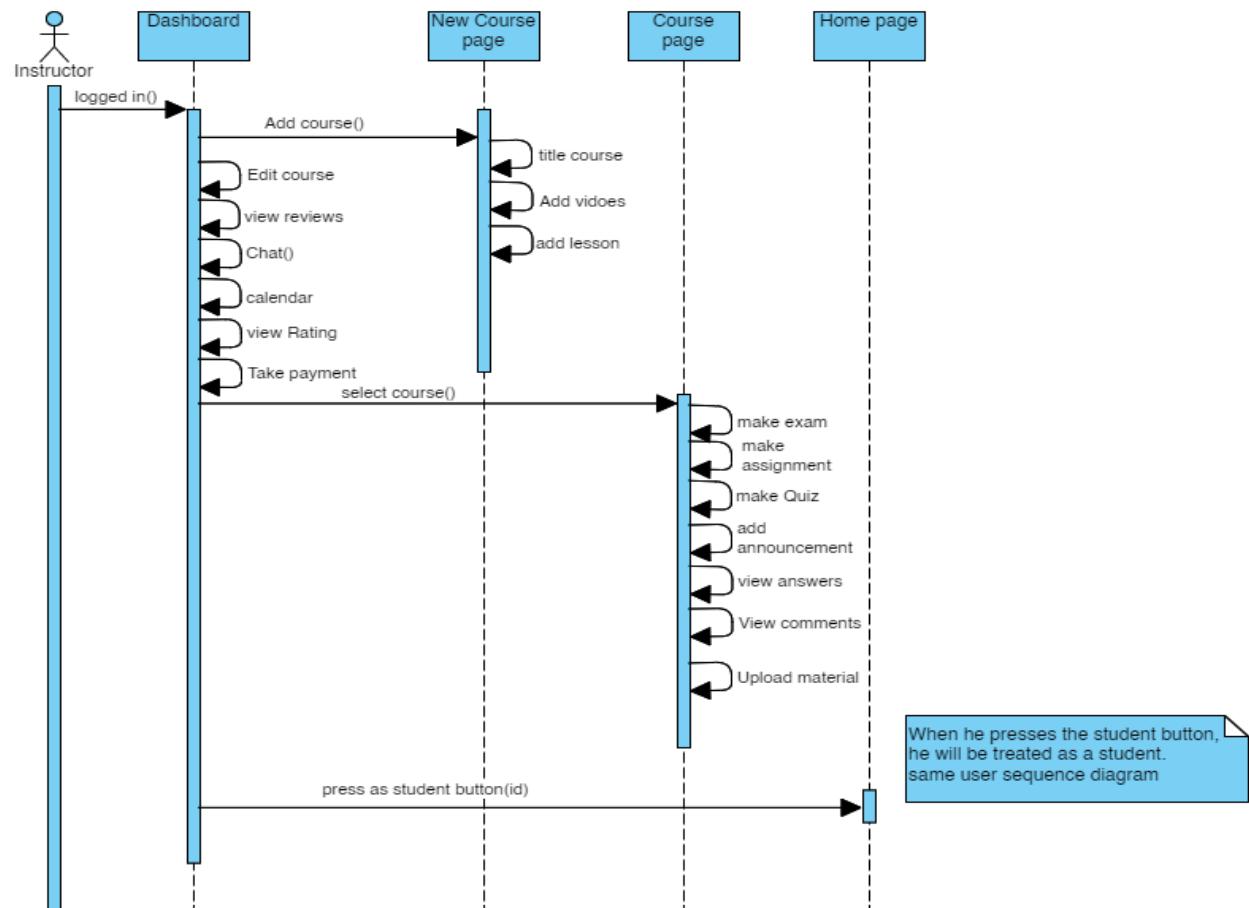


Figure 2.18: Sequence diagram to show student and system

## 2.2.5 ER- Diagram

A database is an organized collection of structured data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

It developed in the early 1960s.

In our platform we will use noncom-relational database (NOSQL) which is type of database, that allows unstructured and semi-structured data to be stored and manipulated in contrast to a relational database, which defines how all data inserted into the database must be composed. NOSQL databases grew popular as web applications became more common and more complex. So, we will use entity–relationship model to describe our platform database

entity–relationship model (ER model) is a high-level data model. it describes data elements, interrelated things of interest in a specific domain of knowledge & composed of entity types which classify the things of interest and specifies relationships that can exist between entities

The diagram shows all system interconnections between different entities that the entity has attributes. The attributes are split into primary key which contains values that uniquely identify each row in a table, string, object refers to other entity that means is foreign key, array means set of data in the same data type, buffer means take photo, The attribute that contains ID is primary key. as shown in figure 2.19

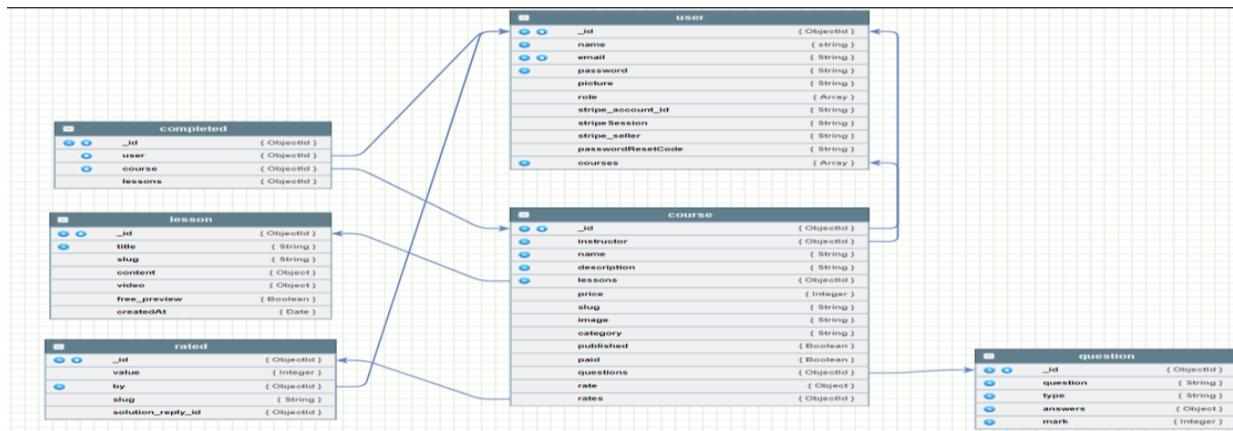


Figure 2.19: ER diagram

The first entity we will describe is user, it has many attributes such as id it is the primary key, email, password, username, tokens, profile picture, age, created at, updated at. It is the parent to student, instructor, admin, that means the attributes of It is also present in children. To enter a child entity, it uses id as shown in figure 2.20

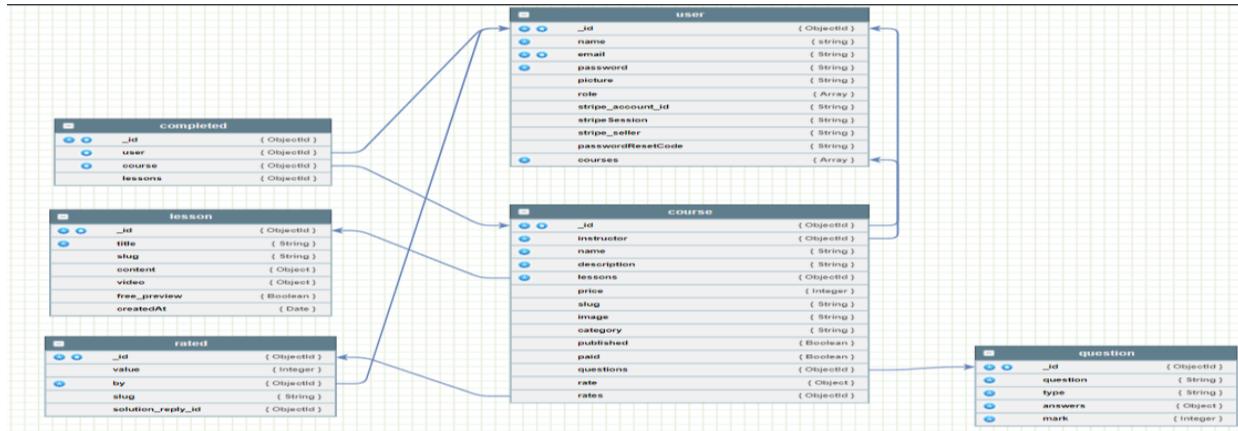


Figure 2.20: ER diagram to user

This entity is used to have all data of course .It has many attributes such as id which is primary key , instructor that type is object means has the group of values of id that his role is instructor in user entity, name, are type of string, lessons are type of object id to lesson entity ,price ,slug, image type of buffer ,categories, published ,paid ,questions are type of object id to question entity , rates are type of object id to rated entity . Id, instructor, name, image, description published should have value. as shown in figure 2.21

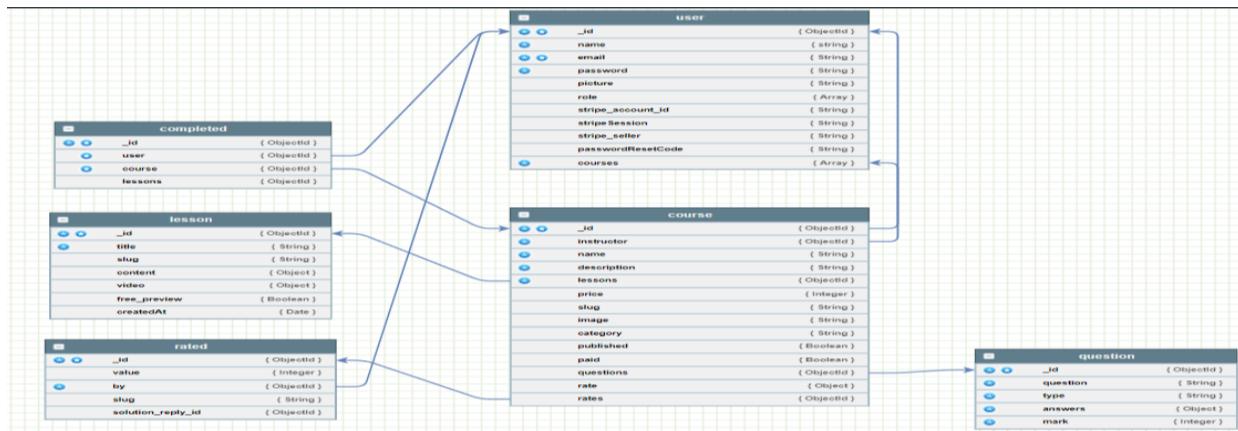


Figure 2.21: ER diagram to course

This entity is used to add lesson to course. It has many attributes such as id is primary key, title of type string, slug of type string, content of type object, video of type object, free preview of type Boolean that refer to if watch or not, created at of type date, id, title, create at should have value. as shown in figure 2.22

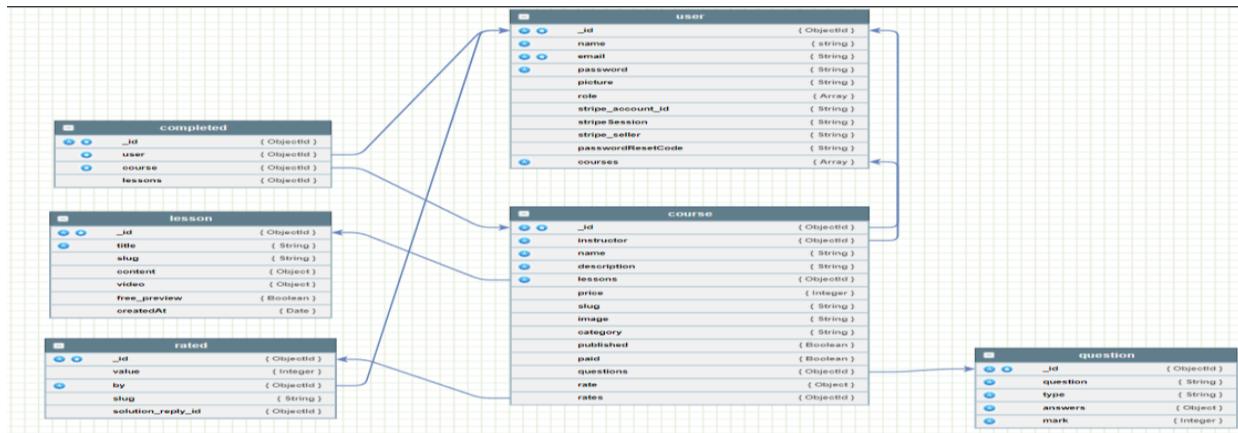


Figure 2.22: ER diagram to lesson

This entity is used to add question to lesson in course by choose type of question then add question then add answer, marks by instructor to specific course. It has many attributes such as id is primary key, question of type string, type of type string, answer of type object, mark of type integer .id should have value. as shown in figure 2.23

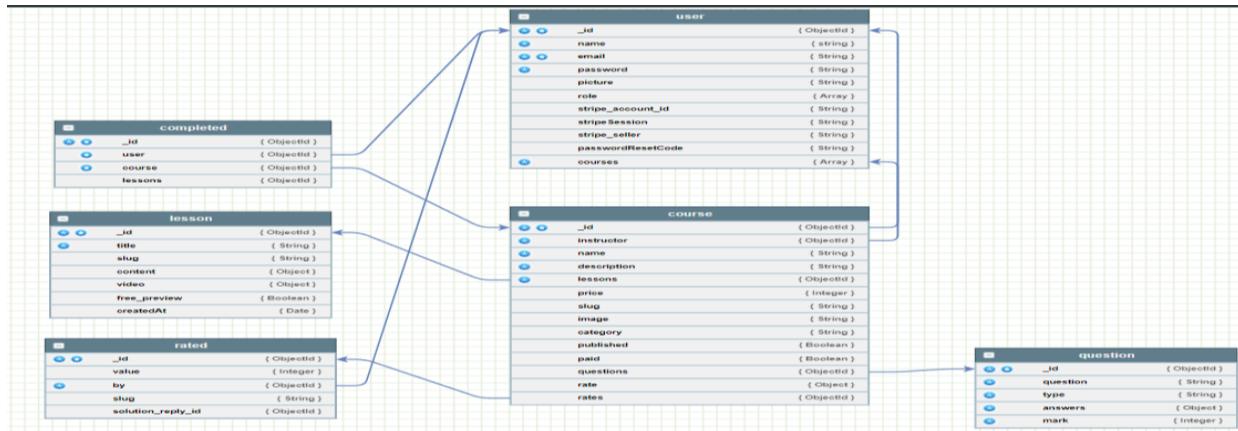


Figure 2.23: ER diagram to question

This entity is used to mark lessons that watch in course. It has many attributes such as id is primary key, user of type object id that refer to user entity, course of type object id that refers to course entity, lessons of type object id that refers to lesson entity .id, user &course should have value. as shown in figure 2.24

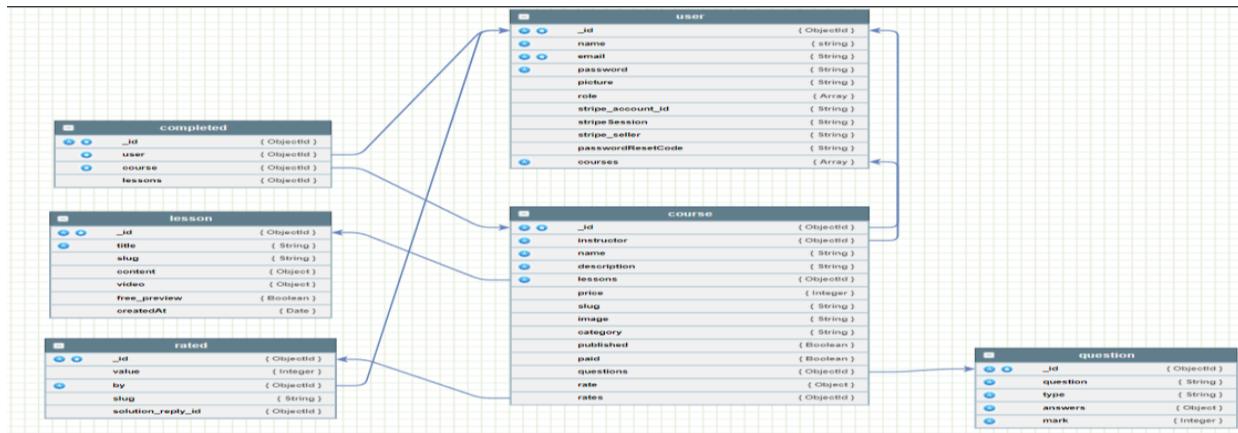


Figure 2.24: ER diagram to completed

This entity is used to rate course by putting value by student and refer to specific course. It has many attributes such as id is primary key, by of type object id that refer to user entity, slug, value of type number, solution-reply-id of type object id that refer to course entity, instructor id of type object .id should have value. as shown in figure 2.25

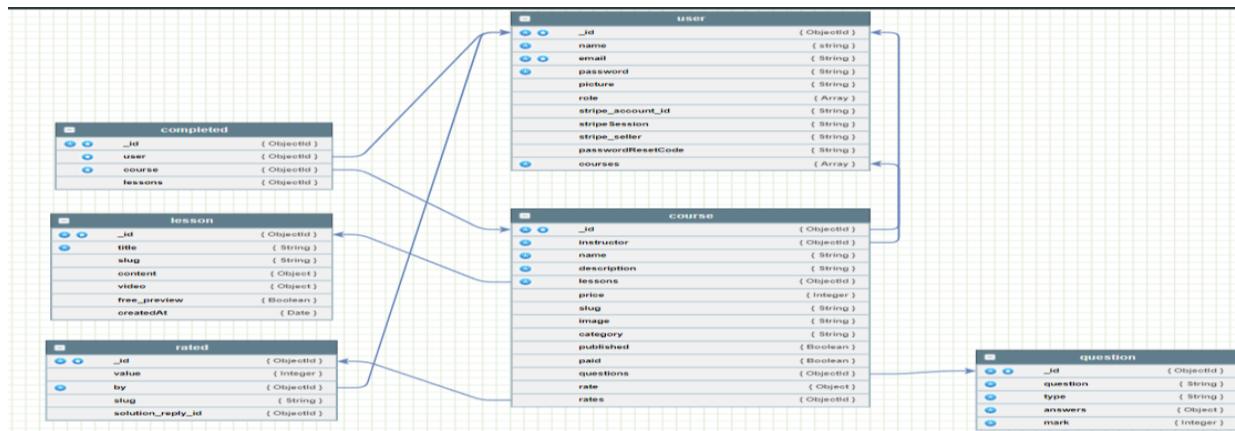


Figure 2.25: ER diagram to rated

# CHAPTER3: SYSTEM SERVICES

in this chapter we will discuss about service that system do for (admin, instructor, student) and tools that help us to build our system

## 3.1 Services

### 3.1.1 The Instructor

1. Register –
2. Edit Account –
3. View Dashboard –
4. Add Course –
5. Edit Course
6. Remove Course –
7. Add Course Details –
8. Take Payment –
9. Add Lessons -
10. Add Videos –
11. Upload Material –
12. Add Calendar –
13. Add Announcement –
14. Add Assignment –
15. Review Answer Assignment –
16. Add Notes -
17. Edit Notes -
18. Add Questions Task -
19. Review Answer Task -
20. Add Quiz –
21. Review Answer Quiz -
22. Add Exam -
23. Review Answer Exam -
24. Review Comment -
25. Add Comment
26. Show Reviews
27. View Rating
28. Add FAQ
29. Respond Message
30. Use Chatbot
31. View Contact Us

### 3.1.2 The Students

1. Register
2. Edit Account
3. Choose Track
4. View Course Details
5. Enroll Course
6. Remove Course
7. Buy Course
8. View Dashboard
9. View Lessons
10. Download Material
11. View Calendar
12. View Announcements
13. View Videos
14. View Notes
15. View Assignment
16. Submit Assignment
17. view Task
18. Submit Task
19. View Quiz
20. Submit Quiz
21. View Exam
22. Submit Exam
23. Add Comment
24. Add Review
25. Rating
26. View FAQ
27. View Contact Us
28. Send Message
29. Use Chatbot

### 3.1.3 The Admin Only

1. Register
2. Add Course
3. Remove Course
4. Edit Course
5. Archive Course
6. Manage Class
7. Manage Categories

- 8.Manage Events Calendar
- 9.Manage Students
- 10.Add Student
- 11.remove Student

## 3.2 METHODOLOGY

### 3.2.1 Software Development Tools

#### VS Code

- The vs code program is the program that will be used by developers to write the code.
- This program was chosen because it is the best in terms of speed and makes writing code easier for developers because we can install a lot of extensions and tools for it.
- We can write [HTML, CSS, JS, React, Node JS, etc.].

[3] [VS Code – The Story and Technology Behind One of the World's Most Popular Desktop Apps for Developers | Tower Blog \(git-tower.com\)](#)

#### CSS

- **Styling:** is a simple mechanism for adding style (e.g., fonts, colors, spacing) to Web documents.
- **SASS:** a preprocessor (or pre-compiler) is a program that processes its input data to produce output that is used as input to another program. The output is said to be a preprocessed form of the input data, which is often used by some subsequent programs like compilers. is an extension of CSS that enables you to use things like variables, nested rules, inline imports, and more. It also helps to keep things organized and allows you to create style sheets faster.

[4] [A brief history of CSS until 2016 \(w3.org\)](#)

#### React

- **React:** React. JS is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications. It's used for handling the view layer for web and mobile apps. React also allows us to create reusable UI components.
- **React Router:** it's a fully featured client and server-side routing library for React, it runs anywhere, runs on the web, on the server with node.js, and on React Native for mobile apps
- **React Redux:** work with React's component model to define how to extract the values your component needs from Redux, and your component updates automatically as needed.
- **Styled Component:** we will use the styled component to avoid redundancy, it gives us the flexibility to write the component only one time then we can use it everywhere in our App, just write the name of the component.

- **React Hooks:** The Hook is a new feature in React that lets us use the state and other React features without writing a lot of code (Class Component).

[5] <https://reactjs.org/tutorial/tutorial.html>

## Next JS

- **Next JS:** Next.js is a JavaScript framework that enables you to build superfast and extremely user-friendly static websites, as well as web applications using React. Next.js allows the building of hybrid applications that contain both server-rendered and statically generated pages. Let's list the main features that make next.js so popular and why many developers consider it the best library right now.
- **Next JS Render:** One of the most important, if not the most important, next.js feature is the different types of rendering that you can do in your application. By default, next JS is using Server-Side Rendering (SSR) and at the same time can be also a great Static Site Generator (SSG).
- [6] [What is Next.js? | Learn Next.js \(nextjs.org\)](#)
- **Next Performance:** Next.js frees the browser from loading and working with all the JavaScript code at once, thus increasing such a metric as a time to first draw (TTFD). It measures the amount of time needed for the user to see the very first content on their screen and should ideally be below 1 second. This factor will improve both User Experience and SEO.
- **React Eco-System:** Next.js is part of a large React ecosystem, with all its benefits in the form of several best practitioners, community help, useful libraries, and already solved corner cases. Next.js was also developed specifically for React, so it will be easy for you to embed this framework into your existing application.

## Node Js

- **Node Js:** It is used for server-side programming, and primarily deployed for non-blocking, event-driven servers, such as traditional websites and back-end API services, but was originally designed with real-time, push-based architectures in mind. Every browser has its own version of a JS engine and node.
- **Express JS:** flexible Node.js framework that provides a for web and mobile applications with a myriad of HTTP utility methods and middleware at your disposal, creating a robust API is quick and easy.
- **Handlebars.js:** it is a simple Template Engine language that uses a template and an input object to generate HTML or other text formats
- **Strip Payment:** an easy tool to deal with Payment in any website or mobile application.
- **Contact APIs:** is an application programming interface that deals with the Database.
- **MVC Pattern:** stands for Model, View, Controller, it's a pattern used to separate the code in different files that make the code more readable and maintainable.
- **Authentication:** we will use Authentication with firebase to make Authentication with Google or Facebook or Microsoft Accounts.

[7] [11 Years Of Node.JS: Timeline & Significant Contributions / Habr](#)

## **MongoDB**

- **MongoDB:** is a document-oriented database that stores data in JSON-like documents with dynamic schema. It means you can store your records without worrying about the data structure such as the number of fields or types of fields to store values.
- **Mangoes:** Library to work with Node JS and Mongo, it gives us an easy system for modeling our data and modeling things like Users and Courses, etc.

[8] <https://www.mongodb.com/nosql-explained>

## **Amazon Web Services (AWS)**

- o **AWS:** secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.
- AWS:** In simple words AWS allows you to do the following things-
1. Running web and application servers in the cloud to host dynamic websites.
  2. Securely store all your files on the cloud so you can access them from anywhere.
  3. Using managed databases like MySQL, PostgreSQL, Oracle, or SQL Server to store information.
  4. Deliver static and dynamic files quickly around the world using a Content Delivery Network (CDN).
  5. Send bulk email to your customers and much more.

[9] [A Brief History Of AWS - And How Computing Has Changed \(digitalcloud.training\)](https://digitalcloud.training/)

### **3.2.2 Machine Learning**

## **Python**

[10] [History of Python - GeeksforGeeks](https://www.geeksforgeeks.org/history-of-python/)

## **Mathematics**

- o **Probability:** Probability theory is very much helpful for making the prediction, Estimates, and predictions form an important part of Data Science.
- o **Statistics:** With the help of statistical methods, we make estimates for further analysis. Thus, statistical methods are largely dependent on the theory of probability.

## **Data analysis**

- o **NumPy:** you can speed up your workflow, and interface with other packages in the Python ecosystem

- **Pandas:** used for data manipulation and analysis. It provides many functions and methods to speed up the data analysis process. Pandas is built on top of the NumPy package.

[11] [NumPy and Pandas Tutorial - Data Analysis with Python | CloudxLab Blog](#)

### **Data visualization**

- **Matplotlib:** is a 2-D plotting library that helps in visualizing figures.
- **Seaborn:** used to make default matplotlib plots look nicer, and also introduces some additional plot types.

[12] [A Brief History of Data Visualization \(dundas.com\)](#)

### **Data processing**

- **Data Cleaning:** the process of identifying the incorrect, incomplete, inaccurate, irrelevant, or missing parts of the data and then modifying, replacing, or deleting them according to the necessity
- Work with Missing data
- Work with Categorical data
- Split data to Train and Test Sets

[13] [Data processing technology history - Dataconomy](#)

### **Natural Language Toolkit**

- work with human language data for applying in statistical natural language processing (NLP). It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning

[14] [A Brief History of Natural Language Processing \(NLP\) - DATAVERSITY](#)

### **Cosine Similarity**

- applications such as data mining and information retrieval. ... This allows for a Cosine Similarity measurement to distinguish and compare documents to each other based upon their similarities and overlap of subject matter

[15] [Cosine Similarity - an overview | ScienceDirect Topics](#)

### **TF-IDF**

- The Tf-idf Vectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new documents

[16] [Understanding TF-ID: A Simple Introduction \(monkeylearn.com\)](#)

### **Flask**

- is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be

some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.

[17] [What is Flask? - Flask: Develop Web Applications in Python \(educative.io\)](#)

## **Streamlit**

- streamlit is an open-source python framework for building web apps for Machine Learning and Data Science. We can instantly develop web apps and deploy them easily using stream lit. stream lit allows you to write an app the same way you write a python code. stream lit makes it seamless to work on the interactive loop of coding and viewing results in the web app.

The framework fills a vital void between data scientists who want to develop a new analytics widget or app and the data engineering typically required to deploy these at scale.

Data scientists can build web apps to access and explore machine-learning models, advanced algorithms, and complex data types without having to master back-end data engineering tasks.

[18] [Introduction to Streamlit \(latentview.com\)](#)

## CHAPTER4: DEMO

when we started to create this platform, we were thinking about how we can solve the problems that we faced in a lot of other learning platforms, also we think about how we can make this platform unique, and how we can implement some important features that we had never seen before in other platforms that we used in our platform, we have many roles for the user, such as Guest, Student, Instructor, and Admin .we start making the platform open for everyone who needs to discover the content and how our platforms work, so for that, we create the Guest role.

The start points or the start page of the platform is the index page, so whether you signed in or not or are an existing user or not, you can discover our platform once you have the live link for the platform. as shown in figure 4.1

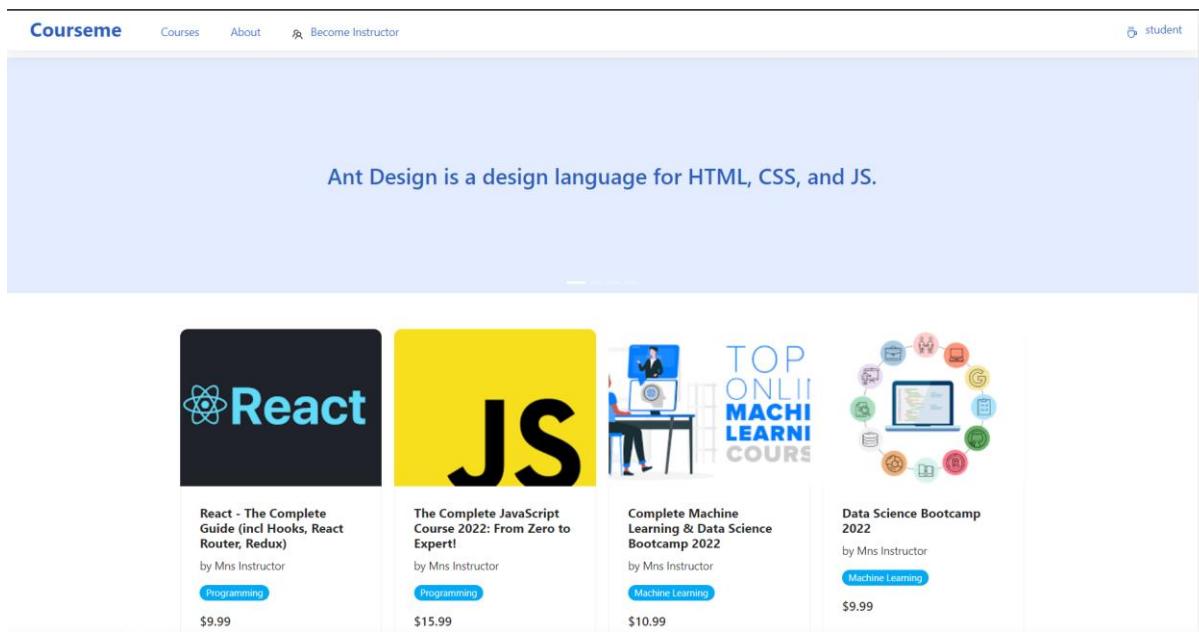


Figure 4.1: Index page

when you open the index page you will see some courses in the center of the page.

as shown in figure 4.2

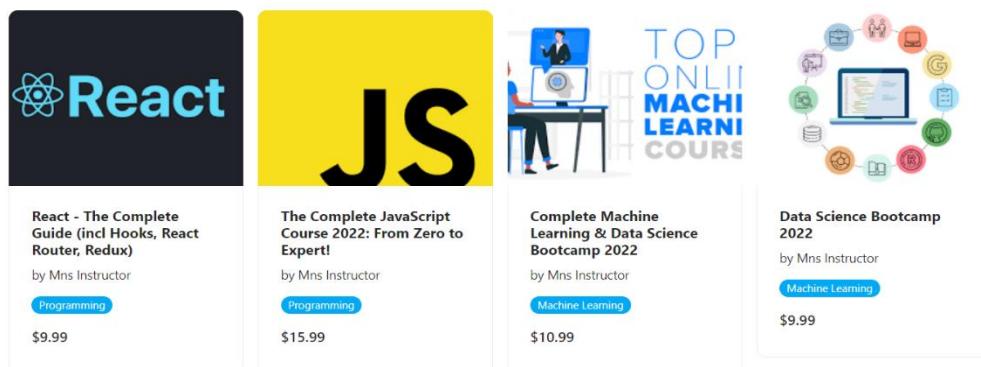


Figure 4.2: Courses in index page

Actually, you can discover all the titles and the description of these courses but if you need to enroll in any course you need to log in first then you can enroll in any course you need.

On our platform, we have two types of courses. as shown in figure 4.3

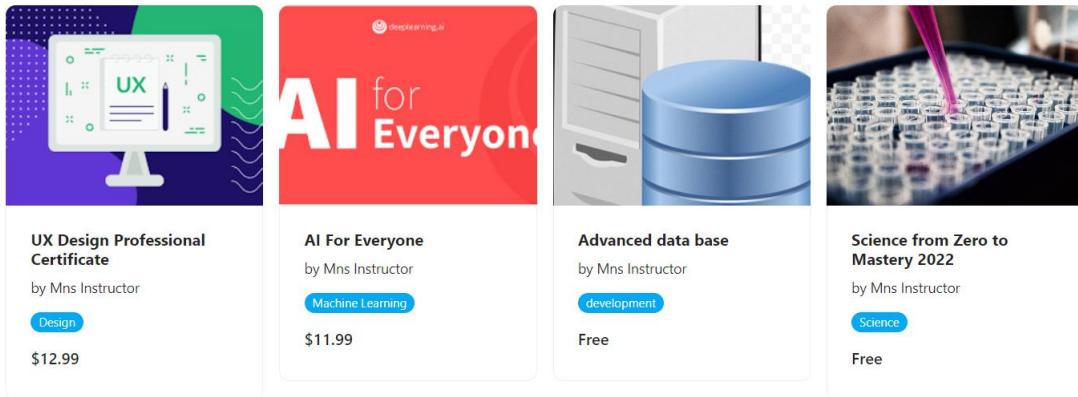


Figure 4.3: Paid and Free Courses in index page

the first type is paid courses, and the second type is the free courses.in the first type (paid courses) it's very easy to enroll in any course and because we use Stripe payment you can use your bank card or any other online payment to take any course just in a few seconds, but as I told you first you need to log in before other any actions so, to log in you can click the login link at the top of the page . as shown in figure 4.4

The login page features a central "Don't have an account?" section with a cartoon illustration of a person sitting at a desk with a laptop and a plant. Below this is a "Join Now" button. To the left is a "Login" form with fields for email and password, and links for forgot password and social media logins (Google, LinkedIn, GitHub). To the right is a sidebar with "Login" and "Register" buttons.

**Courseme**   Courses   About     Login   Register

**Login**

email

password

Forgot password ?

Login

Or continue with email

G   LinkedIn   GitHub

Don't have an account?

Join Now

**Courseme**  
@ 2022 Courseme

About us  
Membership

Privacy Policy  
Contact Us

Terms  
Become a Partner

Figure 4.4: Login page

if you don't have an account, you will have the option to create a new account with your email. as shown in figure 4.5

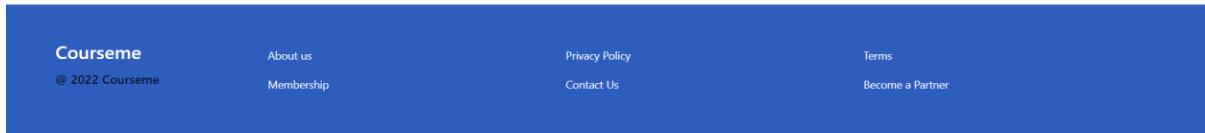
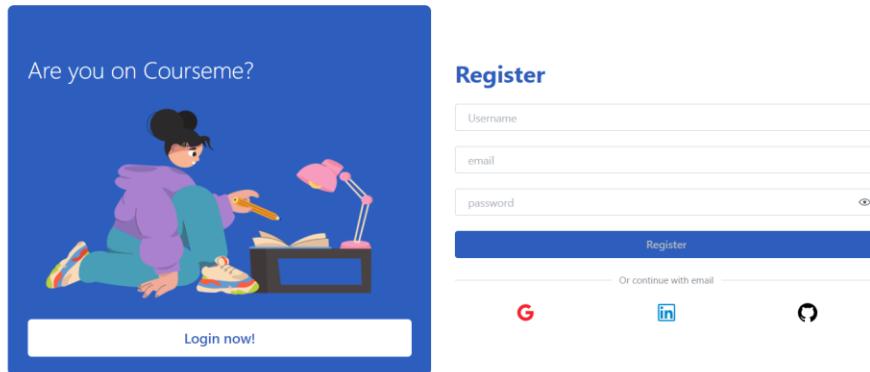


Figure 4.5: Register page

also, you can register with your existing account on Google, LinkedIn, or GitHub very easy, just click on the icon and that's it. as shown in figure 4.6

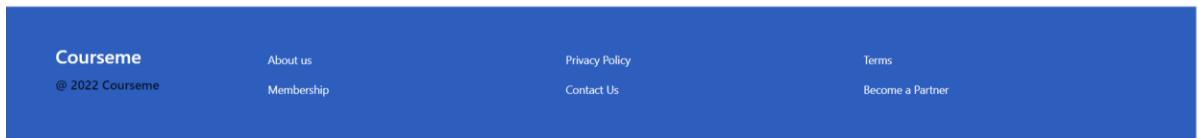
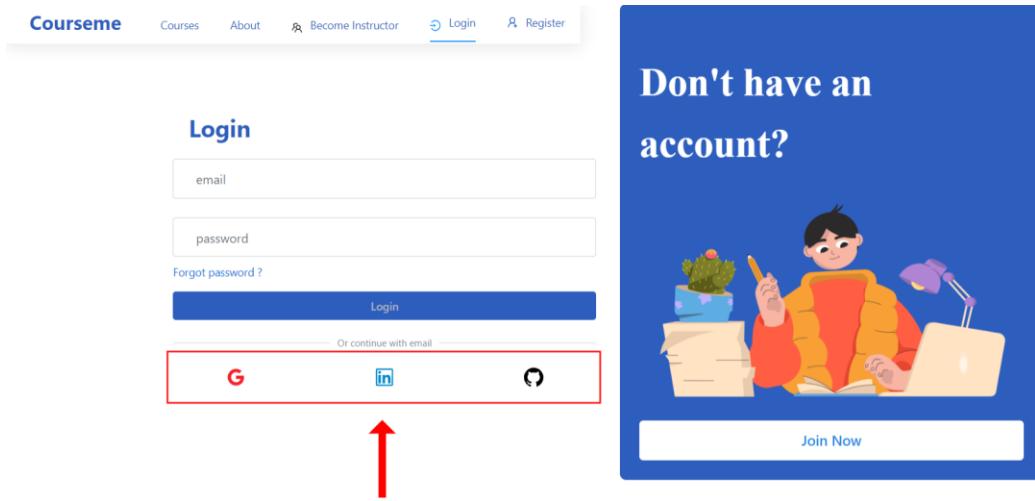


Figure 4.6: Register page with existing accounts

Once you log in or create your account, you will be redirected to the main page of the website, but now you have more options, such as you can enroll in any course. as shown in figure 4.7

The screenshot shows a course page for 'JavaScript'. At the top, there's a navigation bar with 'Courseme', 'Courses', 'About', 'Create Course', 'Instructor', and 'Ioulahkareem-1'. The main content area features a large yellow square with the letters 'JS' in black. Below it is a red button labeled 'Enroll'. To the left of the main content, there's a sidebar with a 'Programming' category, 'Created by Mns Instructor', 'Last updated 6/16/2022', and a rating of '\$15.99 (0 of 5 ⭐)'. Below the main content, there's a section for '5 Lessons' with numbered circles from 1 to 4.

Figure 4.7: Publish course by instructor

also, you can become an instructor if you need to publish a course. as shown in figure 4.8

The screenshot shows the same website interface as Figure 4.7, but with a red arrow pointing to the 'Become Instructor' button in the top navigation bar. Below the navigation bar, there's a large blue button labeled 'student'. The main content area displays four course cards: 'React - The Complete Guide (incl Hooks, React Router, Redux)' by Mns Instructor (\$9.99), 'The Complete JavaScript Course 2022: From Zero to Expert!' by Mns Instructor (\$15.99), 'Complete Machine Learning & Data Science Bootcamp 2022' by Mns Instructor (\$10.99), and 'Data Science Bootcamp 2022' by Mns Instructor (\$9.99). Each card has a small 'Machine Learning' tag.

Figure 4.8: instructor privilege

Generally, we have a lot of pages on the website, these pages depend on your role, for example as a Gest you only can access the admin page (index page) as a user you will have some pages man page sources page, about page, become instructor page

as Admin you will have an extra page (Dashboard) also the instructor has a dashboard but they are different in the permissions and access I will start with the student then I will the extra pages of the other roles

## Student

The student has access to some pages

### Main page

On all pages, you will see the top menu or navbar that has all links that you need to redirect to. . as shown in figure 4.9



Figure 4.9: Menu (navbar)

At the bottom of all pages, you will see the footer that contains some links about the platform that the user may need. as shown in figure 4.10

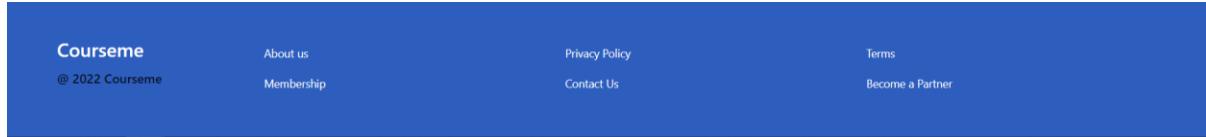


Figure 4.10: Footer

Under the navbar, you will see the slider that has our recent courses and some news about our platform. as shown in figure 4.11

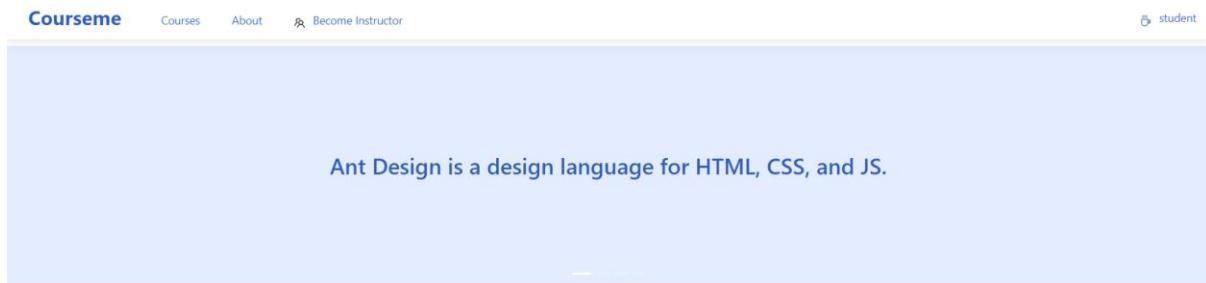


Figure 4.11: Slider

Then the student can see all courses discover them, and enroll in any course if he wants. as shown in figure 4.12

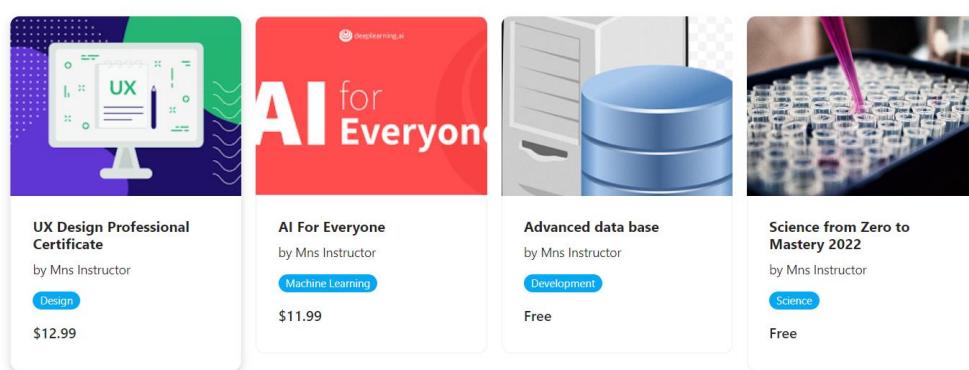


Figure 4.12: Discover courses

## Courses page

You will have access to all courses on this page and you can filter the courses depending on the option that needs . as shown in figure 4.13

**Courseme** Courses About Create Course

Filter By Category All Newest Low to high Apply

Creating 3D environments in Blender by dived larmy Created At: 2022-06-23 <b>Design</b> Free (0 of 5 ⭐)	The Complete Digital Marketing Course - 12 Courses in 1 by nada-1 Created At: 2022-06-23 <b>Digital Marketing</b> Free (0 of 5 ⭐)	Microsoft Excel 2013 Advanced, Online Excel Training Course by nada-1 Created At: 2022-06-23 <b>Software</b> Free (0 of 5 ⭐)	Java 8 Functional Programming: Lambda Expressions Quickly by nada-1 Created At: 2022-06-23 <b>Development</b> Free (0 of 5 ⭐)
PHP CRUD Application by nada-1 Created At: 2022-06-23 <b>Web Development</b> Free (0 of 5 ⭐)	Git & GitHub by nada-1 Created At: 2022-06-23 <b>Software Development</b> Free (0 of 5 ⭐)	RESUME CV by nada-1 Created At: 2022-06-23 <b>Business</b> Free (0 of 5 ⭐)	

Figure 4.13: Filter courses

## About page

On this page, you can read some information about the platform and the students. as shown in figure 4.14

**Courseme** Courses About Create Course Instructor loulahkareem-1

**Our platform**

for people of a variety of ages containing a lot of categories & fields such as development, designing, finance, etc. we want to reach the most number of people possible and help them reach their goals. We are planning to make an easy-to-use website for people of different ages. Courses are going to be divided into different parts (weeks). Tasks are going to be implemented between videos in the week's section, Quizzes will be made between weeks if applicable by the professor

**OUR MISSION**

**Our students**

a community of global learners united in a shared goal of uplift and transformation. Our unique learning model enables an unprecedented degree of engagement with our students, and we are with them through every step of their learning journey—from the first moment a marketing team member might answer a question on Facebook, to the penultimate moment when a career team member receives word that a graduate has gotten a new job. Our mantra is Students First, and this is the light that guides us as we continue our mission to bring the highest quality learning

Figure 4.14: Information about the platform

## Be instructor page

If you need to be an instructor you can do that but first, you need to fill up some information about you to be an instructor. as shown in figure 4.15

**Courseme** Courses About Become Instructor Login Register

**Become Instructor**

**Setup payout to publish courses on Courseme**

Courseme partners with stripe to transfer earnings to your bank account

Payout Setup

You will be redirected to stripe to complete onboarding process.

**Courseme** © 2022 Courseme

About us Membership Privacy Policy Contact Us Terms Become a Partner

Figure 4.15: Information need to fill to be a constructor

## User Dashboard

here you will see the courses that have enrolled in before. as shown in figure 4.16

The screenshot shows the 'User dashboard' page of a learning platform. At the top, there's a navigation bar with 'Courseme', 'Courses', 'About', 'Create Course', 'Instructor' (with a dropdown for 'loulahkareem-1'), and a search bar. Below the header, a large title 'User dashboard' is centered. A list of six courses is displayed, each with a thumbnail, title, number of lessons, and author information. To the right of each course card is a blue circular icon with a white letter 'P'.

Course Title	Author	Lessons
Data Science from Zero to Mastery 2022	By Mina Instructor	5 lessons
English Grammar tenses & structures	By Mina Instructor	5 lessons
C# Problem solving	By nia	5 lessons
Java 8 Functional Programming: Lambda Expressions Quickly	By nia-1	5 lessons
Microsoft Excel 2013 Advanced. Online Excel Training Course	By mouda-1	5 lessons
C++	By loulahkareem-1	5 lessons

At the bottom of the page is a dark blue footer bar with links: 'Courseme', 'About us', 'Membership', 'Privacy Policy', 'Contact Us', 'Terms', and 'Become a Partner'.

Figure 4.16: User dashboard

## Course Page

On the course page you will access all lectures of the course and at the end of every lecture you may take a quiz, if you take this quiz, you will have your grade once you finish the quiz.

The screenshot shows the 'Course Page' for 'Chapter 1'. The left sidebar lists 'Lessons' numbered 0 to 5, with 'Chapter 1' currently selected. The main content area shows a video player with a thumbnail of three people in lab coats and surgical masks. The video player controls include a play button, a progress bar showing '0:00 / 0:29', and a full-screen button. Below the video, a red text message reads 'No Quiz for this lesson'. At the top right of the video player is a button labeled 'Mark as incomplete'.

Figure 4.17: Course page

## Admin Dashboard Page

The screenshot shows the Admin Dashboard for Courseme, a platform for managing courses. The dashboard features a header with the title "Admin Dashboard | Courses Manager" and navigation links for "Courses Manager", "Students Manager", and "Instructors Manager". The main content area displays a table of course listings.

Cover	Course Name	Instructor Name	Price	Categories	Description	Joined Date	Action
	Creating 3D environments in Blender	david larry	Free	DESIGN	<p>Creating stunning unique environments. Organize your workflow to make large environment scenes. More than 250 unique 2K / 4K textures. Lots of medieval reference photos E-book: Old Masters Unveiled ( 250 pages) All scenes assets included. Lots of reference photos and 6 characters to fill up your scene. Full course documentation for the Blender basics chapter. Baking your models and Importing in Unity 3D for games!</p>	2022-06-23	
	Python for Data Science and Machine Learning Bootcamp	david larry	9.99	MACHINE LEARNING	<p>Use Python for Data Science and Machine Learning. Use Spark for Big Data Analysis Implement Machine Learning Algorithms Learn to use NumPy for Numerical Data Learn to use Pandas for Data Analysis Learn to use Matplotlib for Python Plotting Learn to use Seaborn for Data visualization Learn to use TensorFlow for Deep Learning Learn to use PyTorch for Machine Learning Tasks K-Means Clustering Logistic Regression Linear Regression Random Forest and Decision Trees Natural Language Processing and Spam Filters Neural Networks Support Vector Machines</p>	2022-06-23	
	Become a Web Developer from Scratch	david larry	9.99	DEVELOPMENT	<p>"The Complete Freelancer Guide" e-Book Certificate of Completion 30 Free Responsive Templates Awesome Portfolio Website Make REAL life web apps with our final projects Coding Exercises and Challenges PDF Materials and Codes Carousels, Accordions, Off-Canvas, etc... Updated w/ Latest Techs Massive Q&amp;A Community All Lectures and Projects Source Codes Improvement Suggestions for Course 110+ functional forms package Learn to get free hosting forever</p>	2022-06-23	
	Complete Blender Creator: Learn 3D Modelling for Beginners	Ahmed ragab	9.99	DESIGN	<p>This course is continually updated in response to student suggestions - it has been completely updated to Blender 2.8 or later. Blender is a famous platform which enables you to make Animation movies, video games, which is used in game engine, 3D printer, or other software. Here are some of the reasons why you want to learn Blender with this online tutorial... Create assets for video games. Make unique 3D printed gifts. Design your dream house, car, etc Express yourself through 3D artwork.</p>	2022-06-23	
	The Complete Copywriting Course : Write to Sell Like a Pro	Ahmed ragab	14.99	DIGITAL MARKETING	<p>How to sell more products and services (just by being helpful!) How to compel your readers using timeless sales psychology How to get more sales from your website. How to eliminate "stage fright" and "blank page anxiety" Hacks and secrets from the world's most successful copywriters The sales psychology you NEED to grow a profitable business</p>	2022-06-23	
	Instagram Marketing 2022: Complete Guide To Instagram Growth	Ahmed ragab	9.99	DIGITAL MARKETING	<p>Have a powerful Instagram account setup for your Business or personal that you can build your brand and convert your followers into paying customers. Attract 10,000 real targeted followers to your Instagram account! Convert your new followers following your posts into paying customers who love your business. By the end of this course you will have all the competencies to grow your followers &amp; convert your successful Instagram into sales &amp; loyal customers We are up to date with all Instagram features &amp; will guide you step by step on how to utilise these functions to grow your account and market your business.</p>	2022-06-23	
	The Complete Digital Marketing Course - 12 Courses In 1	nada	Free	DIGITAL MARKETING	<p>Grow a Business Online From Scratch Make Money as an Hired Marketer Get Hired as a Digital Marketing Expert Work From Home as a Freelance Marketer</p>	2022-06-23	
	Microsoft Excel 2013 Advanced: Online Excel Training Course	nada	Free	IT&SOFTWARE	<p>To take your basic Excel skills to the next level To gain a solid understanding of the more powerful features and functions In Excel To be able to use Microsoft Excel in a commercial environment</p>	2022-06-23	
	Angular - The Complete Guide (2022 Edition)	nada	15.99	DEVELOPMENT	<p>Develop modern, complex, responsive and scalable web applications with Angular 14 Fully understand the architecture behind an Angular application and how to use it Use the gained, deep understanding of the Angular fundamentals to quickly establish yourself as a frontend developer Create single-page applications with one of the most modern JavaScript frameworks out there</p>	2022-06-23	
	AJAX Development	nada	9.99	DEVELOPMENT	<p>Understand the basics of Ajax interactions To understand and use the XMLHttpRequest object in Javascript To Update the Browser Window's HTML content dynamically through the DOM To Dynamically Create and Send Parameterized Queries to a Server To Monitor Server Response for Process Completion To Display Server Response in Plain Text To Receive and Process XML Objects in the DOM To Receive and Parse JSON Objects in the DOM To Understand the Role of a Web Server in Ajax</p>	2022-06-23	

< 1 2 3 4 5 6 > 10 / page ▾

Figure 4.18: Admin dashboard

The admin has the permission to edit some properties of any course, for example, he can edit the name, price, or add a new category, also he can lock and unlock any course if he sees that this course does not follow our policies. as shown in figure 4.19

The screenshot shows the Admin Dashboard for the Courses Manager. At the top, there are tabs for Courseme, Courses, and About, with Admin buttons for login and logout. The main area is titled "Admin Dashboard | Courses Manager" and features a course list table with columns: Cover, Course Name, Instructor Name, Price, Categories, Description, Joined Date, and Action. One course is listed:

Cover	Course Name	Instructor Name	Price	Categories	Description	Joined Date	Action
	Creating 3D environments in Blender	dived larrry	Free	DESIGN	Creating stunning unique environments Organize your workflow to make large environment scenes More than 250 unique 2K / 4K textures Lots of medieval reference photos E-book Old Masters Unveiled ( 250 pages) All scene assets, including buildings, rocks, grass, trees and more 6 Characters to fill up your scene Full course documentation for the Blender basics chapter Baking your models and importing in Unity 3D for games	2022-06-23	

Below the table is a form to edit the course details:

Course Name Creating 3D environment: <input type="text"/>	Price <input type="text"/> \$ 0	Categories Please select category <input type="button" value="New Category"/> <input type="button" value="Add"/>	Description Creating stunning <input type="text"/>
<input type="button" value="Save"/>			

Figure 4.19: Admin dashboard (courses manager)

The admin has the permission to access all students in the student manager tap, also he can see the courses that the student enrolled in. He can edit some properties of the student like the name and email, also he can block and unblock any student account.as shown in figure 4.20

The screenshot shows the Admin Dashboard for the Students Manager. At the top, there are tabs for Courseme, Courses, and About, with Admin buttons for login and logout. The main area is titled "Admin Dashboard | Students Manager" and features a student list table with columns: Student Name, Email, Courses, Joined Date, and Action. One student is listed:

Student Name	Email	Courses	Joined Date	Action
esam	esamrezk4@gmail.com	No Courses	2022-06-03	

Below the table is a form to edit the student details:

Student Name <input type="text"/> esam	Student email <input type="text"/> esamrezk4@gmail.com	<input type="button" value="Save"/>
---	---	-------------------------------------

Three other students are listed below:

loulahkareem	loulahkareem@gmail.com		2022-06-03	
Kareem	kareemloulah8@gmail.com		2022-06-03	
student	student@gmail.com		2022-06-04	

Figure 4.20: Admin dashboard (students manager)

The admin can access all instructors in the instructor's manager tap, for example, he can see what the course for every instructor is, and edit the name and email of the instructor, also he can block and unblock any instructor. as shown in figure 4.21

## Instructor Dashboard

in the Instructor Dashboard, the instructor can see the number of sources he published before and access all these courses, also he can see every course with the number of lectures, and he can make any course accessible for the students from the publish icon, or disable it also from the same icon . as shown in figure 4.22

The screenshot shows the 'Instructor' tab selected in the top right corner of the CourseMe interface. On the left, a sidebar menu includes 'Dashboard' (which is active and highlighted in blue), 'Course Create', and 'Revenue'. The main content area is titled 'Instructor Dashboard' and displays a message 'Number of courses : 22'. Below this, five course cards are listed, each with a green circular icon containing a checkmark in the top right corner, indicating they are live in the marketplace:

- Resume (CV) Writing/Hacking for Freelancers & Entrepreneurs** (5 Lessons)
- Front End Web Design using WYSIWYG Web Builder like a pro** (6 Lessons)
- Complete Photography Masterclass: 4 Courses IN 1** (7 Lessons)
- Intro to Digital Marketing - تعلم أساسيات التسويق الرقمي** (5 Lessons)
- Making Games With Phaser 2.X** (6 Lessons)

Each course card also includes a small note: 'Your course is live in the marketplace'.

Figure 4.21: Instructor dashboard

If the instructor clicks on any course in the dashboard, he will have the access to see the content of this course and see the number of students that are enrolled in this course. as shown in figure 4.23

The first part of this course is for Freelancers/Entrepreneurs who need a Resume to send to prospective clients in order to get hired for specific jobs. Resumes for Freelancers are very different from those of employees and office workers.

What are the differences?

This class covers all of that, going through a real example, step by step.

The Second part is for those of you who wish to earn a living from Freelancing. Those of you who would like to use your skills as an artist/designer/writer/translator/researcher/photographer/consultant/editor/programmer/etc. etc. and find a way to earn money, so you can have extra income, or even to earn a living, so you can quit your job and do your own work full-time.

Chances are, you would like to start earning a living from being a Freelancer, but you're not sure where or how to start. Well then, this Course is for you!

This Course will explain what you should do when you're starting from Zero. It will go through what to prepare ahead of time (so you won't be caught unprepared later!), as well as how to then find clients, approach them, and get chosen for freelance jobs.

Once you register for this course:

1. You will have lifetime access
2. You will receive a certificate upon completion
3. You can ask questions at any point and I will respond
4. The Course will be updated continuously, so you can always use this course as a reference.

See you in the course!

- Robert G

What you'll learn Adapt your Resume to Working Online Make your CV suitable for Entrepreneurship Learn What to Prepare Learn How to Find Clients Earn Extra Income Get Chosen for Freelance Jobs Use your skills to make money Are there any course requirements or prerequisites? This Course is for Anyone who wishes to Become a Freelancer Who this course is for: This Course is meant for people who wish to become Freelancers and are just starting out or going through their first steps

[Add Lesson](#)

### 5 Lessons

- 1 My example
- 2 Initial Resume
- 3 Top highlights
- 4 Descriptive Paragraph
- 5 Areas of Expertise

Figure 4.22: Course dashboard

The instructor can edit the content of the course, add new lectures, delete any previous lecture, and add a quiz for every single lecture. as shown in figure 4.23

**Courseme** Courses About Create Course

Dashboard Course Create Revenue

## Update Course

Resume (CV) Writing/Hacking for Freelancers & Entrepreneurs

The first part of this course is for Freelancers/Entrepreneurs who need a Resume to send to prospective clients in order to get hired for specific jobs. Resumes for Freelancers are very different from those of employees and office workers.

What are the differences?

This class covers all of that, going through a real example, step by step.

Free

Business

Upload Image

Save & Continue

**5 Lessons**

- 1 My example
- 2 Initial Resume
- 3 Top highlights
- 4 Descriptive Paragraph
- 5 Areas of Expertise

Edit Delete

Edit Delete

Edit Delete

Edit Delete

Edit Delete

**Courseme** @ 2022 Courseme

About us  
Membership

Privacy Policy  
Contact Us

Terms  
Become a Partner

Figure 4.23: Update course

## Add New Course

At the top of the page, there is a (Create course) button that the instructor can click if he needs to add a new course. as shown in figure 4.24

**Courseme** Courses About Create Course

Instructor Mns Instructor

Filter By Categories Categories Categories Categories Categories

**React**

React - The Complete Guide (incl Hooks, React Router, Redux)

by Mns Instructor

Programming

\$9.99

**JS**

The Complete JavaScript Course 2022: From Zero to Expert!

by Mns Instructor

Programming

\$15.99

**TOP ONLINE MACHİ LEARNİNG COURSE**

Complete Machine Learning & Data Science Bootcamp 2022

by Mns Instructor

Machine Learning

\$10.99

**Data Science Bootcamp 2022**

by Mns Instructor

Machine Learning

\$9.99

Figure 4.24: Add new courses

If the instructor clicks this icon, he will be redirected to create a course page and add the content of the new course that he needs. the instructor can decide if the course will be free or paid, if the course is paid, he can receive his money online in his bank account using the Stripe payment that we integrate into the platform. as shown in figure 4.25

The screenshot shows the 'Create Course' page of the Courseme platform. At the top, there's a navigation bar with 'Courseme' logo, 'Courses', 'About', and 'Create Course' (which is underlined). On the right, there are 'Instructor' and 'Mns Instructor' icons. Below the navigation, the main content area has a title 'Create Course'. It contains several input fields: 'Name' (empty), 'Category' (empty), 'Upload Image' (empty), and a dropdown for 'Paid' status set to '\$9.99'. A 'Save & Continue' button is at the bottom. At the very bottom of the page is a dark blue footer bar with links: 'Courseme', 'About us', 'Privacy Policy', 'Terms', '@ 2022 Courseme', 'Membership', 'Contact Us', and 'Become a Partner'.

Figure 4.25: Choose free or paid course

# CHAPTER 5: Recommendation system

## 5.1 Description

recommendation system which gives the user suggestions based on title, description and courses. Convert each description into vectors using TF-IDF. As shown in figure.5.2.

We are building two recommendation engines, one with a book title and another one with a description or course. The model recommends a similar book based on title and description or courses and books.

Calculate the similarity between all descriptions using cosine similarity.

Define a function that takes book title or course name and genre as an input and returns the top five similar recommended books or Courses based on the title and description

### 5.1.1 Context

This dataset holds a list of 230 books in the field of computer science and programming related topics. The list of books and Courses was constructed using many popular websites which provide information on book ratings and of all books in those websites the 230 most popular were selected.

### 5.1.2 Content

Inside that dataset, you will find general information about the book including the number of pages in the book, the book types, the book descriptions, and the book prices.

### 5.1.3 Dataset

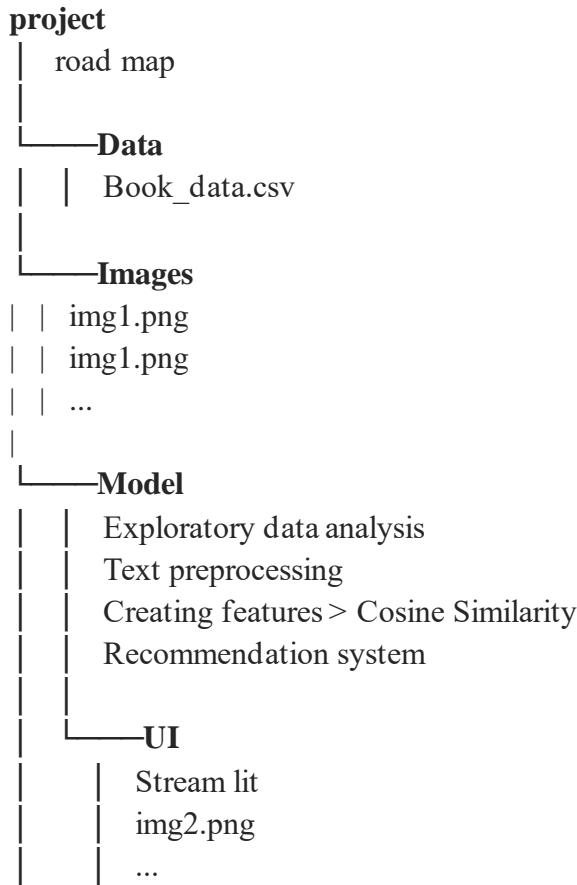
The dataset is provided by Kaggle

Book_title	Description	Course Name	Rating	Reviews
The Elements of Style	This style manual offers practical advi...	Style of Practical Advice on Improving ...	4.17	3,829
Responsive Web Design Overview For Begi...	In Responsive Web Design Overview For B...	Web Design For Beginners	3.33	0
The Meme Machine	What is a meme? First coined by Richard...	What is a meme? and how it spreads	3.84	117
Algorithms to Live By: The Computer Sci...	A fascinating exploration of how insight...	Algorithms The Computer Science of Huma...	4.15	1,817
Weapons of Math Destruction: How Big Da...	A former Wall Street quant sounds an al...	How Big Data Increases Inequality and T...	3.87	2,093
ZX Spectrum Games Code Club: Twenty fun...	This book is ideal for ZX Spectrum fans...	Learn or Practice Building Simple Games	4.62	0
The Master Algorithm: How the Quest for...	A thought-provoking and wide-ranging ex...	Algorithms for Machine learning discove...	3.78	481
Superintelligence: Paths, Dangers, Stra...	Superintelligence asks the questions: w...	What Happens when Machines Surpass Huma...	3.87	1,255

Figure 5.1: Display the data

**We can use "Book title" and "Description" columns to find books like each other.**

## 5.2 Project Structure



A simplified explanation of the project content

## 5.3 Structure model

### 1) Data Preparation:

First step --> Data Analysis: to analysis the data and collect all information we need

Second step --> Data Pre-Processing: to clean the data from any null values

Third step --> Feature Extraction: to extract the important feature of our data

### 2) Model

We use Cosine Similarity model to compare the weights of each word to do a recommendation

### 3) Deployment Plan

After doing a prediction by recommendation system, we need to deploy the result on our web site

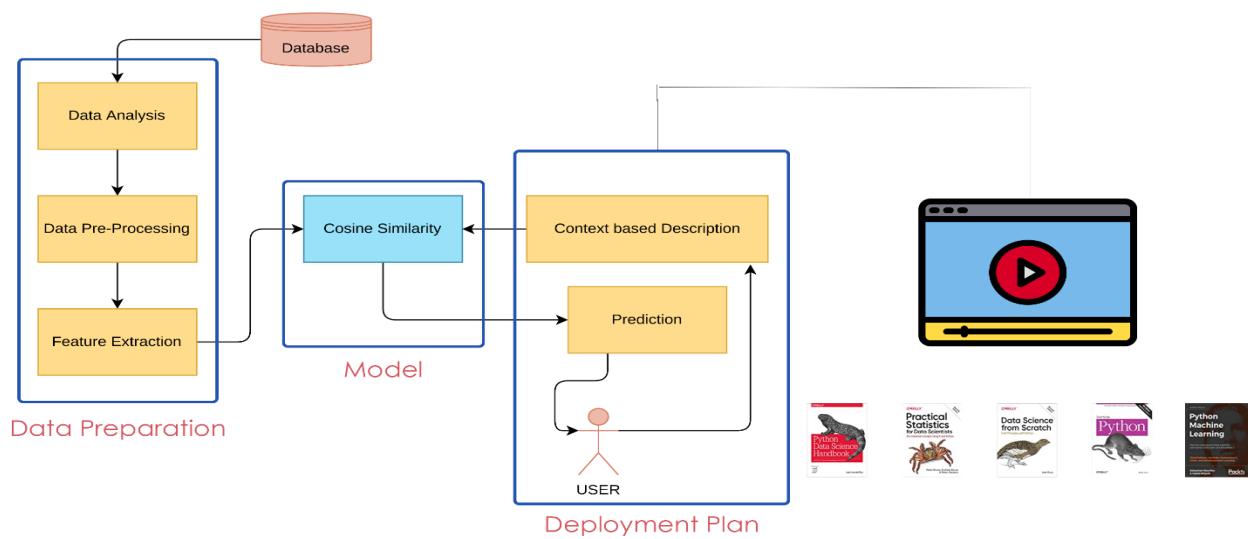


Figure 5.2: Structure Model

## 5.4 Data Analysis

We need to analysis our data to know more details and clarify all hidden information  
 So, we do some analysis in data such as top books respect to ratings, reviews, price, etc.  
 Things to do:

Configure data in the form of graphics to know its content

```
books.shape
```

```
(230, 8)
```

```
books.dtypes
```

Book_title	object
Description	object
Course Name	object
Rating	float64
Reviews	object
Number_Of_Pages	int64
Type	object
Price	float64
dtype:	object

```
sns.heatmap(books.isnull(), yticklabels=False, cbar=False, cmap='viridis')
```

```
<AxesSubplot:>
```

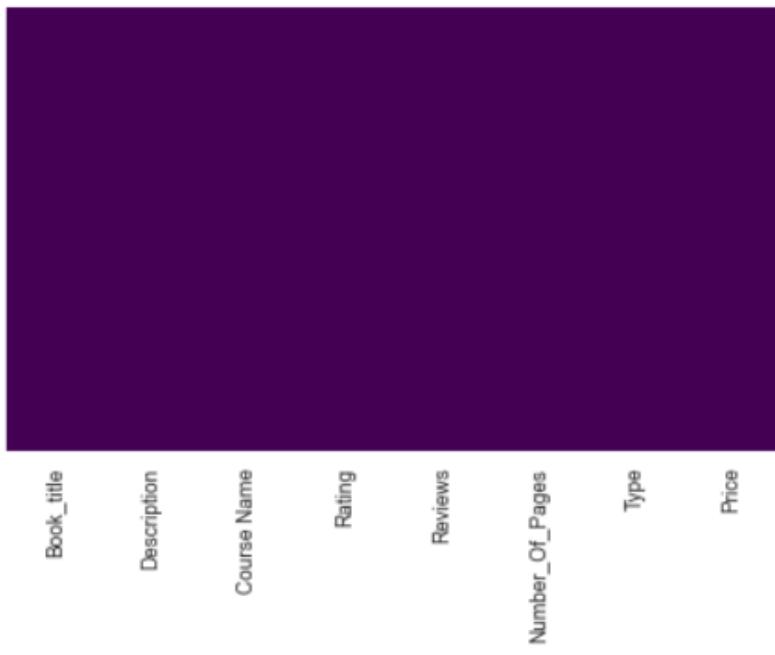


Figure 5.3: Shape of the data

**Shape of data** : consist of (230 columns) & (8 rows)

**Data types** : Number Of Pages -----< Int (64-bit)

- Book\_title -----< object
- Description -----< object
- Course Name -----< object
- Reviews -----< object
- Type -----< object
- Rating -----< Float (64-bit)
- Price -----< Float (64-bit)

**Heat map explanation** : we create this heat map to ensure that we have not (non-values)

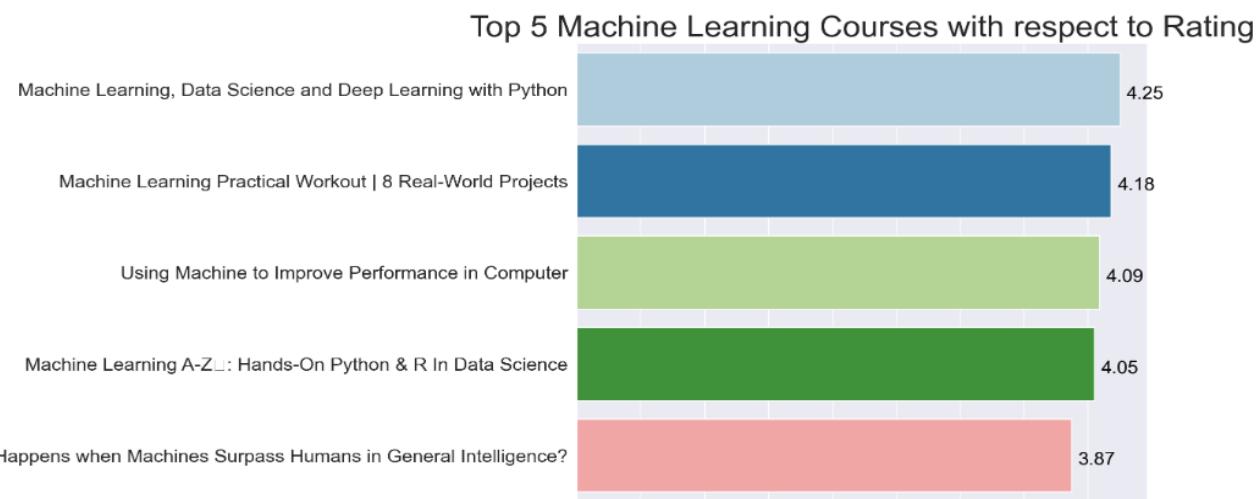


Figure 5.4: Represent top 5 machine learning courses respect to rating

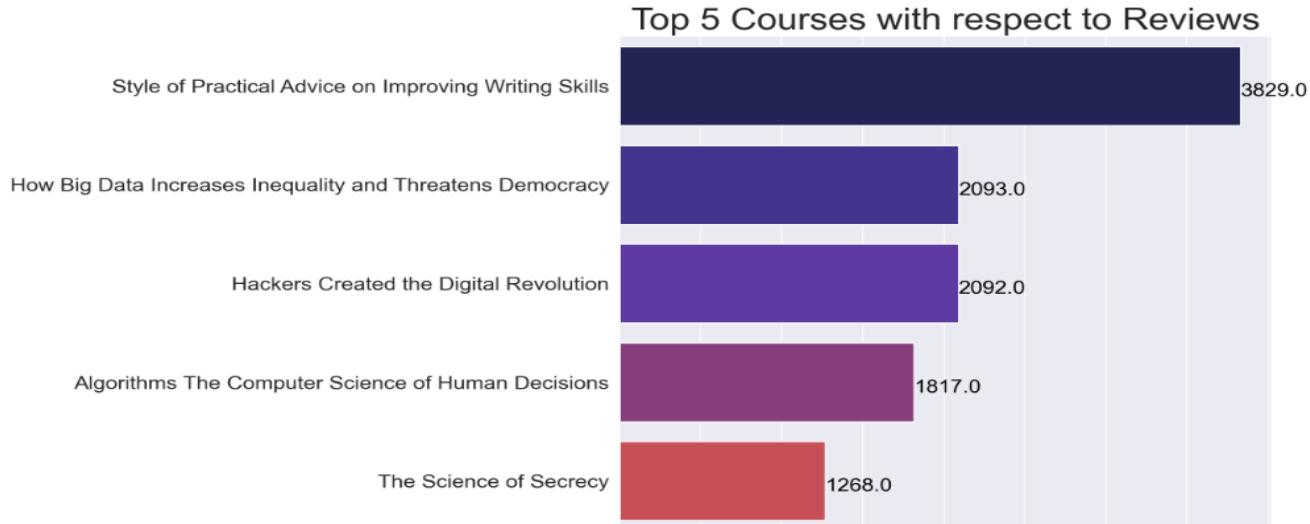


Figure 5.5: Represent top 5 courses respect to reviews

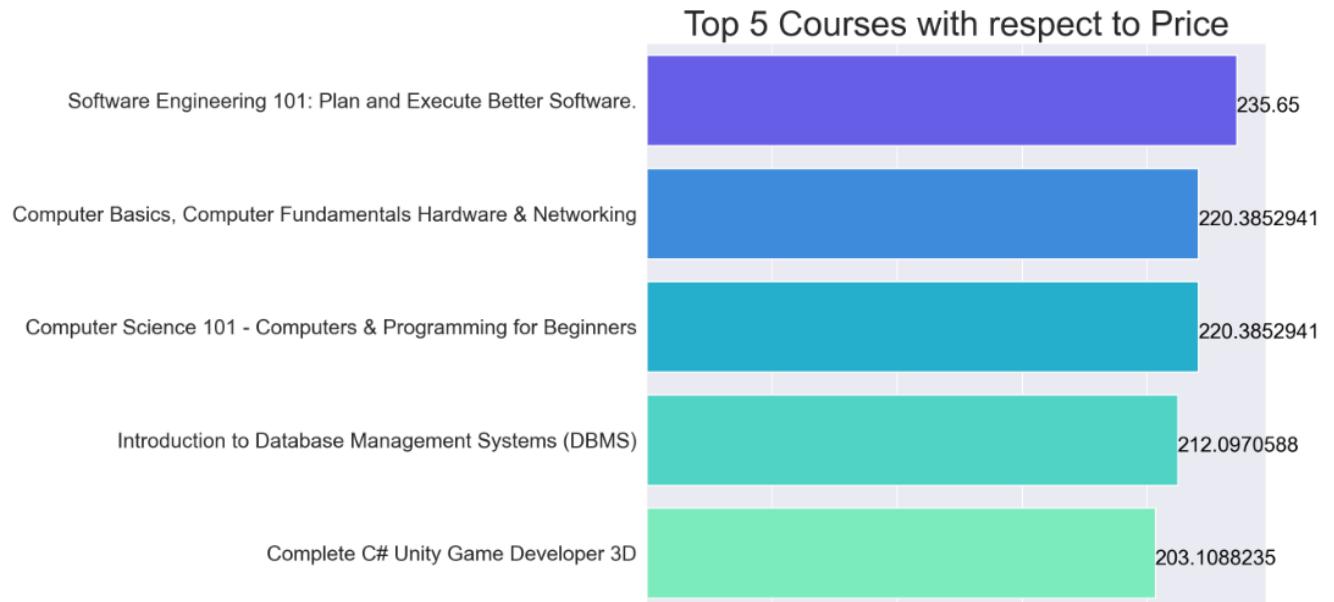


Figure 5.6: Represent top 5 courses respect to price

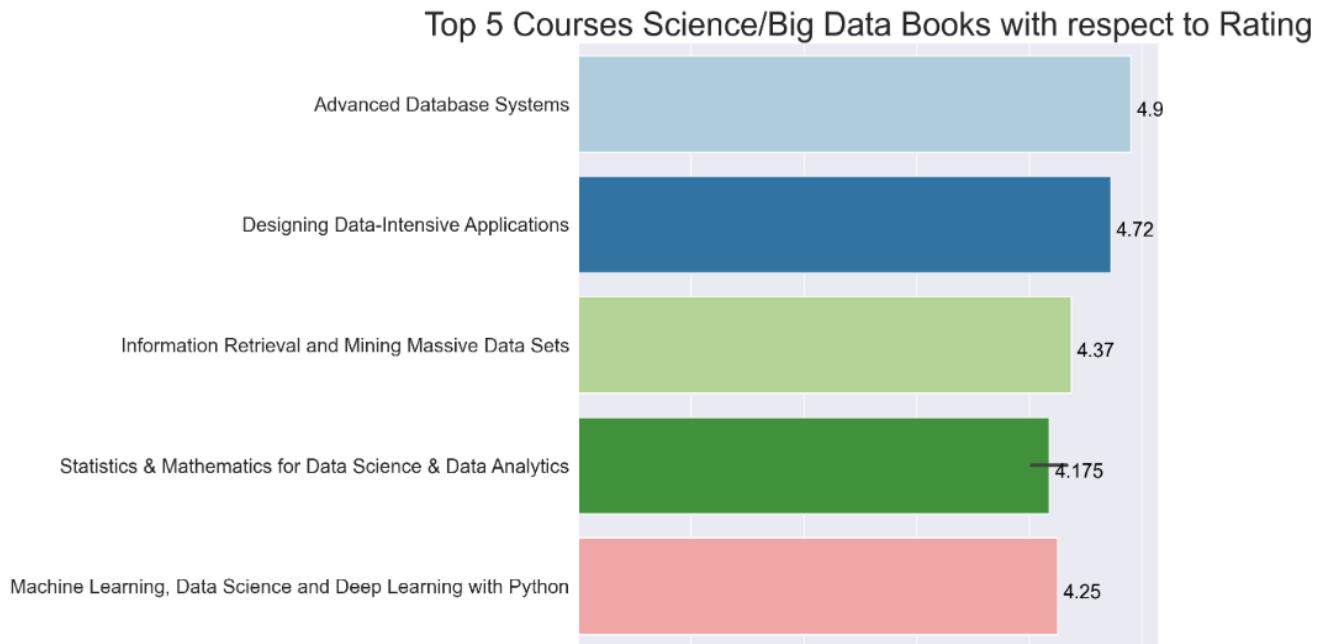


Figure 5.7: Represent top 5 data science/big data courses respect to rating

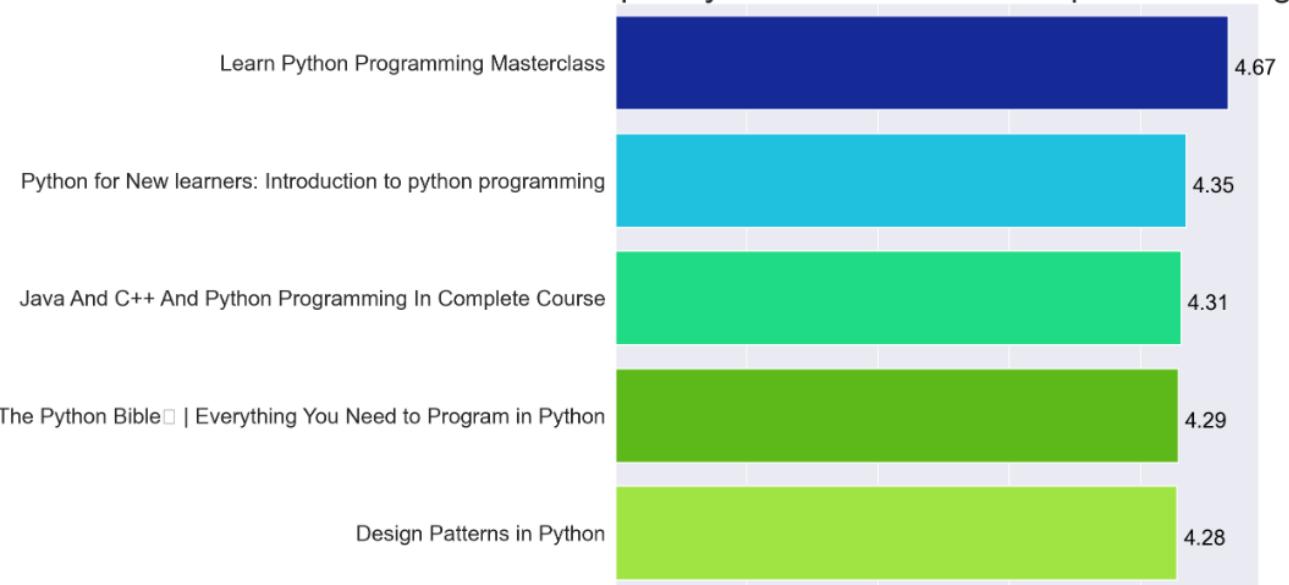
**Top 5 Python Courses with respect to Rating**

Figure 5.8: Represent top 5 python courses respect to rating

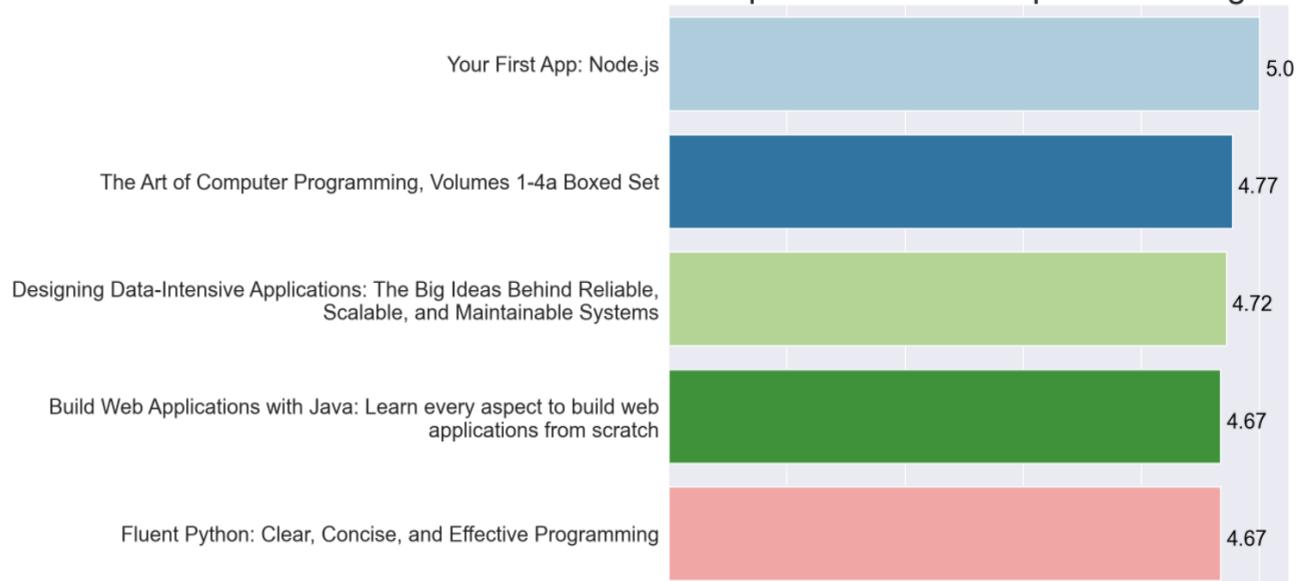
**Top 5 Books with respect to Rating**

Figure 5.9: Represent top 5 books respect to rating

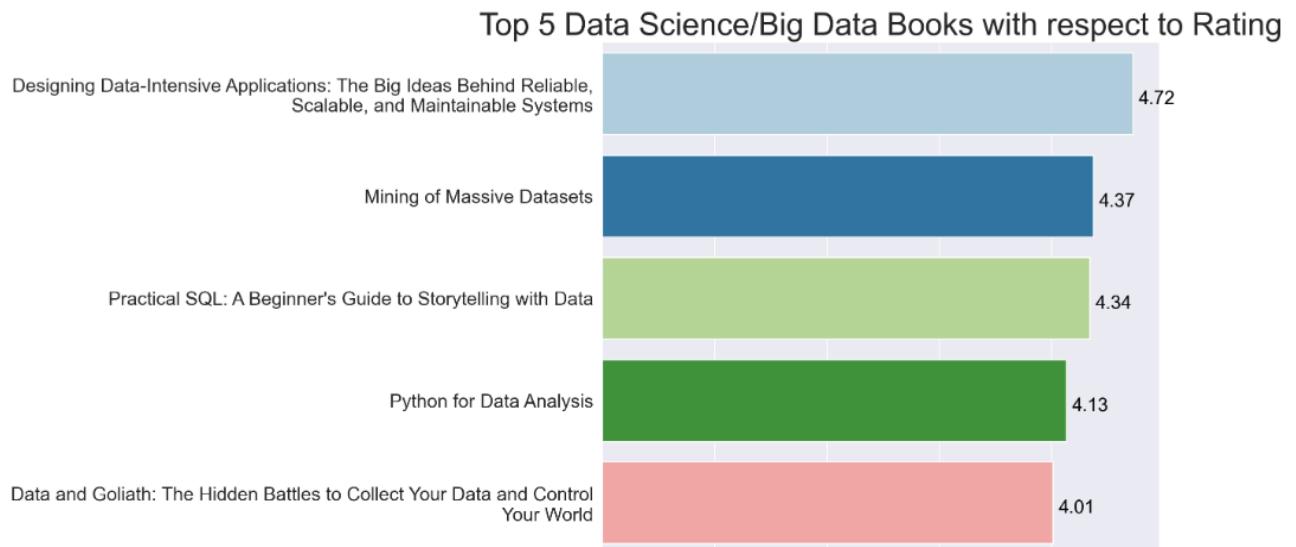


Figure 5.10: Represent top 5 data science/big data books respect to rating

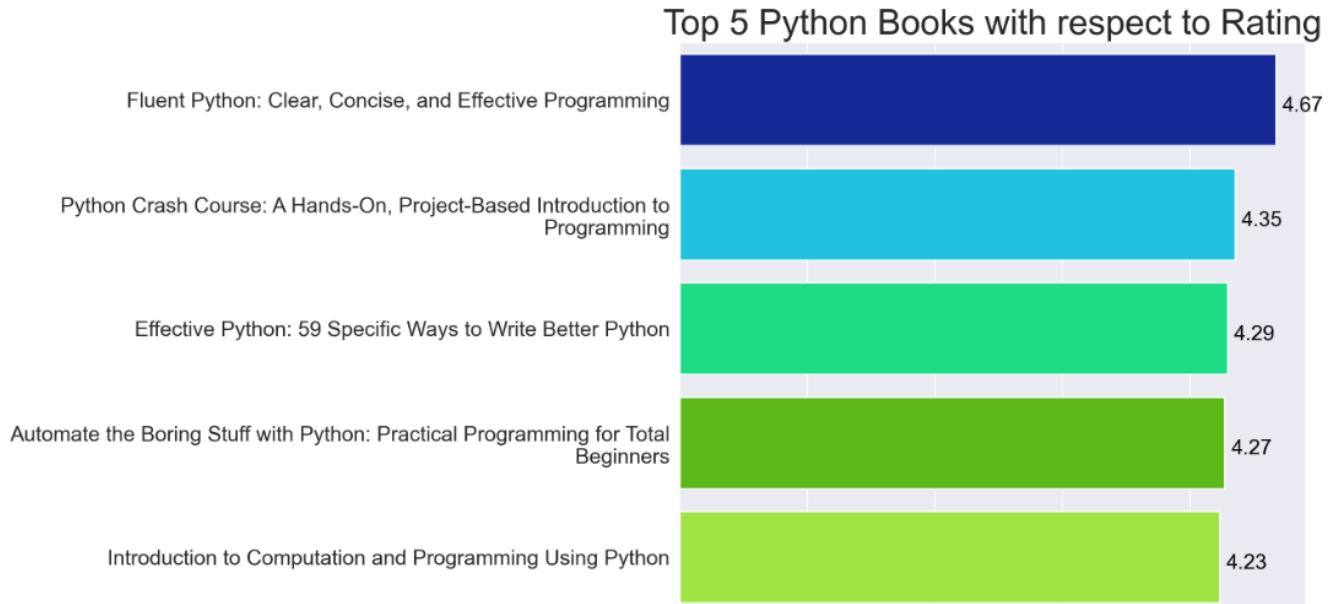


Figure 5.11: Represent top 5 python books respect to rating

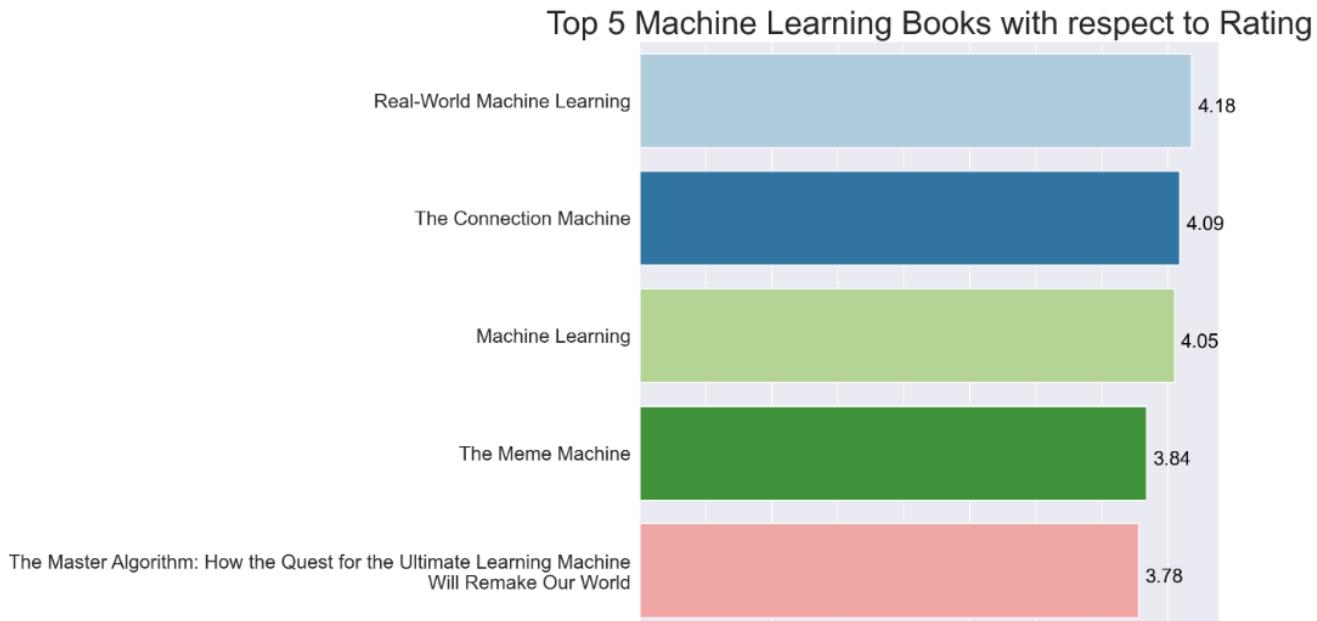


Figure 5.12: Represent top 5 machine learning books respect to rating

## 5.5 Data Pre-processing

Things to do:

- Extract only relevant columns (the ones having text we need)
- Check if there are null values in extracted columns
- Clean data from stopwords and punctuation marks
- Removing all words with digits and standalone digits

## Apply Code :

I will make a simple explanation of the processing work on the data, then we will collect all the functions in one function to save time

```

1 text="The Art of Computer Programming, Volumes 1-3 Boxed Set"
2 pu=text.translate(str.maketrans('', '', punctuation))
3 print(pu)

```

The Art of Computer Programming Volumes 13 Boxed Set

```

1 text2 = "The Art of Computer Programming, Volumes 1-3 Boxed Set"
2 result = re.sub(r'\d+', '',text2)
3 print(result)

```

The Art of Computer Programming, Volumes - Boxed Set

```

1 text = 'The Art of Computer Programming, Volumes 1-3 Boxed Set'
2 text = ' '.join([word for word in text.split() if word not in (stopwords.words('english'))])
3 print(text)

```

The Art Computer Programming, Volumes 1-3 Boxed Set

```

1 text5 = 'The Art of Computer Programming, Volumes 1-3 Boxed Set'
2 text5.lower()

```

'the art of computer programming, volumes 1-3 boxed set'

Figure 5.13: Explanation of data processing

```

1 # Set of punctuation signs to remove
2 from string import punctuation
3
4 def lower(text):
5     return text.lower()
6
7 def remove_punctuation(text):
8     return text.translate(str.maketrans('', '', punctuation))
9
10
11 #Join the list of strings into a string based on delimiter (' ')
12 def remove_stopwords(text):
13     return " ".join([word for word in str(text).split() if word not in stop])
14
15 import re #re - Regular expression operations
16
17 # Removing all words with digits and standalone digits
18 def remove_digits(text):
19     return re.sub(r'\d+', '', text)
20
21 # One function to clean it all
22 def clean_text(text):
23     text = lower(text)
24     text = remove_punctuation(text)
25     text = remove_stopwords(text)
26     text = remove_digits(text)
27     return text

```

Figure 5.14: Explanation of data processing II

Now we will apply what we did to the data to see the result

Description	Course Name	all1_clean
A revelatory exploration of the hottest...	Big Data: A Revolution That Will Transf...	revelatory exploration hottest trend te...
Take your kids from browsing to buildin...	Programming for Kids and Beginners: Lea...	take kids browsing building web latest ...
In his first book since the bestselling...	The Science of Secrecy	first book since bestselling fermat's e...
'Game Programming Golden Rules' provide...	Game Programming Golden Rules	game programming golden rules provides ...
Classic text considers general theory o...	Mathematical logic, for Public Application	classic text considers general theory c...
Here's your ticket to a world of advent...	Develop Minecraft Plugins (Java)	heres ticket world adventures minecraft...
Did you know that Bill Gates, cofounder...	Programming for Kids and Beginners: Lea...	know bill gates cofounder microsoft wro...
Written for experienced C programmers w...	C Programming For Beginners - Master th...	written experienced c programmers want ...

Figure 5.15: Result after apply processing

### What is the target of this process?

Remove unwanted words so that we get a text that contains the important words to be entered in Vector to extract the words with the important meaning.

## 5.6 Features engineering

For content-based book recommendation we have to use NLP techniques like

- **Tf-Idf Vectorizer** - Tf-Idf finds the no. of times a word occurs in a document and then compares this count to the no. of times the word shows up in a bunch of other documents in a collection. It then generates a rank for each word where it is important to a document if it shows up a lot in that particular document but doesn't show up a lot in all the other documents.

### Term Frequency

Term frequency is the frequency of a keyword compared to the number of words in a document. (1)

$$TF = \left( \frac{\text{Number of times keyword is found in document}}{\text{Number of words in document}} \right) \quad (1)$$

### Inverse Document Frequency

Inverse document frequency (IDF) is a dampening factor to reduce the importance of queries that are common to a lot of documents.

$$IDF = \log \left( \frac{\text{Number of documents}}{\text{Number of documents containing the keyword}} \right) \quad (2)$$

### How is TF-IDF is Calculated?

TF-IDF is essentially the multiplication of the Term Frequency (TF) and the inverse document frequency (IDF).

$$TF-IDF = TF * IDF$$

***TF-IDF = Term frequency \* inverse document frequency***

### Things to do:

- Initialize & Fit CountVectorizer into 'title' -> to create count\_matrix this is useful for cosine similarity
- Check all words/ features in the vocabulary

### Apply Code :

I will show an example of how it works.

I have stored a set of sentences for us to apply the idea to them

```

1 vector = ['JavaScript The Good Parts',
2           'Sams Teach Yourself JavaScript',
3           'Eloquent JavaScript A Modern Introduction',
4           'The Principles of Object Oriented JavaScript']

```

Figure 5.16: create variable and store sample of data

**The next step** is to build the dataframe from the list of articles stored in the vector variable.

There are main ways to build a TF-IDF word frequency Dataframe with **Scikit-learn**:

These solutions will create a sparse matrix that can be used to create the Data Frame.

```

1 from sklearn.feature_extraction.text import TfidfVectorizer
2
3 # Initialize the model
4 vec = TfidfVectorizer(stop_words=stop)
5
6 # Train the model
7 tf_idf = vec.fit_transform(vector)

```

Figure 5.17: import TF-IDF process

Now we will look at the data after applying the Tf-idf to find out the words of importance

```

1 print(vec.get_feature_names())
['eloquent', 'good', 'introduction', 'javascript', 'modern', 'object', 'oriented', 'parts', 'principles', 'sams', 'teach']

1 print(tf_idf.shape)
(4, 11)

1 pd.DataFrame(tf_idf.toarray())

```

	0	1	2	3	4	5	6	7	8	9	10
0	0.000000	0.663385	0.000000	0.346182	0.000000	0.000000	0.000000	0.663385	0.000000	0.000000	0.000000
1	0.000000	0.000000	0.000000	0.346182	0.000000	0.000000	0.000000	0.000000	0.000000	0.663385	0.663385
2	0.552805	0.000000	0.552805	0.288477	0.552805	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	0.000000	0.000000	0.000000	0.288477	0.000000	0.552805	0.552805	0.000000	0.552805	0.000000	0.000000

Figure 5.18: Result of TF-IDF process

After removing the stopwords, the words became important, and when comparing the resulting words, we will find that one of the most important words is JavaScript, because it was repeated in all the four texts.

### What is the target of this process?

The target of this process is to extract the words of interest and compare them with other texts to make recommendations for similar courses

Few Applications of TFIDF:

1.Information Retrieval

2.Text mining

3.User Modeling

4.Search Engine

### 5.Keyword Extraction

- **Keyword extraction** -> Extract keywords from Description
- **Cosine Similarity** -> Find cosine similarity between all Courses titles

Keyword Extraction : is automatic detection of terms that best describe the subject of a document.

For keyword extraction we use one of the below based o

## Cosine Similarity

Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis. (3)

We can calculate the similarities between the Descriptions from our matrix above, this can be done using cosine. This is based on the dot product operator from linear algebra and can be computed as:

$$\text{cosine}(\mathbf{v}, \mathbf{w}) = \frac{\mathbf{v} \cdot \mathbf{w}}{|\mathbf{v}| |\mathbf{w}|} = \frac{\sum_{i=1}^N v_i w_i}{\sqrt{\sum_{i=1}^N v_i^2} \sqrt{\sum_{i=1}^N w_i^2}}$$

(3)

The cosine values range from 1 for vectors pointing in the same directions to 0 for orthogonal vectors.

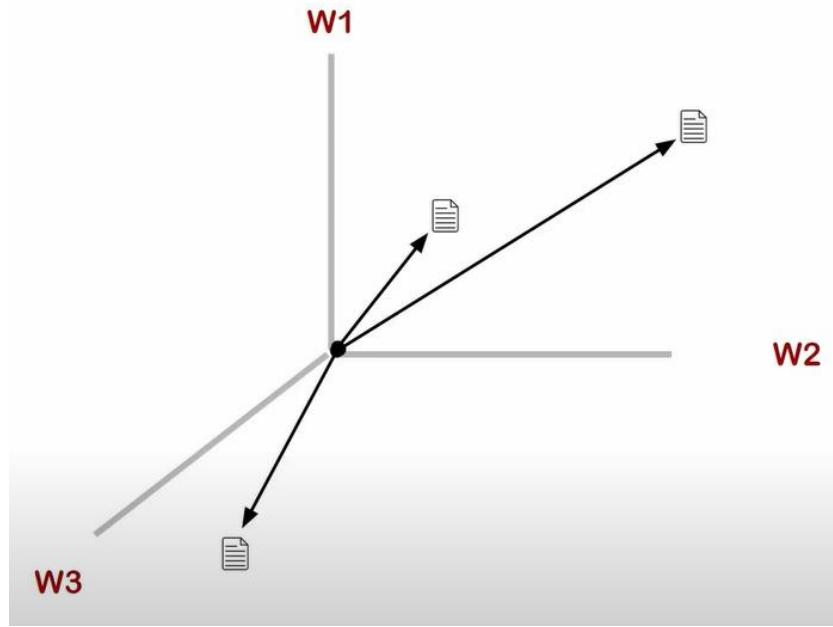


Figure 5.19: Similarity between two documents

## Apply Code :

I will explain a simple example with the same previous data with the addition of a title that has nothing to do with the existing data to calculate the similarity.

```

1 ex1 = 'JavaScript The Good Parts'
2 ex2 = 'Sams Teach Yourself JavaScript'
3 ex3 = 'Eloquent JavaScript A Modern Introduction'
4 ex4 = 'The Principles of Object Oriented JavaScript'
5 ex5 = 'Algorithms to Live By: The Computer Science of Human Decisions'
```

Figure 5.20: Example Simple Data

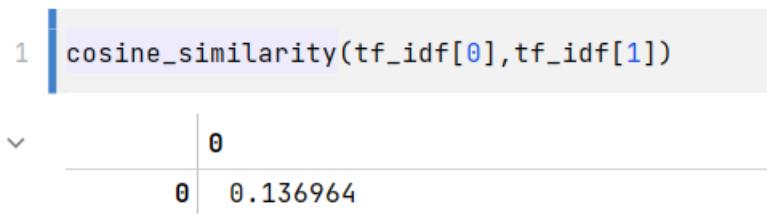


Figure 5.21: Result of similarity between (ex1) and (ex2)

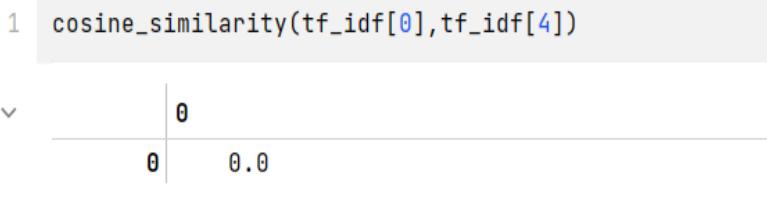


Figure 5.22: Result of similarity between (ex1) and (ex4)

Things to do:

- Initialize Cosine similarity into title matrix
- Extract features from book title
- Using Cosine similarity between this Description and all other Descriptions to be recommended the top 5 books or top 5 courses .

## 5.7 get\_recommendations

Now we have finished all the processing of the data and it is ready for use. We will connect all we have done through a simple code to make recommendations.

[19] [What Are Recommendation Systems in Machine Learning? | Analytics Steps](#)

```

1 def get_recommendations(value_of_element, feature_locate, df, vectors_array, feature_show):
2     # Locating target element by its specific value
3     index_of_element = df[df[feature_locate]==value_of_element].index.values[0]
4     # Finding its value to show
5     show_value_of_element = df.iloc[index_of_element][feature_show]
6     # Dropping target element from df
7     df_without = df.drop(index_of_element).reset_index()
8     # Dropping target element from vectors array
9     vectors_array = list(vectors_array)
10    target = vectors_array.pop(index_of_element).reshape(1,-1)
11    vectors_array = np.array(vectors_array)
12    # Finding cosine similarity between vectors
13    most_similar_sklearn = cosine_similarity(target, vectors_array)[0]
14    # Sorting coeffs in desc order
15    idx = (-most_similar_sklearn).argsort()
16    # Finding features of similar objects by index
17    all_values = df_without[[feature_show]]
18    for index in idx:
19        simular = all_values.values[index]
20
21    recommendations_df = pd.DataFrame({feature_show: show_value_of_element,
22                                         "rec_1": simular[0][0],
23                                         "rec_2": simular[1][0],
24                                         "rec_3": simular[2][0],
25                                         "rec_4": simular[3][0],
26                                         "rec_5": simular[4][0]}, index=[1])
27
28    return recommendations_df

```

Figure 5.23: Get recommendations

## Recommendation Course from All Courses

```
get_recommendations("Advanced Database Systems", 'Course Name', df, tag_vectors, 'Course Name')
```

rec_1	rec_2	rec_3
Practical SQL: A Beginner's	Introduction to Database Management Sys...	Fundamentals of Database Engineering
rec_4	rec_5	
Information Retrieval and Mining Massiv...	Data Analysis with Pandas and Python	

Figure 5.24: Example data from recommendation system

## 5.8 What is New Feature ?

It is usual in any educational system to recommend similar courses so that the user can take advantage of other courses available to develop his skills, but this is not enough because some people want to know the source from which the translation was made because there are people who are looking to solve their problems through scientific books so that they can be used and refer to it when they have a problem, that's why we added this feature to benefit everyone.

Recommendation Book from All Courses

```
1 get_recommendations("Pattern Recognition & Neural Networks", 'Course Name', df, tag_vectors, 'Book_title')
```

rec_1	rec_2	rec_3
Deep Learning	Machine Learning for Hackers	Machine Learning
rec_4	rec_5	
Real-World Machine Learning	The Master Algorithm: How the Quest for...	

Figure 5.25: Recommend books based on courses

## UI using Streamlit

### What is Streamlit?

Stream lit is a free and open-source framework to rapidly build and share beautiful machine learning and data science web apps. It is a Python-based library specifically designed for machine learning engineers. Data scientists or machine learning engineers are not web developers and they're not interested in spending weeks learning to use these frameworks to build web apps. Instead, they want a tool that is easier to learn and to use, as long as it can display data and collect needed parameters for modeling. Stream lit allows you to create a stunning-looking application with only a few lines of code.

# Book Recommendation System Based in Courses

Enter Course

recommend by :

HTML

Reference

Search

Please replace `st.beta_columns` with `st.columns`.`st.beta_columns` will be removed after 2021-11-02.

Recommending books similar to HTML

	Reference	rec_1	rec_2	rec_3	rec_4
0	HTML	Build Web Applications with Java: Learn every aspect to build web applications from scratch	Responsive Web Design Overview For Beginners	Mining of Massive Datasets	Programming in Lua, Fourth Edition

Figure 5.26: Using streamlit with recommendation system

## What's next in the future?

### 1. Recommendations by rating

Updating the system in the future based on instructor-recommended book reviews alongside the course to find out which books are the most useful and which will have a high rating so that our system always updates the data and filters the best

### 2. Providing assistance to the instructor by analyzing the data of the course users to identify the negative points so that he can change them to improve his course

Providing the instructor with services to analyze his own data in order to identify the strengths and weaknesses of his content to improve it so that the content is always effective, and this is based on user data

## REFRENCES

- [1] [A brief history of E-learning \(infographic\) - eFront Blog \(efrontlearning.com\)](#)
- [2] [A Brief History of Machine Learning - DATAVERSITY](#)
- [3] [VS Code – The Story and Technology Behind One of the World's Most Popular Desktop Apps for Developers | Tower Blog \(git-tower.com\)](#)
- [4] [A brief history of CSS until 2016 \(w3.org\)](#)
- [5] <https://reactjs.org/tutorial/tutorial.html>
- [6] [What is Next.js? | Learn Next.js \(nextjs.org\)](#)
- [7] [11 Years Of Node.JS: Timeline & Significant Contributions / Habr](#)
- [8] <https://www.mongodb.com/nosql-explained>
- [9] [A Brief History Of AWS - And How Computing Has Changed \(digitalcloud.training\)](#)
- [10] [History of Python - GeeksforGeeks](#)
- [11] [NumPy and Pandas Tutorial - Data Analysis with Python | CloudxLab Blog](#)
- [12] [A Brief History of Data Visualization \(dundas.com\)](#)
- [13] [Data processing technology history - Dataconomy](#)
- [14] [A Brief History of Natural Language Processing \(NLP\) - DATAVERSITY](#)
- [15] [Cosine Similarity - an overview | ScienceDirect Topics](#)
- [16] [Understanding TF-IDF: A Simple Introduction \(monkeylearn.com\)](#)
- [17] [What is Flask? - Flask: Develop Web Applications in Python \(educative.io\)](#)
- [18] [Introduction to Streamlit \(latentview.com\)](#)