

InQuizitiveAI

Final Year Project

Session 2022-2026

A project submitted in partial fulfillment of the degree of

BS in Gaming & Multimedia



Department of Computer Science

Faculty of Computer Science & Information Technology

The Superior University, Lahore

Spring 2025

Type (Nature of project)	[☒] Development [☐] Research [☐] R&D			
Area of specialization	Game Development			
FYP ID	FYP-BGMM-S25-005			
Project Group Members				
Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	Bsgmm-s24-007	Mahnoor Fatima	Su92-bgmm-s24-007@superior.edu.pk	
(ii)	Bgmm-s22-003	Muhammad Umair	bgmm-s22-003@superior.edu.pk	
(iii)	Bgmm-s22-005	Mahar Faiq Lak	bgmm-s22-005@superior.edu.pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I **Mahnoor Fatima** group leader of FYP under registration no **Bsgmm-s24-007** at Computer Science Department, The Superior University Lahore. I declare that my FYP report is checked by my supervisor.

Date: _____ Name of Group Leader: Mahnoor Fatima Signature: _____

Name of Supervisor: Mr.Nabeel Akram

Designation: Lecturer

Signature: _____

HoD: Dr. Muhammad Azam

Signature: _____

InQuizitiveAI

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Mahnoor,Umaid,Faiq	1.0	19/03/25	Initial draft with project overview and main objectives.	
Mahnoor,Umaid,Faiq	1.1	19/06/25	Updated content based on supervisor's suggestions and refined requirements.	
Mahnoor,Umaid,Faiq	1.2	25/10/25	Revised after academic review, improved diagrams and documentation flow.	
Mahnoor,Umaid,Faiq	1.3	04/11/25	Added complete project plan with timeline and milestones.	
Mahnoor,Umaid,Faiq	1.4	02/12/25	Made further updates according to supervisor's latest review.	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Dedication

I dedicate this Final Year Project to my beloved parents, whose unwavering love, prayers, and support have been the foundation of all my achievements. To my teachers and mentors, for their invaluable guidance, knowledge, and encouragement throughout my academic journey.

And to my friends and classmates, for their constant motivation, collaboration, and companionship during challenging times.

Without your support, this journey would not have been possible.

Acknowledgements

First and foremost, I am profoundly grateful to Almighty Allah for granting me the strength, patience, and knowledge to complete this Final Year Project successfully.

I would like to express my heartfelt gratitude to my supervisor, *Mr. Nabeel Akram*, for their invaluable guidance, continuous support, and constructive feedback throughout the development of this project. Their expertise and encouragement played a crucial role in shaping this work.

A special thanks to the faculty and staff of the Department of Computer Science for providing the necessary resources, a productive learning environment, and continuous support during the entire academic journey.

I am also thankful to my family, especially my parents, for their unconditional love, prayers, and constant encouragement, which kept me motivated throughout this project.

Executive Summary

Our project presents a unique approach to quiz games with a three-genre quiz developed in Unity using C#. The game encompasses Mathematics, General Knowledge, and English categories to offer a diverse and enriching experience for players. By integrating an AI-powered chatbot for providing hints, we enhance user engagement and facilitate problem-solving. Unlike conventional quiz games, our project offers limitless levels that dynamically adjust difficulty levels based on player progress, fostering continuous learning and challenge. With the aim of creating an interactive and intelligent learning platform, our project not only entertains but also educates users effectively, catering to a diverse range of audiences seeking both entertainment and knowledge enrichment.

Table of Contents

InQuizitiveAI	i
Plagiarism Free Certificate	ii
Dedication	v
Acknowledgements	vi
Executive Summary	vii
Table of Contents	viii
List of Figures	xi
List of Tables.....	xii
Chapter 1	1
Introduction	1
1.1. Background	2
1.2. Motivations and Challenges.....	2
1.3. Goals and Objectives	3
1.4. Literature Review/Existing Solutions.....	3
1.5. Gap Analysis	4
1.6. Proposed Solution	4
1.7. Project Plan	4
1.7.1. Work Breakdown Structure.....	5
1.7.2. Roles & Responsibility Matrix.....	6
1.7.3. Gantt Chart.....	7
1.8. Report Outline.....	7
1.9. Empathy Map.....	8
Chapter 2.....	9
Software Requirement Specifications.....	9
2.1. Introduction	10
2.2. Purpose.....	10
2.2.1. Document Conventions.....	10
2.2.2. Intended Audience and Reading Suggestions.....	11
2.2.3. Product Scope.....	11
2.2.4. References.....	12
2.3. Overall Description.....	12
2.3.1. Product Perspective	12
2.3.2. User Classes and Characteristics.....	13
2.3.3. Operating Environment.....	13
2.3.4. Design and Implementation Constraints.....	14
2.3.5. Assumptions and Dependencies	14
2.4. External Interface Requirements	15
2.4.1. User Interfaces.....	16
2.4.2. Hardware Interfaces	16
2.4.3. Software Interfaces.....	17
2.4.4. Communications Interfaces.....	17
2.5. System Features	17

2.5.1.	System Feature 1	18
2.5.1.1.	Description and Priority	18
2.5.1.2.	Stimulus/Response Sequences	18
2.5.1.3.	Functional Requirements.....	19
2.5.2.	System Feature 2	19
2.5.2.1.	Description and Priority	19
2.5.2.2.	Stimulus/Response Sequences	20
2.5.2.3.	Functional Requirements.....	20
2.5.3.	System Feature 3 (and so on)	21
2.6.	Nonfunctional Requirements.....	21
2.6.1.	Performance Requirements	22
2.6.2.	Safety Requirements.....	22
2.6.3.	Security Requirements.....	22
2.6.4.	Usability Requirements	23
2.6.5.	Reliability Requirements	23
2.6.6.	Maintainability/Supportability Requirements	24
2.6.7.	Portability Requirements	24
2.6.8.	Efficiency Requirements.....	25
2.7.	Domain Requirements.....	25
Chapter 3.....		26
Use Case Analysis		26
3.1.	Use Case Model	27
3.2.	Use Cases Description	28
Chapter 4.....		30
System Design		30
4.1.	Architecture Diagram	32
4.2.	Domain Model	33
4.3.	[Entity Relationship Diagram with data dictionary.....	34
4.4.	Class Diagram.....	35
4.5.	Sequence / Collaboration Diagram	36
4.6.	Operation contracts	36
4.7.	Activity Diagram	37
4.8.	State Transition Diagram	38
4.9.	Component Diagram	39
4.10.	Deployment Diagram.....	40
4.11.	Data Flow diagram [<i>only if structured approach is used - Level 0 and 1</i>].....	41
Chapter 5		42
Implementation.....		42
5.1.	Important Flow Control/Pseudo codes.....	43
5.2.	Components, Libraries, Web Services and stubs.....	43
5.3.	Deployment Environment.....	44
5.4.	Tools and Techniques	45
5.5.	Best Practices / Coding Standards.....	45
5.6.	Version Control	46

Chapter 6.....	47
Testing and Evaluation.....	47
6.1. Use Case Testing	48
6.2. Equivalence partitioning	48
6.3. Boundary value analysis	49
6.4. Data flow testing	49
6.5. Unit testing	50
6.6. Integration testing.....	50
6.7. Performance testing.....	50
6.8. Stress Testing.....	51
Chapter 7.....	52
Summary, Conclusion and Future Enhancements.....	52
7.1. Project Summary.....	53
7.2. Achievements and Improvements	53
7.3. Critical Review.....	54
7.4. Lessons Learnt.....	54
7.5. Future Enhancements/Recommendations	54
Appendices.....	56
Appendix A: Information / Promotional Material	57
Appendix [no.]: Appendix Title	60

List of Figures

Figure 1.1 Gantt Chart.....	7
Figure 1.2 Empathy Map	8
Figure 3.1 Use Case Model.....	28
Figure 4 System Design	32
Figure 4.1 Architecture Diagram.....	33
Figure 4..2 Domain Model.....	34
Figure 4.3 ERD	35
Figure 4.4 Class Diagram	36
Figure 4.5 Sequence / Collaboration Diagram.....	37
Figure 4.7 Activity Diagram	38
Figure 4.8 State Transition Diagram	39
Figure 4.9 Component Diagram.....	40
Figure 4.10 Deployment Diagram	41
Figure 4.11 Data Flow diagram.....	42

List of Tables

Table 1 Roles & Responsibility Matrix	6
---	---

Chapter 1

Introduction

Chapter 1: Introduction

This chapter presents our creative project, a three-genre quiz game developed in Unity using C#. Players will be presented with questions from Mathematics, General Knowledge, and English to provide entertainment as well as educational content. One of the features is the presence of an AI chatbot that provides hints to help players overcome difficult questions, engaging users. Unlike traditional quiz games, our project features limitless levels that increment in level of difficulty according to the advancement made by players. Our main purpose is to develop a dynamic, intelligent learning setup that seeks both fun and meaningful learning for our clients.

1.1. Background

Our project is a three-genre quiz game created in Unity with C#. The game tests players with questions from different categories, including Mathematics, General Knowledge, and English, making it an interesting and learning experience. For greater user interaction, an AI-driven chatbot is implemented, offering hints to help players answer tough questions. As opposed to common quiz games, this project includes unlimited levels with dynamically adjusted difficulty based on the performance of players. The aim is to develop an intelligent and interactive learning platform that not only amuses but also educates the user efficiently. The blending of varied quiz types, AI bot support, and flexible difficulty levels will help learners have fun while playing and get captivated at the same time, regardless of age.

1.2. Motivations and Challenges

The motivation behind our project is to transform the quiz game experience by fusing entertainment with learning, providing players with a dynamic and interactive platform to build their knowledge in various subjects. However, this ambitious goal comes with its fair share of challenges. Creating a three-genre quiz game with an AI chatbot feature involves complex design and careful coding to achieve smooth functionality and user-friendly interface. In addition, designing an algorithm that dynamically changes the

difficulty level according to player performance makes the gameplay mechanics more complicated. Despite these difficulties, we are motivated by the desire to provide a state-of-the-art and interactive learning tool that engages users and enables them to develop intellectually through play.

1.3. Goals and Objectives

Our game project is dedicated to transforming the conventional quiz game into an enjoyable and learning-friendly experience for the players. Integrating three game genres into one game and implementing different categories such as Mathematics, General Knowledge, and English, we are making an effort to provide a mixed learning platform. The implementation of an AI chatbot provides the players with customized support, maximizing user interaction and satisfaction. With infinite levels that are adjusted according to player performance, our intention is to design a challenging as well as fulfilling experience that encourages ongoing learning. Through this interactive and smart platform, our aspiration is to engage and educate users efficiently, encouraging people to be knowledge-friendly and curious.

1.4. Literature Review/Existing Solutions

Our product, a three-genre quiz game created on Unity with the use of C#, improves current solutions by adopting novel features in increasing user participation and learning. In contrast to common quiz games that have set numbers of levels, our game excels at having unlimited levels that dynamically modify depending on individual performance. Based on questions cut across various disciplines such as Mathematics, General Knowledge, and English, we plan to offer all-rounded learning experience. By combining an AI-driven chatbot that provides tips to aid players, we aim to provide a platform that not only entertains but also learns well. Through a literature review, we identified the need for of interactive and smart learning tools became apparent, and thus we endeavored to create a game that fulfills these needs and goes beyond typical quiz games.

1.5. Gap Analysis

The gap analysis of our three-genre quiz game developed in Unity using C# will identify what the reality is in those areas where the current state falls short of achieving our goals. One of the key gaps is the absence of detailed analytics for keeping a record of the user's progression and adjusting the difficulty level accordingly. Secondly, there is more refinement needed to make the AI chatbot replies more accurate and helpful as hints tailored for a particular player. This is coupled with missing feedback mechanisms that would give users an avenue to provide input for constant improvement. Addressing these gaps will be critical in ensuring our quiz game truly delivers an engaging and educational experience that meets the expectations of the target audience.

1.6. Proposed Solution

The proposed solution is a Unity-based three-genre quiz game in C# that subjects the players to interesting challenges in categories such as Mathematics, General Knowledge, and English. An AI chatbot provides hints to the user for solving hard questions, with a view to increasing user interest. This creative work includes unlimited levels of difficulty, wherein levels automatically get updated based on player performance-a novelty that distinguishes this work from traditional quiz games. The aim is to provide interactive learning that should be entertaining yet effectively educate the user on various things, making it a dynamic and intelligent knowledge-acquiring platform.

1.7. Project Plan

The proposed project will design a Unity-based three-genre quiz game in C#. The game will allow players to choose questions between Mathematics, General Knowledge, and English, thus making the game interesting as well as informative. We will embed an AI-powered chatbot that will provide hints and help the player with difficult questions and thus enhance the users' interaction. Unlike other common and mundane quiz games, our project doesn't have limited levels; rather, it will change dynamically depending on the performance of the player, leading

to increased difficulty as time progresses. We intend to provide an interactive and intelligent platform for learning, entertaining users while educating them about various concepts effectively.

1.7.1. Work Breakdown Structure

Unity Game Engine

C# Programming Language

Mathematics Category

General Knowledge Category

English Category

AI Chatbot Integration

Hint Generation Algorithm

Level Generation System

Performance Tracking Mechanism

Engagement Metrics

Educational Content

Interactive User Interface

Dynamic Difficulty Adjustment

User Experience Enhancement

Unity Game Engine

1.7.2. Roles & Responsibility Matrix

Table 1 Roles & Responsibility Matrix

Numbering	Deliverable	Number of Days	Responsible Team Member
1	Coding.	25 days	Faiq
2	Testing.	1 Days	Mahnoor, Umair
4	Asset Collection	2 Days	Umair
5	Designing	10 Days	Mahnoor
6	Quality Check	2 Days	Umair
7	UI Design	5 Days	Mahnoor
8	AI-powered Chatbot	28 Days	Faiq
9	Unlimited levels	15 Days	Mahnoor, Umair

1.7.3. Gantt Chart

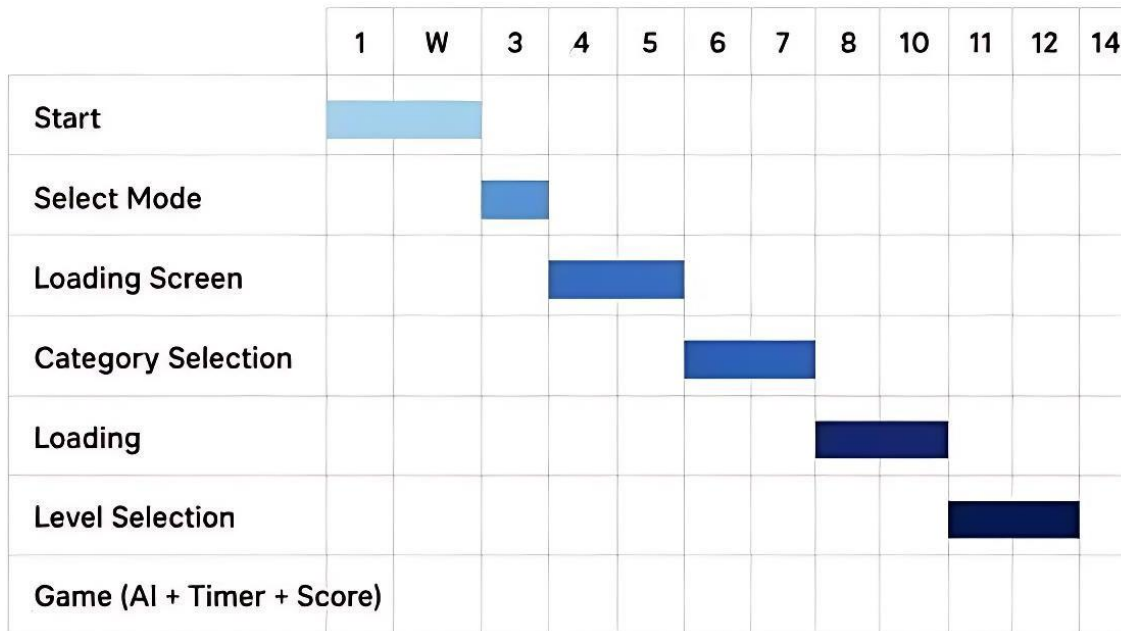


Figure 1 Gantt Chart

1.8. Report Outline

Chapter 1: Introduction

Chapter 2: Software Requirement Specifications

Chapter 3: System Design and Implementation

Chapter 4: Testing, Results, and Conclusion

Appendices: UI designs, Gantt chart, code snippets

1.9. Empathy Map

Says • I want a quiz game that is easy to understand and enjoyable • Hints should help me, not confuse me • Questions should match my skill level • I hope the game runs smoothly without errors	Thinks • Will the next question be harder? • The AI chatbot should guide me at the right moment • Is this game helping me improve my knowledge? • I want a fair challenge that keeps me learning
Does • Selects categories like Mathematics, General Knowledge, and English • Uses AI-powered hints when stuck • Plays multiple levels as difficulty increases automatically	Feels • Curious to discover new questions • Confident when the chatbot provides useful hints • Challenged but motivated as difficulty rises

Figure 2 Empathy Map

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

Our Unity-developed three-genre quiz game in C# will challenge players with questions in Mathematics, General Knowledge, and English for an edutaining experience. Through the use of an AI-powered chatbot, the users will be able to get hints for questions that they find difficult, which will increase user interaction. Unlike other quiz games, our project will feature unlimited levels that become progressively difficult based on the performance of every player. The goal is to provide a stimulating, interactive, intelligent learning interface that is both entertaining and educating, clearly differentiating this from any other quiz game.

2.2. Purpose

This SRS report details the software requirements for the Three-Genre Quiz Game project, with special emphasis on the game development platform of Unity using C#. The product seeks to present players with a fun and learning experience through intellectually challenging questions across genres such as Mathematics, General Knowledge, and English. The inclusion of a chatbot powered by artificial intelligence provides an added dimension of user interaction through the provision of hints to guide players in answering tough questions. The feature of unlimited levels and dynamic difficulty adjustment in the game differentiates it from other quiz games. The final aim of this project is to develop an interactive and smart learning platform that engages and educates users efficiently.

2.2.1. Document Conventions

While writing this Software Requirements Specification (SRS) for our three-genre quiz game project coded in Unity using C#, we have adhered to typographical conventions to be clear and consistent. Priority levels for requirements are shown through the use of different fonts and highlighting styles to show their importance. Top-level requirements are presumed to be inherited by low-level requirements, with each requirement statement carrying its own priority for easy reference. SRS specifies the creation of a quiz game that asks questions from subjects

such as Mathematics, General Knowledge, and English to provide an entertaining and informative experience. The addition of an AI-based chatbot gives players hints to help them solve difficult questions, enhancing user engagement. With unlimited levels that have dynamically escalating difficulty, our project seeks to provide an interactive and smart learning portal that informs and amuses users in a most efficient manner.

2.2.2. Intended Audience and Reading Suggestions

The document is written for a wide range of readers such as developers, project managers, marketing personnel, users, testers, and documentation writers. It has in-depth information on the three-genre quiz game created in Unity with C#, such as the game mechanics, question categories, implementation of an AI chatbot, and the novel feature of unlimited levels. For developers, it has technical details of game development and coding. Project managers will have project milestones and timeline information. Marketing personnel will see the user engagement strategies. The users will receive a guide for playing the game. Testers will have details on testing procedure. Documentation authors will receive guidelines for creating user manuals. We recommend beginning with the overview sections to get an idea of the game concept and then moving on to sections pertaining to each type of reader for a complete understanding of the project.

2.2.3. Product Scope

The software is a three-genre quiz game built in Unity with C#. The game tries to give the user an enjoyable and learning experience by testing their knowledge with questions from various domains such as Mathematics, General Knowledge, and English. It contains an AI chatbot to give hints, boosting user engagement. The game is unique with its unlimited levels that change difficulty according to the performance of the players, unlike other quiz games. The aim is to develop an interactive and smart learning platform that is entertaining and educating in nature. Through the use of this software, our corporate objective is to provide a novel and engaging gaming experience that supports our business strategy of delivering value-added educational entertainment to our audience.

2.2.4. References

Our Software Requirements Specification (SRS) of our three-genre quiz game project for Unity using C# refers to various primary documents and web pages that are core to its creation and implementation. Some of them include the User Interface Style Guide in developing the game's graphic aspect, the System Requirements Specifications with regard to its technical components for functionality, and the Use Case Documents to identify user actions in the game. Further, the project also relies on a Vision and Scope Document to give a strategic perspective and define specific goals for the game development. All of these references are significant in directing the design and development process so that the game fulfills its purpose of offering an entertaining and educational experience to different age and skill groups of players.

2.3. Overall Description

Our project is a three-genre quiz game built in Unity with C#. The game quizzes users with questions from diverse categories, including Mathematics, General Knowledge, and English, in order to provide an interactive and educational experience. For improved user interaction, an AI-based chatbot is implemented, offering hints to help players solve challenging questions. In contrast to conventional quiz games, this project has unlimited levels, dynamically escalating the difficulty according to the performance of the player. The aim is to develop an interactive and smart learning platform that not only engages but also teaches users efficiently.

2.3.1. Product Perspective

The project defined in this SRS is a three-genre quiz game developed in Unity with C#. It is a new, standalone product designed to deliver an entertaining and informative experience to the users. The game tests the users with questions from different genres such as Mathematics, General Knowledge, and English. For enhanced user interaction, the game has an AI-based chatbot that provides hints to assist users in answering difficult questions. In contrast to traditional quiz games, this project provides unlimited levels that dynamically adjust difficulty depending on the performance of the players. The main aim of this project is to provide an interactive and smart learning environment that is both entertaining and very effective in

educating users. A simple diagram presenting the main elements of the system, subsystem interactions, and external interfaces would be helpful in gaining an idea about the overall design of the game.

2.3.2. User Classes and Characteristics

In creating our three-genre quiz game product in Unity with C#, we have developed several user classes that we expect to interact with the product. These user classes differ according to their technical skill, educational level, and usage frequency. The initial user class includes casual gamers for entertainment and learning purposes, who may have differing levels of technical competence and educational attainment. The second class of users is competitive gamers with a higher level of experience in quiz games and wanting demanding experiences. The third user class is educators or parents seeking educational software for their children or students, needing a higher level of content management and observation. There are particular requirements for each user class based on their demands, with an emphasis on offering interactive gameplay and learning material. The most valuable user class in this product is casual players and educators, and they are our main target population that we should fulfill with our intelligent and interactive learning platform.

2.3.3. Operating Environment

The software shall run in the Unity environment and use C# programming language. It needs to run on a stable hardware system that can efficiently run Unity-based applications. The operating system should be Unity compatible and preferably the latest version so that it will run with optimum performance. Also, the software shall have to coexist with AI-based chatbot modules for hint provision without any conflicts. It is essential that all software elements integrate harmoniously with each other to provide a strong and interactive user experience. The intention is to provide a dynamic and informative quiz game that tests players in multiple categories with an interactive and intelligent learning platform.

2.3.4. Design and Implementation Constraints

Limitations that will affect the choices available to the developers for this project are the specialized technologies and tools needed to create Unity games with C#. Moreover, the incorporation of an AI-based chatbot into the game will present constraints on communication protocols and security aspects to facilitate smooth and secure interaction between the chatbot and players. Additionally, the requirements for unlimited levels and adaptive difficulty will mean attention to memory requirements and conventions of design to ensure continuous playability experiences. These requirements will similarly affect database and parallel operation selection to enable the game's sophisticated features while respecting programming conventions aimed at real-world interactive and intelligent learning goals.

2.3.5. Assumptions and Dependencies

Assumptions and considerations that may influence the requirements mentioned in the SRS for the three-genre quiz game project implemented in Unity with C# are:

Third-party chatbot integration:

The performance and usability of the AI-driven chatbot may influence user experience and game performance if the integration is not smooth.

Sources of data for quiz questions:

Assumptions regarding the availability, quality, and consistency of question databases for various categories such as Mathematics, General Knowledge, and English may affect the diversity and difficulty of questions in the game.

Unity plugin compatibility:

Reliance on third-party Unity plugins or assets to develop the game may result in compatibility problems or restrictions if the plugins are not maintained or updated.

User engagement metrics: Player behavior and preference assumptions may influence the creation and progression of unlimited levels upon player performance, influencing the game's balancing and user retention mechanics.

Educational effectiveness:

It is assumed that the game learns users effectively, both educating them and entertaining them, depending on the accuracy and relevance of contents and the level of difficulty adapts to user performance. These dependencies and factors emphasize the need for proper documentation and communication throughout the project to ensure the right implementation of requirements and control over risks.

2.4. External Interface Requirements

Graphic User Interface (GUI):

The game needs an easy-to-use interface with smooth navigation for a hassle-free gameplay experience on various devices.

Audio Interface:

Incorporation of sound effects, background score, and voiceovers to support the gaming experience and offer feedback to players.

Chatbot Integration:

The chatbot powered by AI should be a responsive interface, offering hints and instructions based on player requirements during the gameplay.

Multiplayer Interface:

Online multiplayer support, enabling people to compete against or work alongside each other in real-time.

Social Media Integration:

Sharing facilities for achievements and high scores as well as requesting friends through social media for greater interaction.

In-App Purchases:

Implementing a payment interface for security to enable purchases of extra hints, power-ups, or other new features from within the game.

2.4.1. User Interfaces

In the user interface design for our three-genre quiz game built with Unity and C#, the logical properties of each interface between the software product and the users will be of utmost importance. This means maintaining a visually appealing experience with sample screen snapshots showing the varying quiz subjects such as Mathematics, General Knowledge, and English. GUI standards and product family style guides will be used to ensure consistency and user familiarity. The constraints of the screen layout will be optimized for the user experience with standard buttons and functions such as a help option on all screens. Keyboard shortcuts can be included for fast navigation. Error message display standards will be applied to give good feedback. The user interface design will emphasize improving user interaction with the AI-driven chatbot for hints, developing a dynamic and learning gameplay experience through unlimited levels, adjusting difficulty according to player performance. These aspects will be covered in a standalone user interface specification to provide a consistent and enjoyable user experience.

2.4.2. Hardware Interfaces

To play our game, you just need an Android Device. It does not matter at all what kind of processor the user's device should have. Because, our game is small enough that it does not need a very good and gaming processor to run it. Even if our user has a low end device, he can easily play the game on that to. The minimum requirement of our game is only 1GB RAM. It will also run smoothly in a Low End Device

2.4.3. Software Interfaces

Our project highlights a next-generation three-genre quiz game developed in Unity with C#. Providing a wide variety of questions across Mathematics, General Knowledge, and English, the game guarantees an engaging and informative gaming experience. Adding a level of user engagement, the inclusion of an AI-based chatbot introduces a dynamic element by providing hints to navigate through difficult questions. In contrast to traditional quiz games, our project has unlimited levels, with the difficulty levels adjusting according to the progress of the players. Our goal is to create an interactive and smart learning center that harmoniously weaves entertainment and educational enrichment together.

2.4.4. Communications Interfaces

In our three-genre quiz game implemented in Unity with C#, we have established solid communications interfaces to give an effortless and interactive user experience. The game presents players with questions across Mathematics, General Knowledge, and English, giving a varied and stimulating quiz atmosphere. To help user interaction, we have introduced an AI-based chatbot which provides useful hints to assist players through difficult questions. In contrast to traditional quiz games, our project provides infinite levels, dynamically adapting difficulty levels based on player achievement. Our final goal is to develop an intelligent and interactive learning platform that combines entertainment with good education, giving users a meaningful and enriching experience.

2.5. System Features

Our project is a three-genre quiz game created in Unity with C#. The game presents users with questions from various genres such as Mathematics, General Knowledge, and English, providing a fun and educational experience. In order to ensure user interaction, we have implemented an AI chatbot that offers hints to users in order to answer difficult questions. What makes this game unique is its feature of unlimited levels, where the level of difficulty can change dynamically as per the performance of players. Our vision is to build an interactive and smart

learning platform that amuses and teaches users efficiently, providing a new and innovative way of quiz gaming.

2.5.1. System Feature 1

AI-Powered Chatbot for Hint Assistance

2.5.1.1. Description and Priority

Implementation of an AI-based chatbot in our three-genre quiz game in Unity with C# is of High priority. This feature will contribute meaningfully towards user engagement by offering hints to help players answer tricky questions, thereby enhancing the overall user experience. The benefit factor is scored 8, since it increases user interaction and the chances of successful completion of levels. The cost factor is scored 6, taking into account the resources needed for AI chatbot development. The chances of risks of technical faults or user dissatisfaction due to the inclusion of the chatbot are fairly low because it has been comprehensively tested and user feedback deployed. Generally speaking, the AI chatbot feature is value-enhancing to the educational content of the game as well as for user satisfaction.

2.5.1.2. Stimulus/Response Sequences

The user action and system response sequences that trigger the behavior specified for this feature in our project, a three-genre quiz game created in Unity with C#, include an interactive and learning experience for users. Users interact by choosing a category (Mathematics, General Knowledge, English), providing answers to questions, and receiving feedback on answers. The chatbot offers suggestions and guidance as players get stuck on challenging questions, making user interaction more engaging. As levels are navigated, the game automatically adjusts difficulty depending on performance, ensuring that the game remains challenging yet fun. The fact that there are unlimited levels guarantees players are constantly engaged and encouraged to do better. Globally, our vision is to develop an interactive and smart learning platform that engages and enlightens users efficiently using a smooth user-system experience.

2.5.1.3. Functional Requirements

Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.

REQ-SF1-1: The chatbot should present proper hints as per the question difficulty level.

REQ-SF1-2: The game should monitor player performance in order to dynamically change difficulty.

REQ-SF1-3: The questions for the quiz should be divided into Mathematics, General Knowledge, and English.

REQ-SF1-4: The game should respond to error conditions in a graceful manner, giving good feedback to the user.

REQ-SF1-5: The AI chatbot should be capable of interpreting and answering natural language queries from the players.

2.5.2. System Feature 2

AI-Powered Chatbot Hint System

2.5.2.1. Description and Priority

The functionality of incorporating an AI-driven chatbot into our three-genre quiz game built with Unity and C# is of High priority. This functionality enriches user engagement by offering hints to help players answer challenging questions, thereby boosting player engagement and retention. The advantage of this functionality is an 8, as it strongly improves the user experience.

2.5.2.2. Stimulus/Response Sequences

In our project, a three-genre quiz game developed in Unity with C#, the user input and system output are so well intertwined that they evoke the desired behavior. While players interact with questions covering Mathematics, General Knowledge, and English categories, the system

automatically adjusts the difficulty level to keep the players engaged and challenged. The chatbot powered by AI intervenes to provide useful tips when players get stuck, making the gameplay experience smoother. Unlike other conventional quiz games, our project prefers unlimited levels to ensure that with each passing level, complexity rises gradually. By this engaging and interactive learning system, we wish to amuse and teach users in the most effective manner, making learning a fun experience.

2.5.2.3. Functional Requirements

Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.

REQ-SF2-1:

The quiz game should accommodate questions from three categories: Mathematics, General Knowledge, and English.

REQ-SF2-2:

The chatbot powered by AI should offer hints to help players answer tough questions in real-time.

REQ-SF2-3:

The game should change the level of difficulty in questions dynamically depending on the performance of the players to provide a challenging yet entertaining experience for users.

2.5.3. System Feature 3 (and so on)

The game provides an adaptive learning experience by monitoring player progress and varying question difficulty in response. This means that players are challenged and kept active at their best level of learning all the time. The game also features a full performance analytics package giving deep insights into each player's strengths and weaknesses in various categories. These analytics can assist players in finding areas of improvement and adapting their learning strategy

to optimize educational gain. By marrying adaptive difficulty levels with thoughtful analytics, the game promotes ongoing learning and improvement, and it is an excellent resource for both education and entertainment.

2.6. Nonfunctional Requirements

Some of the nonfunctional requirements for our three-genre quiz game project are performance efficiency to support seamless gameplay and responsiveness despite the game dynamically adjusting difficulty levels. Scalability is important to support unlimited levels and categories, integrating easily with user participation and progress. The reliability of the game is important to ensure seamless gameplay and chatbot support, improving user satisfaction and experience. Security features should be put in place to ensure user data and the integrity of the game world. Usability is also critical, with a focus on ease of use and simplicity of interface and instructions to provide an engaging and educational experience for users at all levels. Finally, maintainability and extensibility are necessary for ongoing updates and expansion to continually enhance the content and features of the game. Scalability will be useful for unlimited levels and categories, and it will integrate with user participation and progress seamlessly. The game should be designed with functionality that will enable additional subjects, question sets, and game attributes without interfering with previous functionality.

A reliable game is an imperative aspect that needs to be ensured for uninterrupted gaming and smooth functions on the chatbot. The system needs to have error-free responses and should be able to tackle unexpected inputs and interruptions.

The implementation of security features should be done with the intention of securing user information and ensuring that the integrity of the game world is kept intact. Examples include the safety of player profiles and gaming progress.

2.6.1. Performance Requirements

Performance Requirements of our three-genre quiz game project in Unity with C# include smooth embedding of AI chatbot for providing hints, scaling for unlimited numbers of levels with dynamically growing complexity depending on performance of the players, and achieving smooth user interactions during the whole game. Performance metrics will target ensuring rapid response times for interactions with the chatbot, optimizing the speed of level generation and difficulty adaptation, and reducing lag or delays to ensure continuous player engagement. Furthermore, the game should be efficient in processing a large pool of questions from diverse categories in order to provide a varied and challenging gameplay experience for players. The long-term goal is to provide a high-performance, user-friendly, and educational quiz game that surpasses player expectations in terms of entertainment and learning value.

2.6.2. Safety Requirements

To protect the players while playing the three-genre quiz game built in Unity through C#, there are some safety needs to be addressed. These include the use of age-friendly content for all the categories, a strong data privacy policy to safeguard players' information, and frequent updates in the game to cover any security flaws. Moreover, the deployment of the AI-driven chatbot must be strictly tested to ensure that it does not pose any harmful interactions with users. There is a need to offer detailed safety instructions on the game interface and urge users to report inappropriate content or behavior. By putting safety protocols at the forefront, the project will be able to establish a safe and fun space for users to interact with the game and absorb its educational material effectively.

2.6.3. Security Requirements

Security needs for the three-genre quiz game project that has been developed in Unity using C# include incorporating encryption technologies to secure personal user information, securing user authentication processes to safeguard against unauthorized users, performing regular security audits and penetration testing to discover and patch vulnerabilities, and incorporating

secure coding methodologies to help secure against SQL injection and cross-site scripting attacks. Moreover, installing user permissions and access controls to limit unauthorized activities, constructing an impenetrable network infrastructure to avoid data breaches, and compliance with appropriate data protection laws are essential to protecting the integrity and confidentiality of the game and users' details. By giving topmost consideration to security features during the development and upkeep process, the project can offer a secure and reliable platform for players to have fun with the interactive and learning experience without sacrificing their personal information or privacy.

2.6.4. Usability Requirements

Usability specifications for our three-genre quiz game project designed in Unity with C# encompass user-friendly navigation for a smooth gaming experience, obvious and concise descriptions for players to comprehend game goals without difficulty, aesthetically pleasing interface design with interest-provoking graphics and animations, responsive feedback mechanism to present instant responses to player input, flexible difficulty levels that automatically change according to player performance to provide a challenging yet fun experience, and effective incorporation of the AI-driven chatbot for offering hints as and when required without interrupting the course of the game. These requirements aim to create a user-friendly and immersive environment that promotes learning and entertainment simultaneously.

2.6.5. Reliability Requirements

Reliability is an essential element of our three-genre quiz game application created in Unity with C#. The functionality of the game has to be rock-solid and reliable at all times to provide an uninterrupted player experience. The AI chatbot intended to offer suggestions and improve user engagement should be running smoothly without glitches or interruptions. Given that the game has limitless levels that change every time in terms of difficulty, it is crucial that the progression system is consistent and shows player performance accordingly. To provide a smooth experience on varied platforms and devices, it is essential that the game does not have any inconsistency. Reliability specifications cover not just the technological factor but also the

general user satisfaction, working towards the aim of the project to develop an interactive and smart learning platform that is both enjoyable and informative.

2.6.6. Maintainability/Supportability Requirements

The Maintainability/Supportability Requirements for our three-genre quiz game project implemented in Unity with C# involve making the game maintainable and supportable to be easily updated and enhanced in the future. This entails organized and documented code, modular design patterns to support future expansions, and easy guidelines for troubleshooting and bug fixing. Furthermore, routine testing and feedback systems will be in place to solve any problems quickly and provide a smooth user experience. Ongoing support and maintenance of the AI chatbot feature will be essential in order to assist players with useful tips and information. The intention is to build a strong platform that can respond to changing user requirements and advancements in technology, keeping the game exciting and informative in the long term.

2.6.7. Portability Requirements

The portability needs of our project, a three-genre quiz game created in Unity with C#, involve making it run smoothly on various operating systems and devices. It must be compatible with Windows, macOS, iOS, and Android platforms to make it more accessible to people. The assets of the game, such as questions, animations, and the AI chatbot, need to be optimized for fast loading and smooth play on both mobile phones and computers. To do this, the code for the game should be organized well, using platform-independent coding habits and following Unity's best practices for portability. By putting portability first, we are hoping to get our interactive and educational quiz game in front of users of a wide variety of devices, thus increasing its impact and reach. The system supports structured interactions between the players and the game environment through defined use cases. Each of these use cases explains in detail how the user will interact with menus, quiz attempts, ask for hints, display results, and progress through dynamically generated levels. These use cases also help point out the role of the player, the AI chatbot, and the quiz manager module within the system.

2.6.8. Efficiency Requirements

Efficiency is important in the creation of our three-genre quiz game project in Unity with C#. For smooth gameplay experience, the game must load quickly, contain little lag, and perform well on different devices. The chatbot powered by AI must offer timely and precise hints to guide players effectively. The code for the game should also be performance-optimized to manage the infinite levels and dynamic difficulty adjustments effortlessly. The utilization of effective algorithms in question generation, player scoring, and level advancement is central to the upkeep of user interest and provision of a seamless learning process. In all, our aim is to develop a high-performance and interactive learning game that provides entertainment as well as learning effectively.

2.7. Domain Requirements

The requirements of our three-genre quiz game built in Unity with C# include a wide variety of questions from Mathematics, General Knowledge, and English topics to keep players entertained in an educational way. The inclusion of an AI chatbot for hinting is expected to make user interaction better and help players solve difficult questions. With infinite levels that adapt difficulty dynamically according to player performance, the aim is to develop an extremely interactive and smart learning environment. This project aims to educate and entertain users in an efficient manner by providing a novel quiz game experience that transcends conventional forms. The system supports structured interactions between the players and the game environment through well-defined use cases. Each use case defines in detail how a user will navigate through menus, attempt quizzes, request hints, review results, and progress through the game through dynamically generated levels. These use cases help in clearly demarcating the roles of the player, the AI chatbot, and the quiz manager module within the system. Documentation of these interactions at the design level ensures smooth player experience and consistency in functionality and handling of player inputs through the length of game playing.

Chapter 3

Use Case Analysis

Chapter 3: Use Case Analysis

Use case analysis is an important chapter that focuses on the concrete interactions between the users and the quiz game. It describes the different situations where the players would interact with the game, from initializing a new game to taking hints from the AI chatbot. This chapter establishes the different roles of users, including players and administrators, and their related actions in the game. By charting out these use cases, game developers can assure that the game caters to the varied demands and requirements of its users. Furthermore, the analysis assists in recognizing any areas of gaps or inefficiencies within the game design, resulting in a more organized and user-centric gaming experience.

3.1. Use Case Model

In our project, the use case model includes a range of interactions and functionalities intended to promote user experience and engagement. Players can initiate a new round of quiz, choose the preferred categories, and respond to questions in the game interface. The AI-based chatbot functionality provides hints and tips to guide players when necessary, introducing a layer of intelligence into the game. As players move on to higher levels, the level of difficulty dynamically changes depending on how well they perform, providing a rewarding but challenging experience. The unlimited levels option provides players with the opportunity to keep challenging themselves in the three genres featured in the game indefinitely. In general, the use case model is aimed at developing an interactive and learning platform that provides a fun learning experience for users while testing their knowledge of Mathematics, General Knowledge, and English.

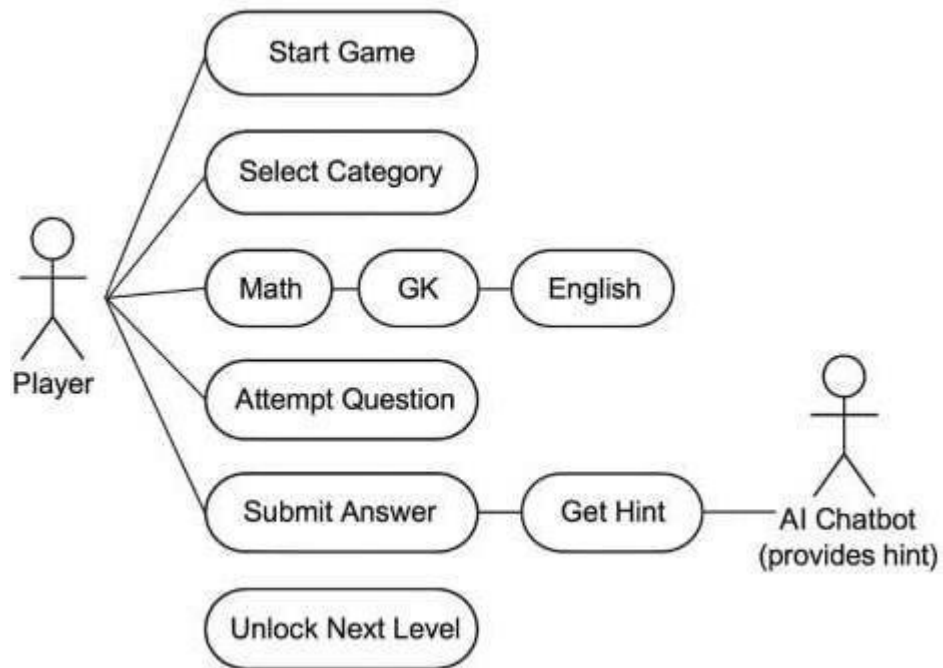
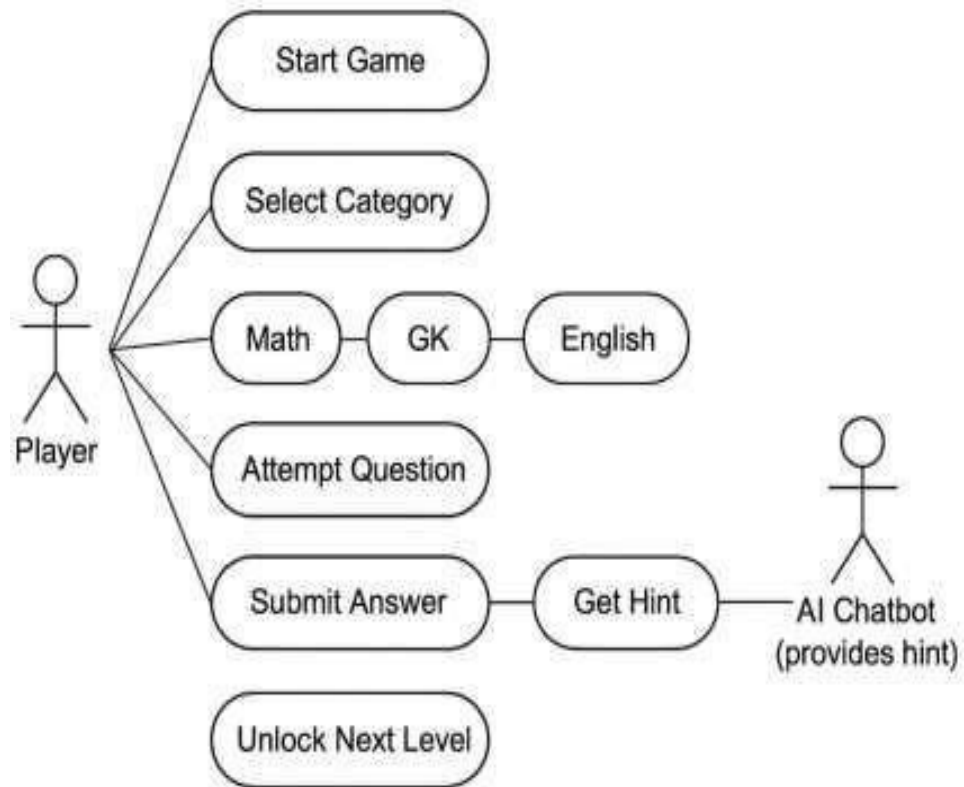


Figure 3 Use Case Model

3.2. Use Cases Description

The game allows players to select various difficulty levels and game types, test their knowledge, and monitor their performance in each type. The AI chatbot provides custom hints to the player based on their expertise, and the game is fun and easy to learn. Unlimited levels and dynamically adapting difficulty ensure that the game provides a fun yet challenging experience for players regardless of their expertise. This project seeks to revolutionize the quiz game category through the blending of education and entertainment. The system supports structured interactions between players and the game environment through well-defined use cases. Each use case describes how the users will navigate the menus, attempt quizzes, receive hints, view results, and progress through dynamically generated levels. Such use cases have been of immense help in clearly demarcating the roles of the player, the AI chatbot, and the quiz manager module within the system. Documentation of such interactions will guarantee smooth player experience, consistency in the functionality provided, and efficiency in handling player inputs at all phases of gameplay.



Chapter 4

System Design

Chapter 4: System Design

The system design chapter describes the three-genre quiz game's architecture and functionality that was built in Unity with C#. It covers the technicalities of how the game functions, such as the implementation of the AI-driven chatbot and the sorting of questions. The user interaction flow and how the game dynamically adjusts difficulty levels according to player performance are also covered in this chapter. Moreover, the chapter also identifies the unlimited levels aspect and its incorporation to present an uninterrupted and stimulating experience to users. On a whole, the system design chapter is focused on presenting an entire picture of how the game operates and how it presents an interactive and insightful learning platform for consumers.

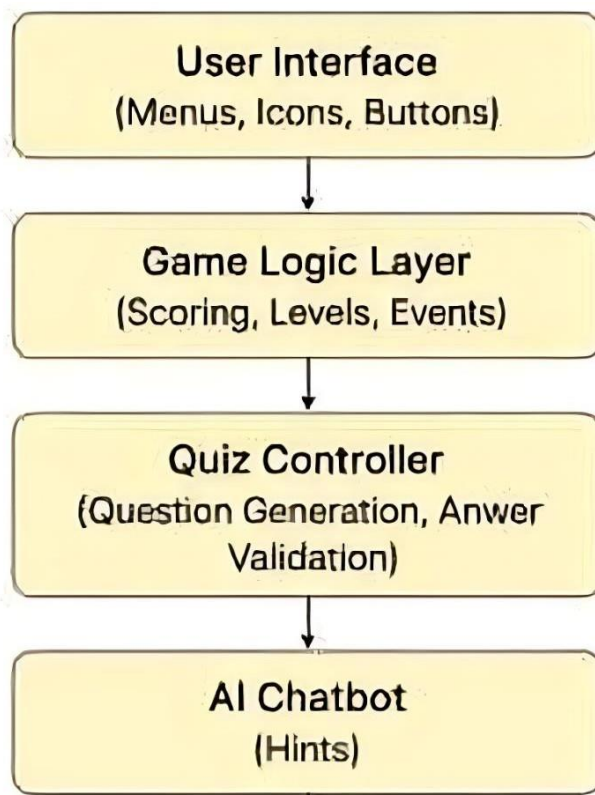


Figure 4 System Design

4.1. Architecture Diagram

The Unity game engine is the foundation, with C# scripts used to implement game logic and AI integration. The UI design is centered on user-friendly navigation and intuitive gameplay experience. The AI chatbot feature uses natural language processing to offer intelligent hints and guidance to players throughout the quiz game.

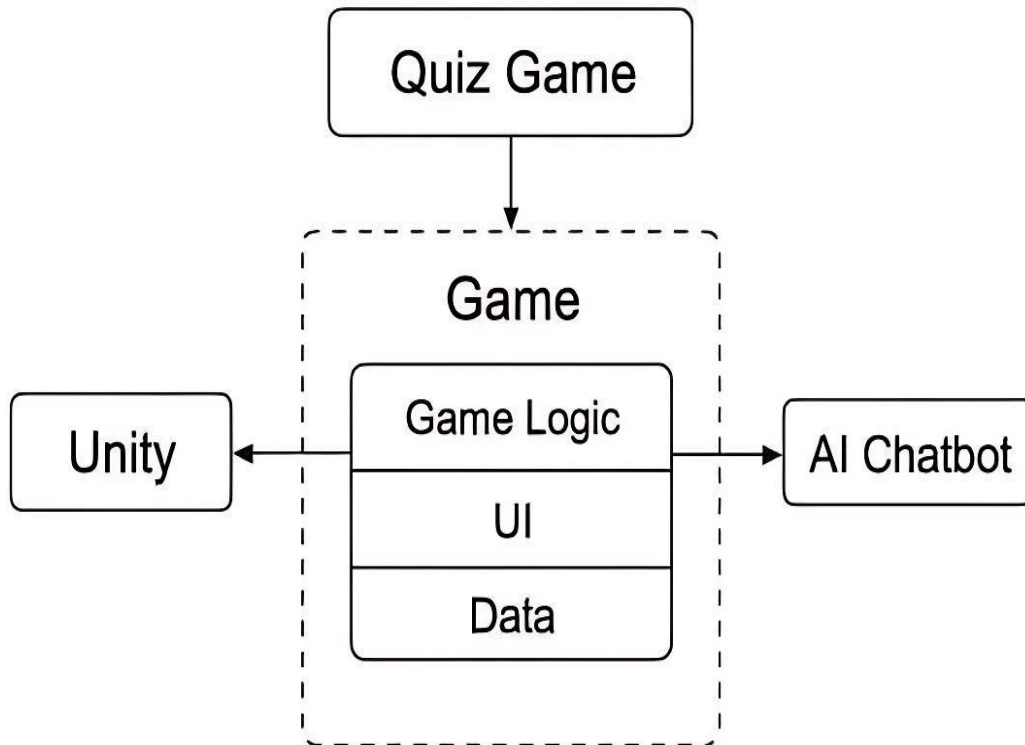


Figure 4.1 Architecture Diagram

4.2. Domain Model

The domain model of our project includes three vital parts: the quiz game itself, the question categories, and the chatbot based on AI. The quiz game is the central part, intended to test players with questions from Mathematics, General Knowledge, and English categories. Each category has a collection of questions to evaluate the player's knowledge and skills. The chatbot based on AI is a facilitative feature that gives players hints and guidance when they face tough questions. The game architecture supports infinite levels so that the level of difficulty adjusts dynamically according to the performance of the player. The model is working towards developing an interactive and smart learning system in which entertainment and education blend perfectly together to give users a fulfilling experience.

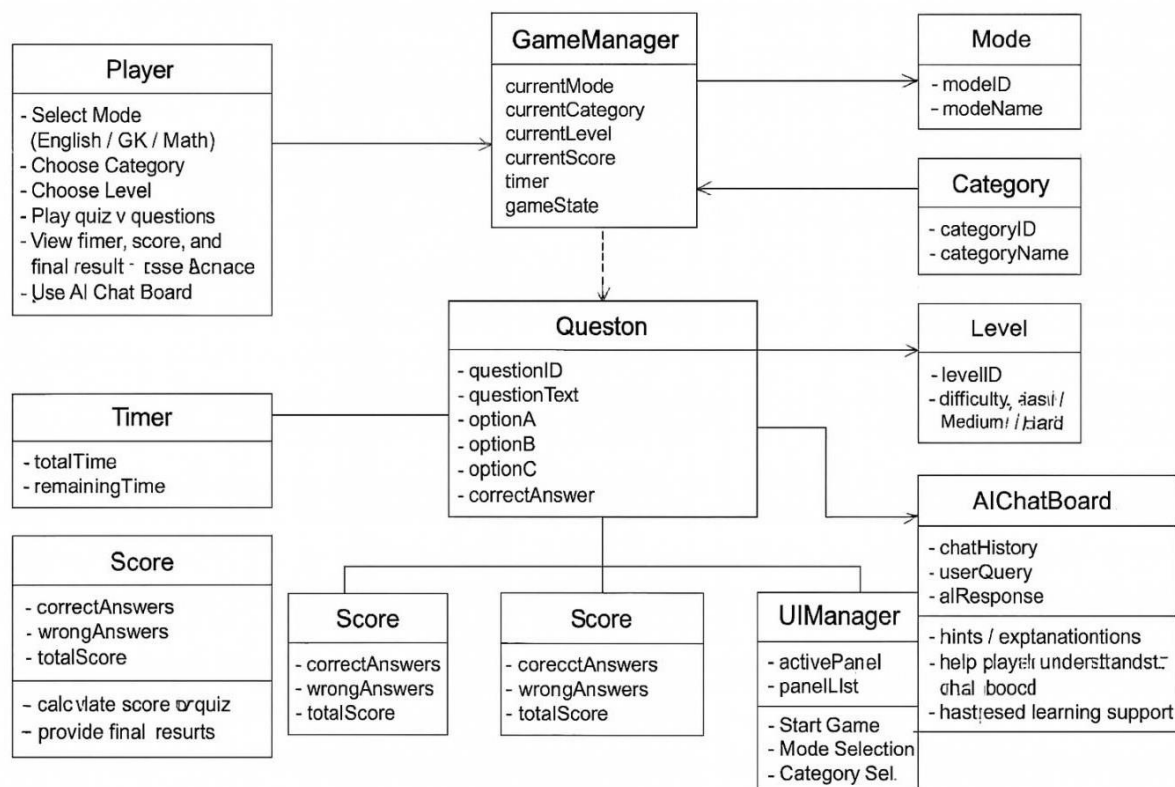


Figure 4.2 Domain Model

4.3. Entity Relationship Diagram with data dictionary

In our three-genre Unity quiz game, developed with C#, we have incorporated an Entity Relationship Diagram with a data dictionary that will make the management of data easier. This diagram shows relationships between players' questions, categories, and levels for a well-structured database that will assist in efficient game operations. Our game will involve three different categories: Mathematics, General Knowledge, and English, ensuring a well-rounded quiz experience for all participants. The incorporation of the AI-powered chatbot allows for hints for the player on challenging questions, thereby enhancing user interaction and learning. We also intend to provide unlimited levels with dynamically adjusting difficulty based on the performance of the player to ensure an interactive, intellectually stimulating gaming environment. Our ultimate objective is to create a quiz game that will not only entertain users but educate them effectively, hence promising an engaging and enriching experience for players of all ages.

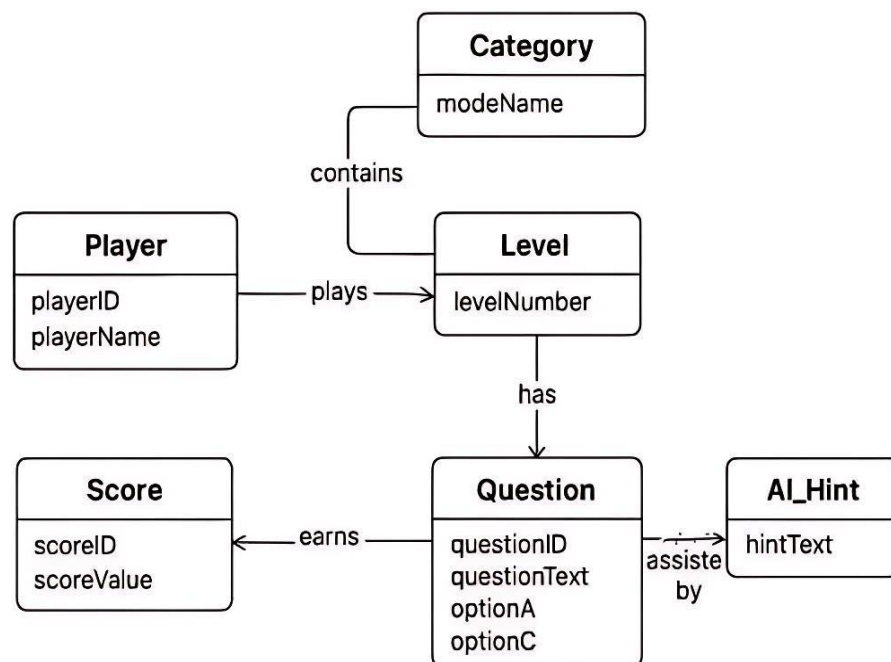


Figure 4.3 ERD

4.4. Class Diagram

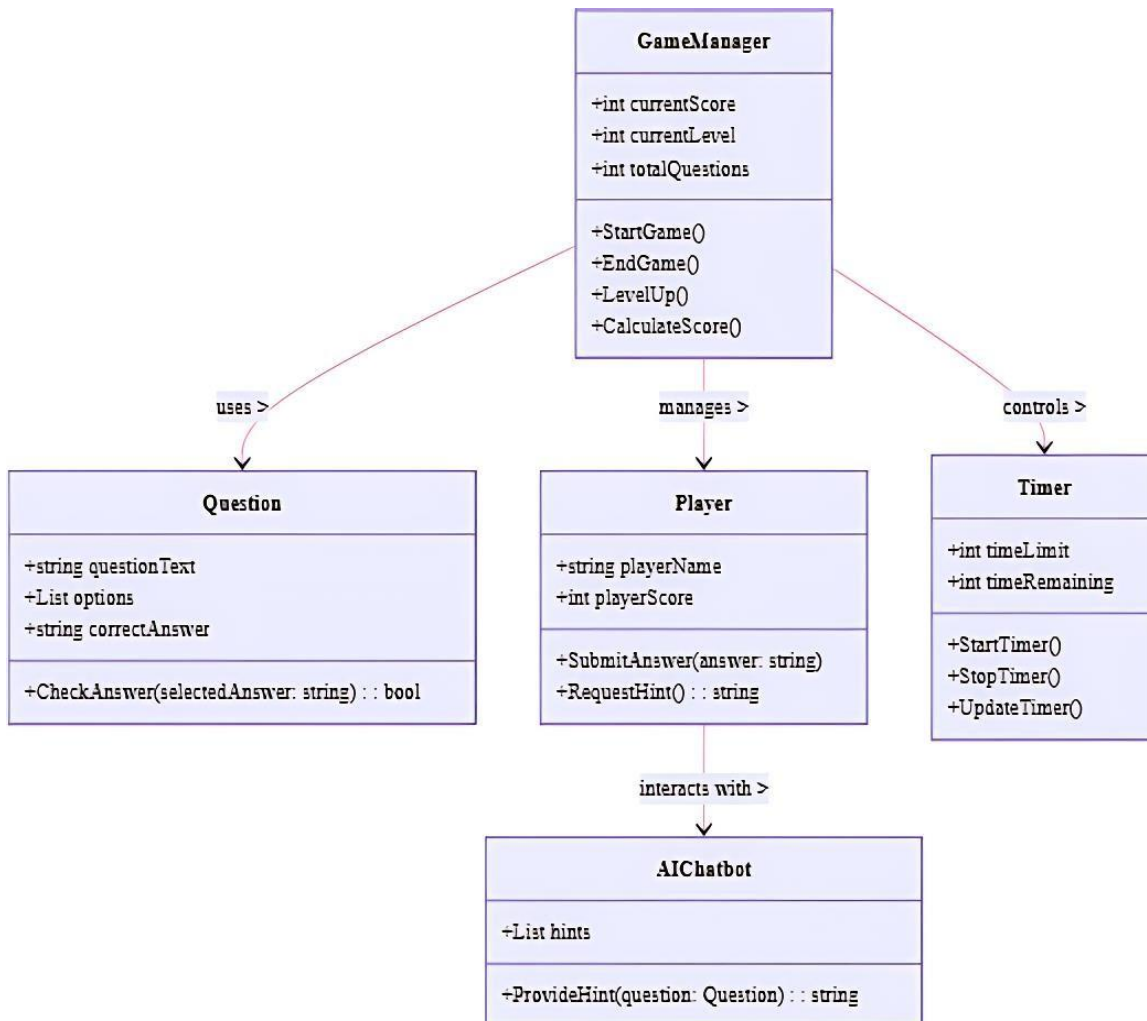


Figure 4.4 Class Diagram

4.5. Sequence / Collaboration Diagram

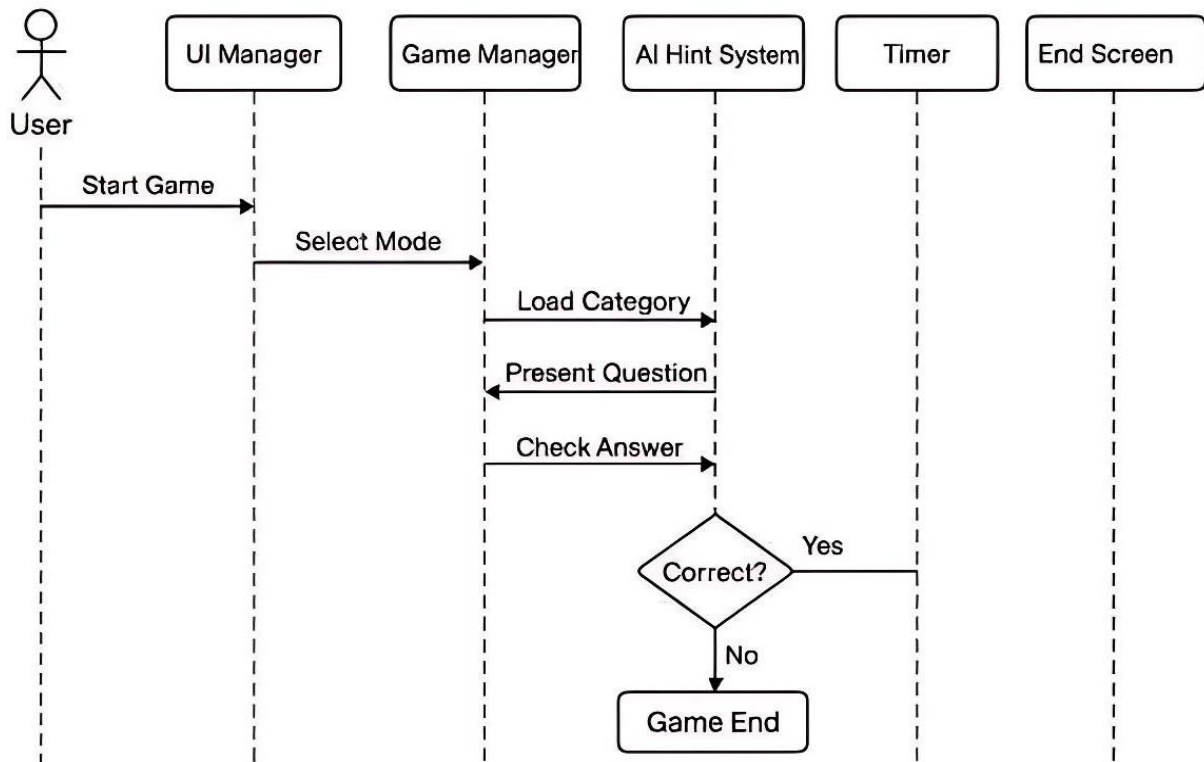


Figure 4.5 Sequence / Collaboration Diagram

4.6. Operation contracts

The Operation contracts for our three-genre quiz game project in Unity with C# will specify the exact behaviors and functionalities of our game elements. These contracts will explicitly state the input parameters, expected results, preconditions, and postconditions for every operation in the game, making it consistent and reliable. For example, the contract for the quiz question generator operation will state that it accepts a category as input and produces a valid question from the category as output. Likewise, the contract for the player answer validation module will state the conditions for checking the correctness of player answers and giving proper feedback. By strictly defining these operation contracts, we can make our game work the way we expect it to, preserve data integrity, and provide a smooth user experience.

4.7. Activity Diagram

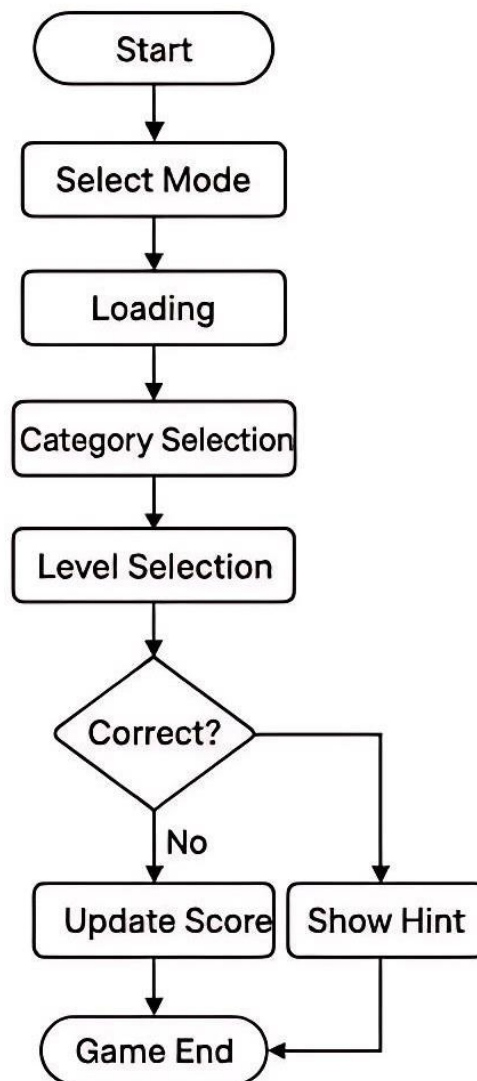


Figure 4.7 Activity Diagram

4.8. State Transition Diagram

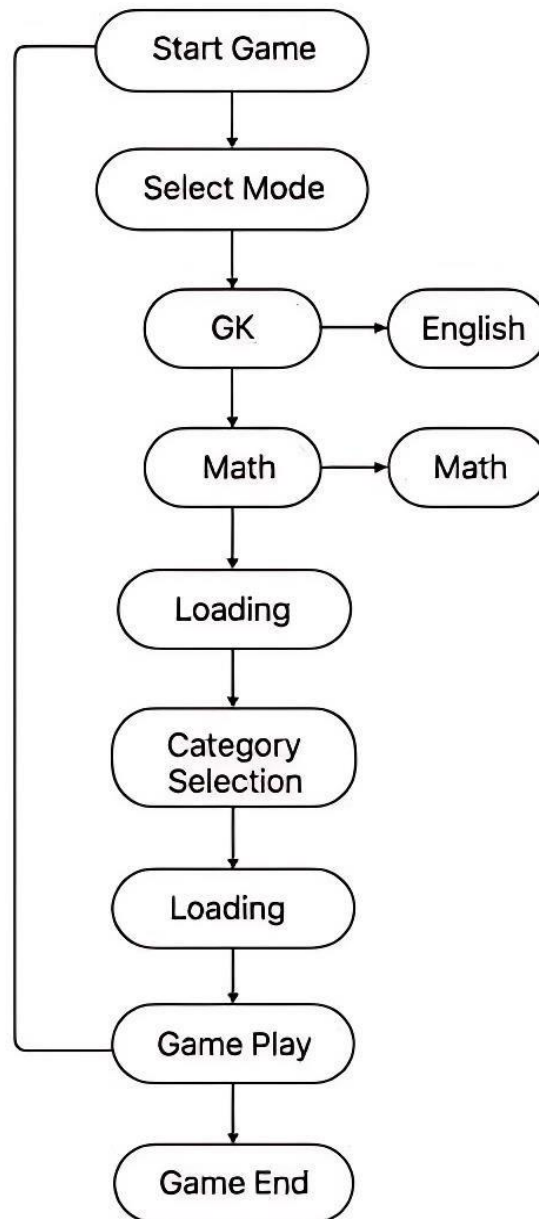


Figure 4.8 State Transition Diagram

4.9. Component Diagram

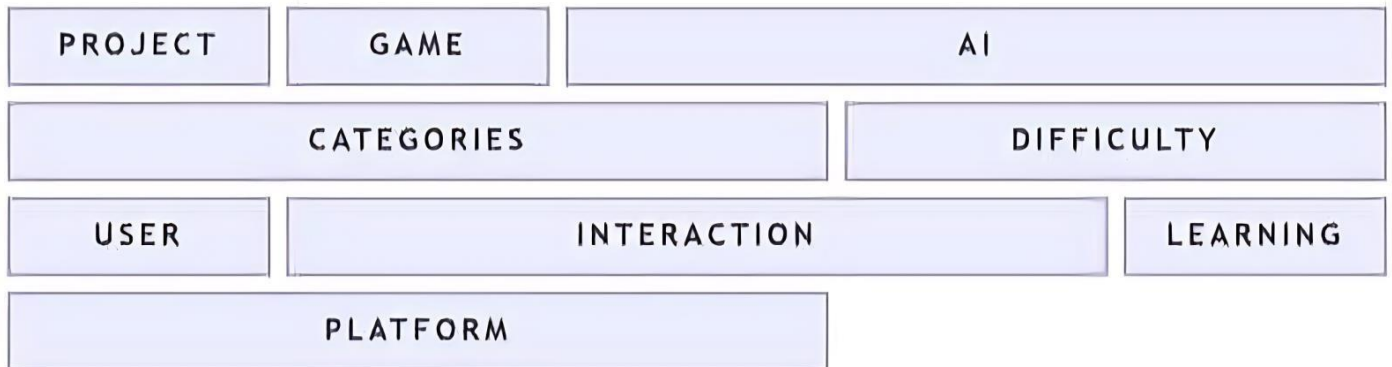


Figure 4.9 Component Diagram

4.10. Deployment Diagram

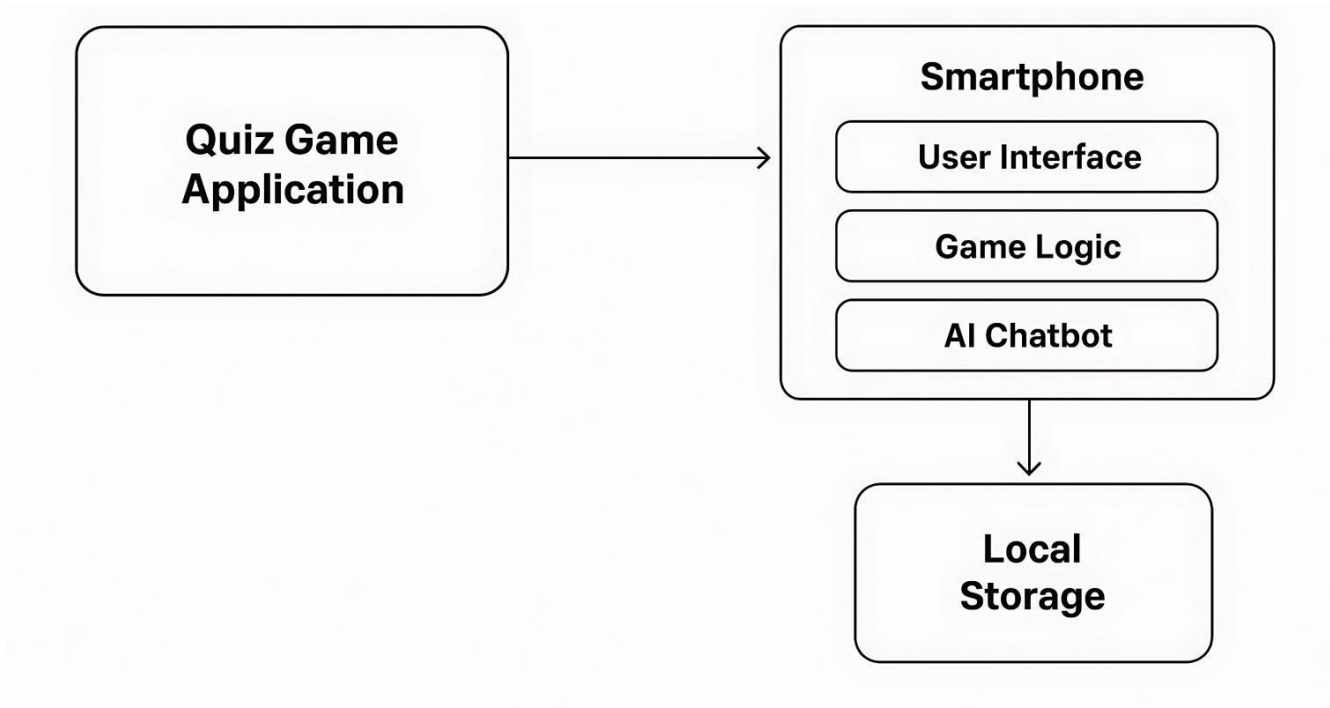


Figure 4.10 Deployment Diagram

4.11. Data Flow diagram

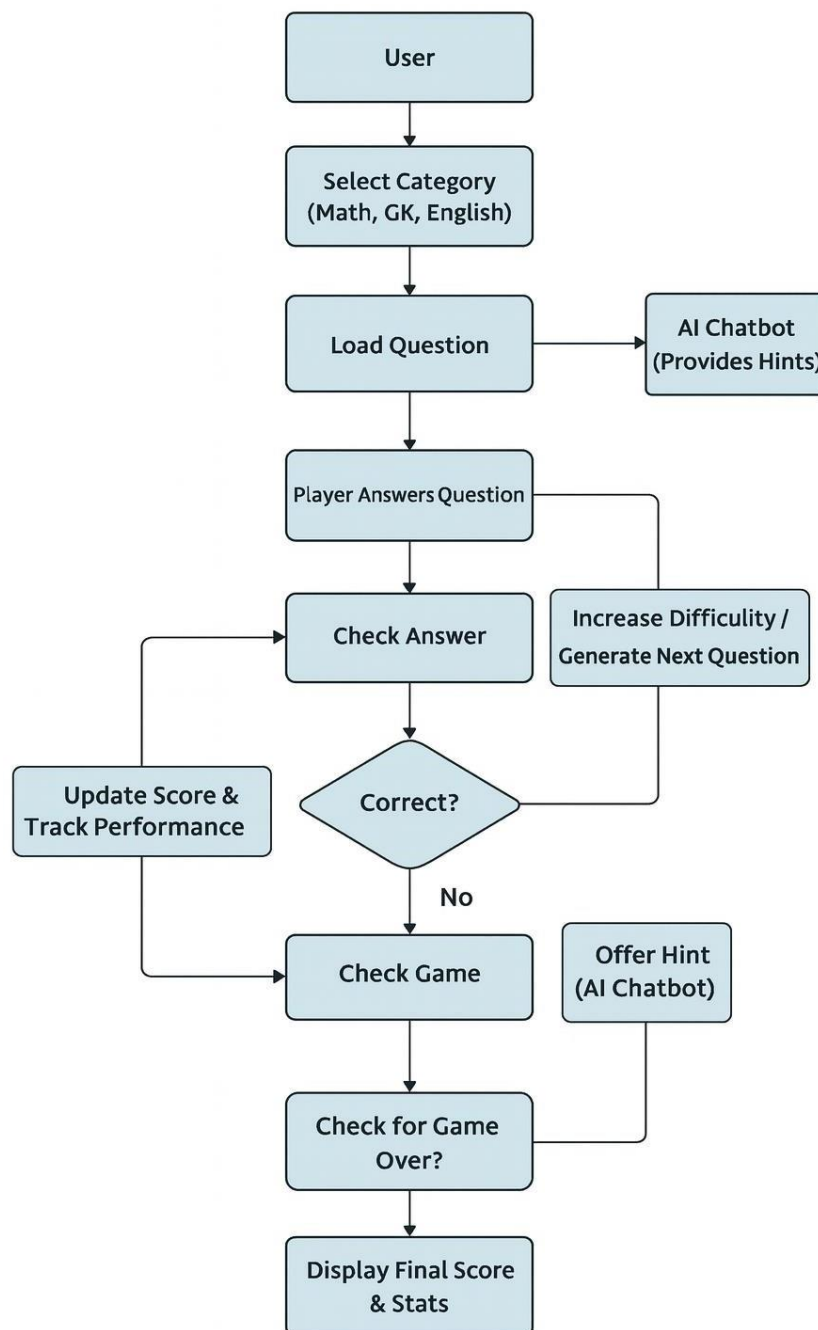


Figure 4.11 Data Flow diagram

Chapter 5

Implementation

Chapter 5: Implementation

The Implementation chapter explores the real-world implementation of our three-genre quiz game developed in Unity through C#. It describes the process of combining questions of various categories such as Mathematics, General Knowledge, and English to offer a challenging and learning-oriented gameplay experience. The chapter also addresses the implementation of an AI-driven chatbot that is capable of providing hints and assistance to players when they are stuck with difficult questions. The chapter highlights the special aspect of unlimited levels and adaptive difficulty, which changes dynamically from the performance of individual players. Overall, the aim of this chapter is to illustrate how all these factors combine to form an interactive and smart learning environment that seeks to engage and educate users efficiently.

5.1. Important Flow Control/Pseudo codes

In the flow control of our three-genre quiz game project implemented in Unity with C#, we have adopted important pseudo codes to make gameplay smooth. The game uses conditional statements to guide players to questions from various categories like Mathematics, General Knowledge, and English. Through the use of loops, we make sure that players are able to go through levels without a hitch, with questions becoming increasingly difficult. The introduction of functions enables the incorporation of an AI-driven chatbot that provides tips to players who are stuck on challenging questions. Through these flow control features, we hope to achieve a fun and instructional experience for users, with adaptively changing difficulty levels that respond to personal performance. Our long-term vision is to establish an interactive and smart learning platform that amuses users while optimizing knowledge gain.

5.2. Components, Libraries, Web Services and stubs

We used a common tool to develop our game and that tool is “Unity” on which millions of games have been made.

Unity is a best tool to develop any game for Android. And it is easy to develop a game on Unity.

The other tool is “Visual Studio” on which we write the scripts and codes for our game.

Tools:

Unity 3D.

Unity Hub.

Visual Studio Code.

Techniques:

We will use C# techniques and 4 pillars of OOP.

Encapsulation

Inheritance

Polymorphism

Abstraction

C# is a best and easy programming language for game development. That is why, we use C# for our game.

Website:

Unity Asset Store

Mixamo

Pixabay

5.3. Deployment Environment

We have written our code so neat and clean that if any programmer opens the code of our game, he will easily understand our code that what we have written where. This is the standard of our game.

Naming Conventions.

Indentation and Formatting.

Comments.

Function Length and Modularity. Error

Handling.

Object-Oriented Principles.

File Organization.

Performance and Optimization.

Version Control and Collaboration.

Testing.

5.4. Tools and Techniques

Our game project is developed using Unity as the game development environment and C# as the programming language. We apply different tools and methods to create the three-genre quiz game. In the quiz aspect, we have a dynamic question database system per category, such as Mathematics, General Knowledge, and English. To enhance the gaming experience of the player, we have an AI-driven chatbot offering useful hints throughout the game. The game has a level-up system that adapts the difficulty level according to player performance, providing an engaging yet rewarding experience. By providing unlimited levels, we want to make players interested and willing to learn more. Our project integrates creativity, technology, and learning content to provide an interactive and smart learning platform that educates and entertains simultaneously.

5.5. Best Practices / Coding Standards

In order to ensure consistency and efficiency in implementing our three-genre quiz game in Unity with C#, we implement the following coding standards and best practices. First, we use clear and descriptive naming conventions for classes, functions, and variables to make the code more readable. Second, we implement a modular and object-oriented structure to ensure code

reusability and maintainability. Thirdly, we adhere to the solid principles to create strong and agile components that are simple to extend or change. Furthermore, we apply efficient error handling and logging mechanisms to diagnose problems and enhance debugging capabilities. Finally, we perform regular code reviews and include feedback to maintain code quality and encourage collaboration among the development team. By following these best practices and coding guidelines, we believe that we can provide a good-quality and interactive quiz game that offers an entertaining and educational experience to our users.

5.6. Version Control

In the realm of our Version Control, is an interesting expedition to the development of a three-genre quiz game meant to entertain and educate users at the same time. Built in Unity with C#, this groundbreaking project provides a mix of Mathematics, General Knowledge, and English questions to test players in an entertaining way. What makes this game unique is the incorporation of a chatbot powered by AI, ready to offer a helping hand in the form of hints for difficult questions, thereby improving the overall user experience. With unlimited levels with dynamically adjusting difficulty levels according to the player's capability, this project seeks to revolutionize conventional quiz games by providing an interactive and smart platform for learning. Plunge into the universe of entertainment and learning, where a complete experience awaits users who desire to be entertained as well as to learn.

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

Testing and Evaluation in our project is geared toward ensuring the functionality and user experience of the three-genre quiz game developed in Unity, using C#. This chapter involves testing the functionality of the game, its performance, and the integration of the AI-powered chatbot. Testing and evaluations shall be conducted to identify issues or bugs that can hinder or affect game performance and the player's experience. Additionally, the learning value in the game and effective testing to educate players in the different categories shall be considered. The goal of this chapter is to fine-tune the quiz game for seamless and enriching experiences in entertainment coupled with learning in a dynamic and interactive manner.

6.1. Use Case Testing

Use Case Testing is an important approach to be incorporated into our project for the smooth functioning of our three-genre quiz game using Unity with C#. Testing numerous different scenarios concerning the use of the AI-powered chatbot to receive hints about certain questions will determine whether or not it is truly useful for users during gameplay. Use Case Testing will let us verify if, according to the player's performance, the game automatically adjusts its difficulty level to provide a challenging but rewarding experience. This will definitely allow us to deliver an interactive, intelligent learning platform that engages a user with appealing content and at the same time gives valuable educational insights into Mathematics, General Knowledge, and English.

6.2. Equivalence partitioning

Equivalence partitioning is an important part of our three-genre quiz game project, whereby precision in categorizing user inputs is of utmost importance if seamless gameplay is to be ensured. This Unity game, developed in C#, ensures that each question pertains to one category, be it Mathematics, General Knowledge, or English, to ensure the integrity of the quiz experience. By partitioning user responses effectively, we can give feedback that is accurate and provide hints relevant to the AI-powered chatbot and adjust the gameplay to the skill level

of the player. This meticulous approach enhances user engagement and ensures a smooth and educative journey across endless levels with challenging quizzes. We are committed to delivering an interactive, intelligent learning platform-and changing ordinary gameplay into an enriching and rewarding experience for all users.

6.3. Boundary value analysis

Boundary value analysis is one of the most important testing techniques that we will apply to our dynamic three-genre quiz game in Unity using C#. This novel project presents a unique gaming experience by merging Mathematics, General Knowledge, and English questions to take players on an entertaining and educative ride. Besides the variation in quiz topics, players will also be assisted by an AI-powered chatbot to help them get through challenging questions. What really makes it different is the 'infinite levels' feature where the difficulty intensifies or relaxes depending on the progress of a player, hence always keeping him challenged. Our aim is to provide an interactive, intelligent learning platform that entertains and educates players effectively, making learning a delightful adventure.

6.4. Data flow testing

Data flow testing, we apply an all-inclusive approach in ensuring information transition within our innovative three-genre quiz game is smooth. The game was designed using Unity and C# and combines entertainment with education in challenging players through questions in the fields of Mathematics, General Knowledge, and English. We focus on data flow testing beyond traditional analysis by incorporating AI into improving user experience through an interactive chatbot to always try to help players with hints to find the answer. With unlimited levels that adjust according to the performance of every player, we aim at developing a cutting-edge learning platform that will be entertaining yet educating. We prioritize data flow testing to ensure a seamless and enriching gaming experience for the users.

6.5. Unit testing

We can propose something different in unit testing for our Unity project of a three-genre quiz game, having unique features and goals in mind. In the testing of the game functionality, emphasis will be placed on the generation and presentation of questions in Mathematics, General Knowledge, and English categories. It will also include hints given by a chatbot that should be helpful in enhancing the experience of players. There will also be scenarios of unlimited levels and dynamic adjustment of difficulty while considering the performance of players, which will make gameplay smooth and challenging. Comprehensive unit testing will, therefore, ensure the reliability and efficiency of this interactive and intelligent learning platform, raising the bar higher for innovation in quiz game development.

6.6. Integration testing

Integration testing is an indispensable process in verifying the smooth operation of our innovative three-genre quiz game developed in Unity using C#. Due to the unique blend present in this project, which involves Mathematics, General Knowledge, and English categories, thorough testing has to be done on the integration of AI-powered chatbot features. It can be assured that players get accurate hints and feedback through simulated interactions between different components and, thus, enhance gameplay. With unlimited levels and dynamically adjusting difficulty levels, our aim is to provide users with an interactive and intelligent learning platform, which not only entertains but also educates effectively. Rigorous integration testing aims to deliver a flawless and engaging quiz game, offering a new benchmark for user experience and educational value.

6.7. Performance testing

In the performance testing of this ambitious three-genre quiz game project in Unity with C#, we are confident to guarantee seamless game experiences that stand by our promise of a fun, interactive, and intelligent learning environment. The game invites players to take part in a

variety of questions about Mathematics, General Knowledge, and English in a spirited quest for entertainment and education. To further enhance the experience for players, there is an advanced AI chatbot ready to provide helpful hints for questions that might be a bit more challenging to solve. One of the distinguishing features of this project involves it having endless levels, with its difficulty adjusting automatically as per individual performance. Our ultimate objective is to craft a stimulating platform that entertains and enlightens, transcending the boundaries of traditional quiz games.

6.8. Stress Testing

Stress Testing for the three-genre quiz game project involves simulating extreme user loads and challenging scenarios to ensure that a Unity-C#-developed game is robust and stable. By putting the game in conditions of high stress, we aim to find possible bottlenecks, performance problems, and weak points in the system that may influence the user experience. The Stress Testing phase will determine how heavy traffic, AI chatbot interactions, or dynamic level generation is handled by the game and whether or not it will scale appropriately to user demands without functional compromise. We are committed to delivering a reliable and seamless quiz game through rigorous testing for the entertainment and education of its players.

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

The proposed project is a unique kind of three-genre quiz game designed in Unity using C#. Players will play this thrilling trivia game that covers questions from categories like Mathematics, General Knowledge, and English. What makes us unique is the AI chatbot we have integrated into this game, which provides valuable hints to the players for solving difficult questions. Unlike other quizzes, the levels are unlimited and dynamically change as the proficiency of the players gets better or worse. Our ultimate goal is to provide an interactive and astute learning tool that entertains while educating users on the go.

7.2. Achievements and Improvements

Our three-genre quiz game has reached incredible heights through innovative design and further development. With the power of Unity and C# as driving technologies, we've created an engaging gameplay experience that successfully combines learning with entertainment. By challenging players in the categories of Mathematics, General Knowledge, and English, we have been able to bring about a stimulating learning environment that brings out curiosity and intellectual growth.

The inclusion of an AI chatbot in the project adds value to each participant by providing personalized hints and guidance, making the user experience richer and problem-solving easier. Unlimited levels are generated according to the performance level of an individual for a smooth, progressive journey and ensure a challenging but rewarding experience. A truly interactive and intelligent learning platform is our dream come true; it embarks on a holistic approach toward education through immersive gameplay.

7.3. Critical Review

Until today, it was a pleasure to dive into the world of the three-genre quiz game designed and developed in Unity using C#. Evidently, its developers were able to think outside the box. With categories as varied as Mathematics, General Knowledge, and English, the experience would indeed be both challenging and intellectually engrossing. Its integration with an AI chatbot would enhance the user experience with personalization through hints or other forms of hints or assistance. The concept of unlimited levels, whose difficulty levels are progressing accordingly with or against the proficiency of the players, stands at a completely different angle from that of a traditional quiz game. This is indeed an interactive and dynamic learning platform. Overall, entertainment and education are perfectly combined in this project to fulfill its missions and set new standards for quiz games within the gaming industry.

7.4. Lessons Learnt

Our three-genre quiz game has proven to be a unique gem in the game development arena. While developing this project using Unity with C#, we learned how effectively an AI-powered chatbot could be embedded into the game for player engagement and learning. Offering hints and dynamically modifying difficulty levels have opened a new dimension toward interactivity and intelligence in the gaming experience.

Looking back, we realize that the essence of innovation lies in pushing beyond the boundaries and challenging convention. Our game is a perfect example of how entertainment can combine with education seamlessly, not only to entertain but also to enlighten. The important thing we learned throughout the whole process is that it isn't about building a game but about creating an experience for your players, enriching and endearing.

7.5. Future Enhancements/Recommendations

Going forward, we have many ways to enhance the three-genre quiz game user experience. Ideas for improvement could include adding multiplayer capability so that players can compete in real-time quizzes against friends or other online users. Another great feature could be a

function that allows players to create and share custom quiz questions, which would increase user engagement and build a broader content library. It would also be a good idea to include a system to track the player's progress and give them personalized feedback and recommendations on how to improve. Through data analytics, we are able to further fine-tune the difficulty scaling algorithm to better reflect the documented performance ability of each player and provide an appropriately challenging yet fun experience for the users.

Appendices

Appendix A: Information / Promotional Material

This appendix gives a detailed overview of the different interactive elements of our unique three-genre quiz game designed in Unity with C#. The players will be thrilled to experience the richness of different categories, such as Mathematics, General Knowledge, and English, which guarantee an engaging and rewarding experience. The inclusion of an AI-based chatbot enhances the gameplay by providing useful tips to address difficult questions efficiently. Additionally, the game's innovative layout provides infinite levels that adapt difficulty dynamically to provide an enriching experience for every player based on their skill level. Learn how our project seeks to transform the quiz game category by merging entertainment with education, providing a rich platform that is both interactive and smart.

A.1. Broacher

In this brochure, we would like to invite you to discover our groundbreaking three-genre quiz game created in Unity with C#. Be immersed in a difficult playing experience crossing categories such as Mathematics, General Knowledge, and English. Make your learning process more superior with our AI chatbot, which is ready to provide hints and assistance as you face difficult questions. With an unlimited number of levels that learn and respond to your ability, this game defies conventions to deliver a responsive and exciting experience. Our aim is simple and potent: to present an interactive and smart platform that entertains and teaches, making each moment spent playing a step towards enrichment and development.

A.2. Flyer

Participate in the most exciting quiz game experience ever! Our project presents a novel three-genre quiz game implemented in Unity through the use of C#. With categories such as Mathematics, General Knowledge, and English, our game provides a difficult but informative gameplay. What makes us stand out is the inclusion of an AI chatbot that supports players with difficult questions, providing an entertaining learning experience. With infinite levels that evolve according to player ability, the game offers a dynamic and stimulating experience.

Immerse yourself in a world where learning and fun come together, and challenge your knowledge like never before!

A.3. Standee

The standee for our three-genre quiz game project is made visually engaging and informative. It has bright graphics depicting the various categories in the game - Mathematics, General Knowledge, and English. The standee has a QR code that takes users to download the game on their smartphones, thus becoming easily accessible. The subtitle "Challenge Your Mind, Expand Your Knowledge" is prominently featured to grab the attention of individuals passing by and communicate the learning aspect of the game.

This standee is strategically positioned in top-traffic locations to grab the attention of the target market - people eager to enjoy fun and learning gaming experiences. Its attention-grabbing design and engaging messaging are designed to entice interested players and create interest in the game's innovation and advantages.

A.4. Banner

Our project presents a revolutionary three-genre quiz game designed in Unity using C#. Our innovative game engulfs participants in an engaging quiz adventure that traverses Mathematics, General Knowledge, and English subjects. To promote maximum engagement, we have incorporated an AI-powered chatbot that provides beneficial hints to overcome tricky questions. Moving away from conventional quiz games, our project features unlimited levels that dynamically shift based on the skills of players. Our final aim is to create an interactive and smart learning device that amuses and instructs users with unprecedented efficiency. Our system also integrates features for handling data seamlessly, which include tracking player progress, adjusting levels of questions posed, and promoting a personalized learning curve.

.

Exploring the Boundaries of Interactive Learning

Our project weaves together the worlds of entertainment and learning with a fresh concept. By blending three different genres in a quiz game and integrating an AI chatbot, we have created an engaging and interactive experience for players. Every level offers a fresh challenge, responsive in real-time to the player's ability and knowledge. As learners make their way through, they open new avenues of learning, broadening their toolkit in Mathematics, General Knowledge, and English. This project dreams of a future where interactive technology not only delights but informs, building a greater appreciation for the pursuit of knowledge.

Improve User Experience with AI Integration:

Integrating Artificial Intelligence into our three-genre quiz game created in Unity with C# has transformed the manner in which the players experience the game. By implementing an AI-driven chatbot, we have offered players customized hints to solve difficult questions, thus improving the immersive and helpful nature of the gaming experience. This innovative concept differentiates our project from conventional quiz games, presenting unlimited levels dynamically responding to players' performance. Our aim is to create a state-of-the-art learning platform that not only engages but also teaches users efficiently, fostering interactive and smart gameplay.

Evolution of Learning Dynamics:

Here, we discuss the new approach adopted in designing the learning dynamics of our three-genre quiz game project. With the help of an adaptive difficulty system, the game automatically adjusts the level of difficulty based on the performance of the individual player. This makes sure that the players are always motivated and active, with constant incentives to enhance their knowledge base over various categories. The smooth incorporation of an AI-based chatbot also

enhances the learning experience by offering users personalized tips and advice when confronted with challenging questions. Through these advanced features, our project hopes to set a new standard for interactive and intelligent learning platforms, creating an enjoyable and productive means for users to increase their knowledge base while partaking in the gameplay.

Appendix no 2 :Development Process and Game Features:

Appendix herein discusses the nitty-gritty of our three-genre quiz game project in Unity developed with C#. The game design presents a challenging structure to players with diverse sets of questions covering Mathematics, General Knowledge, and English topics. With the integration of an AI-based chatbot, players are presented with useful hints to overcome difficult questions, encouraging a more interactive gameplay experience. The game stands out through the provision of unlimited levels with dynamically adjusted difficulty, depending on the player. Essentially, our main objective is to design a compelling and smart learning device that entertains users and transfers knowledge successfully.

Inspirational Title that Challenges Brains:

Our project is a distinctive three-genre quiz game built in Unity using C#. This revolutionary game engulfs players in a dynamic quiz setting covering subjects such as Mathematics, General Knowledge, and English, and provides a stimulating and interactive experience. With an added touch of finesse, we have incorporated an AI-based chatbot providing useful tips to assist users in solving tricky questions. Stepping away from traditional quiz schemes, our project features an endless progression system that dynamically changes difficulty levels depending on player competence. Our goal with this project is to create an interactive and smart learning platform that not only engages but also effectively teaches individuals.

Game Mechanics Overview:

Our innovative three-genre quiz game designed in Unity using C# provides the user with an interactive and interesting game play. This game falls under categories such as Mathematics, General Knowledge, and English. The game promises a challenging yet educating journey for the users. Equipped with an AI chatbot, we hope to provide users with valuable hints to properly tackle difficult questions. What distinguishes our project is the infinite levels it provides, adjusting difficulty levels dynamically based on the progress of the player. Our long-term vision is to create a smart and interactive learning platform that entertains and teaches users both at the same time.

Improved Gameplay Features Implementation:

Our team's project is a three-genre quiz game built within Unity through C#. The game presents players with questions within different categories, including Mathematics, General Knowledge, and English, to provide an immersive yet educational experience. For improved user interaction, an AI-based chatbot feature is implemented, offering hints to aid players in answering complex questions. In contrast to conventional quiz games, this project has unlimited levels that dynamically adjust the difficulty of gameplay according to the performance of players. The objective is to develop an intelligent and interactive learning environment that not just entertains but also educates the users efficiently. The improved gameplay feature implementation intends to deliver a smooth and engaging gaming experience with advanced functionalities that make learning enjoyable and rewarding.

