



# Language Learning Website

**Submitted By**

**Saikat Barman**

**211-35-3174**

**Supervised By**

**Nuruzzaman Faruqui**

**Assistant Professor**

This project report has been submitted in fulfilment of the requirements for the degree  
of **Bachelor of Science in Software Engineering**

@ All right Reserved by Daffodil International University

# DAFFODIL INTERNATIONAL UNIVERSITY

## DECLARATION OF THESIS AND COPYRIGHT

I declare that this thesis is classified as:

I acknowledge that Daffodil International University reserves the following rights:

1. The Project is the Property of Daffodil International University.
2. The Library of Daffodil International University has the right to make copies of the Project for the purpose of research only.
3. The Library of Daffodil International University has the right to make copies of the Project for academic exchange.

Certified by:



\_\_\_\_\_  
(Student's Signature)

\_\_\_\_\_  
Student ID: 211-35-3174  
Date: 04 January 2025



\_\_\_\_\_  
(Supervisor's Signature)

\_\_\_\_\_  
Name of Supervisor: Nuruzzaman  
Faruqui (Assistant Professor)  
Date: 04 January 2025

### APPROVAL

This project titled on "Language Learning Website", submitted by Salkat Barman ( 211-35-3174) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

### BOARD OF EXAMINERS



---

**Dr. Imran Mahmud**  
**Associate Professor & Head**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Chairman**



---

**Nuruzzaman Faruqi**  
**Assistant Professor**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

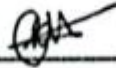
**Internal Examiner 1**



---

**Md. Rajib Mia**  
**Lecturer (Senior Scale)**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Internal Examiner 2**



---

**Md. Fazle Munim**  
**Associate Director & Vice President**  
Government & Public Sector  
Ernst & young (EY)

**External Examiner**

## SUPERVISOR'S DECLARATION

I hereby declare that I have checked this project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Science.



---

(Supervisor's Signature)

Full Name : Nuruzzaman Faruqui  
Position : Assistant Professor  
Date : 04 January 2025

## **STUDENT'S DECLARATION**

I hereby declare that the work in this project is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Daffodil International University or any other institution.



---

**(Student's Signature)**

Full Name : Saikat Barman  
ID Number : 211-35-3174  
Date : 04 January 2025

# **Language Learning Website**

**Saikat Barman**

Project submitted in fulfillment of the requirements  
for the award of the degree of  
Bachelor of Science

Department of Software Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

04 January 2025

# ACKNOWLEDGEMENTS

In this present world of competition, there is a race of existence in which those who are having will come forward to succeed. A project is a bridge between theoretical and practical working. Indeed, I have put in a lot of effort into this project. Nevertheless, it wouldn't have been possible without the kind support and help of many people.

As a beginning of this report, I would like to extend my sincere & heartfelt gratitude to all those who helped me to accomplish this task. I would not have made progress on this project without their active assistance, help, cooperation and encouragement.

It is my honor to first thank Almighty God for guiding me on the right path in life. Without his grace, I would not have been able to complete this project.

It is my pleasure to extend my gratitude to prof. Dr. Imran Mahmud, Head, Department of Software Engineering. Furthermore, I am very grateful to all my distinguished teachers who teach me in such an interesting and easy way, full of enjoyment and who make extra efforts to teach me. I will always be grateful for their kindness and support.

I am extremely grateful to Nuruzzaman Faruqui (Assistant Professor) for his guidance, constant supervision and providing necessary information for the project, as well as the valuable suggestions and assistance in completing this project in its current state.

I am indebted to my department staff, lab technicians and non-teaching staff. They have been extremely helpful throughout the entire project.

Last but not least, I would like to convey my gratitude to my classmate who has directly or indirectly helped me finish this project.

## **DEDICATION**

I therefore declare that I have done this project under the oversight of **Nuruzzaman Faruqi, Assistant Professor**, Department of Software Engineering, Daffodil International University. Also declare that neither entire record nor any portion of this record has been submitted somewhere else for my degree.

# ABSTRACT

My system intends to furnish students with an online platform for language acquisition, specifically centered on coding languages, which provides interactive tutorials, coding challenges, and real-time feedback. Highlighting prevalent programming languages such as Python, This website provides interactive classes, coding challenges, and real-time feedback for both novice and experienced users in JavaScript and Java. The system documents student progress and integrates a customized learning environment with varying task complexity. The integrated code editor, featuring syntax highlighting and debugging functionalities, facilitates hands-on practice. Students can engage, cooperate, and inquire with educators and fellow students at a community forum. Project-based learning courses, which allow students to apply their knowledge in real-world contexts, are also available on the platform. Regular assessments and assignments evaluate knowledge and facilitate comprehension. Leaderboards and badges are gamification tools designed to motivate and engage students. The system offers supplementary educational materials, including video courses, cheat sheets, and manuals. It also provides diverse training pathways tailored to certain career objectives, such as web development or data science. Ultimately, the platform is designed to be accessible on desktop computers, tablets, and smartphones, so facilitating study at any location and time.

## **TABLE OF CONTENT**

ACKNOWLEDGEMENTS	i
DEDICATION	ii
ABSTRACT	iii
TABLE OF CONTENT	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
LIST OF ABBREVIATIONS	ix
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.1.1 Context and Relevance	1
1.1.2 Problem Identification	1
1.1.3 Purpose and Justification	2
1.1.4 Scope	2
1.2 Project Planning and Initiation	2
Feasibility Study (Step-by-Step)	2
1.3 Target User Profile and Tentative Elicitation Process	4
1.3.1 Target User	4
1.3.2 User profile	4
1.3.3 Elicitation Process	6
1.4 Project Block Diagram	7
1.5 System Requirements	7
1.5.1 Hardware Requirements	7
1.5.2 Software Requirements	7
1.5.3 Constraints and Dependencies	8

1.6 Project Scheduling	8
1.7 Summary	9
CHAPTER 2 DESIGN AND IMPLEMENTATION	10
2.1 Introduction	10
2.2 Functional Requirements	10
2.3 Non-Functional Requirements	12
2.3.1 Performance	12
2.3.2 Scalability	13
2.3.3 Availability and Reliability	13
2.3.4 Usability	13
2.3.5 Security	14
2.3.6 Portability	14
2.3.7 Maintainability	14
2.4 Object-oriented System design using UML	15
2.4.1 Use Case Diagram	16
2.4.2 Case Description	17
2.4.3 Activity Diagram	28
2.4.4 Sequence Diagram	39
2.4.5 Class Diagram	44
2.4.6 ER Diagram	45
2.5 Coding: Appendix A	46
2.6 Summary	46
CHAPTER 3 SOFTWARE TESTING	47
3.1 Introduction	47
3.2 Testing Features	47
3.2.1 Feature to Be Tested	47

3.3 Testing Strategies	47
3.3.1 Test Approach	47
3.3.2 Pass/Fail Criteria	48
3.4 System Testing (Test Cases with Report)	48
3.5 Summary	52
CHAPTER 4 DEPLOYMENT AND MAINTENANCE	53
4.1 Introduction	53
4.2 Try to follow the SRLC (software release life cycle)	53
CHAPTER 5 USER MANUAL	54
5.1 Introduction	54
5.2 Project Functionalities	54
5.3 Summary	60
CHAPTER 6 PROJECT SUMMARY	61
6.1 Introduction	61
6.2 Project Limitation	61
6.3 Scope	61
6.4 Future Work	62
6.5 Conclusion	62
REFERENCES	63

## LIST OF TABLES

Table 1.3.1: User Profile for Admin	4
Table 1.3.2: User Profile for Student	5
Table 1.3.3: User Profile for Teacher	5
Table 2.4.1: Case Description-01 Registration	17
Table 2.4.2: Case Description-02 Login	18
Table 2.4.3: Case Description-03 My Courses	19
Table 2.4.4: Case Description-04 Course Feature	20
Table 2.4.5: Case Description-05 Search Course	21
Table 2.4.6: Case Description-06 Buy Course	22
Table 2.4.7: Case Description-07 Payment	23
Table 2.4.8: Case Description-08 Enroll Course	24
Table 2.4.9: Case Description-09 Curriculum	25
Table 2.4.10: Case Description-10 Quiz	26
Table 2.4.11: Case Description-11 Logout	27
Table 3.4.1: Test Case Report for User Registration.	48
Table 3.4.2: Test Case Report for User Login	49
Table 3.4.3: Test Case Report for Admin Login.	49
Table 3.4.4: Test Case Report for Course Feature.	50
Table 3.4.5: Test Case Report for Buy Course.	50
Table 3.4.6: Test Case Report for Enroll Course.	51
Table 3.4.7: Test Case Report for payment.	51
Table 3.4.8: Test Case Report for Quiz.	52

## LIST OF FIGURES

Figure 1.4.1: System Block Diagram	7
Figure 1.6.1: Project Scheduling Gantt Chart.	8
Figure 2.4: Use case Diagram	16
Figure 2.4.1: Activity Diagram for User Registration.	28
Figure 2.4.2: Activity Diagram for Login	29
Figure 2.4.3: Activity Diagram for My Courses	30
Figure 2.4.4: Activity Diagram for My Course Feature.	31
Figure 2.4.5: Activity Diagram for Search Course.	32
Figure 2.4.6: Activity Diagram for Buy Course	33
Figure 2.4.7: Activity Diagram for Payment.	34
Figure 2.4.8: Activity Diagram for Enroll Course.	35
Figure 2.4.9: Activity Diagram for Curriculum.	36
Figure 2.4.10: Activity Diagram for Quiz	37
Figure 2.4.11: Activity Diagram for Logout	38
Figure 2.4.12: Sequence Diagram for Registration	39
Figure 2.4.13: Sequence Diagram for Login	39
Figure 2.4.14: Sequence Diagram for My Course	40
Figure 2.4.15: Sequence Diagram for Course Feature	40
Figure 2.4.16: Sequence Diagram for Search Course	41
Figure 2.4.17: Sequence Diagram for Buy Course	41
Figure 2.4.18: Sequence Diagram for Payment	42
Figure 2.4.19: Sequence Diagram for Enroll Course	42
Figure 2.4.20: Sequence Diagram for Curriculum	43
Figure 2.4.21: Sequence Diagram for Quiz	43
Figure 2.4.22: Class Diagram for Language Learning Website	44
Figure 2.4.23: Entity Relationship Diagram for Language Learning Website	45
Figure 5.2.1: User Log in page	55
Figure 5.2.2: User Registration page	55
Figure 5.2.3: Language learning Home page	56
Figure 5.2.4: Display course and resource page	56
Figure 5.2.5: Display add to cart page	57
Figure 5.2.6: Display course description page	58
Figure 5.2.7: Display course curriculum page	59
Figure 5.2.8: Display course quiz page	59

## LIST OF ABBREVIATIONS

UI	User Interface
UX	User Experience
Frontend	Next.js, JavaScript, CSS
Backend	Python, Django, Django Rest Framework (DRF), JWT
Database	SQLite3
ER	Entity Relationship
SRLC	Software Release Life Cycle

# CHAPTER 1 INTRODUCTION

## 1.1 Background

For prospective programmers of all stripes, my language learning website presents a thorough and interesting experience. Enrolled students may quickly access and manage their active courses, track development, and resume learning where they left off on a customized dashboard found in My Courses. Interactive tutorials with embedded code editors, practical coding tasks with instantaneous feedback, downloadable materials and access to a dynamic community forum for peer support and teacher engagement comprise course features. Users of the strong Search Course capability can rapidly locate courses depending on programming language (Python, Java, JavaScript, etc.), skill level (beginning, intermediate, advanced), or specific topic. Simplified and safe, the Buy Course and Make Payment systems provide several ways to pay for easy purchase. Students can immediately Enroll Course and start their path of learning as a transaction is finished. Every course uses a set Curriculum with well define modules, learning goals, and approximated completion times. Throughout the courses, regular quizzes are included to evaluate understanding, support learning, and offer insightful analysis of student development. From course discovery to mastery of coding, this combined approach guarantees a flawless and efficient learning experience.

### 1.1.1 Context and Relevance

A coding language learning website you may know of that ideally corresponds with the broad packages of on-line transparency, e-studying and the quick-growing EdTech (Educational Technology) market, particularly within the area of software development and computer science education. Current trends that drive rising demand for online coding bootcamps and micro-credentials include rising interactive and gamified learning experiences, personalized learning paths powered by artificial intelligence, and virtual and augmented reality integration for immersive learning. Maintaining good completion rates and effective knowledge transfer in e-learning are two of the biggest challenges. It's also vital to develop engaging content suitable for multiple learning contexts and proficiency levels. You can help close the world digital skills gap through readily accessible and affordable coding training, using data analytics to enhance learning outcomes, and developing strong online communities to help motivate and support student collaboration. The increasing focus of companies on remote working coupled with the digitalization of industries further drives the need for qualified programmers, thus making a serious market for creative coding training solutions.

### 1.1.2 Problem Identification

Through offering easily available and efficient online coding instruction, this coding language learning website seeks to overcome the ongoing skills gap in the technology sector. One of the main issues is the conventional educational model's incapacity to keep up with the fast changing tech scene, so leaving graduates with obsolete skills. Though helpful, prior solutions including in-person coding bootcamps and university computer science programs can be costly, time-consuming, and geographically limited. One alternative that has surfaced online is but many have constraints. Some offer only sparse guidance without sufficient opportunity for practice or

individual feedback. Others have limited community assistance, which can make it especially difficult for students to stay focused and get help. Existing research in publication reinforces the necessity of interactive learning, personalized feedback and community involvement as effective online learning experiences. This project aims to satisfy the need for a more interactive, efficient, and accessible coding education platform by overcoming these limitations and providing interactive tutorials, real-time feedback on coding exercises, project-based learning modules, a supportive online community, and personalized learning paths.

### **1.1.3 Purpose and Justification**

An online platform to learn coding for the growing demand for programmers & software engineers worldwide. It provides a low-cost, easy path into upskilling in popular coding languages. The platform allows flexible learning schedules as opposed to traditional academic education; therefore, individuals learn at their own pace and tailor to suit their existing commitments. Real-time feedback interactive tutorials and project-based learning hone practical skills, better equipping students with real-world coding problems. Spaces with communal aspects not only encourage peer support and collaboration but also make learning much more exciting. This initiative bridges the education and industry gap, encourages tech-related careers and improves digital literacy. The ability to learn code has never been more available to anyone who has internet access, regardless of their background or where they live on the map.

### **1.1.4 Scope**

With this coding language learning website project, we will provide the all-in-one online learning environment for the programming language aspirants of all skill levels. To begin, we will focus on some of the most popular ones with a target list of Python, JavaScript, Java and potentially others based on user demand. The platform's interactive courses with built-in code editors allow users to practice coding in their browsers. By offering diversity in coding assignments and exercises, giving instant feedback, it will facilitate the retention of relevant fundamentals in the student's mind. Students will be using project-based learning modules to create a portfolio of high-quality work that can be used in the real-world as part of this course. One goal of a community forum is to connect with teachers, make new friends, and work together.. The platform's unique learning paths can change based on the preferences and progress of each user. Assessment of learning objectives and provision of useful feedback will culminate in the use of exams, quizzes, and progress tracking.

## **1.2 Project Planning and Initiation**

### **Feasibility Study (Step-by-Step)**

The initial phase of project planning and beginning is a thorough feasibility study. Analyzing the market helps one identify the target audience and competition surroundings. Evaluating the required knowledge and technologies helps to determine technical feasibility. Economic feasibility is the evaluation of income, expenses, potential profitability. Finding the organizational needs and practices will help to establish operational feasibility. Examining the legislation concerning data privacy and online learning helps one to determine their legal feasibility. 6) Project schedule feasibility: determining a reasonable one A SWOT analysis is a synopsis of the internal and outside factors influencing the success of a project. This study

shapes project scope definition, goal-setting, team formation, and resource allocation thereby enabling the project to start off nicely.

### **Phase 1 Preliminary Analysis & Project Scope Definition:**

The preparatory study for this website teaching coding languages closely investigates the needs of the target audience, the competitive environment, and the technical requirements. This study informs the project scope definition, which will include the following significant areas: developing interactive coding tutorials for many programming languages (e.g., Python, JavaScript, and Java); designing coding exercises with automated feedback; implementing project-based learning modules; creating a community forum for peer support and interaction; integrating a user progress tracking system; and designing an intuitive interface working on a range of devices

### **Phase 2 Market Feasibility Analysis (or Market Research):**

The online coding education sector is examined in a coding language learning platform market feasibility analysis. This involves assessing the market's size and growth, identifying major competitors (e.g., Code Coursera, Udemy), and understanding their pros and cons. The method involves determining the target audience (novices, career transitioners, professionals seeking skill upgrade), their learning preferences, and their willingness to take online coding lessons. Market trends, programming languages, pricing strategies, and effective marketing strategies will be collected using surveys, online forum analysis, and competition analysis. This study will evaluate platform demand, identify market niches, and help create a competitive marketing plan.

### **Phase 3 Technical Feasibility Analysis:**

The technical feasibility analysis for a coding language learning website evaluates the technological prerequisites and the practicality of developing and sustaining the platform. This entails assessing the requisite physical infrastructure (servers, databases), software technologies (programming languages for front-end and back-end development, frameworks, databases), and interaction with external services (code editors, payment gateways, video hosting). The assessment encompasses the availability of proficient developers and technical experience, the evaluation of scalability and security factors, and the determination of the feasibility of integrating essential features such as interactive coding environments, automatic feedback mechanisms, and user progress monitoring. The analysis will evaluate performance, reliability, maintainability, and security to verify the platform meets user requirements and manages expected traffic.

### **Phase 4 Financial Feasibility Analysis:**

The website for learning to code's business feasibility analysis checks to see if the project can make money. To do this, you need to figure out how much it will cost to create, run, and market the business (this includes things like infrastructure, staffing, content creation, and advertising), guess how much money it could make (through things like subscriptions, course sales, and partnerships), and then do a cost-benefit analysis. To figure out if the project will make money and last, important numbers like return on investment (ROI), payback time, and break-even point will be used. This study will check to see if the project can make enough money to cover its costs and give a good return on investment.

### 1.3 Target User Profile and Tentative Elicitation Process

#### 1.3.1 Target User

A website that uses coding to teach language has a wide range of target users and stakeholders. The users are mostly aspiring and experienced programmers, divided into four groups: novices with no coding experience, students studying computer science or related subjects, professionals looking to advance their careers in technology, and enthusiasts who want to learn how to code for personal projects. Stakeholders include educational institutions working together to integrate or accredit courses, investors or funding groups providing financial support, instructors and content creators planning and teaching the courses, and the tech industry as a whole using a more skilled workforce. Therefore, for the platform to work, it needs to meet the specific needs and demands of each of these groups.

#### 1.3.2 User profile

Table 1.3.1: User Profile for Admin

User Class	Note on Characteristics
Type of user	Admin
Age range	25-45 years
Frequency of use	Many times of a day
Mandatory	Yes, ensure the website operates smoothly
Computer experience	High (familiar with learning platforms, website management tools)
Education	Higher education in business, e-commerce, or IT
goal	Ensure seamless online for managing student
Language skills	Fluent in Bangla and English

Number of users	1-3
Training	Training on system-specific features
Others system use	Experience with online learning platforms
Way of working	Manages learning platforms, processes payment and updates course.

Table 1.3.2: User Profile for Student

User Class	Note on Characteristics
Type of user	Student
Age range	15-35
Frequency of use	Many times of a day
Mandatory	Yes, to learn the course
Computer experience	High
Education	Varies, but most are educated to at least secondary level
goal	Learn and complete the course
Language skills	Fluent in Bangla and English
Number of users	Many
Others system use	Yes
Way of working	Learn the course properly

Table 1.3.3: User Profile for Teacher

User Class	Note on Characteristics
Type of user	Teacher
Age range	25-45
Frequency of use	Many times of a day
Mandatory	Yes, learning the student online.
Computer experience	High (familiar with learning platforms, code management tools)
Education	BSc and MSc in computer engineering
goal	Learning the student to perform best skills

Language skills	English, Bangla, Coding language
Number of users	1-5
Training	Knowing the coding language
Others system use	Yes
Way of working	Manage course curriculum and quiz

### 1.3.3 Elicitation Process

The eliciting technique of a coding-based language learning website is gathering requests and comments from several people. One can reach this aim in several ways. User surveys help one to find out about people's preferred learning styles, desired characteristics, and areas of difficulty. Interviewing those who would use the platform—beginners, seasoned developers, teachers—is another. By use of a competitive study of present platforms, one can ascertain the performance of rival platforms as well as areas for development. Focus groups can assist to organize conversations and compile group comments. Last but not least, a study of internet communities and forums connected to coding instruction might reveal what people need and are talking about. This multi-pronged approach directs the design of a user-centered platform ensuring a comprehensive awareness of stakeholder requirements.

## 1.4 Project Block Diagram

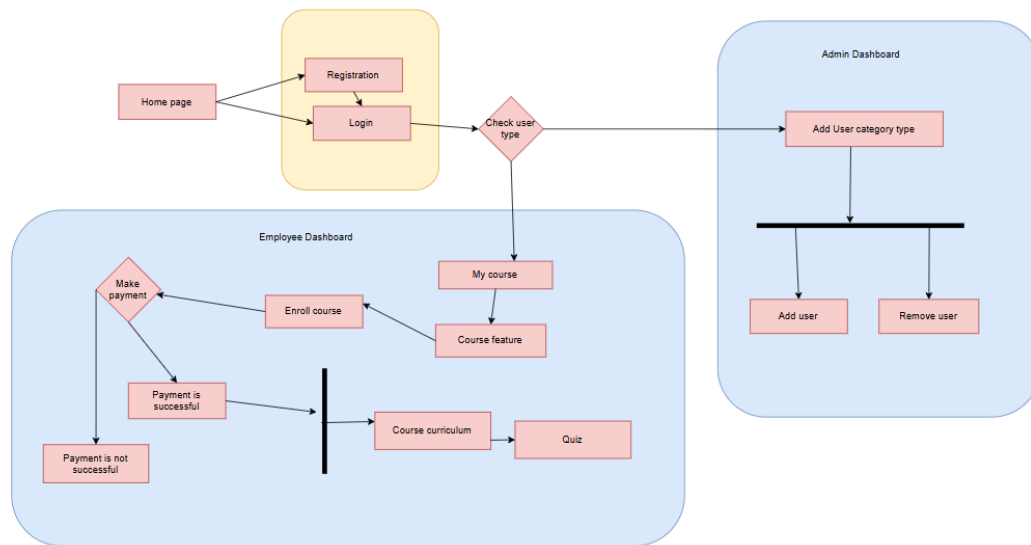


Figure 1.4.1: System Block Diagram

## 1.5 System Requirements

### 1.5.1 Hardware Requirements

The size and expected traffic of a coding-based language learning website determine the hardware requirements. A cloud-based virtual private server (VPS) with at least 2-4 CPU cores, 4-8 GB of RAM, and 50-100 GB of storage would be adequate for early development and small-scale deployments. Dedicated servers or a more reliable cloud architecture with load balancing and auto-scaling features is required as the user base increases. In order to maintain security and isolate user code, this involves web servers to manage user requests, database servers to store user information and course materials, and maybe separate servers for code execution environments (sandboxes). A seamless user experience depends on having enough bandwidth and network connectivity. Additional storage and backup infrastructure are needed for regular backups and disaster recovery procedures.

### 1.5.2 Software Requirements

A coding-based language learning website needs software that supports a variety of technologies. Usually, a stable operating system like Linux (such as Ubuntu or CentOS) is utilized on the server side. HTTP requests are handled by a web server such as Apache or Nginx. User information, course materials, and progress data are stored in a database system like MySQL, PostgreSQL, or MongoDB. The database and application logic are managed by a backend programming language (such as Python with frameworks like Flask or Django, Node.js, or Ruby on Rails). Sandboxing user code execution in interactive coding

environments requires the use of containerization technologies such as Docker or virtual machines. Front-end development is done on the client side using HTML, CSS, and JavaScript. Frameworks like React, Angular, or Vue.js are frequently used to improve the user interface and experience. Other

### 1.5.3 Constraints and Dependencies

Constraints for this coding-oriented language learning website project encompass budgetary limits, potentially constraining the range of functionality or the caliber of resources (e.g., employing seasoned developers, obtaining premium content). Time limitations, dictated by deadlines or market opportunities, can influence the development timeline and require the prioritization of features. Technical limitations, such the restrictions of selected technology or the necessity to integrate with current systems, can impact execution. Dependencies encompass dependency on external services such as cloud hosting providers (AWS, Google Cloud, Azure), payment gateways (Stripe, PayPal), code execution environments (e.g., Judge0), and maybe external APIs for content integration or user authentication. These dependencies create possible failure areas or necessitate meticulous management of integrations.

### 1.6 Project Scheduling

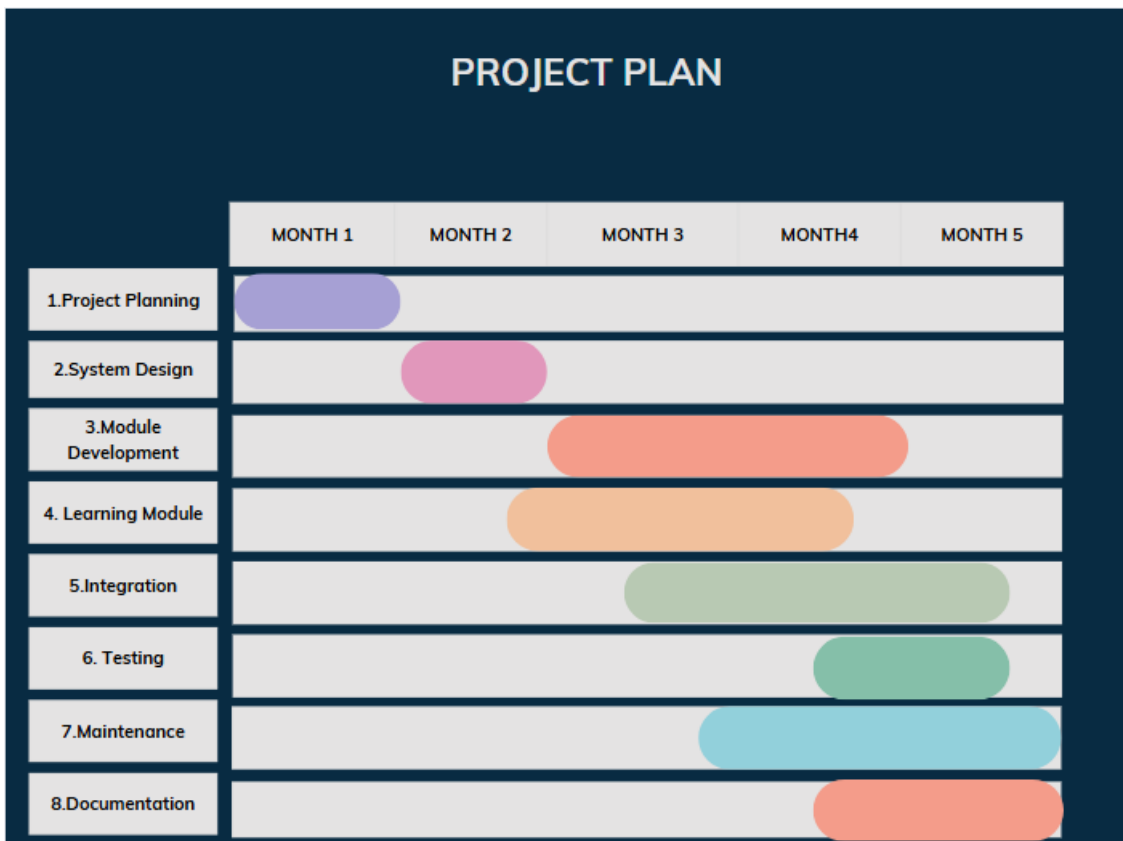


Figure 1.6.1: Project Scheduling Gantt Chart.

Figure 1.6.1 shows the time-bound division of development into phases with simplified time references for achieving planned goals in development. This project is to be accomplished in 5

months where 1st month is for basic planning through gathering user 14 needs for requirement analysis. The design phase will extend into the second month, focusing on establishing system architecture, wireframes, and database schemas. In the third and fourth months, the development phase will take place. The integration and testing phase will take place in the fifth month, ensuring that all modules work flawlessly together with comprehensive testing for functionality, safety and performance. Finally, the documentation and maintenance phase in the fifth month will deploy the system to the production environment and provide user training for shop owners and staff. Regular milestones will be followed through a Gantt chart to monitor progress and guarantee adherence to deadlines.

## **1.7 Summary**

In Chapter 1, the basic parts of the website project for learning to code are explained. Goals and purposes are to make coding education more accessible and useful by using engaging lessons, personalized learning paths, and a helpful community. The goal of this project is to make learning easier, give people real-world writing experience, and give them the tools they need to get in-demand tech skills. During the planning and start part, the project's market, technical, and financial factors are carefully looked into to see if it can work. A user profile table shows that the main target audience is people who want to become programmers, from complete beginners to expert developers who want to learn new skills. To find out what users wanted, surveys, interviews with possible students and teachers, and a comparison of different platforms were used during the elicitation process. The main modules are Interactive Tutorials, Coding Exercises & Feedback, and Community Forum. A block model shows how these modules are connected. The project schedule lists the main steps: planning, system design, programming, testing, deployment, maintenance, and documentation. Each step has a set amount of time (for example, 20 weeks). In conclusion, Chapter One gives a complete history of the project, including its beginnings, goals, target users, and development roadmap.

# CHAPTER 2 DESIGN AND IMPLEMENTATION

## 2.1 Introduction

In the Design and Implementation phase of the coding language learning website project, the collected specifications and desired features become a working website. It comprises essential stages such as designing the system architecture, creating the database schema for users, courses, and progress, UI/UX design for an optimal learning experience, and front-end development using HTML, CSS, and JavaScript to build the user interface. The focus is on interactive development, which includes ongoing testing and feedback to ensure software quality and compliance with project goals.

## 2.2 Functional Requirements

The functional requirements of the coding language learning website include user registration and authentication, course exploration and selection, interactive tutorials with code editors, coding exercises with real-time feedback, progress tracking and reporting, community forum participation (posting, commenting, searching), user profile management, safe payment processing for course purchasing, and administrative functions for course, user,

<b>FR01</b>	<b>Registration</b>
<b>Description</b>	Users can register on the platform to create an account with an email and password, or a social network login. Upon registration, users may establish profiles, monitor their learning progress, and access acquired courses.
<b>Stakeholder</b>	Admin, Teacher, Student

<b>FR02</b>	<b>Login</b>
<b>Description</b>	Users can securely access their customized learning experience by entering their registered email and password on the login page.
<b>Stakeholder</b>	Admin, Teacher, Student

<b>FR03</b>	<b>My Courses</b>
<b>Description</b>	My Courses offers a customized dashboard for enrolled students to access, administer, and monitor their educational progress on the platform.
<b>Stakeholder</b>	Admin, Student

<b>FR04</b>	<b>Course Feature</b>
<b>Description</b>	The course encompasses interactive tutorials, coding exercises with immediate feedback, project-oriented learning, progress monitoring, and community assistance.
<b>Stakeholder</b>	Admin, Teacher, Student

<b>FR05</b>	<b>Search Course</b>
<b>Description</b>	Users can efficiently locate courses by keyword, programming language, skill level, or topic through a comprehensive search function.
<b>Stakeholder</b>	Student

<b>FR06</b>	<b>Buy Course</b>
<b>Description</b>	Users may explore available courses, add them to a cart, and finalize a secure checkout process utilizing several payment options to obtain access.
<b>Stakeholder</b>	Student

<b>FR07</b>	<b>Payment</b>
<b>Description</b>	Secure online payment processing for course acquisitions through many ways (bKash, Nagad).
<b>Stakeholder</b>	Student

<b>FR08</b>	<b>Enroll Course</b>
<b>Description</b>	Users can readily enroll in selected courses upon purchase or via complimentary enrollment alternatives, providing immediate access to course materials and learning activities.
<b>Stakeholder</b>	Student

<b>FR09</b>	<b>Curriculum</b>
<b>Description</b>	The curriculum provides organized learning pathways in prominent programming languages (e.g., Python, JavaScript, Java), advancing from fundamental grammar to sophisticated topics via interactive tutorials, coding exercises, and projects.
<b>Stakeholder</b>	Admin, Teacher, Student

<b>FR10</b>	<b>Quiz</b>
<b>Description</b>	On a website for learning to code, quizzes are short, interactive tests that help students remember what they've learned, make sure they understand important ideas, and get instant feedback.
<b>Stakeholder</b>	Admin, Teacher, Student

<b>FR11</b>	Logout
<b>Description</b>	Secure logout lets users end their current session, which stops other people from getting into their account without their permission.
<b>Stakeholder</b>	Admin, Teacher, Student

## 2.3 Non-Functional Requirements

The coding-based language learning website features non-functional standards that are primarily concerned with how well the system performs and looks. Performance criteria include rapid page loading and code execution with little to no delay. Security needs include secure user authentication as well as vulnerability protection. Usability standards include easy navigation, clear directions, and responsive design across all devices. Scalability requirements include the ability to support more users and data. Reliability requirements include minimal downtime, data backups and recovery, and compliance with WCAG criteria for impaired users. These standards ensure that users have a positive experience, that data is kept secure, and that the site will remain operational for a long time.

### 2.3.1 Performance

The performance of the coding language learning website is first priority for a decent user experience. Key performance criteria are website availability (with little downtime), code execution speed inside interactive settings (offering quick feedback), database query performance (retrieving and storing data rapidly), and

server response time (processing user requests efficiently). The platform should be able to manage several users at once without compromising speed. Performance optimization and load testing will be part of the development process to ensure the website meets these performance criteria and provides a responsive and flawless learning environment.

### **2.3.2 Scalability**

Scalable – a site that teaches people how to code. Its architecture must handle an increasing volume of users, courses, and data with uncompromising speed. Achieving this involves a multidisciplinary approach, including the use of cloud infrastructure (such as AWS or Google Cloud) that can create more infrastructure on-demand (CPU, RAM, storage), load balancing that distributes traffic to multiple servers, carving your database properly to allow quick retrieval of data, caching that takes some load off the database and async processing for things that do not require the user to wait. To ensure that the platform can expand successfully as the user base grows and to detect bottlenecks, regular performance testing and monitoring are required.

### **2.3.3 Availability and Reliability**

People who want to learn how to code need to be able to easily reach and trust a website that does this. With high availability, users never have to worry that they won't be able to access the platform because of maintenance, server problems, or failures that were not planned for. This is possible with backup systems, strong design, and load balancing. The quality of how well the platform works is called its reliability. It includes how well it can consistently provide correct information, safe places to run code, and keep data clean.

### **2.3.4 Usability**

Usability is crucial for the success of a coding-based language learning website. Key usability considerations include intuitive navigation, allowing users to easily find courses and resources; clear and concise content presentation, ensuring learning materials are easy to understand; consistent design and layout, creating a familiar and predictable user experience; responsive design, making the platform accessible across various devices (desktops, tablets, smartphones); effective feedback mechanisms,

providing clear and timely feedback on coding exercises; and accessibility for users with disabilities, adhering to accessibility guidelines (e.g., WCAG). Usability testing with target users should be conducted throughout the development process to identify and address any usability issues.

### **2.3.5 Security**

The website for learning coding is really secure. Data privacy regulations (GDPR, CCPA) and secure code execution environments (sandboxing) are among the measures taken to prevent distributed denial-of-service (DDoS) attacks, secure storage of sensitive data (encryption of passwords and personal information), secure user authentication (strong passwords, password hashing, two-factor authentication), input validation to prevent injection attacks (SQL injection, cross-site scripting), and regular security audits and penetration tests. To keep the platform secure, it is essential to regularly update and patch all software components.

### **2.3.6 Portability**

The portability of this website for learning languages through code means that it can be used on many different types of devices and computers. This means the website should be responsive, which means it should be able to change how it looks and works to fit desk tops, computers, tablets, and phones of different sizes. It is very important that the website works well on Chrome, Firefox, Safari, and Edge. This is called cross-browser support. The platform should also work with many operating systems, like Windows, macOS, Linux, Android, and iOS. People can use standard online technologies (like HTML, CSS, and JavaScript) and make sure that everything works the same on all devices and platforms by following web accessibility rules (WCAG).

### **2.3.7 Maintainability**

It realizes long-term stability and adaptability which are two of the main factors for a coding education website. The above can be tackled using a number of techniques are as below – modular design that separates out functionality into an independent component making it easy to update and troubleshoot; thorough documentation that describes the code logic and functionality; adherence to coding standards across the development team making the code readable and maintainable; automated testing including unit tests and integration tests to catch regressions; the use of tried and tested frameworks and libraries where community support matter and time is saved during development; and regular code reviews to share knowledge, catch mistakes early. These methods allow for maintenance work such as updates, bug fixes,

and added functionality to be carried out over time, as opposed to incurring downtime and limiting the operational longevity of the website.

## **2.4 Object-oriented System design using UML**

### **2.4.1 Use Case Diagram**

2.4.2 The Use Case Diagram for this project illustrates the interaction between various actors and the system's core functionalities. It provides a visual representation of the system's scope and the relationships between the system's modules and its stakeholders.

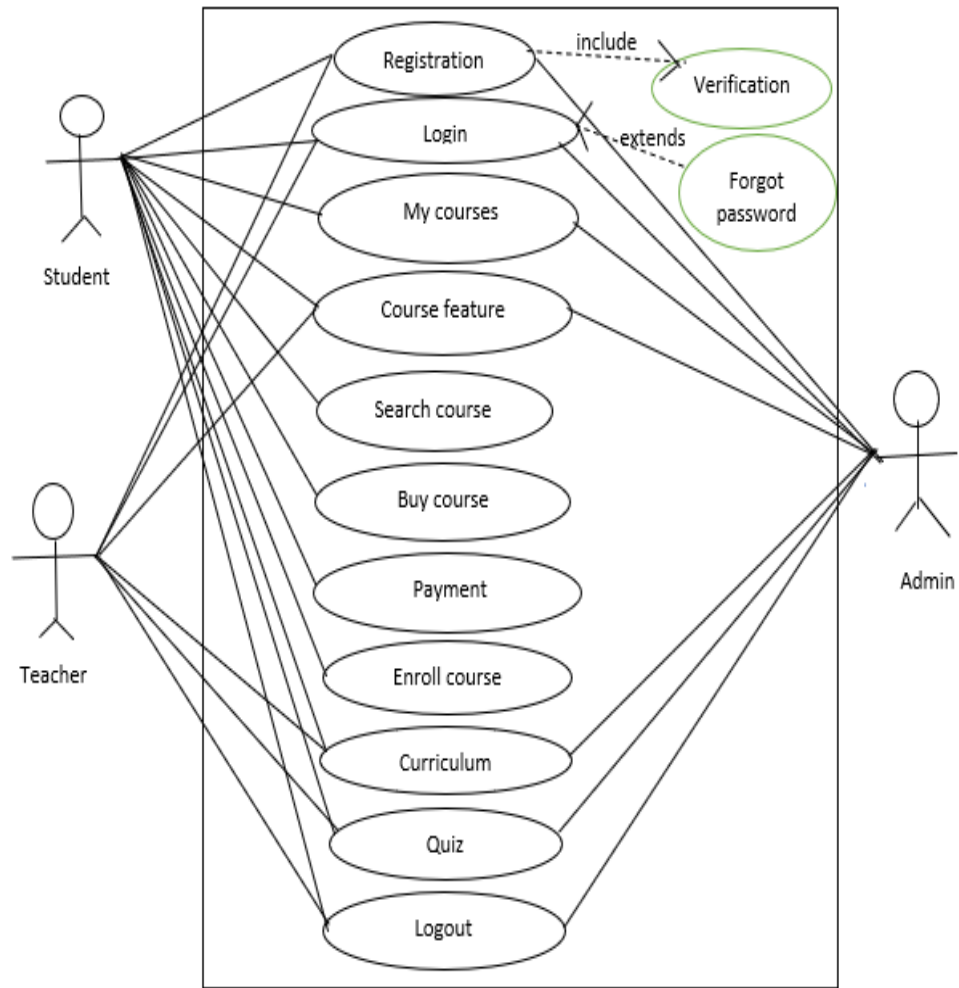


Figure 2.4: Use case Diagram

## 2.4.2 Case Description

**Table 2.4.1: Case Description-01 Registration**

<b>Use Case</b>	<b>Registration</b>													
<b>Goal</b>	<b>Successfully Registration this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully join with this.</b>													
<b>Failed End Condition</b>	<b>Registration process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>														
<b>Trigger</b>	<b>Request for Submit Registration from .</b>													
<b>Description/Main Success Scenario</b>	<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 15%;">Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"><b>1</b></td> <td><b>Request for member registration.</b></td> </tr> <tr> <td style="text-align: center;"><b>2</b></td> <td><b>Admin will accept the from.</b></td> </tr> <tr> <td style="text-align: center;"><b>3</b></td> <td><b>Admin check All the information and if all information is okay then click the submit button</b></td> </tr> </tbody> </table>		Step	Action	<b>1</b>	<b>Request for member registration.</b>	<b>2</b>	<b>Admin will accept the from.</b>	<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>				
Step	Action													
<b>1</b>	<b>Request for member registration.</b>													
<b>2</b>	<b>Admin will accept the from.</b>													
<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>													
<b>Alternative Flows</b>	<table border="1" style="width: 100%;"> <tbody> <tr> <td style="text-align: center;"><b>1</b></td> <td><b>Admin will not connect with the system.</b></td> </tr> <tr> <td style="text-align: center;"><b>1.1</b></td> <td><b>Server not found</b></td> </tr> <tr> <td style="text-align: center;"><b>2</b></td> <td><b>Admin can't receive the from.</b></td> </tr> <tr> <td style="text-align: center;"><b>2.1</b></td> <td><b>Download fail .</b></td> </tr> <tr> <td style="text-align: center;"><b>3</b></td> <td><b>Format dose not match .</b></td> </tr> <tr> <td style="text-align: center;"><b>3.1</b></td> <td><b>Fill up the registration from properly</b></td> </tr> </tbody> </table>		<b>1</b>	<b>Admin will not connect with the system.</b>	<b>1.1</b>	<b>Server not found</b>	<b>2</b>	<b>Admin can't receive the from.</b>	<b>2.1</b>	<b>Download fail .</b>	<b>3</b>	<b>Format dose not match .</b>	<b>3.1</b>	<b>Fill up the registration from properly</b>
<b>1</b>	<b>Admin will not connect with the system.</b>													
<b>1.1</b>	<b>Server not found</b>													
<b>2</b>	<b>Admin can't receive the from.</b>													
<b>2.1</b>	<b>Download fail .</b>													
<b>3</b>	<b>Format dose not match .</b>													
<b>3.1</b>	<b>Fill up the registration from properly</b>													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.2: Case Description-02 Login

<b>Use Case</b>	<b>login</b>													
<b>Goal</b>	<b>Successfully login this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully join with this.</b>													
<b>Failed End Condition</b>	<b>Login process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>														
<b>Trigger</b>	<b>Request for Submit Login from .</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th><b>Step</b></th> <th><b>Action</b></th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td><b>Request for member login.</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin will accept the from.</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Admin check All the information and if all information is okay then click the submit button</b></td> </tr> </tbody> </table>		<b>Step</b>	<b>Action</b>	<b>1</b>	<b>Request for member login.</b>	<b>2</b>	<b>Admin will accept the from.</b>	<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>				
<b>Step</b>	<b>Action</b>													
<b>1</b>	<b>Request for member login.</b>													
<b>2</b>	<b>Admin will accept the from.</b>													
<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td><b>1</b></td> <td><b>Admin will not connect with the system.</b></td> </tr> <tr> <td><b>1.1</b></td> <td><b>Server not found</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin can't receive the from.</b></td> </tr> <tr> <td><b>2.1</b></td> <td><b>Download fail .</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Format dose not match .</b></td> </tr> <tr> <td><b>3.1</b></td> <td><b>Fill up the registration from properly</b></td> </tr> </tbody> </table>		<b>1</b>	<b>Admin will not connect with the system.</b>	<b>1.1</b>	<b>Server not found</b>	<b>2</b>	<b>Admin can't receive the from.</b>	<b>2.1</b>	<b>Download fail .</b>	<b>3</b>	<b>Format dose not match .</b>	<b>3.1</b>	<b>Fill up the registration from properly</b>
<b>1</b>	<b>Admin will not connect with the system.</b>													
<b>1.1</b>	<b>Server not found</b>													
<b>2</b>	<b>Admin can't receive the from.</b>													
<b>2.1</b>	<b>Download fail .</b>													
<b>3</b>	<b>Format dose not match .</b>													
<b>3.1</b>	<b>Fill up the registration from properly</b>													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.3: Case Description-03 My Courses

<b>Use Case</b>	<b>My courses</b>													
<b>Goal</b>	<b>Successfully see information of this site.</b>													
<b>Preconditions</b>	<b>To see the information press my courses button.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully see the details.</b>													
<b>Failed End Condition</b>	<b>Information does not found.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for see the information.</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th><b>Step</b></th> <th><b>Action</b></th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td><b>Request for see the information.</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin will accept the request.</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Admin check All the information and if all information is okay then click the submit button</b></td> </tr> </tbody> </table>		<b>Step</b>	<b>Action</b>	<b>1</b>	<b>Request for see the information.</b>	<b>2</b>	<b>Admin will accept the request.</b>	<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>				
<b>Step</b>	<b>Action</b>													
<b>1</b>	<b>Request for see the information.</b>													
<b>2</b>	<b>Admin will accept the request.</b>													
<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td><b>1</b></td> <td><b>Admin will not connect with the system.</b></td> </tr> <tr> <td><b>1.1</b></td> <td><b>Server not found</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin can't receive the from.</b></td> </tr> <tr> <td><b>2.1</b></td> <td><b>Download fail .</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Format dose not match .</b></td> </tr> <tr> <td><b>3.1</b></td> <td><b>Fill up the registration from properly</b></td> </tr> </tbody> </table>		<b>1</b>	<b>Admin will not connect with the system.</b>	<b>1.1</b>	<b>Server not found</b>	<b>2</b>	<b>Admin can't receive the from.</b>	<b>2.1</b>	<b>Download fail .</b>	<b>3</b>	<b>Format dose not match .</b>	<b>3.1</b>	<b>Fill up the registration from properly</b>
<b>1</b>	<b>Admin will not connect with the system.</b>													
<b>1.1</b>	<b>Server not found</b>													
<b>2</b>	<b>Admin can't receive the from.</b>													
<b>2.1</b>	<b>Download fail .</b>													
<b>3</b>	<b>Format dose not match .</b>													
<b>3.1</b>	<b>Fill up the registration from properly</b>													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.4: Case Description-04 Course Feature

<b>Use Case</b>	<b>Course feature</b>												
<b>Goal</b>	<b>Successfully see information of this site.</b>												
<b>Preconditions</b>	<b>To see the information press course feature button.</b>												
<b>Success end Condition</b>	<b>After precondition everybody will be successfully see the details.</b>												
<b>Failed End Condition</b>	<b>Information does not found.</b>												
<b>Primary Actors</b>	<b>User</b>												
<b>Secondary Actors</b>	<b>Admin.</b>												
<b>Trigger</b>	<b>Request for see the information.</b>												
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for see the information.</td> </tr> <tr> <td>2</td> <td>Admin will accept the request.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>	Step	Action	1	Request for see the information.	2	Admin will accept the request.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action												
1	Request for see the information.												
2	Admin will accept the request.												
3	Admin check All the information and if all information is okay then click the submit button												
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>	1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.												
1.1	Server not found												
2	Admin can't receive the from.												
2.1	Download fail .												
3	Format dose not match .												
3.1	Fill up the registration from properly												
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>												

Table 2.4.5: Case Description-05 Search Course

<b>Use Case</b>	<b>Search course</b>													
<b>Goal</b>	<b>Successfully see information of this site.</b>													
<b>Preconditions</b>	<b>To see the information press search course button.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully see the details.</b>													
<b>Failed End Condition</b>	<b>Information does not found.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for see the information.</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th><b>Step</b></th> <th><b>Action</b></th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td><b>Request for see the information.</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin will accept the request.</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Admin check All the information and if all information is okay then click the submit button</b></td> </tr> </tbody> </table>		<b>Step</b>	<b>Action</b>	<b>1</b>	<b>Request for see the information.</b>	<b>2</b>	<b>Admin will accept the request.</b>	<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>				
<b>Step</b>	<b>Action</b>													
<b>1</b>	<b>Request for see the information.</b>													
<b>2</b>	<b>Admin will accept the request.</b>													
<b>3</b>	<b>Admin check All the information and if all information is okay then click the submit button</b>													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td><b>1</b></td> <td><b>Admin will not connect with the system.</b></td> </tr> <tr> <td><b>1.1</b></td> <td><b>Server not found</b></td> </tr> <tr> <td><b>2</b></td> <td><b>Admin can't receive the from.</b></td> </tr> <tr> <td><b>2.1</b></td> <td><b>Download fail .</b></td> </tr> <tr> <td><b>3</b></td> <td><b>Format dose not match .</b></td> </tr> <tr> <td><b>3.1</b></td> <td><b>Fill up the registration from properly</b></td> </tr> </tbody> </table>		<b>1</b>	<b>Admin will not connect with the system.</b>	<b>1.1</b>	<b>Server not found</b>	<b>2</b>	<b>Admin can't receive the from.</b>	<b>2.1</b>	<b>Download fail .</b>	<b>3</b>	<b>Format dose not match .</b>	<b>3.1</b>	<b>Fill up the registration from properly</b>
<b>1</b>	<b>Admin will not connect with the system.</b>													
<b>1.1</b>	<b>Server not found</b>													
<b>2</b>	<b>Admin can't receive the from.</b>													
<b>2.1</b>	<b>Download fail .</b>													
<b>3</b>	<b>Format dose not match .</b>													
<b>3.1</b>	<b>Fill up the registration from properly</b>													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.6: Case Description-06 Buy Course

<b>Use Case</b>	<b>Buy course</b>													
<b>Goal</b>	<b>Successfully buy course of this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition course will be successfully buy.</b>													
<b>Failed End Condition</b>	<b>Buy course process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for Submit buy course from .</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for buy course.</td> </tr> <tr> <td>2</td> <td>Admin will accept the from.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for buy course.	2	Admin will accept the from.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for buy course.													
2	Admin will accept the from.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.7: Case Description-07 Payment

<b>Use Case</b>	<b>Payment</b>													
<b>Goal</b>	<b>Successfully make payment of this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition payment will be successful.</b>													
<b>Failed End Condition</b>	<b>Make payment process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for Submit make payment from .</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for make payment.</td> </tr> <tr> <td>2</td> <td>Admin will accept the from.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for make payment.	2	Admin will accept the from.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for make payment.													
2	Admin will accept the from.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.8: Case Description-08 Enroll Course

<b>Use Case</b>	<b>Enroll course</b>													
<b>Goal</b>	<b>Successfully enroll course of this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition enroll course will be successful.</b>													
<b>Failed End Condition</b>	<b>Enroll course process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for enroll course.</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for enrol course.</td> </tr> <tr> <td>2</td> <td>Admin will accept the from.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for enrol course.	2	Admin will accept the from.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for enrol course.													
2	Admin will accept the from.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.9: Case Description-09 Curriculum

<b>Use Case</b>	curriculum													
<b>Goal</b>	Successfully see information of this site.													
<b>Preconditions</b>	To see the information press curriculum button.													
<b>Success end Condition</b>	After precondition everybody will be successfully see the details.													
<b>Failed End Condition</b>	Information does not found.													
<b>Primary Actors</b>	User													
<b>Secondary Actors</b>	Admin.													
<b>Trigger</b>	Request for see the information.													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for see the information.</td> </tr> <tr> <td>2</td> <td>Admin will accept the request.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for see the information.	2	Admin will accept the request.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for see the information.													
2	Admin will accept the request.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	You will confirm it within 30 seconds													

Table 2.4.10: Case Description-10 Quiz

<b>Use Case</b>	<b>Quiz</b>													
<b>Goal</b>	<b>Successfully see quiz of this site.</b>													
<b>Preconditions</b>	<b>To see the information press quiz button.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully see the details.</b>													
<b>Failed End Condition</b>	<b>Information does not found.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>														
<b>Trigger</b>	<b>Request for see the quiz.</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for see the quiz information.</td> </tr> <tr> <td>2</td> <td>Admin will accept the request.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for see the quiz information.	2	Admin will accept the request.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for see the quiz information.													
2	Admin will accept the request.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

Table 2.4.11: Case Description-11 Logout

<b>Use Case</b>	<b>logout</b>													
<b>Goal</b>	<b>Successfully logout this site.</b>													
<b>Preconditions</b>	<b>All information must be submit.</b>													
<b>Success end Condition</b>	<b>After precondition everybody will be successfully join with this.</b>													
<b>Failed End Condition</b>	<b>Logout process unsuccessful.</b>													
<b>Primary Actors</b>	<b>User</b>													
<b>Secondary Actors</b>	<b>Admin.</b>													
<b>Trigger</b>	<b>Request for Submit Logout from .</b>													
<b>Description/Main Success Scenario</b>	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Request for member logoui.</td> </tr> <tr> <td>2</td> <td>Admin will accept the from.</td> </tr> <tr> <td>3</td> <td>Admin check All the information and if all information is okay then click the submit button</td> </tr> </tbody> </table>		Step	Action	1	Request for member logoui.	2	Admin will accept the from.	3	Admin check All the information and if all information is okay then click the submit button				
Step	Action													
1	Request for member logoui.													
2	Admin will accept the from.													
3	Admin check All the information and if all information is okay then click the submit button													
<b>Alternative Flows</b>	<table border="1"> <tbody> <tr> <td>1</td> <td>Admin will not connect with the system.</td> </tr> <tr> <td>1.1</td> <td>Server not found</td> </tr> <tr> <td>2</td> <td>Admin can't receive the from.</td> </tr> <tr> <td>2.1</td> <td>Download fail .</td> </tr> <tr> <td>3</td> <td>Format dose not match .</td> </tr> <tr> <td>3.1</td> <td>Fill up the registration from properly</td> </tr> </tbody> </table>		1	Admin will not connect with the system.	1.1	Server not found	2	Admin can't receive the from.	2.1	Download fail .	3	Format dose not match .	3.1	Fill up the registration from properly
1	Admin will not connect with the system.													
1.1	Server not found													
2	Admin can't receive the from.													
2.1	Download fail .													
3	Format dose not match .													
3.1	Fill up the registration from properly													
<b>Quality Requirements</b>	<b>You will confirm it within 30 seconds</b>													

### 2.4.3 Activity Diagram

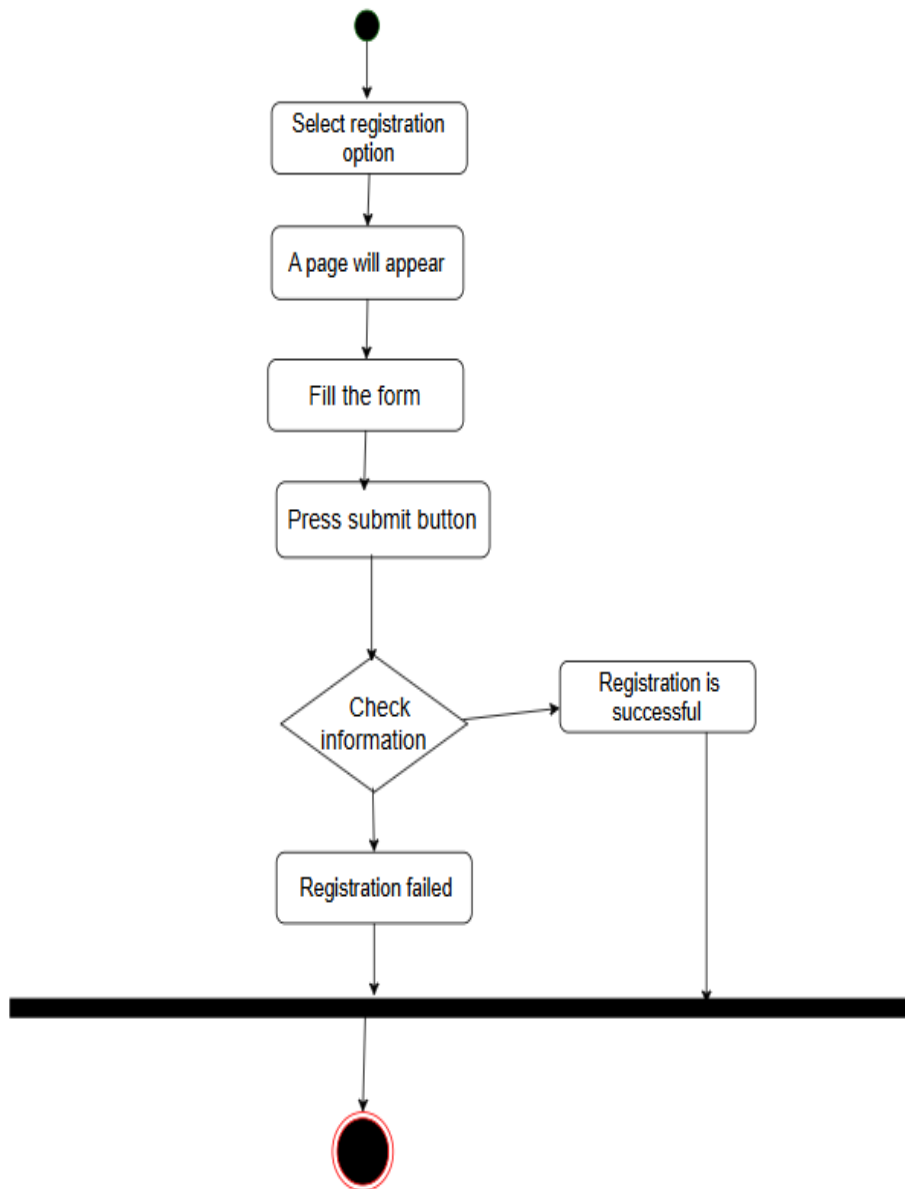


Figure 2.4.1: Activity Diagram for User Registration.

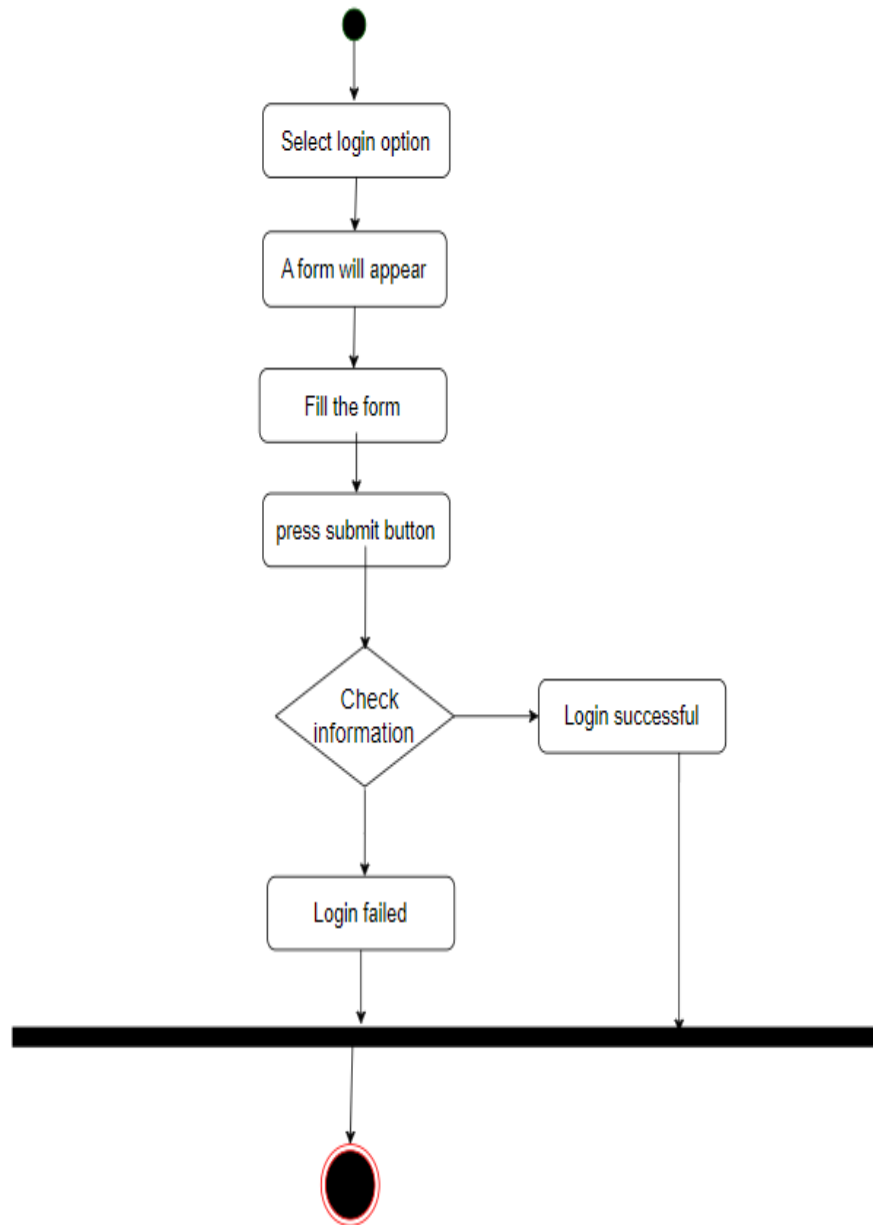


Figure 2.4.2: Activity Diagram for Login.

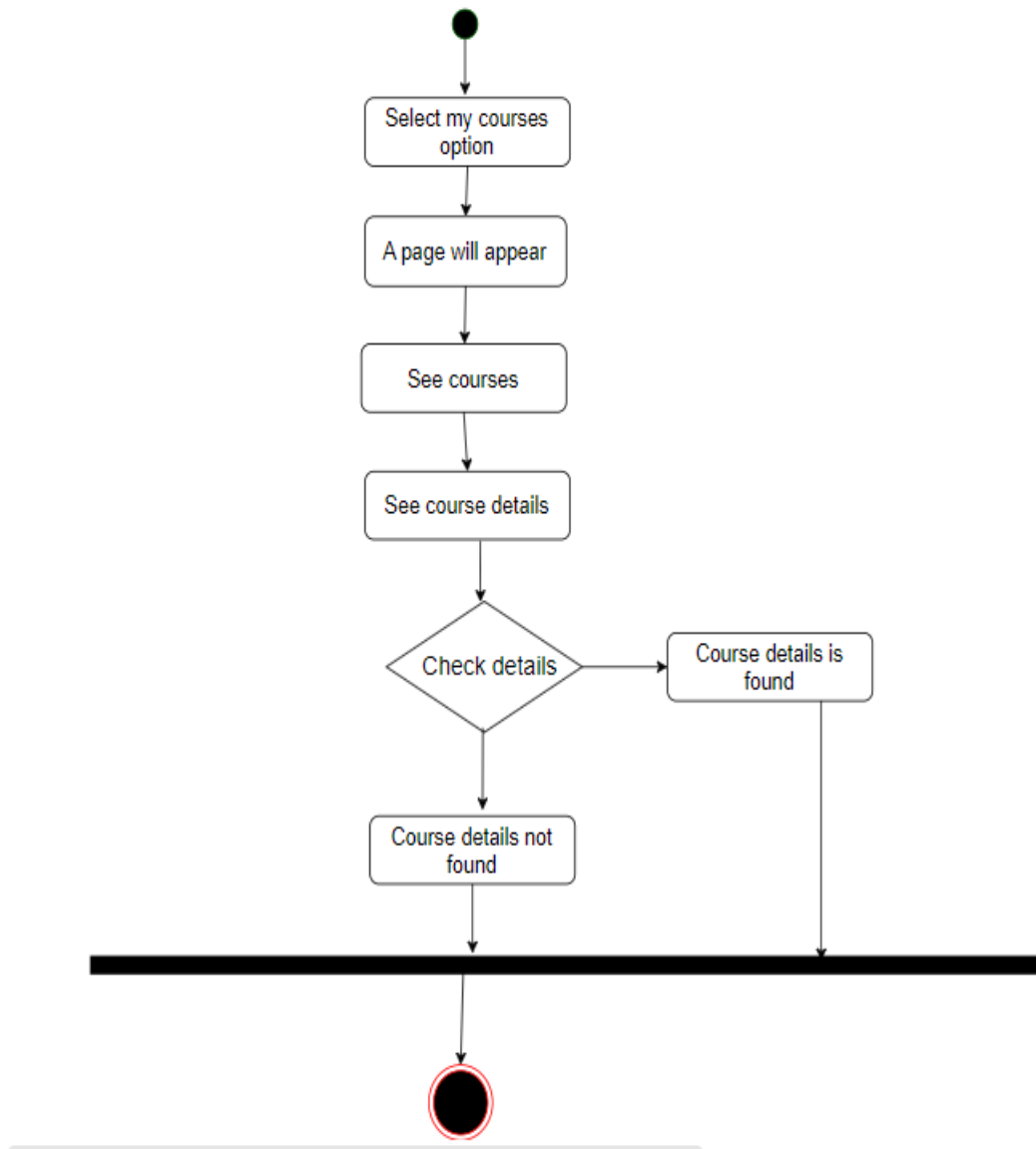


Figure 2.4.3: Activity Diagram for My Courses.

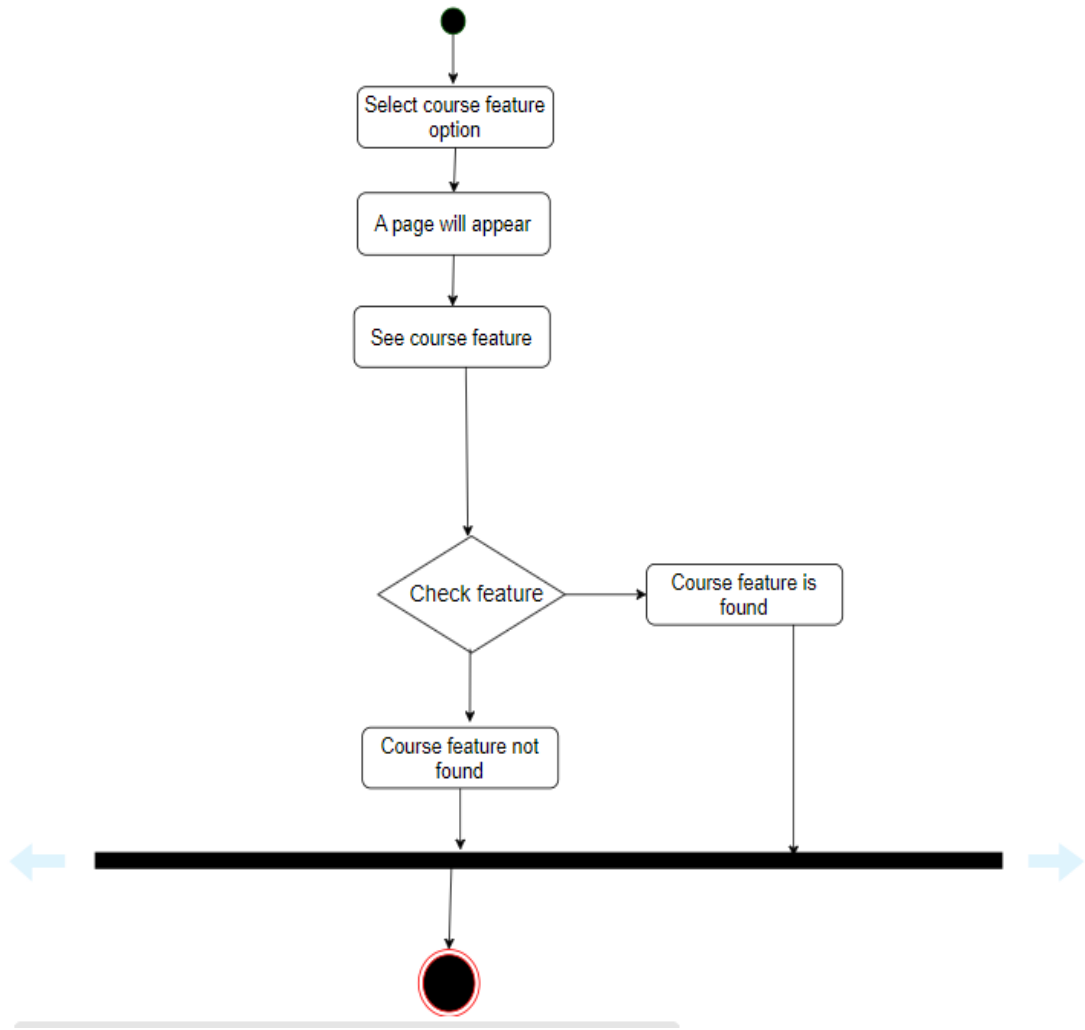


Figure 2.4.4: Activity Diagram for My Course Feature.

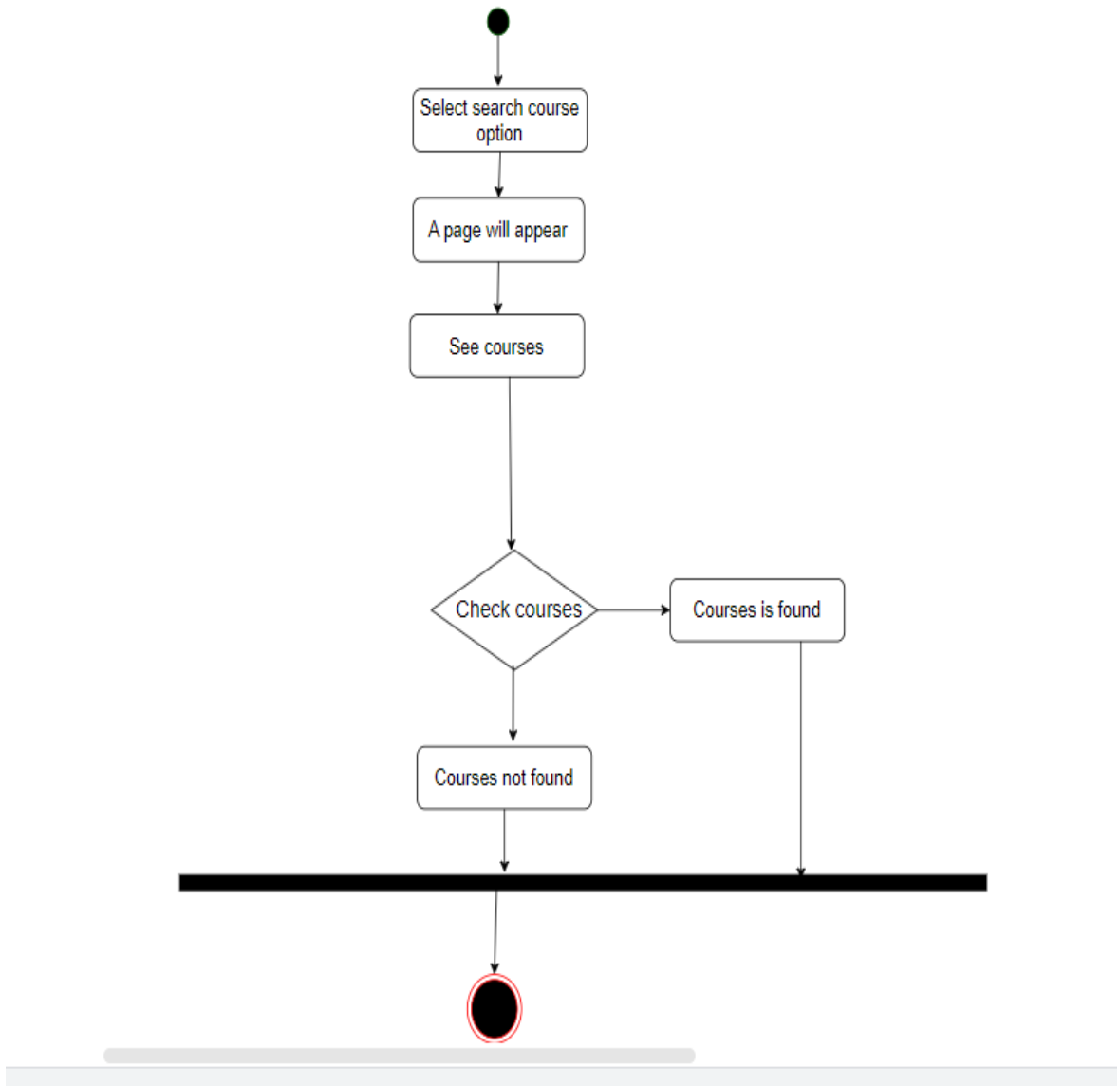


Figure 2.4.5: Activity Diagram for Search Course.

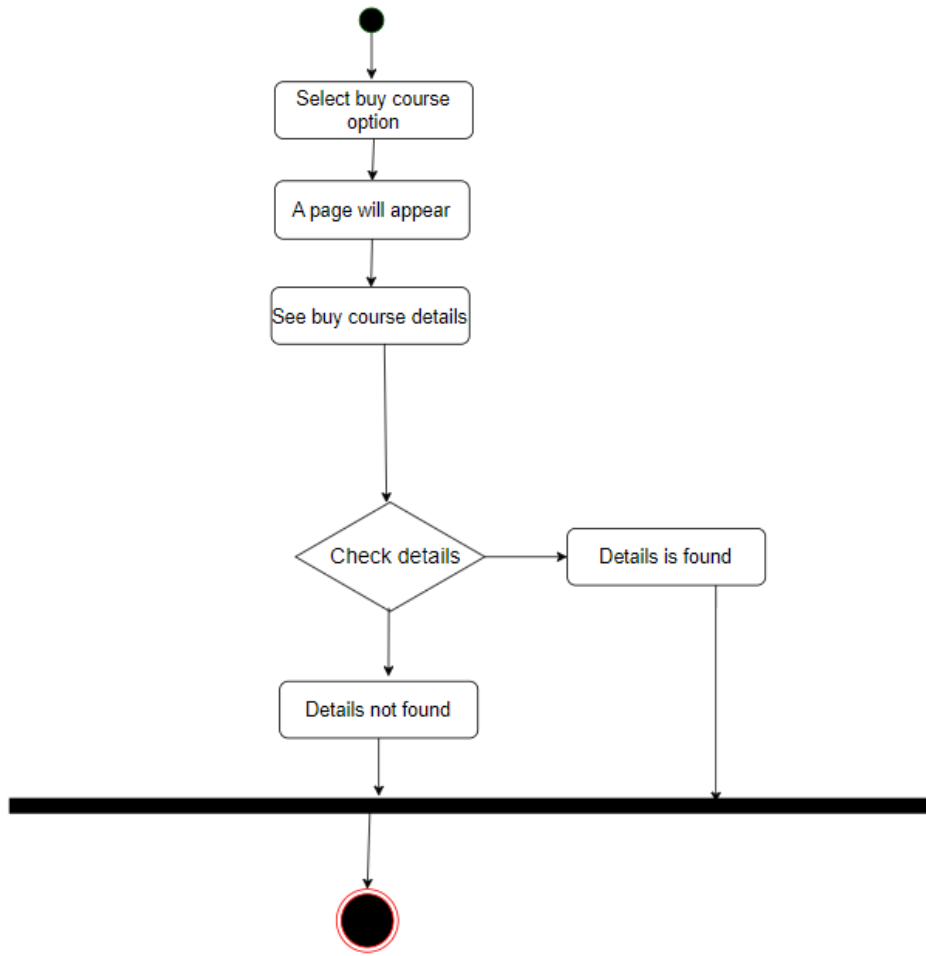


Figure 2.4.6: Activity Diagram for Buy Course

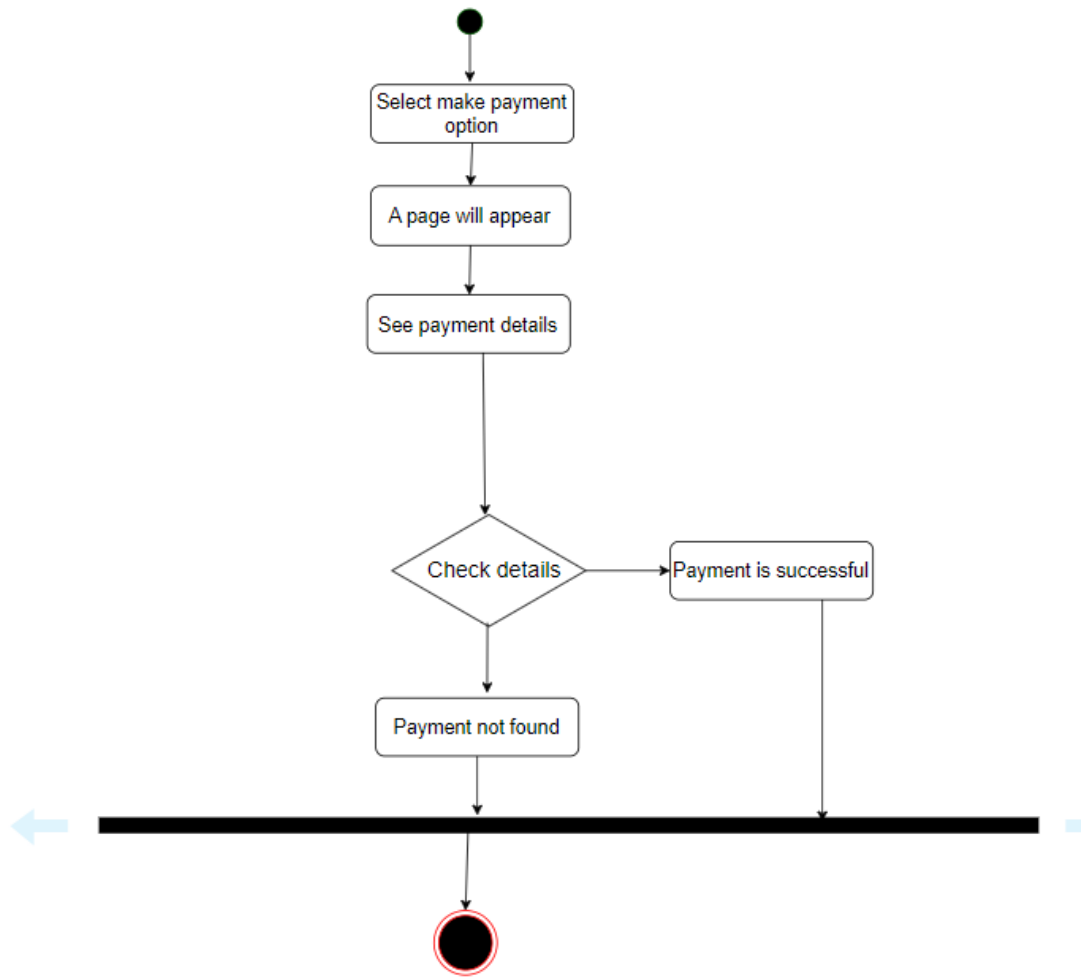


Figure 2.4.7: Activity Diagram for Payment.

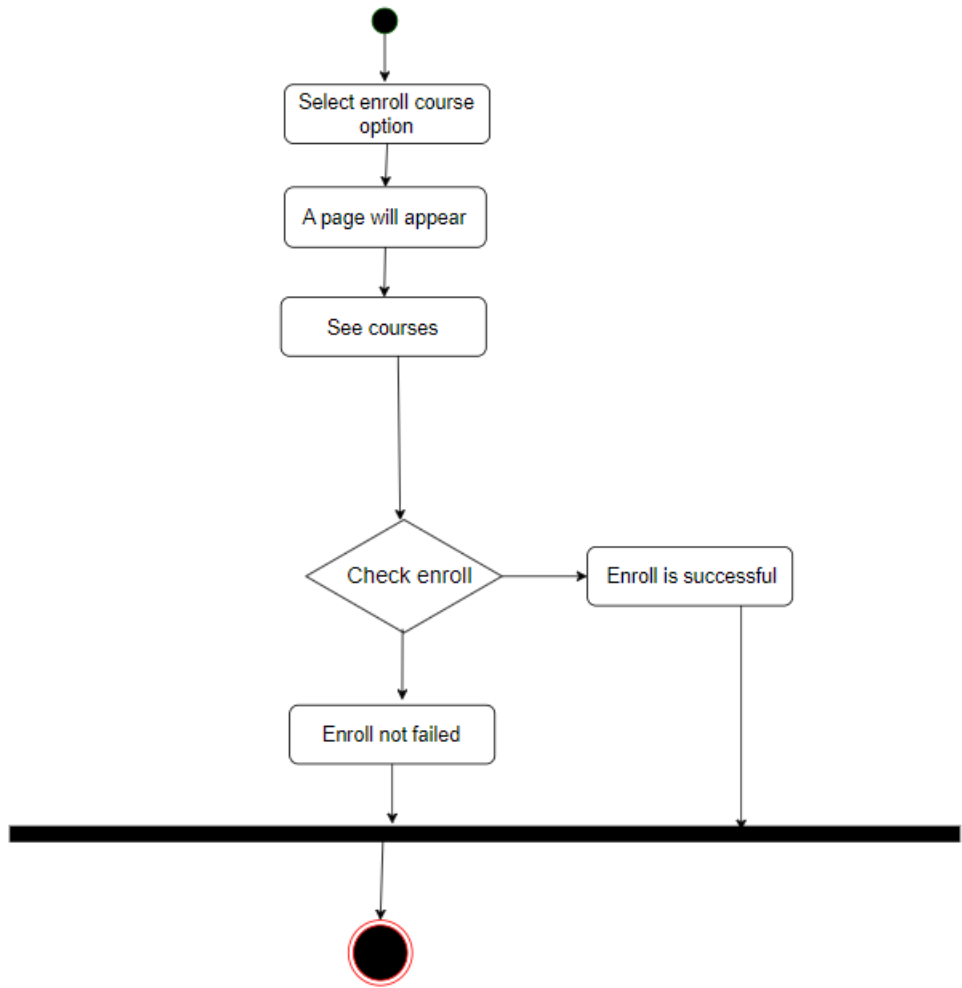


Figure 2.4.8: Activity Diagram for Enroll Course.

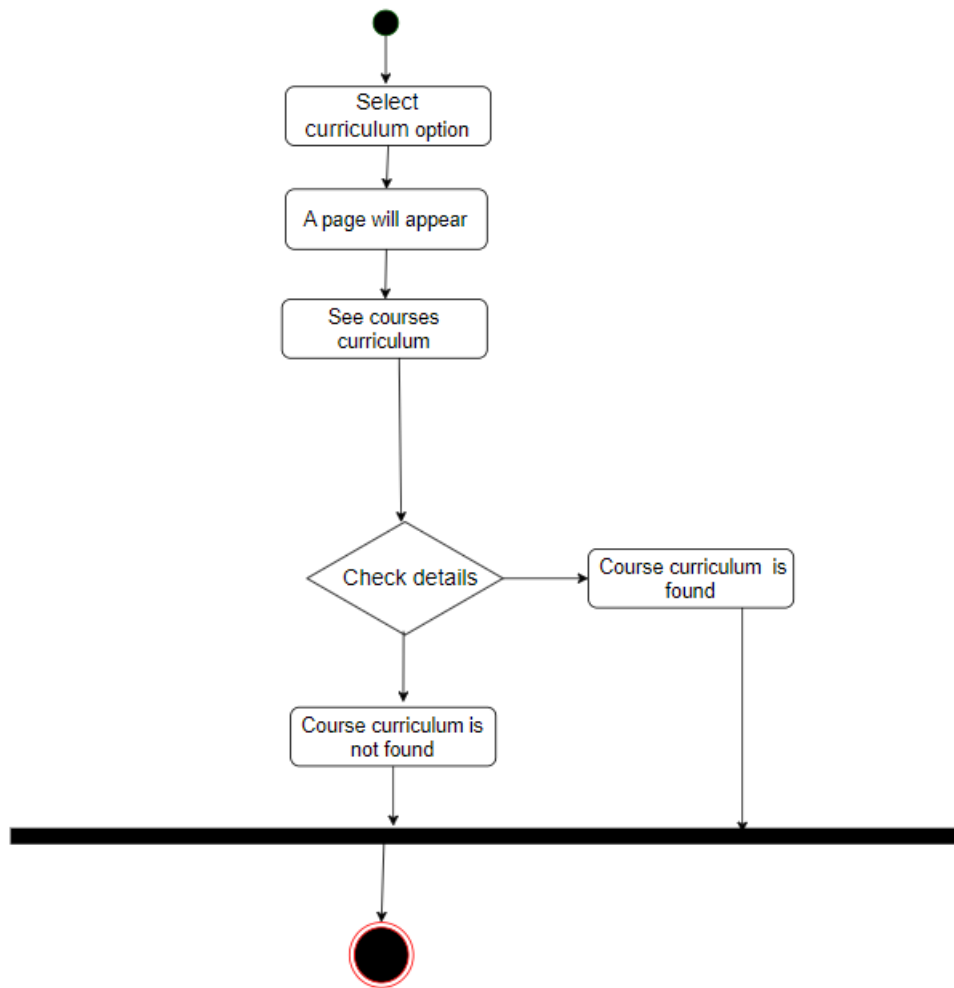


Figure 2.4.9: Activity Diagram for Curriculum.

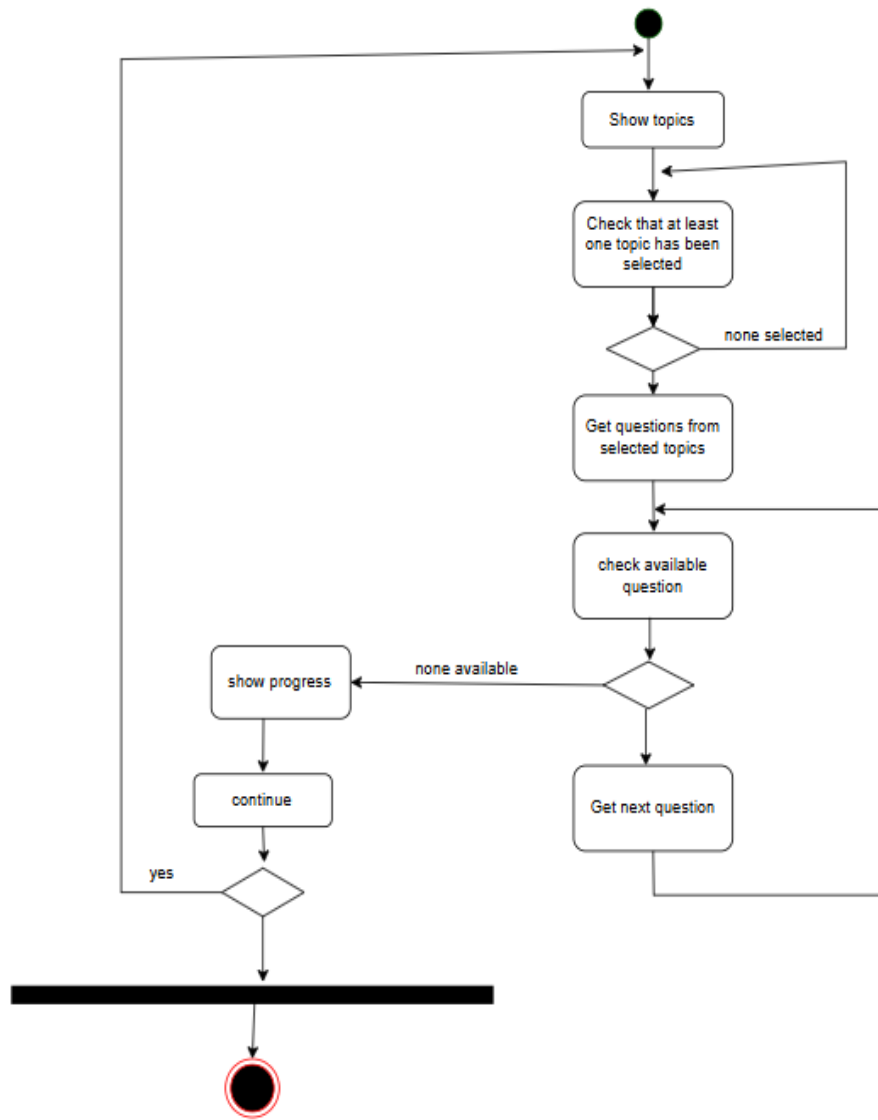


Figure 2.4.10: Activity Diagram for Quiz

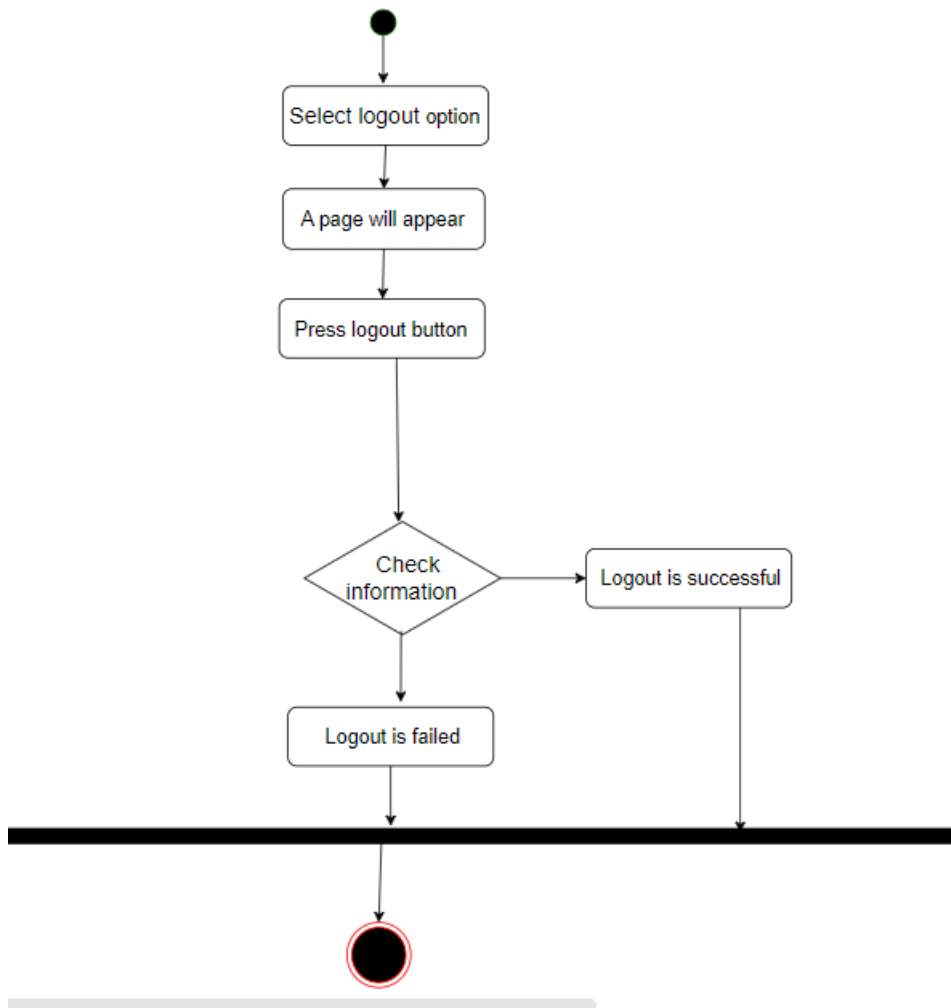


Figure 2.4.11: Activity Diagram for Logout

## 2.4.4 Sequence Diagram

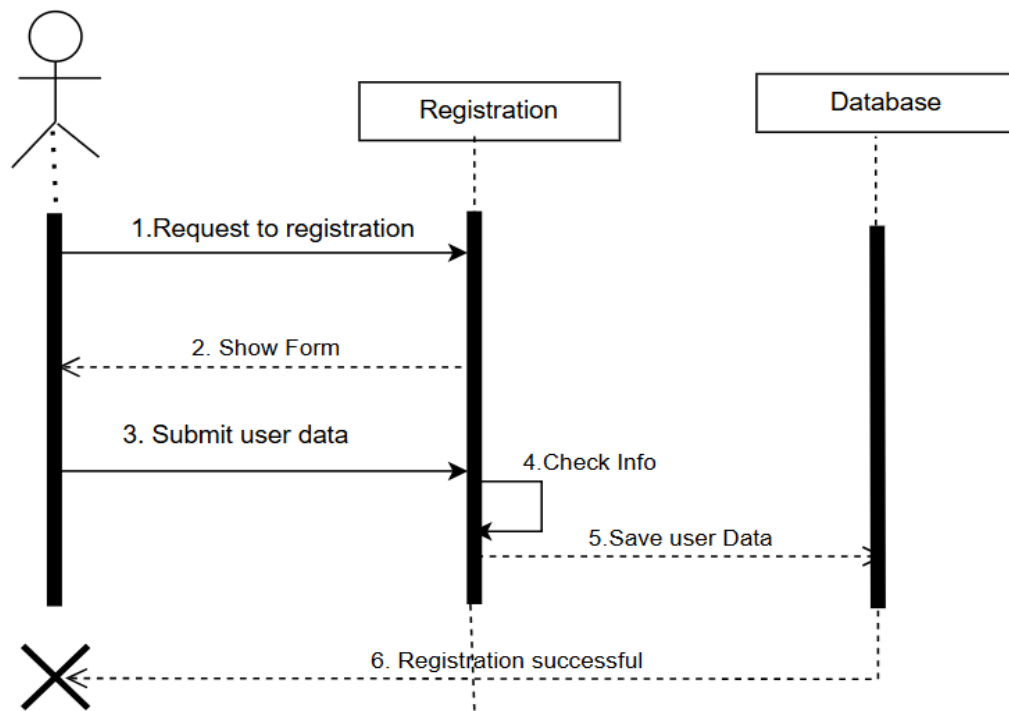


Figure 2.4.12: Sequence Diagram for Registration

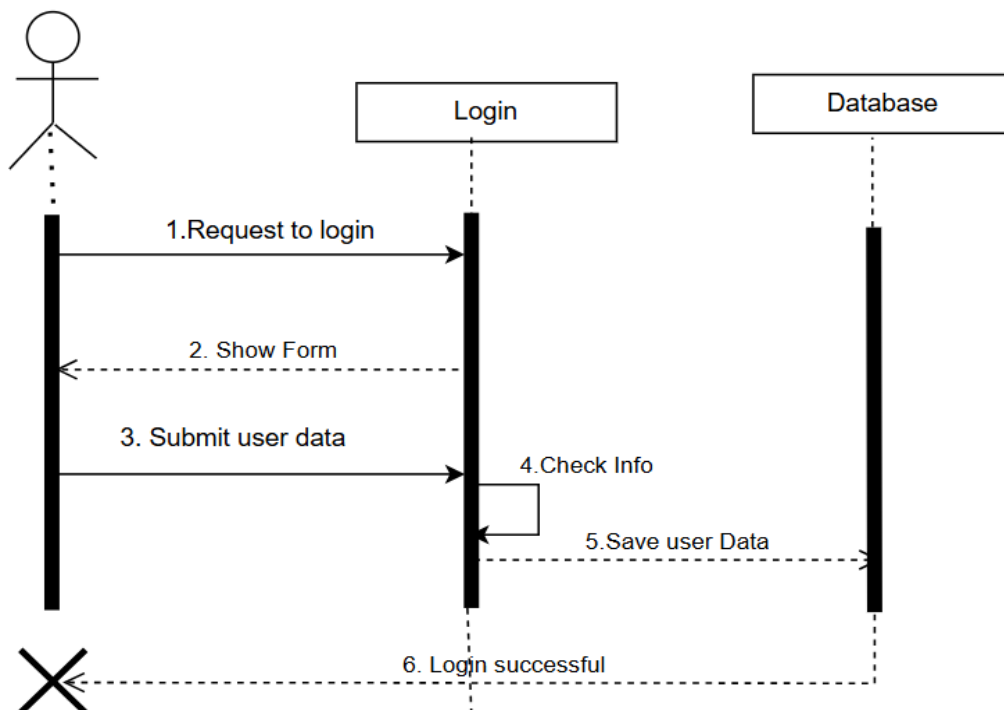


Figure 2.4.13: Sequence Diagram for Login

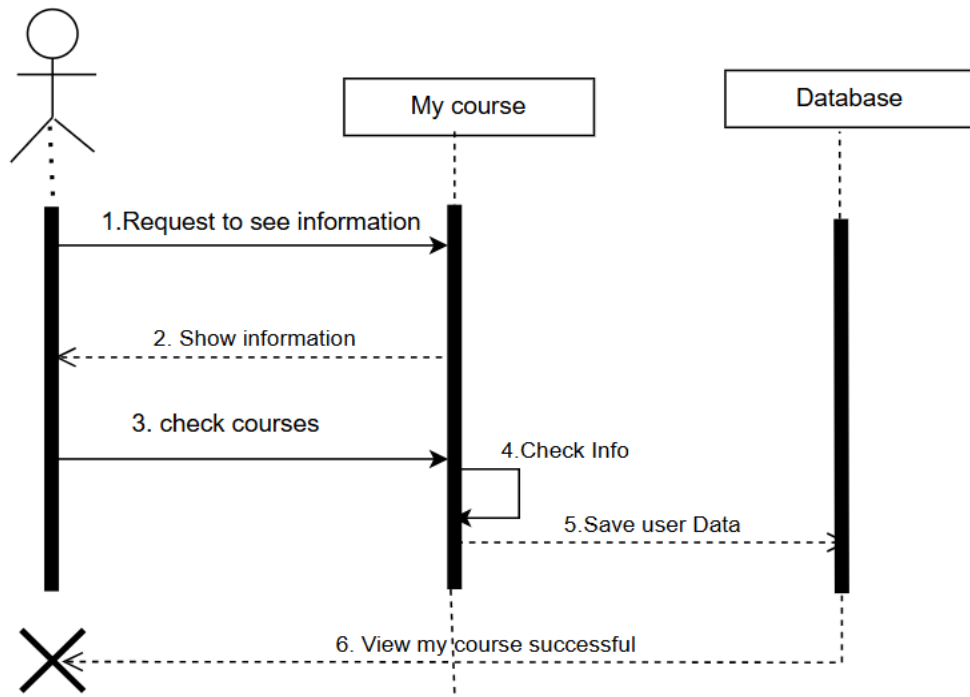


Figure 2.4.14: Sequence Diagram for My Course

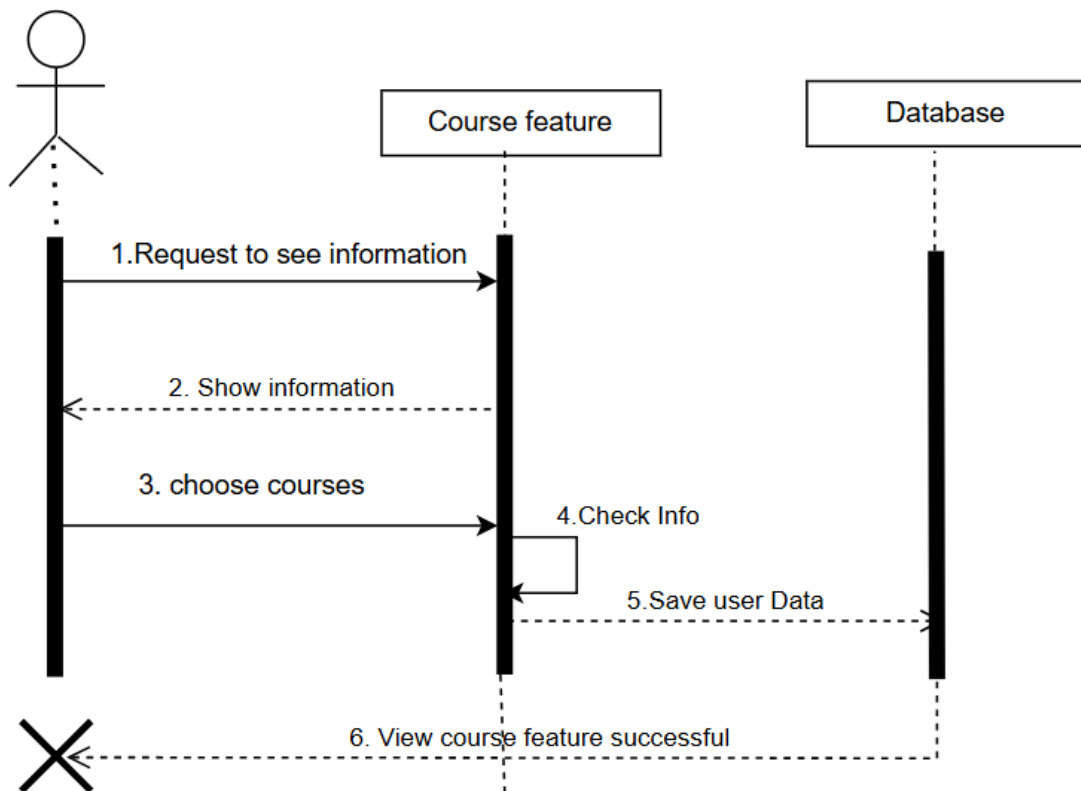


Figure 2.4.15: Sequence Diagram for Course Feature

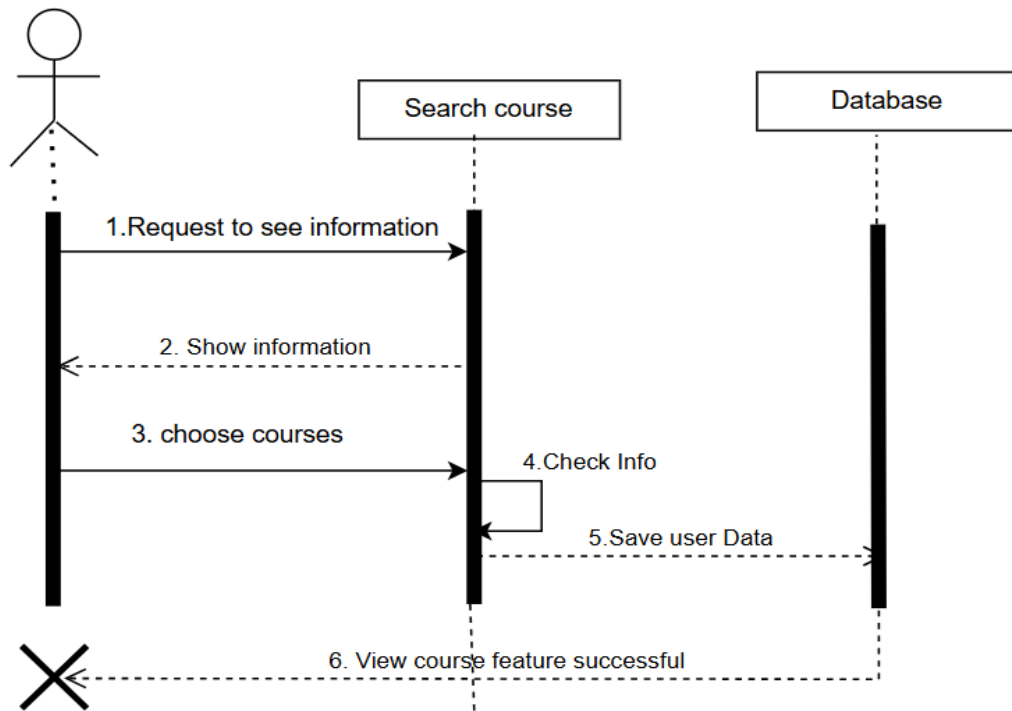


Figure 2.4.16: Sequence Diagram for Search Course

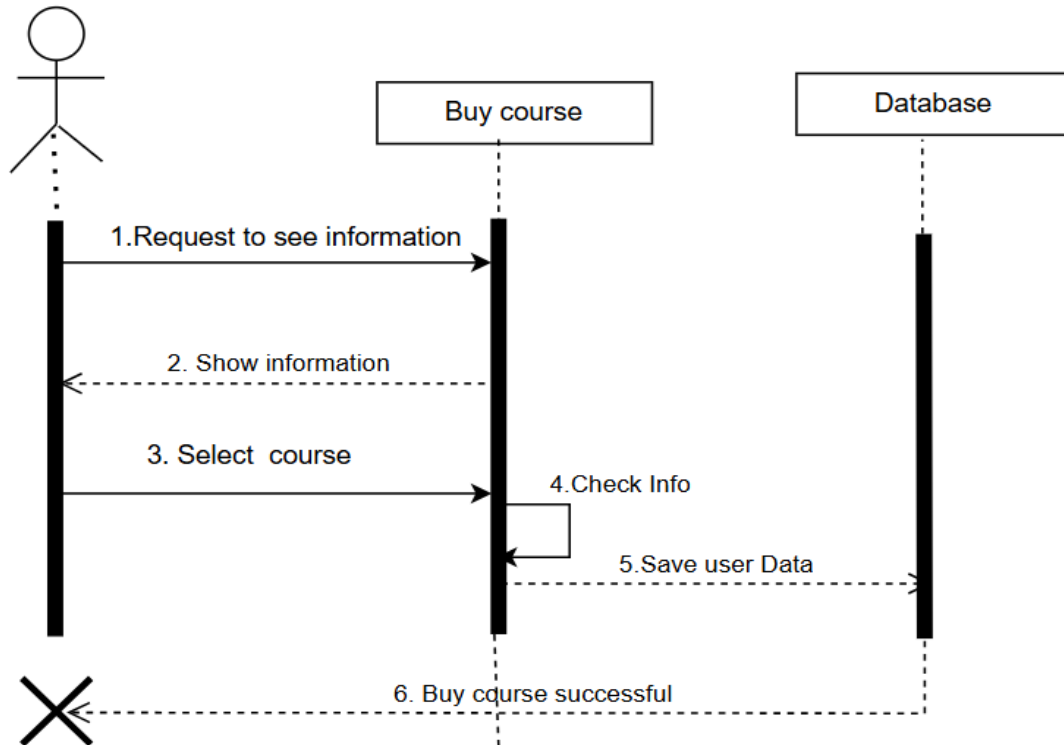


Figure 2.4.17: Sequence Diagram for Buy Course

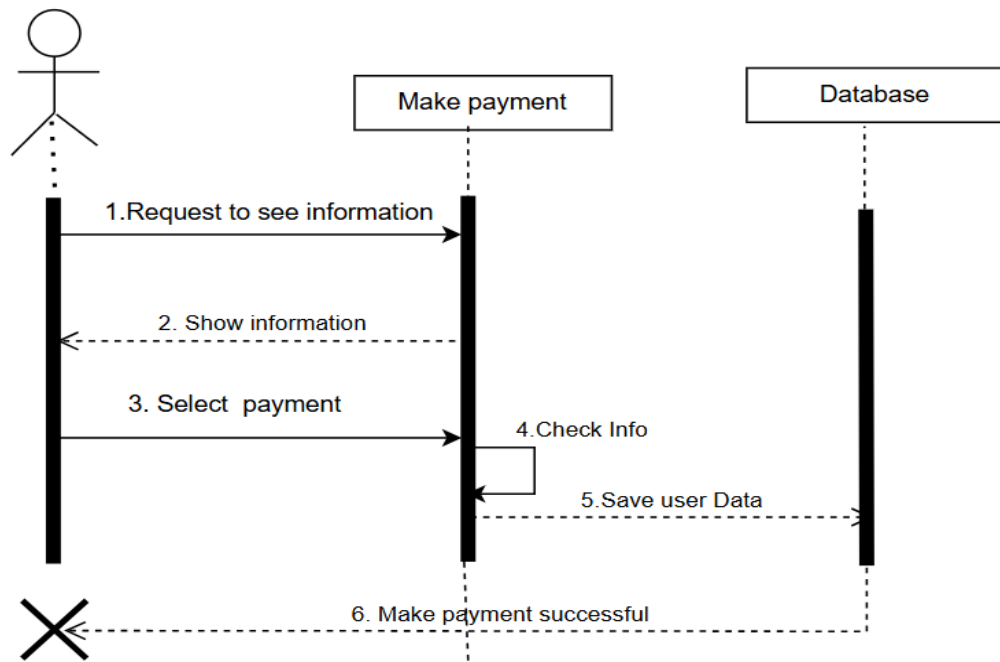


Figure 2.4.18: Sequence Diagram for Payment

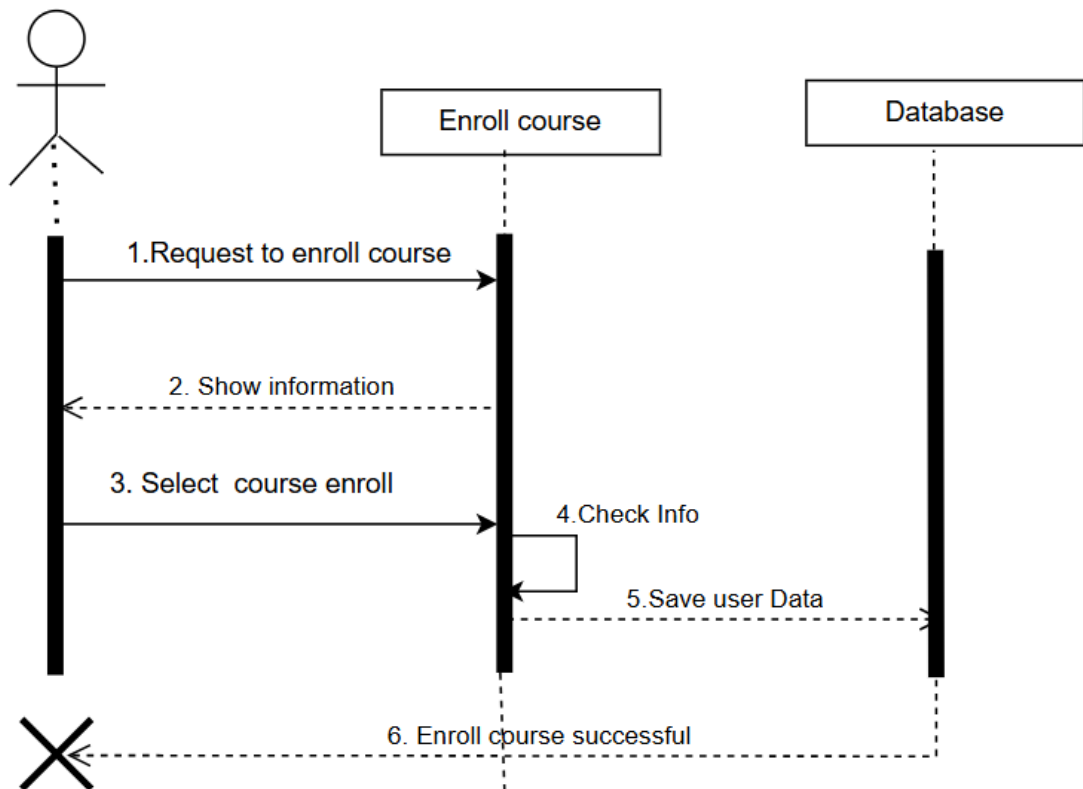


Figure 2.4.19: Sequence Diagram for Enroll Course

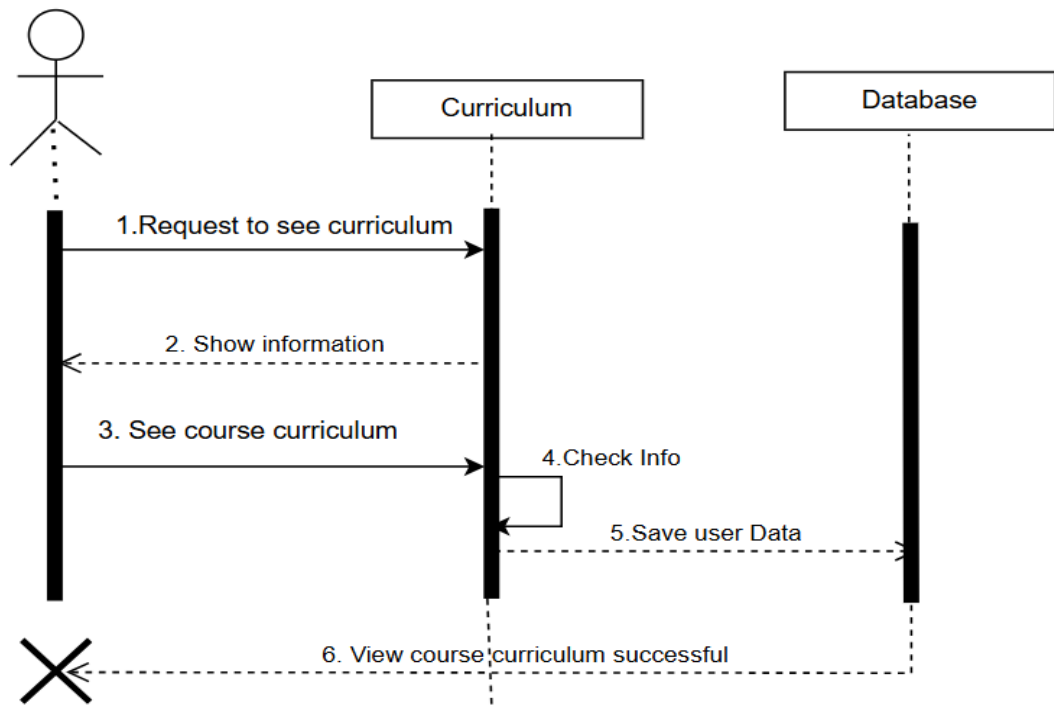


Figure 2.4.20: Sequence Diagram for Curriculum

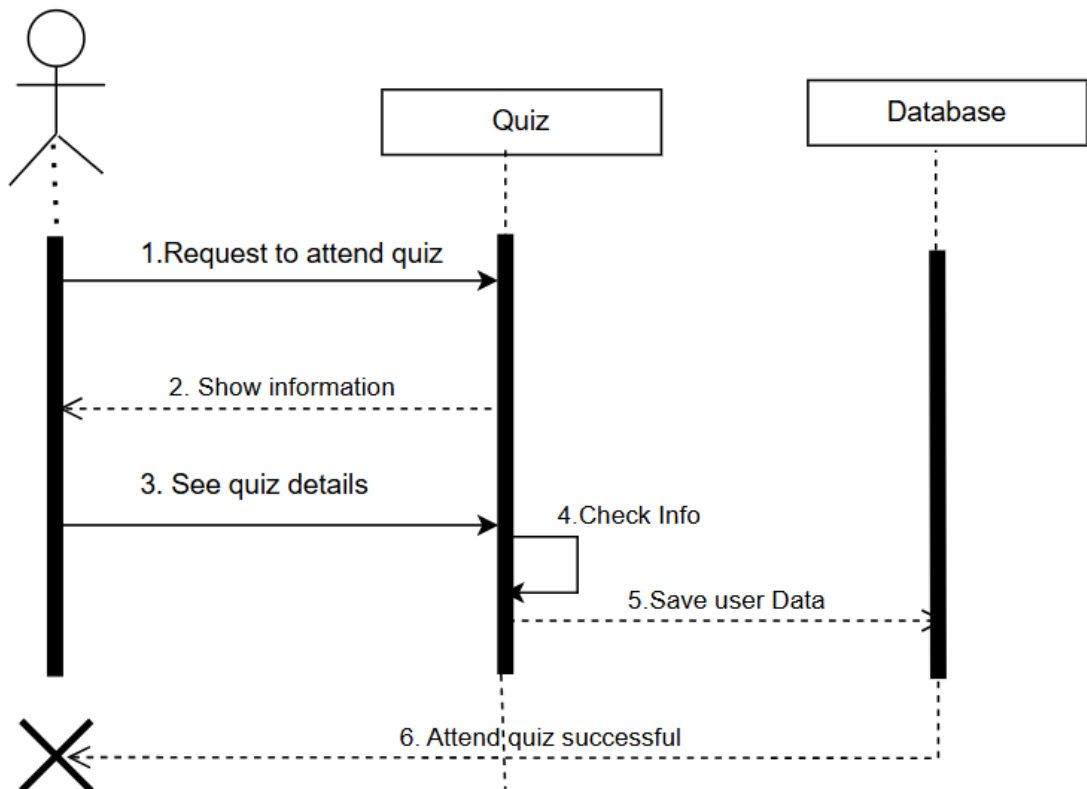


Figure 2.4.21: Sequence Diagram for Quiz

## 2.4.5 Class Diagram

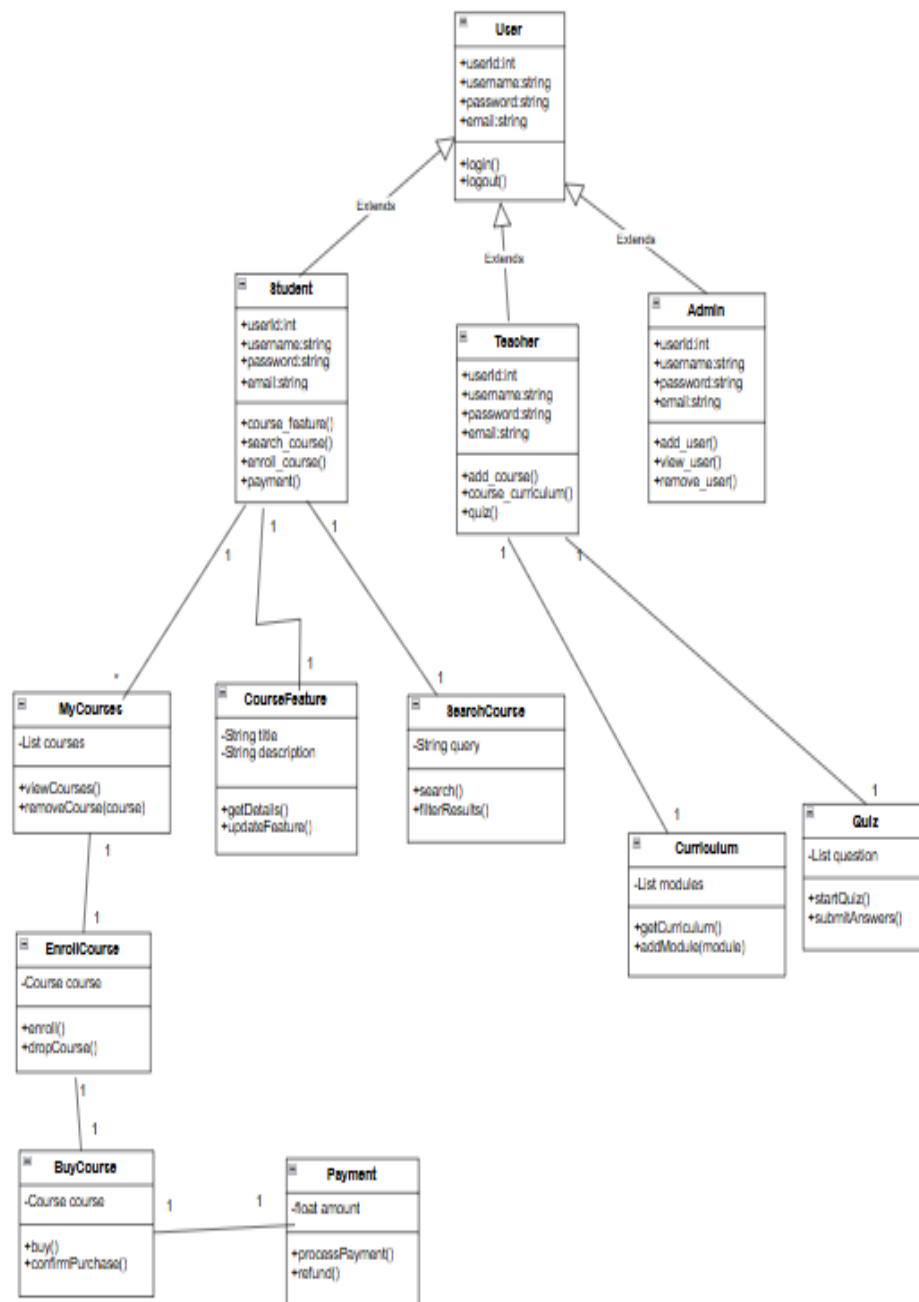


Figure 2.4.22: Class Diagram for Language Learning Website

## 2.4.6 ER Diagram

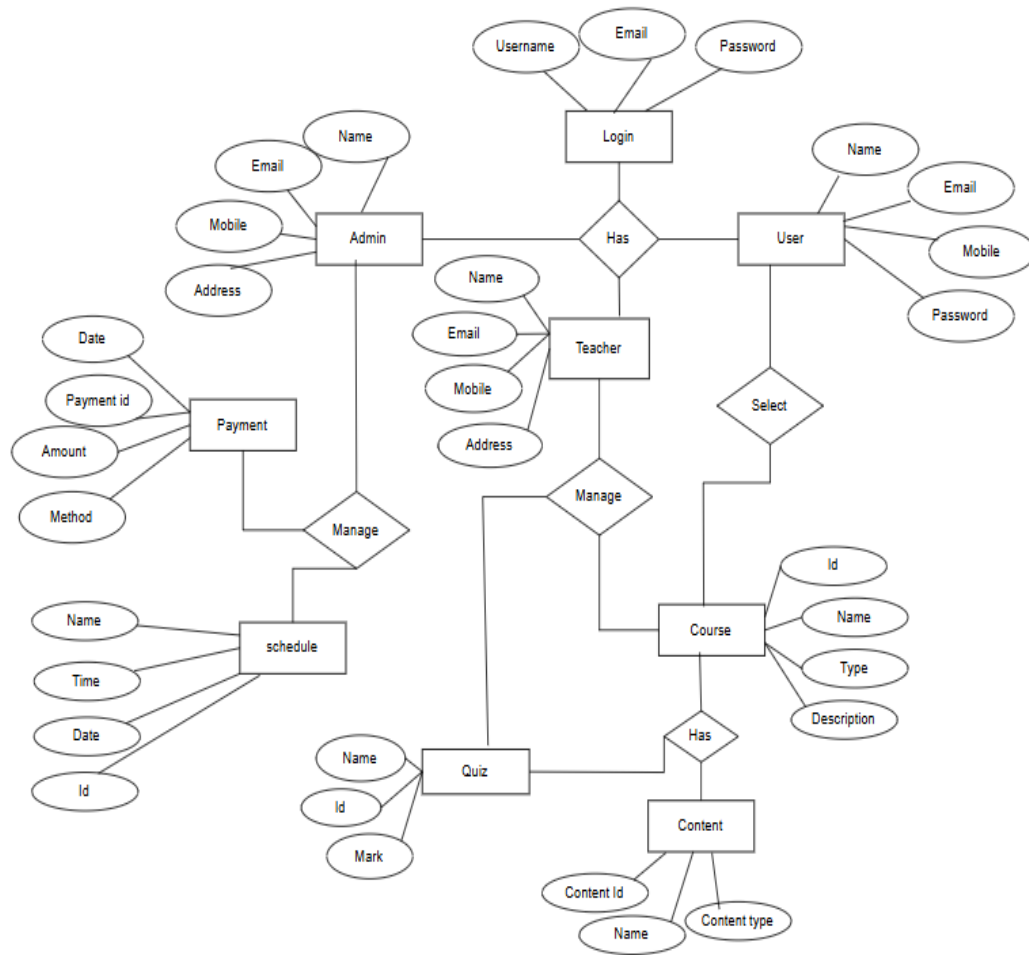


Figure 2.4.23: Entity Relationship Diagram for Language Learning Website

## 2.5 Coding: Appendix A

Application testing is critical to confirm the coding language learning site meets its functional and non-functional requirements, thus providing a reliable and efficient learning experience. Before launching, testing involves identifying and diagnosing standards, usability challenges, and differences. Project uses a rigorous testing method to ensure platform interoperability across multiple browsers, operating systems, and devices; study performance under different load conditions; and test stability against different user actions. This encompasses everything from the assessment of interactive code environments, auto-feedback systems, user tracking, community forum features, and charging for a course. Performance, security, accessibility, and usability will take center stage as non-functional testing focuses on aspects beyond just the core function. It will ensure perfect compatibility with all code editors, browsers, and devices. Testing will also assess whether things function offline, where applicable. Test cases reproducing real user scenarios and edge cases will be created. A combination of manual testing practices and automated testing technologies will be used to enhance the efficiency and accuracy of the testing process. The site for learning the coding language wants to be a reliable, user-driven, high-performing place for coding education, which is why it goes through such a thorough testing process.

## 2.6 Summary

Following consistent testing, the project guarantees interoperability between different browsers and operating system systems. A variety of user interactions, devices, and realistic load performance are all examples of reliability. Interactive coding, automatic feedback, user progress monitoring, community forums, and money processing are all areas that are being tested to ensure their functionality. The speed, accessibility, security, and usability of a system are all evaluated during non-functional testing. Through compatibility testing, a simple connection may be established with code editors, browsers, and devices. A possible investigation into offline functioning will take place. Multiple test cycles will be coordinated with the development process in order to imitate the conditions and behaviors of users that are encountered in the actual world. The efficiency and efficacy of our testing will be improved through the utilization of both human and automated testing methods. The website for learning coding language aims to provide a platform for teaching coding that is optimal, efficient, and user-friendly. This is accomplished through extensive testing.

# Chapter 3 Software Testing

## 3.1 Introduction

Writing a coding language learning website requires software testing to ensure it meets functional and non-functional requirements and provides a good user experience. Before launch, the platform is carefully tested for flaws, performance bottlenecks, security vulnerabilities, and usability. Using multiple testing methods, the website's interoperability across browsers, operating systems, and devices, performance under varying load conditions, and accessibility are verified. Delivering a stable, secure, and user-friendly learning environment requires thorough testing.

## 3.2 Testing Features

### 3.2.1 Feature to Be Tested

- a. User Registration
- b. User Login
- c. Admin Login
- d. Course Feature
- e. Buy Course
- f. Enrol Course
- g. payment
- h. Quiz

## 3.3 Testing Strategies

### 3.3.1 Test Approach

This coding language instruction website will use manual and automated testing throughout construction. Unit testing checks individual components and functions, integration testing checks module interactions, system testing checks the entire platform against requirements, and user acceptance testing (UAT) gathers user feedback. User registration, course navigation, interactive coding environments, and feedback will be functionally tested. Non-functional testing evaluates usability, security, performance, and accessibility. For repeated processes like regression testing, test automation will be employed to improve efficiency and coverage. This comprehensive strategy identifies and fixes faults early in development to ensure a high-quality, reliable learning environment.

### 3.3.2 Pass/Fail Criteria

The pass/fail criteria for this coding language learning website are determined by pre-established standards and user expectations. A feature passes functional testing if it functions as defined in the requirements specification, giving the proper result for valid inputs while gracefully managing faulty inputs. Non-functional testing will use specific metrics: performance testing will have thresholds for page load times, response times, and concurrent user capacity; security testing will require the absence of identified vulnerabilities and adherence to security best practices; usability testing will assess ease of navigation, clarity of instructions, and overall user satisfaction, with predefined acceptance levels; and accessibility testing will verify compliance with Any variation from these requirements will result in a "fail" classification, indicating that the feature or component requires remedial activity before it can be considered ready for release.

### 3.4 System Testing (Test Cases with Report)

Table 3.4.1: Test Case Report for User Registration.

<b>Test Case: 3.4.1</b>			<b>Test Case Name:</b> User Registration				
<b>System:</b> Language Learning website			<b>Subsystem:</b> User Authentication				
<b>Designed by:</b> Saikat Barman			<b>Design Date:</b> 25/12/2024				
<b>Executed by:</b> Saikat Barman			<b>Execution Date:</b> 25/12/2024				
			<b>Description:</b> The user registers for the Language Learning System by providing valid registration information.				
			<b>Pre-condition:</b> The user accesses the registration page.				
Step	Name	Email	Password	Retype password	Response	Pass / Fail	Comment
1	Saikat Barman	Saikatbarman60@gmail.com	1234	1234	Registration Successful	Pass	Registration is successful with valid input.
2	Saikat Barman	Saikatbarman60@gmail.com	123456	123546	Password mismatch	Fail	Password and retype password must match.
3	Saikat Barman		123456	123456	Email field empty	Fail	User must input an email.
4		Saikatbarman60@gmail.com	123456	123456	Name field empty	Fail	User must input a name.
<b>Post-condition:</b> The user is successfully registered with valid information, and the process is complete.							

Table 3.4.2: Test Case Report for User Login.

<b>Test Case: 3.4.2</b>		<b>Test Case Name: User Login</b>			
<b>System:</b> Language Learning website		<b>Subsystem:</b> User Authentication			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user logs into the system by providing valid credentials.			
		<b>Pre-condition:</b> The user accesses the login page.			
Step	Email	Password	Response	Pass/ Fail	Comment
1	Saikatbarman60@gmail.com	123456	Login successful	Pass	User logs in successfully with valid credentials
2	Saikatbarman601@gmail.com	123456	Login failed	Fail	Email not registered.
4	Saikatbarman60@gmail.com	100000	Login failed	Fail	Incorrect password.
<b>Post-condition:</b> The user successfully logs into the system with valid credentials.					

Table 3.4.3: Test Case Report for Admin Login.

<b>Test Case: 3.4.3</b>		<b>Test Case Name: Admin Login</b>			
<b>System:</b> Language Learning website		<b>Subsystem:</b> User Authentication			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The admin logs into the system by providing valid credentials.			
		<b>Pre-condition:</b> The admin accesses the login page.			
Step	Email	Password	Response	Pass/ Fail	Comment
1	admin60@gmail.com	123456	Login successful	Pass	Admin logs in successfully with valid credentials
2	admin601@gmail.com	123456	Login failed	Fail	Email not registered.
4	admin60@gmail.com	100000	Login failed	Fail	Incorrect password.
<b>Post-condition:</b> The admin successfully logs into the system with valid credentials.					

Table 3.4.4: Test Case Report for Course Feature.

<b>Test Case: 3.4.4</b>		<b>Test Case Name:</b> Course feature			
<b>System:</b> Language Learning Website		<b>Subsystem:</b> Learning Platform			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user browses the course feature and views course details.			
		<b>Pre-condition:</b> The user accesses the course feature page.			
Step	Name	Action	Response	Pass/ Fail	Comment
1	View Course	Click on "python"	Course displayed	Pass	Course under the category are displayed.
2	Search Course	Search for "php"	Results displayed	Pass	Relevant results are shown.
3	Invalid Search	Search for "xyz123"	No course found	Pass	Proper message displayed for no results.
<b>Post-condition:</b> The user successfully browses course feature and views details.					

Table 3.4.5: Test Case Report for Buy Course.

<b>Test Case: 3.4.5</b>		<b>Test Case Name:</b> Buy Course			
<b>System:</b> Language Learning Website		<b>Subsystem:</b> Learning Platform			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user browses the buy course and views buy course details .			
		<b>Pre-condition:</b> The user accesses the buy course page.			
Step	Name	Action	Response	Pass/ Fail	Comment
1	Valid Sale	Enter course details and total	Receipt generated	Pass	Sales processed successfully
2	Invalid Course	Enter non-existent course ID	Error: course not available	Fail	Only valid course IDS are accepted
3	Empty Cart	Proceed without adding course	No course found	Fail	Purchasing cannot be processed without course
<b>Post-condition:</b> The user successfully browses buy course and views details.					

Table 3.4.6: Test Case Report for Enroll Course.

<b>Test Case: 3.4.6</b>		<b>Test Case Name:</b> Enroll Course			
<b>System:</b> Language Learning Website		<b>Subsystem:</b> Learning Platform			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user enroll course and views details .			
		<b>Pre-condition:</b> The user accesses the enroll course page.			
Step	Name	Action	Response	Pass/ Fail	Comment
1	Enrollment course	Click on enroll course	Enrollment id generated	Pass	Enroll course successfully
2	Invalid Enrollment	Enter non-existent enrollment ID	Error: enrollment id not available	Fail	Only valid enroll IDS are accepted
3	Empty enrollment key	Proceed without adding enrollment id	No enroll id found	Fail	Course enrollment cannot be processed without buying
<b>Post-condition:</b> The user successfully enroll course and views details.					

Table 3.4.7: Test Case Report for payment.

<b>Test Case: 3.4.7</b>		<b>Test Case Name:</b> Payment			
<b>System:</b> Language Learning Website		<b>Subsystem:</b> Learning Platform			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user browses the payment details.			
		<b>Pre-condition:</b> The user accesses the payment page.			
Step	Name	Action	Response	Pass/ Fail	Comment
1	Payment	Enter payment details and total	Receipt generated	Pass	Payment processed successfully
2	Invalid Payment	Enter non-existent payment ID	Error: payment not found	Fail	Only valid course IDS are accepted
3	Empty Cart	Proceed without adding course	No course found	Fail	Payment cannot be processed without course
<b>Post-condition:</b> The user successfully payment and views details.					

Table 3.4.8: Test Case Report for Quiz.

<b>Test Case: 3.4.8</b>		<b>Test Case Name:</b> Quiz			
<b>System:</b> Language Learning Website		<b>Subsystem:</b> Learning Platform			
<b>Designed by:</b> Saikat Barman		<b>Design Date:</b> 25/12/2024			
<b>Executed by:</b> Saikat Barman		<b>Execution Date:</b> 25/12/2024			
		<b>Description:</b> The user browses the quiz details.			
		<b>Pre-condition:</b> The user accesses the quiz page.			
Step	Name	Action	Response	Pass/Fail	Comment
1	Quiz	Click on quiz	Quiz displayed	Pass	View quiz successfully
2	Invalid Quiz	Enter non-existent quiz ID	Error: quiz not found	Fail	Only valid quiz IDS are accepted
3	Empty quiz	Proceed without completing course	No quiz found	Fail	Quiz cannot be processed without completing course
<b>Post-condition:</b> The user successfully view quiz details.					

### 3.5 Summary

To ensure it meets all functional and non-functional requirements and provides a stable, effective learning environment, the website for learning computer programming must undergo thorough software testing. Testing is done continuously during development to identify and resolve issues with performance, consistency, and problems before launch. We target to deliver strong performance under diverse conditions and also target to evaluate the compatibility among diverse combinations of hardware and software. Test Domains include but Are not limited to safe financial transactions, community forum features, automatic feedback systems, user progress tracking and interactive coding environments. Non-Functional Testing is based on four pillars which are Usability, Accessibility, Reliability, and performance. But that isn't what compatibility testing is all about — Far from it. We shall consider offline capabilities if it is applicable.

# Chapter 4 Deployment and Maintenance

## 4.1 Introduction

The Deployment and Maintenance phase takes the development of the coding language learning website from a model to an active, user-facing application, maintaining career website up and running. Deployment also involves the appropriate server setup. This is followed up with infrastructure setup, software configuration, deployment, and production testing. Maintenance can include ongoing tasks like bug fixes, security updates, performance monitoring, content updates, and support. This stage includes collecting user feedback and building adjustments to align with trends in the market allowing the platform to succeed and stay relevant for the long-term.

## 4.2 Try to follow the SRLC (software release life cycle)

The code language learning website will have a solid and user-centric solution with the help of systematic Software Release Life Cycle (SRLC). Philosophy obviously thus the journalist perspective, we are starting with pre alpha phase focused on needs collecting, feasibility study and preliminary architecture of systems. For example, early prototypes analyze stakeholder comments in order to construct functions that are needed such as interactive lessons, coding exercises, and community features. The alpha phase comprises the creation and testing in controlled internal environment of basic modules. Established and tested against user requirements are fundamental learning characteristics including automated feedback systems and code execution environments. This phase includes constant debugging and improvement to ensure that key parts run as expected. A prototype is distributed to a small group of target users—aspiring coders, teachers—for empirical review during the beta phase. We gather usability information and assess pragmatic situations such handling concurrent users, different browsers, and several code editors. To produce almost bug-free versions, the release candidate process assures the implementation and thorough testing of all necessary aspects. Excellent problems have been fixed, and the system is now dependable and top performance ready. Assessments of security and accessibility take place. All of the documentation—user manuals and FAQs—has been finished. The general availability phase marks the public release of the website, therefore making it reachable to every user. Administered are post-launch continuous maintenance, bug corrections, security enhancements, feature additions. Following the SRLC helps the project to offer a consistent, effective, and user-friendly coding instruction platform.

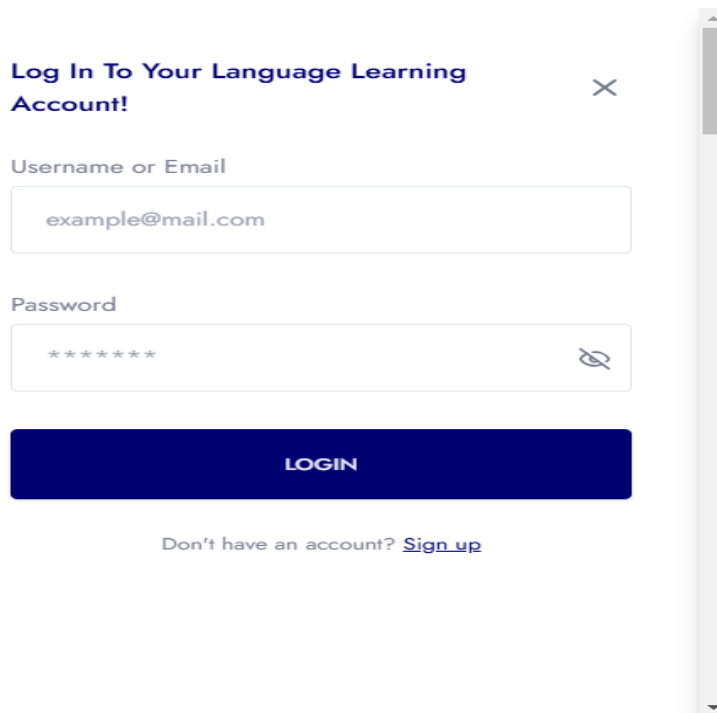
## **Chapter 5 User Manual**

### **5.1 Introduction**

A language learning website follows a similar Software Release Life Cycle (SRLC) to ensure strength and efficiency. The pre-alpha phase establishes core features such to track progress, expand vocabulary, and practice pronunciation. We create functioning mockups and prototypes to visualize UI and UX. Technical checkout thoroughly tests audio playback, speech recognition, and database interfaces. A beta implementation gathers input from a small portion of the target population on the product's technical performance, usability, and lessons. Gamification, community forums, and personalized learning routes are routinely included at this level. The release candidate stage indicates near-completion after implementing all key features, evaluating performance and security, and creating full documentation (including user manuals and API documentation). Later, the public premiere is characterized by broad availability, and there will be ongoing maintenance, bug fixes, and upgrades that add new content, features, and performance. These updates reflect user feedback and language learning trends. This deliberate approach ensures a user-focused, reliable, and adaptable platform that serves several needs.

### **5.2 Project Functionalities**

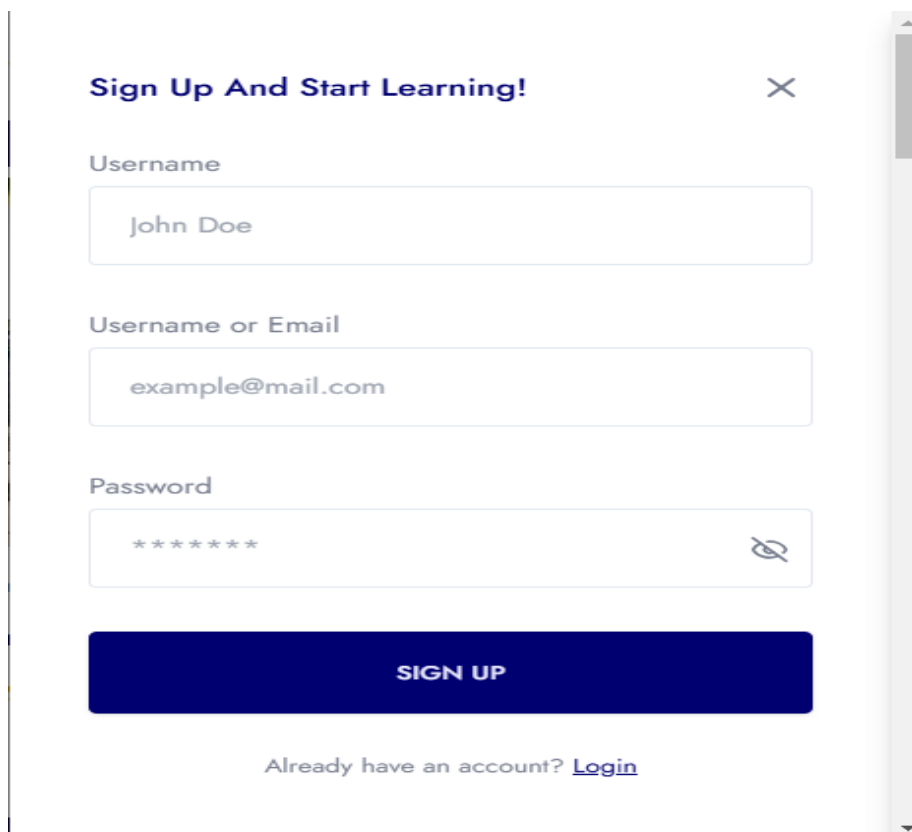
## User Login



The image shows a user login form with a dark blue header and a white background. The header contains the text "Log In To Your Language Learning Account!" and a close button (X). Below the header are two input fields: "Username or Email" with the placeholder text "example@mail.com" and "Password" with placeholder text "\*\*\*\*\*" and a toggle icon. A dark blue "LOGIN" button is positioned below the fields. At the bottom, there is a link: "Don't have an account? [Sign up](#)". A vertical scrollbar is visible on the right side of the form.

Figure 5.2.1: User Log in page

## User Registration



The image shows a user registration form with a dark blue header and a white background. The header contains the text "Sign Up And Start Learning!" and a close button (X). Below the header are three input fields: "Username" with the placeholder text "John Doe", "Username or Email" with the placeholder text "example@mail.com", and "Password" with placeholder text "\*\*\*\*\*" and a toggle icon. A dark blue "SIGN UP" button is positioned below the fields. At the bottom, there is a link: "Already have an account? [Login](#)". A vertical scrollbar is visible on the right side of the form.

Figure 5.2.2: User Registration page

Home Page:



Figure 5.2.3: Language learning Home page

Display course and resource page:

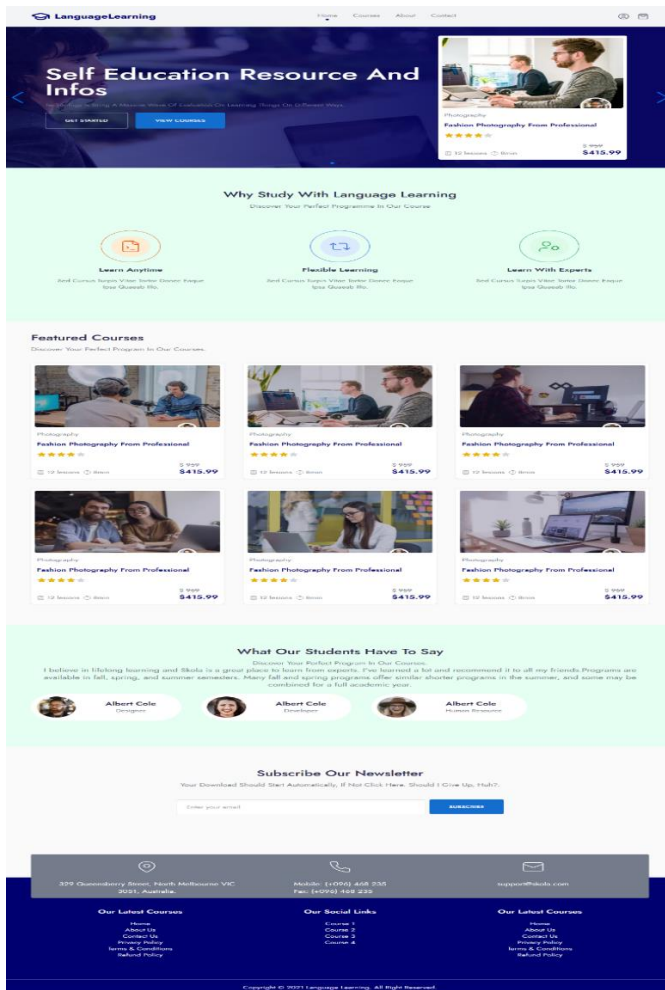
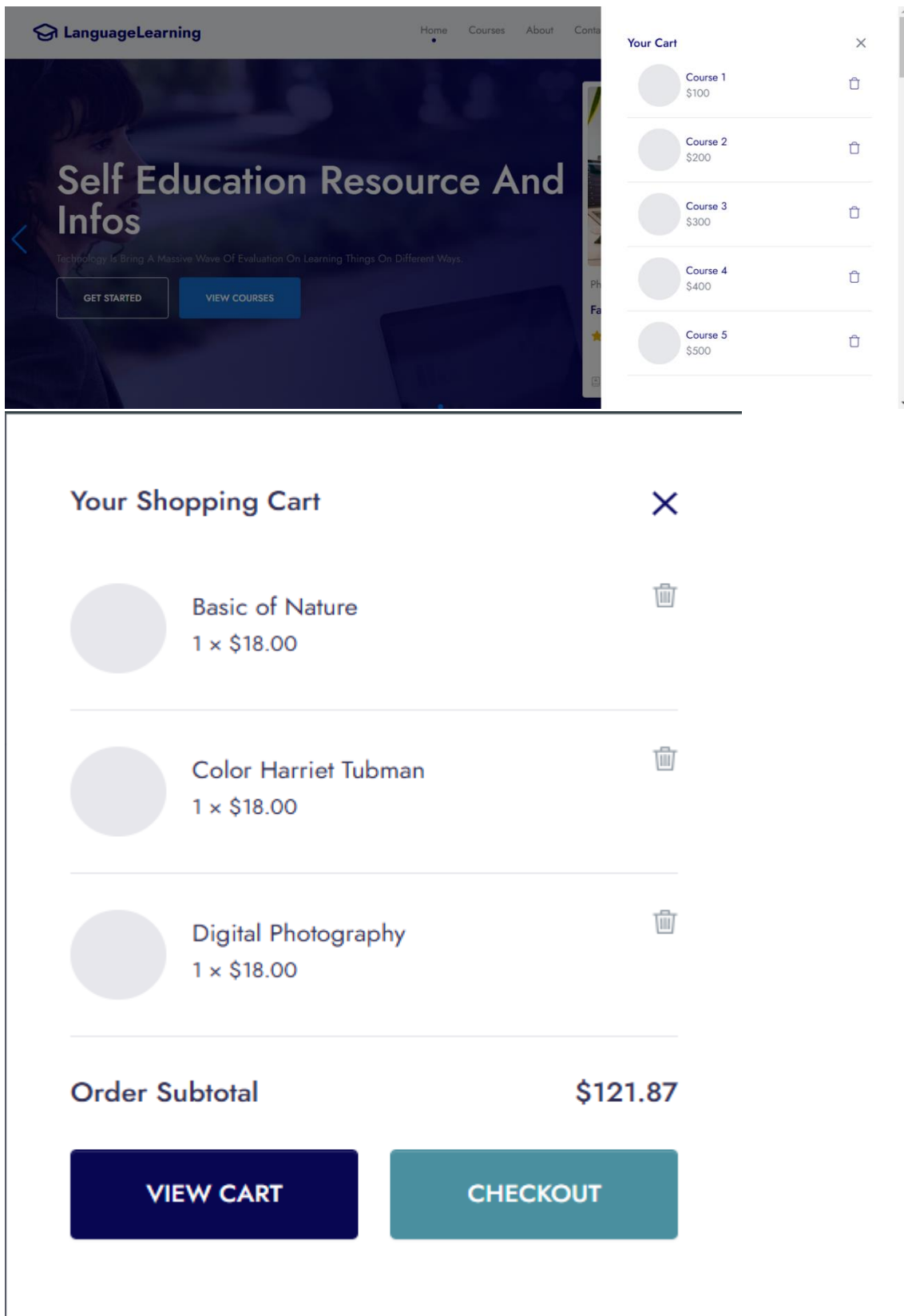


Figure 5.2.4: Display course and resource page

**View Cart:**



**Figure 5.2.5: Display add to cart page**

## Course description:

Home / Courses List / Courses Single

### Learn Figma: User Interface Design Essentials - UI/UX Design

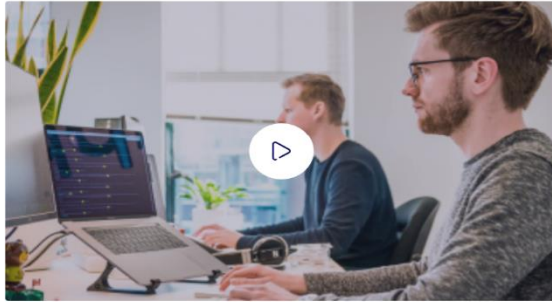
Master Figma app to get a job in UI Design, User Interface, User Experience design, Web Design & UX design.



**Created by**  
Alison Dawn

**Categories**  
Design

**Review**  
★★★★★



- Overview
- Curriculum
- Reviews

#### Course Description

Do you want to become a UI/UX designer but you don't know where to start? This course will allow you to develop your user interface design skills and you can add UI designer to your CV and start getting clients for your skills.

Hi everyone, I'm Arash and I'm a UI/UX designer. In this course, I will help you learn and master Figma app comprehensively from scratch. Figma is an innovative and brilliant tool for User Interface design. It's used by everyone from entrepreneurs and start-ups to Apple, Airbnb, Facebook, etc.

[Read More](#) +

#### What you'll learn

- ✓ Become a UI/UX designer.
- ✓ You will be able to start earning money skills.
- ✓ Build a UI project from beginning to end.
- ✓ Work with colors & fonts.
- ✓ You will create your own UI Kit.
- ✓ Build & test a complete mobile app.
- ✓ Learn to design mobile apps & websites.
- ✓ Design 3 different logos.
- ✓ Create low-fidelity wireframe.
- ✓ Downloadable exercise files.

#### Requirements

- We do not require any previous experience or pre-defined skills to take this course. A great orientation would be enough to master UI/UX design.
- A computer with a good internet connection.
- Adobe Photoshop (OPTIONAL)

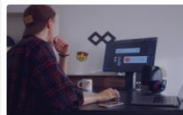
### Related Courses

Discover Your Perfect Program In Our Courses.



Photography  
**Fashion Photography From Professional**

5 lessons ★★★★★



Marketing  
**Productivity and Time Management for workplace**

5 lessons ★★★★★



Personal Development  
**Learn Ethical Hacking From Scratch**

5 lessons ★★★★★



Development  
**The Complete JavaScript Course 2020: Real...**

5 lessons ★★★★★



**\$89.99** ~~339.99~~ 91% OFF

🕒 2 days left at this price!

**BUY NOW**

**ENROLL**

🕒 Duration	43 weeks
📅 Lectures	32
👤 Enrolled	1982 students
🗣️ Language	English
📖 Skill level	beginner
📅 Deadline	06 April 2020
📄 Certificate	Yes

[Share this course](#)

Figure 5.2.6: Display course description page

### Course curriculum:

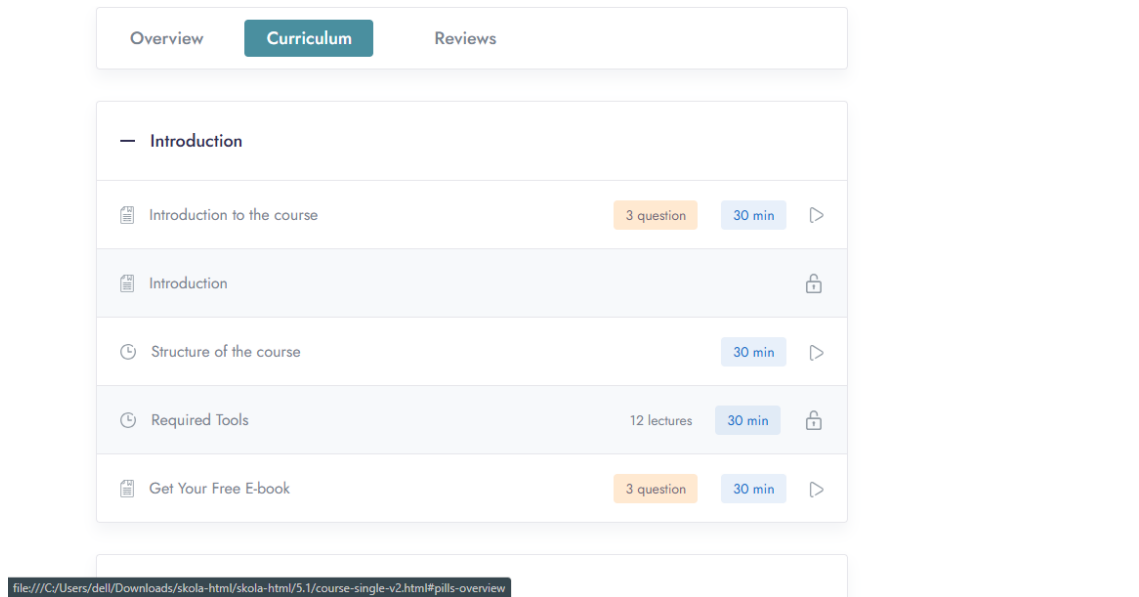


Figure 5.2.7: Display course curriculum page

### Course quiz:

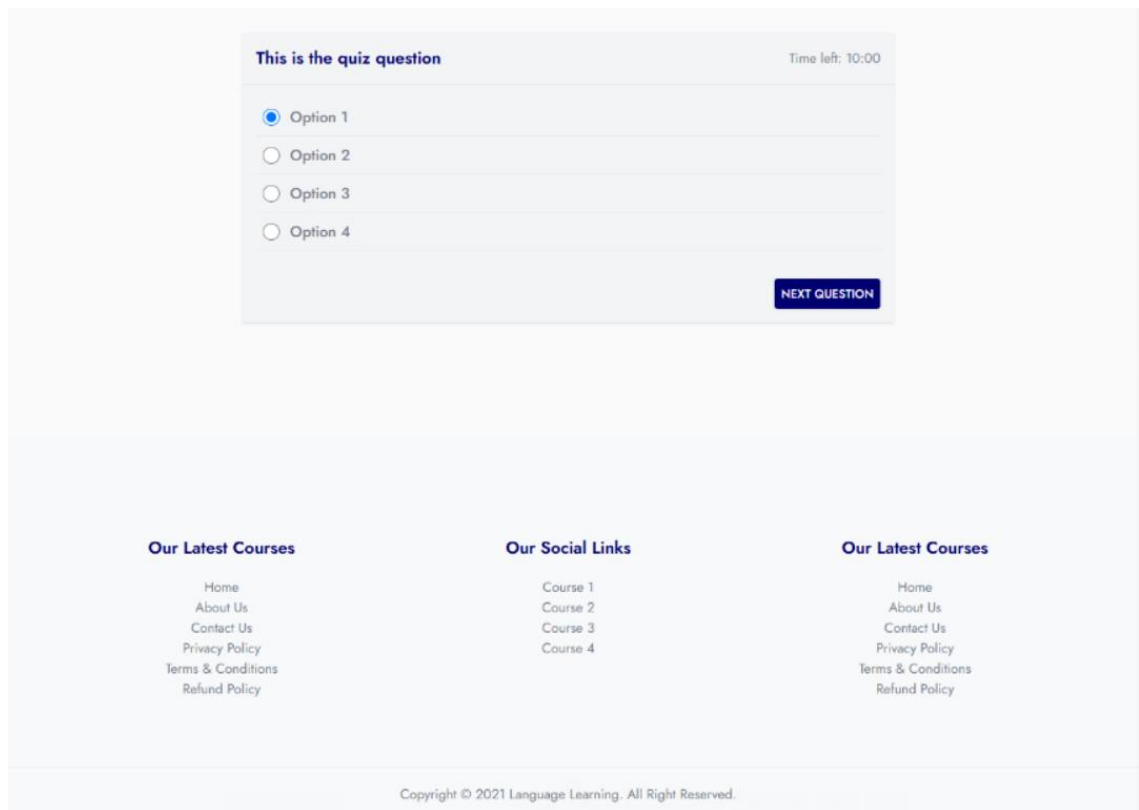


Figure 5.2.8: Display course quiz page

### **5.3 Summary**

The language learning website's user manual explains how to navigate its coding-focused features. This platform for budding programmers includes interactive coding exercises, a code editor with syntax highlighting and auto-completion, and more. Built-in compiler/interpreter provides rapid feedback. The web-based interface allows administrators to manage course material, add/update/delete courses, set coding challenges, and measure student progress. The coding environment supports many programming languages (e.g., Python, JavaScript, and Java) and offers debugging tools for error resolution. Integrating with version control systems like Git enables students to maintain and collaborate on code. The platform provides automatic code evaluations, sandboxes for experimentation, and interaction with online coding communities. The website, available in numerous languages and accessible across platforms, enables users to learn coding via interactive and practical experiences.

# Chapter 6 Project Summary

## 6.1 Introduction

This project is an online tool that is all connected and meant to help people learn how to code. The system improves many parts of learning, such as interactive lessons, coding exercises with automatic feedback, and community contact. It does this by giving people a central place to learn how to code and connect with other students. Some of the most important features are running code and getting feedback in real time, creating unique learning paths, and keeping track of progress. Some features that make learning easier are in-browser code editors, built-in instructions, and a helpful forum. The platform also lets users securely log in and control their profiles. The website for learning coding languages was made with the goal of being easy to use, accessible, and relevant to modern coding practices. It gives people the tools they need to learn useful tech skills and do well in the digital world.

## 6.2 Project Limitation

This website project for learning to code meets some basic educational goals, but it's not perfect. Some features, like advanced personalized learning algorithms or complex progress analytics, may not have been as deep as they could have been because of time limits. Core coding tasks and feedback tools are already in place, but in future versions, features like AI-powered code analysis or automatic code generation could be added. The choice of open-source tools and cloud infrastructure was based on cost, which could make it harder to get enterprise-level features like advanced security or dedicated support. For now, the main focus is on helping individual students, with only a small amount of help for big schools or corporate training programs. The platform tries to work with all browsers and devices, but some older or less popular browsers may not be fully supported at first. People who haven't used online learning tools much before might find it hard to get started because the focus is on functionality rather than providing a lot of training materials. Core learning material is planned, and the course library will continue to be updated and added to all the time. These problems will be fixed in later versions of the platform to make it more stable, scalable, and user-friendly generally.

## 6.3 Scope

This website for learning the coding language has tutorials that you can interact with, tasks that give you feedback automatically, and a community forum where you can get help from other people. Some of the most important features are code editors that run in the browser, real-time code processing and feedback, personalized learning paths, progress tracking, and safe user authentication. The platform is made to work with all major web browsers and be available on desktop computers, tablets, and smartphones. Users can keep track of their learning progress and find places where they can improve with the help of reporting and tracking tools. Features beyond the purview of this project include artificial intelligence-

powered code analysis, automated code development, sophisticated customized learning algorithms, interface with other learning platforms, and support of major educational institutions. Right now, the platform is designed for single students and may not initially include sophisticated project management capabilities or collaborative coding environments. Although most browsers are tried to be compatible with the platform, older or less popular browsers may not be fully supported. There are not any thorough user training tools available in the first version for including the platform into present teaching environments. Still, these elements could become accessible in the future. These omissions indicate that the project is focused on creating a system that can be utilized and upgraded in next editions to satisfy the demands of additional users and educational environments.

## **6.4 Future Work**

More development and growth would help the learning to code website to be better. Feedback and code analysis driven by artificial intelligence might provide consumers customized and thorough help. Creating autonomous code generating tools might improve education. Show good techniques and quick learning to get experience. Integrating external learning platforms and APIs may provide access to resources and material. Supporting collaborative coding environments and project management tools may enhance team-based learning and larger projects. Improving user analytics and reporting may provide insights into student learning and identify areas for program improvement. Adaptive learning algorithms may customize learning routes by tailoring content and speed to each user's requirements. Adding game-like elements, such as virtual or augmented reality, may increase user engagement and motivation. Providing comprehensive user training and assistance for large educational institutions will enhance the platform's usefulness and reach.

## **6.5 Conclusion**

This coding language learning website project combines interactive courses, coding exercises with automatic feedback, and a community forum, thereby creating a working online learning platform for coding. The initiative has demonstrated that it is feasible to present future

programmers of all ability levels have an easily available and simple learning environment. Among the most crucial lessons gained are the benefits of iterative development, continuous user feedback, and comprehensive technical limited and scalability analysis. The project reaches its core objectives by providing a forum for tailored learning experiences, practical coding skills, and a supportive learning community. Though several capabilities were intended for future development, the platform's basic functionality has been successfully established, therefore laying a firm basis for future expansion.

# REFERENCES

[1] "Python Crash Course, 2nd Edition" by Eric Matthes:

<https://archive.org/details/python-crash-course-2nd-edition-a-hands-on-project-based-introduction-to-programming/page/n5/mode/2up>

[2] "HTML and CSS: Design and Build Websites" by Jon Duckett:

[https://books.google.com.bd/books?id=aGjaBTbT0o0C&printsec=copyright&redir\\_esc=y#v=onepage&q&f=false](https://books.google.com.bd/books?id=aGjaBTbT0o0C&printsec=copyright&redir_esc=y#v=onepage&q&f=false)

[3] "Eloquent JavaScript, 3rd Edition" by Marijn Haverbeke:

<https://www.amazon.com/Eloquent-JavaScript-3rd-Introduction-Programming/dp/1593279507>

[4] "Clean Code: A Handbook of Agile Software Craftsmanship" by Robert C. Martin:

<https://www.amazon.com/Clean-Code-Handbook-Software-Craftsmanship/dp/0132350882>

[5] "The Pragmatic Programmer: From Journeyman to Master" by Andrew Hunt and David Thomas::

[https://books.google.com.bd/books?id=5wBQEp6rulAC&pg=PA61&source=gbs\\_selected\\_pages&cad=1#v=onepage&q&f=false](https://books.google.com.bd/books?id=5wBQEp6rulAC&pg=PA61&source=gbs_selected_pages&cad=1#v=onepage&q&f=false)

[6] "The C Programming Language, 2nd Edition" by Brian W. Kernighan and Dennis M. Ritchie:

[https://books.google.com.bd/books/about/The\\_C\\_Programming\\_Language.html?id=FGkPBQAAQBAJ](https://books.google.com.bd/books/about/The_C_Programming_Language.html?id=FGkPBQAAQBAJ)

[7] "Effective Java, 3rd Edition" by Joshua Bloch:

<https://github.com/david-sauvage/effective-java-summary>

## Barman 1.0

---

### ORIGINALITY REPORT

---

**16%**

SIMILARITY INDEX

**10%**

INTERNET SOURCES

**5%**

PUBLICATIONS

**13%**

STUDENT PAPERS

---

### PRIMARY SOURCES

---

<b>1</b>	<b>Submitted to Daffodil International University</b> Student Paper	<b>5%</b>
<b>2</b>	<b><a href="https://dspace.daffodilvarsity.edu.bd:8080">dspace.daffodilvarsity.edu.bd:8080</a></b> Internet Source	<b>5%</b>
<b>3</b>	<b>Submitted to NCC Education</b> Student Paper	<b>2%</b>
<b>4</b>	<b>Submitted to Informatics Education Limited</b> Student Paper	<b>1%</b>
<b>5</b>	<b>Howard Anderson, Sharon Yull, Bruce Hellingsworth. "Higher National Computing - Core Units for BTEC Higher Nationals in Computing and IT", Newnes, 2004</b> Publication	<b>1%</b>
<b>6</b>	<b>Submitted to Universiti Teknologi Malaysia</b> Student Paper	<b>1%</b>
<b>7</b>	<b><a href="http://www.coursehero.com">www.coursehero.com</a></b> Internet Source	<b>&lt;1%</b>
<b>8</b>	<b><a href="http://www.slideshare.net">www.slideshare.net</a></b> Internet Source	<b>&lt;1%</b>

---

9	Submitted to Midlands State University Student Paper	<1 %
10	Submitted to Kingston University Student Paper	<1 %
11	Submitted to A.B. Paterson College Student Paper	<1 %
12	Submitted to University of Colombo Student Paper	<1 %
13	hugepdf.com Internet Source	<1 %
14	dev.to Internet Source	<1 %
15	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
16	Submitted to Wawasan Open University Student Paper	<1 %
17	access.atis.org Internet Source	<1 %
18	exatosoftware.com Internet Source	<1 %
19	eprints.utar.edu.my Internet Source	<1 %
20	odr.chalmers.se	

Internet Source

<1 %

21

[northcoders.com](http://northcoders.com)

Internet Source

<1 %

22

[www.ijntse.com](http://www.ijntse.com)

Internet Source

<1 %

23

Jacob Cohen. "Statistical Power Analysis for the Behavioral Sciences", Lawrence Erlbaum Associates, 2013

Publication

<1 %

24

[collegeesays369.blogspot.com](http://collegeesays369.blogspot.com)

Internet Source

<1 %

25

[www.cs.ru.nl](http://www.cs.ru.nl)

Internet Source

<1 %

26

[umpir.ump.edu.my](http://umpir.ump.edu.my)

Internet Source

<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On

