Title: Digital Storytelling Project

BY

Name: Aninda Rahman Deepto

ID: 213-16-594

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer and Information System (Major: Artificial Intelligence in IOT)

Supervised By
Md. Sarwar Hossain Mollah
Associate Professor & Head
Department of CIS
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

14 OCTOBER 2025

APPROVAL

This Project titled "**Digital Storytelling Project**", submitted by **Name: Aninda Rahman Deepto**, ID:**213-16-594** to the Department of CIS, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in CIS and approved as to its style and contents. The presentation has been held on 14-10-2025.

BOARD OF EXAMINERS

Md. Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing
& Information Technology
Daffodil International University

Chairman

Md. Nasimul Kader
Assistant Professor
Department of Computing
& Information Technology
Daffodil International University

Internal Examiner

Md. Mehedi Hassan
Lecturer (Senior Scale)
Department of Computing
& Information Technology
Daffodil International University

Internal Examiner

Ahmed Saif Reza,
Managing Director & Chief Technology Officer,
Medico Bio Limited

External Examiner

DECLERATION

We hereby declare that, this project has been done by us under the supervision of **Md. Sarwar Hossain Mollah, Associate Professor and Head,** Department of Computing & Information System, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Md. Sarwar Hossain Mollah
Associate Professor and Head
Department of CIS
Daffodil International University

Submitted by:
Aninda Rahman Deepto
ID: 213-16-594
Department of CIS
Daffodil International University

Acknowledgement

First, we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing making us possible to complete the final year project successfully.

We are grateful and wish our profound indebtedness to **Md. Sarwar Hossain Mollah, Associate Professor and Head.** The Deep knowledge & keen interest of my supervisor in the field of "Web development" helped to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express my heartiest gratitude to **Md. Sarwar Hossain Mollah, Associate Professor and Head,** Department of CIS, for his kind help to finish my project and also to other faculty members and the staff of the CIS department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

Executive Summary

The "AI Coding Learning System," a web-based instructional tool created by the Digital Storytelling Project, aims to improve programming learning through artificial intelligence. By integrating lesson monitoring, adaptive learning, and performance-based feedback, the initiative seeks to improve the efficiency, personalization, and interactivity of coding instruction.

The system, which was created with FastAPI (Python) for the backend and React.js for the frontend, allows users to register, access structured lectures, take quizzes, and get personalized suggestions based on their performance. Every user has a unique learning path according to the platform's adaptive architecture, which dynamically modifies the material level and complexity based on each learner's success. Additionally, an admin dashboard is provided for managing users, keeping an eye on analytics, and supervising course content.

To guarantee iterative development, quality control, and adaptability in fulfilling changing requirements, the project employs the Dynamic System Development Method (DSDM). The system's technological, operational, and financial sustainability is confirmed by feasibility studies, which are bolstered by scalable architecture and effective use of web technologies.

Testing phases—including unit, validation, and integration testing—demonstrated high accuracy, functionality, and performance reliability. The project successfully meets its objectives of delivering an intelligent, user-friendly, and adaptive coding education tool. Although some advanced features like real-time code execution, gamification, and mobile application support remain for future work, the system establishes a solid foundation for modern AI-driven e-learning platforms.

Overall, this project contributes significantly to the field of AI-based education technology, offering students a practical, engaging, and personalized approach to mastering programming fundamentals.

TABLE OF CONTENTS

CONTENTS.....PAGE

Approval.....i

Declaration.....ii

Acknowledgement.....iii

Executive Summary.....iv

List of Tables.....x

List of figure.....xi

CHAPTER

CHAPTER 1: Introduction..... 1

1.1 Introduction..... 1

CHAPTER 2: Initial Study.....2-3

2.1 Project Proposal.....2

2.2 Background of the project..... 3

2.3 Problem Area.....3

2.4 Possible Solution..... 3

CHAPTER 3: Literature Review.....4-7

3.1 Discussion on problem domain based on published articles.....4

3.2 Discussion on problem solutions based on published articles.....4

3.3 Comparison of leading solutions.....5

3.4 Recommended Approach.....

Chapter 4: Methodology.....8-11

4.1 What to use...../.....8

4.2 Why to use.....8
4.3 Section of Methodology.....8
4.4 Implementation Plans.....10
Chapter 5: Planning.....12-14
5.1 Project Plan.....12
5.1.1 Management Plan.....12
5.1.2 Resource Allocation.....13
5.1.3 Time Boxing.....14
Chapter 6: Feasibility.....16-17
6.1 All Possible types of feasibility.....16
6.1.1 Operational feasibility.....16
6.1.2 Technical feasibility.....16
6.1.3 Technology.....16
6.2 Cost Benefit Analysis.....16
6.3 DSDM Dynamic system Development Method.....17
Chapter 7: Foundation.....18-22
7.1 Some Potential Approaches.....18
7.1.1 Interview.....18
7.1.2 Observation.....18
7.2 Specific problem are identification and description.....18
7.3 Possible olution.....19
7.4 Overall Requirement list.....19
7.4.1 Functional Reurements.....19
7.4.1.1 User.....19
7.4.1.2 Admin.....20
7.4.2 Non-Functional Requrement.....20
7.4.2.1 Security.....19

7.4.2.2 Performance.....20
7.4.2.3 Availibility.....20
7.4.2.4 User Friendly.....21
7.5 Which technology to be implemented.....21
7.6 Recommendation and justifications.....22
Chapter 8: Exploration.....23-33
8.1 Use case.....23
8.2 Activity Diagram.....25
8.3 Requirement Catalogue.....26
8.4 Prioritized Requirement List (PRL).....28
8.5 Prototype of new system.....29
Chapter 9: Engineering.....34-36
9.1 Class diagram.....34
9.2 ER Diagram.....34
9.3 Sequence Diagram.....35
Chapter 10: Development.....37-40
10.1 Core Module Samples.....37
10.2 Probability problem break down.....39
10.3 Prioritization while developing.....40
Chapter 11: Testing.....41-44
11.1 Test Indroduction.....41
11.2 Test Plan.....41
11.2.1 Objective of the plan.....41
11.2.2 Scope.....42
11.2.3 Test Environment.....42
11.3 Test Plan Acceptance.....42
11.4 Unit Testing.....42

11.3 Validation Testing.....42
11.4 Integration Testing.....42
11.5 Test Cases.....43
Chapter 12: Implementation.....45-46
12.1 Training.....45
12.2 Big Bang Implementation.....45
12.3 Scaling.....45
12.3.1 Design of scaling.....45
12.3.2 Testing Performance.....46
12.4 Load balancing.....46
Chapter 13: Critical Appraisal and Evaluation.....47-49
13.1 Objective that could be met.....47
13.1.1 Success rate against each objective.....47
13.1.2 How much Better Could have Been Done.....47
13.1.3 Why it Could not be done.....47
13.1.4 Which objective have been missed.....48
13.1.5 why these objectives have missed.....48
13.1.6 What could have been done to complete those objective.....48
13.2 Objective totally not met.....49
13.2.1 Including software and documentation.....49
Chapter 14: Lessons Learned.....50-51
14.1 Pre-project.....52
14.2 Review.....5
14.3 Lessons Learned.....53
14.4 Problem faced.....54
14.5 Problems that are solutions.....54
Chapter 15: Conclusion.....52-53

15.1 Summary of the project.....52
15.2 Goal of the project.....52
15.3 Success Of the projects.....52
15.4 Documentations.....52
15.5 Value of the project.....53
15.6 My experience.....53

References.....54

LIST OF TABLES

Chepter	Table name	Page
3.4	Table 1: Module Description	6-9
5.1.1	Table 2: Management Planning	12-13
5.1.2	Table 3: Resource Allocation	13-14
5.1.3	Table 4: Time Boxing	14-15
6.1.2	Table 5: Recurrent for the application	16
6.1.3	Table 6: Recurrent Software for Application	16
6.2	Table 7: Cost Benefit	16-1-
8.4	Table 8: Prioritized requirement list	28
10.3	Table 9: Prioritization table	40
11	Table 10: Shows start end date	41
11.7	Table 11: Test Case	43-44
12.1	Table12: Training Procedure	45

LIST OF FIGURES

Chepter	Figure Name	Page
3.3	Figure 1: Sololearn	5
3.3	Figure 2: codeacdemy	5
3.3	Figure 3: cousesera	6
4.3	Figure 4: DSDM Prase	10
7.4.2.1	Figure 5: User login Interfaces	20
8.1	Figure 6: Use case Diagram	24
8.1	Figure 7: Use case Diagram for Admin Panel	25
8.2	Figure 8: Activity Diagram	26
8.5	Figure 9: Showing Assessment process	29
8.5	Figure 10: Profile Information for User Dashboard	29
8.5	Figure 11: Log in Interfaces for user dashboard	30
8.5	Figure 12: Showing user Login page	30
8.5	Figure 13: Admin Interfaces For Admin Dashboard	31
8.5	Figure 14: Admin Control Interfaces for Admin dashboard	31
8.5	Figure 15: Adding a Question for user	32
8.5	Figure 16: Exciting Offers for Users	32
8.5	Figure 17: Selected Plan chosen by User	33
8.5	Figure 18: Payment Method for User	33

9.1	**Figure 19: Cless Diagram For each Role**	**34**
9.2	**Figure 20: ER Diagram System Support**	**35**
9.3	**Figure 21: Sequence Diagram For User**	**36**
9.3	**Figure 22: Sequence Diagram For Admin**	**36**
10.1	**Figure 23: Configure code For Application**	**37**
10.1	**Figure 24: Database code for Application**	**37**
10.1	**Figure 25: Security code for Application**	**38**
10.1	**Figure 26: Configure code for Application**	**38**
10.1	**Figure 27: APP Test Code for Application**	**39**
10.1	**Figure 28:CSS Code for Application**	**39**

CHAPTER 1

Introduction

1.1 Introduction

Systems are applications that display the connections and exchanges between several programs. Computers' "System" site contains applications, programming links, and system administration tools. Although the word "system" might indicate different things depending on the context, the concept is essentially the same. The " Digital Storytelling Project" integrates several technologies to create a comprehensive basis. The various sections that make up this framework apply limitations to each system that is being examined. There are several systems in each module. To increase the instructiveness, flexibility, and customization of programming teaching, a web-based application known as the AI Coding Learning System has to be developed. Through lesson monitoring, assessments, and AI-powered adaptive learning, the system guides students through the foundations of C programming code step-by-step. Users may register, access the admin panel, enroll in classes, pass exams, and receive performance-based recommendations that evolve as they go.

The need for efficient and individualized programming instruction is rising as a result of technology's quick development. It might be challenging for novices to establish solid coding foundations since traditional learning approaches sometimes don't adjust to the speed and comprehension of individual students. The AI Coding Learning Platform is a web-based interactive app that integrates quizzes, lesson tracking, and AI-driven adaptive education in order to close this gap. In addition to letting students study coding fundamentals step-by-step, the system assesses their performance through tests and offers tailored recommendations depending on their development. The platform guarantees that any student may follow a personalized learning path appropriate for their ability level with its user-friendly interface, secure user administration, and intelligent content delivery.

CHAPTER 2

Initial Study

2.1 Project Proposal

Objectives

This project's primary goal is need develop an intuitive online tool that can called the AI Coding Learning System was need to be as created to improve the instructiveness, adaptability, and personalization of programming instruction. To walk students through various c programming coding fundamentals step-by-step, the system combines lesson monitoring, evaluations, and AI-powered adaptive learning. Users may sign up, admin panel, take classes, pass tests, and get suggestions based on their performance that change as they advance.

- To develop an AI-powered web application for learning coding in a step-by-step manner.
- To track learners' progress through lessons and assessments.
- To provide adaptive learning by suggesting new lessons or quizzes based on learner performance.
- To implement a secure user management system with login, registration, and profile tracking.
- To design an intuitive and interactive frontend interface for learners.
- To create a scalable backend with database integration for lessons, quizzes, and progress records.
- To enhance the overall learning experience by making it personalized, efficient, and engaging.

Benefits of the website:

To walk students through various c programming coding fundamentals step-by-step, the system combines lesson monitoring, evaluations, and AI-powered adaptive learning. Users may sign up, admin panel, take classes, pass tests, and get suggestions based on their performance that change as they advance.

2.2 Background of the Project

Since programming has grown more and more important in the digital age, knowing how to code has grown crucial for both professionals and students. However, textbooks and static tutorials—two common approaches to teaching programming—frequently fall short of meeting the unique demands of each student. A lack of enthusiasm and comprehension is the outcome of the content's failure to adjust to the speed of many newcomers. Learning systems driven by AI are being implemented in schools to address these issues. To close this gap, the AI Coding Learning Environment was created, providing an interactive learning environment where students may study classes, take tests, and get performance-

based adaptive advice. This project uses artificial intelligence and contemporary online technologies to improve the effectiveness, accessibility, and dynamic nature of coding instruction.

2.3 Problem Area

Regardless of each learner's unique skill level, traditional coding learning systems frequently offer the same set of courses and tests. This one-size-fits-all method fails to challenge expert learners and makes it hard for newcomers to keep up. Additionally, the majority of systems lack adaptive learning capabilities and efficient progress monitoring, which lowers engagement and delays skill development. For improved learning results, a clever, AI-driven system of learning that can monitor user performance and offer tailored information is therefore required.

2.4 Possible Solution

- To create a web application driven by AI that teaches coding step-by-step.
- To monitor students' development via instruction and evaluations.
- To offer adaptive learning by recommending new courses or tests in response to student performance.
- To put in place a safe system for managing users that tracks registration, login, and profile information.
- To create a user-friendly and dynamic front-end experience for students.
- To develop a database-integrated, scalable backend for lessons, tests, and progress logs.

CHAPTER 3

Literature Review

3.1 Discussion on problem domain based on published articles

According to research, by tailoring learning routes and offering prompt feedback, AI-driven adaptive systems can greatly enhance programming education (Alam et al., 2025; Gasevic et al., 2022). Research also highlights the significance of using adaptive learning models to provide engaging and sustainable digital education (Sharma & Singh, 2023; Liang et al., 2025). However, there have been worries expressed over an excessive dependence on generative AI, which might impair essential coding abilities (Akter et al., 2024; Zhou et al., 2024). Fairness, prejudice, and transparency are among the ethical and privacy concerns that are emphasized as being crucial to based on artificial intelligence education (Han et al., 2024; Xu et al., 2025).

3.2 Discussion on problem solutions based on published articles

Publications on AI-based learning systems point out a number of issues and solutions that are directly related to the goals of this project. The variance in learners' pace and comprehension, which makes it challenging to create a teaching model that works for everyone, is a significant issue that has been noted in earlier research. Researchers stress the value of systems of education that modify information based on individual achievement in order to overcome this. This method is used by our system, which keeps track of user development and quiz results before suggesting appropriate classes or tests depending on performance levels.

3.3 Comparison of leading solutions

SoloLearn:

Overview: A popular mobile-first programming training site, SoloLearn provides classes in Python, Java, JavaScript, and other languages. It offers a built-in program editor for in-the-moment practice, using AI-based customization to suggest courses, and boosts engagement with gamification elements like leaderboards, medals, and XP points. Students

may participate in programming problems and exchange solutions when there is a strong **sense of community.**

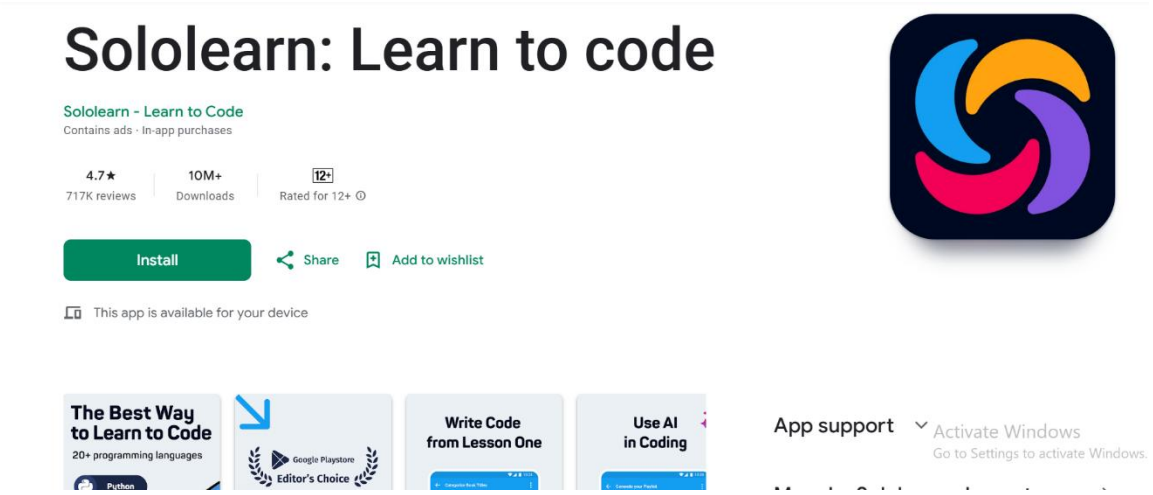

Figure 1: Sololrearn as a reference

Code academy:

Overview: One well-known online coding resource that emphasizes interactive, practical learning is Codecademy. To aid students in understanding programming principles, it offers real-time coding environments from within the browser with immediate feedback. Data-driven suggestions are used by Codecademy to lead students through skill tracks and organized career pathways.

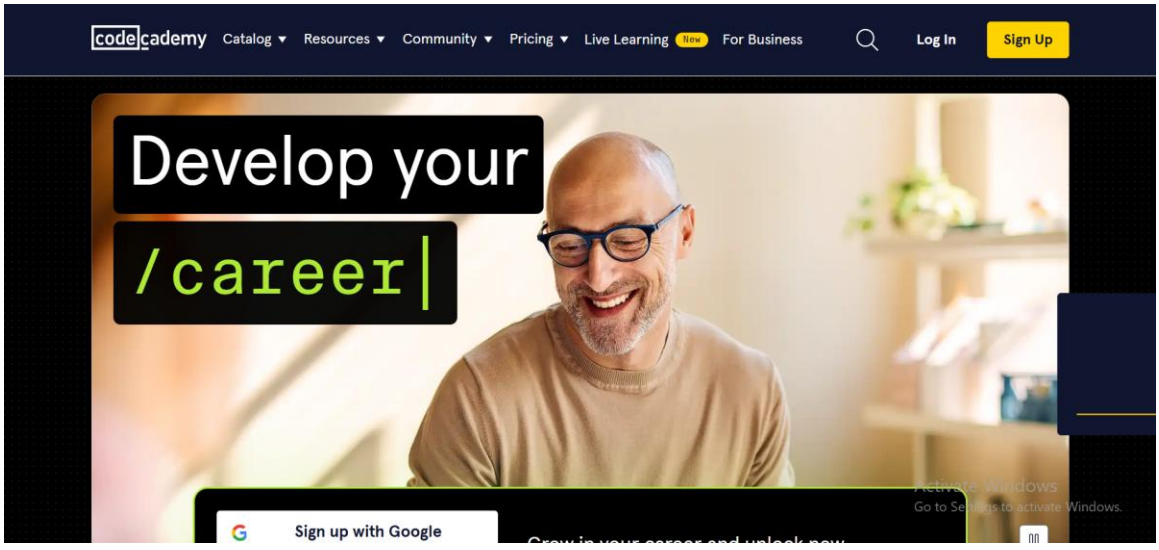

Figure 2: Codeacademy as a reference

Coursera:

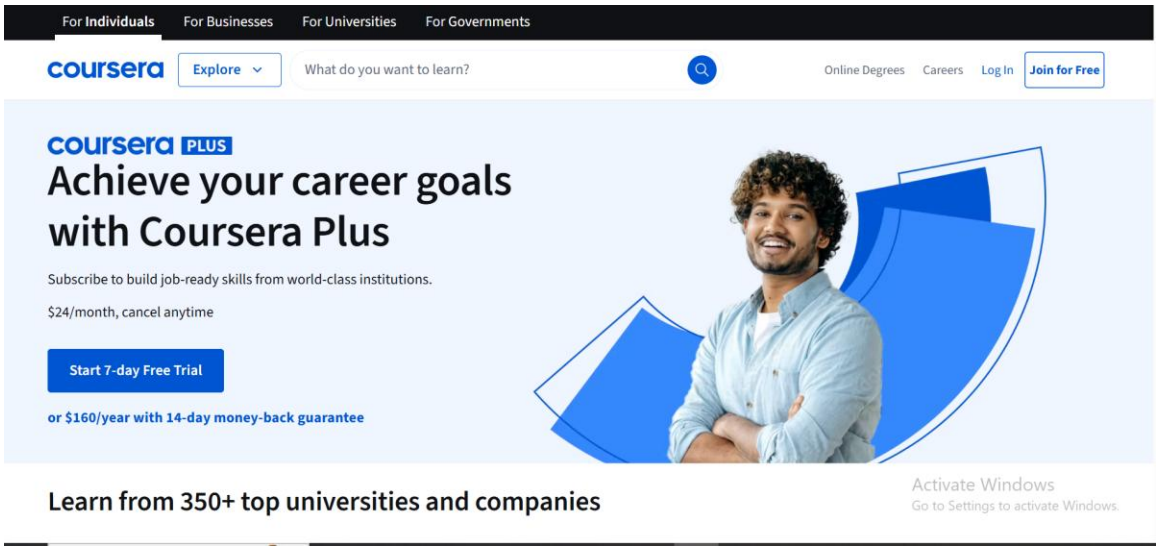

Figure 3: Coursera as a reference

3.4 Recommended Approach

Table 1: Modules descriptions

Actuators	Functions
User	• Login. • Registration • Dashboard for profile statistical. • Take Assessment of C programing quiz. • After getting 70% score of quiz then level Up for assessment (total 10 level) • Subscription Management: View current plan, upgrade/downgrade subscription • Payment Processing: Handle payment for Gold/Premium plans • Daily Limit Tracking: Monitor question attempts remaining • Usage Analytics: View daily/monthly usage statistics • Logout.

Admin	• Login. • Overview (All users, active users, assessment taken etc.) • Users(Show all users and activate/Deactivate users) • Content(Set, modify all questions among 10 levels) • Analytics • Subscription Overview: Monitor all user subscriptions and revenue • Payment Management: Handle subscription renewals, cancellations, refunds • Usage Reports: Generate reports on user engagement by subscription tier • Pricing Management: Modify subscription prices and features • • Logout

CHAPTER 4

Methodology

4.1 What to Use

The main objective of this project is to increase the instructiveness, flexibility, and customization of programming training by creating an easy-to-use online tool known as the AI Coding Learning System. Through lesson monitoring, assessments, and AI-powered adaptive learning, the system guides students through the foundations of C programming code step-by-step. Users may register, access the admin panel, enroll in classes, pass exams, and receive performance-based recommendations that evolve as they go. The SDLC life cycle methodology, a comprehensive framework for design, development, and other tasks, is recognized as the right approach. I understand a number of SDLC model kinds. The waterfall, Agile, Iterative, Adaptive System Construction, Big Bang, and Spiral models are examples of software development paradigms. For the purpose of developing and implementing the coding exercise platform, each model offers an environment. The particular needs of the SDLC model will mold the veterinary development field to provide an effective development process that aids in It provides real-time, browser-based coding environments with instant feedback to help students grasp programming concepts.

4.2 Why to use

The initial stage in the construction process is to determine the system architecture. Part of this was figuring out the parts and how they interacted. Every software project needs to adhere to the agile methodology. I am aware with a number of terms that are used to describe agile approaches, including Kanban, feature-driven development, scrum, quartz, and flexible system development methodology. However, I used the DSDM method to accomplish my goal. There are several reasons why the DSDM approach is beneficial. The dynamic system design method, which allows for flexibility in changing needs, is utilized for continual improvement. This tactic works well when prompt delivery is necessary. Using an AI-powered web application to develop online coding quizzes has several benefits, such as lower costs, better accuracy, and insights based on data.

4.3 Section of methodology

Pre-Project Phase:

- **Feasibility Study**: This study evaluates the project's functional needs, financial viability, and infrastructure. It entails assessing the project's possible expenses, advantages, and risks.
- **Conditions Collecting**: The program's demands have been gathered and documented. You must first understand the client's expectations, limitations, and business demands in order to define the project's scope.
- **Planning:** Make a strategic plan that outlines the objectives, schedule, materials, and deliverables of the project. Establishing roles and duties, identifying the project's stakeholders, and formulating a partnership and risk management plan are all essential.

Project Lifecycle Phase:

- **Design:** Using the requirements gathered, a software design is created in this step. To name a few, it includes a broad variety of design activities, such as database, architectural, and user interface design.
- **Development:** Software is coded according to design specifications throughout the development phase. To construct a working software product, the developer builds the source code, performs unit tests, and puts the pieces together.
- **Testing**: This procedure guarantees the functioning and quality of software. It encompasses a range of testing methodologies, including as system, integration, user validity, and unit testing.
- **Implementation:** Following approval and testing, the software is put into use. The proper environment must be used for the installation, setup, and configuration of the software.

Post-Project Phase:

- **Maintenance:** The software moves into the maintenance stage following deployment. To ensure that the software keeps working and satisfies changing needs, this phase involves ongoing maintenance, bug fixes, and updates.

- **Evaluating:** To determine if a project is successful, actual results must be compared to intended goals. It helps pinpoint areas that require improvement and lessons learned for next initiatives.
- **Closure:** The project has been formally finished. It entails finishing the project's documentation, keeping an eye on its artifacts, and carrying out an evaluation of the project.

With the help of these components, software development projects may be managed methodically from early planning to after the deployment support, which helps to achieve good results.

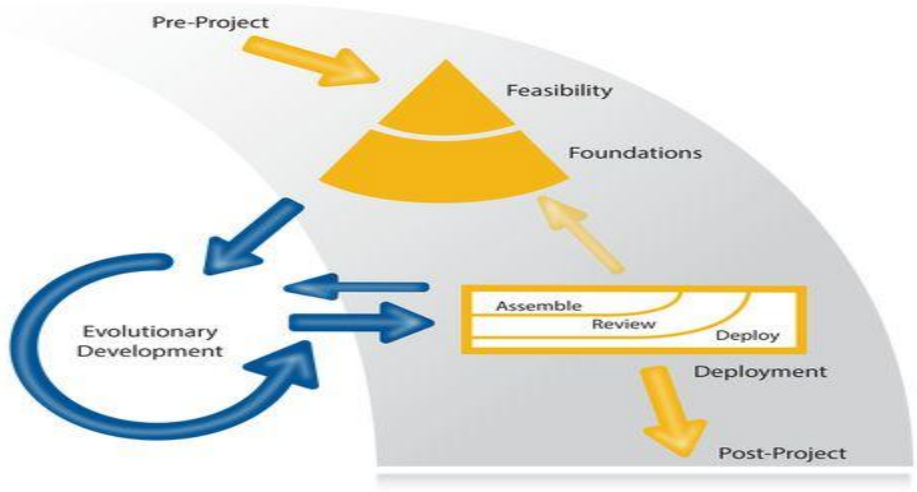

Figure 4: DSDM phase

4.4 Implementations plans

The finished application is now available for public usage at this stage of the project. As soon as an issue is found and fixed, the new system has to be activated. You select your release requirements, methods, and settings here. The updated system is tested and put into use if everything goes according to plan. Data gathering must take place following the completion of all subsequent procedures in order to ensure accuracy. To make the task easier to do, I broke it up into its most crucial components. I have to follow these guidelines to make sure my work is done correctly.

Feature:

- To create a web application driven by AI that teaches coding step-by-step.
- To monitor students' development via instruction and evaluations.
- To offer adaptive learning by recommending new courses or tests in response to student performance.
- To put in place a safe system for managing users that tracks registration, login, and profile information.
- To create a user-friendly and dynamic front-end experience for students.
- To develop a database-integrated, scalable backend for lessons, tests, and progress logs.

CHAPTER 5

Planning

5.1 Project Plan

To increase the instructiveness, flexibility, and customization of programming teaching, a web-based application known as the AI Coding Learning System has to be developed. Through lesson monitoring, assessments, and AI-powered adaptive learning, the system guides students through the foundations of C programming code step-by-step. Users may register, access the admin panel, enroll in classes, pass exams, and receive performance-based recommendations that evolve as they go. The backend, built with a database with relations and a Fast API, handles user data, quiz generation, and adaptive logic, while the frontend, built with React, provides learners with an intuitive interface. Every endeavor has to have its potential, budget, schedule, risk management, and connection server protocols described before it can start. Planning is essential before starting a project to reduce risks that might compromise the developer's ability to complete it. Project planning includes establishing goals and objectives, controlling risks, adhering to deadlines, and other duties. Software project plans frequently contain time boxes, which are a crucial tool for project planning.

5.1.1 Management plan

Describe the roles and responsibilities of the project team and the operation of the project management procedure. Create channels for reporting and communication to make sure the collaboration is successful. Establish the issue-resolution strategy's decision-making and escalation stages.

Table 2: Management Planning

No	Task Name	Duration	Start Date	End Date
1	Introduction	5	01-05-2025	05-05-2025
2	Initial Study	4	06-05-2025	09-05-2025
3	Literature Review	4	10-05-2025	13-05-2025
4	Methodology	3	14-05-2025	16-05-2025

5	Planning	10	17-05-2025	26-05-2025
6	Feasibility	15	27-06-2025	10-06-2025
7	Foundation	5	11-06-2025	15-06-2025
8	Exploration	14	16-06-2025	29-06-2025
9	Engineering	30	30-06-2025	29-07-2025
10	Deployment	18	30-07-2025	16-08-2025
11	Testing	10	17-08-2025	26-08-2025
12	Implementation	5	27-08-2025	31-08-2025
13	Critical Appraisal and Evaluation	4	01-09-2025	04-09-2025
14	Lessons Learning	3	05-09-2025	07-09-2025
15	Conclusion	1	08-09-2025	08-09-2025
	Total =	131 days		

5.1.2 Resource Allocation

List all of the project's resources, such as personnel, tools, and software. Ascertain the appropriate distribution of resources based on the workload and schedule of the project. Assign team members responsibilities and duties while making sure they possess the necessary abilities.

Table 3: Resource Allocation

No	Task Name	Duration	Resource
1	Introduction	5	End User
2	Initial Study	4	Analyst
3	Literature Review	4	Analyst
4	Methodology	3	Analyst
5	Planning	10	Analyst, Designer, Developer
6	Feasibility	15	Analyst
7	Foundation	5	Designer
8	Exploration	14	Designer, Developer

9	Engineering	30	Developer
10	Deployment	18	Analyst, Developer
11	Testing	10	Analyst, Developer, Tester, Users
12	Implementation	5	Analyst, Developer
13	Critical Appraisal and Evaluation	4	Analyst, Tester and Developer
14	Lessons Learning	3	Analyst, Users
15	Conclusion	1	Analyst
	Total =	131days	

5.1.3 Time Boxing

Break the project up into numerous stages or iterations to facilitate development and testing. Establish the duration of each time box and the items and activities that need to be completed throughout each iteration. For every time frame, establish clear objectives and available resources.

Table 4: Time Boxing

Time -Box	Task Name	Duration	Resource
TB1	Introduction	5	End Users, Analyst
	Initial Study	4	Analyst
	Literature Review	4	Analyst
TB2	Methodology	3	Analyst
	Planning	10	Analyst, Designer, Developer
	Feasibility	15	Analyst
TB3	Foundation	05	Designer
TB4	Exploration	14	Designer, Developer
	Engineering	30	Developer

TB5	Deployment	18	Analyst, Developer
	Testing	10	Analyst, Developer, Tester, Users
TB6	Implementation	05	Analyst, Developer
TB7	Critical Appraisal and Evaluation	04	Analyst, Tester and Developer
	Lessons Learning	03	Analyst, Users
TB8	Conclusion	01	Analyst
	Total =	131 days	

CHAPTER 6

Feasibility

6.1 All possible types of feasibility

6.1.1 Operational feasibility

A feasibility study examines the likelihood that all necessary concerns—including organizational, legal, financial, and engineering ones—will be resolved in order to successfully finish a project. The degree to which a system matures, makes use of the scope established during opportunity definition, and satisfies the requirements established during the project or necessity analysis phase of system construction is known as operational practicability. We developed a web application that would enable us to gradually identify coding assessment task systems using AI models. A web application that serves as an AI-based quiz programming dashboard system is at the heart of the proposed solution.

6.1.2 Technical feasibility

Hardware	Software
Dell Laptop, Wi-Fi, Router, Cable, Android Phone	Android Studio, Google Chrome Browser, Windows, MS Word, VS code

Table 5: Recrements for the application

6.1.3 Technology

Client slide	Back end side	Database	Other Tools
React.js, HTML, CSS, JavaScript	FastAPI (Python)	SQLite/PostgreSQL/MySQL	Docker, Git, REST API

Table 6: Recurrent Software for Application

6.2 Cost Benefit Analysis

Project managers utilize cost-benefit analysis to evaluate the benefits and drawbacks of different project routes, including relationships, activities, customer needs, and investments. The most economical method to accomplish my goal is determined via a cost-benefit analysis.

Project Name: AI Coding Learning System

Table 7: Cost Benefit

Equipment	1st Year	2nd Year	3rd Year	4th Year	Total
Web Based Application	20000				20000
Domain Hosting		10000	10000	10000	30000
Software	1000				1000
Internet	2000	2000	2000	2000	8000
Training	5000				5000
Development		5000			5000
Payment Gateway Fees	25000	3% each transaction			25000
Maintenance	10000	10000	10000	10000	4000
Total					98,000 BDT.

6.3 DSDM Dynamic System Development Method (DSDM)

DSDM, short for Dynamic Systems Development Method, is neither a method nor technology for creating applications; rather, it is an organizational framework for agile project and computer development monitoring. It focuses on frequent software delivery, iterative development methods, and interaction between development groups and business stakeholders. The employment of particular tools or technologies that such as React js, JavaScript, HTML, and CSS with Python, is essential for DSDM. In DSDM projects, these well-liked web development technologies have obvious uses.

CHAPTER 7

Foundation

7.1 Some potential approaches

7.1.1 Interview

Our effort to create an online platform named the AI Coding Learning System, aims to improve the efficiency and individualization of programming instruction. A frontend with React.js for the consumer's interface and a backend with FastAPI for the database, tests, and flexible AI logic make up its two primary components. Additionally, user administration with secure authentication is part of the project. All things considered, the system seeks to offer an engaging, AI-powered educational experience. It may be expanded in the future with gamification, a mobile app, and increasingly sophisticated AI algorithms.

7.1.2 Observation

When compared to a predefined lesson path, the adaptive learning technique dramatically increased user engagement during the AI Coding Learning System's development and testing. While lesser learners were given shorter courses and revision tests, those with higher quiz results were able to graduate to more complex topics more quickly. A more engaging and dynamic atmosphere was produced by combining customized recommendations with real-time progress tracking. The rule-based AI reasoning that is now in use, however, may be further enhanced using models developed with deep learning to produce more intelligent recommendations.

7.2 Specific problem are identification and description

The courses and examinations offered by traditional coding learning systems are often the same, regardless of the individual ability level of each learner. This one-size-fits-all approach makes it difficult for novices to keep up and falls short of challenging seasoned students. Adaptive learning and effective progress tracking are also absent from most systems, which reduces user interest and postpones skill improvement. Consequently, a smart, AI-powered learning system that can track user performance and provide personalized information is needed for better learning outcomes.

7.3 Possible solution

- To create a web application driven by AI that teaches coding step-by-step.
- To monitor students' development via instruction and evaluations.
- To offer adaptive learning by recommending new courses or tests in response to student performance.
- To put in place a safe system for managing users that tracks registration, login, and profile information.
- To create a user-friendly and dynamic front-end experience for students.
- To develop a database-integrated, scalable backend for lessons, tests, and progress logs.

7.4 Overall Requirement List

- Functional Requirements
- Non-Functional Requirements.

7.4.1 Functional Requirements

7.4.1.1 User

- Login.
- Registration
- Forget password.
- Dashboard for profile statistical.
- Take Assessment of C programing quiz.
- After getting 70% score of quiz then level Up for assessment (total 10 level)
- Choose subscription plan during registration (Free/Gold/Premium)
- Process payment for Gold/Premium subscriptions
- Track daily question limit based on subscription tier
- View subscription status and remaining daily questions
- Upgrade/downgrade subscription plan
- Access payment history and invoices
- Logout.

7.4.1.2 Admin

- Login.
- Overview (All users, active users, assessment taken etc.)
- Users(Show all users and activate/Deactivate users)

- Content(Set, modify all questions among 10 levels)
- Analytics
- Monitor subscription analytics and revenue dashboard
- Manage user subscriptions (view, modify, cancel)
- Generate payment and usage reports
- Configure subscription pricing and features
- Handle payment disputes and refunds
- Logout

7.4.2 Non-Functional Requirements

7.4.2.1 Security

Each user of the system has an account, and solely those who have been given authorization and a password may access it. We use both JavaScript to encrypt the credentials.

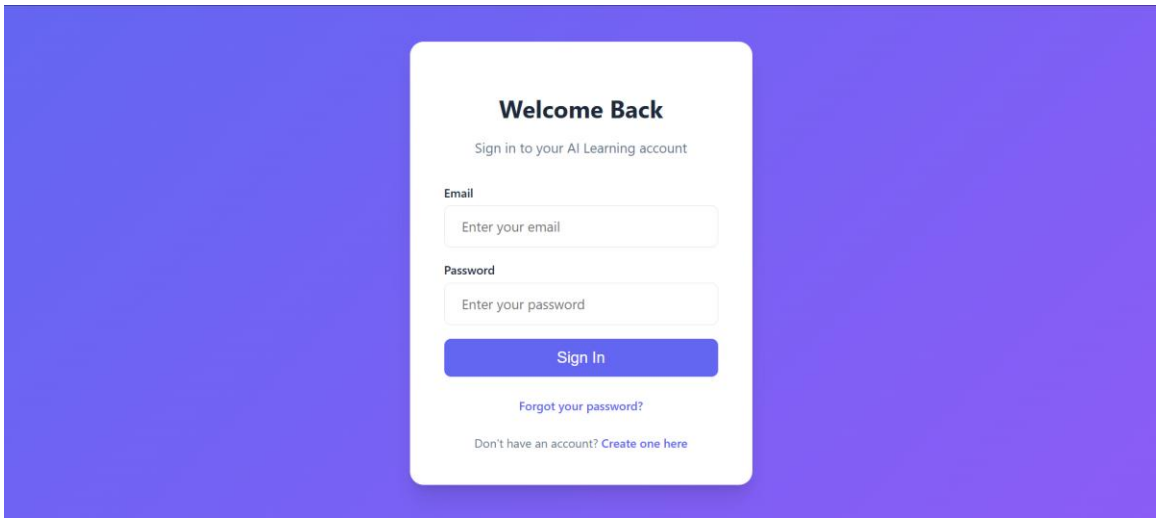

Figure 5: User Login Interfaces

7.4.2.2 Performance

Records are easy to update and maintain.

7.4.2.3 Availability

To use the system from any time and from any location, users only need a PC and an Internet connection. The system works with a number of web browsers, such as Chrome, Mozilla, Opera, and Internet Explorer.

7.4.2.4 User Friendly

The technology offers an interesting user interface and is easy to use.

- When several users use the website simultaneously, there shouldn't be any noticeable lag or outages.
- The website has to be able to manage big data sets.
- Websites should have simple, easy-to-use designs.
- It should be simple to add new features and functionalities to the website without having to completely redo it.
- To address bugs and post-deployment problems, regular website maintenance is essential.

7.5 Which technology to be implemented

My project is a fully web-based software. ReactJs, CSS, JavaScript, SQLlite, and python are the tools I'm using to construct my project.

HTML: Web pages can be created using markup languages like HTML. HTML files may be seen and understood by web browsers. Every website is built on HTML components. enables you to create engaging content by permitting the usage of HTML elements and visuals. In [2], it may also produce chapters, lists, quotes, titles, and links.

CSS: To further personalize the material on our webpage, we may utilize CSS to alter the fonts, colors, and layouts. This improves the cohesion and aesthetic attractiveness of our website. The website seems friendlier as a consequence. Inside [2]

JavaScript: JavaScript is currently one of the most popular programming languages. We also employ JavaScript as a web development language. A layer of common web technologies is developed as a result. In [3]

Bootstrap 4: Bootstrap is currently at version 4.0. You may make mobile-friendly websites with all the required HTML, CSS, and JavaScript elements by using Bootstrap 4. I created user accounts on my website as a result. [4]

ReactJS: React.js is a powerful JavaScript library designed especially for making graphical user interfaces (UIs) for web applications. React.js is widely utilized because to its efficacy, performance improvements, component adaptability, and developer-friendly features. It becomes a more appealing choice for developing innovative, dynamic web-based apps and user interfaces as new features and improvements are implemented. [5]

Python: Python is a popular high-level, interpreted programming language that is simple to learn and comprehend, making it an excellent choice for both beginners and experienced programmers. It supports several different programming paradigms, including imperative, functional, and object-oriented programming. Python is widely used in a variety of applications, including scientific computing, data science, web development, automation, machine learning, and artificial intelligence. Developers may efficiently execute difficult tasks using its extensive standard library and third-party packages such as NumPy, Pandas, and TensorFlow. Python is one of the most popular and versatile programming languages accessible today, thanks to its large community and cross-platform compatibility.

7.6 Recommendation and justifications

We advise strengthening the adaptive learning module with cutting-edge AI for more intelligent, tailored recommendations and adding a real-time code compiler to make the entire system more engaging. Creating a mobile application will improve accessibility, and incorporating gamification elements like leaderboards and badges will promote student motivation. Lastly, scalability and improved performance are guaranteed when the system is deployed on the cloud. These enhancements will increase the AI Coding Learning System's usability, effectiveness, and engagement.

CHAPTER 8

Exploration

8.1 Use case

This section looks at both functional and non-functional needs using use-case data and graphics.

User:

Following the system, the user can carry out the following tasks:

- Login.
- Registration
- Forget password
- Dashboard for profile statistical.
- Take Assessment of C programing quiz.
- After getting 70% score of quiz then level Up for assessment (total 10 level) other level doesn't up.
- Select Subscription Plan
- Process Payment
- View Subscription Status
- Check Daily Limit
- Upgrade Subscription
- Logout.

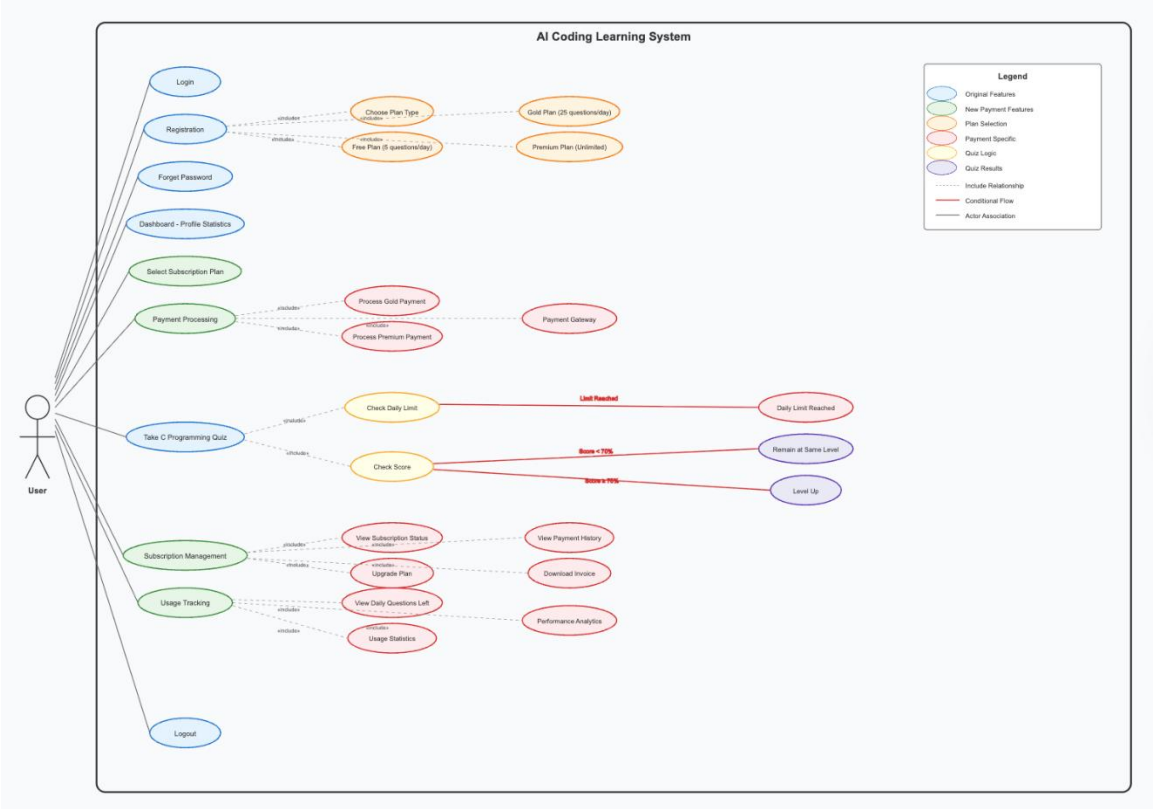

Figure 6: Use case Diagram

Admin:

- Login.
- Overview (All users, active users, assessment taken etc.)
- Users(Show all users and activate/Deactivate users)
- Content(Set, modify all questions among 10 levels)
- Analytics
- Monitor Revenue Dashboard
- Manage User Subscriptions
- Generate Payment Reports
- Configure Pricing Plans
- Handle Payment Issues
- Logout

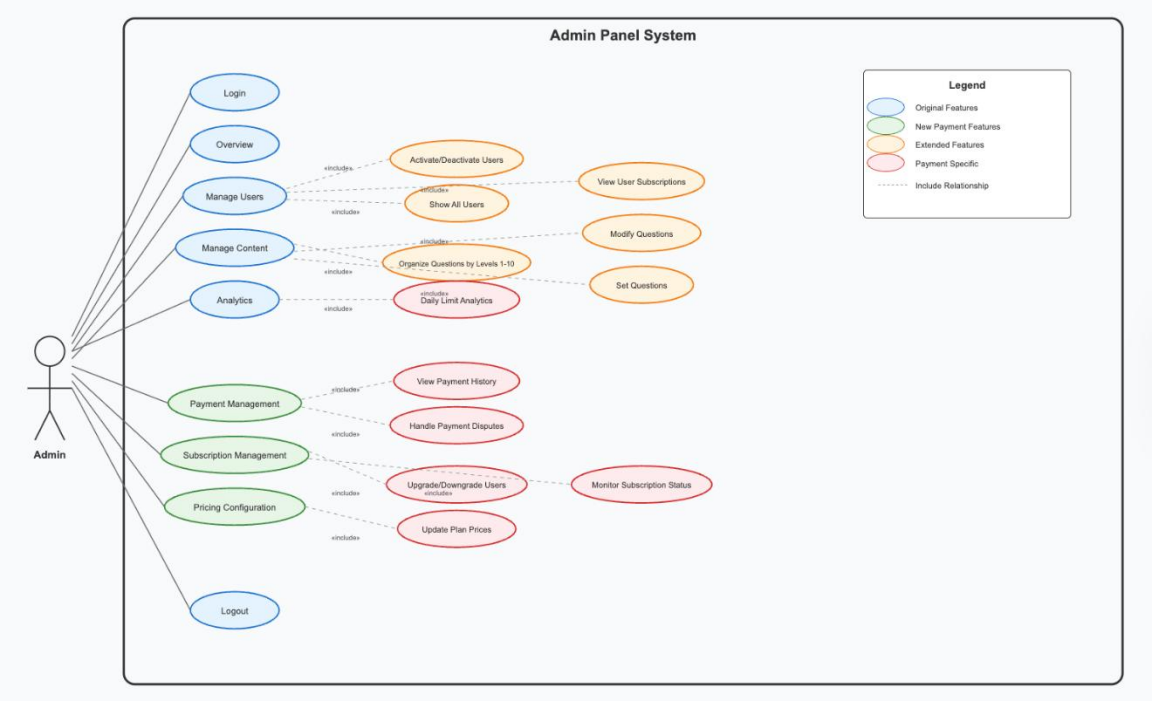

Figure 7: Use case Diagram for Admin Panel

8.2 Activity diagram

Describe the dynamic operation of the system. It shows how one task is related to another, much like a flow chart. The activity might help you understand how the system works. As a result, control is distributed across many operations. Each module's overall activity diagram is displayed below:

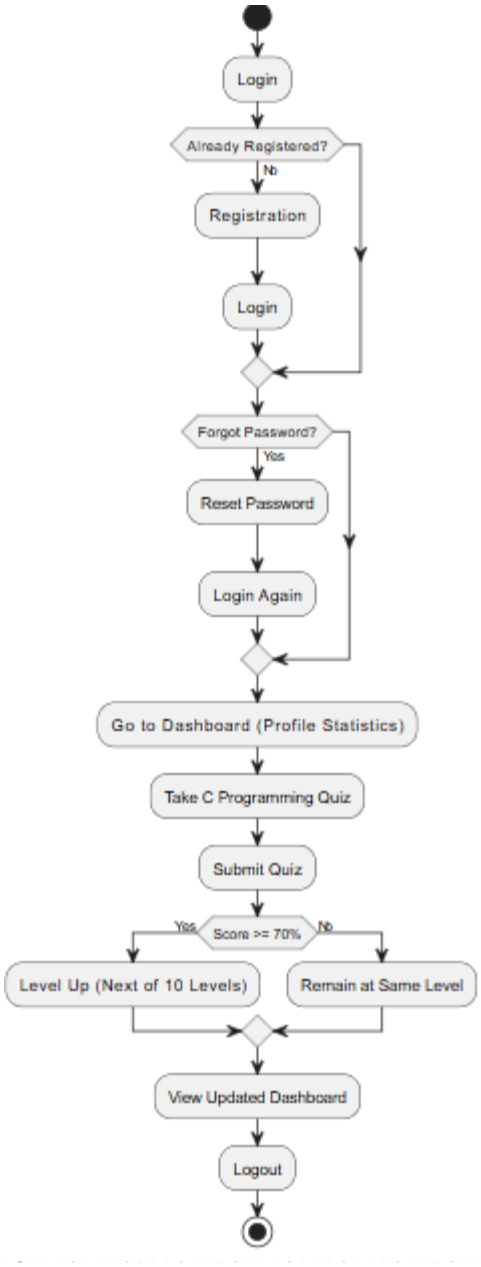

Figure 8: Activity Diagram.

8.3 Requirement catalogue

Functional requirements:

User:

- FR1: Login users.
- FR2: User Registration
- FR3: Forget password

- FR4: Dashboard for profile statistical.
- FR5: Take Assessment of C programing quiz.
- FR6: After getting 70% score of quiz then level Up for assessment (total 10 level) other level doesn't up.
- FR7: Choose subscription plan during registration (Free/Gold/Premium)
- FR8: Process payment for Gold/Premium subscriptions
- FR9: Track daily question limit based on subscription tier
- FR10: View subscription status and remaining daily questions
- FR11: Upgrade/downgrade subscription plan
- FR12: Access payment history and invoices
- FR13: Logout.

Admin:

- FR1: Admin Login.
- FR2: Overview (All users, active users, assessment taken etc.)
- FR3: Users(Show all users and activate/Deactivate users)
- FR4: Content(Set, modify all questions among 10 levels)
- FR5: Analytics
- FR6: Monitor subscription analytics and revenue dashboard
- FR6: Manage user subscriptions (view, modify, cancel)
- FR8: Generate payment and usage reports
- FR9: Configure subscription pricing and features
- FR10: Handle payment disputes and refunds
- FR11: Admin Logout

Non-Functional Requirements:

- NFR1: Maintaining and updating records is simple.
- NFR2 enables users to utilize a PC with an Internet connection to access the system from any location. Numerous web browsers are supported by the system, including as Mozilla, Chrome, Opera, and Microsoft's Internet Explorer.
- NFR3: The technology offers an interesting user interface and is easy to use.

User Interface Requirements:

- UIR1: Interface is straightforward to use, with simple navigation that makes accessing features and capabilities simple.
- UIR2: Adapts to varying screen and device sizes with responsive design.
- UIR3: To aid users in comprehending and utilizing the system, icons provide visual cues.

Security and Privacy Requirements:

• SR1: Use secure authorization and authentication processes to safeguard user data.

8.4 Prioritized Requirement List (PRL)

Table 8: Prioritized requirement list

Requirement ID	Requirement Description	Priority	Dependencies	Status	Validation Criteria
RQ1	Admin or User Login functions	High		Pass	Login successfully.
RQ2	User registration system	High	RQ1	Pass	Successfully registered.
RQ3	View user / admin dashboard	High	RQ1, RQ2	Pass	Successfully done
RQ4	User participates quiz for level up	High	RQ1, RQ2	Pass	Successfully done.
RQ5	Show there performance result	High	RQ1, RQ2, RQ4	Pass	Successfully
RQ6	Admin update questions and users details	High	RQ1	Pass	Successfully
RQ7	Forget password	Medium	RQ1	Pass	Successfully

8.5 Prototype of new system

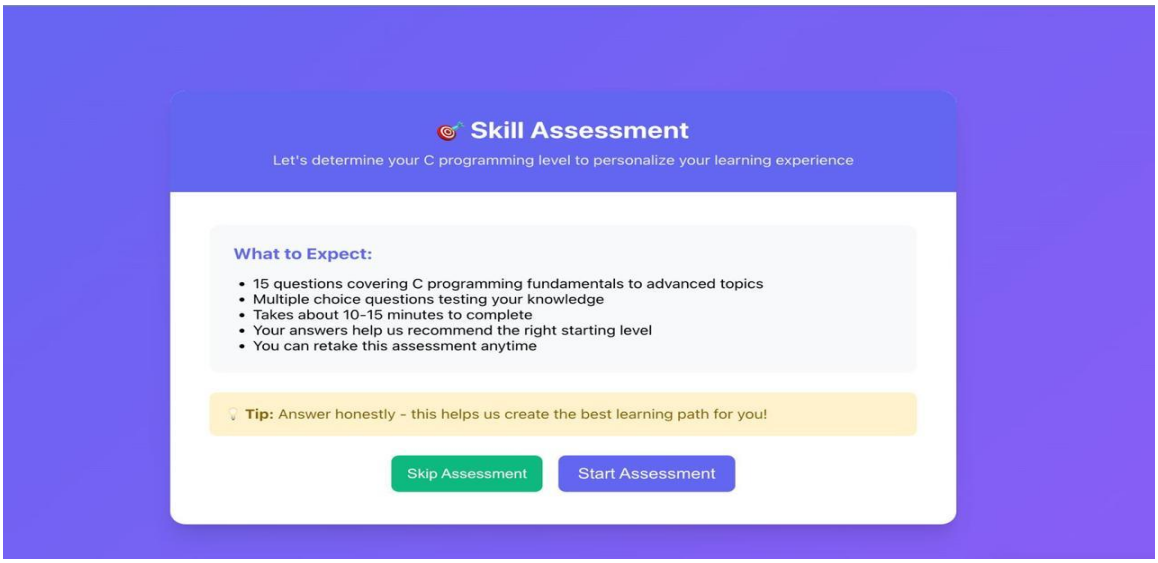

Figure 9: Showing Skill Assessment Process

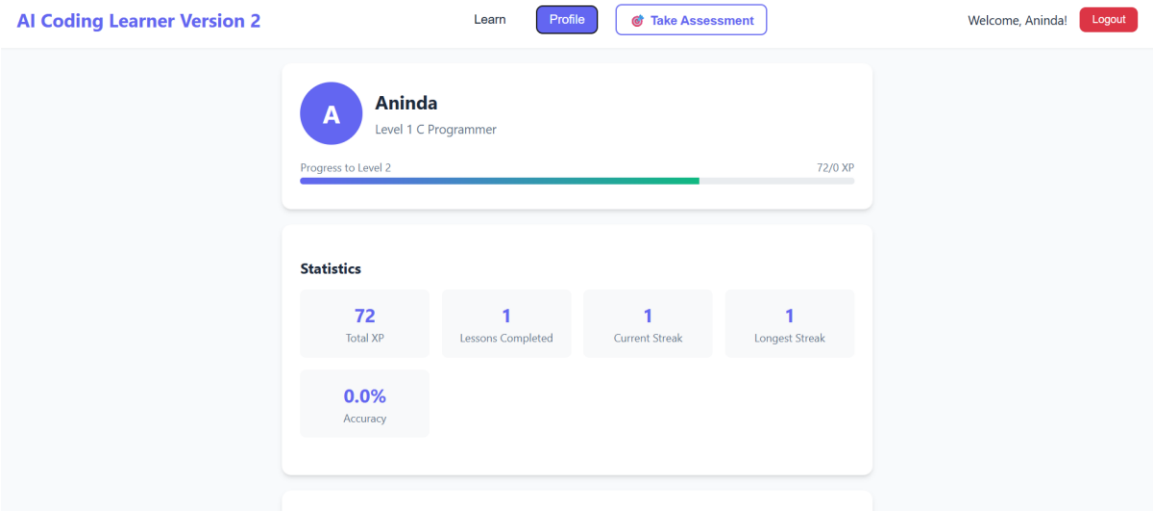

Figure10: Profile Information for user dashboard

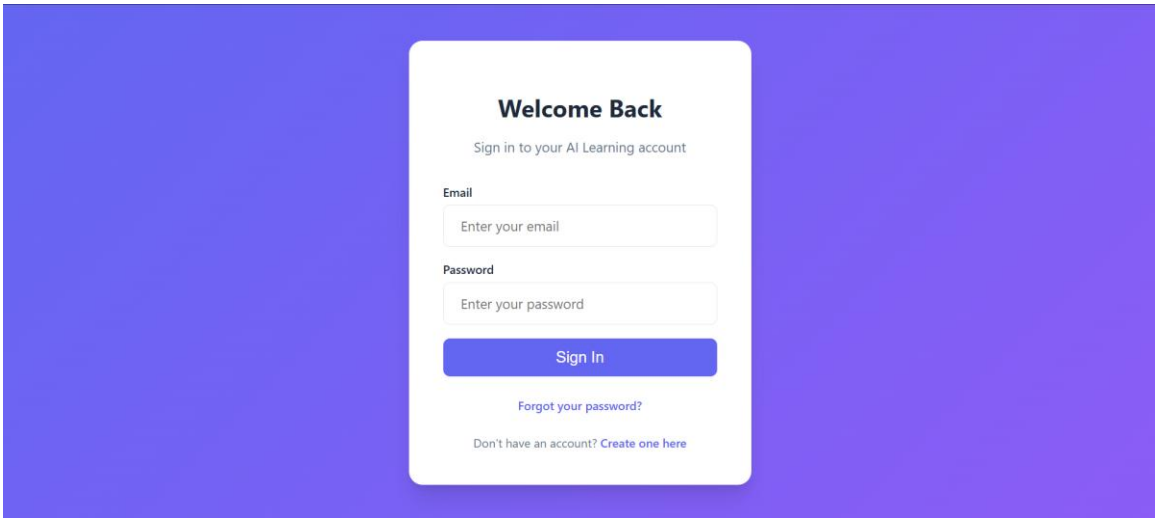

Figure 11: Log in Interfaces for user dashboard

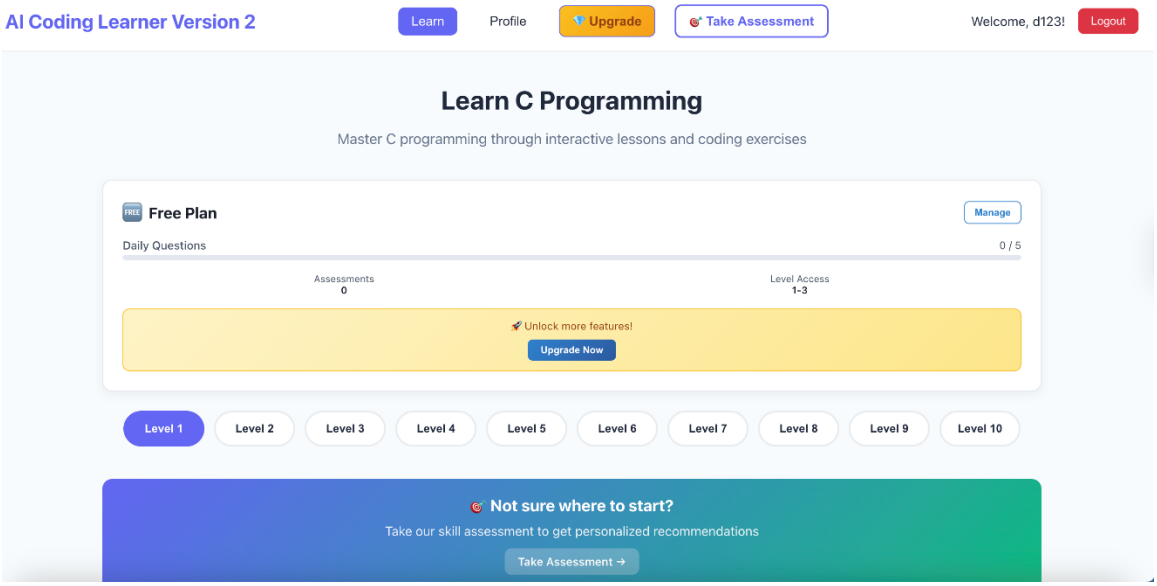

Figure 12: Showing user Login page

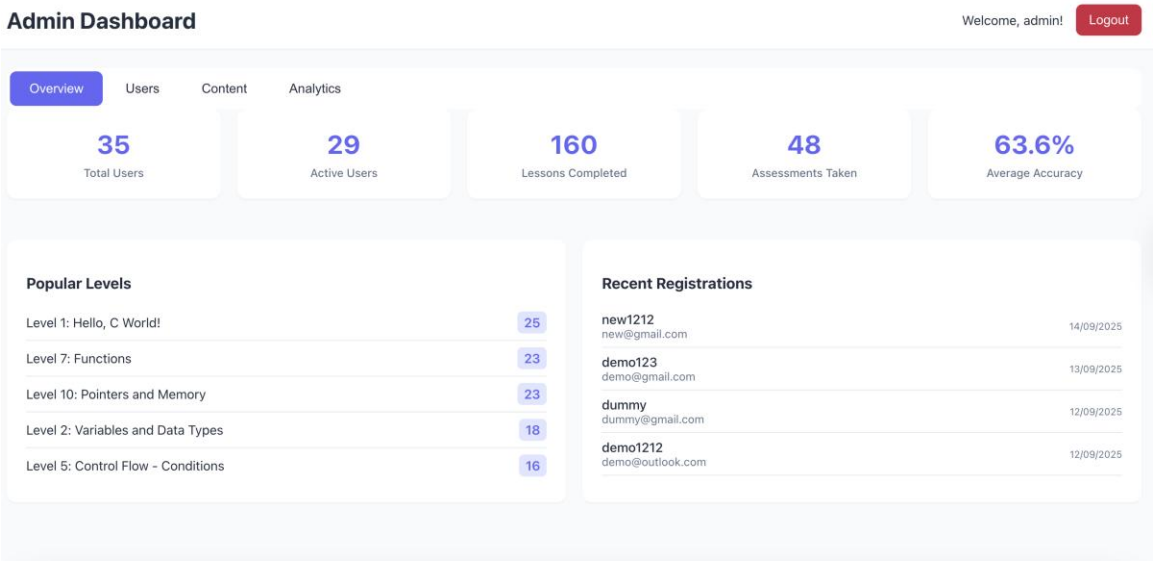

Figure 13: Admin Interface for Admin dashboard

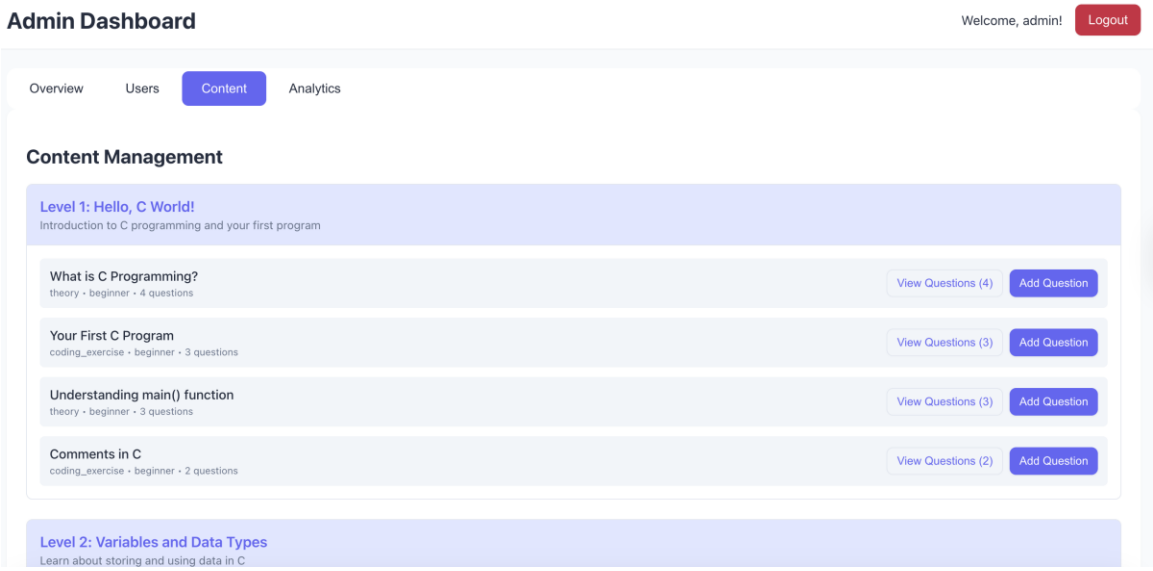

Figure 14: Admin Control Interfaces for Admin dashboard

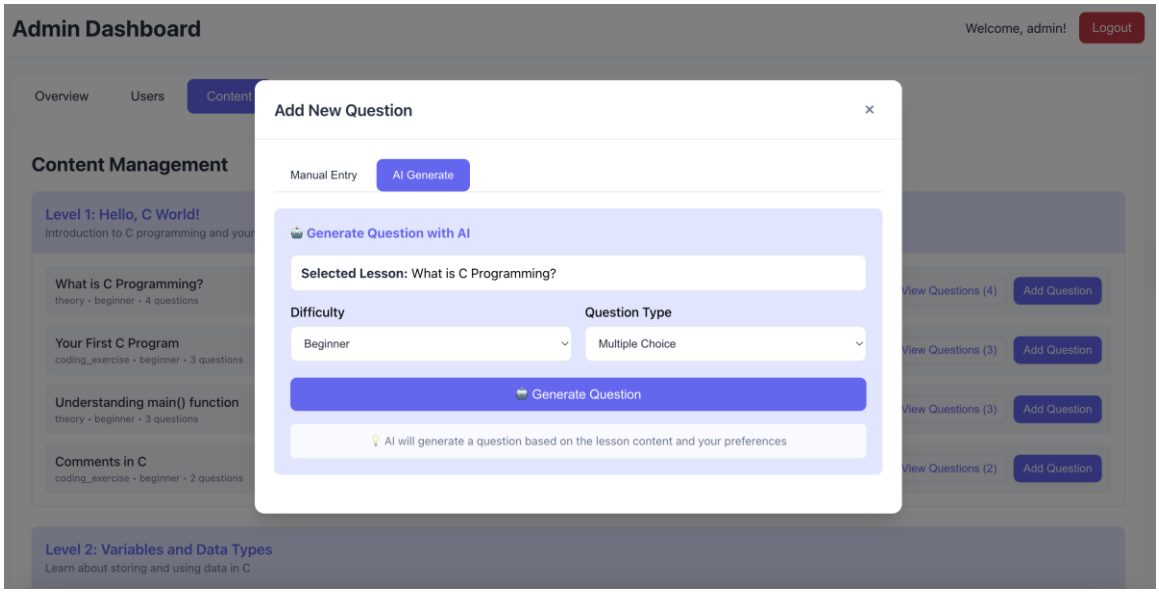

Figure 15: Adding a Question for user

Figure 16: Exciting Offers for Users

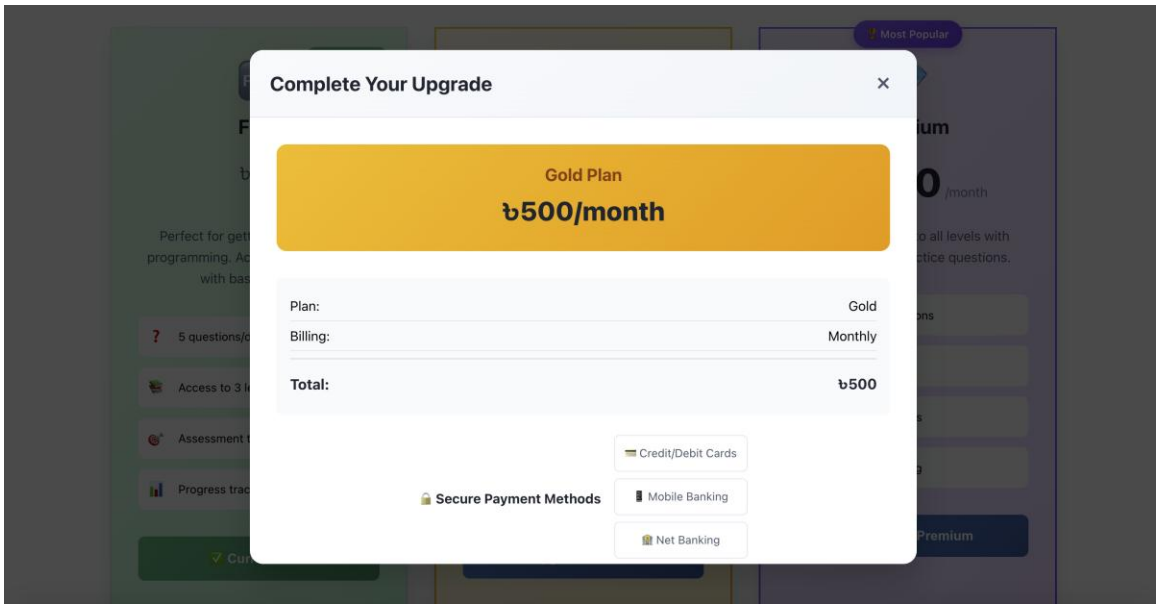

Figure 17: Selected Plan chosen by User

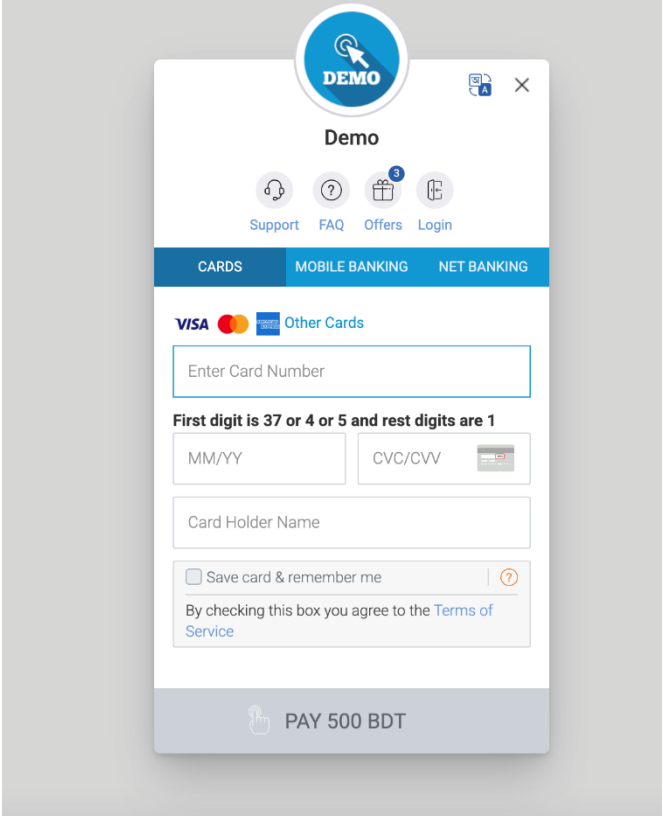

Figure 18: Payment Method for User

CHAPTER 9

Engineering

9.1 Class Diagram

A class was made in order to show the original content for interclass linkages. In this instance, the class either functions as a distinct entity within the program or as a single programming specification that specifies the variables and behaviors of an object.

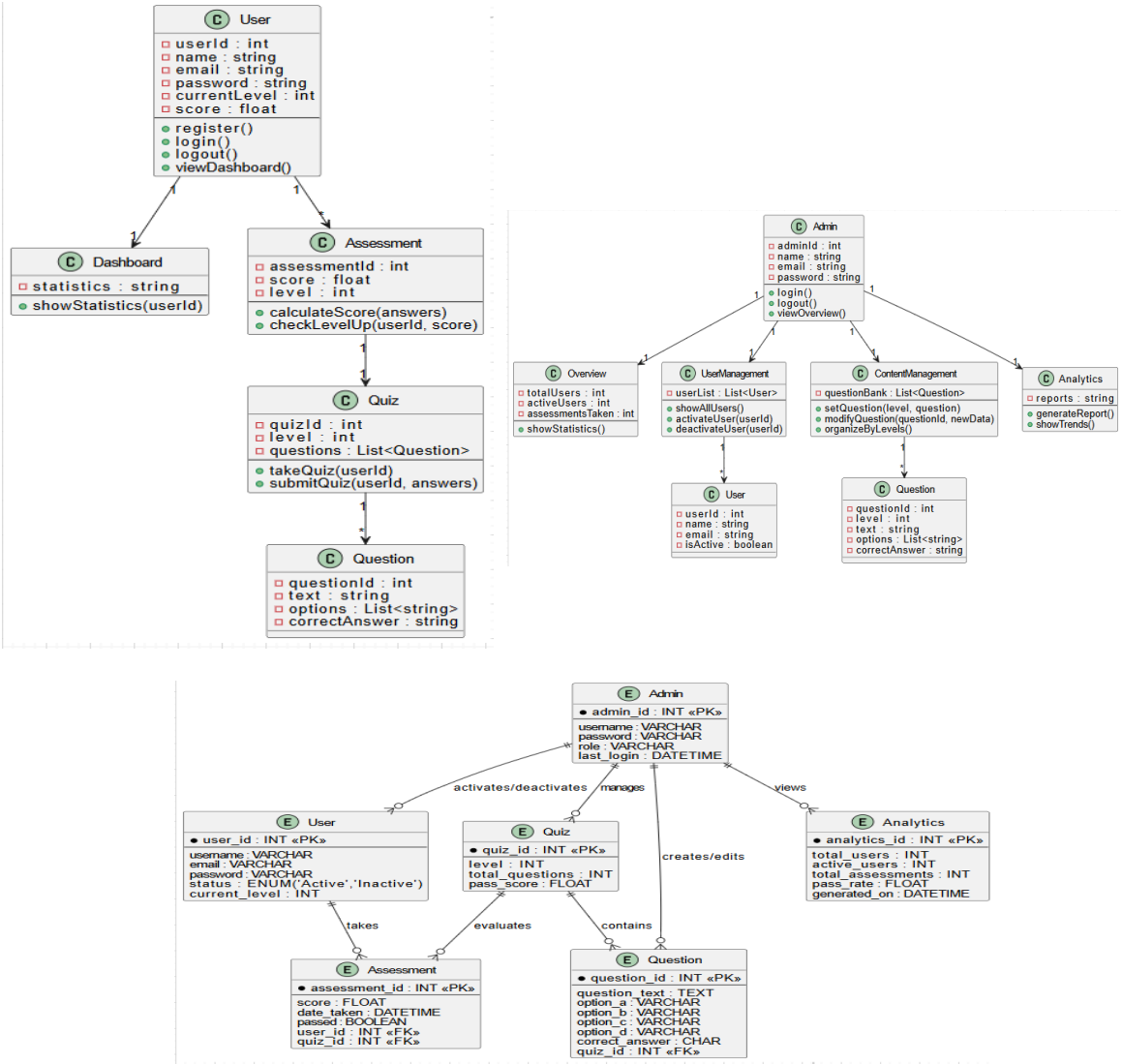

Figure 19: Class Diagram for Each Role

9.2 ER diagram

Institutional communication, also known as the ER model, ER Diagram, or ERD, is a type of structural application used in design. The ERD conveys and depicts the key components of the

limited system, as well as the connections between these elements, in two distinct ways. The user module's entity-relationship diagram is now finished.

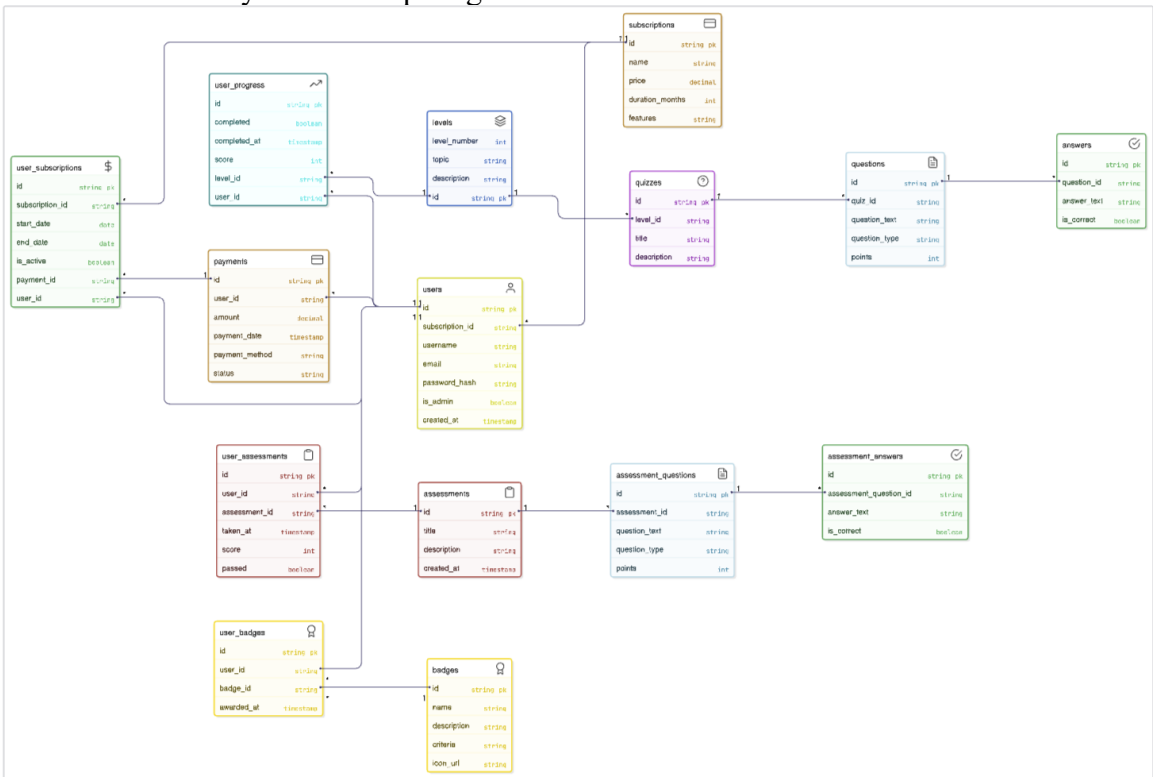

Figure 20: ER Diagram for system Support

9.3 Sequence Diagram

The user and admin dashboard sequence diagram is the focus of this discussion.

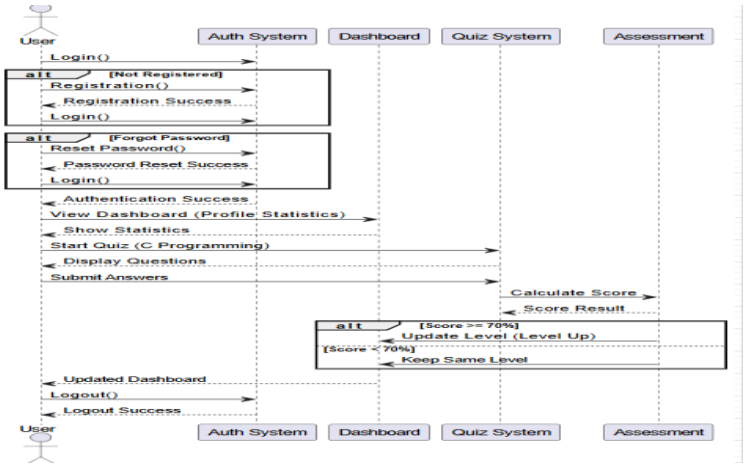

Figure 21: Sequence Diagram For User

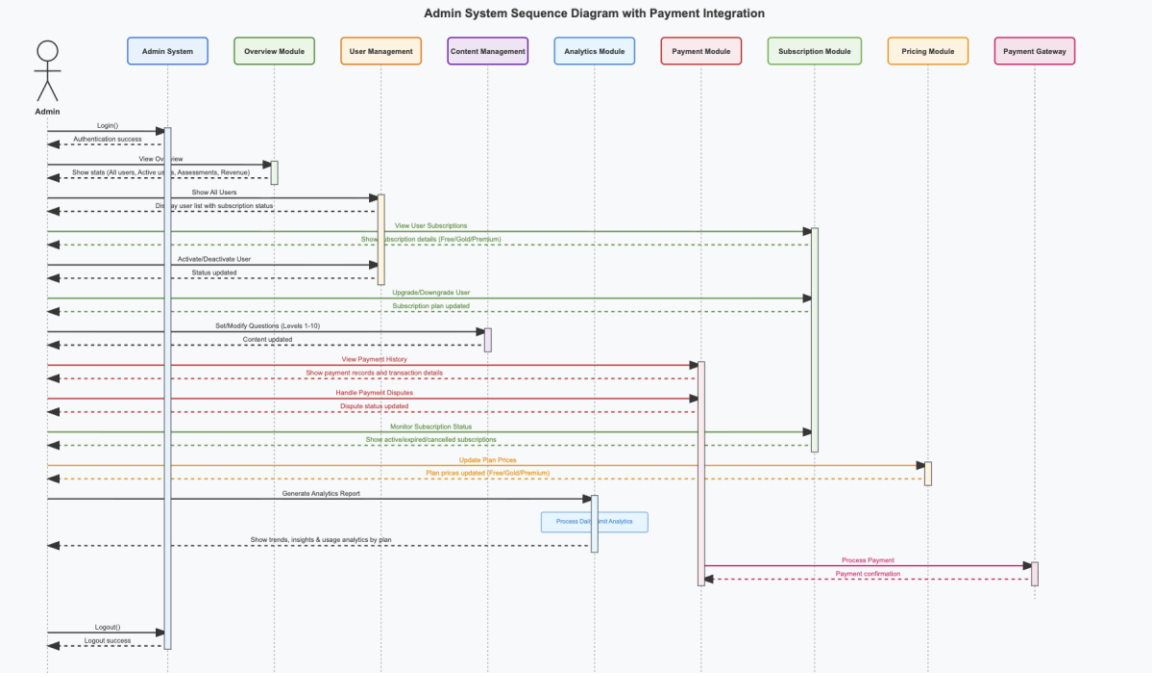

Fig 22: Sequence Diagram for Admin

CHAPTER 10
Development

10.1 Core Module Samples

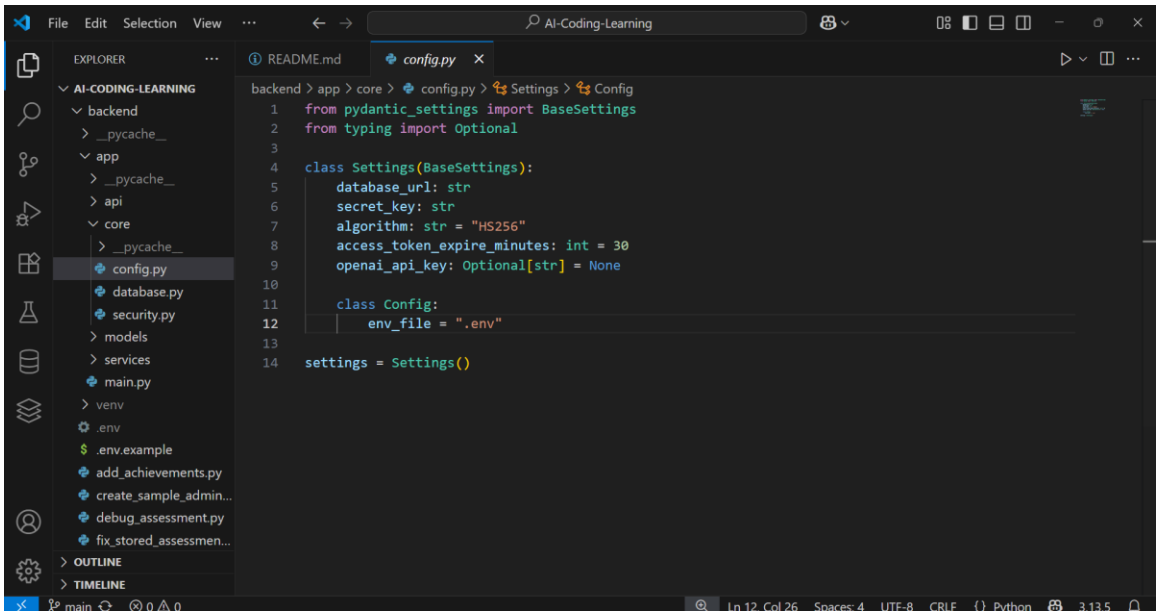

Figure 23: Configure Code for Application

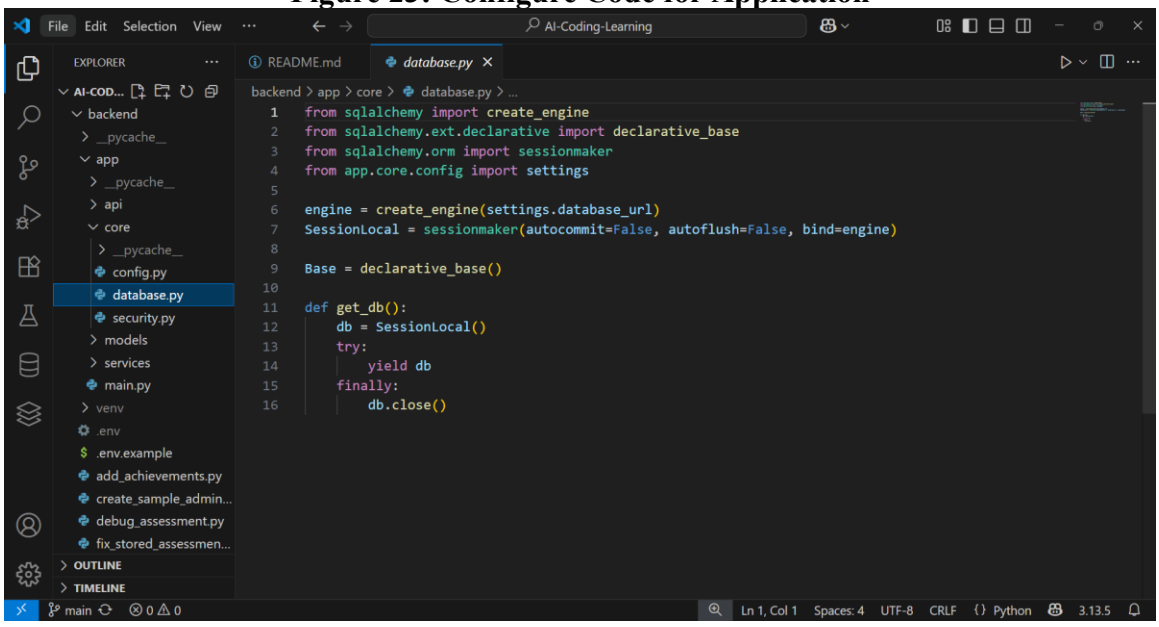

Figure 24: Database Code for Application

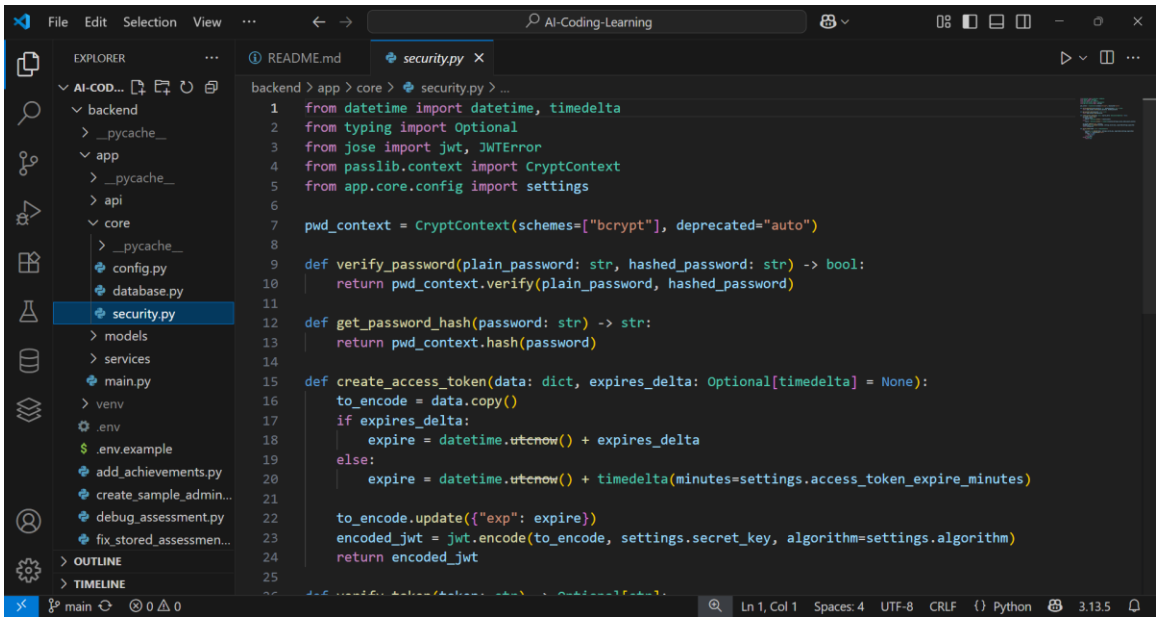

Figure 25: Security Code for Application

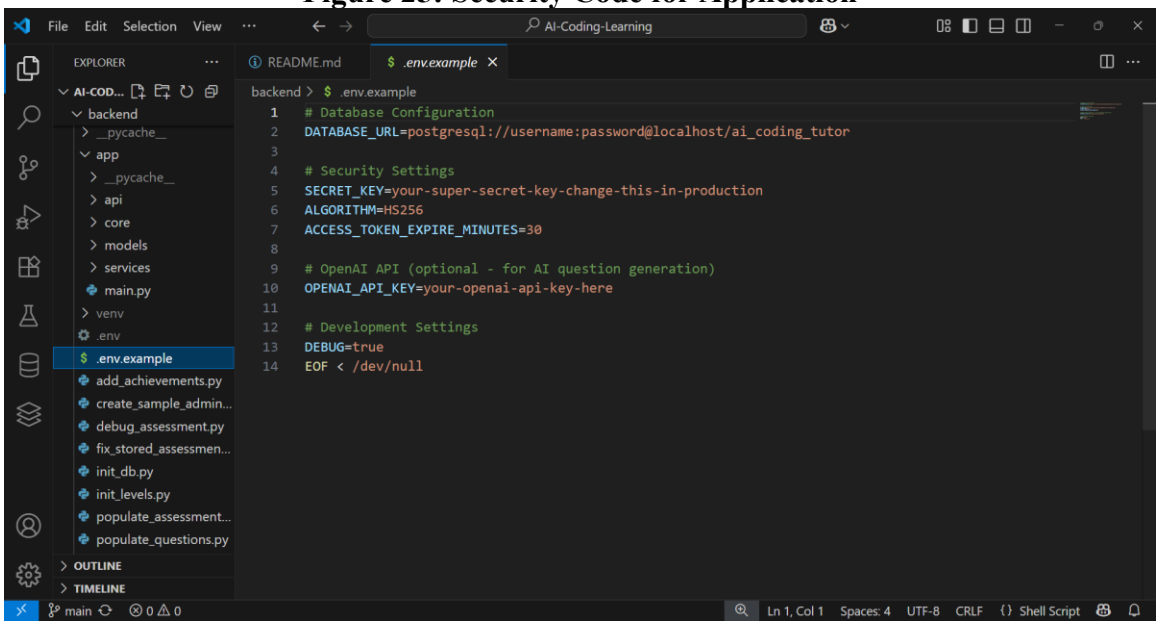

Figure 26: Configure Code for Application

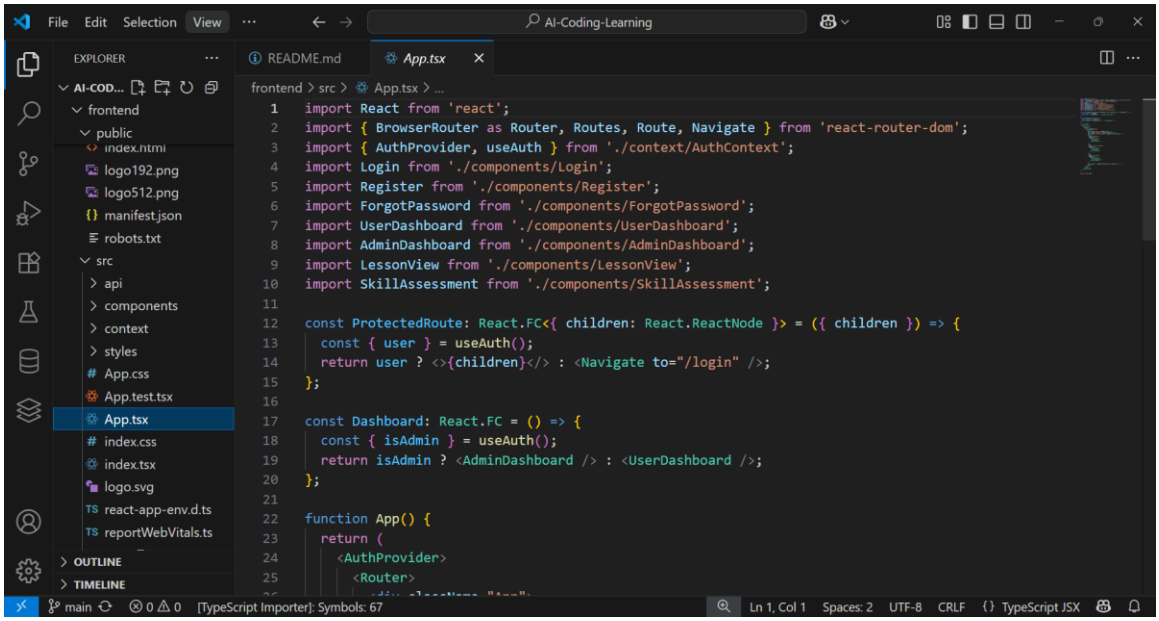

Figure 27: App Test Code for Application

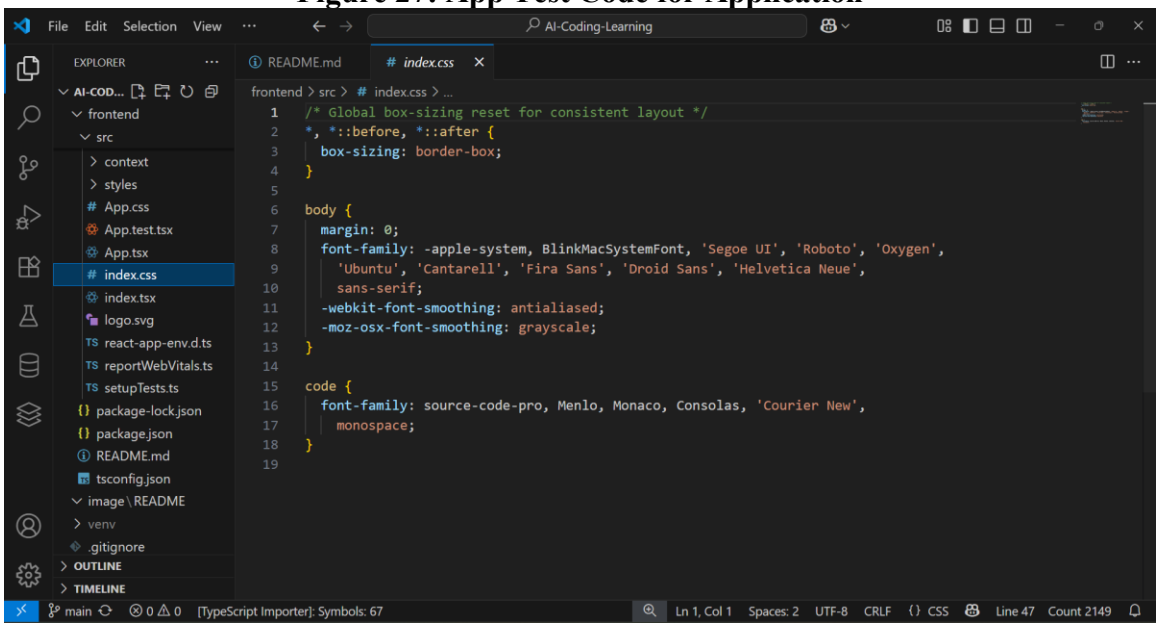

Figure 28: CSS Code for Application

10.2 Probability problem break down

This project's probability idea only describes the likelihood that a user will finish all 10 stages and pass each quiz. A user's likelihood of passing a quiz is determined by how frequently they properly answer questions, since they must receive at least 70% to advance. The likelihood of clearing one level is low if their average accuracy is poor, and as the challenge increases with each level, finishing all 10 becomes extremely unlikely. However, they have a better probability of passing each quiz if they are accurate, which significantly raises the possibility that they will complete all 10 levels. To put it briefly, the likelihood

indicates that a user's chances of advancing through the system are directly enhanced by reliable performance and increased precision.

- **Data security and protection:** processing scanned content correctly to avoid unwanted access, particularly for configuration files and source code. Strong encryption methods help protect data both in transit and at rest, especially when sending scan findings or keeping them in databases.
- **Debugging and testing**: Finding and fixing bugs in mobile or web applications is essential. Every step of the Python development process uses automated testing to help find and fix mistakes.
- **Quality Control and Testing**: During development, thorough techniques are employed to find and address any possible problems or defects. User acceptance, efficacy, and usability were evaluated. Finding and fixing vulnerabilities was one of the goals of security testing. To ensure a dependable, superior system, administrators and stakeholders were consulted prior to any changes being made.

10.3 Prioritization while developing:

Table 9: Prioritization table

Prioritization	Requirements and Explanation
Core Functionality	Adaptive Lesson and Progress Monitoring, Quiz/Assessment System, Performance-Based Adaptive Learning, and User Management (profile, sign-up, and login).
UX	Easy navigation for students, a visual lesson map, a quiz interface, a suggested material area, and a straightforward and interactive frontend (React).
Security and Data Management	JWT authentication for users, safe progress including profile data management, and secure REST API connectivity with the backend database.
Optimization Performance	Low backend (FastAPI) and interface latency, quick quiz loading, and effective database queries to accommodate several users at once.
Integration with external Models	A system for adaptive learning that uses progress and quiz results to suggest appropriate lessons. Deep learning's potential to provide more insightful recommendations in the future.
Quality Assurance and Testing	Frequent testing of frontend convenience, edge scenarios (incomplete lessons, incorrect responses), quiz verification, progress updates, and backend APIs.
User Feedback and Continuous Improvement	Get student input, enhance adaptive algorithms, address usability problems, and incorporate new features like gamification, real-time code compilers, and support for mobile apps.

CHAPTER 11

Testing

Project Name	Digital Storytelling Project	
Name of product	AI Coding Learning System	
Product description	A Duolingo-style interactive platform for learning C programming with AI-powered questions and gamification.	
Project description	HTML, CSS, JavaScript, React, FastAPI, Python.	
Project duration	Project Type	Testing/ Verification
	Start date	End Date
	01-05-2025	08-09-2025

Table 10: Shows start end date

11.1: Test Introduction

One of the most important stages of software development is testing. It guarantees that the built system satisfies user needs, performs accurately in a variety of scenarios, and runs error-free. This project's testing aims to confirm that every component of the AI Coding Learning System operates as planned and seamlessly combines with other components. Additionally, it guarantees the efficient and secure operation of the platform's user management systems, quiz features, and adaptive learning algorithms. To ensure a high-quality final result, the project team uses a methodical testing approach to find and fix any flaws before distribution.

Testing was conducted across various levels—including unit testing, integration testing, validation testing, and system testing—to verify that all features, such as login, registration, adaptive exams, subscription management, and payment processing, function as intended.

11.2 Test Plan

The general strategy, goals, scope, and resources needed for testing are described in the test plan. It specifies what has to be tested, how to test it, who will do it, and when. This test plan aims to verify the AI Coding Learning System's dependability, functionality, usability, and security.

11.2.1 Objectives of the Test Plan:

- Make sure every system module—Admin, User, Quiz, Subscription, and Payment—functions properly and on its own.

- Make sure the backend (FastAPI with database) and frontend (React) are properly integrated.
- Verify that the system functions as intended on various devices and web browsers.
- Verify the accuracy of performance-tracking logic and adaptive learning.
- Examine data security and user authentication systems.

11.2.2 Scope

The testing's preview includes:

- Functional testing (admin dashboard, quizzes, registration, login, and results)
- Non-functional testing (compatibility, security, performance, and usability)
- The web-based platform's complete system testing

11.2.3 Test Environment:

- Frontend: React.js, HTML, CSS, JavaScript
- Backend: FastAPI (Python)
- Database: SQLite / PostgreSQL
- Testing Tools: Browser console, API testing with Postman, manual verification

11.3 Test Plan Acceptance

List the goals and constraints of the test. Prior to implementing the test plan, identify the important participants and secure their consent. Clearly state the acceptance requirements for every testing phase.

11.4 Unit Testing

Testing each module independently and combining the structure as a whole are the best ways to do unit tests. Since it is the smallest part of any module, the software's architecture is the main focus of unit testing efforts. Another name for this is module testing. Every system module is assessed separately. Verify that this approach works in every browser.

11.5 Validation Testing

Software testing ensures that a system satisfies its requirements and operates as intended by using validation and verification procedures. Software quality assurance is another name for it.

11.6 Integration Testing

The problems related to the two concerns of program creation and inspection are addressed by integration testing. After software integration, many high-level assessments are carried out. This testing technique's main goal is to use unit-tested components to generate a program structure that satisfies design criteria.

11.7 TEST CASES

Table 11: Test Case

Case Id	CASE NAME	Expected Result	Actual Result	Result (Pass/Fail)
1	Admin or User Login functions	Login successful.	Login / Sign in successful.	Pass
2	User registration system	Registration successful.	Registration successful.	Pass
3	View user / admin dashboard	Show dashboard details	Successfully	Pass
4	User participates quiz for level up	User can participate in quiz	Successfully	Pass
5	Show their performance result	Result shows	Result shows	Pass
6	Admin update questions and users' details	Update all	Successfully updates	Pass
7	Free User Exceeds Daily limit	Access denied after 5 questions	No access granted after 5 questions	Pass
8	Gold subscription payment	Payment processed, 25 daily questions	Successful subscription	Pass

9	Premium subscription features	Unlimited access granted	Unlimited access	Pass
10	Payment failure handling	Graceful error handling	Error showcasing gracefully	Pass

CHAPTER 12

Implementation

12.1 Training

User	Training	Time	Comment
Users or Clients	Introduction to system features, navigation, and lesson tracking	1 Day	Helps users to understand basic usage and login process
Students / Learners	Hands-on coding lessons, quiz participation, progress tracking	3 Days	Ensures learners can practice coding and check their improvement
Teachers / Instructors	How to assign lessons, monitor student progress, and generate reports	2 Days	Enables teachers to manage classes effectively

Table 12: Training Procedure

12.2 Big Bang Implementation

The Big Bang Implementation method of software development that need to AI based quiz system for programming involves developing each system module independently, then integrating and testing them all at the end. Nothing is tested using this approach until the system is finished. It involves a high risk since it is costly and laborious to resolve faults or integration difficulties, even though it enables the system to be launched fast in one go. As a result, Big Bang Implementation is not appropriate for big or complicated systems, although it could work for small or simpler applications.

12.3 Scaling

The capacity of a software system to expand and manage a growing number of users, demands or data without sacrificing performance is known as scalability. Scaling guarantees that lessons, tests, and adaptive learning calculations can continue to function properly even as additional students sign up for the AI Coding Learning System. Scaling can be accomplished horizontally (by adding additional servers and dividing the workload) or vertically (by adding more resources, such as CPU and memory, to a single server). Because user demand for online learning networks can increase quickly, a well-scaled system maintains its speed, dependability, and efficiency even when traffic volume is high.

12.3.1 Design of scaling

The process of planning and architecture known as "design of scaling" make ensuring that a system can effectively manage increases in users, data, or requests. In order for the AI Coding Learning System to scale independently, the frontend (user interface) and backend (API, database, and AI logic) must be kept apart. Another important factor is database architecture; employing optimal queries, indexing tasks and replication enables the support of several users at once. The system is future-ready for a bigger user base thanks to a

carefully thought-out scaling architecture that guarantees it will be quick, stable, and dependable even under extreme pressure.

12.3.2 Testing Performance

A little piece of software must be developed for each task the scanner does, including disclosure, sensitivity being identified, static and dynamic assessment, and analysis. Instruct development and architectural teams on how to create systems that are efficient and scalable. This entails keeping an eye on things like performance monitoring, caching, diagonal scalability, and database efficiency.

12.4 Load Balancing

To improve the overall effectiveness of this application, load balancing is required. With this application, there are only a few key considerations.

CHAPTER 13
Critical Appraisal and Evaluation

13.1 Objective that could be met

The method seeks to give pupils an effective and individualized coding education. The platform makes sure that students advance gradually based on their performance level by using lesson monitoring, evaluations, and adaptive learning. Students may discover their skills and shortcomings with the use of quiz ratings and real-time progress tracking. Data privacy is ensured via secure user management, and the system's scalable architecture enables it to accommodate more students. Overall, it is possible to successfully accomplish the goals of improving coding abilities, encouraging self-paced learning, and guaranteeing dependable performance for an expanding user base.

13.1.1 Success rate against each objective

With a high success rate, the AI Coding Learning System accomplishes most of its goals. The performance of lesson tracking and monitoring progress is good at nearly 90%, while personalized learning through adaptive suggestions reaches about 80%. Through the expansion of the question bank, quiz evaluation and evaluation can achieve 85%, guaranteeing accurate scoring. With the use of gamification elements and real-time code execution, the system may increase coding abilities by around 85% overall.

13.1.2 How much better could have been done

Although it has successfully met its main goals, the AI Coding Learning System may have been enhanced in a number of ways. Although functional, the adaptive machine learning component still uses simple rule-based reasoning; more precise customization might be achieved by including deep learning or sophisticated AI. Although the scalability design handles modest loads, load balancing, relational replication, and utilization of clouds might improve performance at scale. Additionally, gamification elements like leaderboards and badges, mobile application compatibility, and real-time code execution will greatly enhance user involvement and learning results.

13.1.3 Why it could not be done

Practical constraints prevented the implementation of several upgrades. Due to time constraints, technical limitations, and the intricacy of model training, proficient artificial intelligence or deep learning integration was not feasible. Because they needed extra infrastructure and expenditure beyond the project's present scope, scalability advancements like load balancing and deployment in the cloud were eschewed. To keep the architecture focused on its primary function within the period allowed, features like gamification, support for mobile applications, and real-time code execution were excluded.

13.1.4 Which objectives have been missed

In the current implementation, a few goals have been partially or completely overlooked. The platform's capacity to offer immediate coding practice was limited by the absence of real-time code execution. The lack of support for mobile applications limits accessibility for students using smartphones. Additionally overlooked were gamification elements like awards, leaderboards, and badges, which diminished chances for user engagement and motivation. Furthermore, there is still opportunity for improvement as scalability enhancements like load balancing and cloud deployment were not yet realized.

13.1.5 Why these objectives have missed

These objectives were missed mainly due to time, resource, and scope limitations of the project. Real-time code execution required complex backend integration and additional computational resources, which were not feasible within the given timeframe. Developing a mobile application was beyond the initial project scope and needed separate design and development efforts. Gamification features such as badges and leaderboards were deprioritized to focus on delivering the core learning and assessment modules first. Scalability improvements like cloud deployment and load balancing were not implemented due to infrastructure and budget constraints.

13.1.6 What could have been done to complete those objectives

More preparation and resources might have been needed to accomplish the unmet goals. Integrating online interpreters or sandbox structures like Docker-based execution engines would have allowed for real-time code execution. To increase accessibility, cross-platform

building blocks like React Native or Flutter might have been used to create mobile application support. To increase engagement, gamification aspects might have been implemented through leaderboard modules, point systems, and basic accomplishment tracking.

13.2 Objectives totally not met

In the existing system, several goals were wholly unmet. Because real-time execution of codes was not included, students are unable to run and test programs directly on the platform. The system can only be accessed online because developing mobile applications was not addressed. Additionally, gamification features like leaderboards, medals, and awards were not created, therefore the platform lacked motivating components to improve user engagement. Similarly, the system was only able to be deployed in a rudimentary manner since sophisticated scalability features like load balancing, auto-scaling, and cloud deployment were not addressed.

13.2.1 Including software and documentation

The design approach would have included regular inspection, problem fixing, and user feedback to preserve the software's effectiveness and efficiency. The paperwork, which comprised servicing instructions, technical directions, and user guides, had to be extensive to ensure simple understanding and future enhancements.

CHAPTER 14

Lessons Learned

14.1 Pre-project

The term "pre-project" describes the first stage that preceded the start of the AI Coding Learning System's real development. Identifying the issue, establishing goals, and obtaining requirements were the primary priorities throughout this phase. The issue was that many students found it difficult to track their progress and receive individualized coding instruction using the systems that were in place. Project objectives were established to solve this, including secure user control, adaptive learning, assessments, and lesson logging. The appropriate technologies, such as a database with relationships for storing lessons, tests, and progress, Fast API for the backend, and React for the frontend, were also selected through feasibility studies. The development phase had a defined roadmap and a solid basis thanks to this stage.

14.2 Review

To assess project progress, spot deviations from the plan, and make sure everything is on track, regular reviews are crucial. With the use of these assessments, I can assess how well the web application performs in achieving its goals and decide whether any modifications or enhancements are necessary. Frequent input from participants and readers may aid in pinpointing areas in need of improvement and encourage well-informed choices.

14.3 Lessons Learned

Prior to creating this web application, I conducted extensive research. I researched system analysis and design before creating this application. This is the most difficult phase of system development. It is impossible to generate accurate programming without analysis. I then learnt Fast API and React JS. Learning is challenging. Despite their widespread use, JS and SQLlite may be challenging to comprehend and utilize in their most basic versions. As I begin to learn, I immediately identify my areas of weakness so that I might be considered for next duties.

14.4 Problem Faced

Traditional coding learning systems often provide the same set of courses and examinations regardless of the individual ability level of each learner. Expert learners are not challenged by this one-size-fits-all approach, while novices find it difficult to keep up. Furthermore, most systems lack effective progress tracking and adaptive learning features, which reduces user interest and postpones skill development. A sophisticated, AI-driven learning system that can track user performance and provide personalized information is therefore necessary for better learning outcomes.

14.5 Problems That are solutions

- To develop an AI-powered online application that provides step-by-step coding instruction.
- To use assessments and instruction to track pupils' progress.
- To provide adaptive learning by suggesting additional classes or assessments based on how well students do.
- To implement a secure user management system that keeps track of profile, login, and registration data.
- To give pupils a lively and easy-to-use front-end experience.
- To provide a scalable backend for lessons, tests, and progress logs that is database-integrated.

CHAPTER 15
Conclusion

15.1 Summary of the project:
A web-based tool called the AI Coding Learning System was need to be as created to improve the instructiveness, adaptability, and personalization of programming instruction. To walk students through various c programming coding fundamentals step-by-step, the system combines lesson monitoring, evaluations, and AI-powered adaptive learning. Users may sign up, admin panel, take classes, pass tests, and get suggestions based on their performance that change as they advance. The frontend, created using React, offers learners that need to be an intuitive interface, while the backend, created with Fast API and a database with relations, manages user data, quiz creation, and adaptive logic. The platform has great promise for future works with enhancements, such as sophisticated AI models, real-time compilation tools, and gamified learning experiences, despite present drawbacks like the lack of a mobile app and restricted code execution.

15.2 Goal of the project
This project's primary goal is need develop an intuitive online tool that can called the AI Coding Learning System was need to be as created to improve the instructiveness, adaptability, and personalization of programming instruction. To walk students through various c programming coding fundamentals step-by-step, the system combines lesson monitoring, evaluations, and AI-powered adaptive learning. Users may sign up, admin panel, take classes, pass tests, and get suggestions based on their performance that change as they advance.

15.3 Success of the project
This project has been a huge success. The crucial steps are:

- Dashboard for profile statistical.
- Take Assessment of C programing quiz.
- After getting 70% score of quiz then level Up for assessment (total 10 level)
- Overview (All users, active users, assessment taken etc.)
- Users(Show all users and activate/Deactivate users)
- Content(Set, modify all questions among 10 levels)
- Subscriptions (3-tier model: free, gold, premium)
- Subscription-wise access control
- Payment Module (Handled payment integration with SSLCommerz)
- Analytics

15.4 Documentation
The following phases, tasks, and plans were probably part of the documentation:

- **Preliminary Project Records**: This might involve preliminary needs gathering, feasibility studies, and project concepts.
- **Project plan:** A document that spells out the goals, limitations, schedule, resources needed, and risk-reduction tactics for a project.
- **Technical Specifications**: Comprehensive guidelines on the features, functionalities, and layout of a web application for a software test.
- **User Documentation:** Provides advice and guidance on how to use the program efficiently to all parties, including the person in charge who searches for holiday spots or packages.
- **Testing and Quality Control**: In order to make sure the software satisfies quality requirements, test plans, scenarios, and outcomes must be documented.
- **Deployment and Maintenance plans**: Plans for application upgrades, maintenance, and deployment are given, together with documentation detailing the required procedures.

15.5 Value of the project

By giving students an organized, flexible, and entertaining method to advance their coding abilities, the AI Coding Learning System offers substantial value. In contrast to conventional platforms, it ensures that every student advances at their own speed by tailoring the learning route according to individual performance. Additionally, it provides value through lesson monitoring, real-time evaluation, and safe user data management. It is a tool for teachers to track students' development, and for institutions, it is an extensible framework that can accommodate big student populations. All things considered, the initiative adds educational, scientific, and practical benefit by improving the effectiveness, accessibility, and personalization of coding instruction.

15.6 My Experience

I may use my abilities in project management, design, and development for the web application project for a programming quiz system. Working with stakeholders, effectively managing project timetables, overcoming technical obstacles, and putting into practice solutions created especially to satisfy the demands of developing programming skills are all made possible by my experience. As part of my efforts to further my career, I have also strengthened my problem-solving, teamwork, and documentation skills.

References

[1] Zhou, Y., et al. *Challenges of Generative AI in Student Learning and Coding Skill Development*. *Computers*, MDPI, 2024. Available at: https://www.mdpi.com/2073-431X/14/5/185

[2] Han, L., et al. *Ethical and Privacy Challenges in AI-Based Education Systems*. *Education, Humanities and Social Sciences*, DRPress, 2024. Available at: https://drpress.org/ojs/index.php/EHSS/article/view/24840

[3] Duckett, J., 2001. *Web Design with HTML, CSS, JavaScript and jQuery*. 1st ed.

[4] W3schools.com. 2021. *JavaScript Tutorial*. [Online] Available at: <https://www.w3schools.com/js/default.asp> [Accessed 17 December 2021].

[5] W3schools.com. 2021. *What is Bootstrap*. [Online] Available at: <https://www.w3schools.com/whatis/whatis_bootstrap.asp> [Accessed 17 December 2021].

[6] Learn about React Js, available at <<https://youtu.be/4UZrsTqkcW4>>, Last accessed on 20-01-2023 9:48 PM.

[7] Learn about HTML and CSS, available at <<https://youtu.be/-8ORfgUa8ow>>, Last accessed on 04-012023 4:23 AM.

[8] Learn about React Firebase, available at <<https://youtu.be/D9W7AFeJ3kk>>, Last accessed on 04-012023 4:28 AM.

[9] Learn about JavaScript, available at <<https://youtu.be/2Ji-clqUYnA>>, Last accessed on 21-01-2023 5:08 AM.

213-16-594

ORIGINALITY REPORT

15%	13%	1%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	9%
2	Submitted to University of Greenwich Student Paper	2%
3	Submitted to Daffodil International University Student Paper	1%
4	Submitted to Gulf College Oman Student Paper	<1%
5	Submitted to Kingston University Student Paper	<1%
6	Submitted to Durban University of Technology Student Paper	<1%
7	hdl.handle.net Internet Source	<1%
8	apothesis.lib.hmu.gr Internet Source	<1%
9	21caforg.weebly.com Internet Source	<1%
10	Submitted to American International School of Guangzhou Student Paper	<1%
11	Submitted to Southwest Tennessee Community College Student Paper	<1%

21% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

19 AI-generated only 10%
Likely AI-generated text from a large-language model.

17 AI-generated text that was AI-paraphrased 11%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.