University Of West Attica

Department of Informatics and Computer Engineering

Master's Thesis

Development of an educational web application with integrated digital assistant

Christos Bitas

Georgios Prezerakos, Professor

October 2024

ABSTRACT

The present thesis concerns the development of a web application (Full stack) with educational content by giving the user questions (Quiz) in order to deepen his knowledge in the subject of computer science, providing him with a digital assistant of artificial intelligence (AiBot) that he will be able to interact with in real time and solve important inquiries during the questions that will be given to him.

This educational web application was developed at the front-end and back-end level using modern application development technologies. For the front-end part React (JavaScript library) was integrated. In the part of the management and storage of all data of the application (Back-end) the popular programming language Java was used along with the Spring Boot framework. The data was maintained in a database using the powerful tool PostgreSql which gives the capability of storing, managing and retrieving the data. Botpress which is an open source virtual assistant (chatbot) development platform was used for the digital assistant.

In conclusion, this thesis aims to present the web application that will provide educational content to the user giving him knowledge and at the same time effective interaction between the user and the application interface utilizing modern programming tools.

SCIENTIFIC AREA: Web Applications, Artificial Inteligence
KEYWORD: Fullstack, E-Learning, Educational Software, Aibot

TABLE OF CONTENTS

FIGURES

ABBREVIATIONS

UI = User Interface

UX = User Experience

API = Application Programming Interface

1. Introduction

The rapid evolution of technology has brought the need to adopt new digital media in the field of education, with online applications being updated and offering more and more resources in the way learning is done. Artificial intelligence is becoming increasingly popular, providing important tools that aim to enhance and provide a personalised user experience. The need to access information in real time while providing simultaneous guidance has resulted in the integration of digital assistants into web applications. As a result of this process, is the provision of a more personalized education giving even better tools to both the instructor and the learner. Through this thesis, the development of an educational web application containing an integrated digital assistant will be presented, offering a more comprehensive learning experience proposal.

1.1. The subject of the thesis

The subject of the thesis is the development of an innovative educational web application, which will provide the user in game form with quiz-type questions with the content of the cognitive subject of computer science in combination with the integration of a digital assistant. More specifically, the quiz will include questions covering knowledge related to the field of computing with the aim of consolidating and understanding concepts to better enhance the users' knowledge through a more enjoyable and at the same time interactive process. The existence of the quiz was made with a view to encourage users to participate more actively, making learning more engaging and fun.

The innovative feature of the application is the use of artificial intelligence, which is integrated into the digital assistant, acting as a supporting tool for the user during his interaction with the application, advising him and providing him with useful information for solving the questions he will be given. In addition, the digital assistant will be able to accept any form of question by generating personalised answers based on the user's history. A feature of the digital assistant is the ability to communicate in real time, enabling users to ask for clarification or further information on any questions that may arise during the game.

The application will provide the user, upon completion of the quiz, with a report on the progress made for him/her and the option to see where he/she ranks in relation to the total number of users. This application is aimed at pupils, students and at the same time professionals who may be interested in enhancing their knowledge of computing using a modern and innovative way of learning.

To conclude, significant emphasis in this thesis is given to the way in which the application uses and manages the two levels Front-End and Back-End through which modern technologies were utilized in order for the web application to operate based on the most modern programming tools.

1.2. The objectives of the thesis

The objectives of this thesis on the development of an educational web application with an integrated digital assistant are multidimensional and aim to provide an innovative platform from which the user will have an optimal experience. The application incorporates modern programming technologies and methods combined with artificial intelligence in order to be able to provide real-time feedback to the user by answering any query.

The main and fundamental goal of the work is that the final application can be easy to use, simple and understandable to user. The quiz gives the opportunity through the questions to provide knowledge related to both the web development part and the general subject of the information technology. In addition, the possibility of displaying the user's progress either on an individual level or as a whole will help the user to improve and see up to where he can enhance his learning level as much as possible.

The digital assistant was integrated into the application making it a modern tool to support users. More specifically, the assistant with the use of artificial intelligence utilizing the resources of the whole internet will be able to provide feedback to the user, guidance but also clarifications on the quiz game questions. The digital assistant is a means of support by generating personalised answers based on the user's performance.

The choice of learning in the form of quiz questions is an effective way and incentive in order to increase the participation and interest of the user to participate. Installing special programming functions in the application to calculate points (score), progress as a whole (scoreboard) is intended to improve the user and have an insight into his performance so that he can improve and become better. Through these mechanisms, users will have even more interest to continue participating in the quiz, as learning will be done in a more fun and interactive way.

The web application leveraged artificial intelligence to be able to provide a personalized learning experience with the support of the digital assistant. The aim is to treat all users at the same level of support while emphasizing that the material provided is appropriate based on the content of their queries.

In addition to the learning content, the questions will aim to develop critical thinking and create questions aimed at solving the problems given to the user.

The existence of lifelong learning will provide an environment for the user, which will allow for a continuous education with the possibility of upgrading skills in the field of information technology subject.

In conclusion, this thesis aims to highlight modern programming methods and educational tools to improve learning at different levels and contexts in any subject area. By achieving the above mentioned objectives, the thesis aims to provide a comprehensive and optimal solution for online education by utilizing artificial intelligence and game-based learning.

2. Description of the system's architecture

Chapter 2 will provide programming language definitions with the features used and a complete analysis of the system architecture of the web application. The system architecture of the web application system with an embedded digital assistant is based on a modern approach to web application development and therefore cutting-edge technologies were selected to provide an integrated and efficient educational platform. The application was developed both at the Front-End and Back-End level. In the Front-End part, the React library was used to develop an interactive and user-friendly interface, offering dynamic responsiveness and optimized user experience. For the Back-End part, the popular Java programming language was used along with the well-known Java Web Framework Spring Boot, ensuring security in user connection using encryption, stability and ease of expandability of the application to address future issues. The application data was maintained in a database utilizing PostgreSQL providing reliable and optimal data management. The overall architecture follows a layered approach, providing the ability for easy maintenance and possible future system upgrades to address future challenges. In the following, an extensive discussion will be given on the concepts of Full stack application, API and at the same time the functionality and the way the Front-End and Back-End sections were developed.

Full stack web development is the scenario of simultaneous operation and communication between the two Front-End and Back-End parts of a web application. The comparative term is widely known largely for those who are active as developers in the web development part. In this case, developers have a background of knowledge in the form of code to create the interface that the user will interact with and the display of data with a particular focus on the appearance of the web page (Front-End). In addition, they can ensure the management and storage of data while applying security techniques to cover any gaps in the application. The Front-End section is where users can see or interact with the application.

On the other hand, the Back-End part is not visible to the users, as this part is where the application logic, database, server and other features that are needed in order for the application to run smoothly and seamlessly are implemented. The main features of Full Stack Development are Front-End, Back-End and database usage.

In the Front-End part of the design, the developer focuses on both the user experience (UX) and the user interface (UI). While they may look similar, they are two separate concepts where each refers to a specific type of functionality. Features of a web application that contain visual elements, animations, images, videos, etc. within the website are difficult to create and are an example of good UI design but poor UX and vice versa. An intuitive experience that does not require the user to have specialized knowledge of web usage is a good way to design a website. From the developer's point of view, the Front-End is the part that users see when they visit the website and is mainly concerned with the design and appearance of the website using some programming method [1].

As for the Back-End part, this part unlike the Front-End is hidden from the users, but is normally involved in the application development process. It consists of the server and the application. The server is called to handle all the requests coming from the application. The source code of the application activates its parts, for example calling specific functions based on the server's requests. Therefore, the server is constantly synchronized with the application code. References within the code, for example async-wait, try-catch, sync are an indication of such uses. The server is called to handle all types of requests by giving its response on the client side (Front-End) or the server side (Back-End) depending on the types of callbacks present in the code. The server architecture consists of requests - responses in order to be able to communicate so that it can perform synchronization successfully and achieve smooth operation of the application. The application on its side acts as an intermediary communication medium between the server and the database by sending all the data and requests that the server or the database needs in order to receive the required response. Generally, the application contains all the code that is logical and functional which is executed on the application's website server based on the specific requirements of the website and the application in general.

Modern web applications use an API. The term API (Application Programming Interface) is defined as the set of protocols that enables software applications to be able to communicate with each other. Through an API, the methods and type of data that applications will have in order to pass information are defined giving many advantages to developers to create systems that can interact with each other.

The basic concepts of an API are as follows [1]:

1. Endpoints: Are specific addresses (URLs) through which certain services are accessed.

2. Requests: An API enables an application to be able to make requests to another application.

Requests utilize HTTP methods such as [1]:

- GET: Retrieving data from a server.
- POST: Sending new data to the server.
- PUT: Update data on the server.
- DELETE: Remove data from the server.

3. Responses: When the API performs a request reception, then it gives a response in a JSON or XML format. This response usually consists of the requested data or confirms the success/failure of the original request.

4. Authentication: APIs enable authentication of a user's details using special tokens to ensure that only authorized users or applications can access the particular service ensuring smooth operation of the application.

5. Rate Limiting: It helps to limit the frequency of requests in order to avoid overloading the service. This is called rate limiting (e.g. 100 requests can be made per minute).

There are many types of APIs, some of which will be discussed below. More specifically, Web APIs, which are the most popular, allow communication over the Internet, usually using HTTP or HTTPS protocols. Applications that use such APIs are social media platforms (e.g. Twitter, Facebook). There are Library/API Frameworks which give application developers the right to integrate specific software libraries, such as JavaScript libraries. Another type of APIs are Operating System APIs which relate to operating systems by allowing applications to interact with system functions, such as file systems or hardware components. APIs are important as they provide many tools for developers to integrate pre-existing services into their system instead of developing them from scratch which would increase the difficulty and at the same time the overall production time of the project. The main advantage of using them is that they allow communication between different systems and also automation of complex tasks and processes. In summary, an API acts as a channel in applications to allow interaction with other systems [1].

Spring Boot was used for the needs of the Back-End section. It is one of the most powerful and popular frameworks based on the Spring ecosystem, providing the ability to build applications using the popular Java programming language with minimal customization. One of the features of Spring Boot is its simplified connection to databases via JDBC APIs, allowing for fast and efficient data management. Spring Boot provides auto-configuration, making it easy to use tools such as Hibernate to manage database. The framework itself can generate code based on the needs and specifications required by the application which makes it ideal for rapid application development. Developers can focus more on the logic of the application and less on the complex details of the configuration involved in the framework. It has great support through the Spring community, provides a plethora of pre-installed function tools such as Spring Security and Spring Data, having compatibility with modern architectures such as microservices which makes it ideal for developing web applications [2].

In this training application, the JWT (JSON Web Token) authentication technique was used to secure the user's connection to the application, allowing users to have a secure connection [3]. In addition, with this technique, users were divided into two roles, the administrator (admin) and the ordinary user. The JWT Authentication function works based on the following steps [3]:

1. Authentication: the user enters his/her email and password in the Login page of the application. Then, if the user is detected in the database they are verified on the server and the user successfully logs into the application.

2. Generating JWT token: If the user is successfully found then a JWT token is generated. JWT token is an encrypted data set that has information about the user (claims) such as user ID or role for example if it is user or admin and then signed with a secret key to ensure its integrity.

3. Sending the JWT token: The JWT token is sent to the client (for example to a web application) and saved into the local storage or as a cookie.

4. Sending Requests: On the client side if the client needs some protected content or is going to perform some action then it sends the JWT in the request (usually inside the Authorization Headers with the Bearer prefix).

5. Token Verification: Next the JWT token is verified by the server. A signature check is performed and if the signatures match then it approves the token.

6. Responses: If the JWT is approved by the server then it sends a response from its side and enables access to the protected content of the application. If the token is valid, the server responds and allows access to the protected resources of the application. Otherwise, it rejects the request and thus denies access to the user [3].

The structure of the JWT consists of three parts [3]:

1. Header: Contains the JWT token type and the signature algorithm.
2. Payload: Contains the information (claims) about the user.
3. Signature: It is generated based on the cryptography used by the application in order to confirm the integrity of the JWT token. In this educational application, the HmacSHA256 algorithm was used in this training application.

A feature of JWT is that no information needs to be stored on the server regarding the user, as all the necessary data is contained within the token. This has the effect of making it efficient and scalable for large web applications. Some of the advantages of JWT Aunthentication are providing security to the data since it is encrypted and thus secures the potentially sensitive information of the user. The server is not overloaded by sessions, it works in a scalable manner in order to enable the system to handle a large volume of connected users in real time. Overall, JWT authentication is one of the most secure and effective methods of user identification for modern applications [3].

PostgreSQL was used to store the data for the educational application. It is a powerful, open-source system quite popular for database management in modern applications. The function in this educational web application is to store the user's data, for example username, password, email and also his personal score from the game in a table named users. In addition, a table named quiz_questions was created, in which the quiz questions were stored. In general, PostgreSQL manages relational databases (RDBMS) by providing reliability, expandability and support for advanced functionality. It has been continuously developed for over 30 years and is considered one of the most stable database systems. In addition to managing relationships between data, it also supports data in JSON format, making it ideal for hybrid applications. It has full support for the SQL standard, drivers for programming languages (for example Java), and extensions such as PostGIS for which adds support for geographic objects to the database. Finally, PostgreSQL is popular for its strong security, ease of expandability, and flexibility, making it ideal for maintaining data in modern applications [4].

For the Front-End part, the popular JavaScript library React was used, which was developed by Facebook and is used to create interactive, fast and efficient user interfaces (UI), mainly in Single Page Applications (SPA). It is based on a component-based architecture, where the UI is broken down into small, reusable and independent components, which can manage their state and communicate with each other depending on how the developer defines them. This model enables better organization and expandability of the code, so that web applications are more maintainable and understandable. One of React's most important assets is its use of the Virtual DOM, which allows the user interface (UI) to be updated by updating only the parts that have changed, as opposed to updating the entire page. The result of this process is faster and more efficient applications, especially in environments that require a constant update of their data. In addition, React supports one-way data flow, resulting in data

being transferred in a controlled manner from parent to child component. It allows for greater control over managing the state of the application, making it easier to understand how data and UI changes. Another important feature of React is the use of JSX, a syntax that combines JavaScript and HTML commands. Developers will be able to develop code that is both readable and understandable, making UI development more efficient. React comes with a large number of tools, such as the React Router for managing navigation between pages. These tools make React ideal for creating modern web applications, which can easily manage large amounts of data and complex interactions [5].

The well-known JavaScript library Axios was used to create the HTTP requests in the educational application in order for the data to be able to communicate with both the Back-End part and the Front-End. Some of the features it includes are as follows [6]:

1. GET/POST requests: It enables sending requests to the APIs, for example downloading data (GET request) or sending data (POST request).

2. Automatic JSON conversion: Data sent and received from the server is converted to JSON at the same time.

3. Interceptors support: It is allowed to handle requests and responses before they appear in the UI.

4. Error management: Provides easy-to-use functions to monitor and handle errors that may occur during the creation of requests.

Its ease of use and configurability based on the application specifications make it popular for communication between the Front-End and Back-End departments [6].

Ready-to-use modern software and platforms were used to develop the application in order to develop the Front-End, Back-End, database management and API testing to ensure the integrity of data transfer. Regarding the Front-End, Microsoft Visual Studio Code, one of the most popular code development platforms for web applications and interfaces, was used. More specifically, NodeJs was installed. A vite-project was created using the npm create vite@latest command in the Microsoft Visual Studio Code terminal and given as project name: eLearningWebApp-Frontend and package name: elearningwebapp-frontend, selecting framework: React and Variant: JavaScript. Microsoft Visual Studio Code offers a wide range of support for both programming languages and functionalities related to the Front-end part, for example HTML, CSS, JavaScript, and frameworks such as React. Providing support for modern tools and libraries helps the development of user interfaces to be faster and more organized, while providing specific tools for debugging.

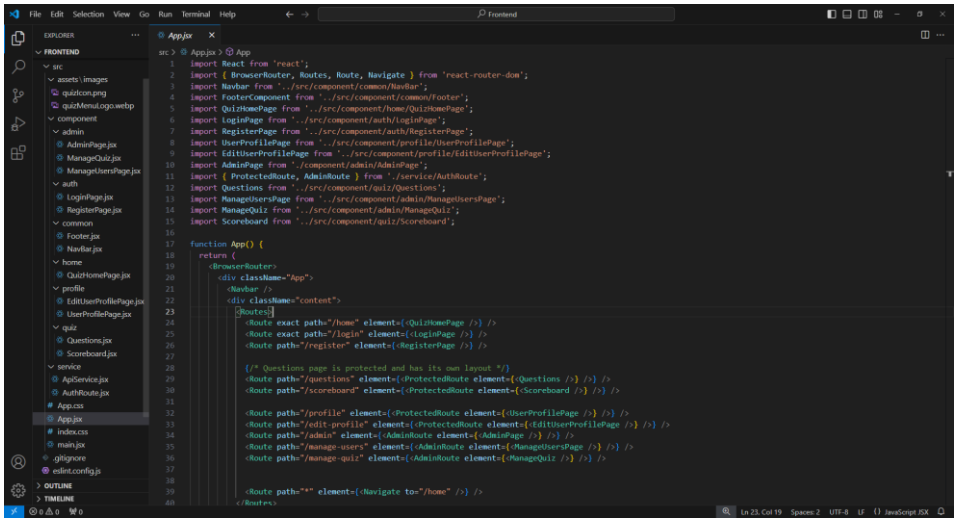

Figure 2.1: Microsoft Visual Studio Code

For the part of the code that deals with the Back-End part of the educational application, IntelliJ IDEA Ultimate integrated development environment was used together with the Java language, which is suitable for the development of scalable and stable Back-End applications. IntelliJ Ultimate provides a rich environment with tools, for example auto-complete code, debugging capability at any point in the code the developer wishes to perform and support for frameworks, such as Spring Boot, that facilitate the creation of applications that need communication to transfer and store application data with databases and APIs. In addition, this IDE allows the organization of code and the management of data flow between users and the server.

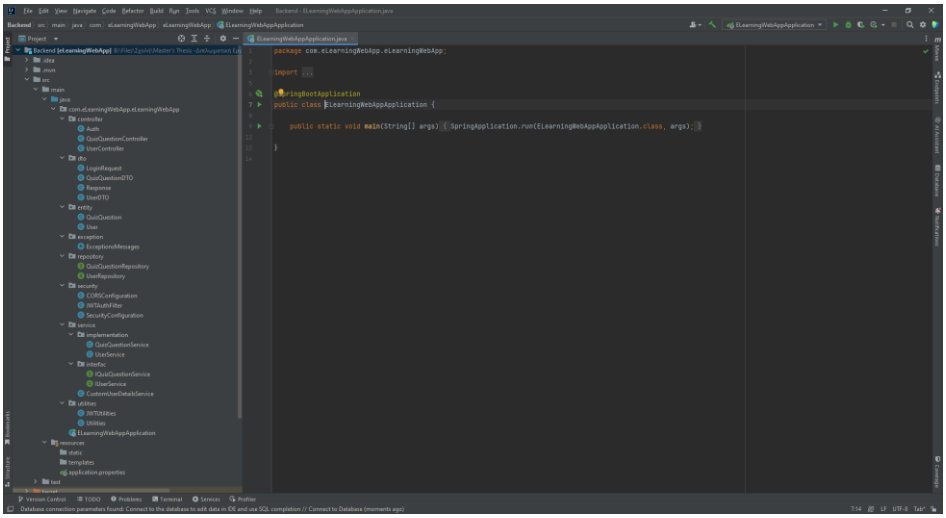

Figure 2.2: IntelliJ IDEA Ultimate

To manage the database, PostgreSQL was used, which is a powerful open source relational database and is specialized for complex systems and data storage. The database was managed through the phpPgAdmin4 application, having an easy-to-use graphical interface for running queries, managing tables, and managing data in general. The ease of use of this tool helps the developer in optimal and efficient data management.

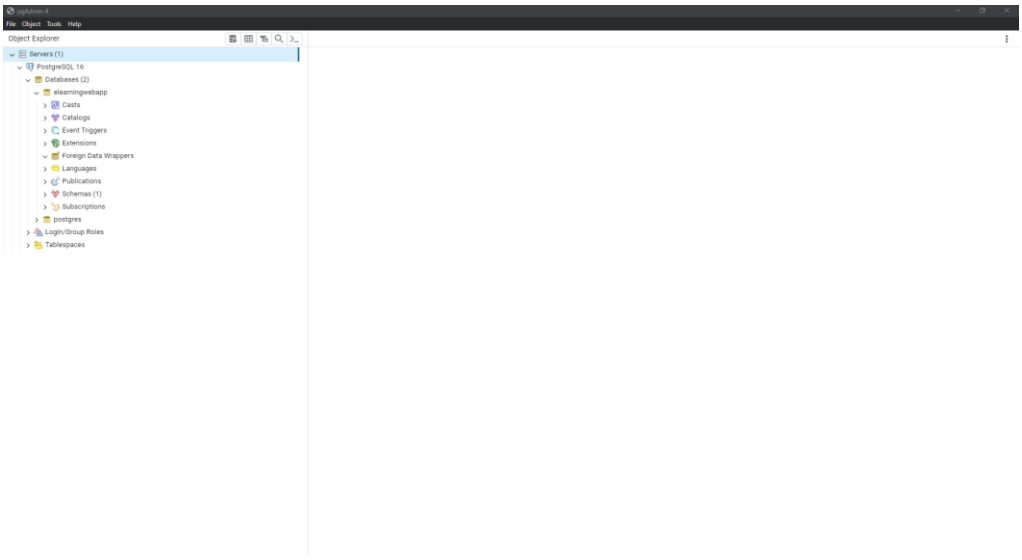

Figure 2.3: PgAdmin4

To test the APIs and generally ensure that the Back-End code part of the project performs the actions correctly, answering the requests with the right data, the well-known Postman application was used. It is a tool that helps in sending HTTP requests, testing the different endpoints of the code and verifying the server responses. Postman ensures with the messages the status of the APIs, so that the developer can see with specific reports and error messages if the APIs are working correctly, but at the same time can perform quick tests in real time to ensure that the code functions return the expected results and check the security of the data [7].

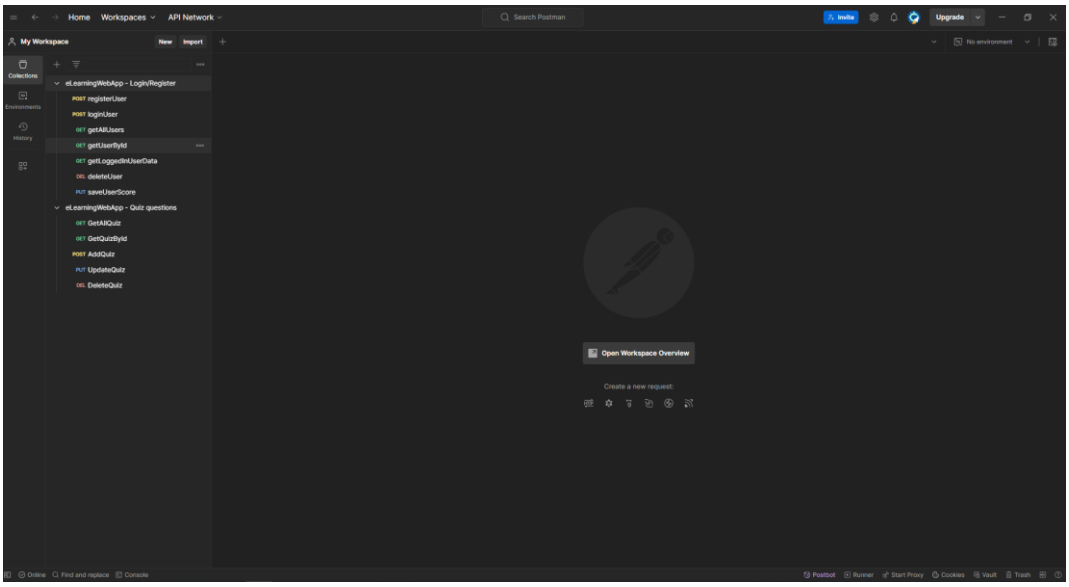

Figure 2.4: Postman

Utilizing the above mentioned ready-made software, the development of the web-based educational application with an integrated assistant was carried out on Microsoft Windows 10 Professional operating system. In the following, a detailed description of the way the web-based application works regarding the E-learing site part will be given, emphasizing both the code and the user interface using screenshots. Reference will be made to the way the digital assistant (Chatbot) is integrated in the application, as well as the use of the artificial intelligence (AI) and closing this chapter there will be a diagram of the database in order to display the tables created for the needs of the educational web application.

2.1. E-learning site

The E-learning site consists of a quiz of questions given to the user in order to offer an innovative learning based on modern standards. The theme of the questions is based on the subject of computer science. The application enables the user to see his/her personal progress (score) in points by completing the quiz and also his/her ranking in relation to other users. When interacting with the quiz questions, the user will have a digital assistant at his/her disposal, which uses artificial intelligence, will be able to provide personalized answers to his/her questions. The operation of the online educational application created for the needs of this thesis will be detailed below.

To start the application, the project needs to be started from the Back-End part of the project. Then the Front-End part requires the execution of the npm run dev command, which starts the project this time from the Front-End part. When the project is started from both parts, the main page of the application is displayed.

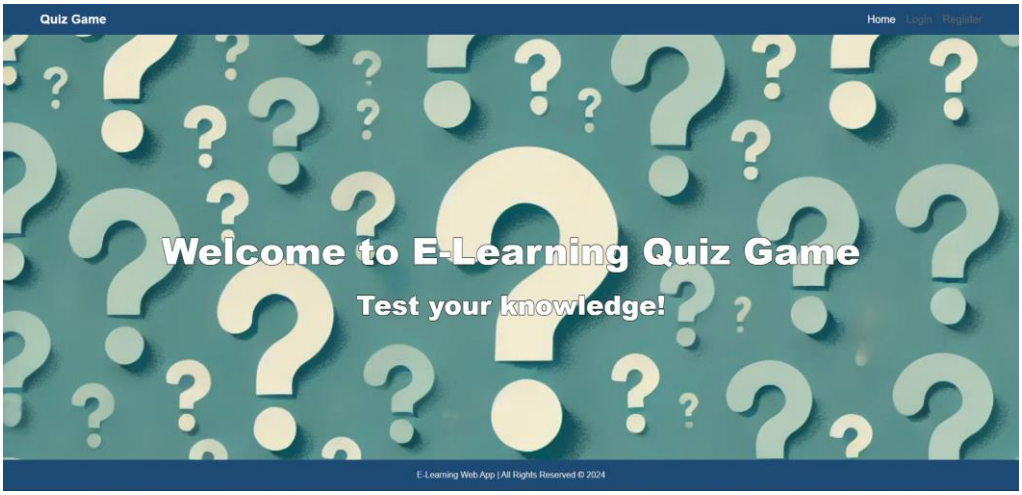

Figure 2.5: E-Learning Web App home page

Observing the main page of the E-Learning Web App on page 17 the user is given the actions through the navbar, the direction to the main page (Home button), the direction to identify the user and automatically login to the application (Login button), but also to enter the data of a new user in the application (Register button). If the user has no data in the application and wishes to be entered in the system, he/she is directed by the Register button to the page where he/she fills in his/her data in order to save the data and be entered as a new user in the database with the rights of a simple user.

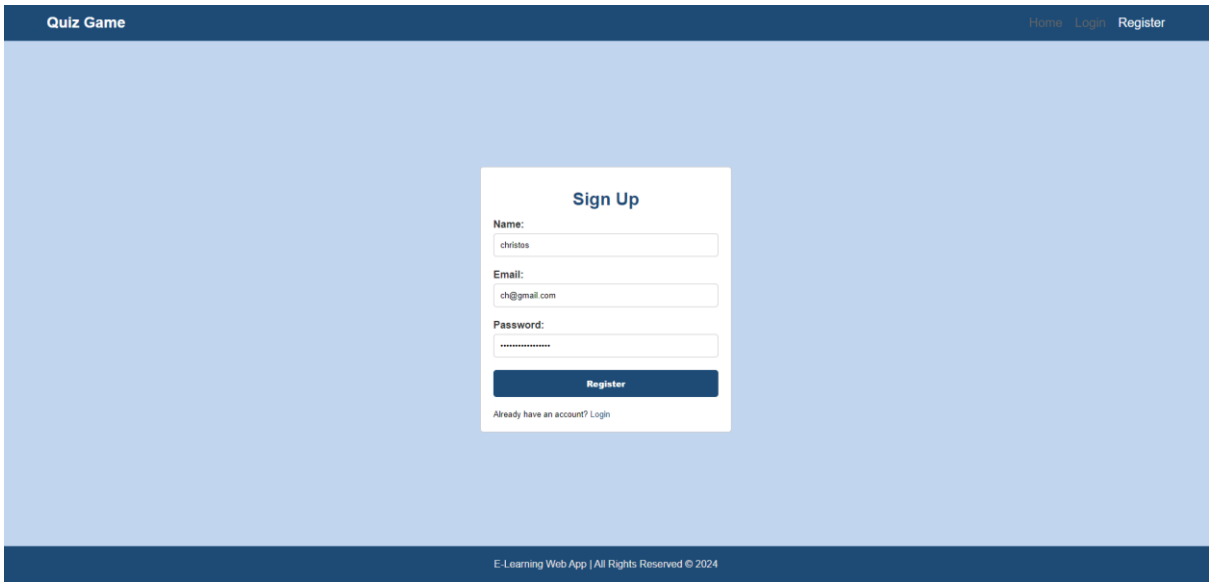

Figure 2.6: New user data entry page

After the user fills in the Name, Email, Password fields and clicks on the Register button, his/her personal details are stored in the database under the name elearningwebapp.

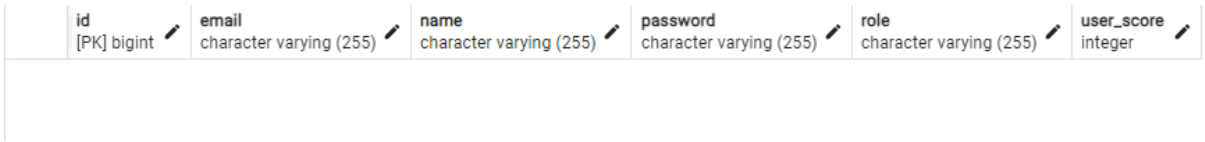

Figure 2.7: Users table (without a user entry in the system)

	id [PK] bigint	email character varying (255)	name character varying (255)	password character varying (255)	role character varying (255)	user_score integer
1	1	ch@gmail.com	christos	$2a$10$Ag6UF3emosOmC448OK/rJO2xp2SFoJk7KLMH8WqDNAVb2TRY13P...	USER	0

Figure 2.8: Users table (after user registration in the system)

By looking at the users table in Figure 2.8, the new user has been successfully introduced to the system with all his data stored and with an encrypted password, but at the same time having the role of a simple user and score = 0 as he has not yet started the quiz. The new user, after successfully registering in the system in order to log in to the application, is directed by the Login button to the user identification page, where he/she enters the Email and Password.

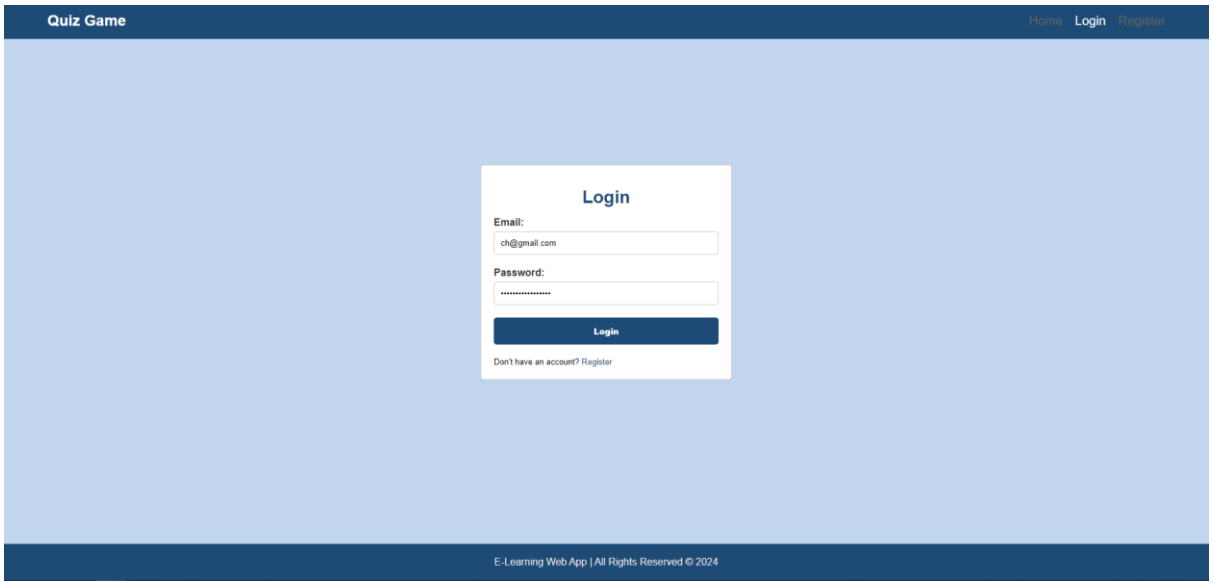

Figure 2.9: User authentication

In case the system authenticate the user successfully, it logins him/her to the application by adding new functions in the navbar. The profile button for viewing his/her personal data, but also for updating them immediately if he/she wishes. The game button that refers to the start of the quiz and finally the two buttons, one of which is the Scoreboard for displaying the final scores of all users and the Logout button with which the user is logged out.

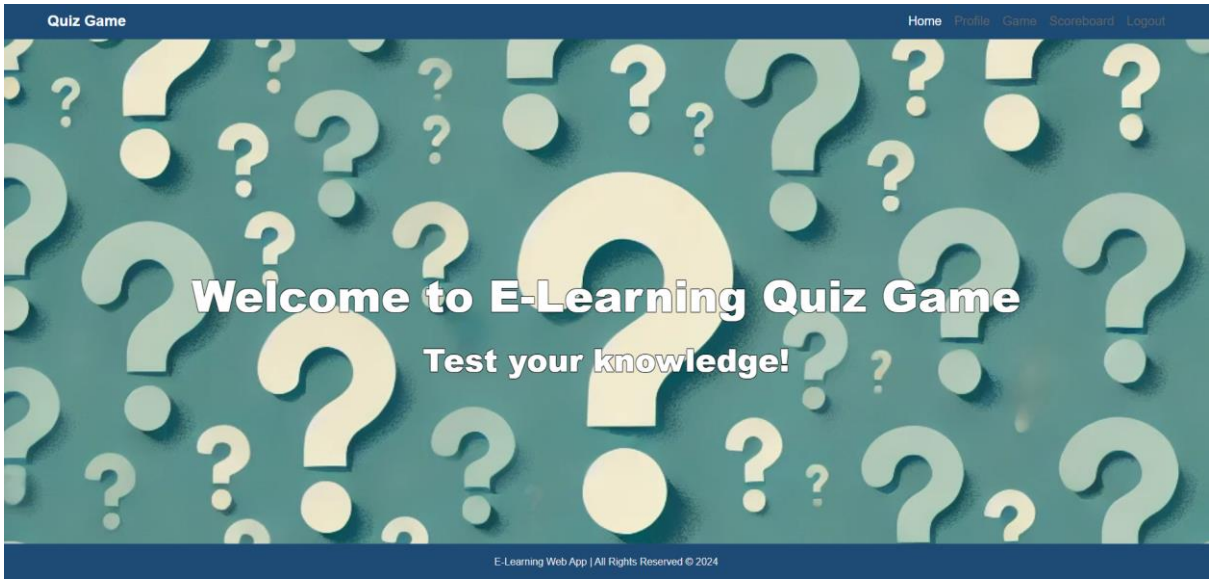

Figure 2.10: Application home page (Successful user login)

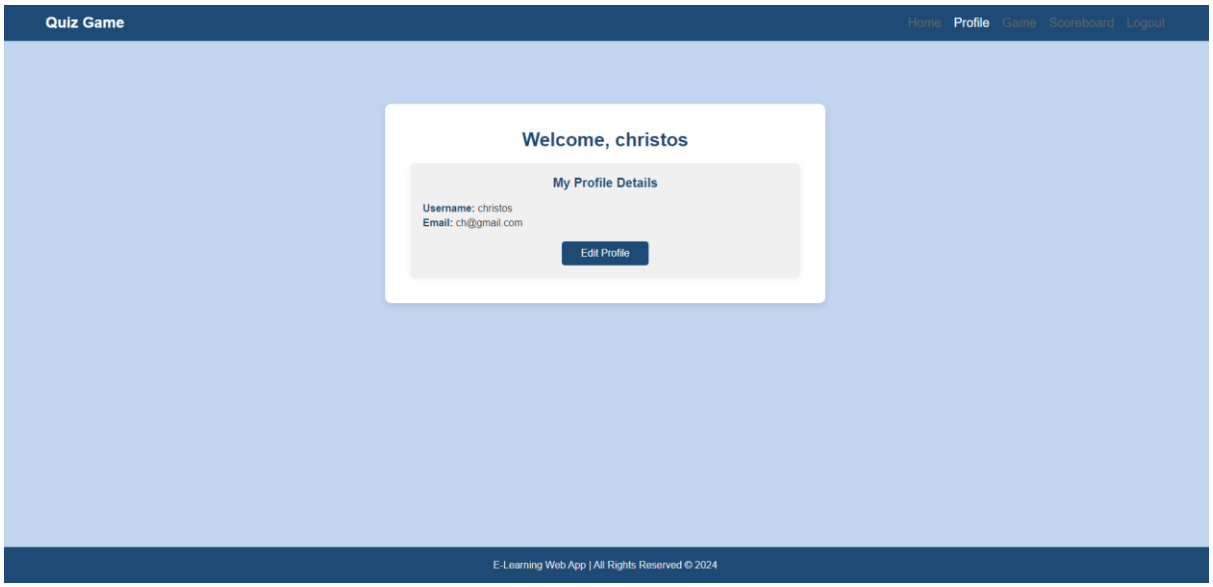

Figure 2.11: View the profile of a logged-in user

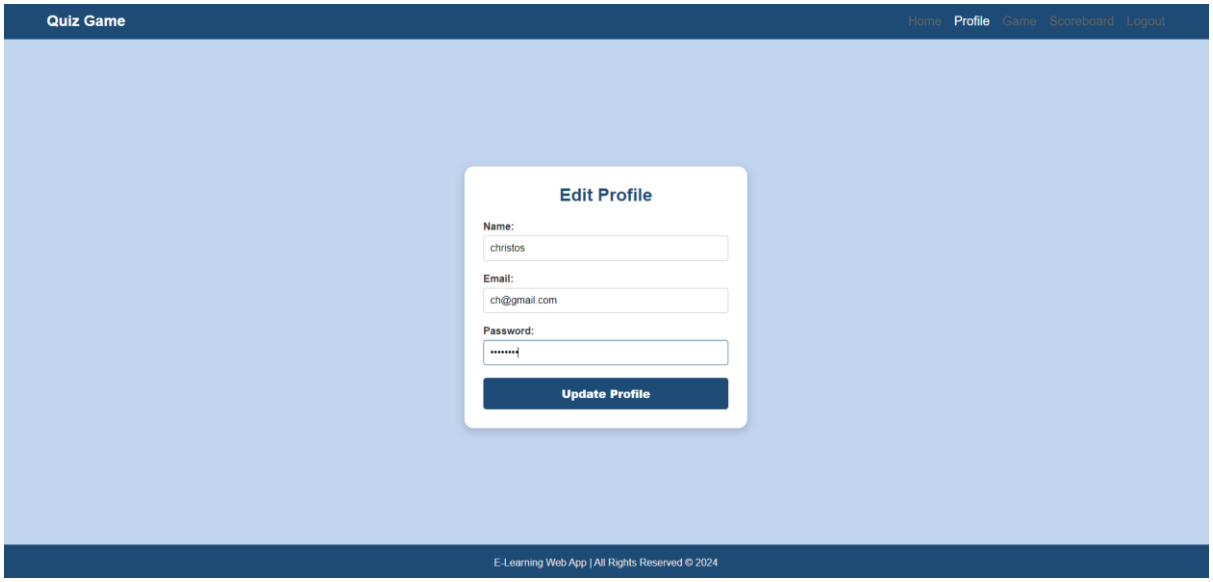

Figure 2.12: Updating personal user information

The questions contained in the quiz are stored in the quiz_questions table of the elearningwebapp database. The quiz consists of the following questions and answers:

- What does CPU stand for?
  Answer -> Central Processing Unit
- Which language is primarily used for Android app development?
  Answer -> Kotlin
- What does RAM stand for?
  Answer -> Random Access Memory
- Which of the following is a programming language?
  Answer -> JavaScript
- What is the time complexity of binary search?
  Answer -> O(log n)
- Which of the following is an operating system?
  Answer -> Windows
- What does HTTP stand for?
  Answer -> HyperText Transfer Protocol
- Which company developed the C programming language?
  Answer -> Bell Labs
- What is the primary purpose of DNS?
  Answer -> Translate domain names to IP addresses
- Which of the following is a NoSQL database?
  Answer -> MongoDB

Figure 2.13 shows the quiz_questions table containing the quiz questions with the answers in the elearningwebapp database.

	id [PK] bigint	answer_option_1 character varying (255)	answer_option_2 character varying (255)	answer_option_3 character varying (255)	answer_option_4 character varying (255)	correct_answer integer	question_text character varying (255)
1	1	Central Processing Unit	Computer Personal Unit	Central Process Unit	Computer Power Unit	1	What does CPU stand for?
2	2	Python	Kotlin	Swift	Java	2	Which language is primarily used for Android app developmen...
3	3	Read Access Memory	Random Access Memory	Random Allocated Memory	Ready Access Memory	2	What does RAM stand for?
4	4	HTML	CSS	JavaScript	SQL	3	Which of the following is a programming language?
5	5	O(n)	O(log n)	O(n^2)	O(1)	2	What is the time complexity of binary search?
6	6	Windows	Google Chrome	Oracle	Intel	1	Which of the following is an operating system?
7	7	HyperText Transfer Protocol	HyperText Transfer Procedure	HyperTransfer Text Protocol	HyperText Temporary Protocol	1	What does HTTP stand for?
8	8	Apple	Microsoft	Bell Labs	IBM	3	Which company developed the C programming language?
9	9	Resolve IP addresses	Store passwords	Translate domain names to IP addresses	Encrypt data	3	What is the primary purpose of DNS?
10	10	MySQL	MongoDB	PostgreSQL	SQLite	2	Which of the following is a NoSQL database?

Figure 2.13: Quiz_questions table

Once the user presses the Game button, the user will start the timed question quiz and at the same time will be given an additional function, that of the built-in digital assistant in order to provide support by giving personalized answers using artificial intelligence.

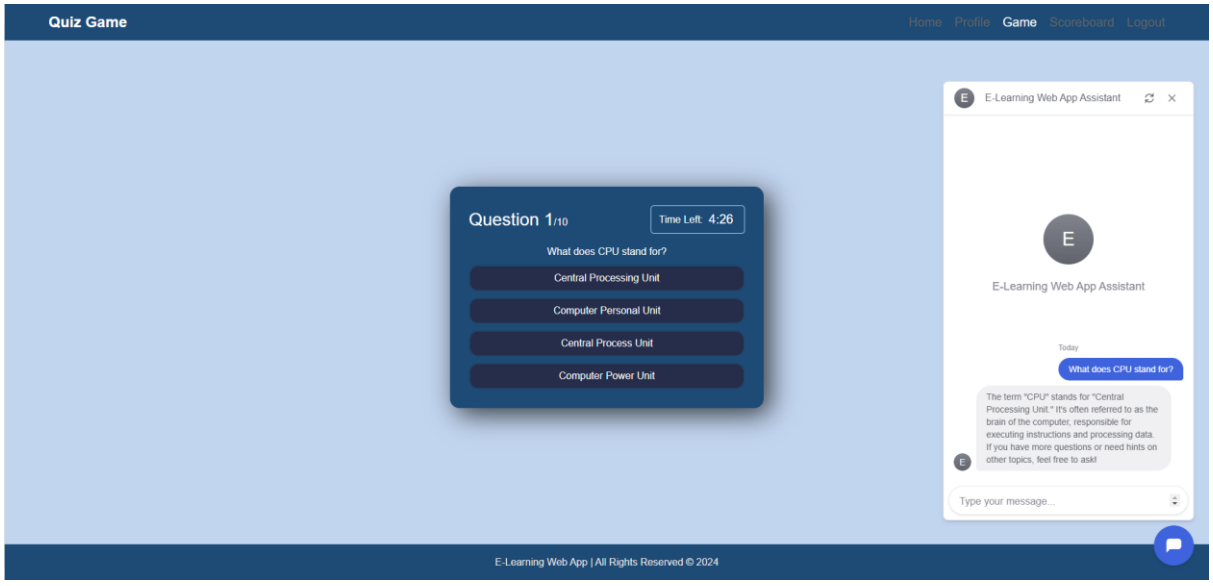

Figure 2.14: Quiz questions with built-in digital assistant

At the completion of the quiz the application will give the user the final score based on the correct answers given from the user. The user has the option of either re-displaying the quiz questions (Restart Quiz button) or going to the main page of the application (Home button) or viewing the ranking of all users (Scoreboard button).

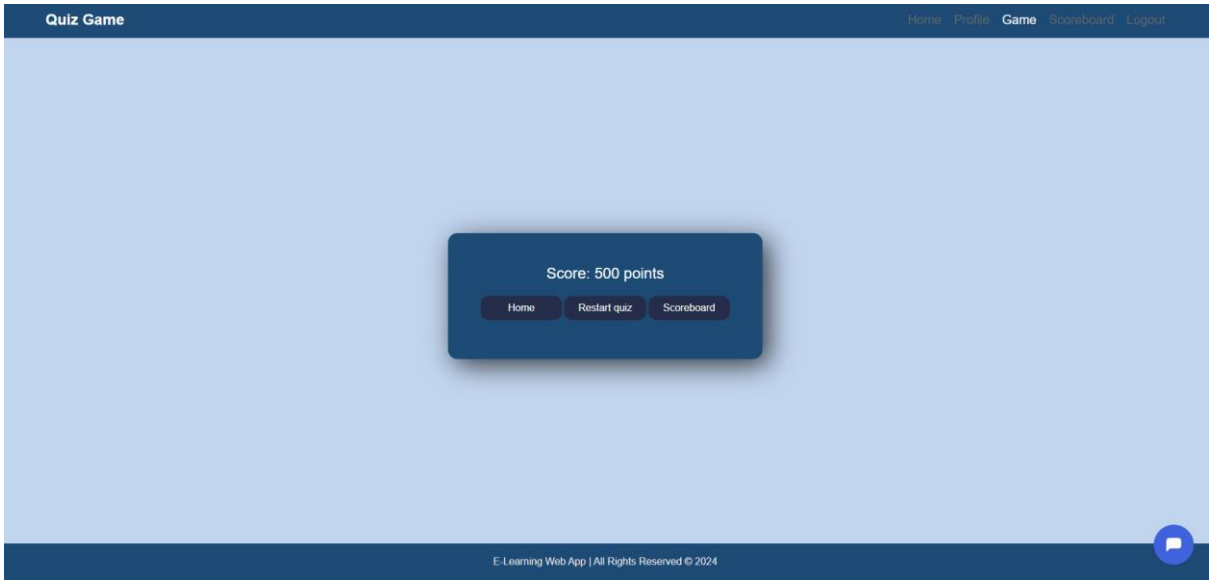

Figure 2.15: Quiz questions (final screen)

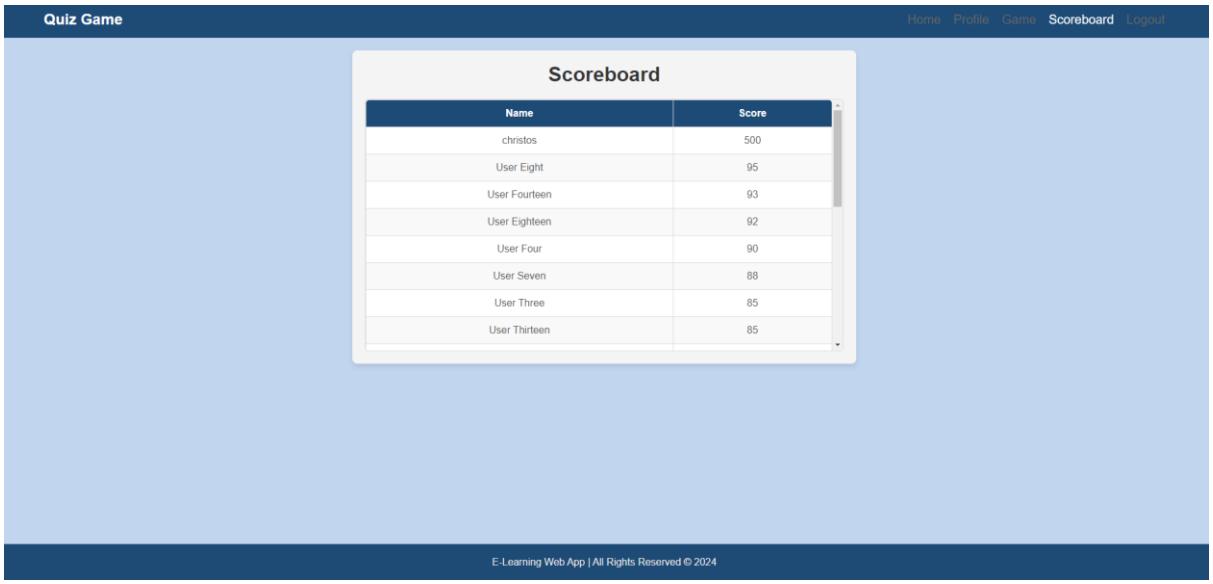

Figure 2.16: Viewing of users' ranking

In the ranking of all users based on their final score from the quiz questions, the admin was excluded and the scores are displayed from the highest upwards to the lowest downwards while displaying the Name of each user respectively as shown in figure 2.17 below.

During the game if the user accidentally presses any button on the navbar and the quiz questions are not completed then the application informs the user how he wishes to proceed.

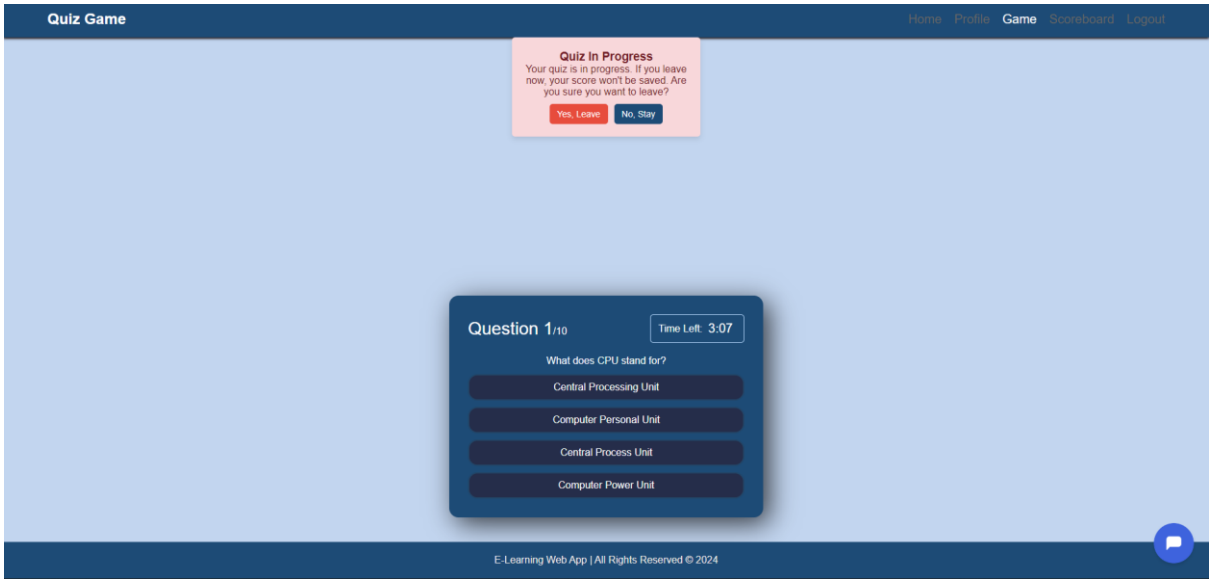

Figure 2.17: User notification for not completing the Quiz

Previously, the educational web application for the user case was demonstrated. In the following, the operation of the system for the case of the administrator (admin) will be analyzed. The Postman application was used to enter the admin in the users table of the elearningwebapp database.

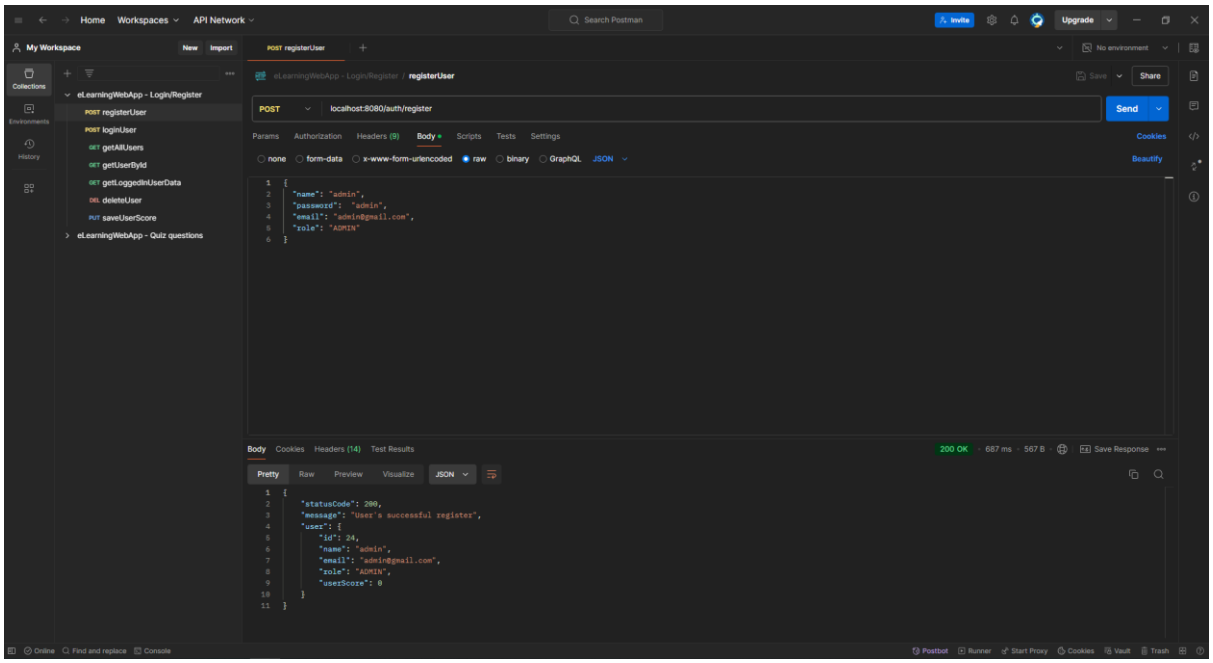

Figure 2.18: Inserting admin data into the database (Postman)

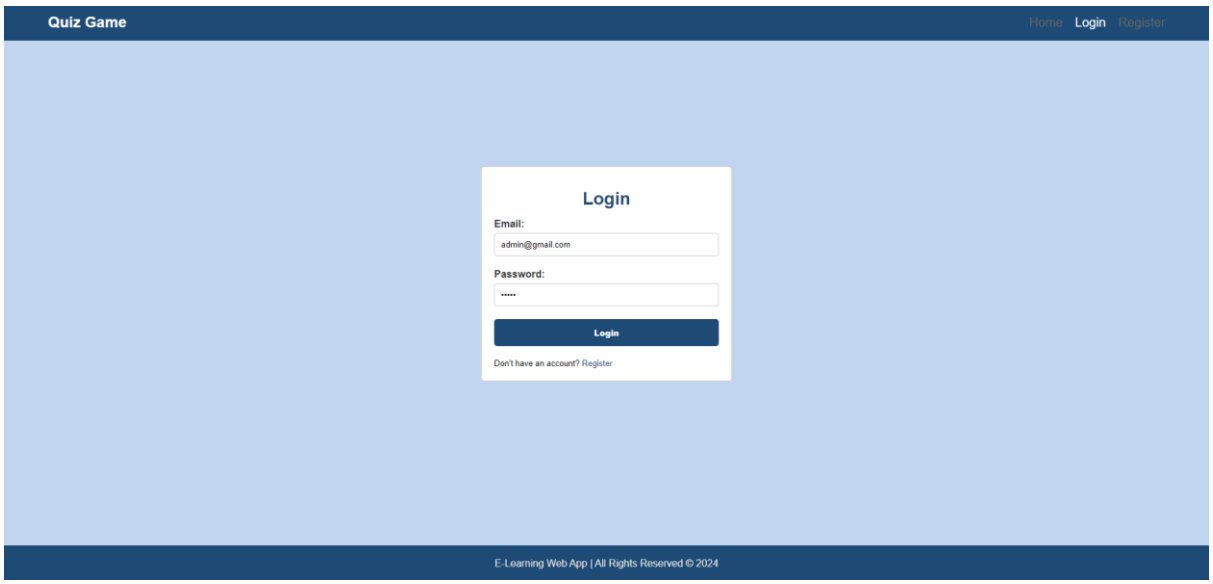

Figure 2.19: Authentication and login of admin to the application

Upon login the application the administrator is directed to the main page of the application having the admin function.

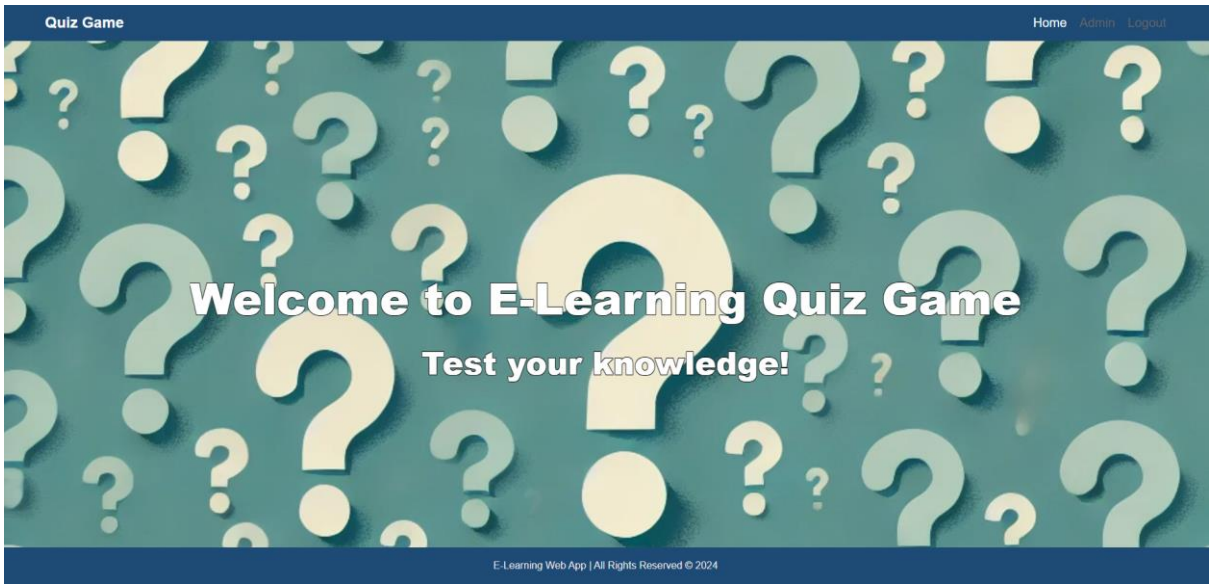

Figure 2.20: Successful admin login to the application

As for the case of the administrator function, the administrator can manage users, but also quizzes. The application allows the administrator to see all registered users and can delete a user from the application and at the same time from the database.

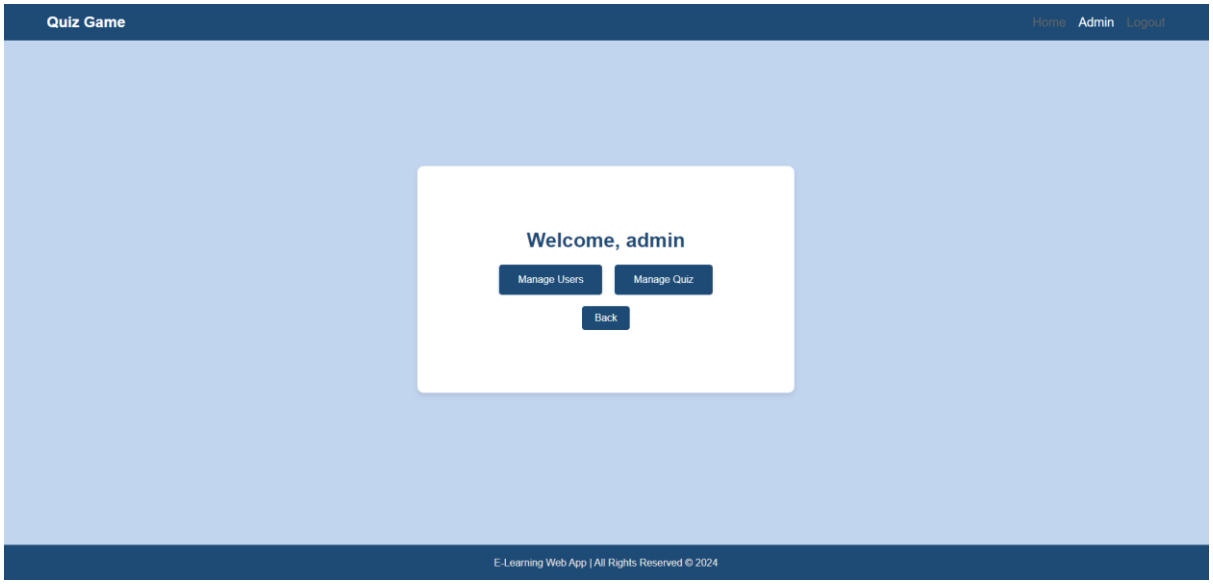

Figure 2.21: Admin functions

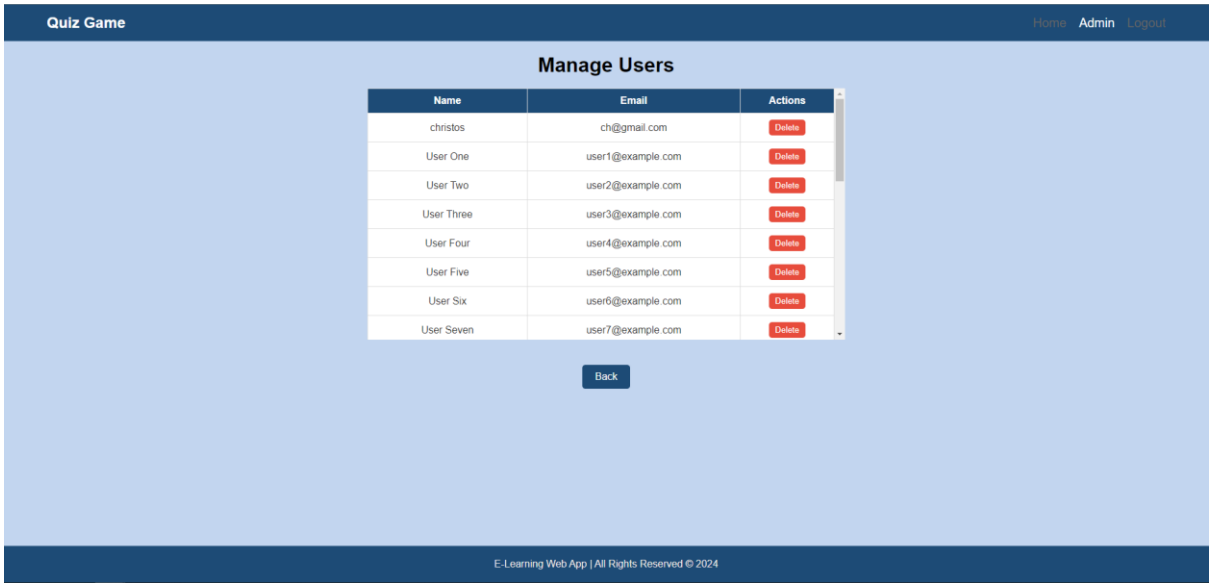

Figure 2.22: Displaying all users with delete function (Admin Role)

In case the administrator of the educational application needs to delete a user or press the delete button for a user he/she does not wish to delete, then the system notifies him/her if he/she wishes to complete this action.

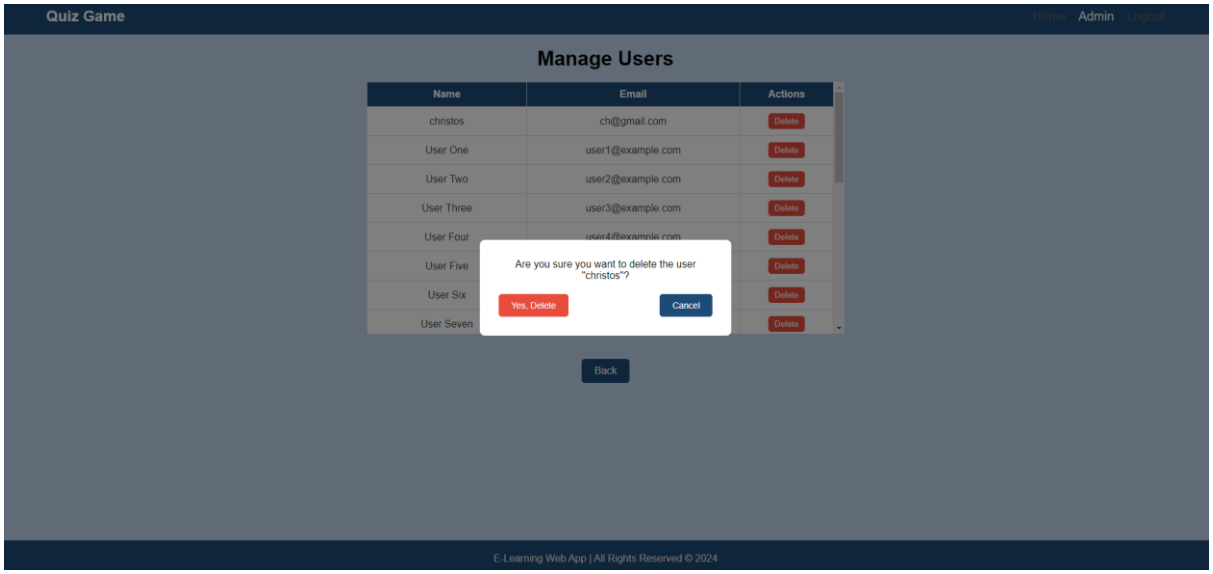

Figure 2.23: Notification for user deletion (Admin Role)

If the administrator wishes to change something regarding the quizzes, the application allows him/her to add a new quiz by giving the question and the corresponding answers, setting one as correct. In addition, he can update and delete if necessary existing questions or answers or correct answer choices stored in the elearningwebapp database.

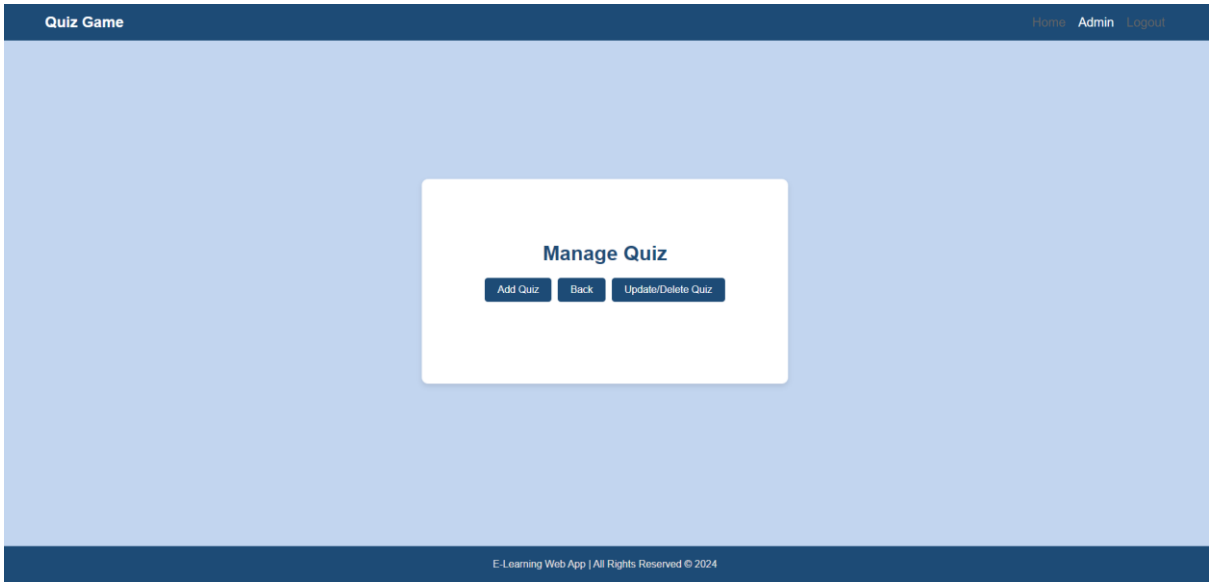

Figure 2.24: Quiz management capabilities (Admin Role)

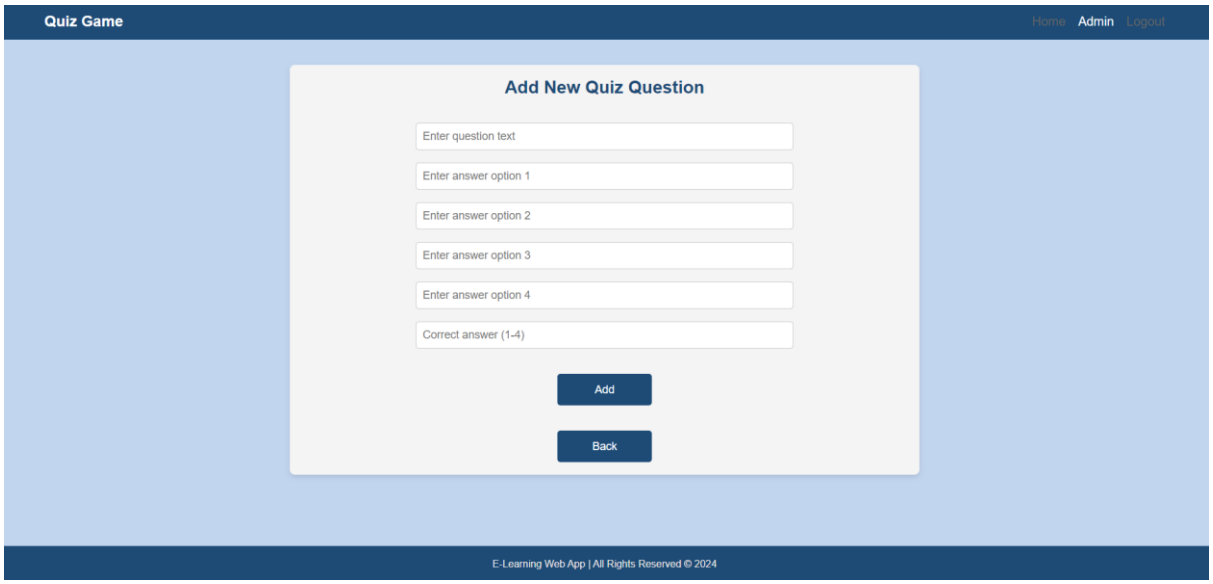

Figure 2.25: Adding Quiz (Admin Role)

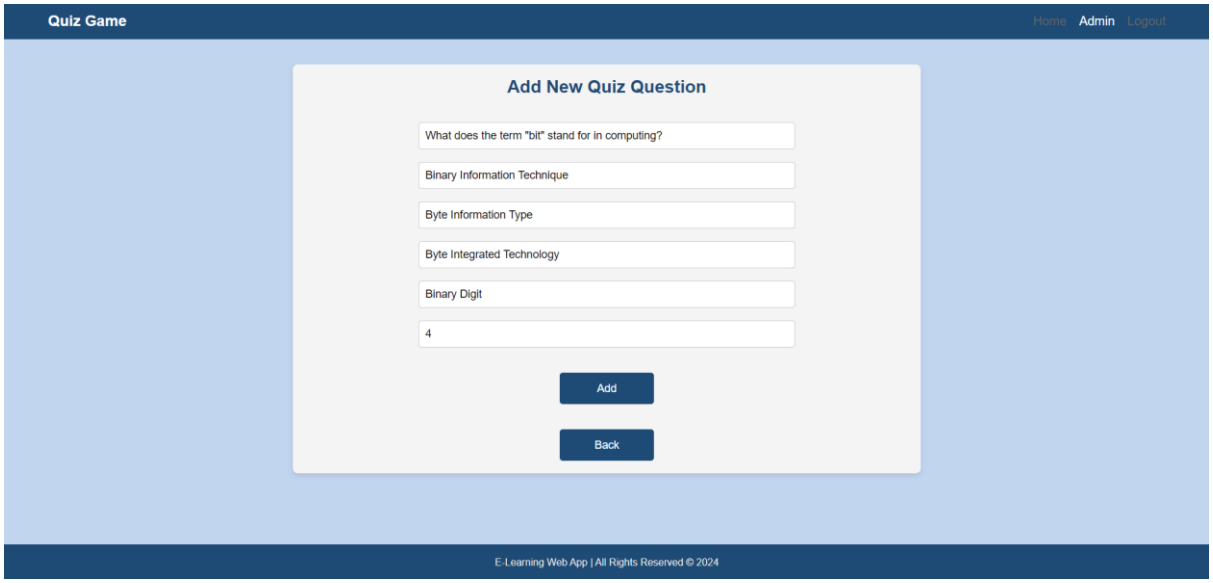

Figure 2.26: Inserting Quiz (Admin Role)

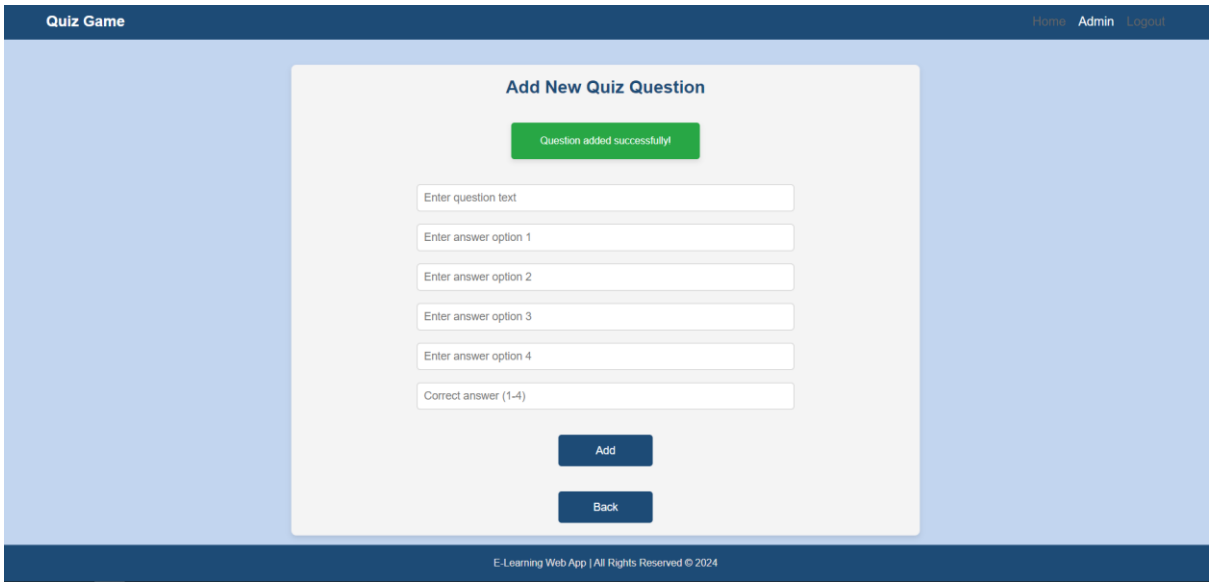

Figure 2.27: Displaying message after successful inserting of a Quiz (Admin Role)

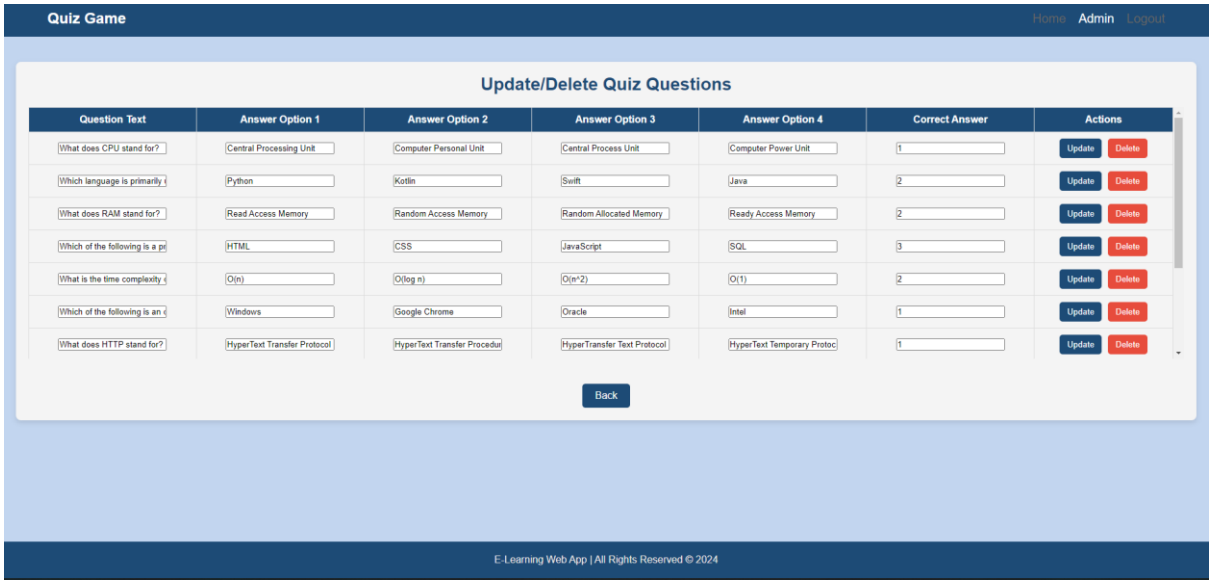

Figure 2.28: Ability to update and delete Quiz (Admin Role)

As shown above in Figure 2.28, the educational web application allows the admin to view all the questions and at the same time to update them in real time.

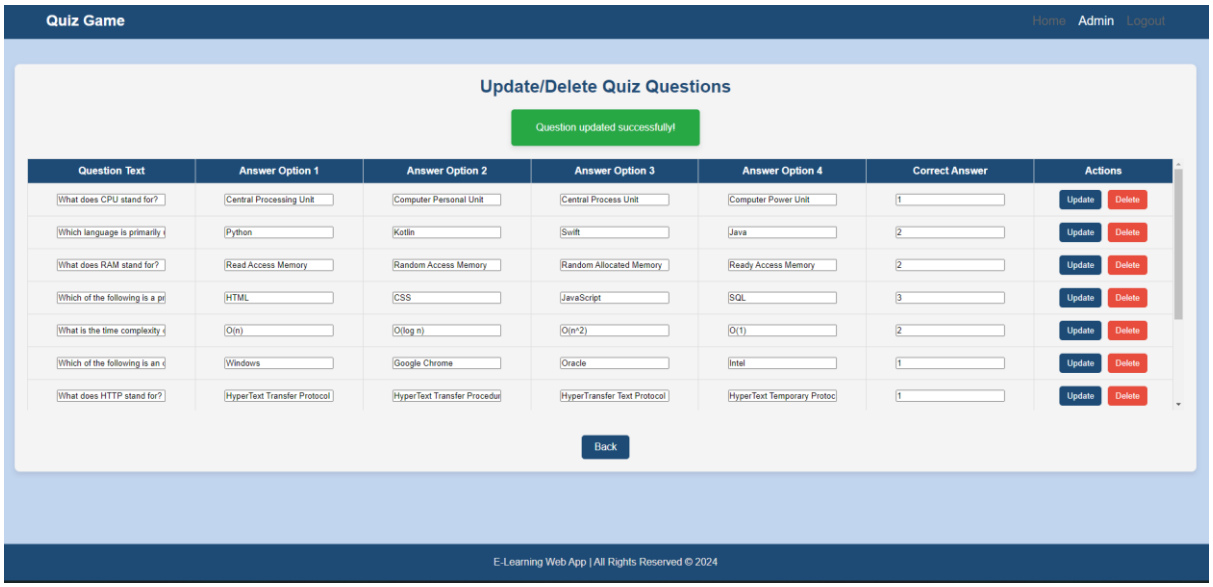

Figure 2.29: Show successful Quiz update message (Admin Role)

In the case of the possible deletion of a question with the answers, the system informs the administrator if he/she wishes to complete the specific action and if he/she wishes to do so then informs him/her with an appropriate message.

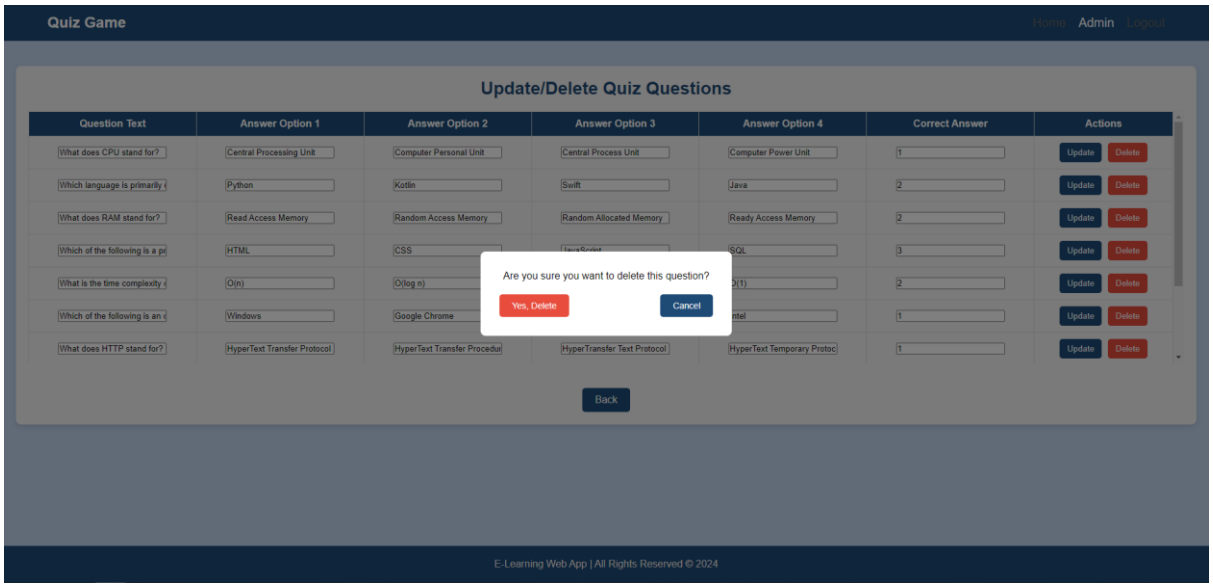

Figure 2.30: Showing confirmation message to delete Quiz (Admin Role)

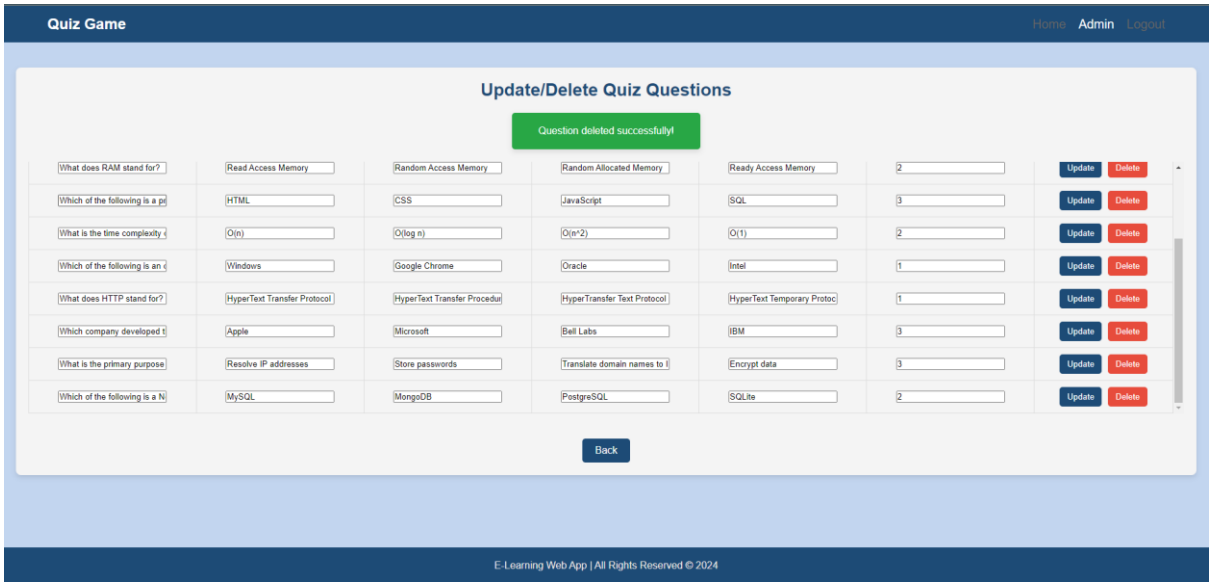

Figure 2.31: Notify admin of successful quiz deletion

When the user or administrator (admin) logs out, the system informs for confirmation of the specific action.

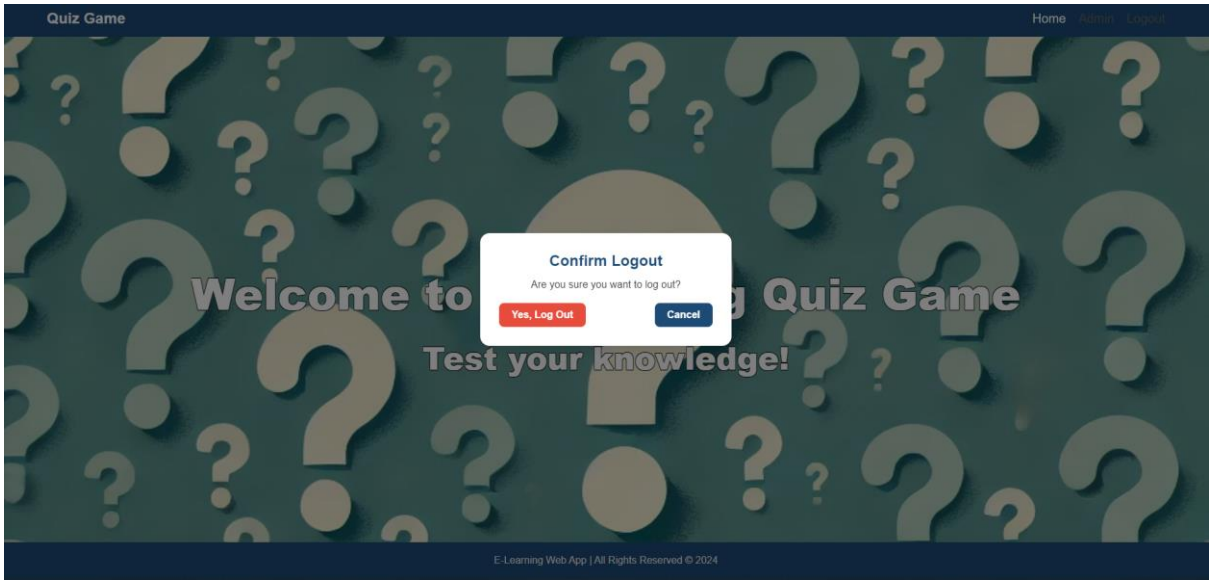

Figure 2.32: Confirmation of account logout from the application (Admin and User)

2.2. Chatbot

For the needs of the educational web application in order to integrate a digital assistant, the popular Botpress platform was used. It is an open source platform for the creation and development of intelligent chatbots. It was designed to allow for interactive interactions, enabling the development of chatbots that can physically interact with users through real-time conversations creating personalized responses. It incorporates artificial intelligence and natural language processing mechanisms to understand and respond to all user questions, regardless of their complexity. Botpress has dedicated tools to develop, test and manage conversations with ease and can be integrated with many different platforms. It also has features for analyzing chatbot performance and can be integrated into existing systems, offering flexibility and adaptability to the future needs of any application [8].

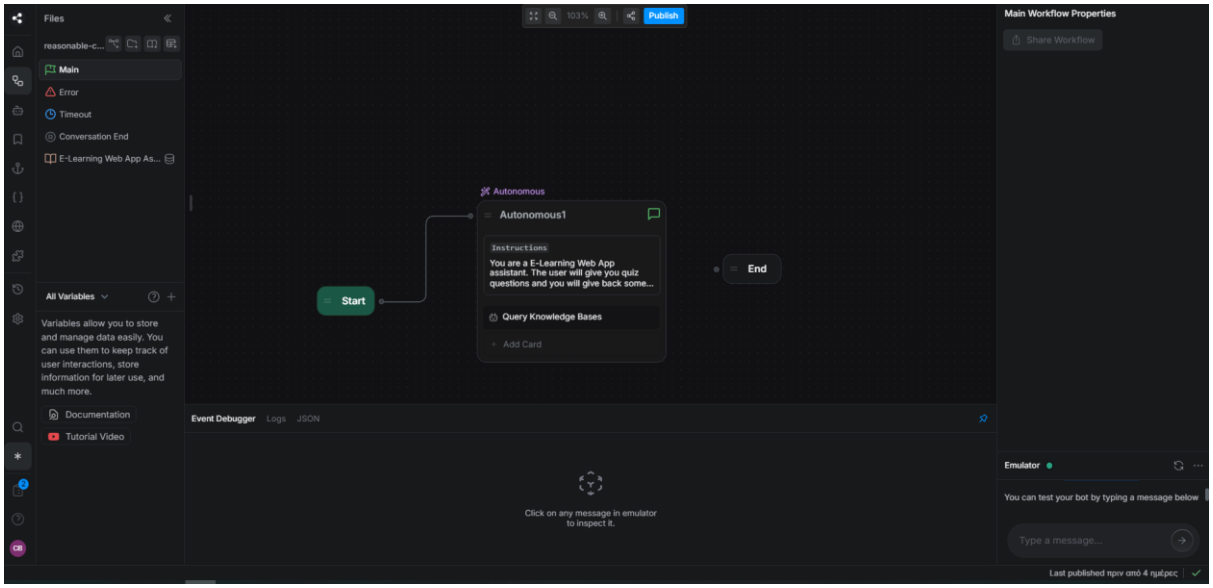

Figure 2.33: Botpress platform

In order to test the chatbot during its development, Botpress provides an emulator and a visual chat builder. It can be utilized to program actions that can help to perform specific tasks using the built-in JavaScript code. In addition, intentions, entities and slots can be defined using the NLU module. The platform was primarily designed with developers requiring an open system in mind. Botpress enables experts with different skills to collaborate and create improved chatbots in order to integrate into companies' applications and give support for any kind of need. It is easy in terms of configuration with the result that even a simple user can create their own virtual assistant in a very quick time. Finally, Botpress leverages variables to store information from one place and use it in another through a stream. The variables are intended to provide conversations personalized to the user by the digital assistant to help track information about users. They have the ability to even, connect to third party APIs and data sources [8].

2.3 Database diagram illustration

Utilizing the Pgadmin 4 tool, the elearningwebapp database was visualized as shown in the following figure 2.34 showing the two tables. The table users, where the specific table contains the user's details such as Name, Username, Email, but also the personal score. The quiz_questions table contains the quiz questions with the answers. Both the users table and quiz_questions table have their id as their primary key.

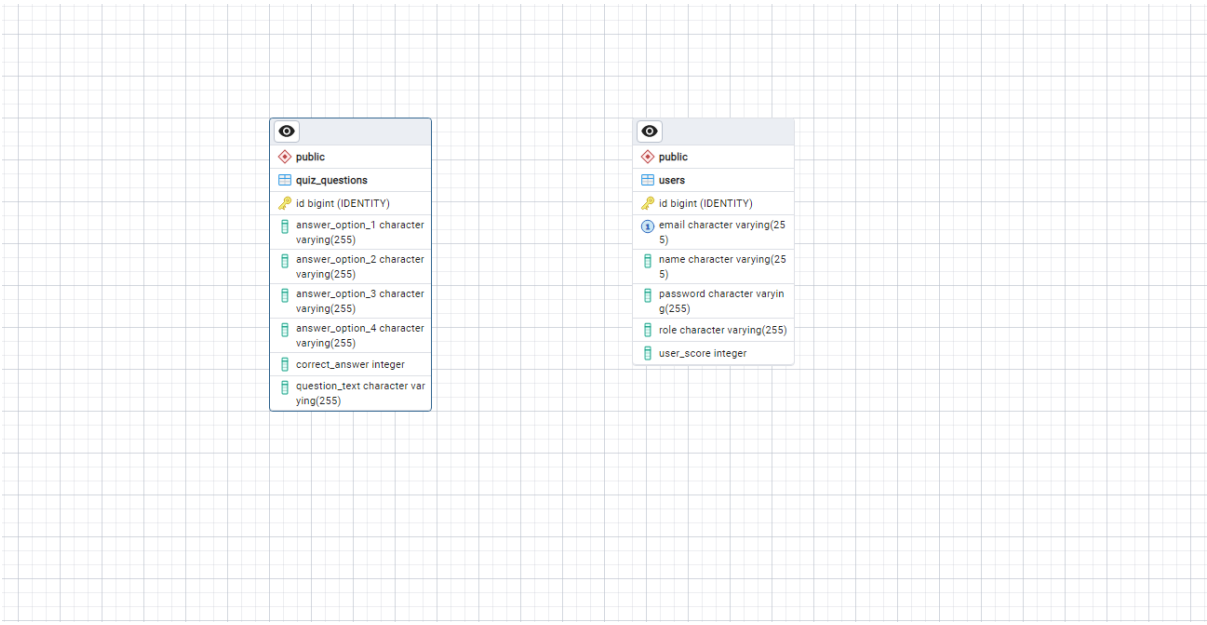

Figure 2.34: Database table relationship of the elearningwebapp database

User introduction to the educational application was carried out. The passwords of admin and user with Name = christos are encrypted, while the rest of the users are not, as for convenience and to demonstrate the application with a wider range of data to contain more user scores, they were imported directly into the database without going through the Back-End application code and performing the encryption algorithm for that data. The remaining users were imported using the following command.

```
INSERT INTO users (email, name, password, role, user_score) VALUES
('user1@example.com', 'User One', 'password1', 'USER', 50),
('user2@example.com', 'User Two', 'password2', 'USER', 75),
('user3@example.com', 'User Three', 'password3', 'USER', 85),
('user4@example.com', 'User Four', 'password4', 'USER', 90),
('user5@example.com', 'User Five', 'password5', 'USER', 65),
('user6@example.com', 'User Six', 'password6', 'USER', 78),
('user7@example.com', 'User Seven', 'password7', 'USER', 88),
('user8@example.com', 'User Eight', 'password8', 'USER', 95),
('user9@example.com', 'User Nine', 'password9', 'USER', 50),
('user10@example.com', 'User Ten', 'password10', 'USER', 80),
('user11@example.com', 'User Eleven', 'password11', 'USER', 60),
('user12@example.com', 'User Twelve', 'password12', 'USER', 72),
('user13@example.com', 'User Thirteen', 'password13', 'USER', 85),
('user14@example.com', 'User Fourteen', 'password14', 'USER', 93),
('user15@example.com', 'User Fifteen', 'password15', 'USER', 70),
('user16@example.com', 'User Sixteen', 'password16', 'USER', 80),
('user17@example.com', 'User Seventeen', 'password17', 'USER', 68),
('user18@example.com', 'User Eighteen', 'password18', 'USER', 92),
('user19@example.com', 'User Nineteen', 'password19', 'USER', 75),
('user20@example.com', 'User Twenty', 'password20', 'USER', 85);
```

Then, by running the select * from users; command in the Pgadmin4 tool, we display the table of users and notice that all the application data has been successfully stored in the elearningwebapp database.

	id [PK] bigint	email character varying (255)	name character varying (255)	password character varying (255)	role character varying (255)	user_score integer
1	2	admin@gmail.com	admin	$2a$10$/4MuzJ7QpemOjY1FtyclZeYNJ1B6z171ohk0fJTNgjsN1gH3WmS...	ADMIN	0
2	3	user1@example.com	User One	password1	USER	50
3	4	user2@example.com	User Two	password2	USER	75
4	5	user3@example.com	User Three	password3	USER	85
5	6	user4@example.com	User Four	password4	USER	90
6	7	user5@example.com	User Five	password5	USER	65
7	8	user6@example.com	User Six	password6	USER	78
8	9	user7@example.com	User Seven	password7	USER	88
9	10	user8@example.com	User Eight	password8	USER	95
10	11	user9@example.com	User Nine	password9	USER	50
11	12	user10@example.com	User Ten	password10	USER	80
12	13	user11@example.com	User Eleven	password11	USER	60
13	14	user12@example.com	User Twelve	password12	USER	72
14	15	user13@example.com	User Thirteen	password13	USER	85
15	16	user14@example.com	User Fourteen	password14	USER	93
16	17	user15@example.com	User Fifteen	password15	USER	70
17	18	user16@example.com	User Sixteen	password16	USER	80
18	19	user17@example.com	User Seventeen	password17	USER	68
19	20	user18@example.com	User Eighteen	password18	USER	92
20	21	user19@example.com	User Nineteen	password19	USER	75
21	22	user20@example.com	User Twenty	password20	USER	85
22	1	ch@gmail.com	christos	$2a$10$5qzkoQZvyglu7Y7bh9pwtOR1iqn8ExOK7fiW0wnRSBICM0AENSc...	USER	500

Figure 2.35: Users table from database elearningwebapp

The following command was used to insert the quiz questions into the quiz_questions table of the elearningwebapp database. The possibility to add or update/delete a quiz is also given directly from the application exclusively to the administrator (admin),

INSERT INTO quiz_questions (question_text, answer_option_1, answer_option_2, answer_option_3, answer_option_4, correct_answer)

VALUES

('What does CPU stand for?', 'Central Processing Unit', 'Computer Personal Unit', 'Central Process Unit', 'Computer Power Unit', 1),

('Which language is primarily used for Android app development?', 'Python', 'Kotlin', 'Swift', 'Java', 2),

('What does RAM stand for?', 'Read Access Memory', 'Random Access Memory', 'Random Allocated Memory', 'Ready Access Memory', 2),

('Which of the following is a programming language?', 'HTML', 'CSS', 'JavaScript', 'SQL', 3),

('What is the time complexity of binary search?', 'O(n)', 'O(log n)', 'O(n^2)', 'O(1)', 2),

('Which of the following is an operating system?', 'Windows', 'Google Chrome', 'Oracle', 'Intel', 1),

('What does HTTP stand for?', 'HyperText Transfer Protocol', 'HyperText Transfer Procedure', 'HyperTransfer Text Protocol', 'HyperText Temporary Protocol', 1),

('Which company developed the C programming language?', 'Apple', 'Microsoft', 'Bell Labs', 'IBM', 3),

('What is the primary purpose of DNS?', 'Resolve IP addresses', 'Store passwords', 'Translate domain names to IP addresses', 'Encrypt data', 3),

('Which of the following is a NoSQL database?', 'MySQL', 'MongoDB', 'PostgreSQL', 'SQLite', 2);

To display the entire contents of the quiz_questions table, the command select * from quiz_questions; was executed in Pgadmin4.

	id [PK] bigint	answer_option_1 character varying (255)	answer_option_2 character varying (255)	answer_option_3 character varying (255)	answer_option_4 character varying (255)	correct_answer integer	question_text character varying (255)
1	1	Central Processing Unit	Computer Personal Unit	Central Process Unit	Computer Power Unit	1	What does CPU stand for?
2	2	Python	Kotlin	Swift	Java	2	Which language is primarily used for Android app developmen...
3	3	Read Access Memory	Random Access Memory	Random Allocated Memory	Ready Access Memory	2	What does RAM stand for?
4	4	HTML	CSS	JavaScript	SQL	3	Which of the following is a programming language?
5	5	O(n)	O(log n)	O(n^2)	O(1)	2	What is the time complexity of binary search?
6	6	Windows	Google Chrome	Oracle	Intel	1	Which of the following is an operating system?
7	7	HyperText Transfer Protocol	HyperText Transfer Procedure	HyperTransfer Text Protocol	HyperText Temporary Protocol	1	What does HTTP stand for?
8	8	Apple	Microsoft	Bell Labs	IBM	3	Which company developed the C programming language?
9	9	Resolve IP addresses	Store passwords	Translate domain names to IP addresses	Encrypt data	3	What is the primary purpose of DNS?
10	10	MySQL	MongoDB	PostgreSQL	SQLite	2	Which of the following is a NoSQL database?

Figure 2.36: Quiz_questions table of the elearningwebapp database

3. Problems / Challenges and ways to overcome them

During the development of the web-based educational application, several problems and challenges arose in order for the application to be modern, easy and with interesting interactive content for the user. More specifically, the scaling of data that involved storing and managing large volumes of data in PostgreSQL can cause temporary performance issues for the application. Therefore, it is necessary to optimize the SQL queries. Establishing the communication between the Back-End and Front-End departments without some experience from some previous creation of a similar project seemed impossible. The search for documentation on both the frameworks used, the tools and the programming languages resulted in an understanding of the specific technologies which led to a better and faster development of the application. The coexistence between the Spring Boot framework (Back-End) and the JavaScript React library (Front-End) is complex, especially in asynchronous data. The need for encryption to protect the user's personal data has raised the level of difficulty in developing the system in order to counter attacks or to prevent a system vulnerability and leakage of all data. The use of JWT token for user authentication did not work properly in the first steps of the application development, as the user data was successfully registered in the elearningwebapp database, but because the communication between the Back-end and Front-end department was not properly established there were problems in viewing the user data or even the application could not verify the user by rejecting the user's entry into the educational application.

Finding a suitable digital assistant (chatbot) was a tedious process to ensure that it could provide the best possible support to the user. After much searching online and observing feature comparisons between the most modern chatbots, the Botpress platform was incorporated into the system. Initially, concerns were raised which related to its visual appearance, as well as its functionality, so that it could receive information from all over the internet. The chatbot was not giving the right answers, as it had not taken into account the sources that were necessary. Its training and programming resulted in the incorporation of artificial intelligence in order to be able to provide feedback to the user by generating personalized and accurate responses.

4. Use Cases in the functionality of the application

Chapter 4 will illustrate the Use Cases concerning the functionality of the educational application with integrated digital assistant. For successful visualization, Microsoft Visio Professional 2021 software was used.

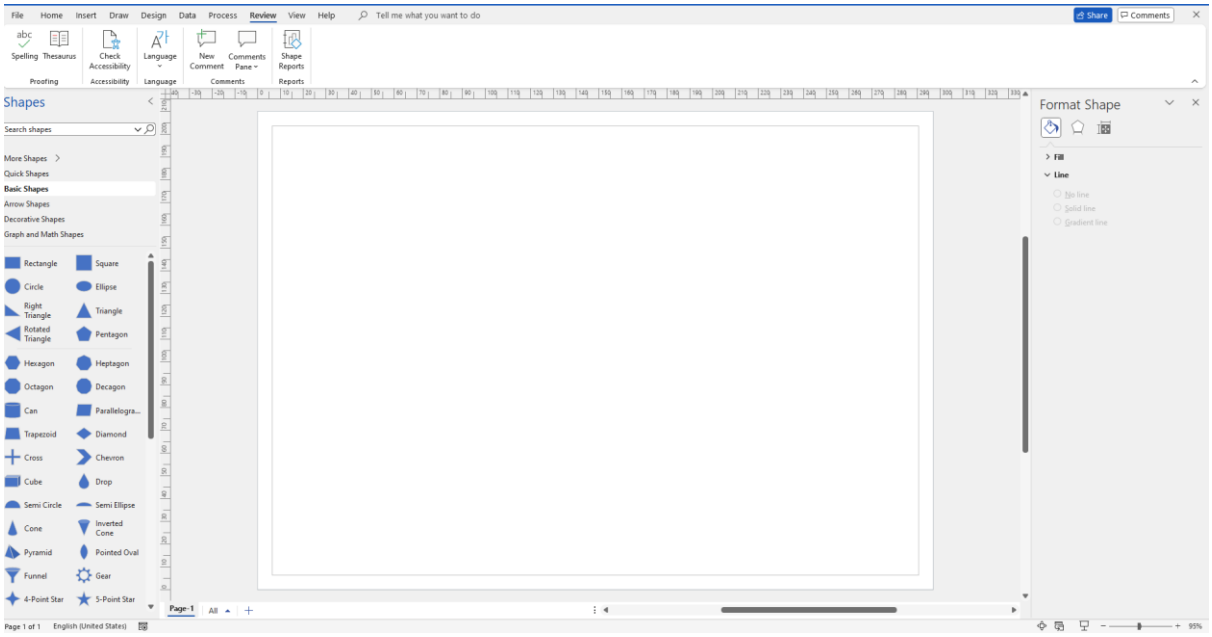

Figure 2.37: Microsoft Visio Professional 2021

Initially, the UML diagram is given, showing the process of user identification (Login) in order to be given access to the application, as well as the entry of data into the database (Register).

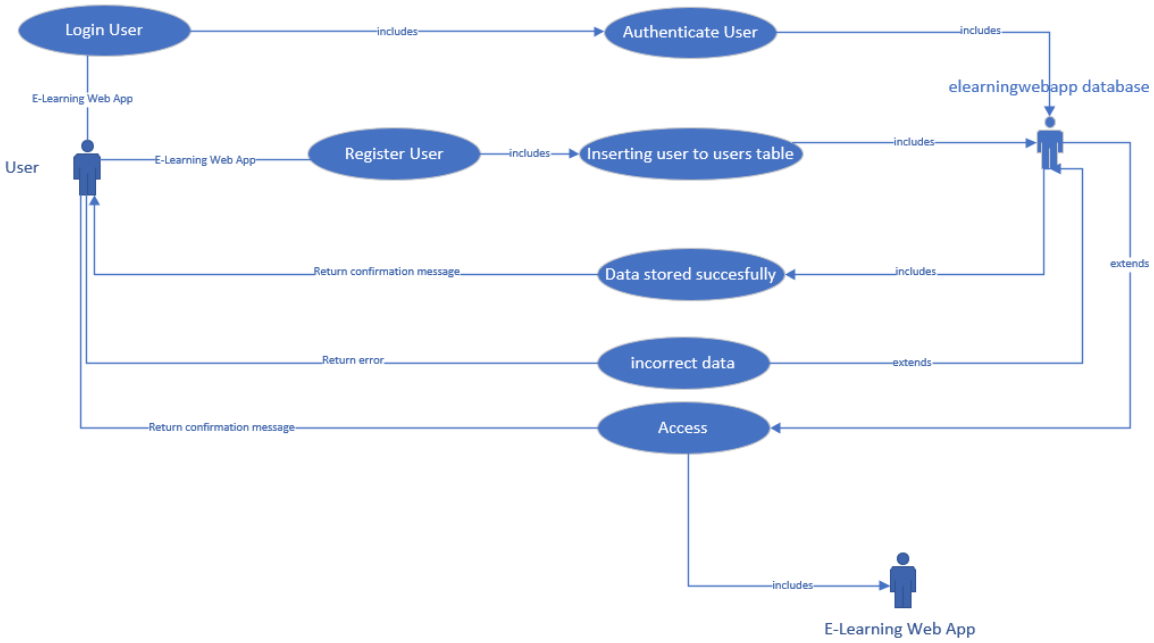

Figure 2.38: Use case of authentication and inserting user data into the database

In case of entering incorrect data during the user's authentication, the application has the possibility to inform the user by displaying an appropriate message. If the user is successfully identified, access to the application is granted. In the process of introducing a new user (register), the system shall inform the user with a message that the process has been successfully completed.

In the following, we will illustrate the use case concerning the authentication of a user with system administration rights (admin).

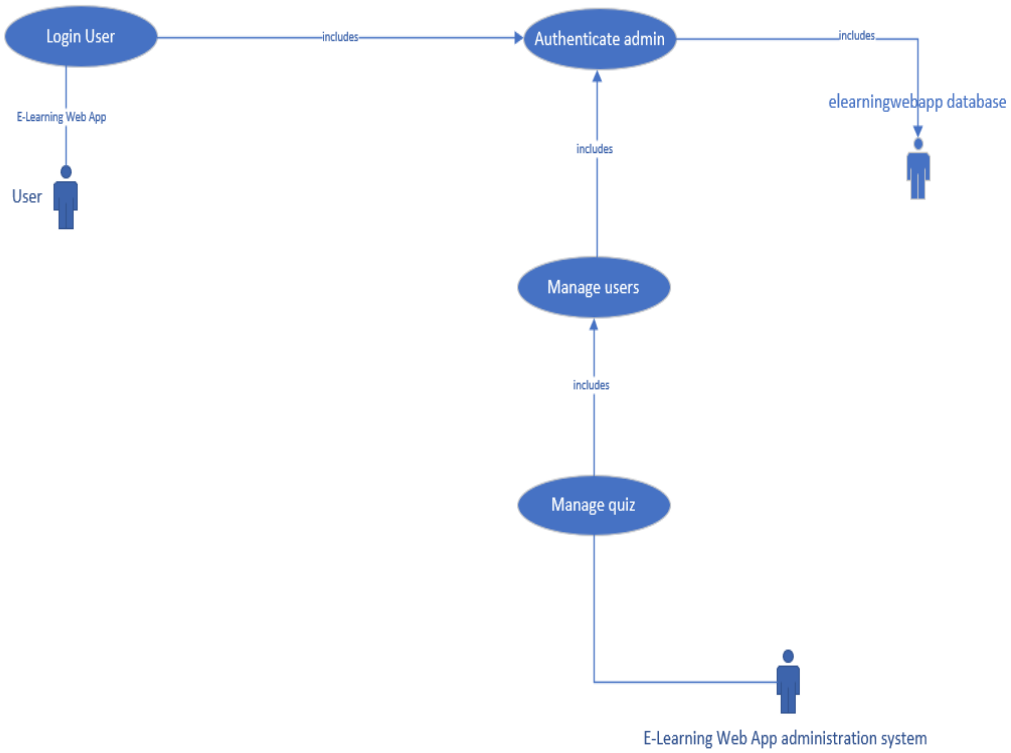

Figure 2.39: Use case for administrator authentication (admin)

Once the admin is authenticated in the base system, the application gives the admin two options. The first is to manage users by allowing him/her to delete a user if he wishes. The second possibility concerns the management of the quiz, as he/she will be able to add questions and answers either to update an existing one or to delete it, with the application displaying the corresponding appropriate message confirming the action.

Once the user is identified in the elearningwebapp database, the user is logged in to the application. The possibilities that a user has are to view his/her personal data and if he wishes to update it in real time. Finally, upon completion of the quiz, the personal score and the ranking of all users based on the score achieved in the quiz questions is displayed.

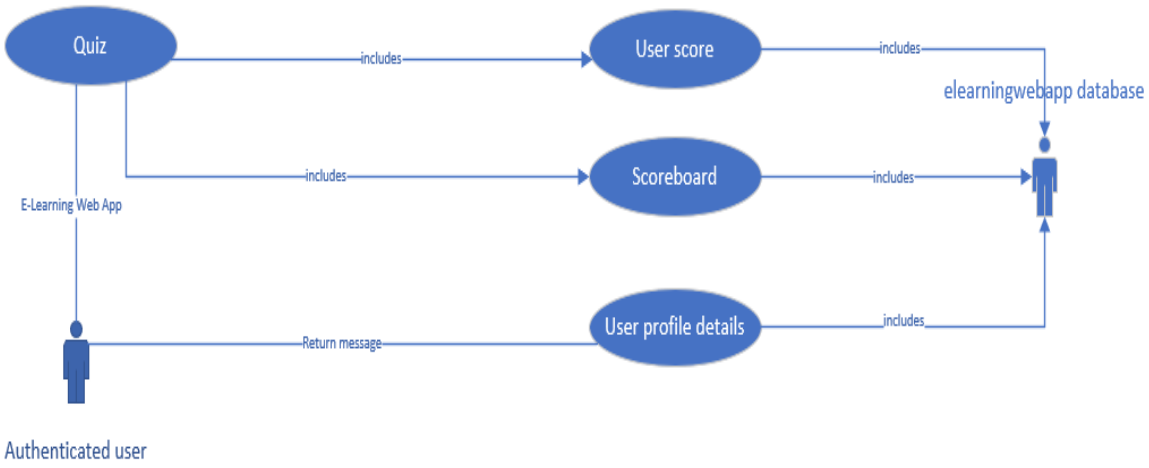

Figure 2.40: Use case of authenticated user capabilities within the application

5. Future extensions

For the needs of this thesis, an educational full stack web application with an integrated digital assistant was developed, which uses Java, JavaScript, Spring Boot, React, PostgreSQL and the open source platform Botpress. In addition to the features of the application discussed in the previous chapters, the following future extensions could be possible:

1. User login via Google API (OAuth 2.0) for authentication and secure login.

2. Multiple ways of user login, for example via Facebook or Apple account.

3. Quiz based on user interests.

4. Dynamic updating of the application's quiz questions utilizing some ready-made API possibly from a popular game.

5. Ability to support Quiz in other languages.

6. Creation of analytics table for system administrators with user participation statistics.

7. Push notifications in case there is a new version of the application

8. Offline operation of the quiz, with data synchronization when the user reconnects to the internet.

9. Better responsiveness of the application for mobile devices.

10. API for third parties, so they can create quiz questions in order to insert them to any platform they wish.

11. Use of Machine Learning to analyze user behavior.

12. More functions during the quizzes, such as showing some video or image.

13. Create a premium version of the application.

14. Ability to report application bugs.

In conclusion, all the above ideas will be able to make the educational application better by improving it in important areas, adding features that will improve the overall experience.

6. Conclusion

Completing this thesis, an extensive presentation of the educational application with integrated digital assistant E-Learning Web App along with the development methods that were applied, showing its integral operation, utilizing modern programming techniques and tools aimed at creating modern applications.

7. References

[1] J. Ofoeda, R. Boateng, J. Effah, “Application Programming Interface (API) Research: A Review of the Past to Inform the Future”, International Journal of Enterprise Information Systems, vol. 15, no. 3, pp. 76-102, July-September 2019.

[2] X. Luan, "Implementation and Analysis of Software Development in Spring Boot", M.S. thesis, Dept. of Computer Science, California State Polytechnic University, Pomona, 2021.

[3] P. Mahindrakar, U. Pujer, “Insights of JSON Web Token”, International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 6, pp. 1707-1712, March 2020.

[4] T. Lockhart, ed., "PostgreSQL User’s Guide", The PostgreSQL Development Team, 1996-2000.

[5] S. Chen, U. R. Thaduri, V. K. R. Ballamudi, "Front-End Development in React: An Overview", Engineering International, vol. 7, no. 2, pp. 117-125, 2019.

[6] K. Parameshwari, S. K, R. M, K. R, “CRUD Application Using ReactJS Hooks”, Dept. of Information Technology, R.M.K. Engineering College, Kavaraipettai, India.

[7] P. P. Kore, M. J. Lohar, M. T. Surve, S. Jadhav, "API Testing Using Postman Tool", MCA Dept., Shivaji University, Kolhapur, and Yashoda Technical Campus, Satara.

[8] B. Deshpande, M. B. Chandak, "A Survey of Designing Tools for Chatbot Application", Dept. of Computer Science and Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, India.