ADDIS ABABA UNIVERSITY

College of Natural and Computational Science

School of Information Science

Enimar Code Learning and Hub

Industrial Project-1

Prepared by:

no	Name	ID No
1	**Esubalew Chekol**	UGR/2592/13
2	**Hayat Ebrahim**	UGR/4156/13
3	**Azeb Mihrete**	UGR/2604/13
4	**Lydia Fikire**	UGR/0373/13
5	**Melkamu Terefe**	UGR/1286/13
6	**Bereket Gebeyeaw**	UGR/8797/13

Table of Contents

List of Tables

List of Figures

List of Acronyms

BR = Business Rule

HTML= Hypertext Markup Language

EN=Enimar

UI = User Interface

FR = Functional requirement

UID=user interface id

CHAPTER ONE: INTRODUCTION

1.1 Overview of the system

Enimar Code Learning and Hub for Ethiopians will be an innovative online educational website having social media features that help to facilitate communication between students and students, students and teachers, and teachers with students. Additionally, it, specifically tailored to meet the unique needs of Ethiopian students interested in learning coding and wishing to work on software development. With a focus on providing high-quality, computer programming relevant content, the website aims to empower Ethiopian learners by equipping them with the knowledge and skills needed to excel in the programming field with interactive way of teaching and learning process.

Chapter one provides an introduction to the project of building an e-learning website. It begins with a background of the organization to provide context for the project. The chapter then proceeds to present the statement of the problem, outlining the specific challenges and issues that the e-learning website aims to address.

The objectives of the project are then described, including the general objective that outlines the overall goal of the project and the specific objectives that detail the specific outcomes and targets to be achieved. Following this, a feasibility study is conducted to assess the practicality and viability of the project.

The significance of the project is discussed, highlighting the potential impact and benefits it can bring to various stakeholders. The beneficiaries of the project are identified, specifying the individuals or groups who will benefit directly or indirectly from the e-learning website.

The methodology section explains the approach and techniques that will be employed throughout the project. It covers data collection methods and the system development methodology that will guide the project's execution. Additionally, the section elaborates on the development tools and technologies that will be utilized, including the fronted technologies, backend technologies, documentation and modeling tools, and the deployment environment.

The scope of the project is defined, outlining the boundaries and extent to which the e-learning website will be developed. Risks, assumptions, and constraints are identified to address potential challenges and limitations that may affect the project's progress and outcomes.

Phases and Deliverables of the project are then presented, providing an overview of the project's timeline and key milestones. A work-breakdown structure is provided to outline the major tasks and activities that need to be accomplished. Lastly, a project schedule is included to provide a timeline for the project's completion and the expected duration.

1.2 Background

Enimar Code Learning and Hub is an independent educational website that will be administered and controlled by our team members. We will not design the system for any specific organization.

E-learning, or electronic learning, is the delivery of learning and training through digital resources-learning enables individuals to learn from the comfort of their homes, eliminating the need for physical attendance in traditional classrooms.

E-learning is a broad concept. It encompasses various ways of using electronic technology, primarily the internet, to facilitate learning. It includes online courses, virtual classrooms, educational websites, digital resources, and interactive multimedia tools. The term covers a wide range of educational experiences, from formal online degree programs to informal self-paced learning modules.

E-learning is like a vast internet-based school encompassing a wide array of subjects, ranging from mathematics and history to diverse topics such as drawing or computer programming. It serves as a comprehensive online educational platform where one can explore various fields of interest. Even Though, the concept of e-learning is broad and inclusive, our project team aims to create a website tailored specifically for code learning, catering exclusively to Ethiopian students.

The research conducted by Tesfaye and Elizabeth suggests that while E-learning is not a new concept in Ethiopia and other parts of the world, its full implementation remains incomplete, particularly in Ethiopia (Tesfaye and Elizabeth, 2008).

The statement regarding its incomplete implementation indicates that it does not align with the preferences of Ethiopian students, failing to fulfill their needs including learning coding as desired.

Coding is the process of designing and building an executable computer program to accomplish a specific task or solve a particular problem. Learning to code entails acquiring the skills to create computer programs by understanding programming languages, algorithms, and problem-solving techniques. In Ethiopia, coding courses are given by many universities and colleges.

According to the article by Wubetu Barud Demilie(2019), titled "Causes of Failure of University Students in Computer Programming Courses: The Case of Wachemo University," many students may face challenges in coding proficiency due to limitations in the availability of appropriate learning resources and educational infrastructure.

E-learning websites designed for remote teaching are being implemented in both Ethiopian government sites and private educational organizations. Regrettably, these platforms primarily rely on video and file sharing, resulting in a unidirectional learning flow from the platform to the learner. Consequently, there is a lack of public accounts for teachers and students, hindering communication between them. This presents the first issue within available Ethiopian e-learning tools—limited interactivity, which prevents users from raising and discussing issues online. Additionally, certain e-learning sites, like Awaki, utilize YouTube videos embedded on their platforms, impeding interaction due to the video's origin from YouTube Because of the limitation of Ethiopian E-learning sites on providing coding content, Ethiopians go to international websites to access online learning.

Coursera, for example, is one of the popular sites to learn coding. Unfortunately, E-learning sites like Coursera are accessible after the payment is made with a credit card which is difficult to pay for Ethiopian students.

This is because Ethiopian banks' VISA cards are valid only in Ethiopia to transfer and receive only Ethiopian birr because of the fact that Ethiopian banks are not in the international standard. (ADHENA, 2008)

To address the above problems, the team plans to develop a website called Enimar Code Learning and Hub for Ethiopians that lets Ethiopians learn coding with Ethiopian teachers.

We are not going to build this system for an organization and we are not tied to an organization rule or restriction.

Our e-learning website is an independent initiative developed and managed by our dedicated team. Enimar Code Learning and Hub website, will be a great educational solution that aims to revolutionize online learning experiences. With a focus on fostering distinct profiles for both students and teachers, the website offers a personalized and efficient environment for education.

1.3 Statement of the problem

In the journal article by Tewodrose Tilahun (2022) titled "Enhancing Third-Year Computer Science Students' Computer Programming Skills at Hawassa University, Ethiopia," Teaching how to code in an efficient and effective method has been the most difficult subject in the world of computing for the last couple of decades. Aside from a lack of laboratory equipment, the most prevalent issues are the lengthy time it takes to understand a problem, devising algorithms, writing coding, syntax, and semantic complexity of the programming language. Because of the wide range of students' backgrounds, misconceptions about the course, traditional classroom teaching methodology, and the limited allotted time available to cover the course, it is extremely difficult for a teacher to go beyond the fundamental concepts that impede the development of students' problem-solving abilities.

So, students should use E-learning websites like Khan Academy, Codecademy, or Coursera, Udemy to learn programming whenever they have free time. This will help them enhance their problem-solving skills by practicing coding exercises and projects.

From the perspective of international E-learning websites However, there are some notable problems associated with choosing these websites for Ethiopian students. Firstly, the majority of instructors on these websites are not Ethiopians, which may impact the relevance of the content to the local context. Secondly, the requirement for payments poses a barrier, as many Ethiopians do not have access to international credit card payment mechanisms and more, most of these websites do not allow students to make chat and to discuss topics. Furthermore, there is a limitation in communication and collaboration features on this website, hindering interaction between students and teachers, as well as among students themselves.

From the perspective of Ethiopian e-Learning websites such as Ethio Programming have limitations in teaching programming. They lack features such as active participation from teachers and in-depth course offerings. Ethio Programming mainly provides introductory information about programming languages through text posts and YouTube videos. Moreover, the website does not support user logins. There is a need for the inclusion of account features to enhance the overall learning experience. Additionally, these websites lack comprehensive course coverage and They don't

provide instruction by teachers who are experts in the respective courses, which hinders the quality of education for students. Moreover, the presence of only one teacher further restricts the diversity of teaching styles and perspectives available to the learners.

Given these challenges, Ethiopian students currently struggle to learn programming. To address this, our team is in the process of developing an affordable e-Learning system tailored for Ethiopians for learning coding.

1.4. Objectives of the Project

1.4.1 General Objectives

The general objective of this project is building a web-based solution for learning code with integrated social interaction for Ethiopians.

1.4.2 Specific objectives

In order to achieve the general objectives stated above, the following activities will be undertaken.

- **Analyzing the existing e-learning systems:** Assess the currently available E-Learning systems like Coursera and Udemy and identify problems that makes them inconvenient to Ethiopians.
- **Generation of requirement of the proposed system:** define and identify several problems as a result of the preceding study. Develop needs for the new system or the new system will be proposed based on the requirements.
- **Conducting business analysis:** Business analysis involves systematically examining and evaluating an organization's processes, systems, and functions to identify opportunities for improvement and align solutions with strategic objectives.
- **Designing the proposed system:** after requirements are generated, the way to develop the new system will be planned.

- **Implementation of the New System:** Implement the new e-learning system according to the design, and align with the formulated system requirements.
- **Testing of System Functionality:** Conduct thorough testing of the new e-learning system to ensure that all features work seamlessly and meet the specified requirements.
- **Deployment of the new System:** transitioning from the development and testing phases to a live production environment where the system is accessible and actively used.

1.5 Feasibility study

A feasibility study is simply an assessment of the practicality of a proposed project plan or method. This is done by analyzing technical, economic, legal, operational and time feasibility factors. Just as the name implies, you're asking, "Is this feasible?". (projectmanager.com)

1.5.1. Economic Feasibility

Economic feasibility refers to the ability of a project or business venture to generate enough revenue to cover its costs and provide a reasonable return on investment. It involves analyzing the costs and benefits of a project, including the costs of materials, labor, and equipment, as well as the projected revenue from sales or other sources of income. (ceopedia.org)

The cost benefit analysis of the project is presented as follows.

Tangible Cost:

1. **Development Costs: including** Expenses related to the purchase of servers for hosting the website, computers and Costs associated with hiring developers.

2. **Operational Costs:** These costs are typically incurred at regular intervals, such as monthly, quarterly, or annually. operational costs are necessary to sustain the day-to-day activities of a business and are distinct from one-time or capital expenditures.

3. **Marketing and Promotion:** Expenses related to promoting the e-learning website through various channels, Costs associated with creating promotional materials, such as brochures and online ads.

4. **Training and Implementation:** Costs for training instructors and support staff on using the website and Expenses associated with deploying the website.

Table 1.1 development costs

no.	item	unit cost	quantity	total cost in birr
1	Desktop	70,000	2	140,000
2	Developer	100/hours	6 people for 300 hours	180,000
3	training	100,000	1	100,000
4	Printing and stationery cost	350	2 copies for both semesters	700
total	420,700			

Table 1.2 Operational costs

no.	item	Cost in birr per month
1	system administration	15,000
2	system maintenance	3,000
3	internet bill	3,000
4	server rent	6000
5	promotion	50,000
6	Website deployment	250

Total	77,000

Intangible Costs:

- Access to accurate, real-time information
- Time and energy spent to work on the project

Tangible Benefits:

We have used NPV (Net present value) to calculate the tangible benefit

According to the research of African development bank group "Ethiopia - Monetary Statistics Discount rate (%)" we take the discount rate in Ethiopia as 12 %.

We generate revenue through a 20% commission on each course enrollment. For instance, if the average course price is 1000 ETB, and we upload 100 courses with an average enrollment of 10 students per course in the first year, our projected revenue for the first year would be 200,000ETB. In the second year, assuming 200 courses are uploaded, each with an average enrollment of 50 students and an average course price of 1120 ETB, the anticipated revenue would be 2,240,000 ETB. And costs starting from year 1 are operational costs from the above table

Project	Year 0	Year1	Year2	total
Cost (cash out flows)	498,000	86,520	98,902.4	
Benefits (inflows)	0	200,000	2,240,000	
Discount factor	1	0.89	0.79	
Discounted cost	498,000	77,002.8	78,132.89	653,135.69
Discounted benefits	0	178,000	1,769,600	1,947,600
Discounted benefits - discounted cost	(498,000)	100,997.2	1,691,467.11	NPV=1,294,464.31 ROI=NPV/Discounted cost =198%

Based on the above cost benefit analysis using NPV and ROI the project is feasible

Intangible Benefits

- **Increased Efficiency:** The time saved by learners and instructors due to the flexibility and accessibility of e-learning.
- **Increased collaboration:** the system facilitates collaboration between students with each other and with teachers.
- **Increased Reach:** The ability to reach a larger audience without proportional increases in infrastructure and personnel costs.
- **Global Expansion:** Opportunities for expanding the user base globally without significant incremental costs.

So based on the above cost benefit analysis the project is economically feasible.

1.5.2. Operational feasibility

In Operational Feasibility the degree of providing service to requirements is analyzed along with how easy the product will be to operate and maintain after deployment. Along with these other operational scopes are determining usability of product. Determining suggested solutions by software development team is acceptable or not to address issues specified in the statement of the problem above. Our operational feasibility analysis outlines key objectives and strategies. Firstly, we aim to establish an accessible payment system tailored to local needs, including mobile banking options. Secondly, we plan to enhance the learning experience by delivering course content in native languages like Afan Oromo and Amharic. Lastly, we aim to overcome interaction limitations by developing enhanced features like Chat or Messaging System:

In conclusion, our e-learning website is operationally feasible, promising to enhance the accessibility, inclusivity, and overall effectiveness of coding education in Ethiopia. Through a localized payment system, multiple languages of instruction, and improved interaction features, our project aims to provide a tailored and conducive learning environment for individuals.

1.5.3. Technical Feasibility

Technical feasibility can be described as the formal process of assessing whether it is technically possible to manufacture a product or service. A technical feasibility study helps organizations determine whether they have the technical resources to convert the idea into a fully functional and profitable working system. (in.indeed.com)

In order to develop this system familiarity with web technologies such as Django and react is needed. The team members have a good web development background in Django. Therefore, we can conclude that this project is technically feasible.

1.5.4 Political Feasibility

The development of this system aims to benefit most Ethiopian students and teachers, ensuring equal access without discrimination. It is important to emphasize that this project is not intended to create any political conflicts or favor any particular groups. Rather, its purpose is to provide interactive service that can assist and support students and teachers in their educational endeavors. The system is designed to be inclusive, unbiased, and free from any harm or negative implications. Its focus is solely on providing a valuable educational resource that can benefit the entire Ethiopian educational community.so the project is politically feasible.

1.5.5. Schedule feasibility

The process of assessing the degree to which the potential time frame and completion dates for all major activities within a project meet organizational deadlines and constraints for affecting change. (teacherscollegesj.org)
As the WBS presented below, the team has adapted the project duration accordingly. Phase one of this project will be completed by the end of the first semester while phase two will extend to the end of the second semester. Our plan is to finish on time

and have it run in a real environment by the end of this academic year. Hence, as it can be clearly seen in the project schedule in section, this project is feasible time wise.

1.6 Significance of the Project

Enimar Code Learning and Hub has a great significance. Firstly, it aims to increase access to coding education by addressing payment system limitations, thereby empowering education and enabling a wider range of Ethiopians to gain valuable coding skills. Moreover, the focus on enhancing interaction between students and teachers through its social media features contributes to a more engaging and personalized learning experience, addressing existing deficiencies in the e-learning landscape.

Furthermore, the website's customization to local educational needs, specifically designed for the Ethiopian context, seeks to incorporate cultural relevance. The contribution to skill development and employability among Ethiopians is another crucial impact, empowering individuals to actively participate in the rapidly evolving technology sector, both locally and globally.

Additionally, the website's role in supporting economic growth, particularly in the local technology sector, has positive implications for innovation and overall economic development within Ethiopia. By promoting a sense of community among learners and instructors, the website encourages collaboration, knowledge sharing, and continuous learning, thus building a strong coding community.

By addressing challenges related to payment systems, and interaction limitations, the website will play a significant role in facilitating coding education in Ethiopia.

1.7. Beneficiaries of the project

The beneficiaries of the Enimar Code Learning and Hub designed to address coding education challenges in Ethiopia would include various stakeholders involved in the education and technology sectors. Here are key beneficiaries:

Students: Individuals in Ethiopia will benefit significantly. The website's focus on providing coding courses in local languages and enabling collaboration among students and instructors due to its social media features can empower students with the skills necessary for success in the technology industry.

Instructors: instructors will benefit from the new job opportunity of teaching through the platform with features that can contribute to a more effective and engaging teaching experience.

Employers and the Technology Sector: Employers in Ethiopia's technology sector will benefit from a pool of skilled and well-educated individuals in coding. The website's contribution to skill development can lead to a more qualified workforce, supporting the growth and innovation of the local technology industry.

Local Communities: The project can have positive impacts on local communities by providing individuals with valuable skills that enhance employability. This, in turn, can contribute to the overall socio-economic development of the communities.

1.8 Methodology

1.8.1. Data Collection

Data collection is one of the important tasks to analyze how activities are done in the existing system and develop the new system. Data for developing this system is obtained from different sources. These data collection methods we are going use are:

1.8.2. Observation

Is one of the information collections that we are using to see what is exactly happening in the problem area. We used this method to analyze the system requirement. This includes sample business form and report, and documentation of Existing system. Analyzing the website structure obtained from the Udemy, Coursera, and other websites which are done in Ethiopia, will help us to understand the problem of existing websites for Ethiopian students and teachers.

1.8.3. Interview

This approach combines two different kinds of approaches. Interviews, both open-ended and closed-ended. In order to identify the current working procedure for the teaching and learning process in the university and to better understand their experiences using various e-learning websites, the team has chosen to conduct open-ended interviews with students and university instructors.

1.8.2. System Development Methodology

For system development, we will use Agile methodology specifically scrum because Scrum provides a flexible approach to project management, allowing development teams to work collaboratively, adapt to changing requirements, and deliver incremental releases of software. It emphasizes iterative development, frequent communication, and continuous improvement.

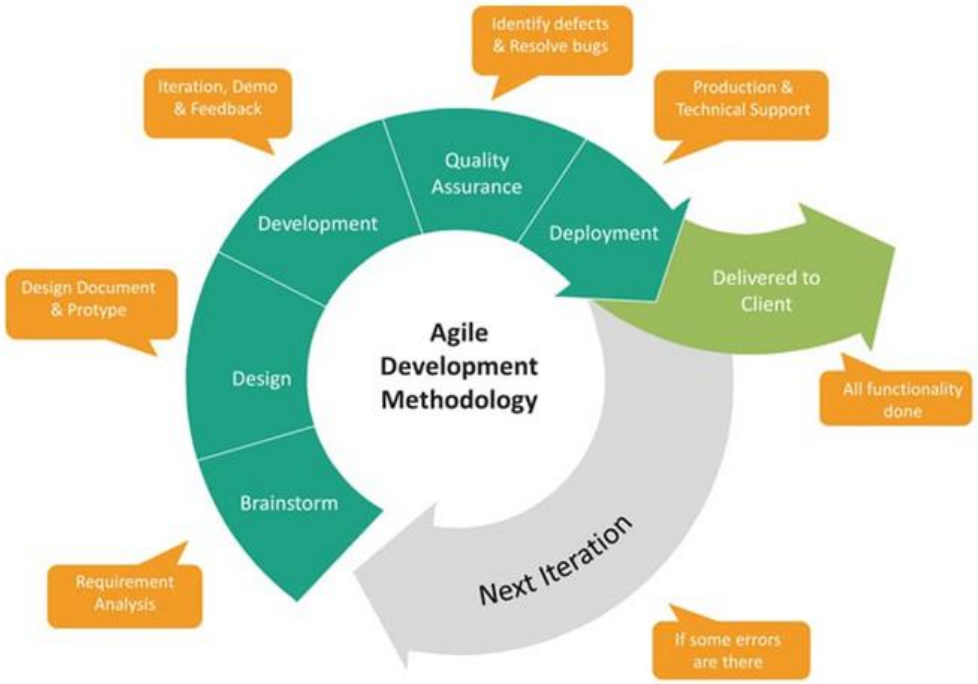

Figure 1 Agile Development Methodology

1.9. Development Tools and Technologies

Choosing the right development technologies and tools for a project is an important decision that can affect the success, cost, and quality of the product. There are many factors to consider when selecting a technology stack, such as the size and type of the project, the complexity of the tasks, the development speed and cost, the availability and expertise of the developers, the presence of development tools and ready-made solutions, the project flexibility and scalability, the technical requirements, and the security risks.

1.9.1. Front end Technologies

As a Front-end tool for designing the interface the project team will use **react.**

REACT: is an open-source JavaScript library for building user interfaces, particularly for single-page applications (SPAs) and complex web applications. It was developed and is maintained by Facebook. React is known for its component-based architecture and its ability to efficiently update and render components as the data in your application changes.

We will use React as a frontend tool because React has modular and reusable UI components. This approach significantly creates faster development processes, better code organization, and enhances maintainability. Another critical aspect is React's utilization of a virtual DOM, enabling efficient updates to the actual DOM by selectively rendering only the components that have undergone changes.

1.9.2. Backend Technologies

As a back-end tool the project team will use Django and **PostgreSQL**.

Django is a high-level Python web framework that encourages rapid development and clean design. Built by experienced developers, it takes care of much of the hassle of web development, so we can focus on writing our app without needing to reinvent the wheel. It’s free and open source.

Django is a great choice for projects that handle large volumes of content, user interactions or heavy traffic, or deal with complex functions or technology (online chat, live stream).

As we need fast development, we wanted to use the Django features that aren't available in most of the other popular web development backends such node.js. (e.g., ORM, Authentication, Default Security and middleware)

Django has its own naming system for all functions and components (e.g., HTTP responses are called "views").

It also has an admin panel, which is deemed easier to work with than in Laravel and other technical Django features, including:

-Simple syntax;

-Its own web server (for development);

-MVC (Model-View-Controller) core architecture;

- "Batteries included" (comes with all the essentials needed to solve solving common cases);

-An ORM (Object Relational Mapper);

-HTTP libraries;

-Middleware support; and

-A Python unit test framework. (DjangoStars, 2023)

PostgreSQL

PostgreSQL is a powerful, open-source object-relational database system with over 35 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance.

Django supports many databases including MySQL, SQLite3, SQL, MariaDB from those we choose Postgres database. It gives more flexibility in data types, scalability, concurrency, and data integrity.

Sources say Postgres database is good if the website is complex and handles many read and writes and indexing actions. PostgreSQL is better suited for enterprise-level applications with frequent write operations and complex queries. (AWS.Amazon.com, 2022).

Because Enimar code learning and hub performs complex actions like loading video, searching and filtering course contents Postgres is better than others in performing these tasks.

PostgreSQL is faster when dealing with massive datasets, complicated queries, and read-write operations (Hevo Data, 2023).

1.9.3. Documentation and Modeling Tools

For effective project documentation and modeling, we will use the following tools:

1. Microsoft Word: This is a word processing application widely used for creating and editing text documents. It offers various formatting tools, templates, and features for document creation and collaboration.
2. UMLet: UMLet is a free, open-source tool for creating Unified Modeling Language (UML) diagrams. It provides a simple and intuitive interface for drawing different types of UML diagrams, including class diagrams, use case diagrams, sequence diagrams, and more
3. Google Docs: Google Docs is a web-based application for creating and editing documents online. It allows real-time collaboration, making it easy for multiple users to work on the same document simultaneously. It's part of the Google Workspace suite.
4. Wondershare EdrawMax: EdrawMax is a versatile diagramming tool that supports the creation of various types of diagrams, charts, and visual representations. It includes templates for flowcharts, mind maps, organizational charts, and more.

5. Figma: we prefer to use Figma because it lets us share design files and collaborate in real-time quickly with our entire team. for documentation and modeling.

1.9.4. Deployment Environment

Django deployment is the process of transferring a Django web application from a development environment to a production environment. In other words, it is the act of making your application accessible to users on the internet.

Django, being a web framework, needs a web server in order to operate. And since most web servers don't natively speak Python, we need an interface to make that communication happen.

Django currently supports two interfaces: WSGI and ASGI.

WSGI is the main Python standard for communicating between web servers and applications, but it only supports synchronous code.

ASGI is the new, asynchronous-friendly standard that will allow your Django site to use asynchronous Python features, and asynchronous Django features as they are developed.

Popular choices for deploying Django include Railway, Python Anywhere, Amazon Web Services, Microsoft Azure, etc.

We deploy using uWSGI which is an extremely fast Python application server. To simplify this process we use Render, Vercel hosting providers

1.10 Scope of the project

In this project, we proposed to develop a web-based code learning platform having social media features. The major deliverable of the project for Enimar Code Learning and Hub will be a web system that will enable:

1. To create accounts for both students and teachers.
2. Teachers create courses.
3. Teachers can view the lists of videos that he/she created.
4. Teachers to upload free videos
5. Students view available courses
6. Students Make payment and enroll in courses
7. Students can track course progress and view assessment results and their grades.
8. Students provide feedback on courses
9. Students get certificates.

We will provide courses related to coding so offering courses outside of that is not part of our project.

1.11. Risks, assumptions and constraints

Risk

It is likely that the new system we will create won't succeed in the company as we don't plan ahead for what would happen both during and after the system is put into place.

Table 1.3 risk management table

Risk	**Likelihood**	**Mitigation Strategy**
Managing and controlling the contents of videos that will be uploaded by teachers	Medium	Carefully test contents to ensure it is high of quality, engaging, and effective.
Lack of Budget and resources to accomplish our project	high	Identifying possible funding sources like NGOs.
Problem in Internet Connectivity	high	Design the website to be accessible in offline mode, allowing learners to download content when connected and access it offline.
Security and privacy issues related to the system	Medium	Prevent unauthorized access to the website by regularly updating the website and Conducting security audits to identify and address vulnerabilities.

Constraints of the project

These are constraints associated with the development and implementation of the project. Here some of the constraints are listed below.

1. Budget Constraints:

Our project may lack basic and adequate financial resource for purchasing equipment and marketing and promotion

2.**Time constraints**:

The time frame for developing the project may be insufficient

3.**Resource constraints:** the amount of resource including human resource may not be enough to finish the project based on target time boundary and scope

4. **Deployment constraints:** Django websites are not easy to deploy for free

Assumptions of the project

- The development of the project will be met with the available resources.
- All students and teachers who will use our website has internet access and basic hardware facilities
- The internet connection is assumed to be accessible 24 hours a day 7 days a week.
- The team members of the group will all see through the development and implementation of the project providing all the required skills.
- All development equipment is expected to remain in good condition.
- The development team assumes to get that necessary information from students and teachers.

1.12 Phases and Deliverable of Project

1. Planning:

It is the phase that we will Plan the project, including schedules, resource allocation, and budgets.

Deliverable:

Project Plan: a comprehensive plan outlining tasks, timelines, resources, and potential risks

(Proposal).

2. Requirements Gathering

It is basically the phase that our team will Understand and document the project requirements.

Deliverable:

A detailed document that will specify the functional and non-functional requirements of the Enimar e-learning.

3. System Analysis

We will try to analysis the requirement of the project

Deliverable:

specification document, which consists of

• System use case modeling

• Class modeling

• Sequence diagram

• Activity diagram

• User interface prototyping

4. Design:

We will Create a blueprint for the system based on the requirements.

Deliverable:

design document, which consist of

• Class type architecture

• State chart diagram

• Collaboration diagram

• Component diagram

• Deployment diagram

• Persistence diagram

• User interface design

4. Implementation (Coding):

We will develop the actual website based on the design.

Deliverable:

We will have actual code written by our team.

5. Testing:

Verify that the Website meets the specified requirements.

Deliverable:

Records of test executions and their outcomes.

6. Deployment:

We will Release the website and make it available for possible users.

Deliverable:

We will have fully accessible website with documentation to guide end-users in utilizing the Website.

7. Maintenance and Support:

We will Address issues, fix bugs, and provide ongoing support.

Deliverable:

Bug Fixes and Patches: Code modifications to address identified issues.

1.13. Work-breakdown structure

Table 1.3 Work-breakdown structure

No	TASKS	RESPONSIBLE
1.	Project proposal	
1.1	Introduction of the Proposal	Esubalew Chekol
1.2.	Statement of the Problem	Esubalew Chekol
1.3.	Objectives of the project General objective Specific objectives	Esubalew Chekol
1.4.	Feasibility Analysis Economic Feasibility	Hayat Ebrahim
	Operational Feasibility Schedule Feasibility Political Feasibility Technical Feasibility	

1.5.	Significance of the Project	Hayat Ebrahim
1.6.	Beneficiary of the project	Hayat Ebrahim
1.7.	Methodology Data collection System Development Methodology	Lydia Fikire
1.8.	Development Tools and Technologies Frontend Technologies Backend Technologies Documentation and Modeling Tools Deployment Environment	Esubalew Chekol
1.9.	Scope	Azeb Mehrete
1.10.	Risks, assumptions and constraints	
1.11	Phases and deliverables of the project	Lydia Fikire
1.12	Work-breakdown structure	
1.13	Project schedule	
2.	Business Area Analysis and Requirement Definition	
2.1	Introduction	Melkamu Terefa

2.2	Business area analysis Activities/functions of the organization Problems of the current system Forms and Reports of the current system Players of the existing system	Esubalew Chekol and Bereket Gebeyaw
2.3	Requirements Definition Functional Requirements	Azeb Mehrete, Hayat Ebrahim and Lydia Fikre

	Non-functional requirement	Bereket Gebeyaw
3.	Object Oriented Analysis	
3.1.	Introduction	Hayat Ebrahim
3.2.	System use case UI identification Business rule identification Actor identification Designing the use case diagram Use case Description	Esubalew Chekol, Melkamu Terefa and Azeb Mehrete

	Non-functional requirement	Bereket Gebeyaw
3.	Object Oriented Analysis	
3.3	Conceptual Modeling Class diagram Class description	Hayat Ebrahim
3.4	Sequence Diagramming	Bereket Gebeyaw
3.5	User interface prototyping	Hayat Ebrahim and Bereket Gebeyaw
4	Conclusion of the first phase	Azeb Mehrete
5.	Object Oriented Design	
5.1.	Review of the First Phase	Melkamu Terefe
5.2.	Introduction to the Chapter	Azeb Mehrete

5.3.	Class Type Architecture User Interface Layer Process/Controller Layer Domain/Business Layer Persistent Layer	Lydia Fikire ,Hayat Ebrahim, and Bereket Gebeyaw
5.4.	Design Class Modeling Class Diagram Description of Classes	Melkamu Tereffa
5.5.	Collaboration Diagram	Esubalew Chekol
5.6.	Component Diagram	Lydia Fikire
5.7.	Deployment Diagram	Bereket Gebeyaw
5.8.	Relational Persistent Mode	Esubalew Chekol
5.9.	User Interface User Interface Flow Diagram User Interface Design	Hayat Ebrahim and Lydia Fikire
6.	Object Oriented Implementation	
6.1.	Introduction	Hayat Ebrahim

6.2.	Implementation Technology	Lydia Fikire
6.3.	Testing and Testing Procedures Unit Testing Integration Testing System Testing	Melkamu Tereffa and Hayat Ebrahim
6.4.	Deployment/Installation Process Hardware and Software Acquisitions	Esubalew Chekol
7.	Conclusion and Recommendations	
7.1.	Conclusion	Bereket Gebeyaw
7.2.	Recommendation	Azeb Mihrete

1.14 Project schedule

Project schedule is a timetable that describes the project plan in terms of time using a Gantt chart as shown below. To implement the proposed project effectively and reach the planned goal, there are actions to be undertaken and performed; these activities are shown in the Gantt chart.

ID	Task Name	Start	Finish	Duration
1	Enimar E-learning	11/3/2023	6/12/2024	159d
2	Planning	11/3/2023	11/20/2023	12d
3	**Phase 1**	11/3/2023	3/7/2024	90d
4	Proposal preparation	11/3/2023	11/16/2023	10d
5	**Problem Identification**	11/3/2023	11/16/2023	10d
6	Data collection	11/3/2023	11/9/2023	5d
7	WBS	11/3/2023	11/8/2023	4d
8	Proposal approval	11/3/2023	11/3/2023	1d
9	**Business area analysis and Requirement Definition**	11/3/2023	12/7/2023	25d
10	Essential modelong	11/3/2023	11/30/2023	20d
11	Requirement approval	11/3/2023	11/3/2023	1d
12	Object oriented analysis	11/3/2023	1/18/2024	55d
13	Use case modeling	11/3/2023	12/21/2023	35d
14	Class modeling	11/3/2023	12/21/2023	35d
15	Sequence modeling	11/3/2023	12/28/2023	40d
16	Approval	11/3/2023	11/3/2023	1d
17	**Phase 2**	11/3/2023	2/14/2024	74d
18	**Object Oriented design**	11/3/2023	12/6/2023	24d
19	System design	11/3/2023	11/22/2023	14d
20	Database design	11/3/2023	11/23/2023	15d
21	Interface design	11/3/2023	11/30/2023	20d
22	**Object oriented implementation**	11/3/2023	1/30/2024	63d
23	Coding	11/3/2023	12/28/2023	40d
24	Testing	11/3/2023	11/16/2023	10d
25	Deployment	11/3/2023	11/7/2023	3d

Figure 2 Gantt chart

CHAPTER TWO: BUSINESS AREA ANALYSIS AND REQUIREMENT DEFINITION

2.1. Overview

This chapter provides an in-depth business area analysis of Two prominent online learning platforms: Coursera and Udemy. Coursera and Udemy enjoy global recognition and usage. The initial section talks about the comprehensive functionalities and operations of Coursera, Udemy. It examines their basic activities. Furthermore, it utilizes the PIECES framework to define the prevalent issues within the current systems, focusing on Performance, Information, Economic viability, Control mechanisms, Efficiency, and Services.

Moreover, this chapter outlines the various enrollment procedures, quizzes and assignment formats, feedback mechanisms, and the teachers' platform for content uploading across these platforms.

In addition, it explores the diverse stakeholders involved in the existing systems, delineating their roles and contributions, thus presenting a comprehensive overview of the players within this domain.

The latter part of this chapter emphasizes the foundational aspects by mentioning both the functional and non-functional requirements necessary for these platforms to operate effectively.

2.2. Business area analysis

Business analysis is identifying business needs and determining solutions to business problems. Solutions often include a software-systems development component, but may also consist of process improvement; organizational change or strategic planning and policy development. This may include examining elements of the organization structures and staff development issues as well as current processes and IT systems. (Hass, Horst, & Ziemski, 2008).

2.2.1. Activities/functions of the organization

E-learning platforms, Udemy and Coursera.com are among e-learning platforms that are currently working and used by many Ethiopians. They do not offer courses in Ethiopian languages, lack a convenient payment method tailored for Ethiopian students, and are deficient in essential social interaction features crucial for an effective teaching and learning process. Our project aims at developing an e-learning and social interaction website where Ethiopians can access coding-related courses in their own language, along with a convenient payment method. Functions of these e-learning are more or less generic; conducting requirement analysis on each of these organizations is expensive at this time. Consequently, we have chosen two e-learning Service Providers for conducting requirement analysis and other activities throughout this project, the first one Udemy and the second one is Coursera. We chose the two websites because they have a well-established working system and are used by most Ethiopians today.

Activities/Functions of Udemy

Udemy is a global online learning platform that provides a vast array of courses on various subjects taught by expert instructors. Founded in 2010 by Eren Bali, Gagan Biyani, and Oktay Caglar, Udemy has become one of the largest and most popular e-learning platforms worldwide. (Wikipedia)

Udemy offers a wide range of courses covering diverse subjects, including technology, business, finance, marketing, personal development, arts, languages, and more. These courses cater to learners at different skill levels, from beginners to advanced.

Udemy offers a mix of free and paid courses. Paid courses often have pricing set by the instructors and can vary widely based on factors such as course content, duration, and instructor expertise. Udemy frequently offers discounts and promotions on course prices. Udemy offers free courses and discounts through promotional campaigns, instructor giveaways, and online platforms and forums.

Enrolling in a course on Udemy involves a straightforward process. First, after creating an account on the Udemy website or app, users can explore a vast array of courses by browsing categories or using search filters. Upon finding a course of interest, users can view its details, including the curriculum, instructor information, and reviews. When ready to enroll, clicking the "Enroll" or "Buy Now" button initiates the process. Users proceed to the payment page where they can select a payment method and complete the purchase. Once enrolled, the course content becomes accessible immediately, offering a variety of learning materials such as video lectures, quizzes, and downloadable resources. Learners can progress through the course at their own pace, track completion, and engage with the content. Throughout the learning journey, Udemy provides support, progress tracking, and the flexibility to revisit completed sections (udemy.com)

Activities/Functions of Coursera

Coursera is a leading online learning platform that partners with top universities and organizations worldwide to offer a wide range of courses, specializations, and degree programs. Founded in 2012 by Andrew Ng and Daphne Koller, Coursera aims to provide universal access to high-quality education, making learning more accessible, flexible, and affordable for learners globally.

Coursera offers a vast array of courses spanning various disciplines, including technology, business, humanities, health sciences, social sciences, and more. These courses cover topics suitable for learners at different levels, from beginners to advanced professionals.

To enroll in a Coursera course coursera allows first, sign up on the Coursera website or app using an email address or social media account. Explore the wide range of courses available, select a course that matches your interests, and review its details including the syllabus and instructor information. Click the "Enroll" or "Start Free

Trial" button and choose your enrollment option—auditing for free or enrolling in a paid course for additional features. If it's a paid course, proceed to payment by selecting a payment method and entering your details. Once enrolled, access the course materials immediately, engage in video lectures, quizzes, and assignments at your own pace, and track your progress. Upon completion, earn a certificate of completion or a shareable certificate if applicable, marking your achievement in the course.

2.2.2. Problem of current system with framework

We will be using PIECES framework to describe the existing problems of the current system. This framework is used to identify the problems within an existing system. These problems can be seen from six different perspectives like Performance, Information, Economic, Control, Efficiency and Services given by the system to users.

Performance

Coursera and Udemy offers a vast number of courses across different subjects. Navigating through this extensive library overwhelm users, leading to difficulties in finding specific courses or relevant information due to the volume of options available.

Information

Coursera and Udemy offers a vast array of courses across different subjects. These information overload leading to difficulties in finding specific courses or relevant information due to the volume of options available.

Students in coursera and Udemy also face lack of information that they require to understand course contents.

Control

Udemy and Coursera face difficulties in overseeing instructors' content to ensure alignment with particular countries' cultures. This presents a challenge for them in controlling the diversity and cultural relevance of the materials offered on their platforms.

Service

Udemy and Coursera lack direct communication services allowing students and teachers to chat within the platform.

Economical

Limited payment option and expensiveness of the courses restrict Ethiopian students' access to these platforms. This stops more people from using the sites and makes it tough for the sites to grow in Ethiopia. It also means Ethiopian students might not get good education chances from these platforms.

Efficiency

Udemy and Coursera are less efficient in terms of making students understand content quickly and easily because they don't offer chatting features for direct communication between instructors and students, hindering student understanding.

2.2.3. Forms and reports of the current system

in this section, we've listed the types of forms and reports found on two popular e-learning platforms, Coursera and Udemy.

I The initial form serves as a means for students to enroll in a course and subsequently provide feedback following enrollment.

II The second form serves as a means for instructors to upload course content.

Udemy employs various forms to facilitate student enrollment in courses, process payments, collect feedback, and administer assessments. These forms are listed below:

Table 2.1 Student enrollment form

Name of field	Description
Full name	Users are typically asked to provide their full name.
Email	A valid email address is required for communication and account verification.
Password	Users create a password to secure their accounts.

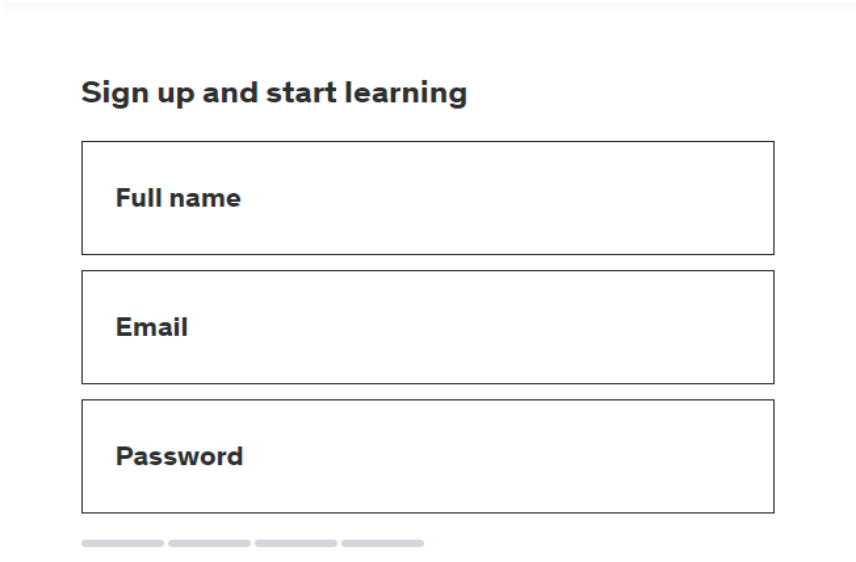

Figure 3 Udemy student enrollment form

In order to enroll in a course, students are required to pay the course fee. Here is a payment form for students.

Table 2.2 Payment form

Name of field	Description
Payment methods	Users are typically required to choose a payment method. Common options include credit/debit cards and PayPal.
Debit card/credit card information	if users choose to pay with a credit or debit card, they need to enter the card number, expiration date, and security code (CVV)
Billing address	Users are often required to provide the billing address associated with the chosen payment method.
Gift card or coupon card	Users have the option to apply coupon codes or use gift cards to avail of discounts on the course price.

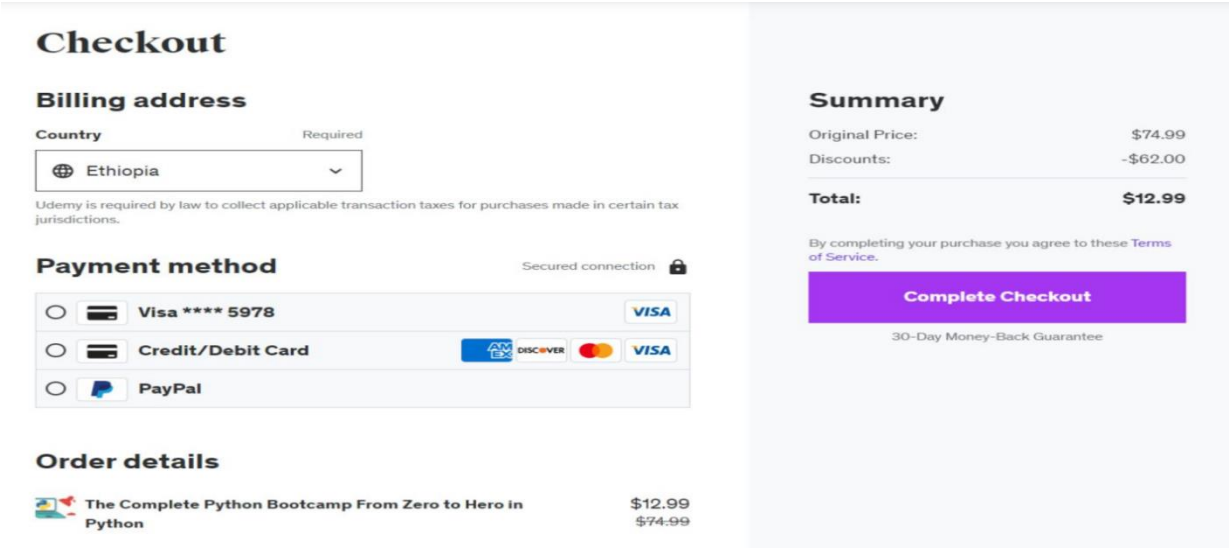

Figure 4 payment form for udemy

In this form student can give feedback about the course contents through the field's list below.

Table 2.3 Feedback form

Name of field	description
Rating	Users are often presented with a scale, usually in the form of stars (e.g., 1 to 5 stars). Each star represents a level of satisfaction,
Written Review	A text area or comment box is provided for users to enter their written feedback. This allows users to provide more detailed comments about their experience with the course.

Table 2.4 Administer Assessments

Name of fields	**description**
Text Input	For answering open-ended questions or providing explanations.
Multiple-Choice	Users choose the correct answer from a set of options.
File Upload	For submitting assignments or additional materials

Table 2.5 Udemy teachers form when uploading course content

Name of fields	description
Title	Enter the title of your course. This should be a clear and concise representation of what your course is about.
Description	Provide a detailed description of what students will gain from your course. Highlight key topics, skills, and the overall value of the course.
Subtitle	If applicable, add a subtitle that complements your course title, providing additional context or emphasis.
Course Image (Thumbnail)	Upload a visually appealing thumbnail image for your course.
Preview Video (Entry Video to Your Course)	Describe the content and purpose of the preview video. This is the introductory video that potential students will see before deciding to enroll in your course.
Instructor Profile	Introduce yourself as the instructor. Highlight your expertise, qualifications, and why students should trust you to teach the course.

Submit for Review	check the appropriate box to submit your course for review by Udemy.

Table 2.6 Coursera student enrollment form

Personal Information	
Country:	country of residence.
Time Zone:	time zone that corresponds to location.

Account Information	
Full Name:	Provide your full name.
Email Address:	Use a valid email address as this will be primary means of communication with Coursera.
Password:	password for Coursera account.

Profile Information	
Profile Picture:	Optionally, upload a profile picture.
Bio:	an option to write a short bio or introduction.

Payment Information	
Credit Card Information:	If a student is enrolling in a course that requires payment, enter credit card details.

Enrollment Information	
Choose Course:	Browse or search for the course a student want to enroll in.
Enrollment Option:	Select whether a student want to audit the course for free or pay for a certificate or full access.

Additional Options (if applicable)	
Financial Aid Application:	Coursera may offer financial aid for some courses.

Payment form

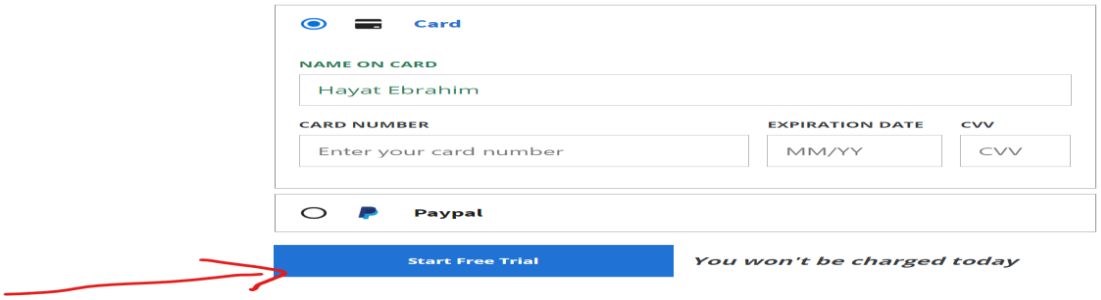

Figure 5 payment form for Coursera

Reports of the current system

Udemy offers reports to help students track their course progress.

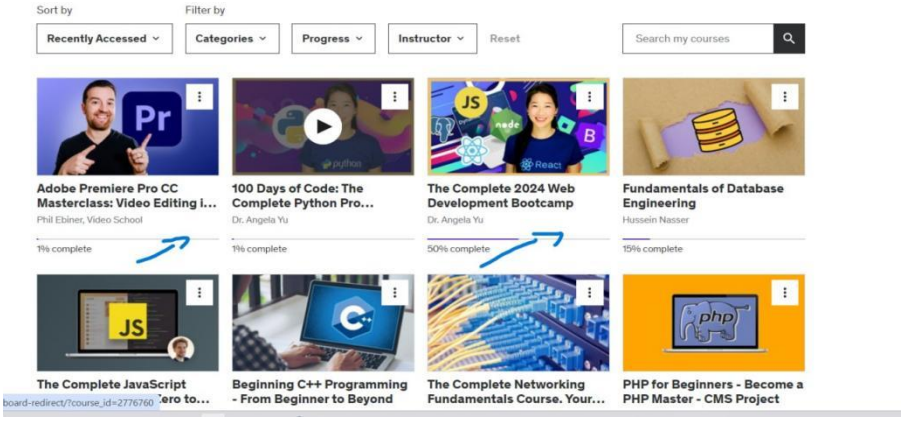

Figure 6 Report for Udemy

Coursera reports for the student to track course Coursera progress

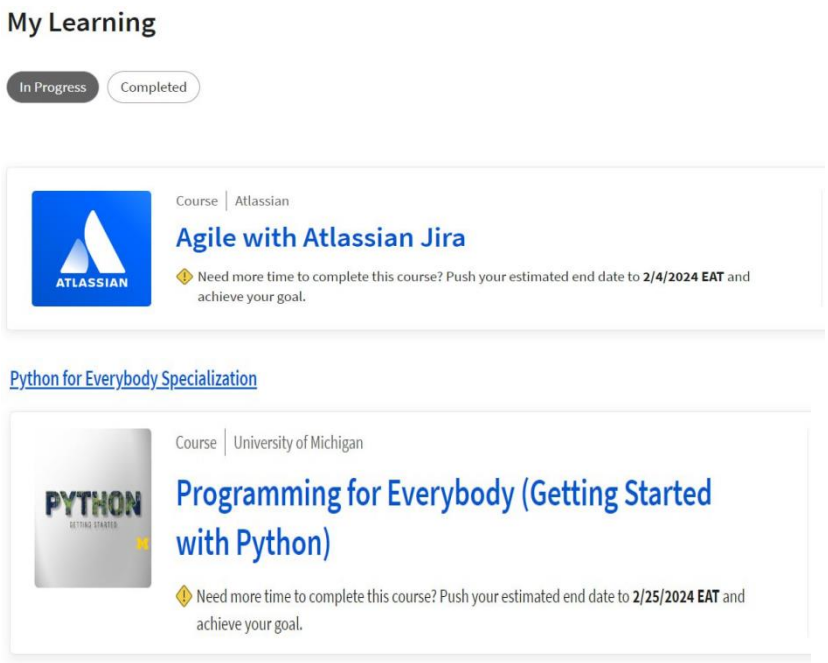

Figure 7 report for Coursera

2.2.4. Players of the existing system

Players represent external entities that interact with the system and are involved in the activities of current system. In our project, we have identified the following players.

Players of the existing system

1. student: Users who enroll in and take courses on the system are referred to as students. These students can do the following activities.

Course Enrollment, Watching Video Lectures, Participating in Discussions, Providing Reviews and Ratings, Downloading Resources, Utilizing Certificates of Completion.

2. Instructors: users who create and teach courses are referred to as instructors.

Instructors do the following activities

Course Creation: instructors create courses. They plan the course structure, outline the content, and develop materials such as video lectures, quizzes, assignments, and supplementary resources. Designing Assessment and Feedback: Instructors design quizzes, assessments, and assignments to evaluate students' understanding of the material. They provide feedback on student work and address common questions or misconceptions, Course Management: Instructors manage their courses on the platform, including setting up course structure, creating and editing content, and responding to student inquiries, Creating Supplementary Resources

3. bank system: links to a bank system for financial transactions such as payments and refunds. A service that facilitates the transfer of funds between a customer's bank and the platform.

4. reviewers: review team will provide feedback on various aspects of the course, such as content quality, video and audio clarity, and compliance with the platforms policies.

5. Support Team: support team, which assists both instructors and students with platform-related inquiries, technical issues, and other support needs.
6. administrators: play a crucial role in maintaining the integrity and functionality of the online learning platform. They are responsible for tasks such as reviewing course content to ensure compliance with policies, providing user support to both instructors and students, addressing technical issues and bugs, enforcing platform policies, communicating updates to users, and maintaining the overall quality of courses.
7. accountants: Accountants would maintain accurate and up-to-date financial records, tracking transactions, expenses, and revenues related to the operation of the platform.

2.3. Requirements Definition

2.3.1. Functional requirement

Online courses are here and there these days especially when it comes to contents related to coding and other technical topics. There are, of course, books and videos everywhere. But these courses are not making Ethiopians capable and the reason is the way these online materials are being presented that can be the language they use, or the way they interact with the learners. Enimar code learning and hub platform is planned to solve the problems faced by Ethiopian to learn coding. Ethiopians will be able to access contents online through easily available online payments and the courses will be in the local languages they learn. Learners will easily contact their teachers, by following, messaging and liking their teachers' contents. Teachers having their own publicly accessible account will have a freedom of posting free and paid courses so that they can get real time feedback and reactions from learners.

Table 2.7 Functional requirement Table

ID	NAME	DESCRIPTION
1	User Registration	The system facilitates the creation of user accounts for users of the system by collecting and verifying necessary information.
2	Show Course Catalog	Students can access a comprehensive course catalog, featuring course titles, detailed descriptions, and instructor information.
3	Enroll Course	Students have the ability to enroll in courses of their choice, and the system securely stores the enrollment details.
4	Process Payment	The system integrated with Ethiopian payment systems, allowing students to conveniently make course fee

ID	NAME	DESCRIPTION
		payments.
5	Evaluate assessment	The system evaluates the assessments of the course
6	Track Progress	Students are empowered to monitor their course progress, including completed modules, grades.
7	Automated Assessment	The system employs automated assessment techniques to evaluate student performance through quizzes, providing prompt grades and feedback.
8	Issue Certification	Upon successful completion of courses, the system generates professional certificates containing essential details such as course name, completion date, and instructor endorsement.
9	Management Course	Instructors can easily upload and manage course content.
10	Manage system functionalities	Administrators are equipped with a centralized admin dashboard, empowering them to efficiently manage user accounts, monitor system performance, and address any issues that may arise.
11	Give Feedback	Students are provided with a user-friendly feedback mechanism, enabling them to share valuable insights and suggestions regarding courses and instructors.
12	Review Course	The system incorporates a comprehensive course review process, ensuring the quality and relevance of

ID	NAME	DESCRIPTION
		courses created by teachers before they are made available to students.
13	Instant Messaging	The system offers a robust communication platform, fostering seamless interaction between students and instructors, and students and students.
14	social interactions	The system must include robust social interaction capabilities, allowing users to post content, follow other users, like posts, and comment on shared content.

2.3.2. Non-functional requirement

Non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors

1. Performance:

- Response Time: The e-learning platform should have a maximum response time 2 seconds for user interaction. For example, clicking a button to start a video lecture, The video should begin playing within 2 seconds of clicking the button.

- Scalability: The platform should be able to support a simultaneous user load of at least 10,000 users without significant performance degradation.

2. Reliability:

- Availability: The e-learning platform should be available to users 99.9% of the time, excluding scheduled maintenance windows.

- Fault Tolerance: The system should be able to recover from a server failure within 5 minutes without data loss.

3. Security:

- Authentication and Authorization: Users must authenticate using a secure, multi-factor authentication process. Role-based access control should be implemented for content and administrative features.

- Data Encryption: All user data, including login credentials and personal information, should be encrypted during transmission and storage using industry-standard encryption algorithms.

4. Usability:

- User Interface Design: The user interface should follow the principles of responsive design, ensuring a consistent and user-friendly experience across different devices and screen sizes.

- Accessibility: The platform should comply with WCAG 2.0 standards, providing accessible features for users with disabilities.

5. Scalability:

- Load Handling: The platform should handle a 20% increase in user load during peak hours without degradation in performance.

- Concurrency: The system should support a minimum of 1,000 concurrent users without a significant impact on response time.

6. Interoperability:

- Integration with Other Systems: The e-learning platform should support integration with popular Learning Management System (LMS) standards such as SCORM and Tin Can API.

- Data Exchange: The platform should allow seamless import and export of course content in standard formats (e.g., XML, CSV).

7. Maintainability:

- Modifiability: The codebase should follow modular design principles to facilitate easy modification and extension. Changes to the system should not require extensive retesting of unrelated components.

- Documentation: Comprehensive technical documentation, including API documentation and user manuals, should be maintained and updated with each release.

8. Compatibility:

- Browser and Device Compatibility: The platform should be compatible with the latest versions of major browsers (Chrome, Firefox, Safari, Edge) and support a responsive design for various devices (desktops, tablets, smartphones).

- Operating System Compatibility: The platform should be compatible with major operating systems (Windows, macOS, Linux).

9. Backup and Recovery:

- Data Backup: User data should be backed up daily, and backup integrity should be regularly tested.

- Disaster Recovery: A documented disaster recovery plan should be in place, specifying procedures for restoring the platform in the event of a catastrophic failure.

10. Training and Support:

- User Training: The platform should provide comprehensive online training modules for both instructors and students.

- Customer Support: A 24/7 customer support system should be available, with response times not exceeding 4 hours for critical issues.

CHAPTER THREE: OBJECT ORIENTED ANALYSIS

3.1. Overview

This chapter focuses on analyzing the business area to determine the requirements for developing a new system within our Enimar Code Learning Platform. The goal is to understand what needs to be built. During this phase, we will engage in various activities such as identifying user interface elements, defining business rules, and modeling system use cases. However, it's important to note that the modeling process will be influenced by the technology chosen for the development of our platform.

In the context of our Enimar Code Learning Platform, we will employ Object-Oriented Analysis (OOA). This methodology involves identifying the requirements of software engineering and specifying them in terms of an object model. The object model consists of interacting objects that combine data and functions. These objects are inspired by real-world entities that our system interacts with. Unlike traditional analysis approaches that consider functions and data separately, the object-oriented approach integrates both aspects. The primary objective is to create a conceptual model of the information relevant to our code learning platform.

By employing OOA within our Enimar Code Learning Platform, we can effectively organize requirements around objects, leading to a more comprehensive understanding of the problem domain. This approach will help us develop a conceptual model that accurately represents the information and processes within our platform.

3.2. Use case Modeling

Use case modeling is a technique used in software engineering to visually represent and document the interactions between users (actors) and a system. It is a part of the Unified Modeling Language (UML), a standardized modeling language widely used in software development. Use case diagrams provide a high-level view of a system's functionality from the perspective of its users.

3.2.1. UI identification

Table 3.1 UI identification

UID	User Interface	Description
UI 01	Login Page	Page the user is authenticated
UI 02	Account creation Page	Page where Teacher creates account
UI 03	Account creation Page	Page where Student creates account
UI 04	Home page	Home page for Teacher
UI 05	Home page	Home page for Student
UI 06	Accounting dashboard	Page where Accountant views Report
UI 07	Teacher Profile	Profile for Teacher

UI 08	Student Profile	Profile page for Student
UI 09	Reviewer Dashboard	Page where Reviewer reviews courses
UI 10	Admin Dashboard	Administrators page
UI 11	Enrolled Course list page	Page where Student sees list of course he has enrolled in
UI 12	Payment Page	Page to make payment
UI 13	Course report Page	Page where Teacher views courses he has created
UI 14	Notification page	Page where users see notifications
UI 15	Chat Page	Page where user chat with each other
UI 16	Course Creation Page	Page where Teacher creates course
UI 17	Post page	Page that shows single Post

3.2.2. Business rules identification

Identification	Business Rule	Description
BR01	Many Admin	There are many admins in our system
BR02	Complete Form	Non optional fields in of our system forms need to be filled to proceed
BR03	Students Must pay	Students must first pay to access courses
BR04	Only Posts are allowed	Making post but not creating courses are allowed for students
BR05	Course reviewal	Courses must undergo a review process before being published.
BR06	Agreeing to terms	All users are required to agree to the terms and conditions of the system.
BR07	Educational content only	Contents out of Education(political,religious,racial)contents are forbidden

3.2.3. Actor identification

1. Teacher: The Teacher is responsible for delivering educational content, creating courses, assessing students, and providing guidance and feedback within the e-learning system.
2. Student: The Student engages in learning activities, accesses course materials, participates in assessments, tracks progress, and interacts with teachers and peers within the e-learning system.
3. Reviewer: The Reviewer evaluates the quality and effectiveness of course content, provides feedback to teachers.

4. Admin: The Admin manages user accounts, system settings, and resolves technical issues to ensure the smooth operation of the e-learning platform.

5. Accountant: The Accountant handles financial transactions, manages course fees, and payment processing.

3.2.4. Designing the use case diagram

A use case diagram is a type of diagram in the Unified Modeling Language (UML) that illustrates the interactions between actors (users or external systems) and a system.

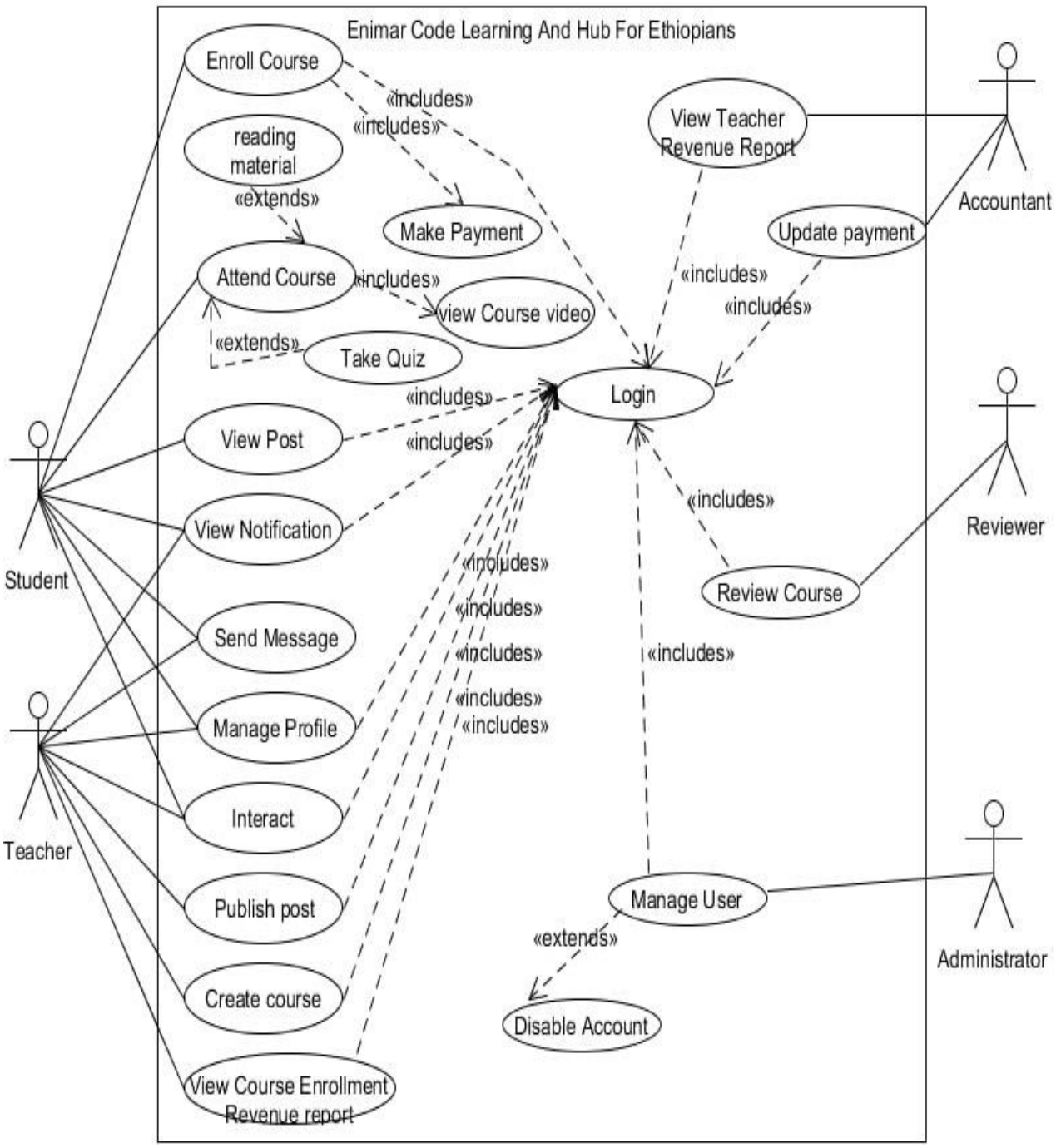

Figure 8 use case diagram

3.2.5. Use case description

A use case description is a detailed and narrative document that provides a comprehensive understanding of how a specific use case within a system functions. It outlines the interactions between various actors (users or external systems) and the system itself.

Table 3.2 Use case description

Name	Create course
Identifier	UC01
Description	A teacher can create a course
Actor	Teacher
Pre-Condition	A teacher must have an account
Extends	None
Includes	Login
Basic Course of Action	1 The use case starts when the teacher clicks on create course button 2. The system displays course creation page: UI 16 3. When teacher finished filling the required fields and if teacher wants to continue, he/she submits the form 4. The system creates a course 5. The use case ends
Alternative Course of Action	A3. If the teacher doesn't want to continue: A3.1. the teacher clicks on cancel button A3.2. The system will display the home page.
Post Condition	The course status will be draft and send to the reviewers

Name	Review course
Identifier	UC02
Description	The reviewer reviews course content and approve the drafted courses or either gives comment so that owner fixes them
Actor	Reviewer
Pre-Condition	The reviewer must create an account
Extends	None
Includes	Login
Basic Course of Action	1. The system displays home page for the reviewers 2. The reviewer clicks on specific course that he/she assigned to review 3. If the course meets the criteria specified in business rule BR07 reviewer changes the course status to published. 4. The system will get the evaluation result from the reviewer. 5. If the course is approved, the system will send notification to the student who follow that teacher. 6. The use case ends
Alternative Course of Action	A4 if the course doesn't meet the criteria specified in business rule A4.1 The course will be discarded. A4.2 The system will notify the teacher about the cancelation of the course
Post Condition	Course reviewed successfully

Name	**Follow public accounts**
Identifier	UC03
Description	Student or teacher can follow the other public accounts
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	Interact
Includes	Login
Basic Course of Action	1. The user wants to follow public accounts 2. The user clicks on the student/teacher's tab on the home page 3. The system display's a list of students /teachers 4. The user clicks on one of the students/teachers he/she wants to follow 5. The system displays the student's/ teacher's profile 6. The user clicks on the follow button 7. The system changes follow button to unfollow button 8. The user successfully follows the account 9. The use case ends
Alternative Course of Action	None
Post Condition	The student can get updates from teachers/students update on courses and posts.

Name	**Unfollow public accounts**
Identifier	UC04
Description	Student or teacher can unfollow other public accounts
Actor	Student, Teacher
Pre-Condition	A user must have followed an account
Extends	Interact
Includes	Login
Basic Course of Action	1. The user wants to unfollow public accounts 2. The user navigates to the student/teacher's tab on the home page. 3. The system displays a list of students/teachers the user is currently following. 4. The user selects the student/teacher they want to unfollow. 5. The system displays the student's/teacher's profile. 6. The user clicks on the "Unfollow" button. 7. The system prompts for confirmation or provides feedback about the successful unfollow action. 8. The user successfully unfollows the account. 9. The use case ends.
Alternative Course of Action	
Post Condition	The student stops getting updates from teachers /students.

Name	**Leave comment**
Identifier	UC05
Description	Student or teacher can leave comment for a post
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	Interact
Includes	Login
Basic Course of Action	1. The user wants to leave comment 2. The user selects comment button on a specific post 3. The system displays the comment box 4. The user writes his/her comment on the comment box 5. The system displays the comments on the comments list 6. The use case ends
Alternative Course of Action	
Post Condition	Comments uploaded and listed successfully

Name	**Publish Post**
Identifier	UC06
Description	Teacher and student can create post which may contain video, text and photo
Actor	Teacher, Student
Pre-Condition	A Teacher must have an account
Extends	None
Includes	Login
Basic Course of Action	1. Teacher or Student want to publish Post 2. The system displays home page 3. The teacher clicks on create post button 4. The system displays the form to create the post 5. Teacher or Student fills out the form with the necessary information and content. 6. Teacher or Student submits the completed form. 7. The system provides feedback or a notification confirming the successful publication of the post. 8. The use case ends
Alternative Course of Action	
Post Condition	Post published successfully

Name	**Share post**
Identifier	UC07
Description	Student or teacher can share posts
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	Interact
Includes	Login
Basic Course of Action	1. User wants share to a post 2. The user clicks on the "Share" icon associated with the chosen post. 3. The system makes the post visible to people who follow the user that shared it. 4. The use case ends.
Alternative Course of Action	
Post Condition	Posts shared successfully

Name	**Manage Profile**
Identifier	UC08
Description	Student or teacher can manage their profiles
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on the profile found on the header of the page. 2. The system displays a list of options to manage their profile. 3. The user selects on the specific option. 4. The system will display a form based on that option. 5. The user fills the form and clicks on save button. 6. The system will save the changes. 7. The use case ends.
Alternative Course of Action	
Post Condition	Profile updated successfully

Name	**View notification**
Identifier	UC09
Description	Student or teacher can view notification
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on notification icon 2. The system displays a list of notifications. 3. The use case ends.
Alternative Course of Action	None
Post Condition	Notifications viewed

Name	View message
Identifier	UC10
Description	Student or teacher can view messages
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on the chats button. 2. The system displays list of chats. 3. The user selects a specific chat 4. The system notifies the sender that the message has been seen if it hasn't been seen already. 5. The use case ends.
Alternative Course of Action	
Post Condition	Message viewed successfully

Name	Send message
Identifier	UC11
Description	Student or teacher can send messages
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	None

Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on message icon of a specific public account. 2. The system will check if the user is blocked or not 3. The system will display a text input field. 4. The user writes a message 5. The user clicks on a send icon 6. The use case ends.
Alternative Course of Action	A2 If the user is blocked A2.1 The system will display disabled text input field
Post Condition	A message sent successfully

Name	View Course Enrollment and Revenue Report
Identifier	UC12
Description	The teacher can view reports detailing the enrollment numbers for each course and the monthly revenue earned.
Actor	Teacher
Pre-Condition	The teacher must have created a course
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the teacher clicks on reports button. 2. The system displays a list of courses he/she has created. 3. The teacher clicks on specific course. 4. The system displays a page that contain information about number of students enrolled and the monthly revenue earned in that course. 5. The use case ends.

Alternative Course of Action	
Post Condition	Course Enrollment and Revenue Report viewed successfully.

Name	View teacher revenue report
Identifier	UC13
Description	The accountant can view reports detailing the monthly revenue earned by each teacher.
Actor	Accountant
Pre-Condition	An accountant must have an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the accountant clicks on reports page. 2. The system displays a report for each teacher. 3. The use case ends.
Alternative Course of Action	
Post Condition	Teacher revenue report displayed

Name	View certificate
Identifier	UC14
Description	The student can view a certificate that he/she got for a given course
Actor	Student
Pre-Condition	The student must have completed a course
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the student clicks on view certificate button of a specific course. 2. The system checks if the student has completed the course 3. The system displays the certificate. 4. The use case ends.
Alternative Course of Action	A2 if the student hasn't completed the course A 2.1 the system will display that the course has not completed yet message.
Post Condition	The student viewed certificate

Name	**Like posts**
Identifier	UC15
Description	Student or teacher can like post
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	Interact
Includes	Login

Basic Course of Action	1. The use case starts when the user clicks a like icon displayed under a post 2. The system changes the color of the like icon 3. The system increases the number of likes by one 4. The use case ends
Alternative Course of Action	
Post Condition	The student liked the post successfully

Name	**Rate course**
Identifier	UC16
Description	Student can rate a course
Actor	Student
Pre-Condition	A student must first enroll in a course
Extends	Interact
Includes	Login
Basic Course of Action	1. The use case starts when the student clicks on a desired number of stars displayed under a specific course 2. The system changes the color of the selected stars 3. The system registers the rate given by the student for that course 4. The use case ends
Alternative Course of Action	
Post	The student rated the course successfully

Condition	

Name	**view course video**
Identifier	UC17
Description	Student can view a selected course video
Actor	Student
Pre-Condition	A student must have enrolled in a course
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on a specific course video 2. The system plays the video 3. The system updates the course progress 4. The use case ends
Alternative Course of Action	
Post Condition	The student views the video

Name	**Read reading material**
Identifier	UC18
Description	Student can read a selected reading material
Actor	Student
Pre-Condition	A student must have enrolled in a course

Extends	Attend Course
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on a specific reading material 2. The system opens the reading material 3. The system updates the course progress 4. The use case ends
Alternative Course of Action	
Post Condition	The student read the reading

Name	Take quiz
Identifier	UC19
Description	Student can take a selected quiz
Actor	Student
Pre-Condition	A student must have enrolled in a course
Extends	Attend Course
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on a specific quiz 2. The system displays list of questions with choices to be selected 3. Student selects one choice as an answer for each question and click submit 4. The system display the result for the user 5. The use case ends
Alternative Course of Action	
Post Condition	The student taken the quiz

Name	Attend Course
Identifier	UC20
Description	Student starts learning the contents of the course
Actor	Student
Pre-Condition	A student must have enrolled in a course
Extends	

Includes	view course video, Login
Basic Course of Action	1. The use case starts when the user clicks on go to course button of a specific course 2. The system display list of course videos, files and assessments 3. The use case ends
Alternative Course of Action	
Post Condition	The student is attending the course

Name	**Enroll in course**
Identifier	UC21
Description	Student can enroll in a selected course
Actor	Student
Pre-Condition	A student must have an account
Extends	None
Includes	Make payment, Login
Basic Course of Action	1. The use case starts when the student clicks on the enroll button of a specific course 2. The system displays a form to verify first name and last name of a student 3. The student fills in the form and click on submit button 4. The system adds that student to list of enrolled students of that course and opens course content. 5. The use case ends
Alternative	

Course of Action	
Post Condition	The student enrolled in a course

Name	Update payment status
Identifier	UC22
Description	The accountant can update payment status of each course
Actor	accountant
Pre-Condition	The accountant must create an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the accountant clicks on the status button in front of a specific course. 2. The system displays a radio button labeled paid and un paid to the course in a specific month 3. The accountant selects one of the radio buttons 4. The system updates payment status of the course for a specific month 6. The use case ends
Alternative Course of Action	
Post Condition	Course payment status updated successfully

Name	Delete account
Identifier	UC23
Description	The administrator can delete an account
Actor	Administrator
Pre-Condition	The administrator must have an account
Extends	Manage User
Includes	Login
Basic Course of Action	1. The use case starts when the administrator clicks on the delete button in front of a specific account. 2. The system displays a confirmation message 3. The administrator selects button yes 4. The system will remove the selected account 5. The use case ends
Alternative Course of Action	A3 If the administrator selects a button NO A3.1 The system will be back to the previous state
Post Condition	Administrator deleted account successfully

Name	Disable account
Identifier	UC24
Description	The administrator can intentionally restrict or revoke access to the user
Actor	Administrator
Pre-Condition	The administrator must have an account
Extends	Manage Users

Includes	Login
Basic Course of Action	1. The use case starts when the administrator clicks on the disable button in front of a specific account. 2. The system displays a confirmation message 3. The administrator selects yes button 4. The system will disable the selected account 5. The use case ends
Alternative Course of Action	A3 If the administrator selects NO button A3.1 The system will be back to the previous state
Post Condition	The administrator disabled account successfully

Name	**Block public account**
Identifier	UC25
Description	Student or teacher can block public account
Actor	Student, Teacher
Pre-Condition	A user must have an account
Extends	Interact
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on one of the students, he/she wants to block 2. The system displays the student's/ teacher's profile 3. The user clicks on the block button 4. The use case ends
Alternative Course of	

Action	
Post Condition	The blocked user will stop getting updates from the user that blocked the account and the blocked user will get disabled to send and receive messages to and from the user that blocked the account.

Name	**Unblock public account**
Identifier	UC26
Description	Student or teacher can unblock blocked public accounts
Actor	Student, Teacher
Pre-Condition	The user must have an account and the account to be unblocked must have been blocked
Extends	Interact
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks on one of the blocked users, he/she wants to unblock 2. The system displays the student's/ teacher's profile 3. The user clicks on the unblock button 4. The use case ends
Alternative Course of Action	
Post Condition	The unblocked user will start getting updates from the user that unblocked the account and the unblocked user will get enabled to send and receive messages to and from the user that unblocked the account.

Name	Log in
Identifier	UC27
Description	A user can log in to an account created previously
Actor	Student, teacher, reviewer, accountant, administrator
Pre-Condition	A user must have an account
Extends	None
Includes	
Basic Course of Action	1. The use case starts when the student clicks on the log in button on the home page 2. The system displays a form 3. The user fills in the form and click on submit button 4. The system will check if the user inputs are valid 5. The system authenticates the user and displays a page associated to that user. 6. The use case ends
Alternative Course of Action	A4 If the inputs are not valid A4:1 The system displays wrong username or password Message
Post Condition	User logged in successfully

Name	Sign up
Identifier	UC28
Description	A user can create an account
Actor	Student, teacher, reviewer, accountant, administrator
Pre-	None

Condition	
Extends	None
Includes	
Basic Course of Action	1. The use case starts when the student clicks on the sign up button on the home page 2. The system displays a form 3. The user fills in the form and click on submit button 4. The use case ends
Alternative Course of Action	
Post Condition	Account created successfully

Name	**Unlike post**
Identifier	UC29
Description	Student or teacher can unlike post
Actor	Student, Teacher
Pre-Condition	A user must have an account and a user must have liked the post
Extends	Interact
Includes	Login
Basic Course of Action	1. The use case starts when the user clicks a like icon displayed under a post 2. The system changes the color of the like icon 3. The system decreases the number of likes by one 4. The use case ends
Alternative	

Course of Action	
Post Condition	The student unlike the post successfully

Name	Make payment
Identifier	UC30
Description	A student can make payment via different options available
Actor	Student
Pre-Condition	A student must have an account
Extends	None
Includes	Login
Basic Course of Action	1. The use case starts when the student clicks on the enroll button of a specific course or buy button of a specific package. 2. The system displays a list of payment modes available. 3. The student clicks on a specific payment mode 4. The system displays a form to confirm the payment 5. The student fills in the form and clicks on confirm button 6. The system validates the payment 7. If the payment is valid, the system informs that the payment information is correct to enroll in a course or purchase package. 8. The use case ends
Alternative Course of Action	7 A if the payment is not valid 7.1 The system displays a page that informs that the payment information is not correct.
Post Condition	The user paid successfully

3.3. Conceptual Modeling

Conceptual modeling is a process in systems engineering and software engineering that involves creating abstract representations of a system's concepts, entities, relationships, and behaviors. The goal of conceptual modeling is to provide a high-level and simplified view of a system, emphasizing its essential aspects without delving into implementation details.

3.3.1. Class diagram

A class diagram is a type of diagram in the Unified Modeling Language (UML) that represents the structure of a system by depicting the classes, their attributes, methods, and the relationships among them.

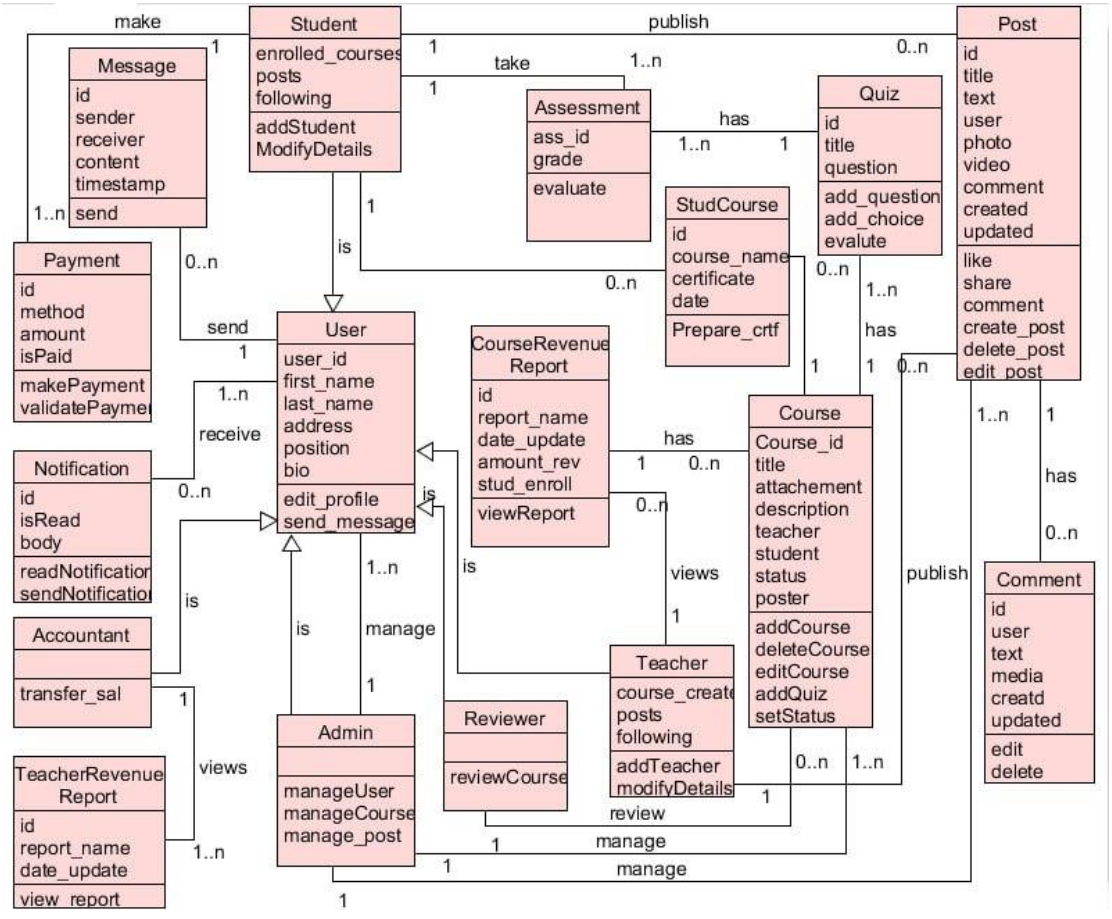

Figure 9 class diagram

3.3.2 Class description

1. **Post:**

   The Post class represents a generic post in the e-learning platform. It serves as the base class for different types of posts, including text, video, photo, and combinations of these. Each post has content (text, video, photo), an author (Either teacher or student), a timestamp indicating when it was created, and a list of comments, likes and shares too.

Attributes:

title: the title of the post being posted
user: the user responsible for the post
photos: photos attached with the post
video: video attached with the post
comments: list comments with the post
created: the date time the post was created
updated: the date time the post was last updated

Methods

like (): liking the post
share (): sharing the post by other account
comment (): commenting adds comment to the post
delete (): remove the post from profile
edit (): edit the contents of the post

2. Course:

   A class representing a course.

Attributes:

title (): The title of the course.
description (): A description of the course.
teacher (): teacher who published the course
status: the status of the Course either PUBLISHED or DRAFT

poster (): Photo representing a course as a poster

videos (): Photo representing a course as a poster

enrolled_users (): A list of users who have enrolled in the course.

3. Comment

   A class representing a comment on a post.

Attributes:

user (): The user who made the comment.

text (): The text content of the comment.

media (): Media content associated with the comment.

created (): The time the comment was first made.

updated (): The last time the comment updated.

Methods:

edit (): edit the comment to a new value.

add (): edit the comment to a new value.

delete (): delete the comment.

reply (): reply to the comment

4. Certificate:

   A class representing a certificate associated with a course.

Attributes:

course: The course the certificate is for.

student_id: the user the certificate is given

date_given : The course the certificate is for.

5. Assessment:

   A class representing an assessment associated with a course.

Attributes:

assessment_id : the id of the assesmment

quiz_id: the quiz associated with the assessment

grade: the grade the student will receive

Methods:

evaluate (): giving grade for the quiz

6. Quiz:

   A class representing an assessment associated with a course.

Attributes:

title (): The title of the quiz.

questions (): A list of questions for the quiz.

Methods:

add_question() : The title of the assessment.

add_choice() : The title of the assessment.

7. User:

   A class representing a user in the e-learning platform.

Attributes:

first_name (): The first name of the user.

last_name (): The last name of the user.

address (): The address of the user.

position (): The position or job title of the user.

bio (): A short bio or description of the user.

notifications (): list of notifications

Methods:

edit_profile(): Edits the user's profile information.

send_message(): sending message to other user

edit_profile (): editing profile

8. Student (User):

   A class representing a student, inheriting from the User class.

Attributes:

followers (): A list of users following this user.

following (): A list of users whom this user is following.

posts (): A list of posts created by the user.

enrolled_courses: number course the student is registered.

Methods:

create_post(): Creates a new text post.
make_payment(): making payment
enroll (): enroll in course

9. Teacher (User):
   A class representing a teacher, inheriting from the User class

Attributes:

followers (): A list of users following this user.
following (): A list of users whom this user is following.
posts (): A list of posts created by the user.
course_created (): the course the teacher posted

Methods:

create_post(): Creates a new text post.
follow (): follow other accounts
unfollow (): unfollow other account

10. Message:
    A class representing a message.

Attributes:

sender (): The user who sent the message.
receiver (): The user who will receive the message.
content (): The content of the message.
timestamp (): The timestamp indicating when the message was sent.

Attributes:

send_message: The user who sends the message.
read_message : The user who will receive the message.

10. Admin (User):

    A class representing an administrator in the e-learning platform.

    Inherits from the User class.

Methods:

    manage_users(): add, inactivate and update user accounts

    manage_courses(): Manage courses created by teachers

    mange_posts(): Manage posts by all users

11. Accountant (User):

    A class representing an accountant in the e-learning platform. Inherits from the User class.

Methods:

    pay_salary (): accountant pays the teacher

12. Payment:

    A class representing the Payment user can make

Attributes:

    amount: accountant process the payment made by the user

    method: the payment gateway user may use to pay

    is_paid: whether the payment is made or not

Methods:

    make_payment(): make payment

13. Reviewer (User):

    A class representing a course reviewer in the e-learning platform. Inherits from the User class.

Methods:

    review_course(): Reviews a course and provides feedback.

14. Report:

    A class representing the Report generated by the system

Attributes:

report_name: the name report is about

Methods:

view_report(): view financial reports generated by system

3.4. Sequence diagramming

A sequence diagram is a type of interaction diagram in the Unified Modeling Language (UML) that illustrates the dynamic interactions between objects or components within a system over time. It focuses on the flow of messages between different entities, typically representing objects or instances of classes, and shows the order in which these messages are exchanged.

Enroll in course sequence diagram

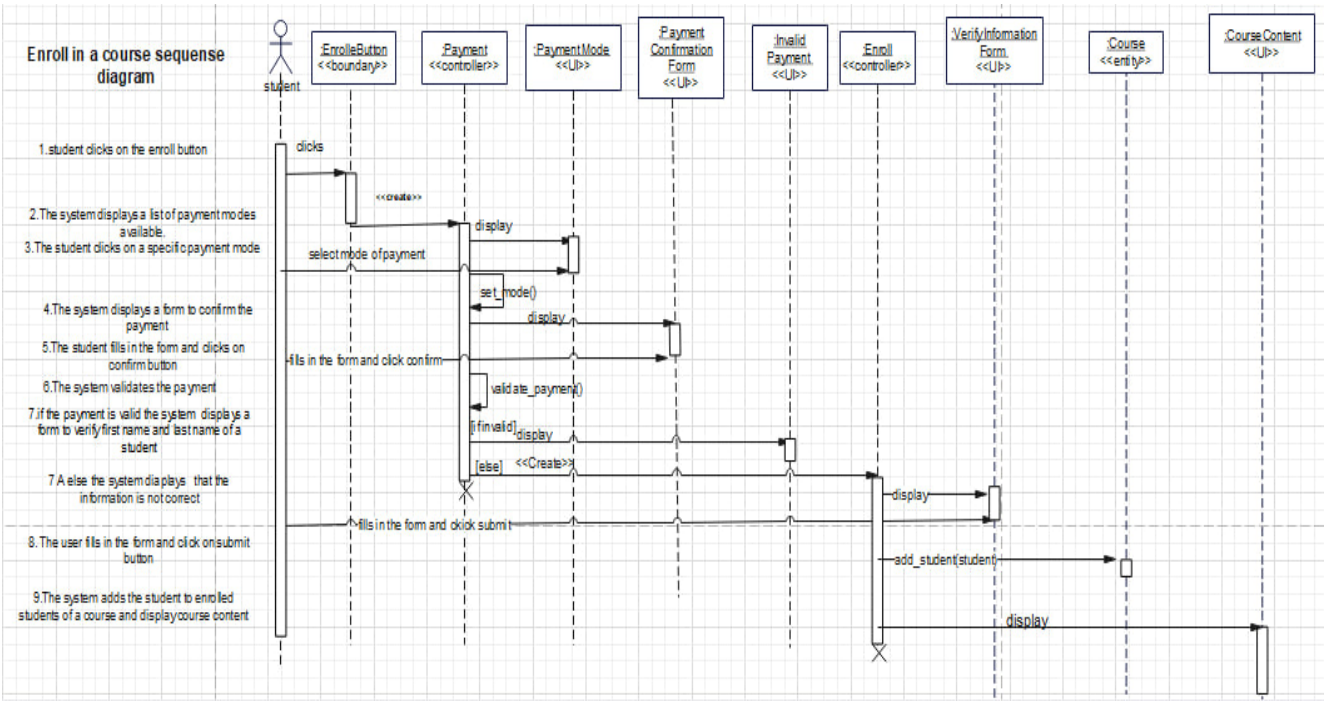

Figure 10 enroll course sequence diagram

Create course sequence diagram

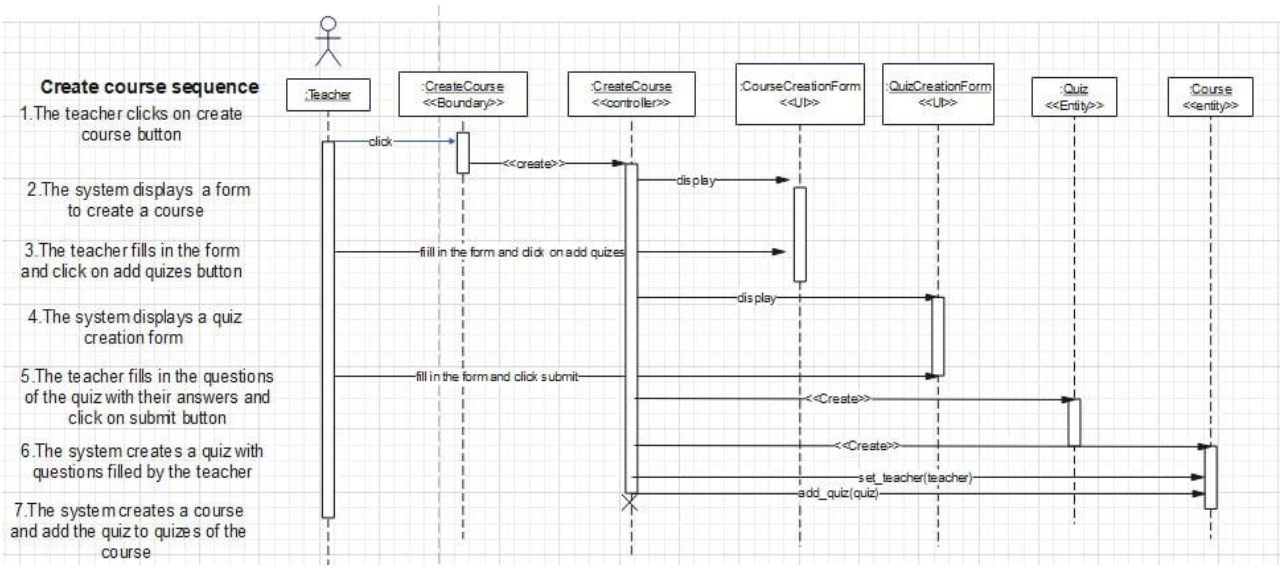

Figure 11 create course sequence diagram

Review course sequence diagram

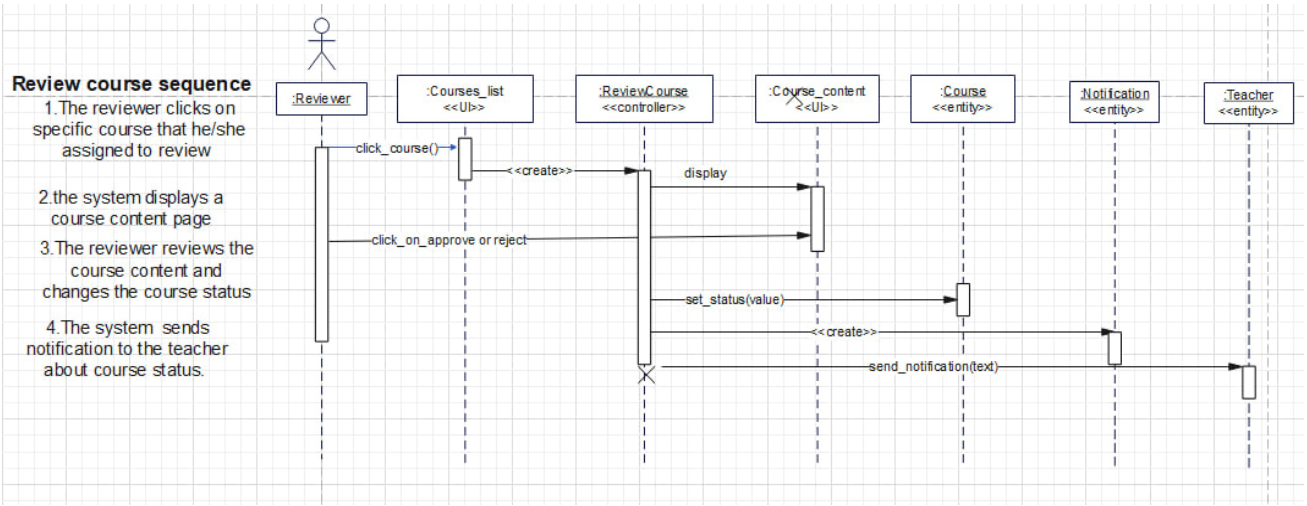

Figure 12 review course sequence diagram

3.5. User Interface Prototyping

User Interface (UI) prototyping is a design technique used in software development to create a visual representation of the user interface of a system before it is built.

Home page

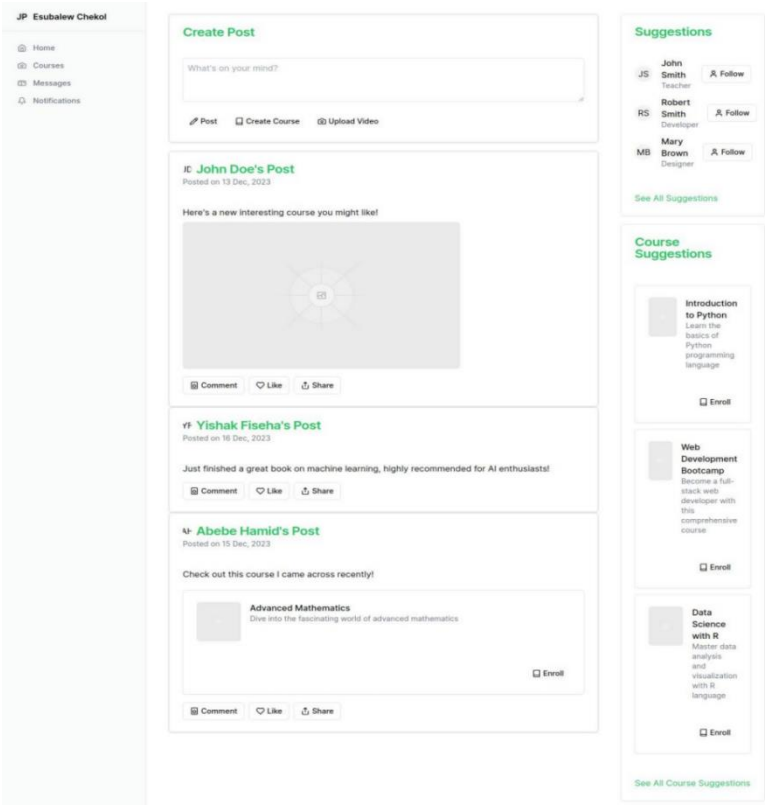

Figure 13 home page

Profile UI

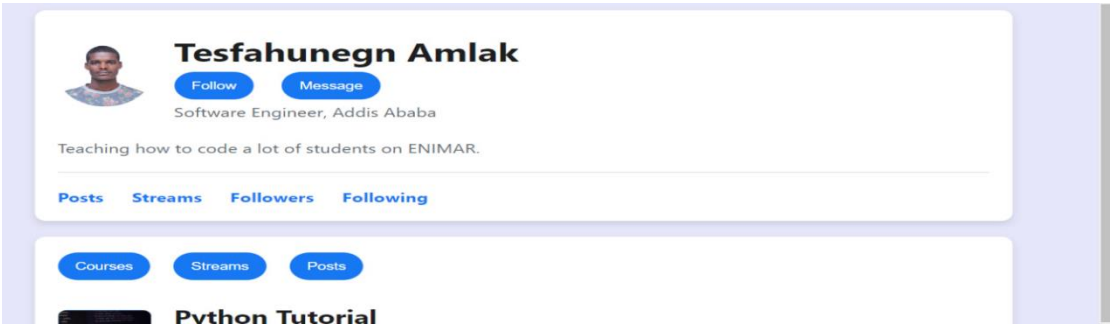

Figure 14 profile UI

Figure 15 purchase package

Login page

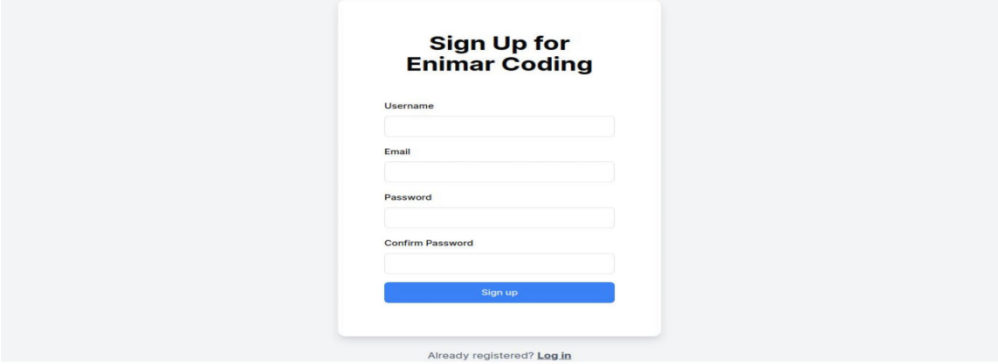

Figure 16 login

Chatting UI

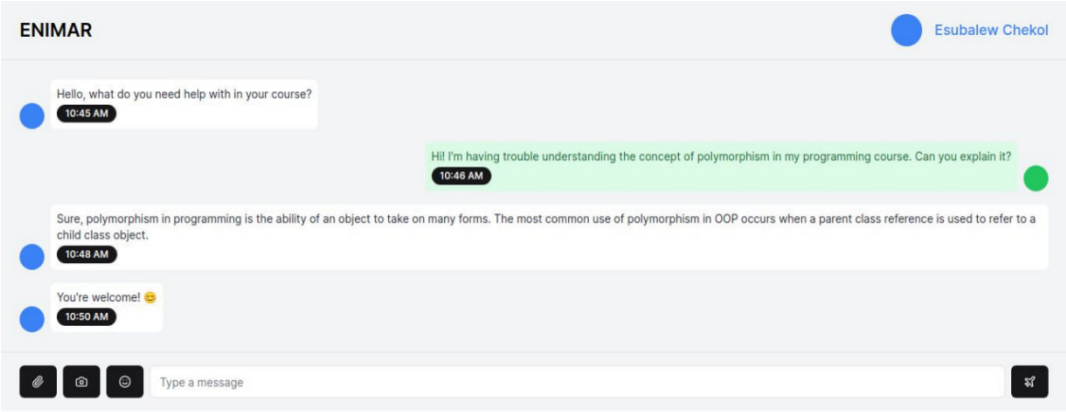

Figure 17 chatting

My courses UI

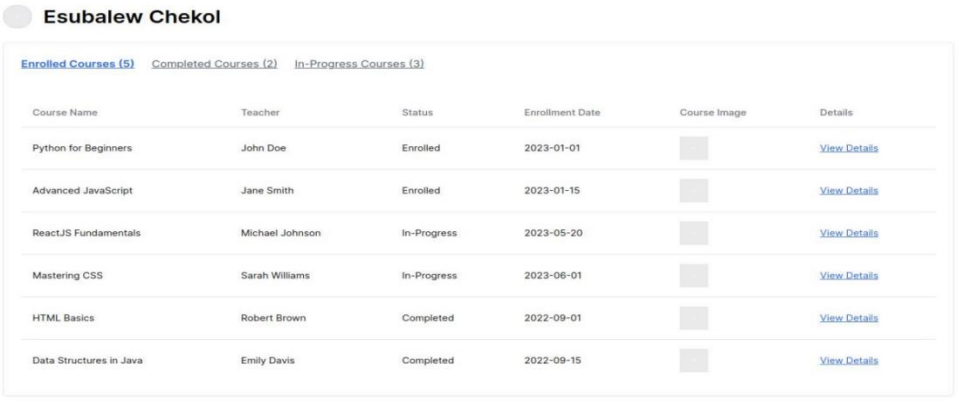

Figure 18 My Courses UI

CHAPTER FOUR: CONCLUSION

In chapter one, we discussed an introduction to the project of building an Enimar-learning website It begins with a background of the organization to provide context for the project. then proceeds to present the statement of the problem, outlining the specific challenges and issues that the e-learning website aims to address. The objectives of the project are then described, including the general objective that outlines the overall goal of the project and the specific objectives that detail the specific outcomes and targets to be achieved.

In chapter two, we have discussed business area analysis of three prominent online learning platforms, identified the problems that take place in the online learning platforms that they are currently using, studied the forms and the way user interact with those platforms and identified the players of the organization. After that, we have identified the functional requirements of the system to solve the problems of the existing system that is identified during the analysis process. In this chapter essential modeling, the nonfunctional requirements of the system and the collaboration among different classes has been modeled.

After covering the business area analysis of the organization, the next chapter, chapter three, described the analysis phase of the object-oriented system development and design of Enimar-learning website. This chapter covered some concepts of object-oriented design as introduction. What system use case is along with its diagram and description, the identification of actors that can use the system, identification of user interface for the system, the conceptual modeling which includes class diagram and the description of the class diagram and sequence diagram of the systems use case as well as the user interface prototyping of the new system to be developed.

Reference

[1] Grady Booch, Robert A. Maksimchuk, Michael W. Engel, Bobbi J. Young, Jim Conallen, Kelli A. Houston - Object-Oriented Analysis and Design with Application. 3rd Ed. 2007

[2] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK Guide). Project Management Institute, 2017. Accessed on Oct. 23, 2023.

[3] J. Kjeldsen, T. D. Skov, and K. M. Hansen, "Using social media to gather user requirements: An empirical study," in Proceedings of the 27th International Conference on Human-Computer Interaction, 2011, pp. 2907-2912. ACM. Accessed on Oct. 25, 2023.

[4] Simplilearn, "Feasibility Study and Its Importance in Project Management," Simplilearn.com. [Online]. Available: https://www.simplilearn.com/feasibility-study-article. Accessed on Oct. 26, 2023.

[5] Krasamo, "The Agile Development Process for Mobile Apps," Krasamo. [Online]. Available: https://www.krasamo.com/agile-development-process/. Accessed on Oct. 23, 2023.

[6] D. Lock and M. Gopalakrishnan, Project Management: Tools and Techniques (7th ed.). Routledge, 2017. Accessed on Oct. 26, 2023.

[7] Joseph Valacich, Joey George - Modern Systems Analysis and Design 8th Ed. Pearson Edu. -2021.

[8] H. Kerzner, Project Management: A Systematic Approach to Planning, Executing, Controlling, and Closing Projects. John Wiley & Sons, 2013. Accessed on Oct. 24, 2023

[9] Whitten Bentley, Systems Analysis and Design Methods 7th ed.-2007

[10] Joseph Valacich, Joey George, Modern system Analysis Design 9th ed. -2021]

[11] Wikipedia

[12] Udemy.com

[13] Tewodrose Tilahun (2022) Enhancing Third-Year Computer Science Students' Computer Programming Skills at Hawassa University, Ethiopia.