THE MAHARAJA SAYAJIRAO UNIVERSITY OF BARODA

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

सत्यं शिवं सुन्दरम्

A MINI PROJECT REPORT

ON

QuizBOT

SUBMITED BY

AAMIR NODHLA	**:**	**8022003407**
BADRINATH PANDEY	**:**	**8022009595**
DEV PATEL	**:**	**8022008148**
DHWANIT PANCHAL	**:**	**8021031365**
DHYEY PATEL	**:**	**8022020011**

IN PARTIAL FULLFILLMENT FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF COMPUTER APPLICATIONS

IN THE

DEPARTMENT OF COMPUTER APPLICATIONS

OCTOBER 2024

GUIDED BY

Ms. KRUPALI PANCHAL

TABLE OF CONTENTS

ACKNOWLEDGEMENT

We, the members of the project team, express our heartfelt gratitude to almighty for his grace and love showered on us and to all those who played an integral part in ensuring the successful completion of our project. We acknowledge that the contributions of each person involved were vital, and without their support, this accomplishment would not have been possible.

We extend our gratitude to the Head of Department Prof. P.K. Mehta for entrusting us with the opportunity to work on this project. His belief in our abilities and willingness to guide us has allowed us to showcase our knowledge and skills. We would also like to acknowledge the invaluable guidance provided by our project guide Ms. Krupali Panchal, her unwavering support, and timely feedback were crucial in the successful completion of this project.

Our Department of Computer Applications played a crucial role in enabling us to gain practical experience and understanding of the professional environment. This opportunity has not only enhanced our skills but also boosted our confidence for our future careers.

Last but not least, we express our deepest appreciation to each other for working collaboratively as a team, like the parts of a machine, to bring this project to function. Your commitment, dedication, and hard work have been crucial in achieving our goals, and we could not have done it without each other.

In conclusion, we thank everyone involved in this project and hope to continue working together in future endeavors. We acknowledge that the contributions of each person involved were vital, and we are grateful for their support and guidance throughout the project.

THE MAHARAJA SAYAJIRAO UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

CERTIFICATE

This is to certify that Mr. **AAMIR NODHLA** a student of **Bachelor of Computer Applications (**PRN**: 8022003407)** in the Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the Mini Project prescribed for the fifth semester of the program during the period from July, 2024 to October, 2024 towards the partial fulfilment for the degree of Bachelor of Computer Applications.

Date: 5/10/2024

Name & Signature of the Guide	Head, Department of Computer Applications

THE MAHARAJA SAYAJIRAO UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

CERTIFICATE

This is to certify that Mr. **BADRINATH PANDEY** a student of **Bachelor of Computer Applications (**PRN**: 8022009595)** in the Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the Mini Project prescribed for the fifth semester of the program during the period from July, 2024 to October, 2024 towards the partial fulfilment for the degree of Bachelor of Computer Applications.

Date: 5/10/2024

Name & Signature	Head, Department of
of the Guide	Computer Applications

THE MAHARAJA SAYAJIRAO UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

CERTIFICATE

This is to certify that Mr. **DEV PATEL** a student of **Bachelor of Computer Applications (**PRN**: 8022008148)** in the Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the Mini Project prescribed for the fifth semester of the program during the period from July, 2024 to October, 2024 towards the partial fulfilment for the degree of Bachelor of Computer Applications.

Date: 5/10/2024

Name & Signature of the Guide	Head, Department of Computer Applications

THE MAHARAJA SAYAJIRAO UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

CERTIFICATE

This is to certify that Mr. **DHWANIT PANCHAL** a student of **Bachelor of Computer Applications (**PRN**: 8021031365)** in the Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the Mini Project prescribed for the fifth semester of the program during the period from July, 2024 to October, 2024 towards the partial fulfilment for the degree of Bachelor of Computer Applications.

Date: 5/10/2024

Name & Signature of the Guide	Head, Department of Computer Applications

THE MAHARAJA SAYAJIRAO UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE

<u>CERTIFICATE</u>

This is to certify that Mr. **DHYEY PATEL** a student of **Bachelor of Computer Applications (**PRN**: 8022020011)** in the Department of Computer Applications, Faculty of Science, The Maharaja Sayajirao University of Baroda, Vadodara, has successfully completed the Mini Project prescribed for the fifth semester of the program during the period from July, 2024 to October, 2024 towards the partial fulfilment for the degree of Bachelor of Computer Applications.

Date: 5/10/2024

Name & Signature
of the Guide

Head, Department of
Computer Applications

SOFTWARE REQUIREMENT SPECIFICATION

1.1 INTRODUCTION

QuizBOT not only simplifies the process of creating quizzes but also offers a flexible approach to education. Teachers can select the difficulty level, topics, and number of questions they want to include, making the quizzes customizable and adaptable to different learning needs. This flexibility allows teachers to create quizzes that align perfectly with their lessons, ensuring that students are tested on relevant material.

The AI-powered question generator further enhances the experience by reducing the time teachers spend on manual quiz creation. With just a few clicks, teachers can generate a set of questions tailored to the subject, ensuring variety and depth. This not only saves time but also introduces students to a wide range of questions they may not encounter in a traditional quiz format. Moreover, the interactive interface keeps students engaged, encouraging them to actively participate and enhance their knowledge.

Students also benefit from the real-time feedback provided by QuizBOT. As soon as a quiz is completed, students receive immediate results, allowing them to understand where they stand and what areas need improvement. This instant feedback loop is key to helping students learn more effectively, as it promotes quick correction of mistakes and reinforces concepts they may have struggled with.

Additionally, QuizBOT supports collaboration by allowing students to take quizzes in a shared classroom environment, fostering a sense of competition and motivation among peers. The platform's user-friendly design ensures that both tech-savvy and non-tech-savvy users can navigate it with ease, making it accessible to a broad range of users. Furthermore, the integration of modern technologies like AI positions QuizBOT as a cutting-edge solution in the realm of educational tools, bringing innovation into the classroom.

In essence, QuizBOT bridges the gap between traditional teaching methods and modern technology, creating a well-rounded educational tool that promotes both learning and assessment in a streamlined manner.

In addition to simplifying quiz creation, QuizBOT empowers teachers to customize quizzes according to the specific needs of their students. Teachers can set different difficulty levels, adjust the number of questions, and select topics that align with their curriculum. This flexibility ensures that the quizzes are tailored to the unique learning objectives of each class or subject. Moreover, the AI-generated questions are diverse and dynamic, providing a fresh set of questions each time, which reduces the chances of repetition and encourages deeper learning among students.

The platform also supports collaboration among teachers. Multiple educators can contribute to the quiz database, sharing resources and knowledge to enhance the overall quality of assessments. This feature promotes a sense of community within the educational institution, fostering teamwork and knowledge-sharing. Additionally, the quizzes can be reused, modified, or expanded, making it easy to adapt them for different classes or future use.

For students, QuizBOT enhances the learning process by offering engaging, interactive quizzes that can be accessed from anywhere. The real-time participation feature allows students to join quizzes from their devices, enabling flexible learning whether in a classroom or at home. With instant feedback, students are not left waiting to understand their performance; instead, they can immediately see their results and learn from their mistakes, improving retention and motivation. Teachers, in turn, benefit from instant analytics, enabling them to quickly assess individual or class performance and adjust their teaching strategies accordingly.

By streamlining the quiz process and integrating AI, QuizBOT creates an environment where both teaching and learning are more effective, personalized, and engaging. It reduces the administrative burden on teachers while offering students a fun and educational way to assess their knowledge.

1.2 OVERALL DESCRIPTION

QuizBOT is primarily intended for educational institutions, including schools, colleges, and other learning environments. It provides a flexible and efficient solution for managing assessments, designed to meet the needs of both teachers and students. The platform offers a wide range of features that ensure a smooth, engaging, and personalized quiz experience:

QUIZ CREATION AND MANAGEMENT

Teachers have complete control over the quiz creation process. They can either create quizzes manually, tailoring questions to suit the curriculum, or use the AI-powered feature to automatically generate questions. This flexibility ensures that teachers can save time while still creating high-quality, relevant quizzes.

AI-GENERATED QUIZZES

By leveraging artificial intelligence, QuizBOT can generate quiz questions that align with the topics and difficulty levels specified by the teacher. This feature not only reduces the time spent on quiz creation but also introduces a variety of question types, making assessments more dynamic and challenging for students. Teachers can refine the generated questions to ensure they match the learning objectives.

REAL-TIME QUIZ PARTICIPATION

Students can easily participate in quizzes using a unique room code provided by the teacher. This ensures a streamlined process for joining quizzes, whether in a physical classroom or a remote learning environment. Real-time participation also allows teachers to monitor progress and engagement during the quiz.

INSTANT FEEDBACK AND RESULTS

One of the most valuable aspects of QuizBOT is its ability to deliver immediate feedback to both students and teachers. Once a quiz is completed, students can instantly view their scores and review correct answers. This helps them identify areas of improvement and gain a deeper understanding of the material. Teachers, on the other hand, can quickly assess student performance and identify trends or gaps in learning.

QUIZ CUSTOMIZATION AND FLEXIBILITY

In addition to AI-generated questions, teachers can fully customize quizzes by selecting question formats, such as multiple-choice, true/false, and short answer. This allows for a more tailored assessment experience that suits the specific needs of the class.

CLASSROOM INTEGRATION

QuizBOT is designed to integrate seamlessly into classroom activities, supporting both individual and group assessments. It can be used for timed quizzes, mid-term assessments, or even casual knowledge checks during lessons, making it a versatile tool for any educational setting.

PERFORMANCE TRACKING

The platform also includes features for tracking student progress over time. Teachers can access detailed reports on quiz performance, allowing them to monitor individual students' growth or identify areas where the class as a whole may need additional focus.

DEFINITION

QUIZBOT

The name of the web application specifically designed to simplify the process of quiz creation, management, and administration for teachers, while providing a user-friendly platform for students to take quizzes.

QUIZ

A structured tool or form of assessment made up of multiple questions, designed to evaluate a student’s knowledge, understanding, or proficiency in a particular subject. Quizzes can vary in format, including multiple-choice, true or false, short answer, or even fill-in-the-blank questions.

AI GENERATIVE QUIZ

A feature embedded within QuizBOT that utilizes artificial intelligence to automatically create quiz questions. This feature allows teachers to specify topics, difficulty levels, and the number of questions, and the AI generates unique, relevant questions, reducing the workload for educators while ensuring a diverse set of assessments.

FRONTEND

The client-side, user-facing part of the QuizBOT application. This is where teachers create quizzes and students interact with them. The frontend includes all the visual elements, buttons, forms, and navigation that users see and interact with, ensuring a smooth and intuitive experience.

BACKEND

The server-side portion of the application that operates behind the scenes. It manages and processes the data entered by users, handles quiz submissions, stores user information, and ensures the overall functionality of the application. The backend is critical for processing quiz results, handling AI quiz generation, and maintaining secure and efficient operations.

DATABASE

A structured system used to store all the data related to the application. In QuizBOT, the database holds information about users (teachers and students), quizzes (questions, answers, etc.), and results. The database ensures that data is securely stored and can be easily retrieved for future use, such as analysing student performance or updating quiz content.

In addition to these definitions, the system also uses secure methods to store and manage user authentication data, ensuring the privacy of both teachers and students. The frontend and backend work together seamlessly to provide a real-time experience where students can access quizzes, submit their responses, and instantly receive results, while teachers can monitor progress and make updates as needed. This interconnected system ensures that both quiz creation and participation are efficient and effective, supporting the overall goals of education and assessment.

ACRONYMS

HTML (HYPER TEXT MARKUP LANGUAGE)

The standard language used to create and design web pages.

CSS (CASCADING STYLE SHEETS)

A style sheet language used to describe the presentation of a web page, including layout, colours, and fonts.

JS (JAVASCRIPT)

A programming language that adds interactivity and dynamic behaviour to websites, allowing features like animations and real-time content updates.

DJANGO - A HIGH-LEVEL PYTHON-BASED WEB FRAMEWORK

It simplifies web development by providing built-in tools and features for building robust, scalable web applications quickly.

AI (ARTIFICIAL INTELLIGENCE)

Technology that enables machines to perform tasks that typically require human intelligence, such as problem-solving, learning, and decision-making.

SQL (STRUCTURED QUERY LANGUAGE)

A programming language used to manage and manipulate relational databases, allowing users to store, retrieve, and update data efficiently.

API (APPLICATION PROGRAMMING INTERFACE)

A set of rules and protocols that allow different software applications to communicate and share data with each other, facilitating integration between systems and services.

UML (UNIFIED MODELLING LANGUAGE)

A visual modelling language used to represent the structure, behaviour, and interactions of a software system during its development.

JSON (JAVASCRIPT OBJECT NOTATION)

A lightweight data interchange format used to transmit data between servers and web applications in a human-readable format.

ABBREVIATIONS

DB (DATABASE)

A structured collection of data that is stored and accessed electronically. In the context of QuizBOT, the database stores all user information, quiz data, and results securely.

HTTP (HYPER TEXT TRANSFER PROTOCOL)

A protocol used for transferring data over the web. It forms the foundation of data communication for QuizBOT's web-based platform.

HTTPS (HYPER TEXT TRANSFER PROTOCOL SECURE)

An encrypted version of HTTP that ensures secure communication between the user's browser and the server. QuizBOT uses HTTPS to protect sensitive user data, such as login credentials and quiz results.

JSON (JAVASCRIPT OBJECT NOTATION)

A lightweight data-interchange format used to transmit data between a server and a web application. QuizBOT uses JSON to exchange quiz data and results between the frontend and backend efficiently.

UML (UNIFIED MODELLING LANGUAGE)

A visual language for modelling the architecture of software applications. It helps in visualizing and designing the structure of QuizBOT's system during development.

UI/UX (USER INTERFACE / USER EXPERIENCE)

It refers to the design and functionality of the application's interface that users interact with. A well-designed UI/UX ensures that

QuizBOT is user-friendly, intuitive, and provides a smooth experience for both teachers and students.

These abbreviations represent key components of the technology and architecture that power QuizBOT, contributing to its performance, security, and user-friendliness.

OVERVIEW

QuizBOT is designed to transform how quizzes are developed and administered in educational settings. The platform is built to be intuitive, allowing teachers to easily create quizzes using either custom questions or AI-generated content. This flexibility means that educators can tailor assessments to meet their specific curricular needs and learning objectives. Students can join quizzes using unique codes, making it simple to access the quizzes from any device with an internet connection. Once they complete their assessments, they receive instant feedback, which is crucial for understanding their performance. This immediate feedback allows students to quickly identify areas where they may need improvement and encourages them to take an active role in their learning.

The backend, powered by the Django framework, ensures that data related to quizzes, results, and users is securely managed. This security is essential, as it protects sensitive information and maintains the integrity of the assessment process. Additionally, the integration of artificial intelligence enables QuizBOT to adapt to the specific needs of both teachers and students, ensuring that quiz content remains relevant and challenging. The AI not only generates questions but can also analyse students' performance over time, helping teachers refine their instructional strategies based on real data.

Moreover, QuizBOT includes a variety of question types, such as multiple-choice, true/false, and open-ended questions, allowing teachers to assess students' understanding from different angles. This diversity in question format helps cater to various learning styles and preferences. Teachers can also set different difficulty levels for questions, ensuring that assessments are appropriately challenging for all students.
The platform is designed with user experience in mind, featuring an appealing interface that is easy to navigate for both educators and students. Resources and tutorials are available to help users make the most of the application, ensuring that they can utilize all its features effectively. Furthermore, QuizBOT supports collaboration among teachers, enabling them to share quizzes and best practices, fostering a community of learning.

In summary, QuizBOT aims to create a more engaging and efficient learning environment where assessments are simplified for educators and more interactive for students. By leveraging modern technologies such as AI and Django, the platform enhances the overall quiz experience, making it a valuable tool in both traditional and digital learning spaces. As educational methods continue to evolve, QuizBOT is positioned to be at the forefront of these changes, providing innovative solutions that support teaching and learning in a rapidly changing world. With its focus on collaboration, adaptability, and user-friendly design, QuizBOT is poised to make a lasting impact on the educational landscape.

1.3 SYSTEM REQUIREMENT

The Quiz Web Application is an innovative platform designed to enhance the learning experience by providing teachers with the tools to create quizzes and enabling students to participate actively in their educational journey. It aims to simplify the quiz creation process through two primary methods: manual input by the teacher and an automated system powered by an AI-based generator. This dual approach ensures that educators can tailor quizzes to their specific needs while also benefiting from the efficiency and creativity of AI-generated content. The system supports both pre-made quizzes, which can be reused, and AI-generated quizzes that are customized based on user input, ensuring a diverse range of assessments that cater to different learning styles and needs.

In addition to facilitating quiz creation, the platform is designed to provide a seamless user experience for both teachers and students. Teachers can easily navigate the interface to manage their quizzes, while students can effortlessly join and participate in quizzes with minimal barriers. This user-centric design promotes engagement and makes learning more interactive, allowing students to take ownership of their educational progress.

PRODUCT FUNCTIONS

TEACHER FUNCTIONS

Create custom quizzes with questions of their choice, allowing for personalized assessments that reflect the curriculum.

Generate quizzes automatically by selecting a topic, question level, and the number of questions. The AI generates the quiz based on these selections, saving time and effort for educators.

Update or delete previously created quizzes, ensuring that the content remains relevant and accurate.

Share a room code with students to facilitate easy access to quizzes, making the process of joining straightforward and efficient.

Review and analyse quiz results to identify trends in student performance, helping to inform future teaching strategies.

STUDENT FUNCTIONS

Join a quiz using a room code provided by the teacher, allowing for quick access to assessments.

Submit answers and receive immediate results and feedback upon completion, enabling students to understand their performance right away.

Track progress over time through a dashboard that displays completed quizzes, scores, and areas for improvement.

AI QUIZ GENERATION

Teachers can opt to generate quizzes by choosing parameters such as topic, difficulty level, and quantity. The AI then automatically generates the quiz based on the input, providing a diverse array of questions that align with learning objectives.

The AI can adapt questions based on the previous performance of students, ensuring that the assessments are neither too easy nor too difficult, thereby maintaining an appropriate challenge level.

Overall, the Quiz Web Application aims to foster an engaging learning environment that not only enhances the assessment experience but also supports educators in their teaching efforts. By integrating advanced technology and user-friendly features, the platform seeks to bridge the gap between traditional education and modern learning techniques, preparing students for future academic success. This approach not only encourages a deeper understanding of the material but also motivates students to take an active role in their education, fostering a culture of continuous learning and improvement.

FUNCTIONAL REQUIREMENTS

TEACHER REGISTRATION AND LOGIN

Teachers are required to sign up by providing a valid email address, a username, and a secure password to gain access to the quiz creation features. After successfully logging in, teachers will be redirected to a comprehensive dashboard. This dashboard will provide easy access to all their quizzes, allowing for effective management and organization of their assessment tools.

STUDENT REGISTRATION AND LOGIN

To participate in quizzes, students must either register for a new account or log in using their existing credentials. This ensures that only authorized users can access the quizzes, maintaining the integrity of the assessment process.

QUIZ CREATION

Teachers have the flexibility to create quizzes manually by entering custom questions that fit their curriculum needs. Additionally, they can utilize the AI question generator, which allows them to select the desired topic, specify the question difficulty level, and determine the number of questions. This feature enables teachers to quickly generate quizzes that are relevant and challenging for their students, saving valuable time in the process.

QUIZ PARTICIPATION

Students can join quizzes using a unique room code provided by their teacher. This code ensures that only students who are meant to participate in the quiz can access it. Once in the quiz, students will be able to answer a series of questions, after which they can view their results immediately upon submission. This feature enhances engagement and allows for a more dynamic learning experience.

RESULT MANAGEMENT

After submitting their quizzes, students will have the opportunity to review their results along with detailed performance metrics. This feedback will help them understand their strengths and areas for improvement, thereby promoting a deeper learning process. Teachers will also have access to aggregate results, allowing them to analyse overall class performance and adjust their teaching strategies accordingly.

NON-FUNCTIONAL REQUIREMENTS

USABILITY

The application must be designed with a focus on usability, ensuring that both teachers and students find it intuitive and easy to navigate. Clear instructions and user-friendly interfaces will enhance the overall experience, making it accessible for users of varying technical skills.

PERFORMANCE

The system is required to handle multiple quiz sessions simultaneously without experiencing delays in processing or generating quizzes. Efficient performance will ensure that both teachers and students can rely on the application during peak usage times, such as during exams or assignments.

SECURITY

Protecting user data is of utmost importance. The application must implement robust security measures to ensure that sensitive information, including login credentials and quiz results, is stored securely. This includes using encryption protocols for data storage and transmission.

SCALABILITY

QuizBOT should be designed to scale effectively as the number of users increases. This includes being capable of handling a growing number of AI-generated quizzes and real-time student participation without compromising performance. The architecture must allow for easy expansion to accommodate more users, ensuring that the application remains functional and efficient as demand increases.

RELIABILITY

The application should exhibit high reliability, ensuring that it is consistently available for both teachers and students. Downtime should be minimal, and the system should have protocols in place to recover from potential failures, ensuring a smooth user experience at all times.

ACCESSIBILITY

QuizBOT must adhere to accessibility standards to ensure that all users, including those with disabilities, can effectively use the application. This involves implementing features such as screen reader compatibility and keyboard navigation to provide an inclusive learning environment.

In conclusion, the specific requirements for QuizBOT encompass a range of functional and non-functional needs designed to create an effective, user-friendly, and secure quiz management system. By addressing these requirements, the application aims to deliver an exceptional experience for both teachers and students, ultimately enhancing the educational process.

SYSTEM QUALITY ATTRIBUTES

The following attributes that are essential for ensuring the system performs well, meets user expectations, and remains robust over time

USABILITY

The platform is designed for ease of use, providing an intuitive interface for both teachers and students, with clear navigation and guidance for quiz creation and participation.

PERFORMANCE

The system ensures quick response times for loading pages, generating quizzes, and displaying results, even under high user loads.

SECURITY

Secure user authentication, data encryption, and protected quiz access via unique room codes ensure that user data and quiz integrity are maintained.

SCALABILITY

The application can scale seamlessly to handle increasing users, quizzes, and AI-generated content without performance degradation.

RELIABILITY

The system guarantees high availability with automatic data backups and periodic auto-saving during quizzes to prevent data loss.

MAINTAINABILITY

The codebase is modular and well-documented to support easy updates, bug fixes, and the addition of new features.

ACCESSIBILITY

The platform follows accessibility standards to ensure it is usable by individuals with disabilities, including screen reader compatibility.

TOOLS & TECHNOLOGIES

HTML (HYPERTEXT MARKUP LANGUAGE)

Used to structure the content and layout of the web pages, including forms, quizzes, and user interfaces.

CSS (CASCADING STYLE SHEETS)

Used for styling the web application, ensuring a visually appealing and responsive design across different devices.

JS (JAVASCRIPT)

Provides interactive functionality on the client side, such as form validation, dynamic content updates, and enhancing user experience.

DJANGO (PYTHON)

A high-level web framework used for backend development, managing the logic for user authentication, quiz creation, AI integration, and data handling.

AI (ARTIFICIAL INTELLIGENCE)

Integrated to automatically generate quizzes based on selected topics, question levels, and quantities, enhancing the dynamic nature of quiz creation.

1.4 MODULES

USER AUTHENTICATION

The User Authentication module is a key part of QuizBOT that ensures only the right people can access the platform. It offers a secure method for users to log in or sign up using their email, username, and password, confirming their identity before allowing access. This module plays a big role in keeping users' private information safe, making sure that unauthorized people can't enter. Besides basic login and signup, it also manages important features like password recovery. If a user forgets their password, they can easily reset it. The module makes the entire login process simple, secure, and easy to use, ensuring that users feel confident their data is protected.

QUIZ MANAGEMENT

The Quiz Management module forms the foundation of QuizBOT, as it enables teachers, administrators, or anyone creating quizzes to manage all aspects of the quiz creation process. With this module, users can easily create, edit, and even delete quizzes, providing full control over the content. From writing questions to adding multiple choices for answers, this module ensures the quiz is set up correctly. Teachers can categorize quizzes by topics or difficulty levels, set time limits for each quiz, and assign specific quizzes to different groups of students. It also allows educators to track the progress of students while giving them options to customize quizzes to fit the learning needs of individuals, ensuring that every student has a fun and educational experience.

AI INTEGRATION

The AI Integration module is one of the most exciting features of QuizBOT. It uses advanced Artificial Intelligence to automatically generate multiple-choice questions (MCQs) based on the topics, levels, and number of questions selected by the user. This makes quiz creation much faster and easier, especially for teachers who want quick, relevant quizzes without spending too much time preparing them. The AI ensures that the questions are well-matched to the chosen topic, providing a challenging yet fair quiz for students. This feature helps teachers focus more on student interaction rather than spending too much time creating quiz content. It also makes QuizBOT highly adaptable, as quizzes can be generated on a wide variety of subjects at different levels of complexity.

RECORDS MANAGEMENT

The Records Management module is essential for keeping track of all quiz-related information in QuizBOT. This includes storing quiz results, user scores, participation data, and performance trends. Teachers can use this module to look back on students' performance in previous quizzes, helping them to analyse strengths and areas for improvement. For students, it provides a way to review their progress over time. The module securely stores all data, ensuring that personal information remains private while still being accessible to users whenever needed. Teachers can use this data to provide feedback and tailor future quizzes based on the student's learning needs, making it an essential part of managing the entire quiz system.

PROFILE MANAGEMENT

The Profile Management module gives each user control over their personal information on QuizBOT. This feature allows users to update their profiles, including their name, email, and password, whenever necessary. Users can also track their quiz history, viewing their past performances and understanding their learning progress over time. By offering these personalized features, QuizBOT makes sure each user has a customized experience, whether they are teachers or students. This module encourages users to take ownership of their personal data, empowering them to manage their accounts effectively while enhancing their overall experience on the platform.

USE CASE DIAGRAMS

A Use Case Diagram is a visual representation of the interactions between users (actors) and a system, outlining the functional requirements of a system. It shows how external actors (such as users or other systems) interact with the system to achieve specific goals, represented as use cases. The diagram typically includes actors, use cases, and relationships like associations, generalizations, and dependencies. Use Case Diagrams are essential for understanding system behavior from an external viewpoint and are widely used in software development to capture system functionality and user interactions.

Actors: An actor represents any entity that interacts with the system. This could be a human user, another system, or an external device.
Primary actors are the main users that initiate a particular action within the system (e.g., Teacher, Student).
Secondary actors might provide support or play a secondary role (e.g., Admin).
Actors are represented as stick figures and are positioned outside the system boundary.

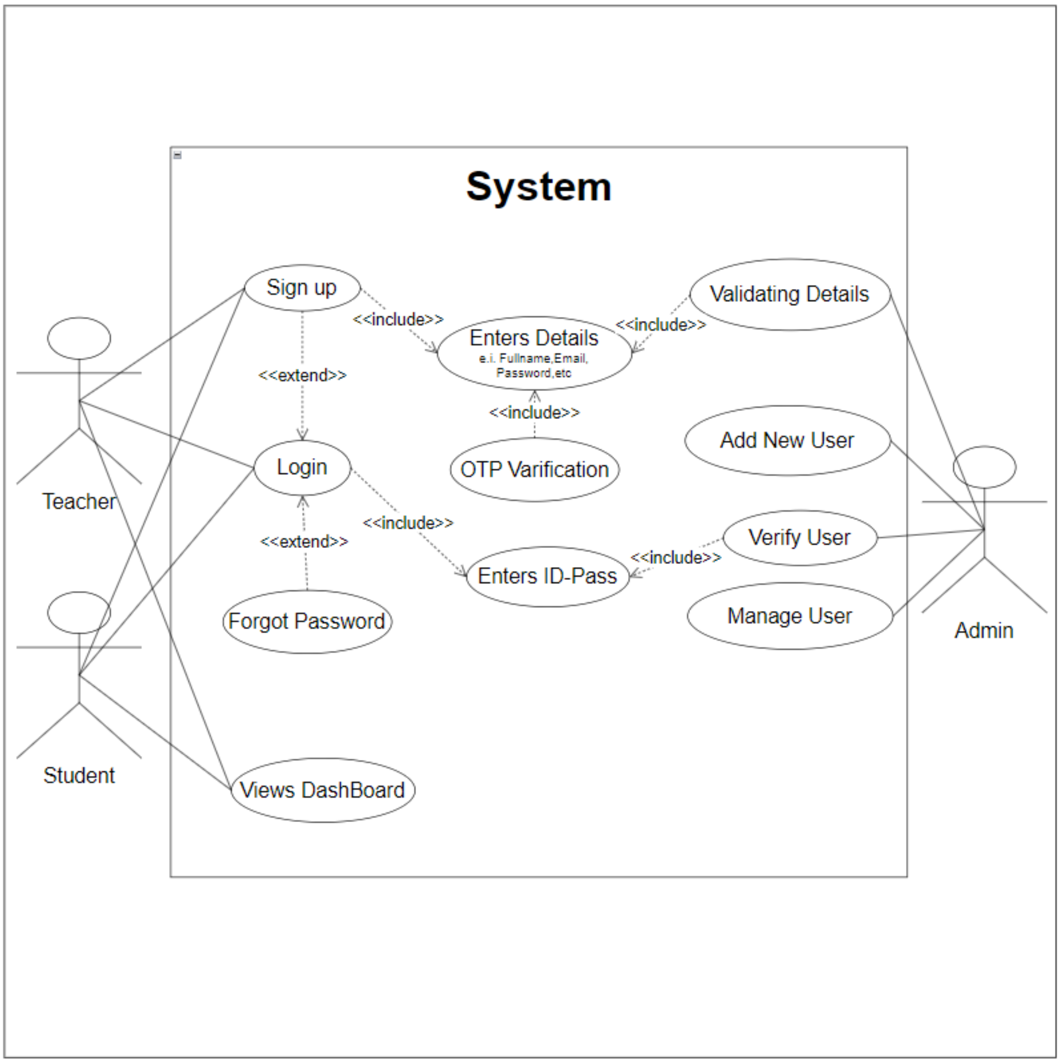
Use Cases: A use case is a specific functionality or goal that the system performs for the actor. It represents a task, activity, or service provided by the system.
Each use case should have a well-defined scope and be independent of other use cases.
Use cases are drawn as ovals inside the system boundary, and they represent activities like "Login," "Signup," or "Submit Quiz."

System Boundary: The system boundary defines the scope of the system, visually separating what is part of the system from what lies outside it. All the use cases lie inside this boundary, while the actors are placed outside.
It helps clarify the interactions between external actors and the system functions.

Relationships:
Include: This relationship shows that one use case always includes the behavior of another use case. It represents a mandatory relationship between use cases.
Extend: This shows optional or conditional behavior. A base use case may extend another use case if certain conditions are met.

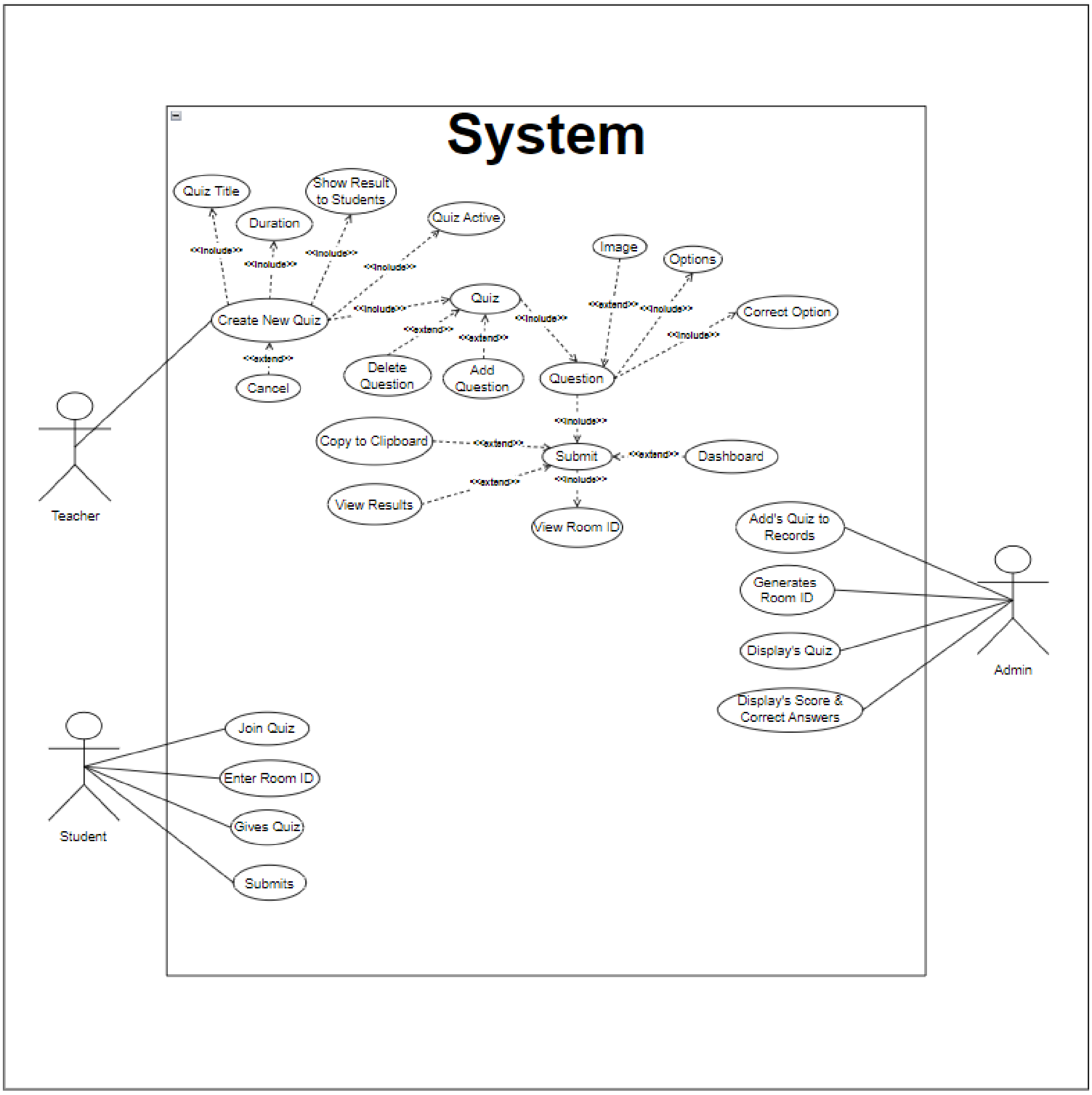
A Use Case Diagram provides a high-level view of the interactions between external actors and the system, capturing the functional requirements in a user-friendly way. By clearly showing how users interact with the system, it enables better understanding, facilitates communication, and helps in defining the system's scope during the early stages of development. As a powerful tool in the UML suite, it aids both technical and non-technical stakeholders in understanding the functional behavior of a system.

1) User Authentication (Login & Signup)



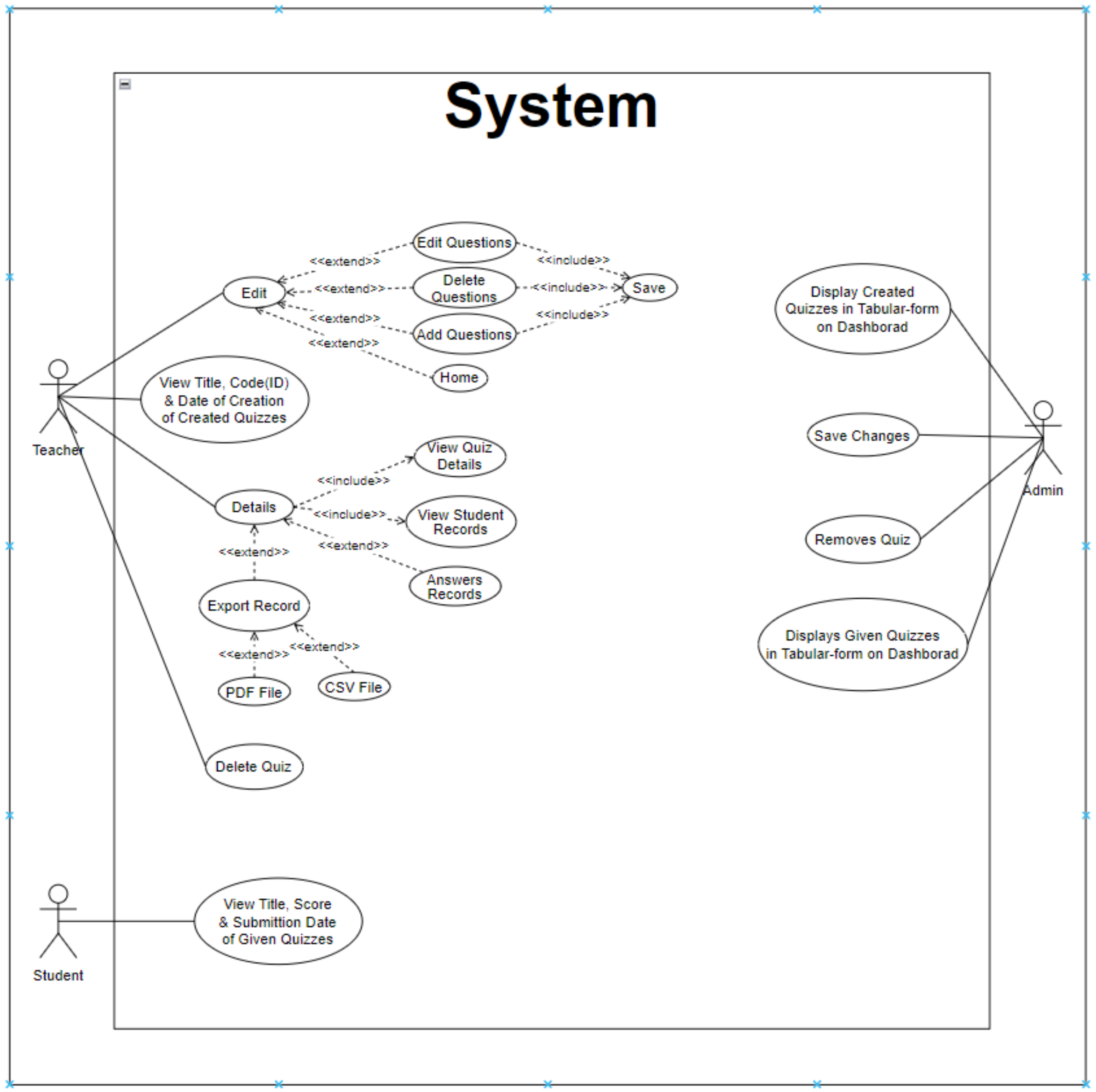
The **User Authentication (Login & Signup)** Use Case Diagram represents the interaction between the system and three actors: Teacher, Student, and Admin. Both Teachers and Students can sign up to create an account and log in to access the system. The Admin has additional privileges to add new users, manage user accounts (including updating or deleting), and validate user credentials. The use case diagram highlights these interactions, showing how each actor engages with the authentication processes.

2) Quiz



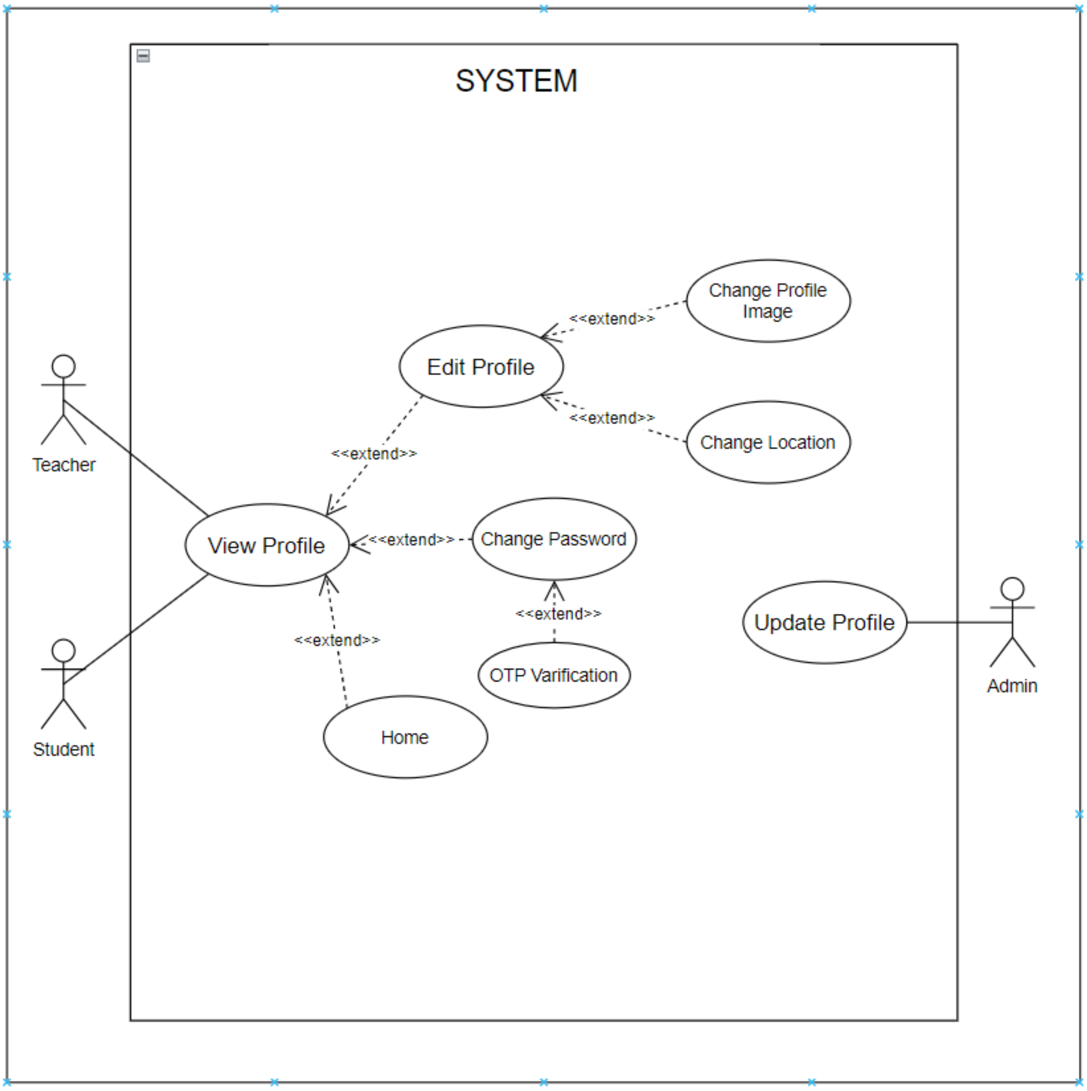
The **Quiz Management** Use Case Diagram illustrates the interactions between the system and three actors: Teacher, Student, and Admin. The Teacher can create new quizzes for students. The Admin is responsible for adding quiz records, generating Room IDs, displaying quizzes, and managing quiz scores. Students can join a quiz by entering the Room ID and submitting their answers. The diagram highlights the flow of quiz creation, management, and participation within the system.

3) **Records**



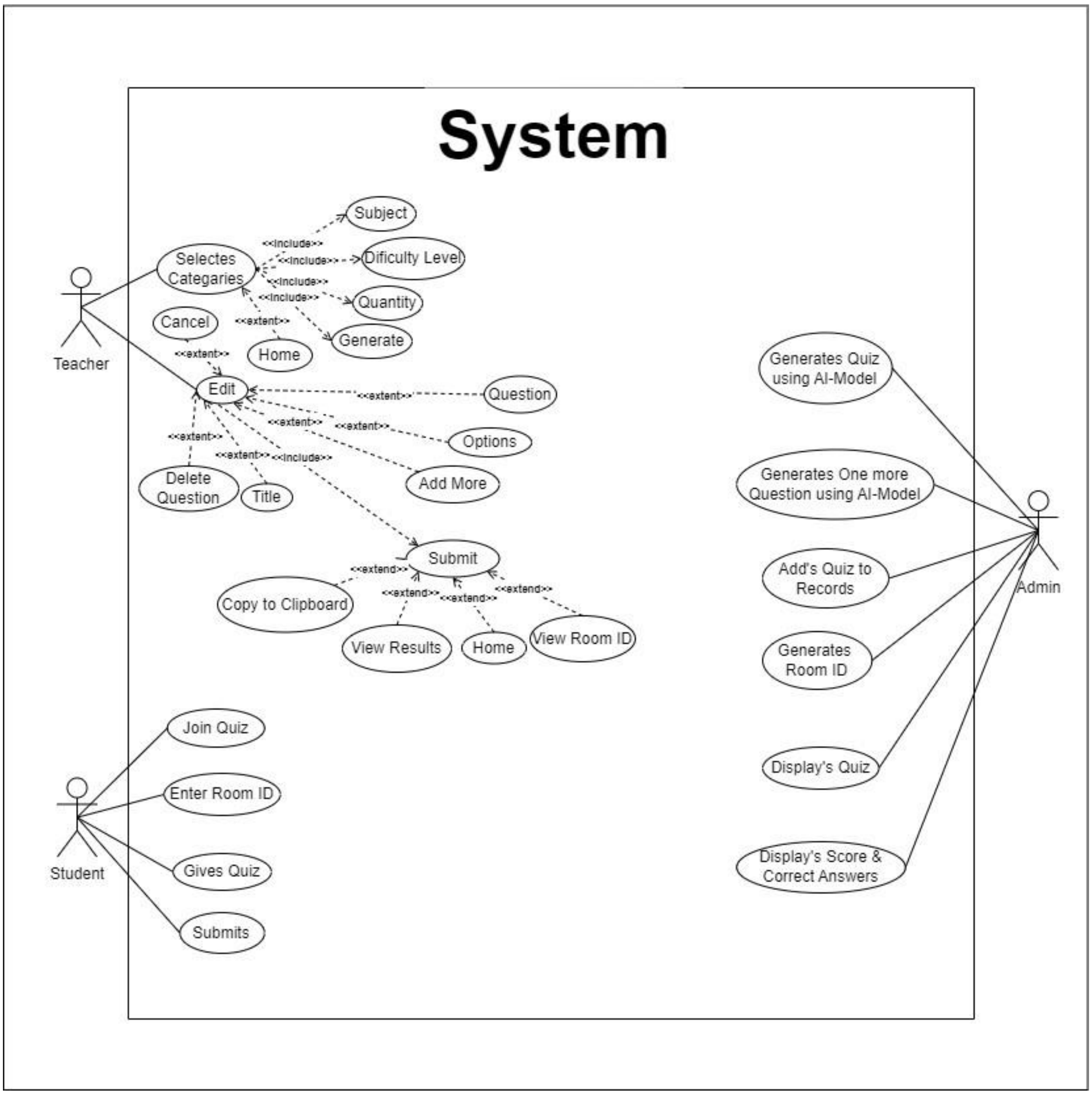
The **Record Management** Use Case Diagram showcases how Teachers, Students, and Admin interact with quiz records. Teachers can edit quizzes and view details such as the title, code, and date of creation. The Admin can display created quizzes in a tabular format on the dashboard, remove quizzes, and also display quizzes submitted by students. Students can view their quiz records, including the quiz name, score, and submission date. The diagram outlines the management and viewing of quiz-related records across these roles.

4) **Profile**



The **Profile Management** Use Case Diagram outlines how Teachers, Students, and Admin manage user profiles. Teachers and Students can view and edit their profiles, including changing their profile picture and updating their password using OTP verification. The Admin has the ability to update profile details for all users. This diagram highlights the interactions involved in profile viewing, editing, and security updates for each user role.

5) AI INTEGRATION



The **AI Integration** Use Case Diagram outlines the interactions with AI-generated quizzes for three actors. Teacher can select categories for AI-generated quizzes, including subject, difficulty level, and question quantity, and generate quizzes based on these preferences. Student can join AI-generated quizzes using a Room ID and submit their answers. Admin has advanced capabilities to generate quizzes using an AI Model, add more questions via the AI model, manage quiz records, generate Room ids, and display quizzes, scores, and correct answers.

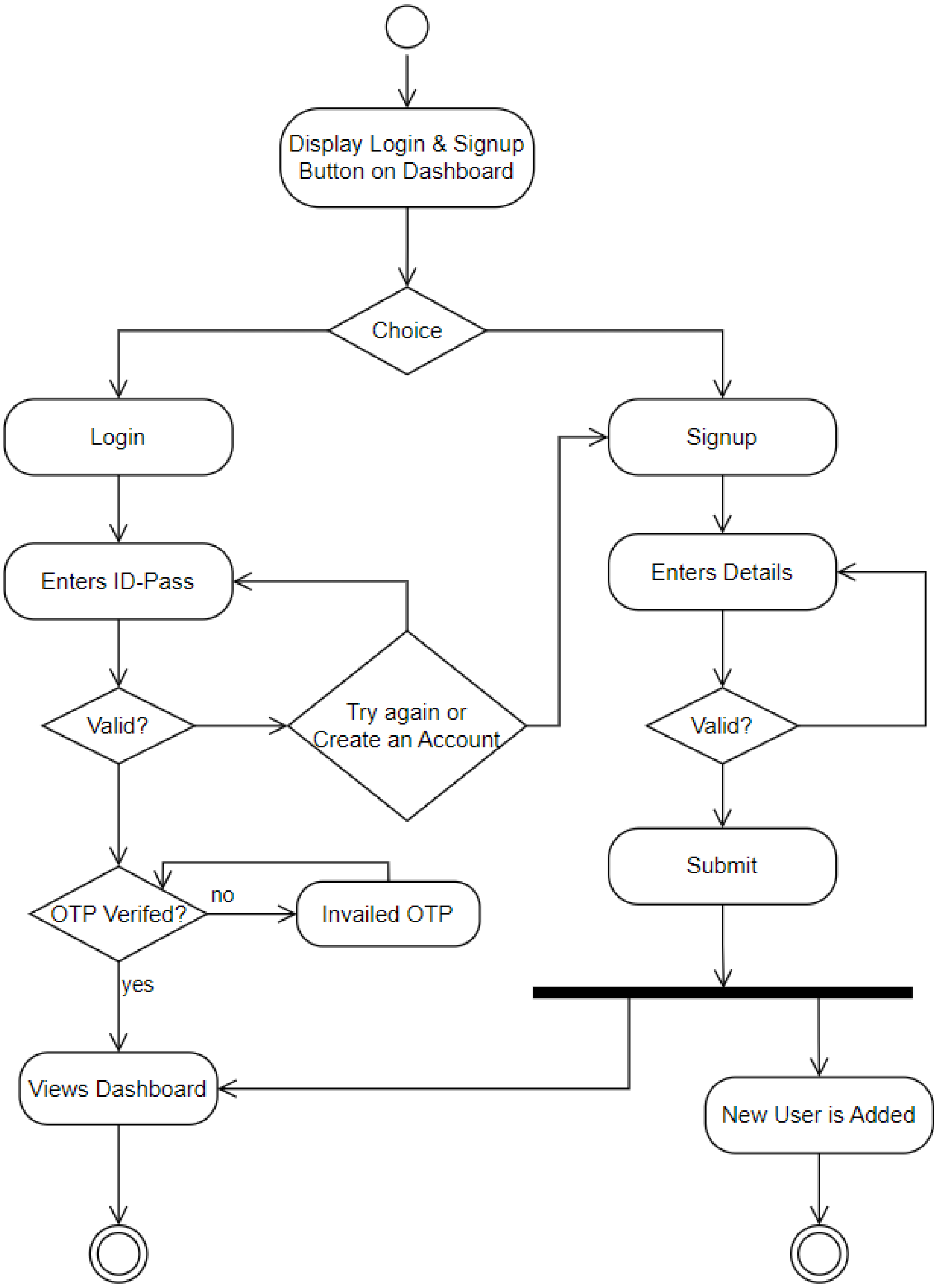
ACTIVITY DIAGRAMS

An Activity Diagram is a UML (Unified Modeling Language) behavioral diagram that represents the flow of activities or actions within a system or process. It is essentially a flowchart that captures the sequence of operations, highlighting decision points, parallel actions, and the control flow of processes. Activity diagrams are commonly used in both software development and business processes to model workflows, system activities, and use case scenarios, making them crucial for understanding how tasks are executed step by step.

Activities: Represent tasks or actions in a workflow, shown as rounded rectangles. They can be simple actions or complex processes.
Initial Node: The starting point of the diagram, depicted as a filled black circle, where the flow begins.
Final Node: The endpoint of the process, shown as a circle with a surrounding ring, marking the conclusion of the activity flow.
Transitions/Control Flows: Arrows that show the sequence of activities and direction of the flow, which can be sequential, parallel, or branched.
Decision Nodes: Diamond-shaped nodes representing a branching point, where the flow follows a path based on conditions (e.g., Yes/No).
Merge Nodes: A diamond shape used to merge multiple flows into one after a decision point.
Fork and Join Nodes: Fork nodes split the flow into parallel actions, while join nodes synchronize them back together.
Swimlanes: Partitions that organize activities by actor or department, showing responsibilities in the workflow.

Activity diagrams provide a detailed view of the sequence and flow of activities, helping stakeholders, developers, and analysts understand how a system or process works**.**

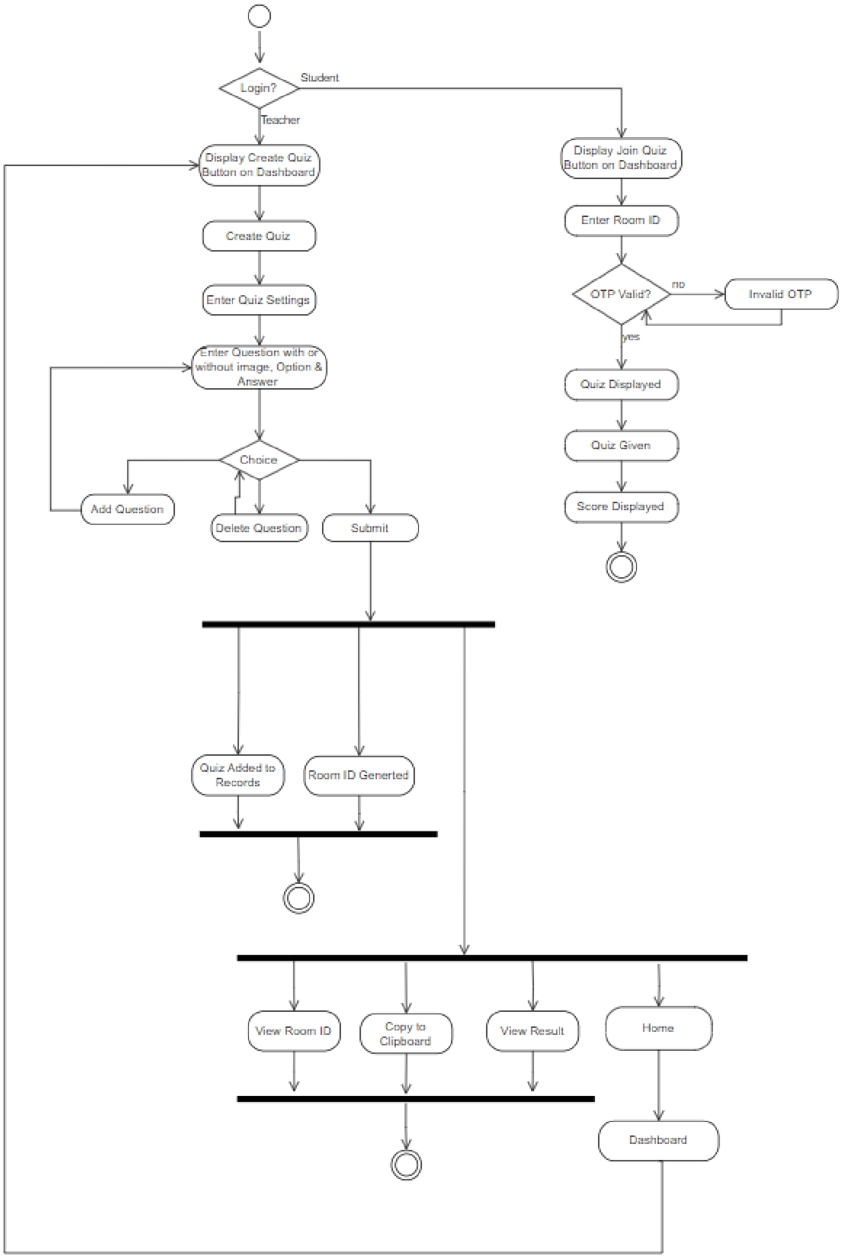
1) User Authentication (Login & Signup)



User Authentication Activity Diagram Description:

The User Authentication Activity Diagram outlines the process of user registration and login in the system. Both teachers and students can sign up for an account, after which they can log in using their credentials. The admin plays a crucial role in managing users, which includes adding new users and validating existing accounts to ensure proper access and security within the system.

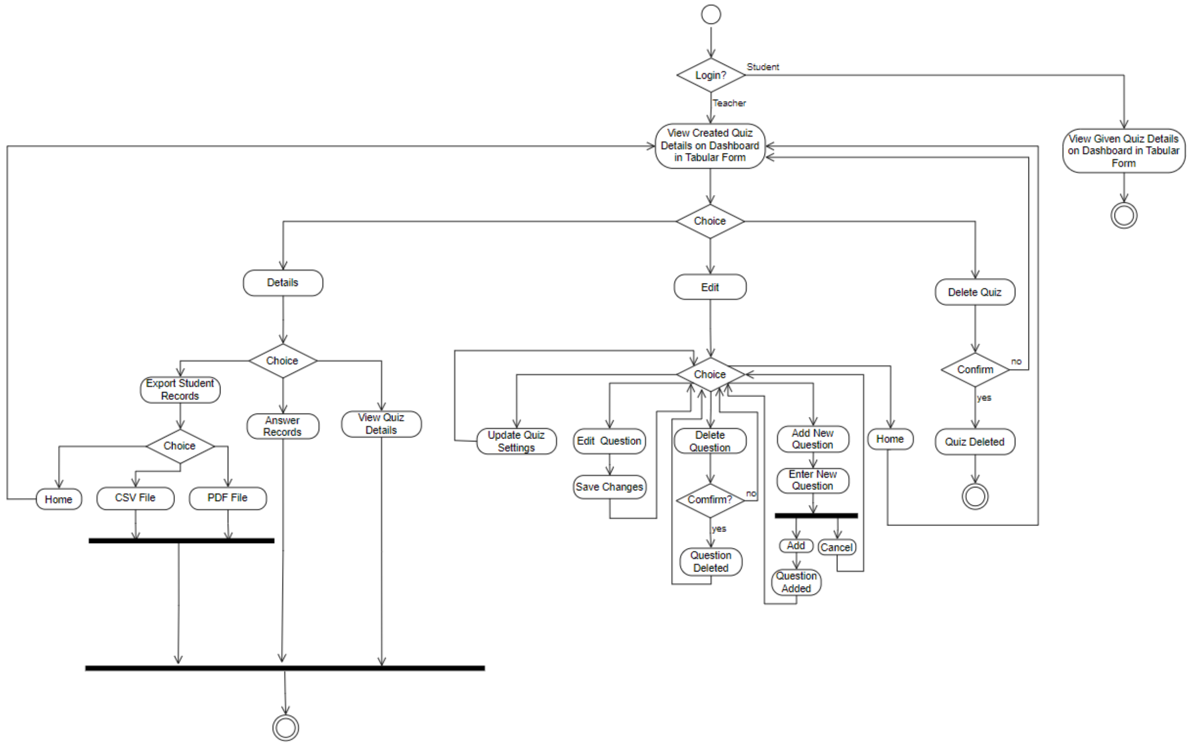
2) Quiz



Quiz Activity Diagram Description:

The Quiz Management Activity Diagram illustrates the workflow of quiz-related activities in the system. Teachers can create new quizzes, specifying details such as title and questions. The admin is responsible for adding quiz records, generating Room ids, and displaying quizzes along with their scores. Students can join a quiz by entering the Room ID and submitting their answers, completing the interaction with the quiz management system.

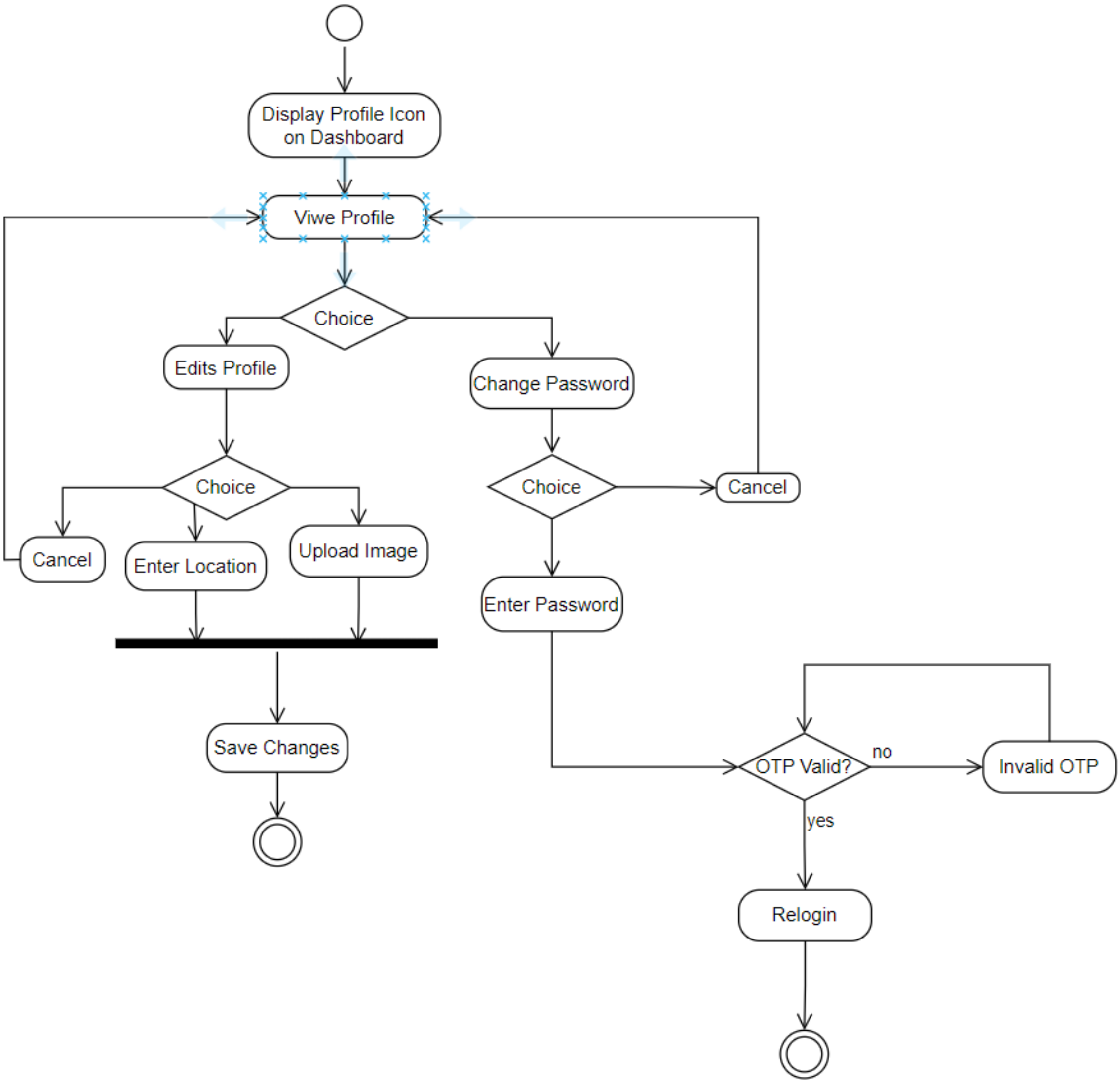
3) Records



Records Activity Diagram Description:

The Record Management Activity Diagram details the interactions related to quiz records within the system. Teachers can edit quizzes, viewing essential details such as the title, code, and creation date. Admins have the capability to display created quizzes in a tabular format on the dashboard, remove quizzes, and showcase submitted quizzes by students. Students can access their quiz records, which include the quiz name, score, and submission date, allowing for effective performance tracking.

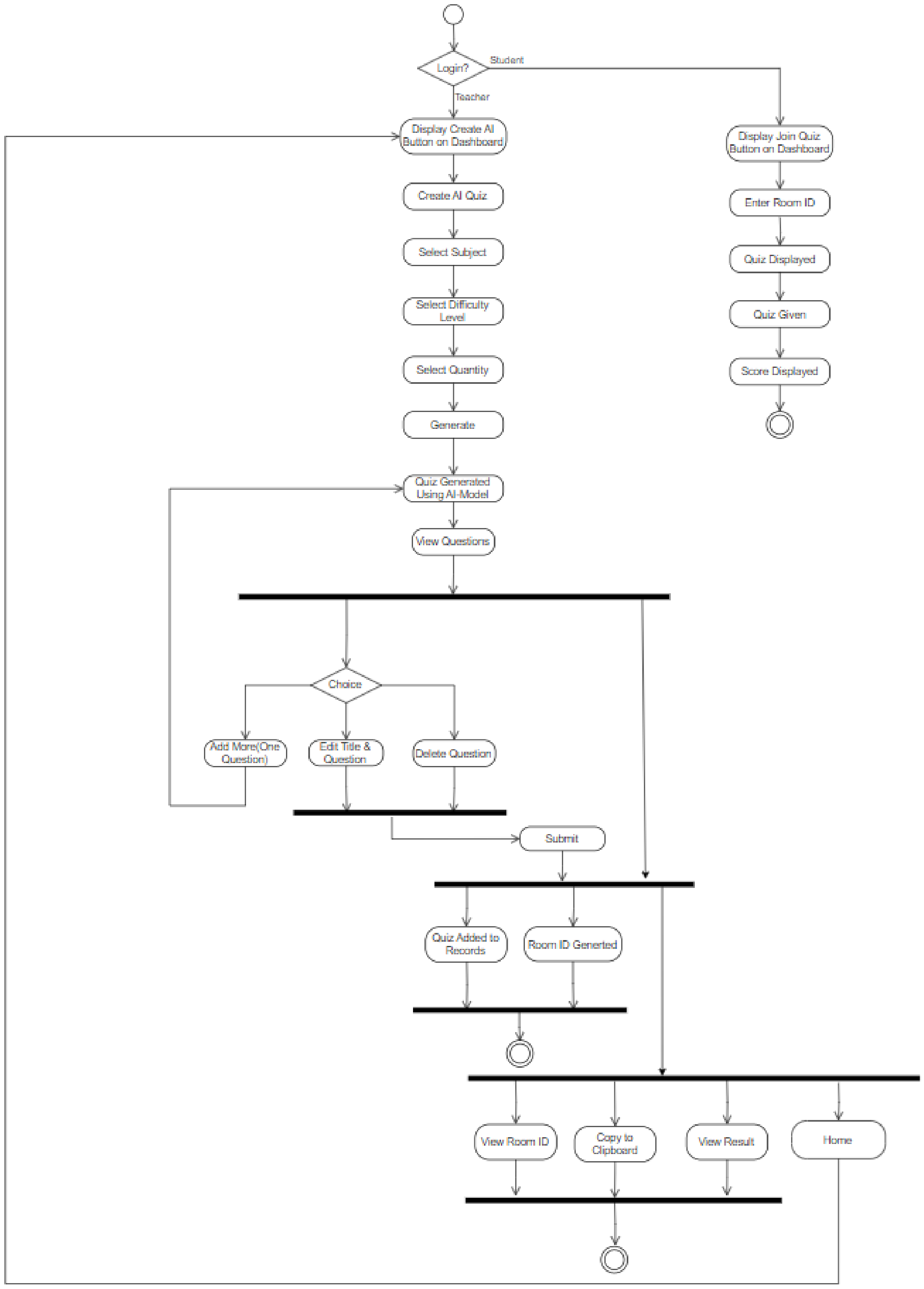
4) Profile



Profile Activity Diagram Description:

The Profile Management Activity Diagram illustrates how users interact with their profiles in the system. Both teachers and students can view and edit their profiles, which includes changing their profile picture and updating their password through OTP verification for added security. The admin has the authority to update user profiles and details, ensuring that all information is current and accurate across the platform.

5) AI Integration



AI Integration Activity Diagram Description:

The AI Integration Activity Diagram details the processes involved in generating AI-based quizzes.

Teacher can select various categories for AI-generated quizzes, including the subject, difficulty level, and quantity of questions, and initiate the quiz generation process.

Students can join the quiz by entering the Room ID and submit their answers upon completion.

Admin can utilize the AI Model to generate quizzes, create additional questions using the AI model, and manage quiz records by adding them to the system.

The Admin is also responsible for generating Room IDs, displaying quizzes, and showing scores along with correct answers after quiz completion.

This diagram captures the interactions and workflow between teachers, students, and admins in managing AI-generated quizzes.

SEQUENCE DIAGRAMS

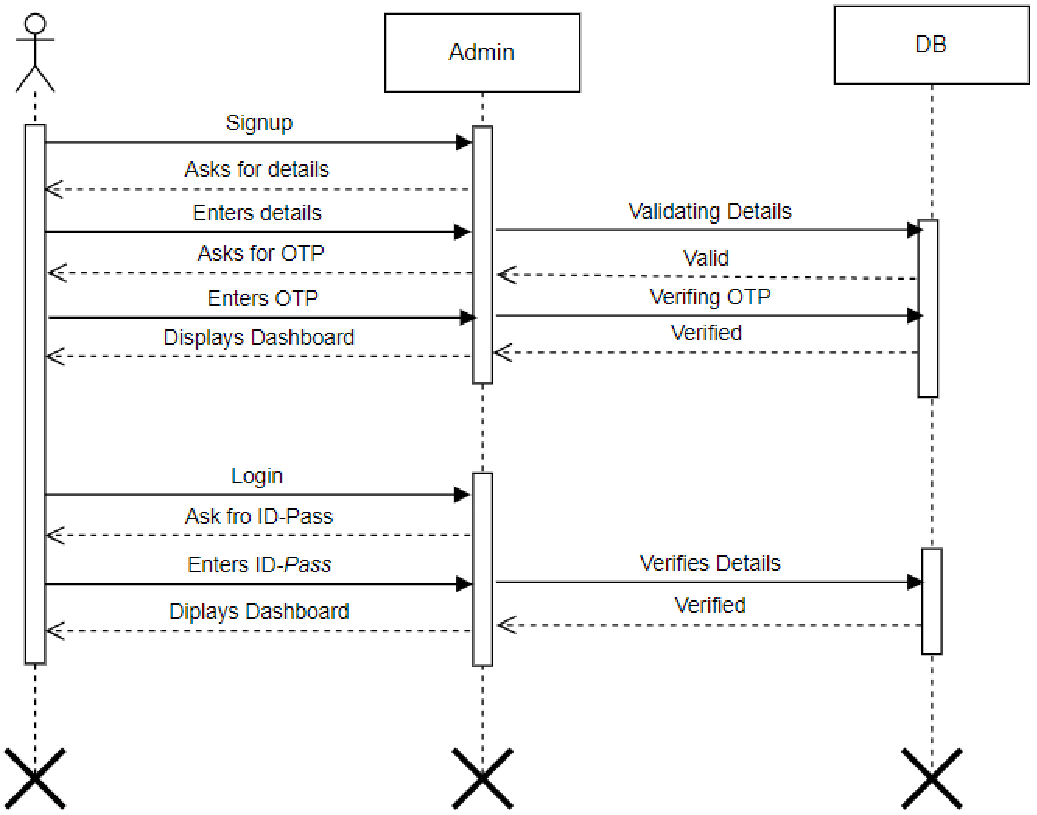
The Sequence Diagram in the project documentation illustrates the dynamic interactions between various actors and the system during specific processes within the application. It captures the flow of messages exchanged among the actors, detailing how they communicate and the order in which interactions occur. Each participant in the diagram, such as Teachers, Students, and Admins, is represented as a lifeline, and the horizontal arrows depict the messages or function calls exchanged during the sequence of events. This visualization aids in understanding how different components of the system work together to achieve a specific goal, providing a clear representation of the system's behavior over time.

The Sequence Diagram further elaborates on the quiz management process, particularly how Teachers, Students, and Admins interact with the system to manage quizzes. When a Teacher creates a quiz, the sequence starts with the Teacher sending quiz details to the system, which then generates a quiz record and assigns a Room ID.

Subsequently, Students can join the quiz by providing the Room ID, which triggers the system to validate their entry and allow access to the quiz. Admins can oversee this process, as they manage records, display quizzes, and monitor participant scores, ensuring that the quiz management workflow is seamless and efficient.

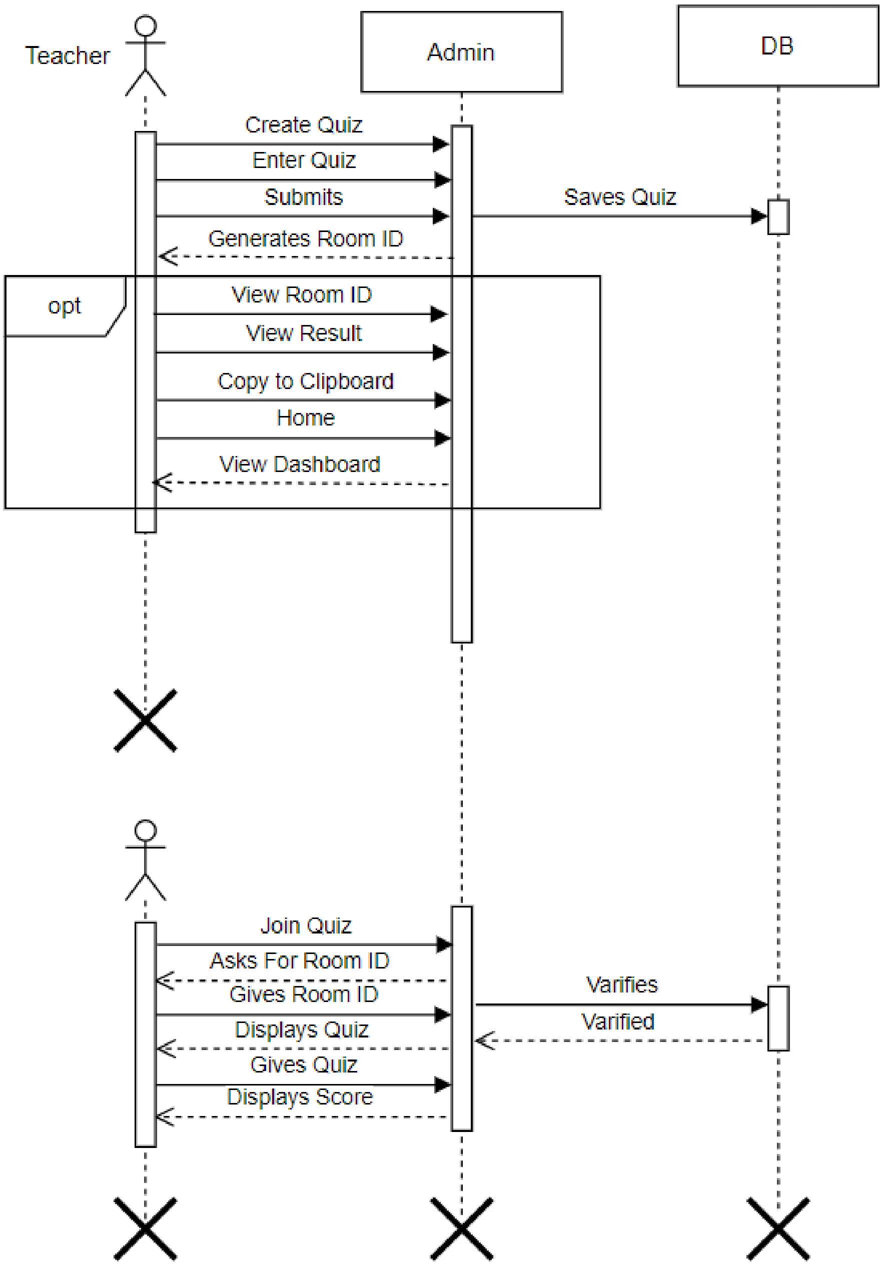
Overall, the Sequence Diagram serves as a crucial tool in the project documentation by providing a clear and organized representation of interactions within the system. It highlights key functionalities such as user authentication, quiz creation, and participation, as well as the roles of different actors in these processes. By outlining the order of operations and the relationships between components, the Sequence Diagram not only enhances understanding but also facilitates future development and troubleshooting efforts. This visualization is instrumental in ensuring that all team members are aligned on how the system operates and the expectations for each user interaction.

1) User Authentication (Login & Signup)



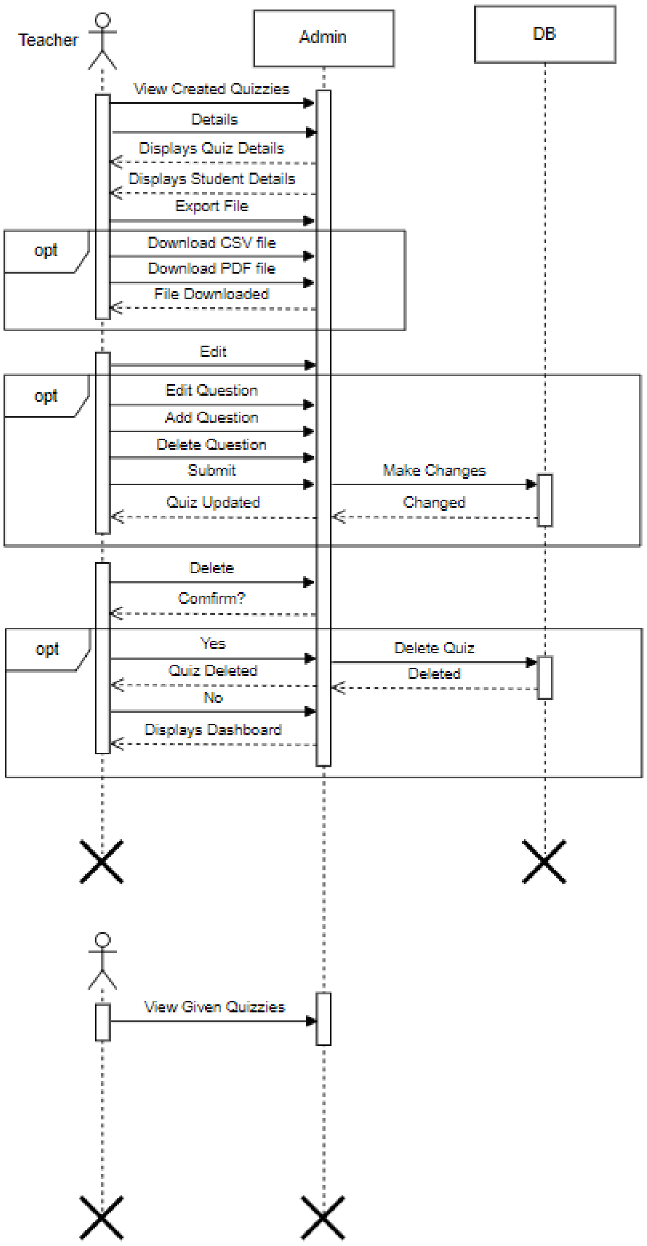
The User Authentication (Login & Signup) Sequence Diagram illustrates the interactions involved when Teachers and Students sign up and log in to the system. Both Teachers and Students initiate the signup process by providing their credentials, which the system validates and creates new user accounts. Once signed up, they can log in by submitting their credentials, prompting the system to authenticate them and grant access to their respective dashboards. Additionally, the Admin plays a crucial role by adding new users, managing existing user accounts, and validating user credentials throughout the process. This diagram effectively outlines the flow of messages and operations involved in user authentication.

2) Quiz



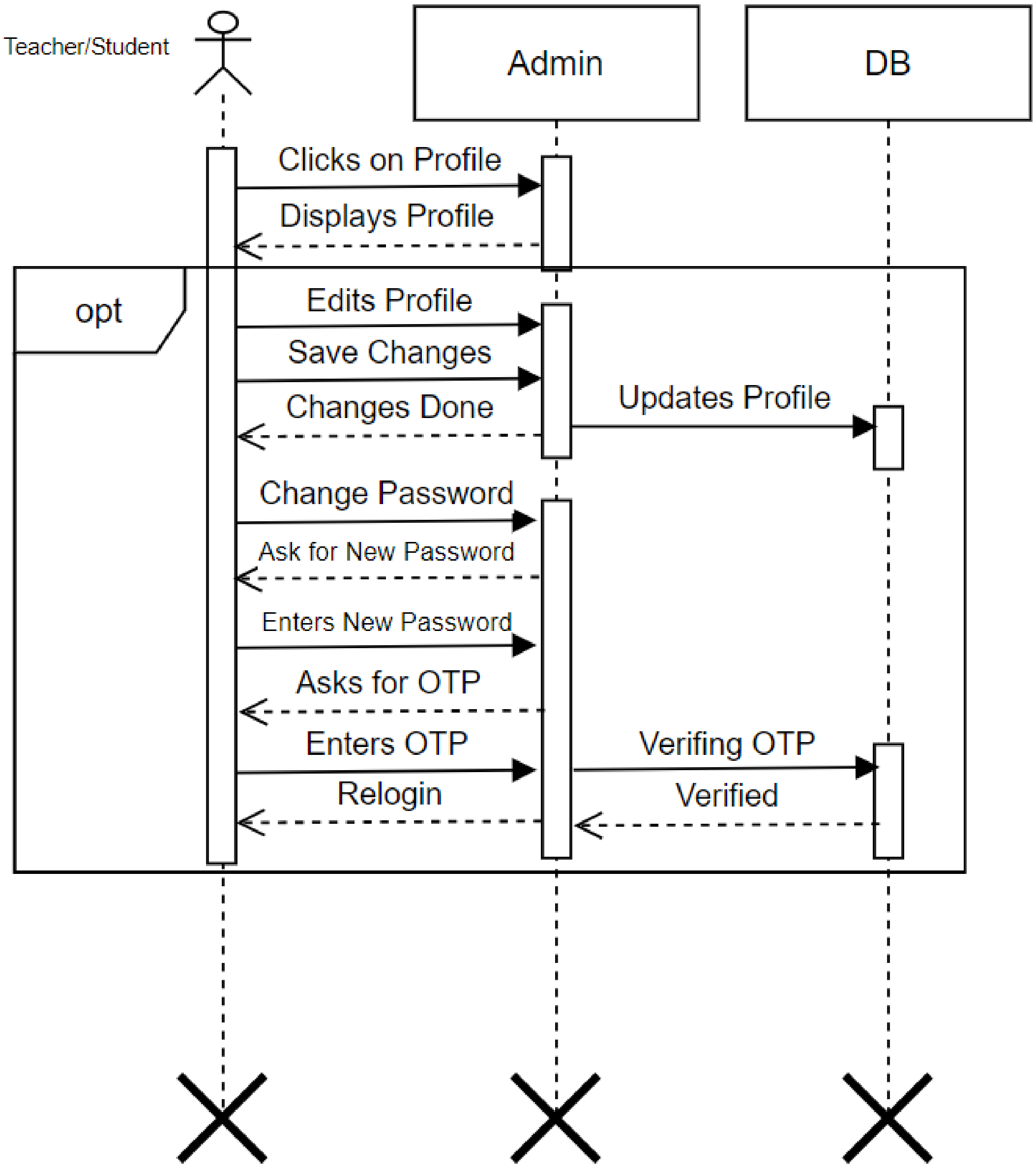
The Quiz Management Sequence Diagram illustrates the flow of interactions involved in quiz management among the Teacher, Admin, and Student actors. When a Teacher creates a new quiz, the system processes the quiz details. The Admin then adds the quiz record, generates a Room ID, and displays the quiz along with the corresponding scores. Students join the quiz by entering the Room ID and submit their responses. This diagram effectively captures the chronological order of operations and the communication between the actors and the system during the quiz management process.

3) **Records**



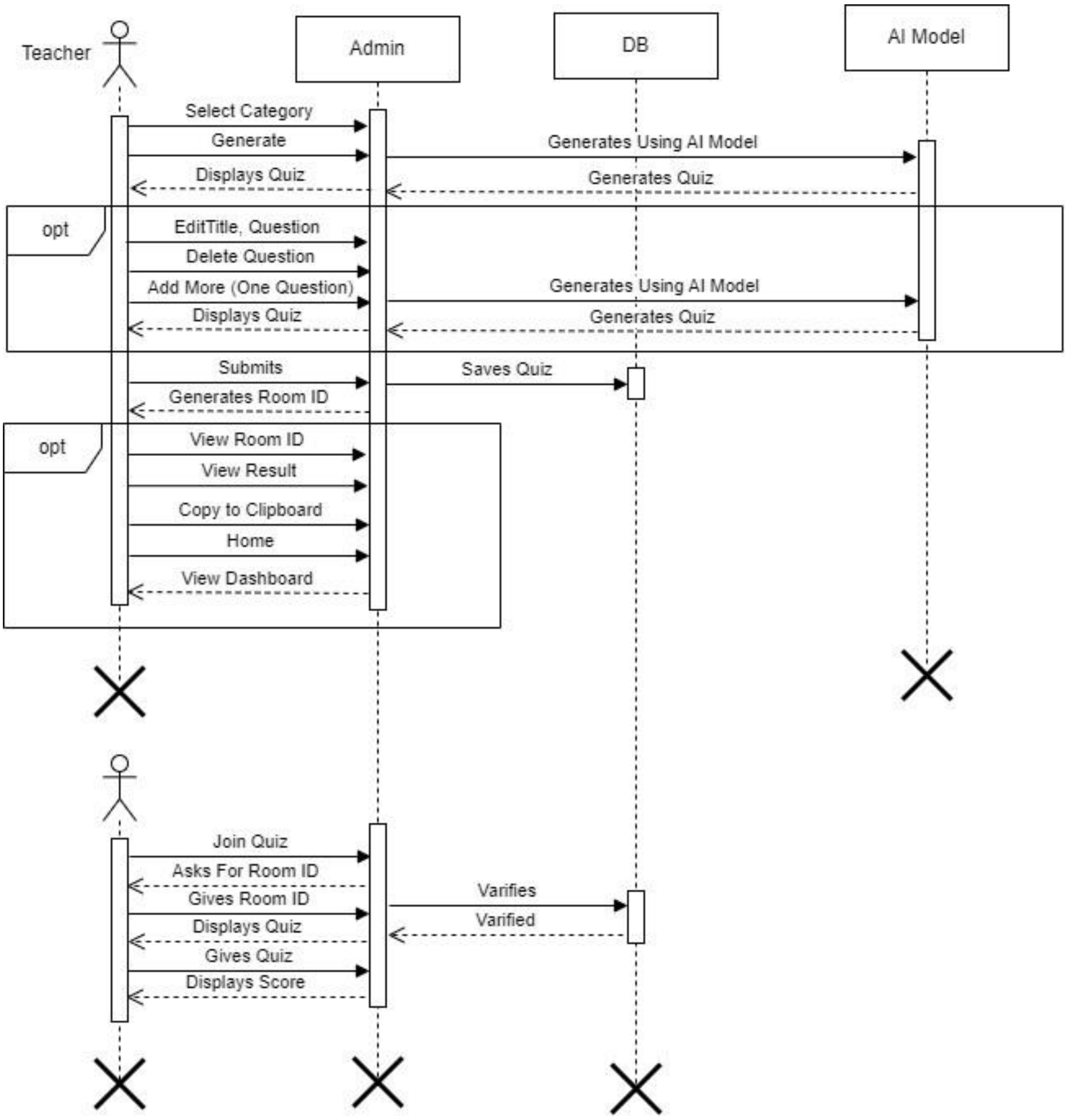
The Record Management Sequence Diagram outlines the interactions between the Teacher, Admin, and Student during the management of quiz records. When a Teacher edits a quiz, the system retrieves the title, code, and date of creation for modification. The Admin can display all created quizzes in a tabular format on the dashboard, remove specific quizzes, and view submitted quizzes from students. Meanwhile, Students can access their quiz records to view the quiz name, score, and submission date. This diagram captures the sequential flow of actions and the exchange of information between the actors and the system for effective record management.

4) Profile



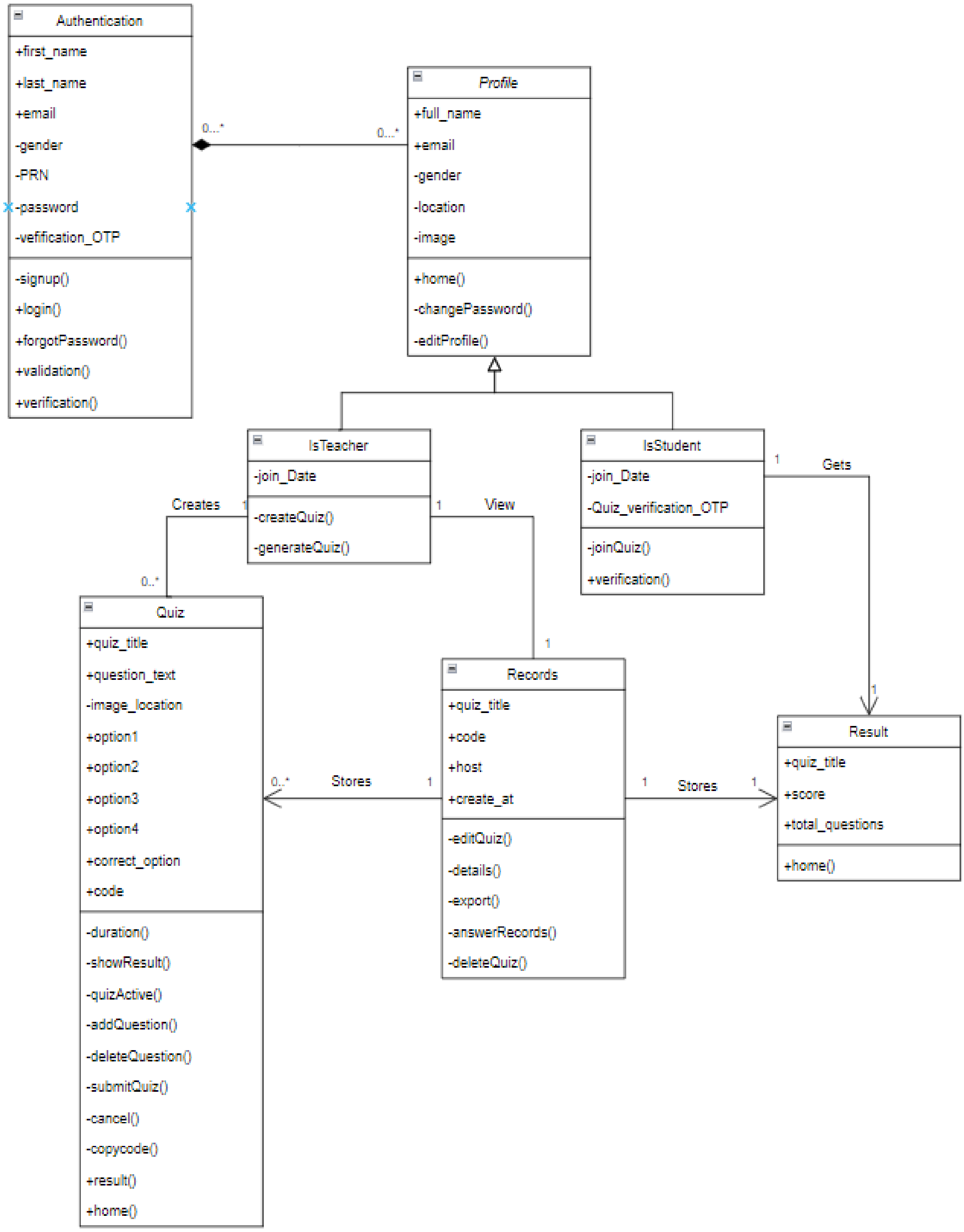
The Profile Management Sequence Diagram illustrates the interactions between Teachers, Students, and Admin regarding user profile management. Both Teachers and Students can view their profiles and edit them, including changing their profile pictures and updating their passwords through OTP verification for enhanced security. The Admin has the authority to update profiles and details for all users. This diagram effectively captures the sequence of actions and communications required for managing user profiles within the system.

5) AI Integration



The AI Integration Sequence Diagram outlines the interactions between teachers, students, and admins during the AI-based quiz generation process. First, the teacher selects quiz categories, including subject, difficulty level, and question quantity, and requests the quiz generation via the AI Model. The admin then utilizes the AI Model to generate the quiz and create additional questions, which are subsequently stored in the system. Students join the quiz by entering the Room ID and submit their answers upon completion. Finally, the admin displays the quiz along with the scores and correct answers after the quiz is completed, ensuring a smooth workflow for managing AI-generated quizzes.

CLASS DIAGRAM



The class diagram for the project consists of seven essential classes: Authentication, Profile, IsTeacher, IsStudent, Quiz, Records, and Results. Each class plays a specific role in the overall architecture of the system, supporting the management of user authentication, quiz creation, and result tracking.

1. Authentication Class: This class handles user authentication and includes key variables such as first name, last name, email, gender, PRN, password, and OTP verification. Its methods include signup(), login(), forgetPassword(), validation(), and verification(), which facilitate user registration, login processes, password recovery, and verification of user details.

2. Profile Class: The Profile class manages user profile information, including variables like fullname, email, gender, location, and profile image. Its methods—home(), changePassword(), and editProfile()—allow users to navigate their profiles, update passwords, and edit personal information.

3. IsTeacher Class: This class inherits properties from the Profile class, with an additional variable for join_date. It includes methods such as createQuiz() and generateQuiz(), enabling teachers to create and generate quizzes within the system.

4. IsStudent Class: Similar to the IsTeacher class, IsStudent manages student-specific data, including join_date and a quizVerificationOTP. Its methods, joinQuiz() and verification(), facilitate student participation in quizzes and verification processes.

5. Quiz Class: The Quiz class is responsible for managing quiz-related data. Key variables include quizTitle, questionText, imageLocation, and answer options. The class also contains methods like duration(), showResult(), quizActive(), addQuestion(), deleteQuestion(), submitQuiz(), cancel(), copyCode(), and result(), which handle various aspects of quiz management, from adding questions to displaying results.

6. Records Class: This class stores and manages records of quizzes taken. It includes variables such as quizTitle, quizCode, quizHost, and createdAt. Its methods—editQuiz(), details(), export(), answerRecords(), and deleteQuizzes()—allow for the editing of

quizzes, accessing details, exporting records, and managing quiz deletions.

7. Results Class: The Results class tracks individual quiz results, containing variables like quizTitle, score, and totalQuestions. Its primary method, home(), allows users to access their results and navigate back to the main interface.

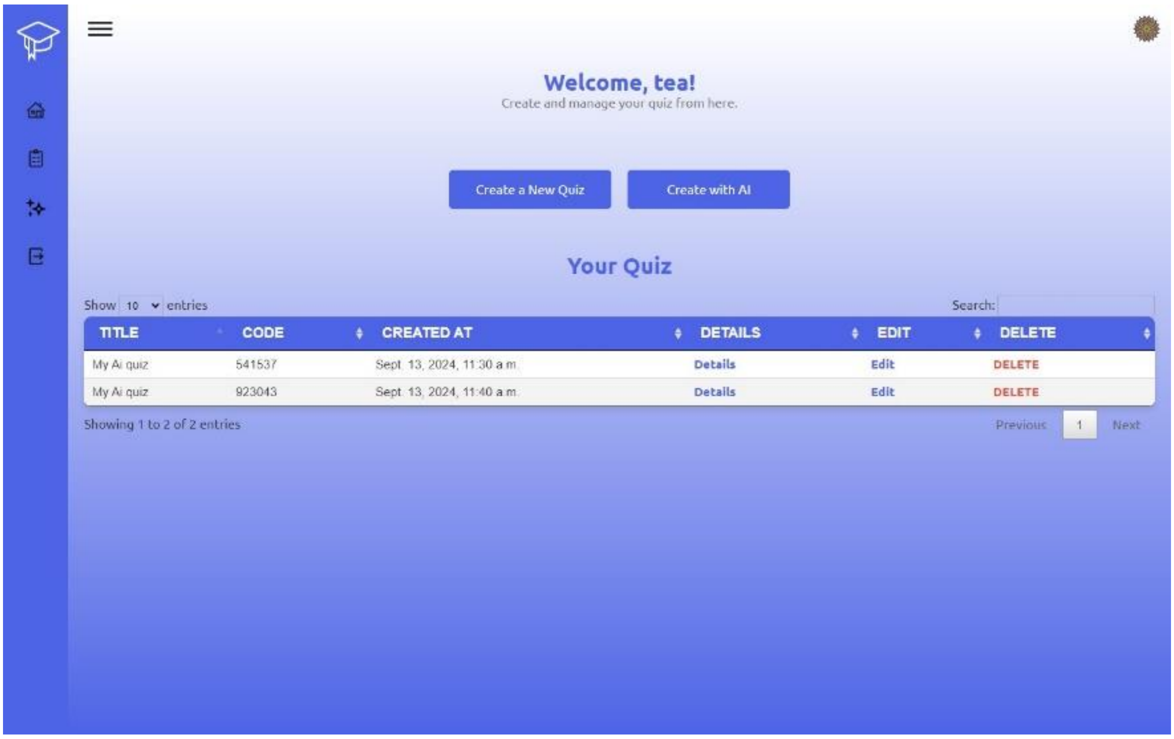
Relationships Between Classes

- The IsTeacher class has the capability to create quizzes and view records related to those quizzes.
- The Records class serves as a repository that stores both quizzes and results for future reference.
- The IsStudent class retrieves results associated with the quizzes they have completed.

This class diagram effectively captures the interactions and relationships among various components of the quiz management system, providing a structured approach to user authentication, quiz handling, and record-keeping.

GLIMPES OF QuizBOT

DASHBOARD OF QuizBOT



Welcome, tea!

Create and manage your quiz from here.

Your Quiz

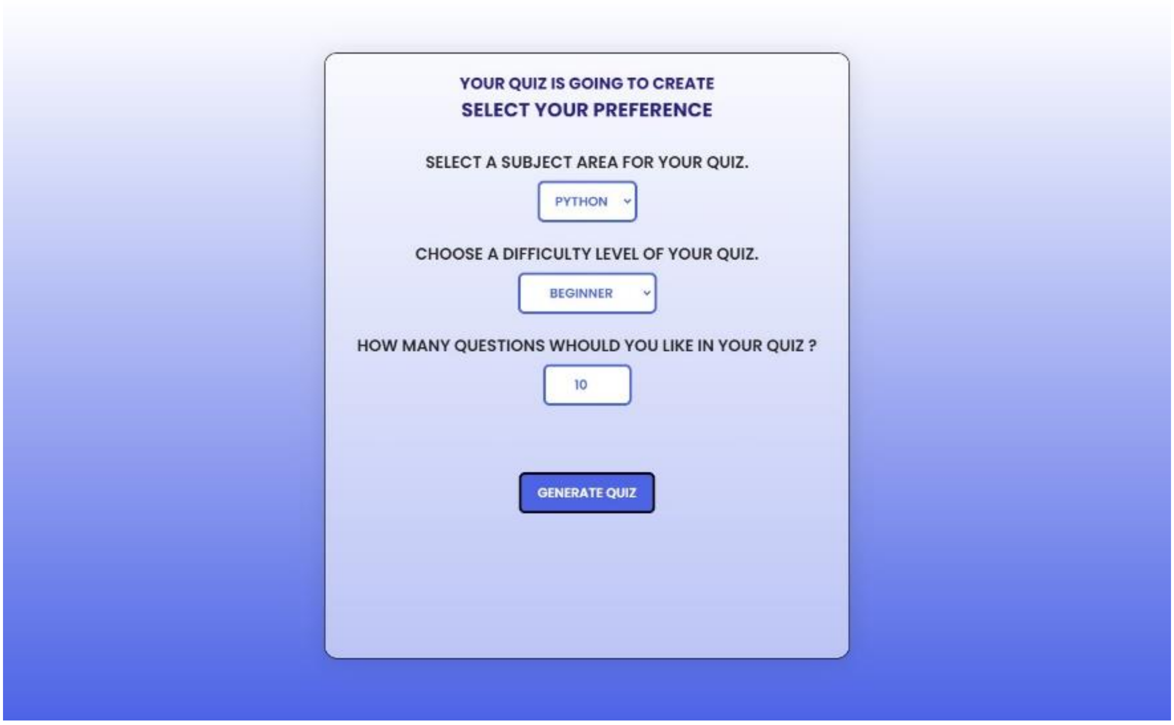
Show 10 entries

Search:

TITLE	CODE	CREATED AT	DETAILS	EDIT	DELETE
My Ai quiz	541537	Sept. 13, 2024, 11:30 a.m.	Details	Edit	DELETE
My Ai quiz	923043	Sept. 13, 2024, 11:40 a.m.	Details	Edit	DELETE

Showing 1 to 2 of 2 entries

AI GENERATIVE QUIZ



YOUR QUIZ IS GOING TO CREATE

SELECT YOUR PREFERENCE

SELECT A SUBJECT AREA FOR YOUR QUIZ.

PYTHON

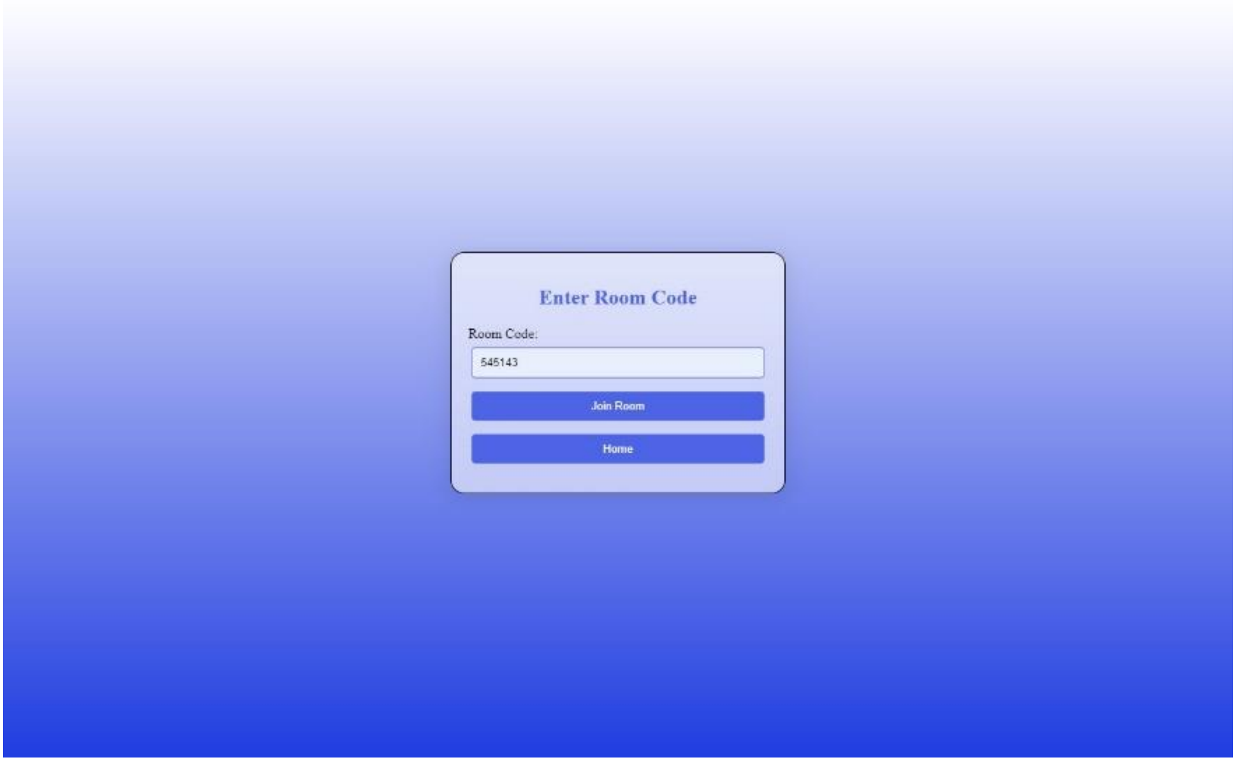
CHOOSE A DIFFICULTY LEVEL OF YOUR QUIZ.

BEGINNER

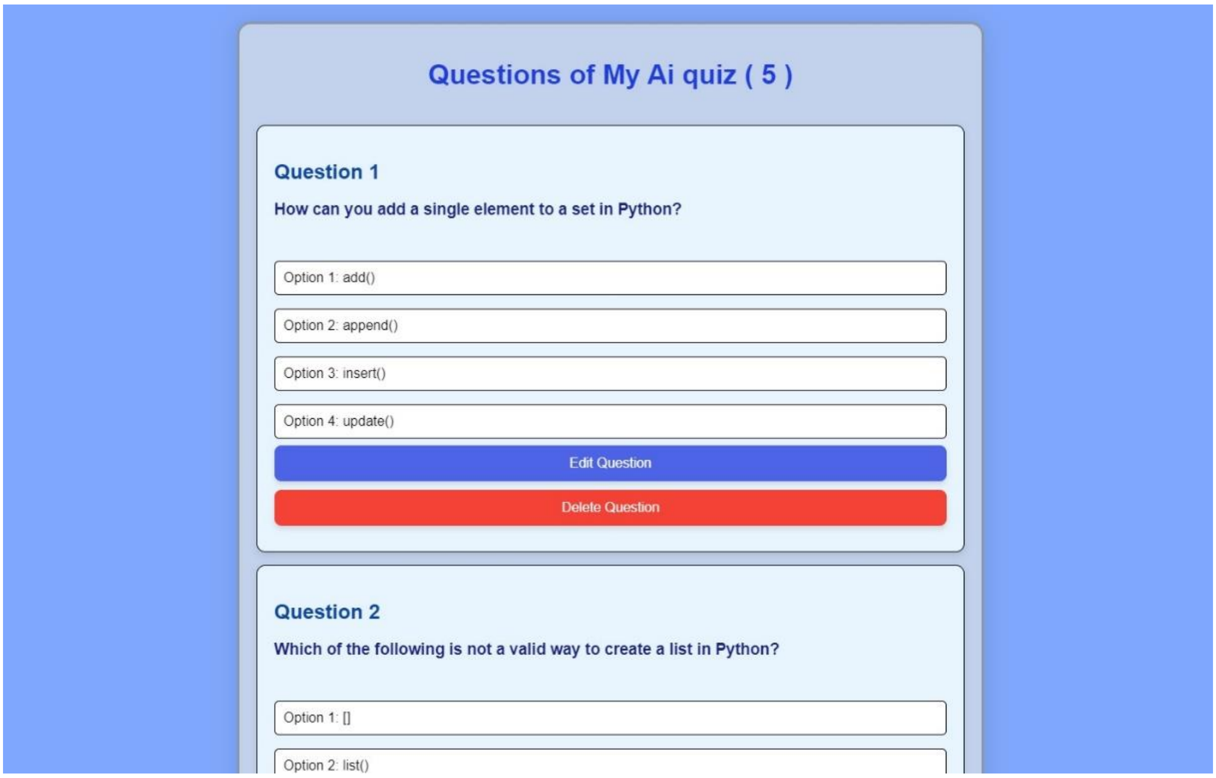
HOW MANY QUESTIONS WHOULD YOU LIKE IN YOUR QUIZ ?

10

JOIN QUIZ USING ROOM CODE



MANAGE QUESTION



Questions of My Ai quiz (5)

Question 1

How can you add a single element to a set in Python?

Option 1: add()

Option 2: append()

Option 3: insert()

Option 4: update()

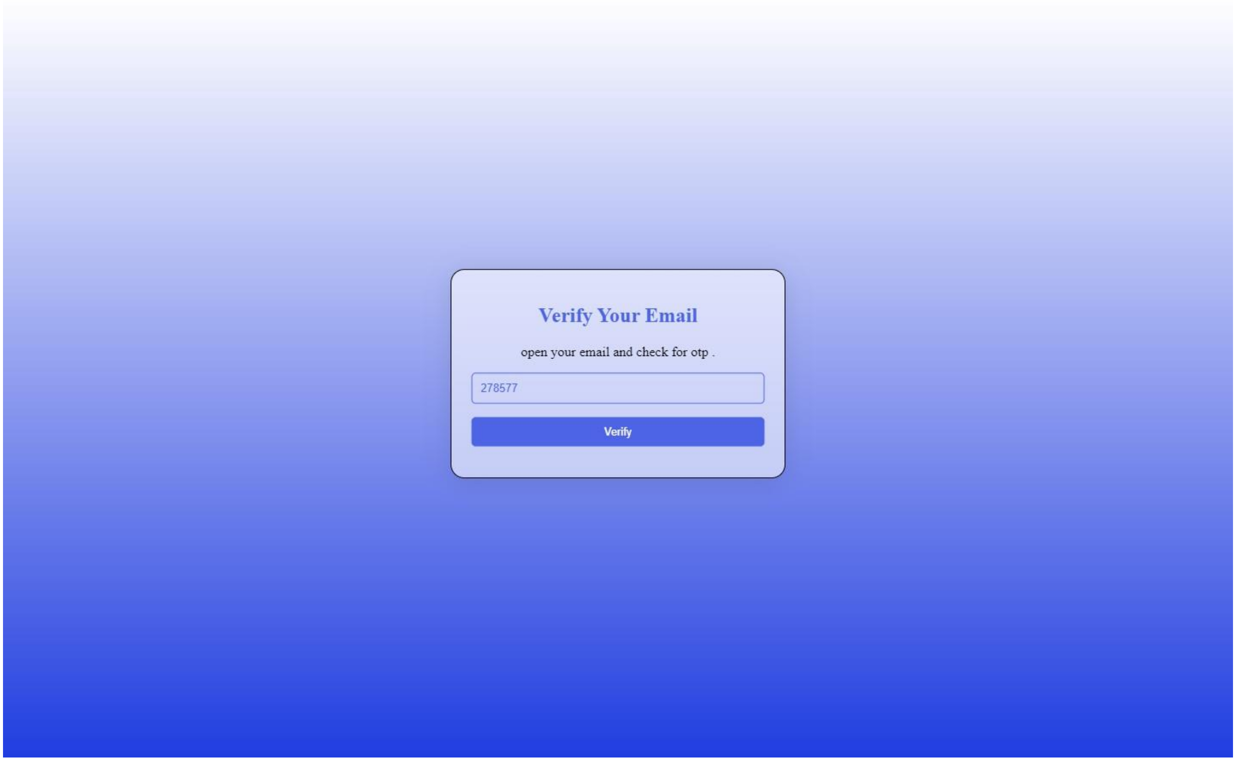
Question 2

Which of the following is not a valid way to create a list in Python?

Option 1: []

Option 2: list()

EMAIL VERIFICATION

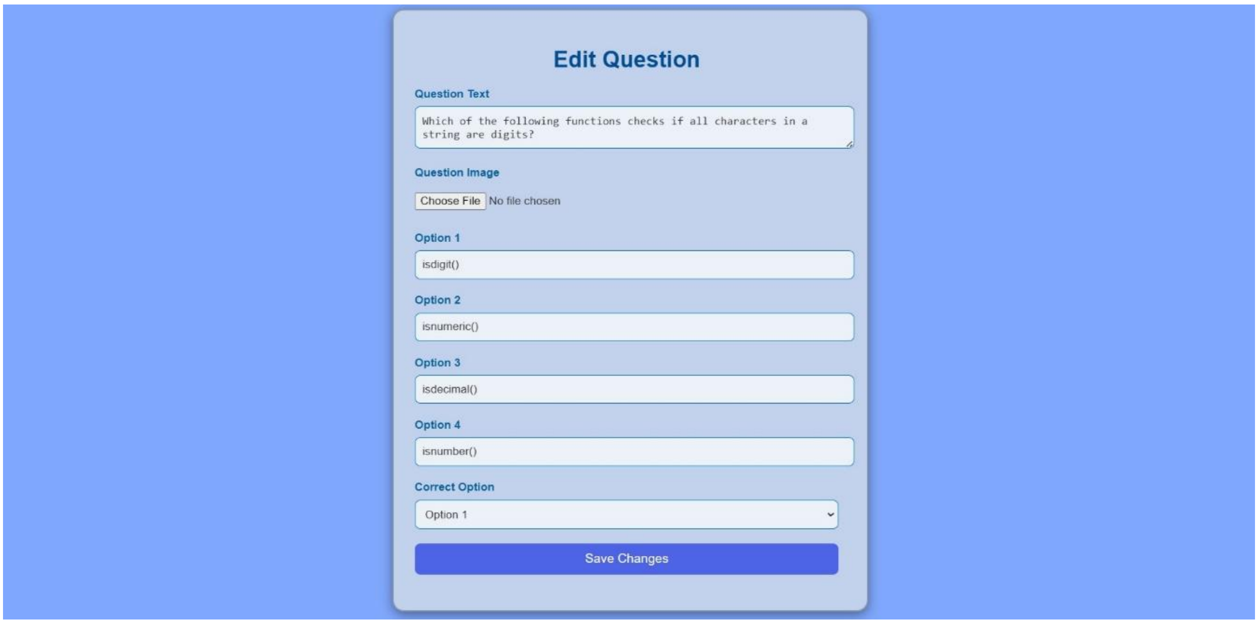


Verify Your Email

open your email and check for otp .

278577

EDIT QUESTION



Edit Question

Question Text

Which of the following functions checks if all characters in a string are digits?

Question Image

No file chosen

Option 1

isdigit()

Option 2

isnumeric()

Option 3

isdecimal()

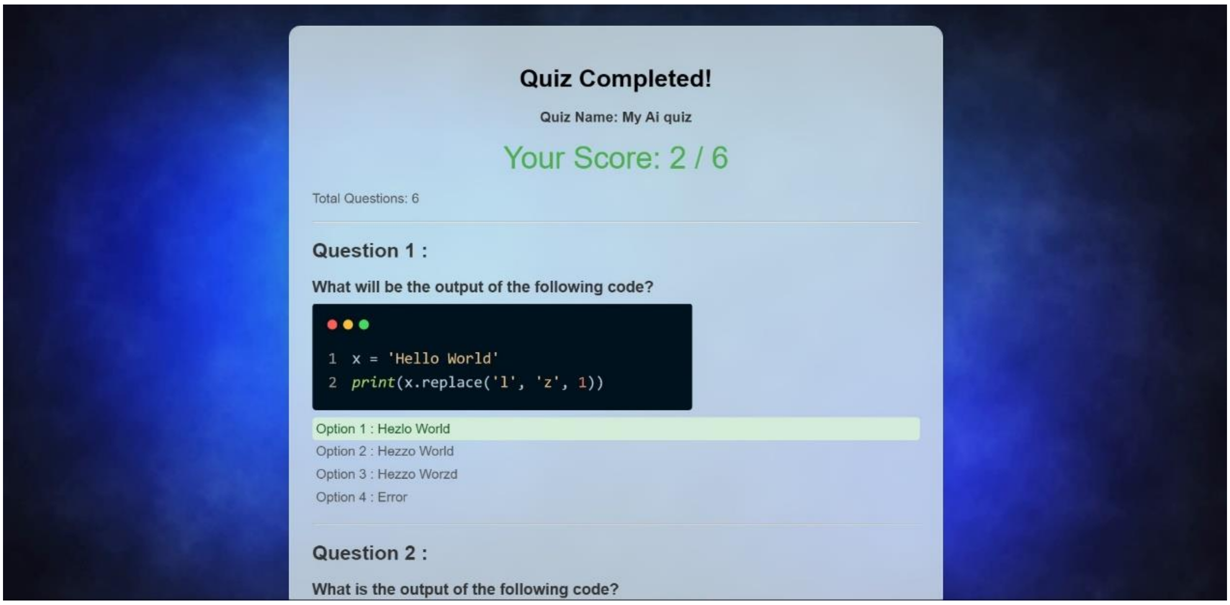
Option 4

isnumber()

Correct Option

Option 1

RESULT



Quiz Completed!

Quiz Name: My Ai quiz

Your Score: 2 / 6

Total Questions: 6

Question 1 :

What will be the output of the following code?

```
x = 'Hello World'
print(x.replace('l', 'z', 1))
```

Option 1 : Hezlo World

Option 2 : Hezzo World

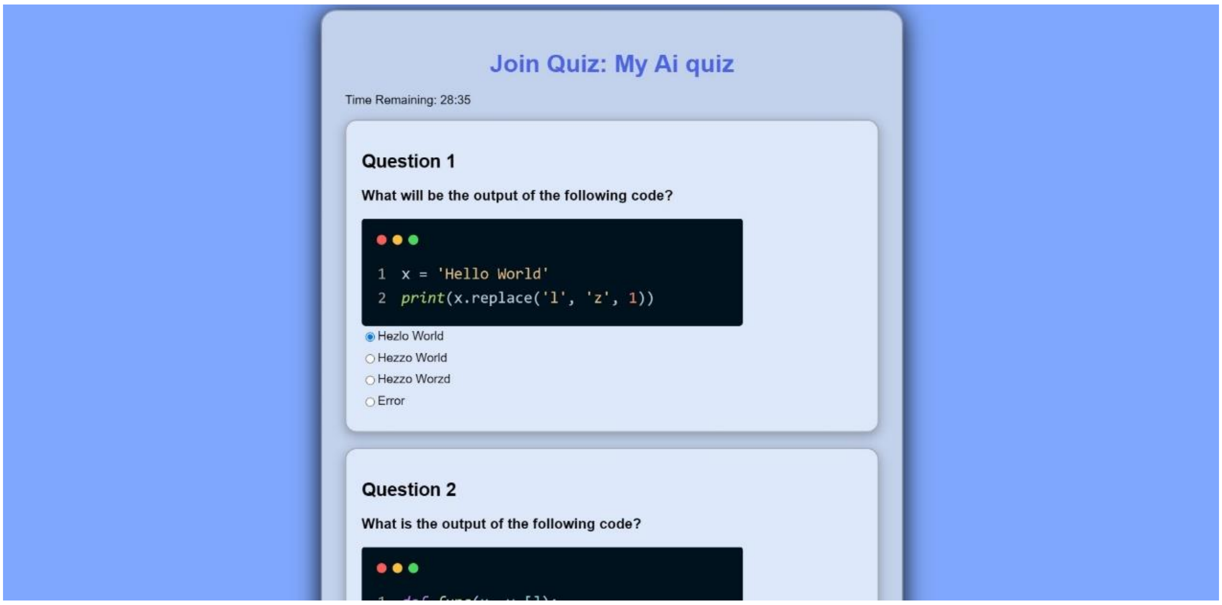
Option 3 : Hezzo Worzd

Option 4 : Error

Question 2 :

What is the output of the following code?

JOIN QUIZ



Join Quiz: My Ai quiz

Time Remaining: 28:35

Question 1

What will be the output of the following code?

```
x = 'Hello World'
print(x.replace('l', 'z', 1))
```

● Hezlo World
○ Hezzo World
○ Hezzo Worzd
○ Error

Question 2

What is the output of the following code?

***