

**UNIVERSITY OF PULA
FACULTY OF INFORMATICS**

Sebastijan Dominis

ECON QUIZ

Rijeka, April 2025

SUMMARY

Econ Quiz is a web application that allows economics enthusiasts to have fun and learn more about their favorite topics. It uses data from World Bank Data to provide the most accurate numbers. Users can test how accurately they know GDPs, total populations, or fertility rates of various countries. There are three types of quizzes, namely multiple choice, timed, and manual input. There are seven difficulties for each topic. There are currently 24 topics, divided into three different groups of indicators, namely economic, demographic, and other. This documentation explains how this website compares to the alternatives found on the internet, describes exactly how it works, and shows screenshots from the actual application, both from a desktop and a mobile device.

1. INTRODUCTION

Trivia and quizzes in general have been popular for a while, and their popularity seems to be persistent. Popular television formats include “The Chase” and “Who Wants to Be a Millionaire?” Trivia is also shown to help with memorizing data of interest. Both pre-answer and post-answer interest seem to play a role, implying that the creation of engaging and interesting quizzes can indeed help individuals memorize better (Fastrich et al. 2017). Furthermore, it is known that age does not matter when it comes to memorizing something that an individual is curious about and interested in. Post-answer interest helps increase memory performance for individuals of all ages (McGillivray, Murayama & Castel 2015).

Trivia games are not among the most popular ones in the world. Action, arcade, casual, and racing games all attract a lot more gamers than trivia (Saifuddin et al. 2024). Nonetheless, there are many popular websites out there that offer individuals a chance to have fun and learn something about various topics. The most popular one is Sporcle. It uses its fun orange-looking brand color to excite the users, and it also allows users to create their own quizzes from different topics. However, there are not many economics-related quizzes on Sporcle. Furthermore, given that anyone can create a quiz on the website, there is no guarantee in the accuracy and up-to-dateness of the data. The types of quizzes are somewhat limited as well, and the website doesn't offer users the option to pick different difficulties when playing the same quiz. On the contrary, the quiz creator personally determines how the questions will be structured, whether there will be a time limit, and if so, how it is going to look. This means that true economics nerds cannot enjoy the trivia regarding some of their favorite topics to the fullest. Indeed, Sporcle is not the only trivia website, but most of the existing trivia websites seem to have similar issues. If anything, Sporcle is probably better than them, given that they invest a lot of effort in UI and UX. However, there are some other websites that offer educational online quiz-based games (EOQBGs), like Kahoot! and Quizziz. These games were created from the so-called “student response system” (SRS), which is a pedagogical concept from the early seventies (Wang, 2020). The practice largely revolves around asking pupils questions and getting their answers. At the most basic level, it can be thought of as posing a survey question to the class, gathering their answers, and writing them down on the chalkboard. In the 1970s, scientists conducted research on the usefulness of utilizing SRS to teach biology and chemistry to students and found that

using SRS enhanced engagement, academic achievement, and interaction. Nevertheless, there are significant disadvantages to SRSs, including waiting times, academic effectiveness, and practicality. It raises questions about how well the current EOQBGs address the shortcomings of the SRS in the past (Wang, 2020). Salsabila & Fi'aunillah (2023) wanted to explore those questions and found out that most students agree with the claim that Quizziz and Kahoot! increase their motivation to learn, as well as their comprehension of the learning material. Students also express that the apps make learning more fun and make their time spent in the classroom feel less boring.

A quick browser search for “econ quiz” shows a lot of results, some of which may be described as good or satisfying, but none of which offer quizzes on a lot of different topics and with different difficulty levels. We can conclude that there is a need for a website that allows economics nerds to enjoy econ-related trivia to the fullest. This implies ensuring that the data is accurate, providing a satisfying amount of different econ topics, allowing different difficulties, and allowing different types of quizzes, that is, different ways of solving them.

2. MOTIVATION

The previous chapter provided an overview of the existing applications that offer similar services to the one hereby presented. This chapter will go more in-depth on that topic. It will also provide a SWOT analysis and describe the application's benefits.

As previously mentioned, the most popular trivia website out there is Sporcle. It does have some economics-related quizzes. They include „Highest GDP Countries A-Z“, „10 Poorest Countries in Europe“, „Economic Terms: What is Economics?“, „Names of Economists“ and many more. Most of the quizzes have a time limit ranging from 5 to 10 minutes. Many of the economics-related topics are covered, often by different quizzes, and a lot of these quizzes are fun to solve. The most common format is the one where the terms, countries, names, etc., are listed below, and the user is expected to type the answers that match them. Some quizzes require the user to write the answers alphabetically or chronologically. Some of them end the quiz if a user inputs a wrong answer. Virtually all of them have a time limit. Most of the time a user can input a part of the answer or an abbreviation (e.g., "uk" for "The United Kingdom of Great Britain and Northern Ireland"), and the quiz automatically recognizes that as a valid answer. Some of the other quiz formats include "clickable", "map," and "multiple choice", and there are nine types of quizzes in total. Results are saved for each logged-in user and possible to view later on. Users can also choose to play a random quiz from any list of quizzes with a chosen tag or any of the predefined categories. Therefore, it is possible to simply search for a random quiz with the "econ" or "economics" tag and play it. Users can also sort and filter their search results when looking for a new quiz in a lot of ways. They can search for all quizzes, published ones, or contributed ones. Published quizzes are created by the Sporcle team, whereas contributed ones are submitted by other players. Users can include or exclude any of the predefined categories, including "movies", "science," and "language", and there are fifteen of these categories in total. Another filtering option is related to the length of the quizzes, which can range from 15 seconds to 25 minutes but does not allow custom length. Instead, there are thirty predefined lengths for the creator of a given quiz to choose from if they wish to include the time limit to begin with. They increment by fifteen seconds up to the one-minute mark, then by thirty seconds up to the three-minute mark, and then by one minute up to the twenty-five-minute mark. Quizzes can be sorted by relevance, date, rating, or play count. Sporcle allows even more detailed sorting and filtering options for its

premium users. For example, they can search for specific subcategories, by play status, by years, and so on. All of this ensures a good user experience. Other than that, Sporcle allows registered users to take lessons, earn badges, appear on leaderboards, compete with other users, and even take math tests and typing tests, in addition to other options. Considering the very rich offer of the website, one can legitimately ask what is there to offer that is not already offered? As previously mentioned, Sporcle does not offer different difficulty levels for its quizzes. They come as they are. Additionally, while the website itself offers users the ability to create nine different types of quizzes, many of the topics are covered only by one or two types at most. For example, a quick search for „maternal mortality rates“ shows two results, namely „highest maternal mortality rate countries“ and „lowest maternal mortality rate countries.“ This is probably interesting enough for the vast majority of users, given that both quizzes have less than one hundred plays. For comparison, "the countries of the world" quiz created by a user named Matt has around 38.6 million plays. However, economics nerds may want something more detailed and structured than that. Econ Quiz aims to satisfy the needs of that small demographic.

Another popular trivia website is Fun Trivia. It boasts having 3 million users and 2.5 million questions from over 160,000 quizzes and 14,000 topics. Some of the most popular topics include geography, history, and American football. That being said, a quick search for the term „economics“ gives around 15 results, whereas a search for „econ“ gives those same results, in addition to many that are actually not related to economics at all. These include "Sports' Greatest Choke Artist"s, "Gone in 60 Seconds," and "End of the Lin"e. Therefore, while the website does offer some economics-related quizzes, it is of lower quality than Sporcle, and its offer does not satisfy the needs of economics nerds.

Make Questions is also a relatively popular trivia website. While it doesn't have an enormous amount of quizzes, it does have some that are related to economics. There seem to be seven of them, and they include "Corporate logos", "Currencies: Guess the country," and "Economics acronyms". However, the questions seem to be very basic to a large degree. For example, "Economics acronyms" has a question about what the acronym GDP stands for. This question is way too simple for anyone who likes economics. It is the equivalent of asking a geography nerd what the biggest country in the world is. In addition to that, seven quizzes regarding the topic are obviously not a whole lot.

Council for Economic Education is a website that emphasizes how little many of the young people know about managing money and how this influences their lives. They offer one basic short quiz regarding the topic of financial literacy and another one on personal finance. They also offer additional quizzes for their customers. This is clearly just a for-profit website that is only marginally related to quizzes regarding economics and in no way satisfies the needs of economics nerds whatsoever.

Kahoot! is a for-profit website that aims to help educators create quizzes and make studying more engaging and impactful. The quizzes are focused on multiplayer mode to foster the competitive spirit. There are very few free quizzes, and it seems like one cannot play them as a single player, since they mostly exist only as a marketing tool. While the paid membership allows educators to create quizzes for their students and the students to solve them, it is a paid service, and there are no different levels of difficulty automatically generated. The quiz comes as it is. While Kahoot! is a useful website for many educators, it has its limitations, and it is not free.

Quizlet is a website that aims to engage the students in learning with study guides, flashcards, and practice tests. It primarily exists to help educators engage their students in studying and provide them with additional support to master the knowledge in given topics. Given that Econ Quiz aims to offer economics nerds a chance to show if they know GDPs of certain countries by heart, for example, Quizlet is not really a competitor to Econ Quiz. ProProfs and BBC Bitesize are two other websites that can be used for education. Similarly, they do not cover the needs of the niche market that Econ Quiz aims to cover.

The main strength of Econ Quiz, as previously mentioned, lies in the fact that it satisfies the so-far unsatisfied needs of a niche market. It allows economics nerds to test their true knowledge of different economic indicators as they apply to different countries. While many quiz platforms ask questions such as “What is GDP?” and “What is the most populous country in the world?”, not many ask “What is the GDP of Togo?” Indeed, most people do not care about the GDP of Togo, the population of Kiribati, or the diabetes prevalence in Japan. However, some people do, and some of them would like to test their knowledge of these things in the form of a quiz. Econ quiz allows them to do so. It offers dozens of topics that are related to economics, all of which come in seven difficulties and can be solved in three different formats. Users who are casually

interested in economics can solve topics such as “Nominal GDP p/c,” “Total population,” and “Life expectancy” on easier difficulties, while the greatest nerds of them all can solve “GDP PPP” or “Arable land (% of land area” on higher difficulties. None of the competitors offer anything like this, and the only website that offers something relatively interesting on these topics is Sporcle, but even their offer pales in comparison to the rich offer of Econ Quiz. The quizzes on Econ Quiz are also generated by an algorithm, which ensures that no two quizzes are the same, thus increasing the users’ engagement. Econ Quiz also offers a study section, and it provides a leaderboard, both of which positively contribute to the user experience. Users can sign up and log in via their Google accounts, in addition to the regular way of doing so, which increases flexibility and user satisfaction. Lastly, the data is based on the numbers provided by the World Bank Data, which ensures high quality, and the application is structured in a way that makes updating it every year a very trivial task.

The main weakness of Econ Quiz is that it currently does not support multiplayer and advanced analysis, or analysis of almost any kind. While users can see their past results, they are not shown which questions they did or did not answer correctly and what the correct answers were. Econ Quiz also doesn’t allow anyone except for the admin users to create new quiz topics or edit or delete the existing ones. Furthermore, the website can’t really be used in a traditional education setting, as it doesn’t cover that sphere of knowledge. It is assumed that the users are already familiar with the basic terms of economics and the most commonly used abbreviations. The website does not offer any instructions on what terms like “GDP” and “PPP” mean, as it is assumed that the user already knows that. If they don’t, there are many other websites that explain the terms, but that doesn’t really matter, as the user who is not familiar with such basic terms will most likely find the website to be unbelievably boring.

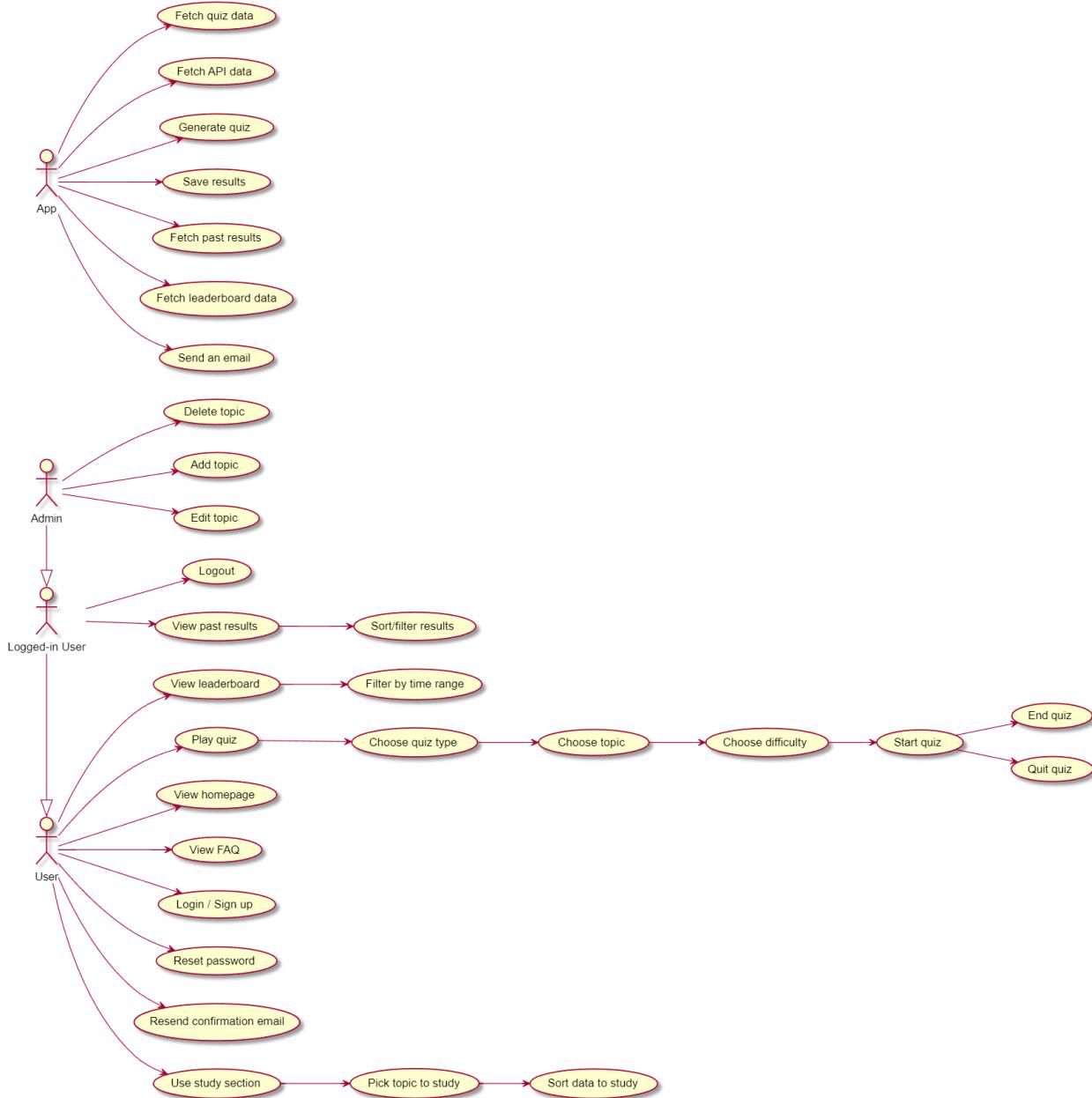
The main opportunities of Econ Quiz revolve around expanding its offer to include multiplayer and advanced analysis. While competing with oneself can be interesting, competing with other players can make the experience feel more competitive and exciting. Advanced analysis would allow players to understand where they are most likely to make mistakes and learn faster if they feel like doing so.

The main threat of Econ Quiz is that a website like Investopedia decides to create a similar offer as a part of its website. Such an offer could blend seamlessly into the rest of what Investopedia

already has to offer, and given its popularity, completely crush Econ Quiz. Another threat is that the World Bank Data simply stops updating its numbers or ceases to exist. This is highly unlikely, but given that Econ Quiz relies on their API, it would mean that the whole website needs to be restructured, or at least, a new API has to be used. Furthermore, while the World Bank Data allows a lot of API calls to be made, there is a limit, and that limit may potentially be reduced in the future. This would require restructuring the website so that the data is stored in the Econ Quiz's database. This would create even more reading from Firestore, which would quickly trigger the monthly limit that is already very likely to get hit if the number of users exceeds a few dozen. Thus, backend limitations of the current model are also a threat, and a different approach to storing data can be considered to deal with it.

3. FUNCTIONALITIES

This chapter goes more in-depth on the actual functionality of the app. It describes how it works for different users and provides a basic overview of what the app looks like. Firstly, the use case diagram will be used to serve as a broad description of the entire app.



(Created by author)

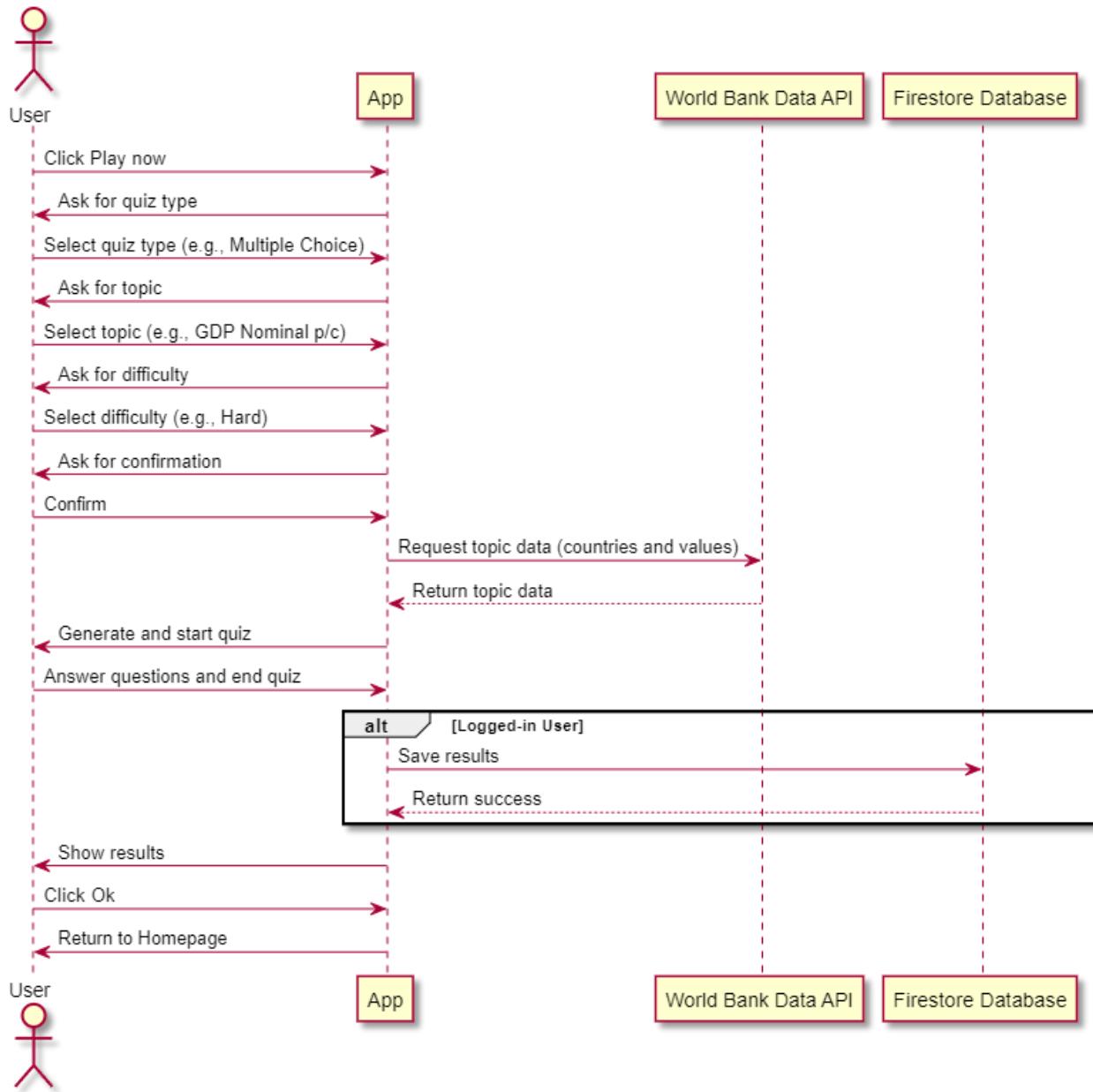
Broadly speaking, there are three different kinds of users. Those that are not logged in, those that are, and those that are admins, in addition to being logged in. All users can access the login and signup forms, even though only registered users with a verified email can actually log into the app. Users that may not have received the verification email or have simply lost it can request another one to be sent to them. Users that have forgotten their password can request an email that helps them reset it. Logged in or not, all users can access the homepage and choose to play a quiz, view frequently asked questions, view the leaderboard, or study. After opting to play a quiz, users choose what kind of quiz they want to play, what topic, and what difficulty. After the quiz starts, users can either finish it or simply decide to quit. If the user decides to study, they have to choose a topic to study, and then they can also sort the data if they wish to do so. Users that choose to view the leaderboard can also choose a time period for which they want to view the top-scoring players. The results of logged-in users are saved, and they can access them whenever they want to do so. Obviously, logged-in users can also log out. In addition to all of that, admin users can add new topics or edit or delete the existing ones. The app itself helps enable the whole process by acting as an intermediary between the user and the database, as well as an external API. The app can fetch quiz data, fetch API data, generate a quiz, save results, fetch past results, fetch leaderboard data, or send a request to the database to send an email to the user.

The main process in the app is the playing of the quiz itself. User chooses to play the quiz by clicking on the “Play now!” button on the homepage. After that, the user can either change his mind or choose a quiz type. There are three quiz types in total—“Multiple Choice,” “Timed,” and “Manual Input.” Multiple-choice quizzes, as the name suggests, prompt the user to choose the correct answer out of the four available options. These quizzes have no time limit, and the user is allowed to return to previous questions if they feel inclined to do so. Timed quizzes are like multiple-choice ones, but they have a five-second time limit on each question, and there is no way of returning to any of the previous questions. Manual input quizzes prompt the user to manually type the answer and then evaluate how close the user’s answer was to the accurate one and calculate their score accordingly. Users can also go back to previous questions in this mode, and there is no time limit set. There are always 20 questions in total. After the user has chosen a quiz type, they are asked to choose a topic as well. At the time of writing this documentation, there are 24 topics overall, but this number is flexible. Topics are categorized by which

indicators they belong to the most. The three indicator types are economic, demographic, and other. The topics currently included for economic indicators are GDP PPP, GDP PPP p/c, inflation, nominal GDP p/c, unemployment, imports as % of GDP, exports as % of GDP, and nominal GDP. The topics currently included for demographic indicators are life expectancy, population growth rate, population 65+ (% of total, total population, urban population (% of total, net migration, fertility rate, and population 0-14 (% of total. The topics currently included for other indicators are Internet users (% of population, arable land (% of land area, diabetes as % of people ages 20 to 79, maternal mortality ratio (per 100k births), forest area (% of land area, poverty headcount ratio, health spending (% of GDP, and literacy rate. After the user has chosen a topic, they can either go back or choose a difficulty. There are seven difficulties in total, namely noob, easy, very easy, normal, hard, very hard, and absolute madman. Friendly UI shows some images and text while the users are hovering over difficulties and trying to decide which one to choose. Users can still just decide to go back, or they can choose a difficulty. At this point, the app requests their confirmation, and if they confirm, the process of creating a new quiz only for this specific user begins. The app sends a request for the topic data (countries and values) to the World Bank Data API. World Bank Data is arguably the most trustworthy website when it comes to economics-related data. They only display official government data they are able to gather. Their database stores data for many different indicators, and it is easy to fetch it, since the only difference lies in the key and year used to fetch it. The limit for the number of API requests is very high, so it is almost impossible to get blocked by them. As long as the internet connection is good and the code is written properly (it is), the fetching of the data should go well. After the data is fetched, it is first filtered so that the values for country groups as well as countries not widely recognized by the UN are not included. After that, the key-value pairs, whereby the keys are the country names and the values are the values for a chosen topic, are saved locally. This data is then used in a function that creates the quiz, but it is first shuffled. The function then takes the correct answer and multiplies it by random values from a range that depends on the difficulty that the user has chosen. The easier the quiz, the higher the factors of multiplication. For example, if the correct answer is 30,000, the options with wrong answers may include 4,000 and 100,000 for noob quizzes, but 30,204 and 29,653 for absolute madman quizzes. Another element that depends on the quiz difficulty is the countries that appear in the questions. For example, noob quizzes will most likely have countries like the United States,

China, and Germany appear in them, whereas absolute madman quizzes will have countries like Tuvalu and Kiribati. If two of the wrong answers are too close to each other, one of them is multiplied by a random factor in a range that also depends on difficulty until the two answers are no longer too close to each other. If it is impossible for an answer to be above 100, above 120, or below 0, the algorithm prevents that as well and looks for a value that is within a reasonable range that is at least hypothetically possible for a given topic. For example, it is hypothetically possible for a country's nominal GDP to be 1,500 American dollars or 120 trillion American dollars, but it cannot be a negative value. Similarly, the percentage of total population or total land area based on whatever factor has to be between 0 and 100. After three false options are generated and satisfy all of the mentioned conditions, they are shuffled so that the correct answer is not always A, for instance. Questions are all the same in the sense that they ask, "What is the value of the topic for the country?" whereby topic is replaced with an actual topic and country is replaced with an actual country name. In total, 21 questions are created, and 20 appear in the quiz. This is because the array starts with zero, but the counter for questions starts with 1. The score for a given quiz depends on the type of quiz. For multiple-choice and timed quizzes, the number of correct answers is simply divided by the total—20. For manual input quizzes, an algorithm evaluates how close the user was to guessing the exact value for each and every answer, and then the average of 20 values is taken as the final score. All of this is only calculated if the user actually decides to end the quiz or wait until the timed quiz ends, and the score is then displayed to the user. If a user is logged in, another kind of score is calculated as well, and it is called a "leadership score." This is later used to calculate the top scores and display the best user's usernames along with their best leadership score on the leaderboard. This score rewards solving timed and manual input quizzes, as well as higher difficulties, and it punishes playing the life expectancy topic, given that it is considerably easier than the other ones. If a user is logged in, his score, leadership score, and some other data are saved to two places in Firestore. Firstly, it is saved to the user's personal results, and secondly to the global collection called results. The global collection also stores the user's username, since it is used to create leaderboards. Storing results into the user's document enables quickly fetching that particular user's results later on, whereas storing them in a global collection enables calculating the leaderboard more quickly. Other than the scores, the app ensures that the type, topic, difficulty, time spent on solving the quiz, and the timestamp of when the quiz was taken are all saved. If this does not go well, the

user is notified that something went wrong and their results were not saved. Finally, after the user sees the results, they can simply press the OK button to get back to the homepage.

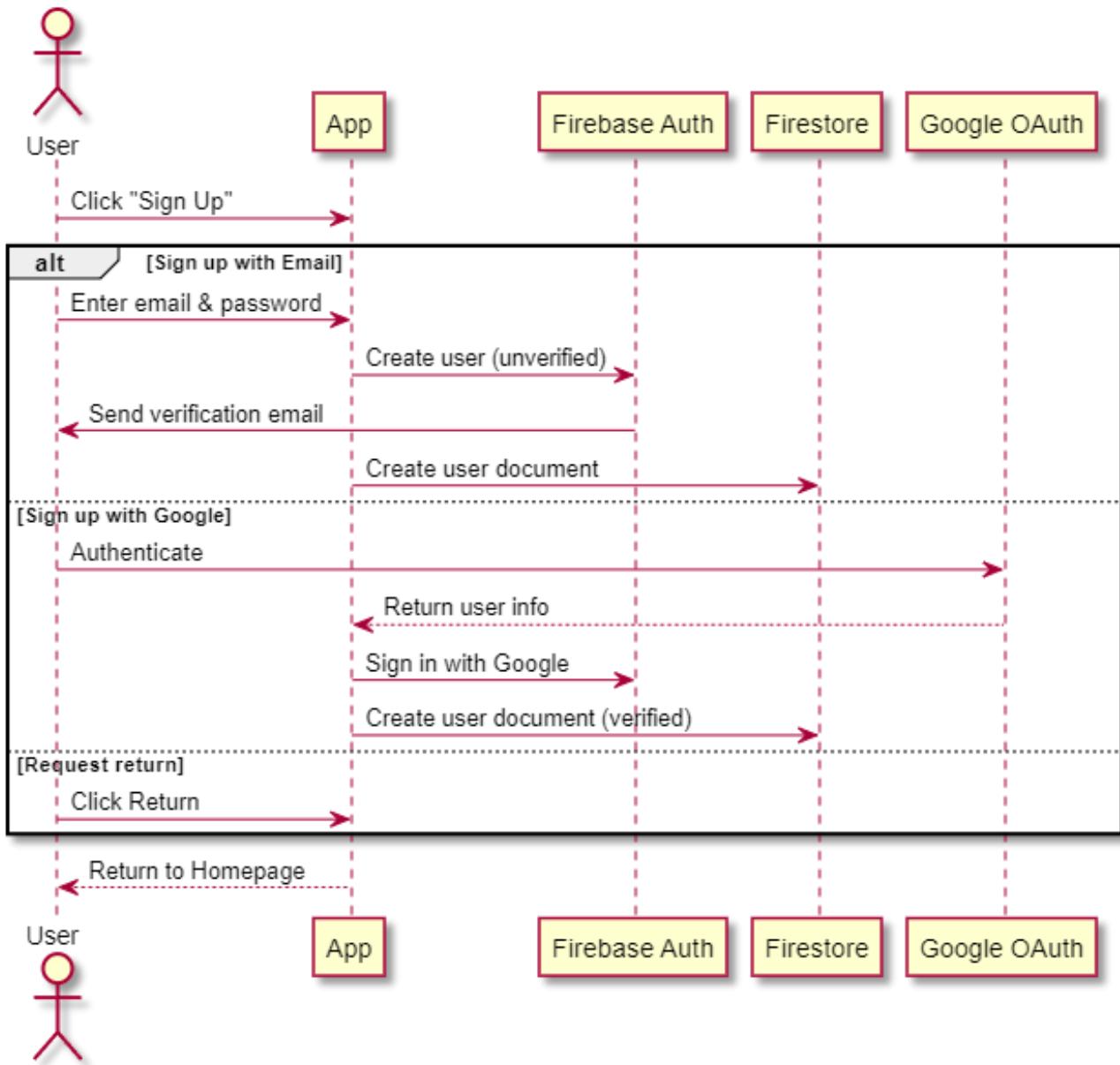


(Created by author)

Clearly, logged-in users have some extra benefits when compared to the other users. Namely, their results are saved, they can view those results later, and they can appear on the leaderboards. Now the processes of signing up and logging in will be explained. In order to start the signup

process, a user has to click on the “Sign up” button. From there, the user can choose between signing up with email and password and signing up with Google. If they choose to sign in with email and password, they need to fill in a very user-friendly form. They are requested to choose a username, enter their email, confirm their email, create a password, and confirm the password. At the beginning, at least on larger screens, a red frowny face appears to the right of each of these five inputs. Whenever a valid value is entered, the red frowny face is replaced by a green smiley face. Username can be between 5 and 25 characters long. Email has to satisfy a proper email format. The repeated email has to be the same as the first one. Password has to be between 8 and 20 characters, include at least one lowercase letter, one uppercase letter, one number, and one special character. The repeated password has to match the first one. Once all of this is satisfied, the user can sign up. The app then sends the user’s credentials to the Firebase database, where the user is first created in Firebase authentication, and then added to Firestore as well. In Firestore, a user’s document contains their username and email, an isAdmin value that is initially set to false by default, and later the past quiz results of that user. The isAdmin value determines whether the user is allowed to use the Admin options or not. After the creation of the user in Firebase authentication is done, a verification email is automatically sent to the user by Firebase authentication, and the user is logged out and redirected back to the homepage. They will only be able to log in once their email is verified. Alternatively, the user can sign up with Google, in which case the Google OAuth is used. The process of saving the user’s data to the Firestore database is the same. The part with the Firebase authentication is slightly different. If the user’s Gmail does not appear in the database, a new user is created, and the email is automatically verified, since the Google OAuth requires a valid Gmail to be used in order for the process to be successful. If the Gmail has already been used to create a different account, then these two accounts are merged, and the display name of the user, which is saved in Firebase authentication, is changed from their Google account’s name to the username they chose initially when creating their first account. This username is later displayed on the homepage to welcome the user. The user is also redirected to the homepage after the Google signup. Alternatively, a quit button

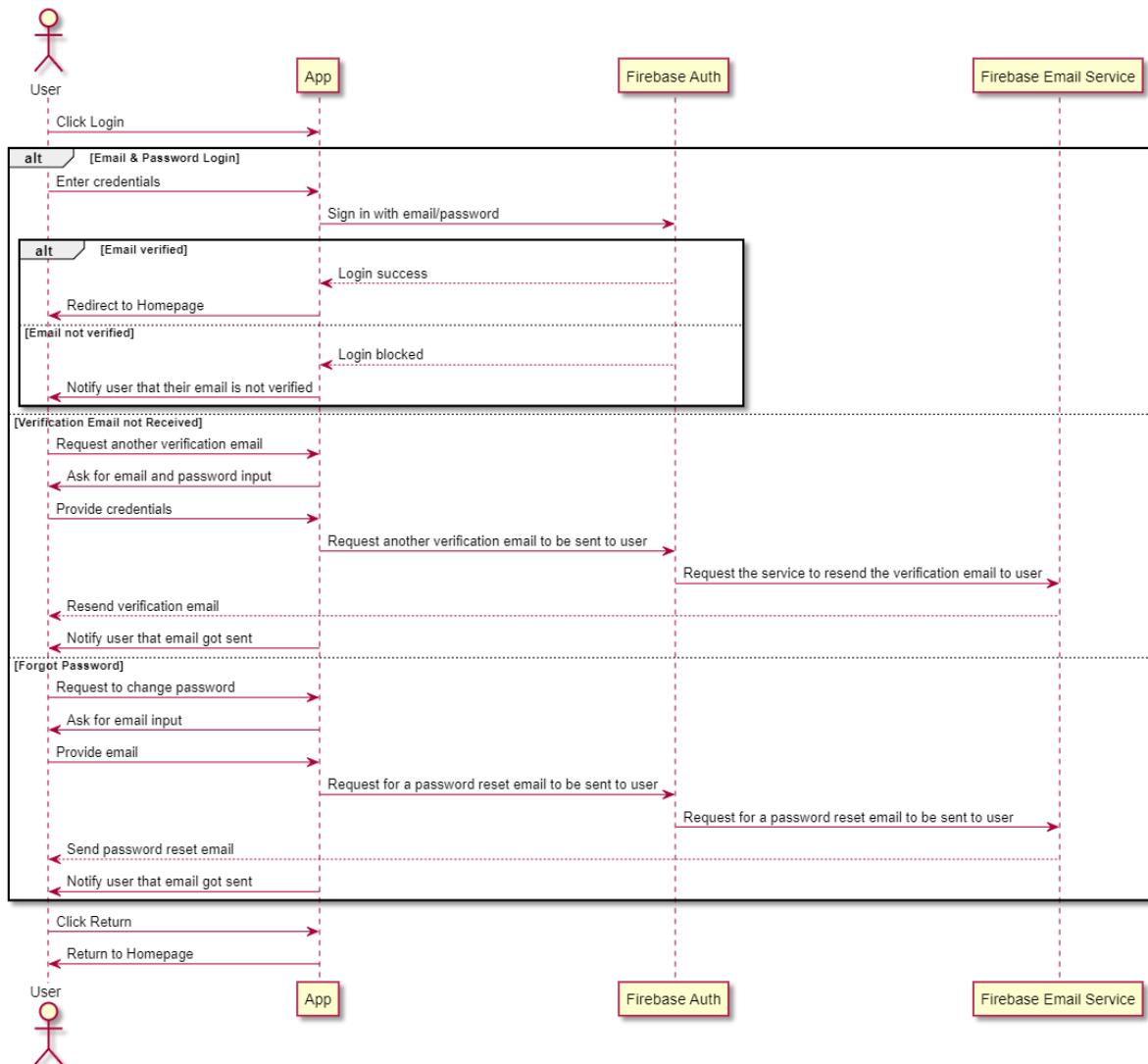
enables the user to quit and return to the homepage at any point in time.



(Created by author)

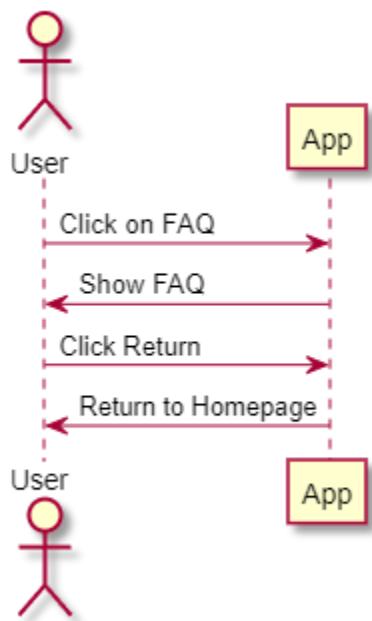
The process of logging in is naturally started by clicking on the login button on the homepage. After that, a user can choose to log in either with email and password or with Google. If they want to log in with email and password, they simply need to input those two credentials and click on the login button. The application then requests Firebase authentication to verify whether the user's email has been verified. If it has, the user is successfully logged in and redirected to the

homepage. If it hasn't, the user is notified, and their login request is denied. If the user either hasn't received the verification email or they've lost it, they can simply request another one. This is simply done by clicking on "Resend verification email" and then inputting the email and password correctly and clicking "Resend". The rest of the process is simply done via an email that the user receives. It is also possible for a user to forget their password. In that case, they can just click on "Reset password." After doing that, the user only has to input their email and click "Resend," and then the rest of the process is done via an email that they receive. The user can also just click quit to get back to the homepage if they want to do so, or click "return" if they are in the password reset form or resend verification email form in order to get back to the login form first.



(Created by author)

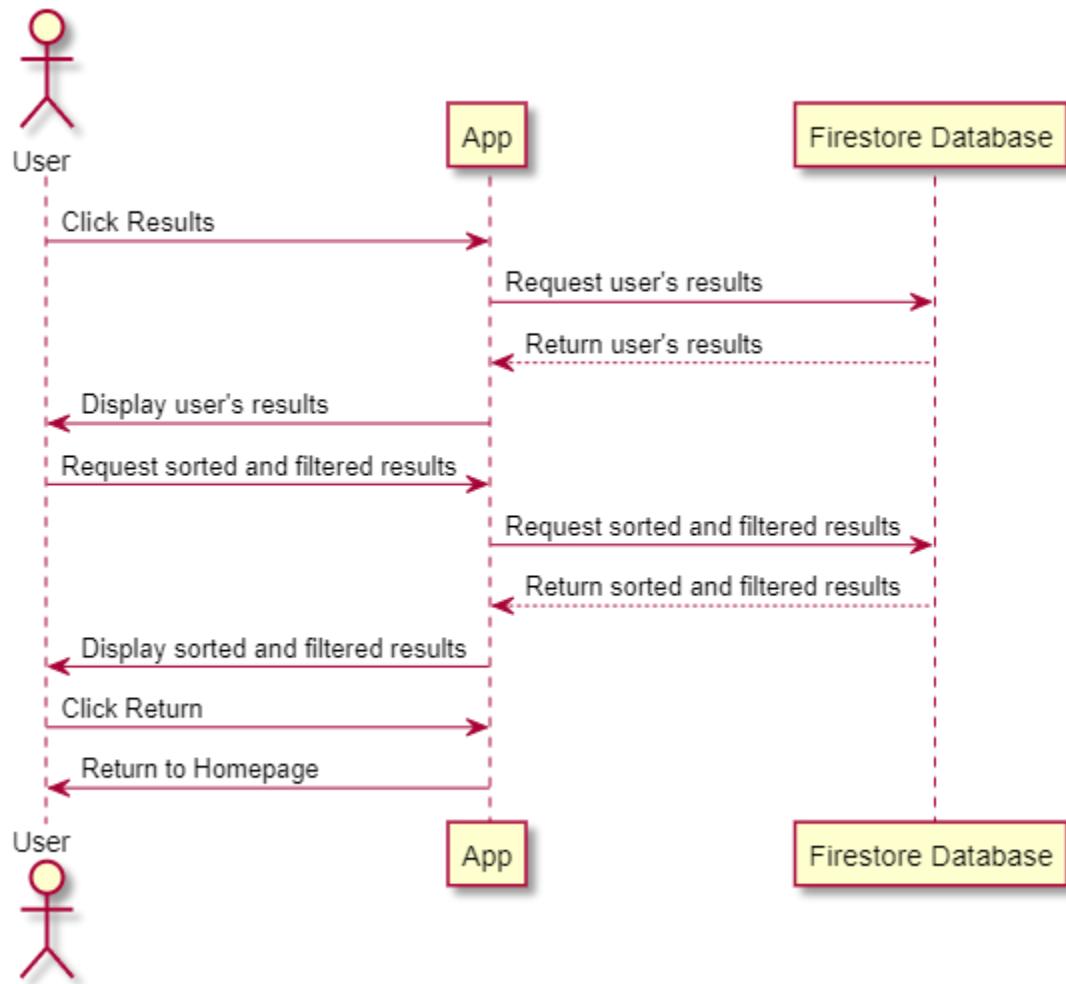
Another simple process that all of the users can do is opening and reading the frequently asked questions. This is simply done by clicking on the FAQ button. After that, the frequently asked questions and their corresponding answers are shown to the user. The user can simply click on the return button to get back to the homepage. It is important to mention that the frequently asked questions and answers are stored in the Firestore database and are fetched whenever the page is loaded or reloaded. They are then saved locally, so when the user clicks on the FAQ button, that same saved data is what is being displayed. This allows for a bit more flexibility in terms of adding and removing some questions and answers in the future.



(Created by author)

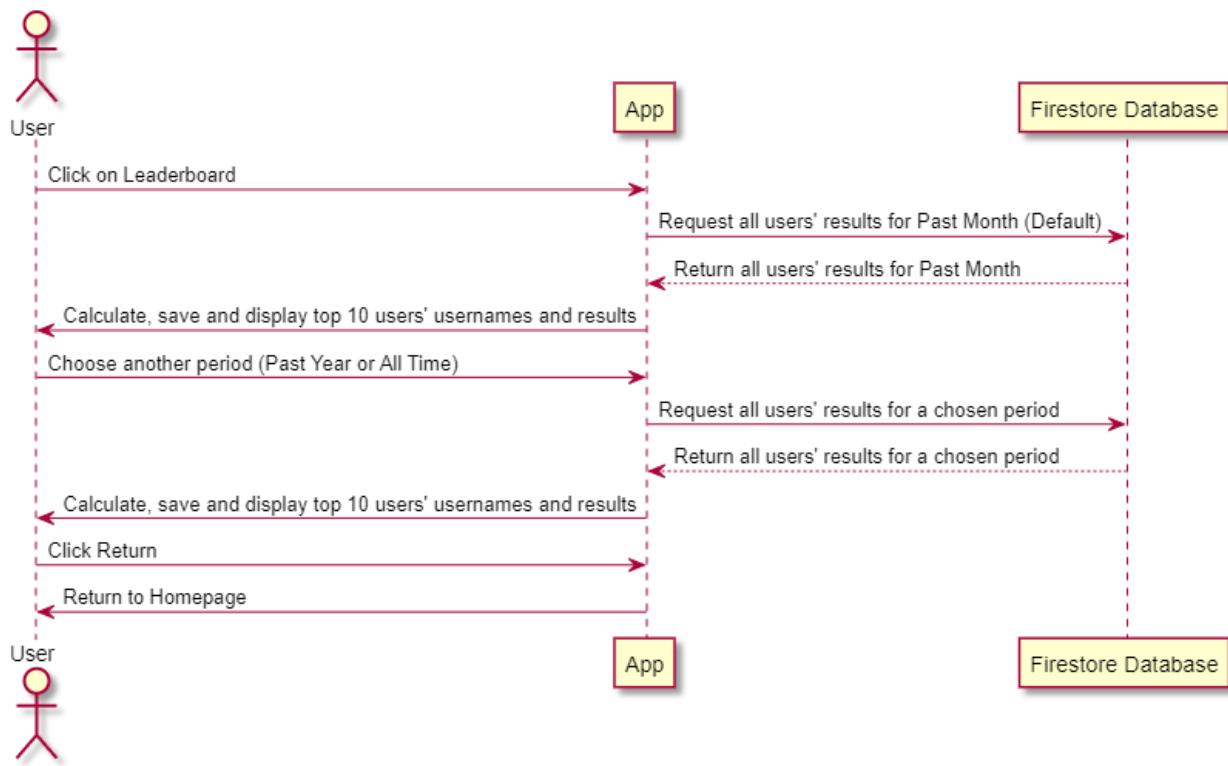
Logged-in users can start the process of viewing their past results by simply clicking on the Results button. It is worth mentioning that logged-out users can also click on this button, but then they are simply welcomed by a message saying, “You are not logged in. Please log in to see your result’s”. Then they can only click on the return button and get back to the homepage. On the other hand, when a logged-in user clicks on the results button, the application requests their past results from the Firestore database. The initial request is sent for the first 10 results sorted by

timestamp, meaning that the results for the 10 quizzes that the user has most recently solved will be the ones to be displayed first. If the user has solved more than 10 quizzes in total, more results are dynamically fetched by scrolling. The results display all of the saved data, namely type, topic, difficulty, score, time spent (on solving the quiz), leaderboard score, and when the quiz was taken. The user can also sort and filter their results. Results can be sorted by score, difficulty, time spent, taken on (timestamp), and leaderboard score, either in an ascending or a descending order. Users can also filter their results by type (multiple choice, timed, manual input) and indicators (economic, demographic, other). Appropriate indices have been created in Firebase Firestore in order to handle these complex queries. Users can also decide to get back to the homepage at any point by simply clicking on the return button.



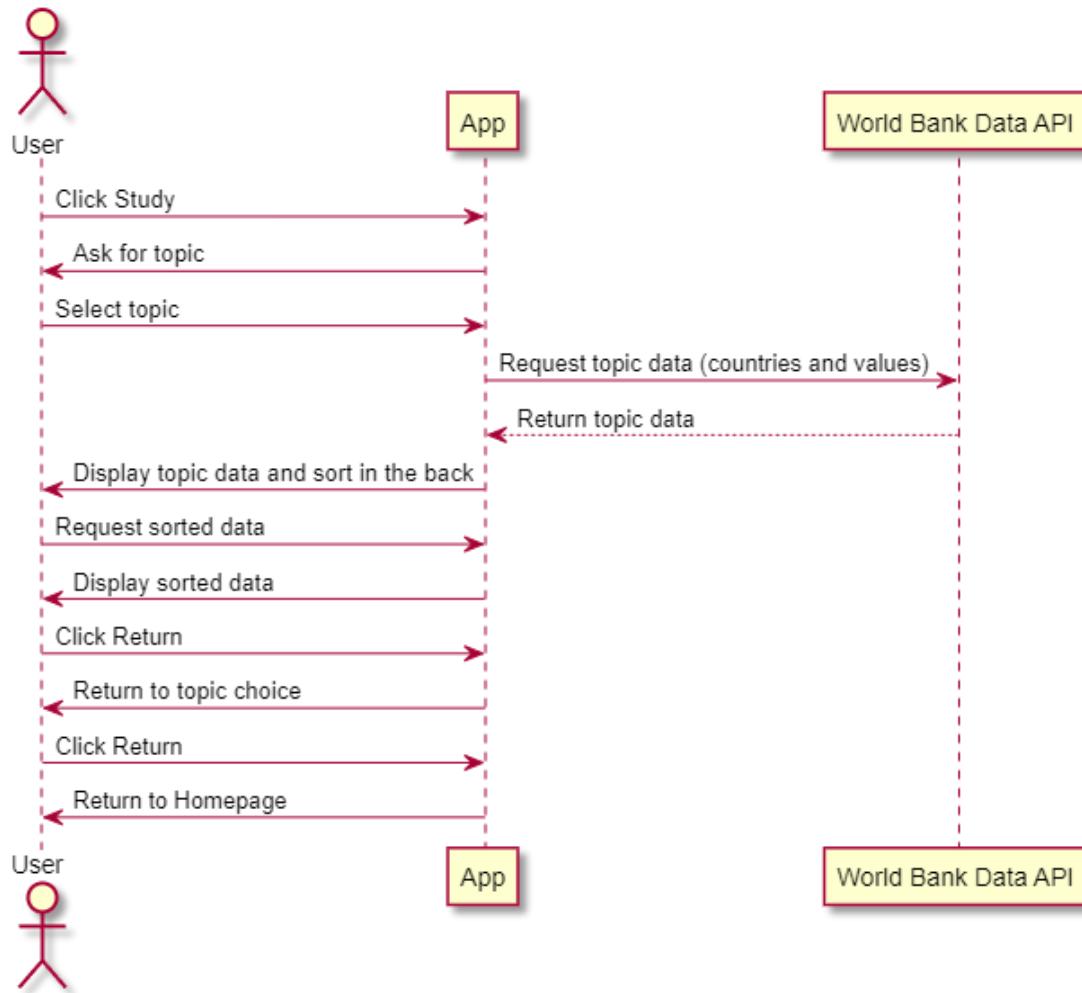
(Created by author)

All users can click on the leaderboard button to see the leaderboard(s). When the user clicks on the button, the app sends a request to the Firestore database to fetch the best leaderboard scores over the last month. If one user has multiple leaderboard scores that could theoretically appear in the top 10, only their best results over the given period are chosen. The top 10 users are picked, and their usernames are displayed along with their best leaderboard score. This data is then saved locally. Users can also choose to see the top players of the past year or all time. In that case, the data is fetched again, and the same process is repeated, only for a different time frame. Since the results are now saved locally, if the user decides to view the best users of the past month again, there is no need to fetch the data again. Instead, the locally saved data is just displayed. The best user's data appears in the gold color, the second best user's data in silver, and the third's in bronze. The rest of it appears in the brand color. Users can click on the question mark in the top right corner to read the instructions on how the leaderboard scores are calculated. Users can also simply return to the homepage whenever they want to do so by simply clicking the return button.



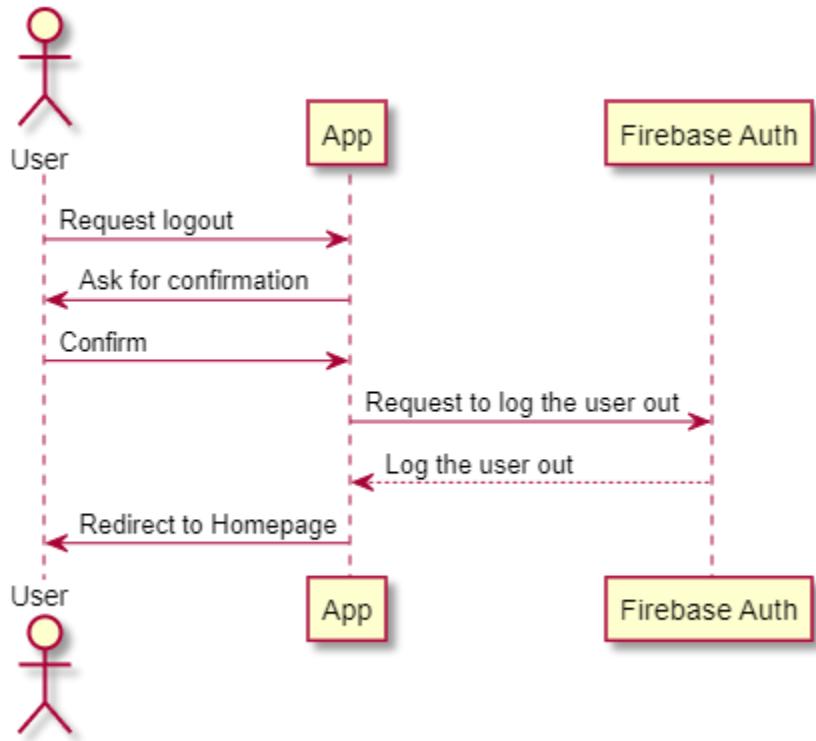
(Created by author)

Another interesting option for users is to study. The process begins when a user clicks on the study button on the homepage. They are then prompted to choose a topic to study. The topics to study are exactly the same as the topics to take quizzes in, and the data is fetched from the exact same source as well (World Bank Data API). When a user clicks on their topic of choice, the data is fetched from the API and displayed to the user. The user is also notified of which year the data is from. They can now view the data, but they can also sort it. Data can be sorted by country names or value, in an ascending or a descending order. It is worth mentioning that the data is actually immediately sorted when it's fetched, so choosing a different sorting option does not trigger the sorting to begin, and instead simply displays the already sorted data on the screen. Users can click the return button to return to the topic choice, and they can then click the return button again to return to the homepage.



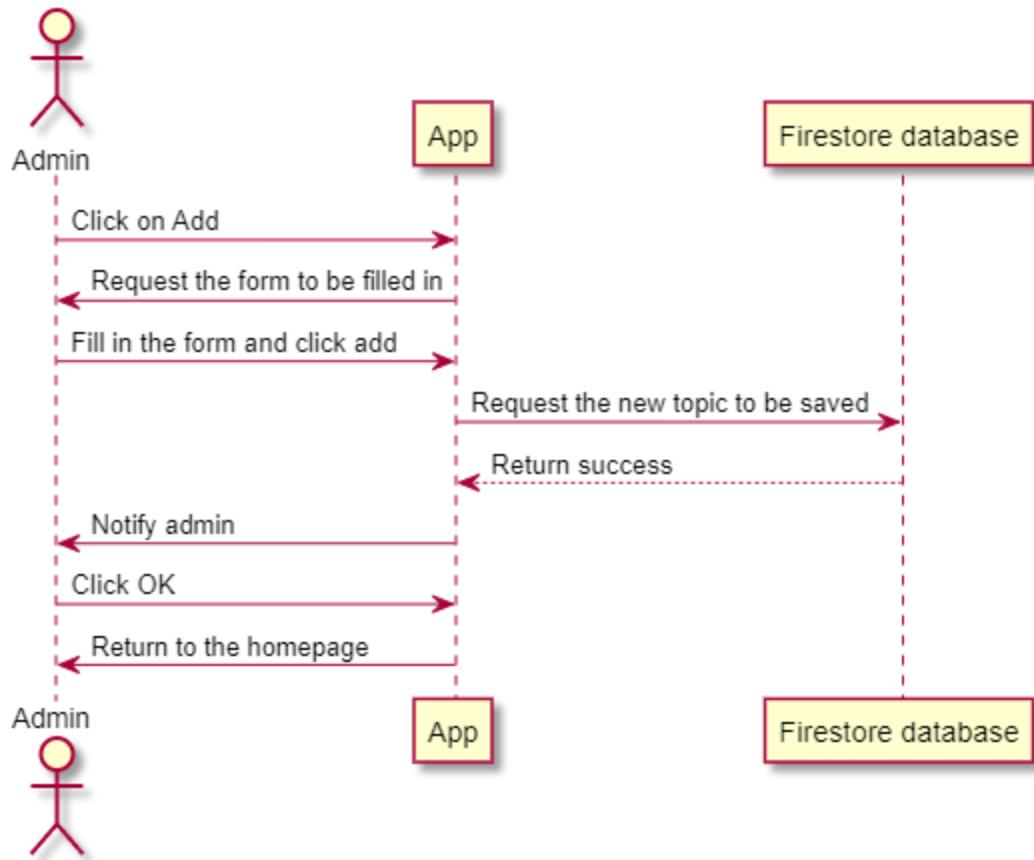
(Created by author)

Obviously, users can decide to log out whenever they want. The process is very simple. A logged-in user clicks on the logout button and is then logged out via Firebase authentication and brought back to the homepage as a logged-out user. Of course, the user is first asked to confirm if they really want to log out, as this improves the user experience.



(Created by author)

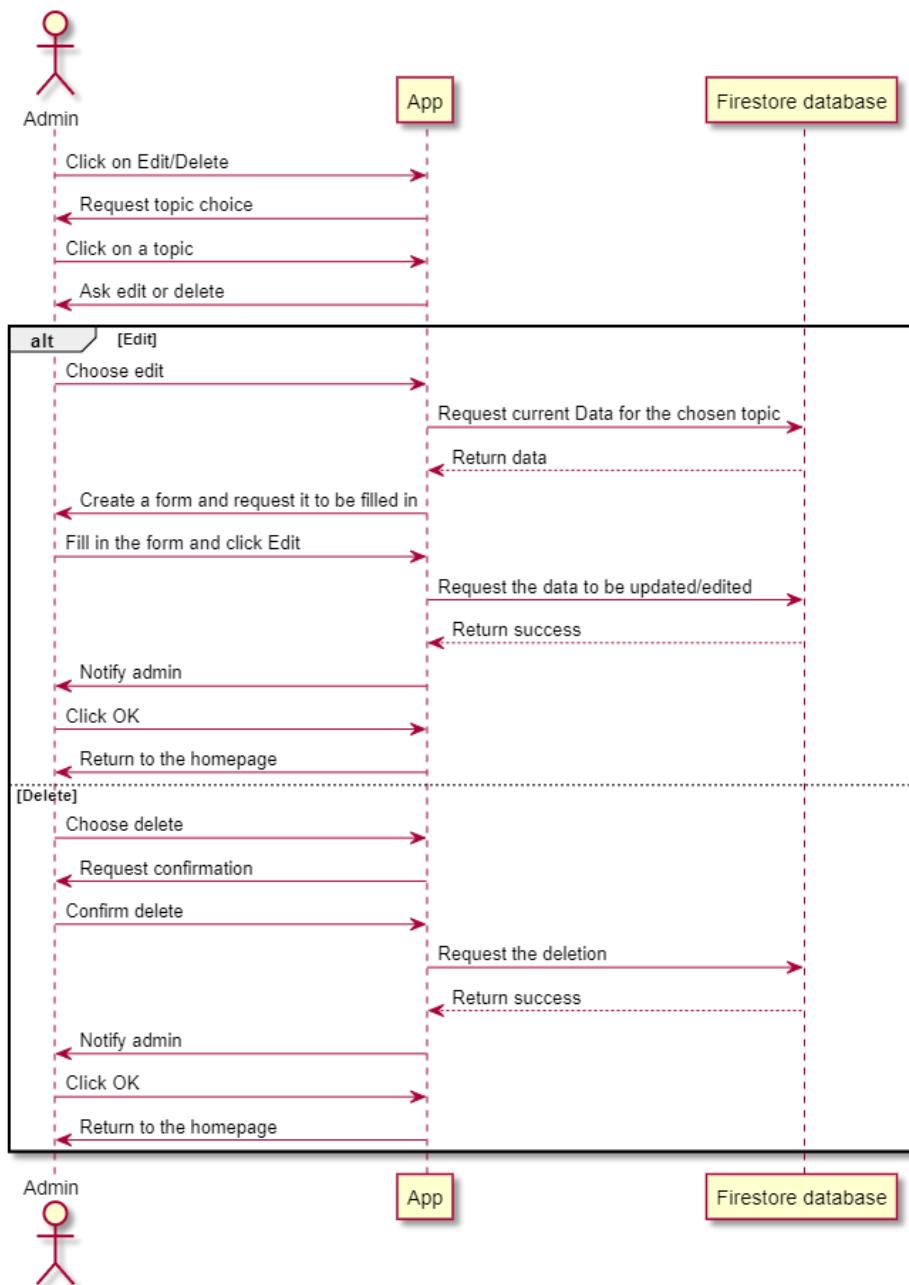
Admins can also add, edit, and delete quiz topics. The process of adding a new one is pretty straightforward. An admin clicks on add and then fills in the user-friendly form to add a new topic. After the admin clicks on the add button, the app sends a request to the Firestore database to store this new data. The Firestore database either returns success or failure. Under normal circumstances, it returns success, and the admin is notified. The admin can then just click the OK button to get back to the homepage. Everything is pretty intuitive. The admin can alternatively click on the question mark button in the top right corner to read the instructions on how the form should be filled in. They can also just click on the return button to get back to the homepage.



(Created by author)

In order to edit or delete a topic, the admin can click on edit/delete. The admin is then requested to choose a topic to be edited or deleted. After that, they are prompted to choose whether they want to edit or delete the chosen topic. If they choose to edit it, the app sends a request to the database to fetch the data that is currently saved for the topic of choice. That data is then returned and used to create the editing form. That form is then displayed to the admin, and they can change whatever they want but should stick to the rules. The rules can be viewed by clicking the question mark button in the top right corner. The admin can then simply click edit and confirm that they really want to do so. The app then requests the Firestore database to save the new, updated data, and the database hopefully does so. Either way, a message is then displayed to the admin, and they can just click OK to go back to the homepage. Alternatively, the admin can choose to delete a quiz. They are then asked for a confirmation where the yes button is red in order to signal the danger. Once the admin confirms their desire, the app requests the Firestore

database to delete the data for this given topic, and the database sends back a response. The response is shown to the user, and they can then just click OK to get back to the homepage. Of course, admins can use the return buttons to navigate back to the homepage on their own whenever they feel like it.



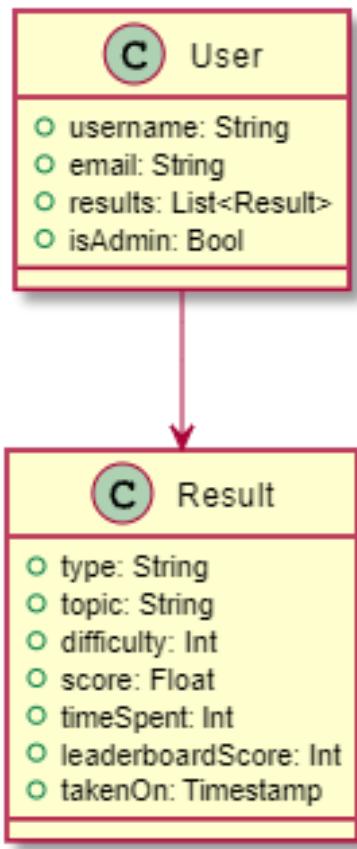
(Created by author)

Some functionalities are not worth including in the UML diagram but can be mentioned nevertheless. Popups are used in various parts of the application, such as the request to choose a type of quiz to play, confirmations for various actions, instructions in some parts of the app, and the sorter/filter in the results section. Users can close all of these popups by simply clicking the escape button on their keyboard. All popups have another way of closing them, namely a button that says ‘close’ or ‘no.’ The homepage includes a few images for large-screen users, and so do some other parts of the application. This is meant to beautify the UI. The forms used in the add and edit sections include input and dropdown elements. The dropdowns are opened by clicking on them and are closed when a new dropdown is opened or when a user clicks outside of the dropdown. Most of the pop-ups in the application can also be closed by clicking outside of them. The sorter in the study section changes its look based on scrolling and screen size and can be closed by clicking outside of it as well. The app also allows the user to play a chill lofi music track by clicking on an intuitive-looking button on the left. The icon inside of the button changes depending on whether the music is being played or not. The user can stop the music at any point by simply clicking the button again. If they decide to then click on it again, the track will continue playing from where it was stopped, unless the page was refreshed or closed in the meanwhile. The track also loops. The timed quizzes include an animation that shows numbers going from 5 to 1 and is accompanied by sounds that sound like a clock. The first second has two sounds, while the other four have only one. These sounds stop when the quiz ends. The manual input quizzes show the accurately formatted version of the number that the user is typing and do not allow the user to move on to the next question if the input value is not a valid number. For example, if the quiz is about nominal GDP and the user input equals 10000000, the displayed formatted number will appear as \$10,000,000. If the user inputs 10e, they will be asked to input a valid number before moving on to the next (or previous) question. Similar guards exist in the signup form, the login form, and the admin-related forms. Users always get clear messages as to what they are missing. Everything is very clear and intuitive. The background image of the application is always the same, except in the study section, where it is replaced by an image of a library. The favicon was generated by ChatGPT, and the colors match those on the website.

As far as Firebase is concerned, users are saved both in Firebase authentication and the Firestore database. In Firebase authentication, most of the process is automated and was already described earlier. Firestore contains each registered user’s username, email, results, and admin status.

Usernames and emails are saved as strings, results are saved as a collection of the result documents, and admin status is saved as a boolean value. Each result contains type, topic, difficulty, score, time spent, leaderboard score, and the timestamp. Type and topic are saved as strings. Difficulty is saved as an integer value, since that allows sorting later on. The integer value is transformed back to a string when results are shown to the user. Score is a floating integer, and leaderboard score is a regular integer. Of course, all numbers are simply saved as a number value in the Firestore database. Time spent is also saved as a number and is then transformed into a format that displays minutes and seconds separately, e.g., “1m 40s.”

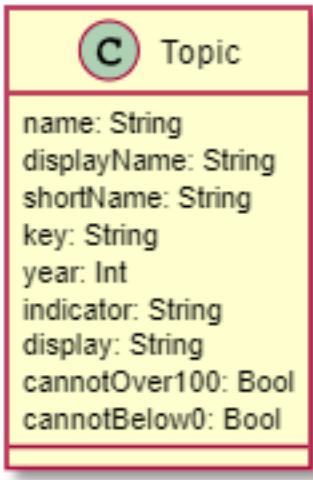
Timestamp is saved as a timestamp, which is a data type that exists in Firebase Firestore.



(Created by author)

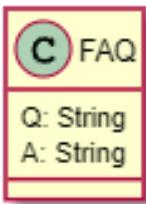
Topics are saved in Firebase Firestore, and the data is used for the core feature of Econ Quiz. Name, as the name suggests, represents the name of the topic (e.g., GDP PPP). This name is allowed to differ from the official one from World Bank Data, but it has to be clear and concise.

This is because displayName will have to fit in the buttons. DisplayName, therefore, is how the name appears in the buttons. It is essentially the same as the name, but “\n” is added where appropriate so that the display names that stretch over two lines appear in a more visually appealing manner. ShortName simply represents a part of the URL, meaning it is used by the Vue router to create a URL. It should generally be a bit more concise than the name but still meaningful and clear enough. Key is used for fetching data from the World Bank Data API, which can be found on their website. GenAI is also pretty helpful in figuring out the appropriate keys for indicators (topics). Year determines which year the data is fetched for. The data should not be more than 5 years old, but it also has to contain values for at least 50 countries. It should be the most recent year, given these restraints. This is usually last or the one before that, since the World Bank Data usually has a major update every April and a smaller one in September. Therefore, the data is for the year before the previous until April and then for the previous year. The indicator represents which of the three indicators appearing on the website (economic, demographic, or other) a given topic fits in the best. It is actually a group of indicators, but named indicators for simplicity. Display determines how the numbers (values) are displayed. LargeNums is used for large numbers that are not money-related (e.g., population), smallNums for small numbers that are not money-related (e.g., fertility rate), largeNumsDollars for large numbers that are money-related (e.g., GDP Nominal), and smallNumsPercentages for small numbers that are money-related or otherwise expressed in percentages (e.g., inflation, arable land (% of land area)). CannotOver100 is used to classify which data cannot have a value higher than 100 (e.g., internet users (% of the population)) and cannotBelow0 is used to classify which data cannot have a value smaller than 0 (e.g., population 0-14 (% of total)). Name, displayName, shortName, key, indicator, and display are all saved as strings. Year is saved as a number (integer) and cannotOver100 and cannotBelow0 are saved as boolean values.



(Created by author)

Frequently asked questions and the corresponding answers are also saved in the database, and in a very simple way. Both the question and answer are saved as a string value.



(Created by author)

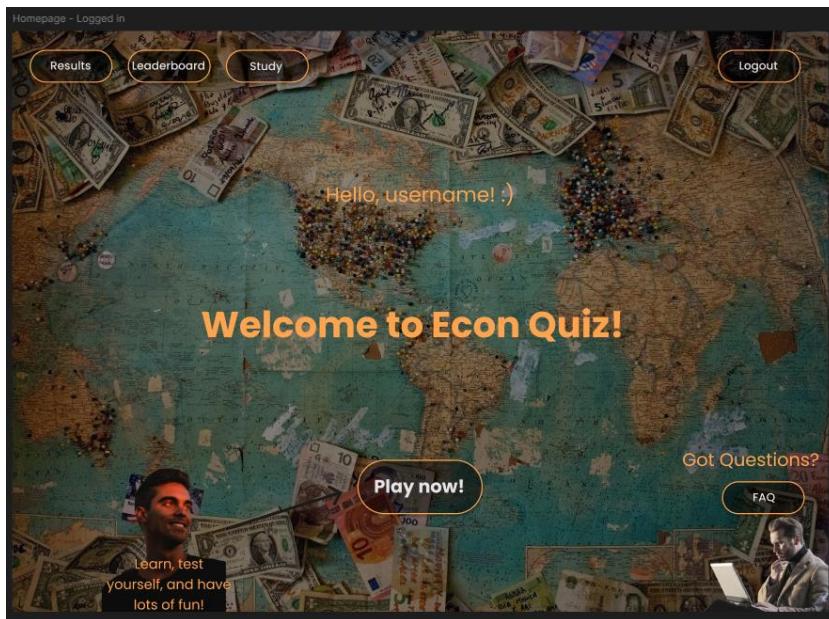
Before the coding even began, the application was sketched in Figma. While some of the screens and functionalities were later added and some changed their appearance, the app largely resembles the initial sketches. The ideas for the desktop screens will be shown first, followed by the ideas for the mobile screens.

The initial idea for the homepage is very similar to the end product. When a user is logged out, it looks like this:



(Created by author)

After the user logs in, it looks like this:



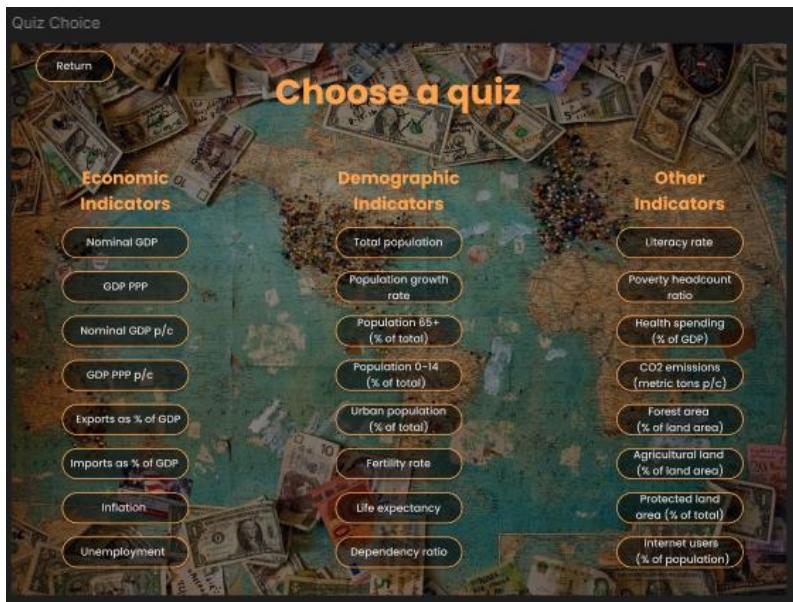
(Created by author)

If the user is also an admin, the homepage looks like this:

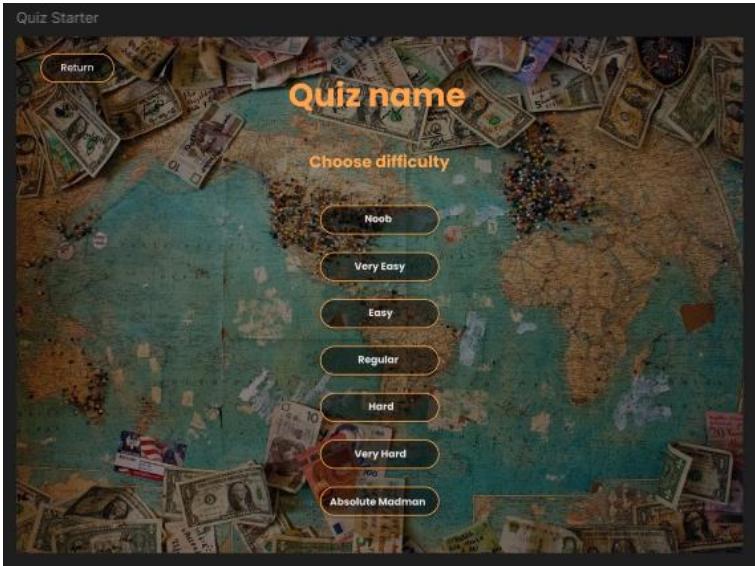


(Created by author)

If a user decides to play a quiz, their experience will look something like this:



(Created by author)



(Created by author)



(Created by author)



(Created by author)

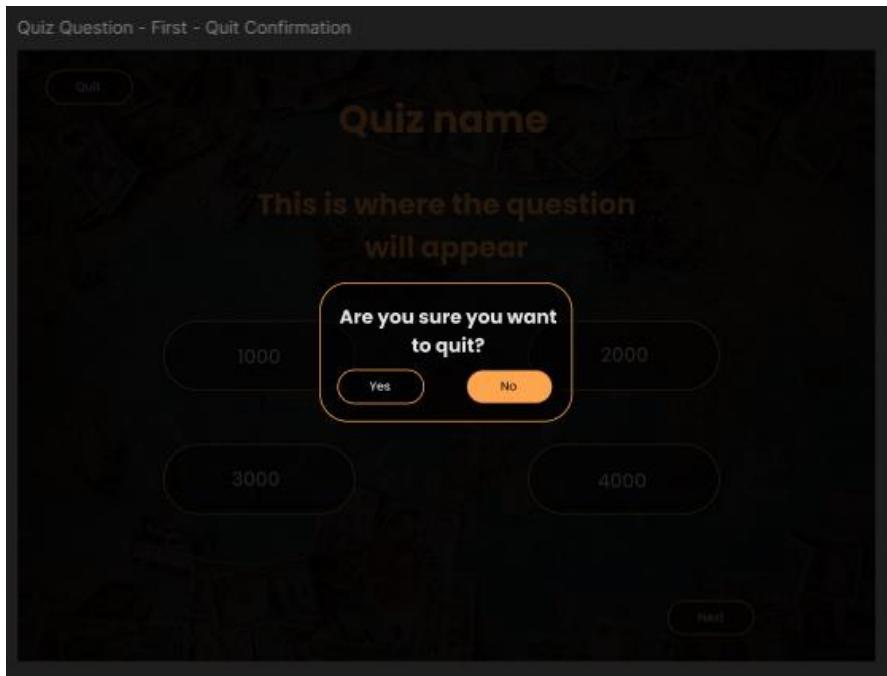


(Created by author)

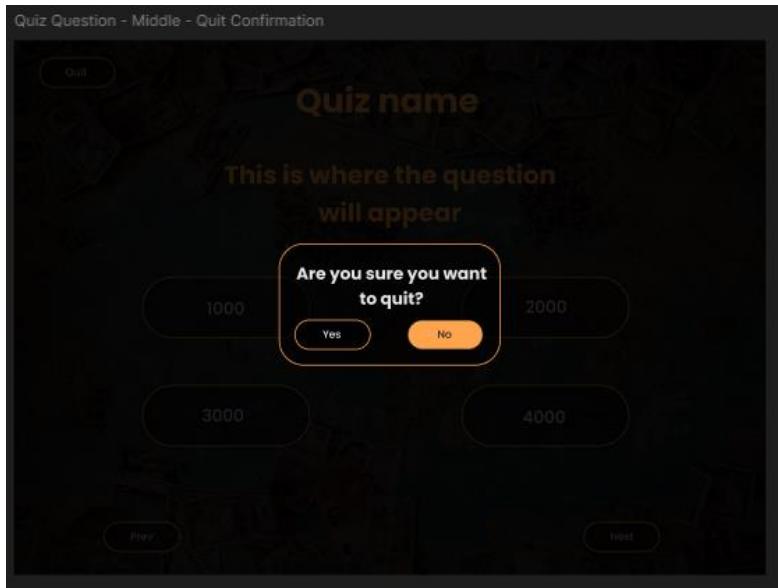


(Created by author)

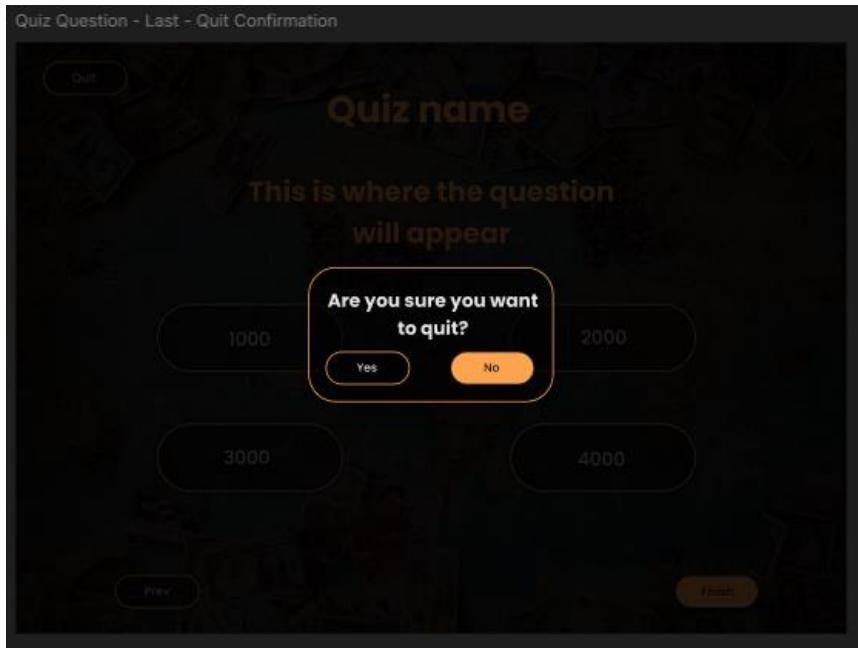
User can decide to quit at any point, and is then met with a popup asking for confirmation.



(Created by author)



(Created by author)



(Created by author)

If the user decides to study, their experience will look like so:

Study - Choice

Return

Choose a Topic to Study

The page features a collage of international banknotes from various countries at the top. Below the title, there are three main categories of indicators:

- Economic indicators**: Nominal GDP, GDP PPP, Nominal GDP p/c, GDP PPP p/c, Exports as % of GDP, Imports as % of GDP, Inflation, and Unemployment.
- Demographic indicators**: Total population, Population growth rate, Population 65+ (% of total), Population 0-14 (% of total), Urban population (% of total), Fertility rate, Life expectancy, and Dependency ratio.
- Other indicators**: Literacy rate, Poverty headcount ratio, Health spending (% of GDP), CO2 emissions (metric tons p/c), Forest area (% of land area), Agricultural land (% of land area), Protected land area (% of total), and Internet users (% of population).

(Created by author)

Study - Main

Return

Library

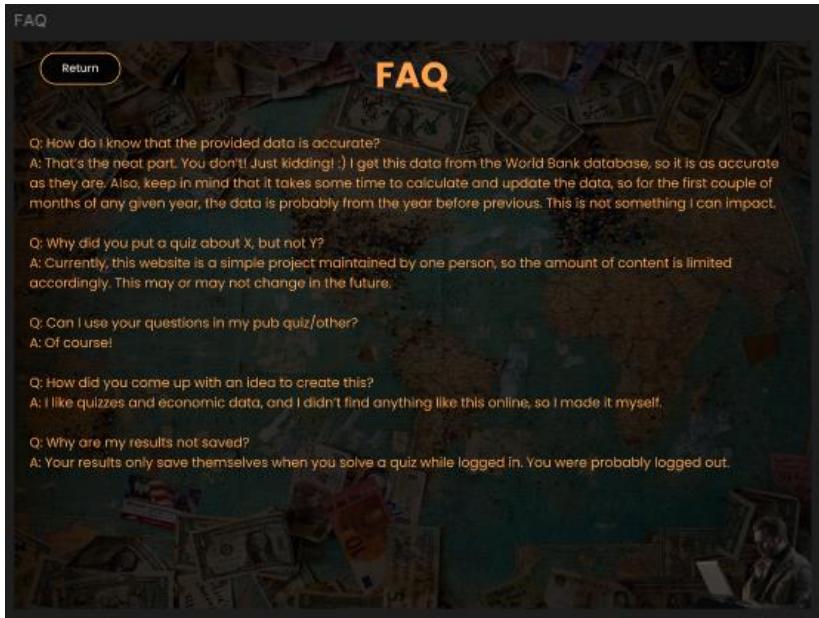
Data: Category name is this and that

The page features a collage of international banknotes at the top. Below the title, there is a heading "Data: Category name is this and that". The data is presented in a table with two columns: "Country" and "Data".

Country	Data
Albania	7000
Antigua and Barbuda	11000
The United Kingdom	52000
Malawi	1400
Nepal	4700
Japan	41000
Brazil	13000

(Created by author)

The FAQ section will appear like this:



FAQ

Return

FAQ

Q: How do I know that the provided data is accurate?
A: That's the neat part. You don't! Just kidding! :) I get this data from the World Bank database, so it is as accurate as they are. Also, keep in mind that it takes some time to calculate and update the data, so for the first couple of months of any given year, the data is probably from the year before previous. This is not something I can impact.

Q: Why did you put a quiz about X, but not Y?
A: Currently, this website is a simple project maintained by one person, so the amount of content is limited accordingly. This may or may not change in the future.

Q: Can I use your questions in my pub quiz/other?
A: Of course!

Q: How did you come up with an idea to create this?
A: I like quizzes and economic data, and I didn't find anything like this online, so I made it myself.

Q: Why are my results not saved?
A: Your results only save themselves when you solve a quiz while logged in. You were probably logged out.

(Created by author)

If a user decides to view the leaderboard, they will be welcomed with this screen:



Leaderboard

Return

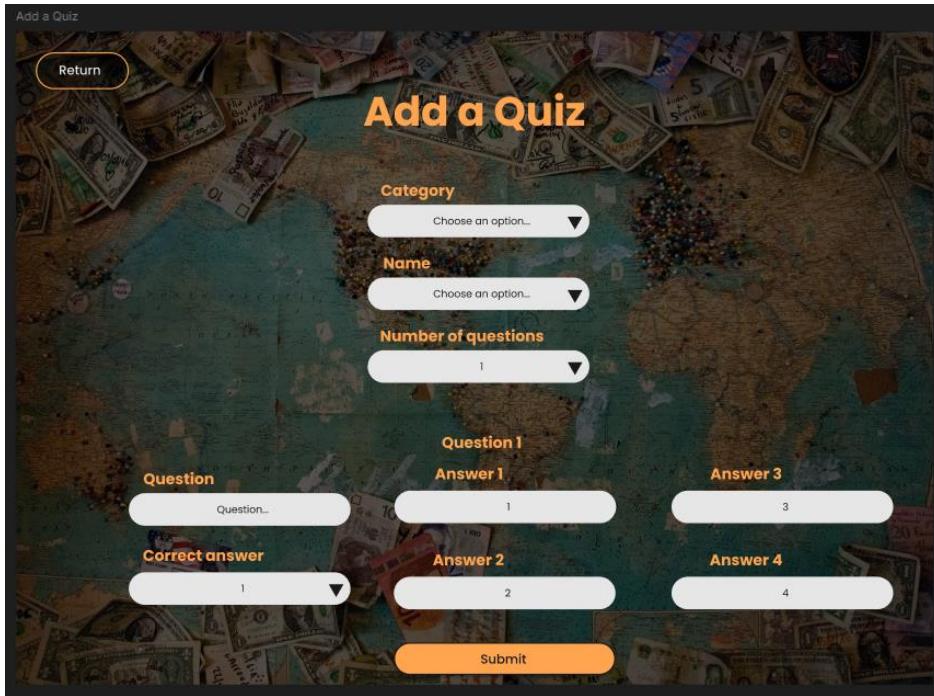
Leaderboard

Past Month **Past Year** **All Time**

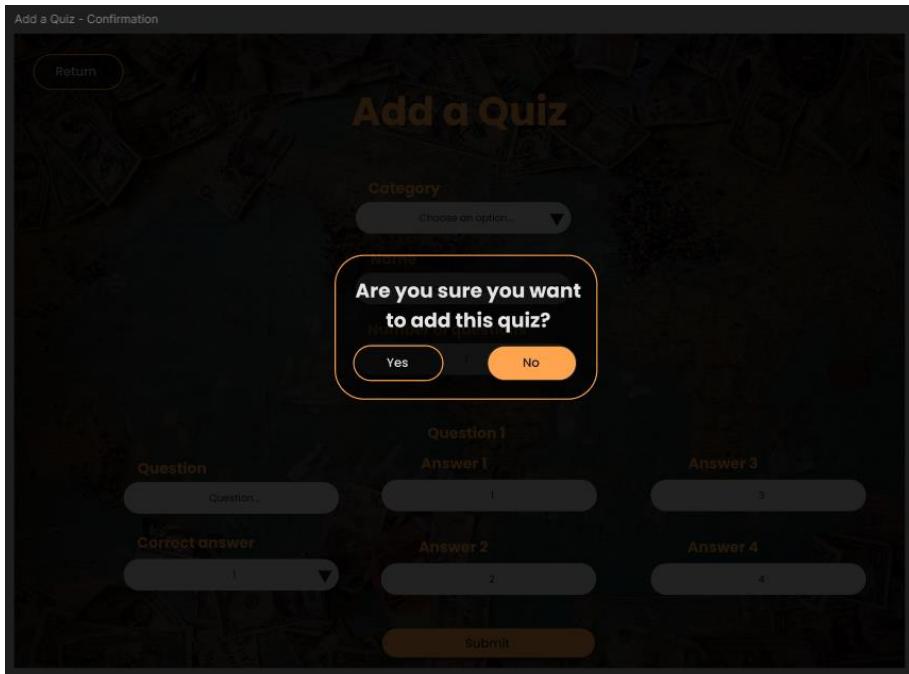
User	Score
ronaldo007	5200
kresol23	4700
andrewlovesecon	3200
mariathelegend	2600
stacystace	2160
batman	2000
cocojellon	1780

(Created by author)

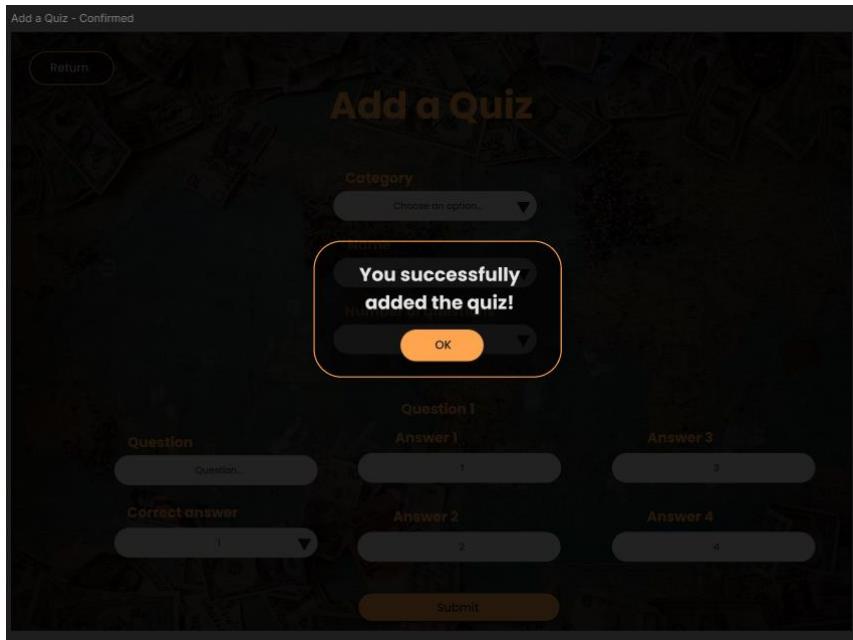
If an admin decides to add a new quiz, the process will look like this:



(Created by author)

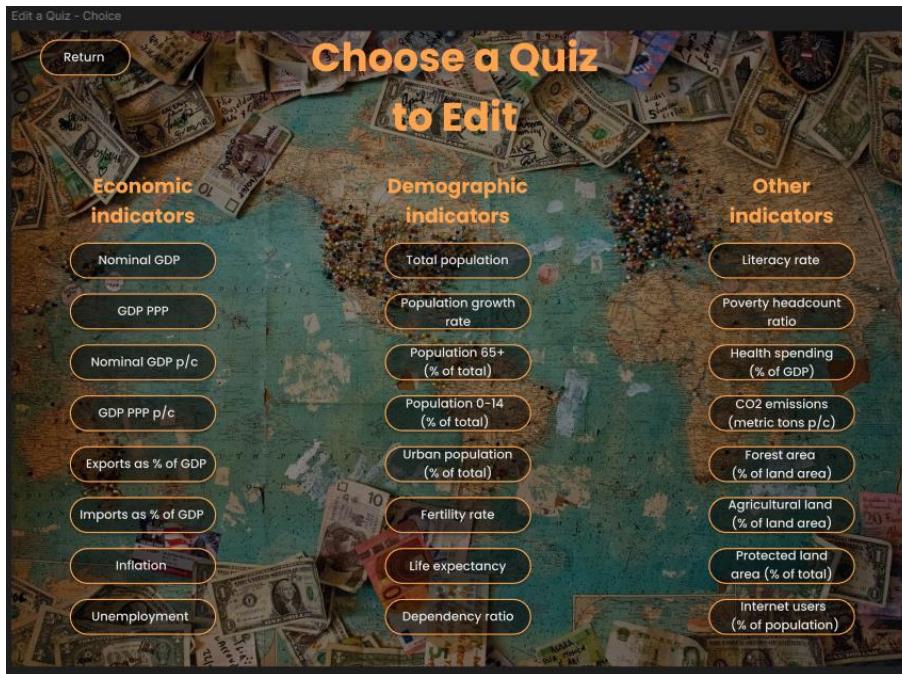


(Created by author)



(Created by author)

Editing or deleting a quiz will go like this:



(Created by author)

Edit a Quiz - Main

Return

Edit a Quiz

Category

Choose an option...

Name

Choose an option...

Number of questions

1

Question 1

Question

Delete

Question...

Answer 1

1

Answer 3

3

Correct answer

1

Answer 2

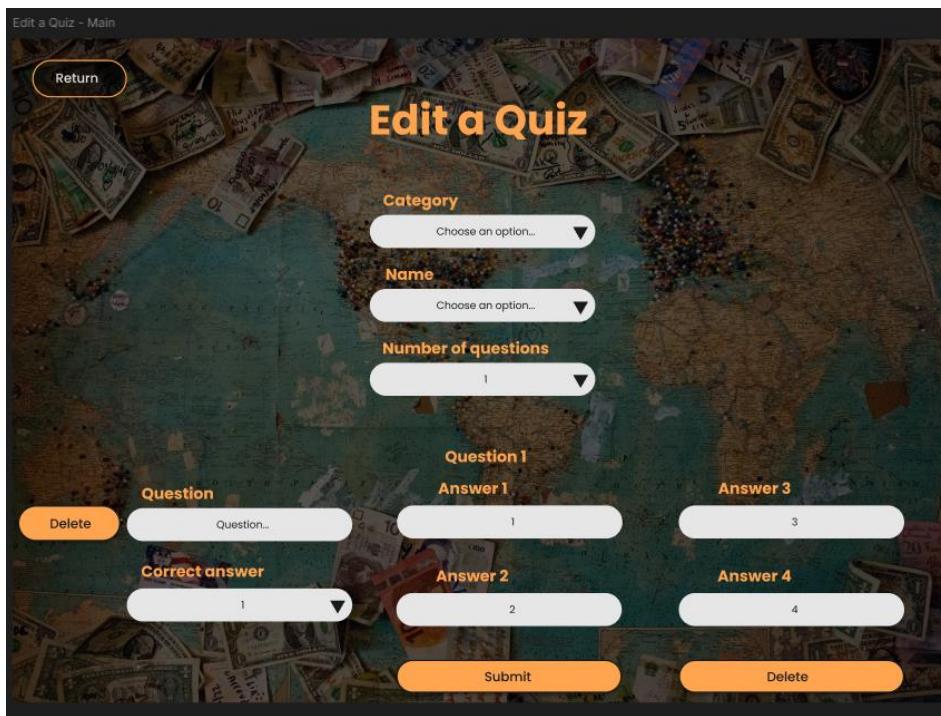
2

Answer 4

4

Submit

Delete



(Created by author)

Edit a Quiz - Confirmation

Return

Add a Quiz

Category

Choose an option...

Are you sure you want
to edit this quiz?

Yes

No

Question 1

Question

Question...

Answer 1

1

Answer 3

3

Correct answer

1

Answer 2

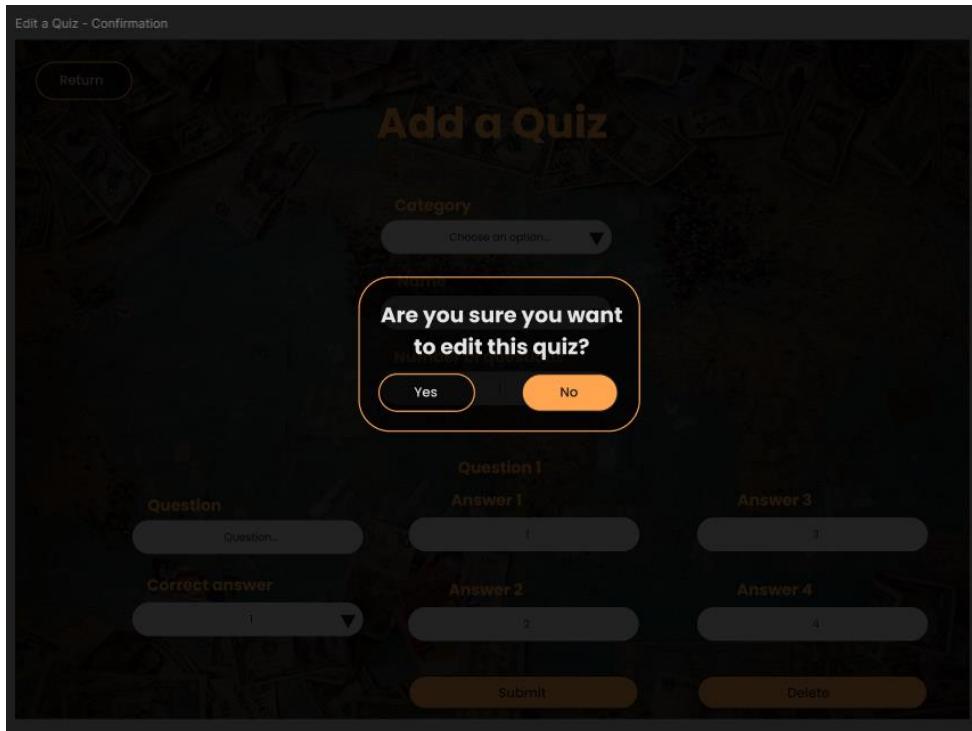
2

Answer 4

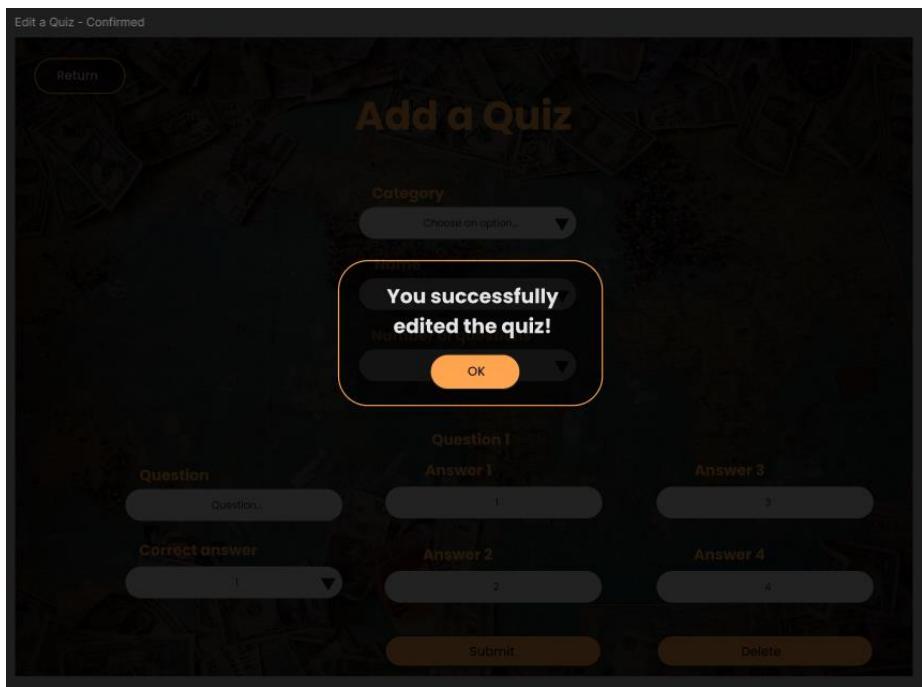
4

Submit

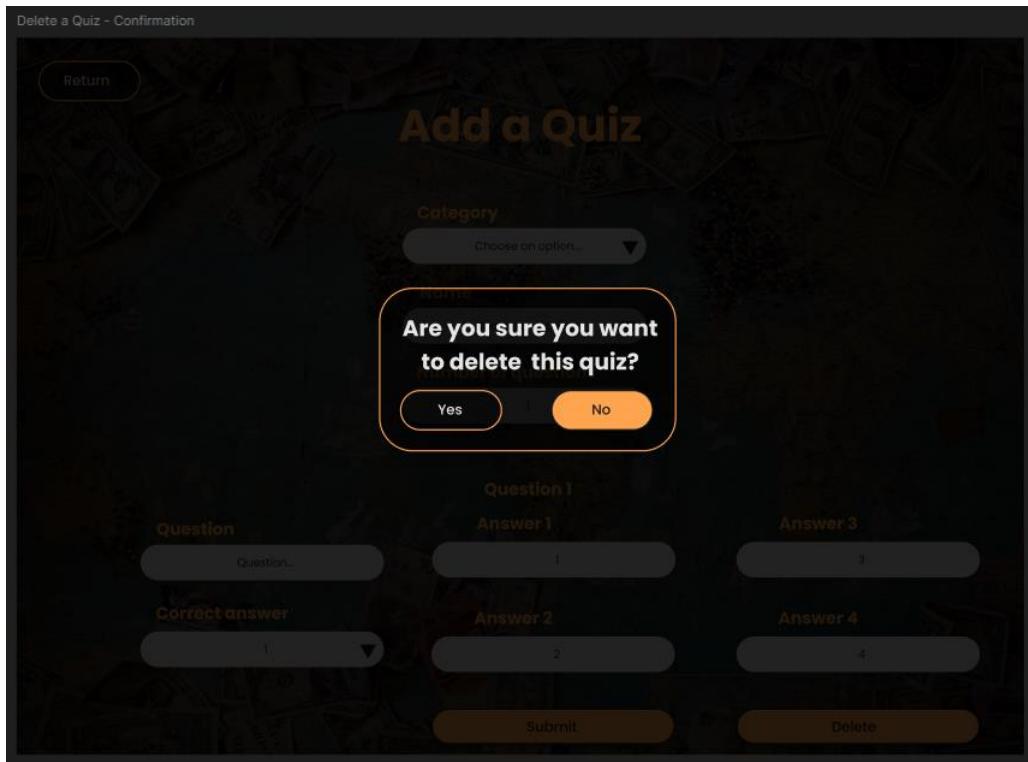
Delete



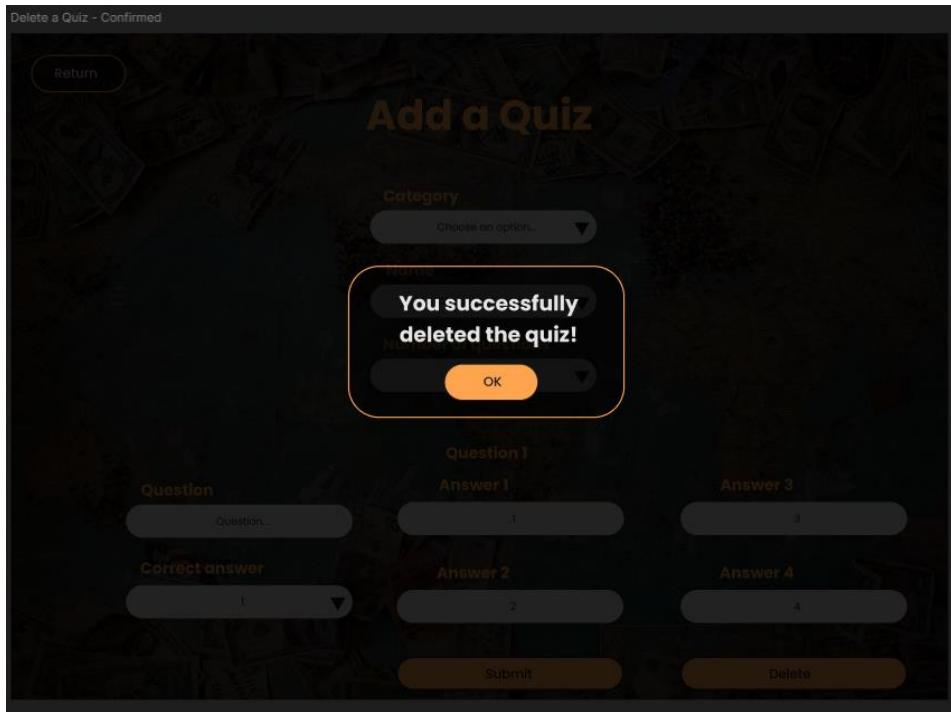
(Created by author)



(Created by author)

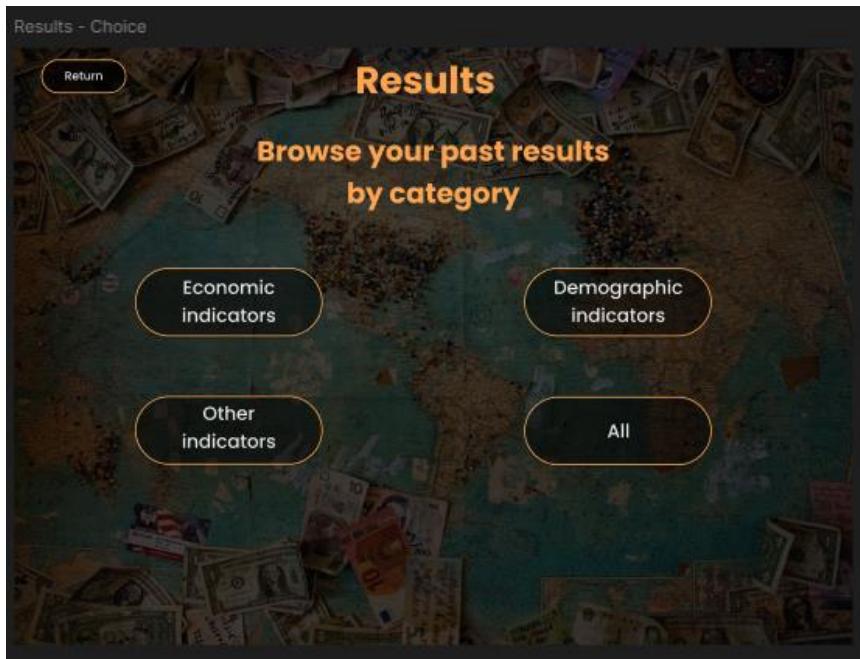


(Created by author)



(Created by author)

Logged in users can also see their past results, which will go something like this:



(Created by author)

Results - Listed

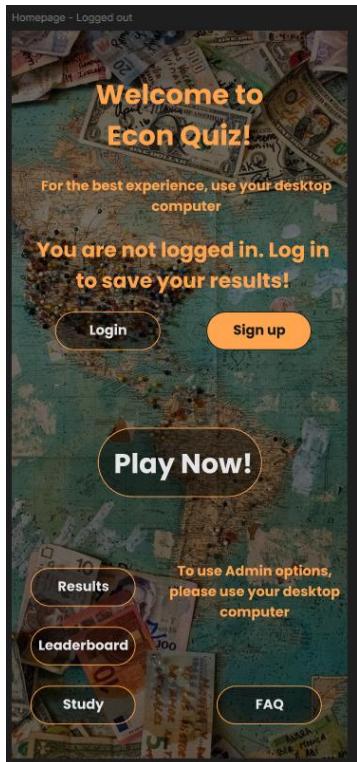
Results

Results: Category Name

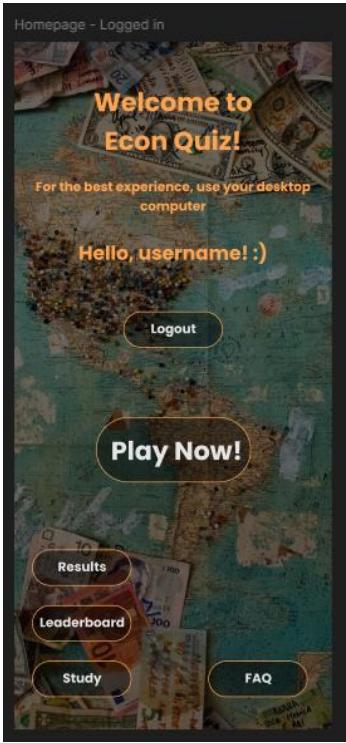
Quiz Name	Difficulty	Result
Nominal GDP	Normal	42/50
GDP PPP p/c	Easy	46/50
CO2 emissions (metric tons p/c)	Very Easy	38/50
Protected land area (% of land area)	Noob	41/50
Internet users (% of population)	Absolute Madman	17/50
Imports as % of GDP	Very Hard	22/50

(Created by author)

The homepage for mobile users is a bit different.



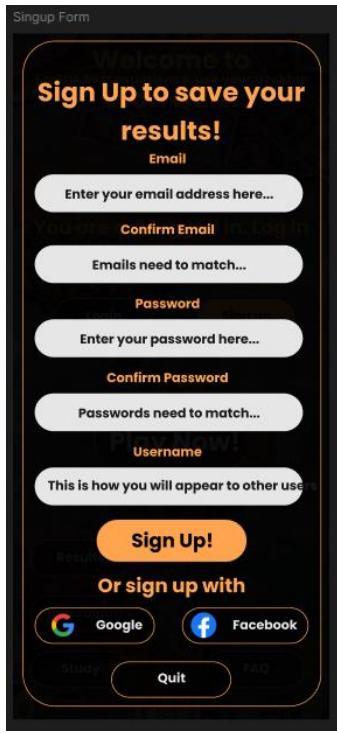
(Created by author)



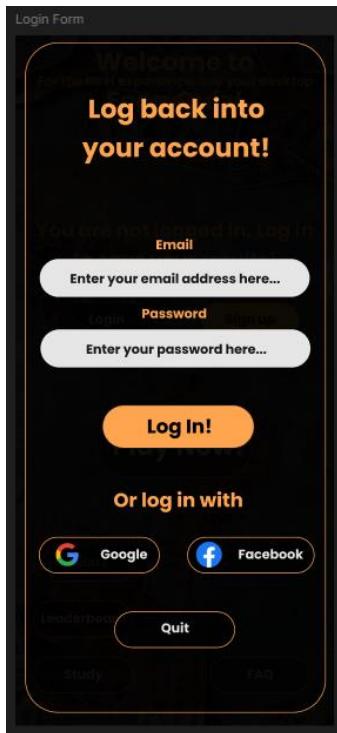
(Created by author)

The rest of it is basically the same, although it was initially planned not to allow users to use admin options on their phones, which was actually enabled later on.

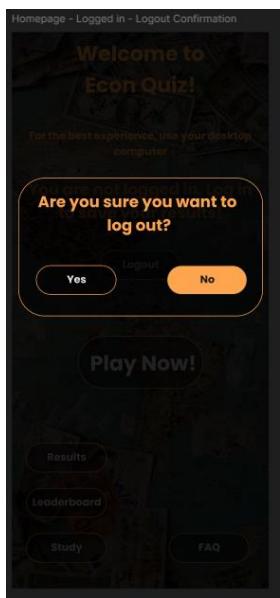
Authentication:



(Created by author)



(Created by author)

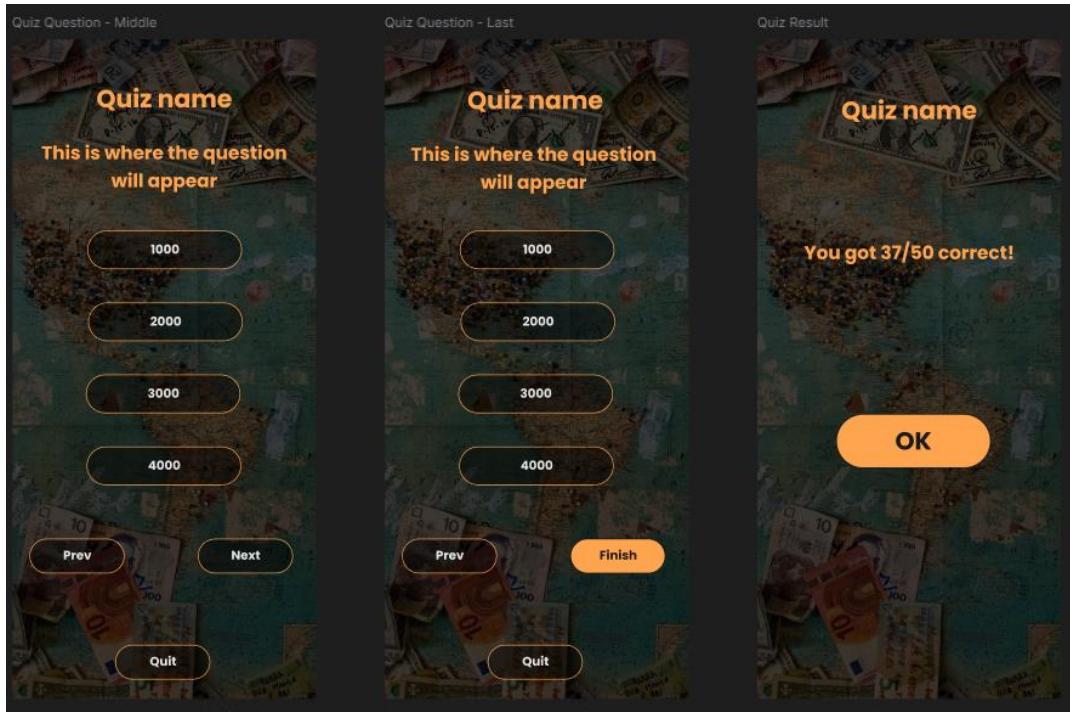


(Created by author)

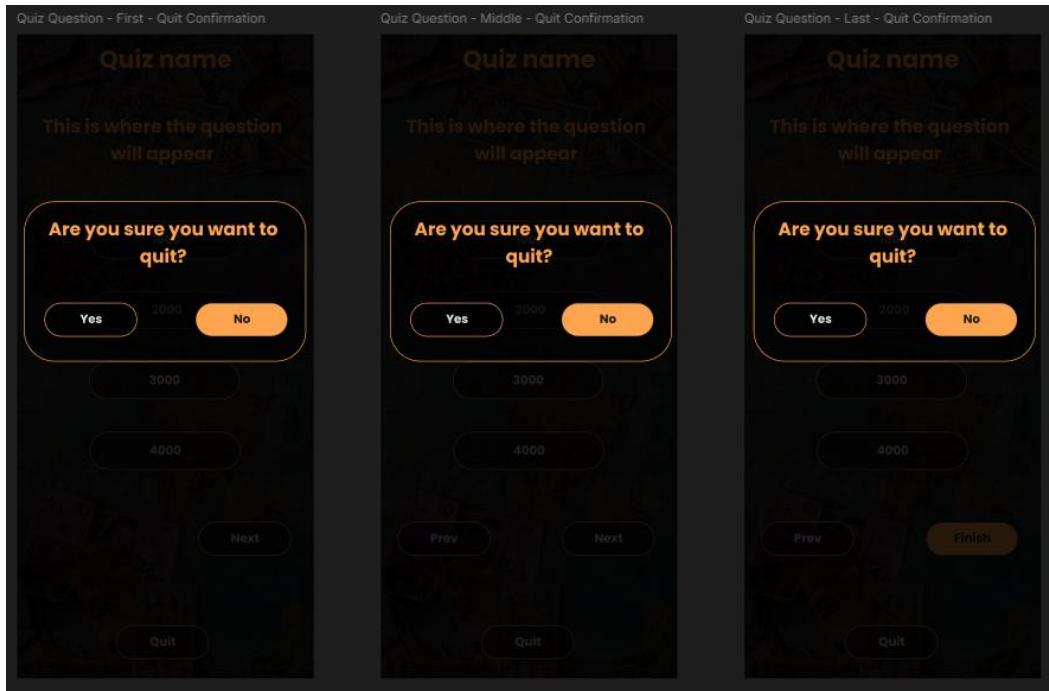
Quiz:

Three screenshots of the Econ Quiz app. The first screen, titled "Quiz Choice", shows a grid of economic indicators: Nominal GDP, GDP PPP, Nominal GDP p/c, GDP PPP p/c, Exports as % of GDP, Imports as % of GDP, Inflation, and Unemployment. The second screen, titled "Quiz Starter", shows a grid of difficulty levels: Noob, Very Easy, Easy, Normal, Hard, Very Hard, and Absolute Madman. The third screen, titled "Quiz Question - First", shows a placeholder text area "This is where the question will appear" and a grid of point values: 1000, 2000, 3000, and 4000. It also includes "Next" and "Quit" buttons.

(Created by author)



(Created by author)



(Created by author)

Study section:

Study - Choice

Economic Indicators

- Nominal GDP
- GDP PPP
- Nominal GDP p/c
- GDP PPP p/c
- Exports as % of GDP
- Imports as % of GDP
- Inflation
- Unemployment

Return

Demographic Indicators

Study - Main

Library

Data: Category name is this
and that

Country	Data
United Kingdom	52000
United States of America	80000
Antigua and Barbuda	25000
Germany	53000
China	16000
India	2800
Australia	64000
Colombia	8900
Benin	1700
	7800

Return

(Created by author)

FAQ section:

FAQ

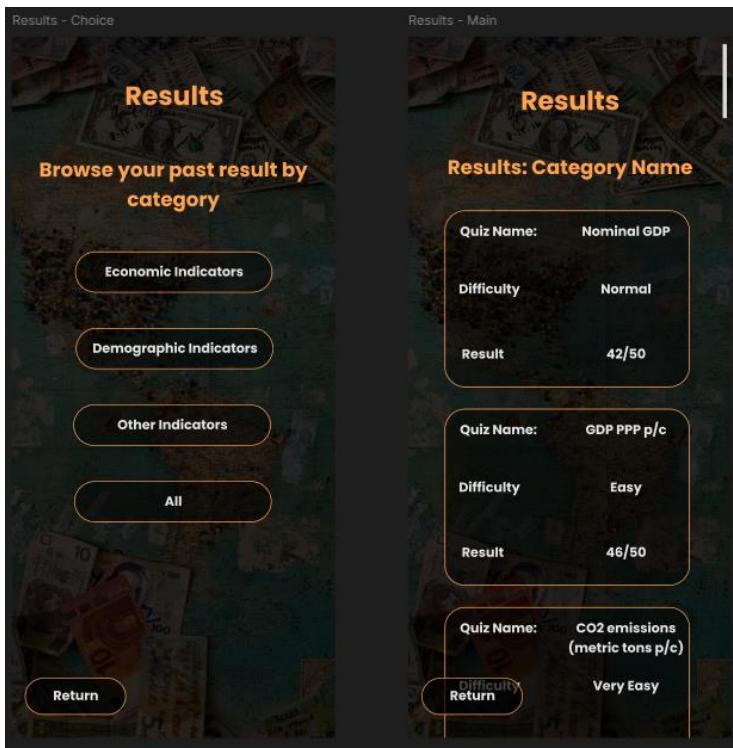
FAQ

Q: How do I know that the provided data is accurate?
A: That's the neat part. You don't! Just kidding! :) I get this data from the World Bank database, so it is as accurate as they are. Also, keep in mind that it takes some time to calculate and update the data, so for the first couple of months of any given year, the data is probably from the year before previous. This is not something I can impact.

Q: Why did you put a quiz about X, but not Y?
A: Currently, this website is a simple project maintained by [Return](#), so the amount of content is limited accordingly.

(Created by author)

Results section:



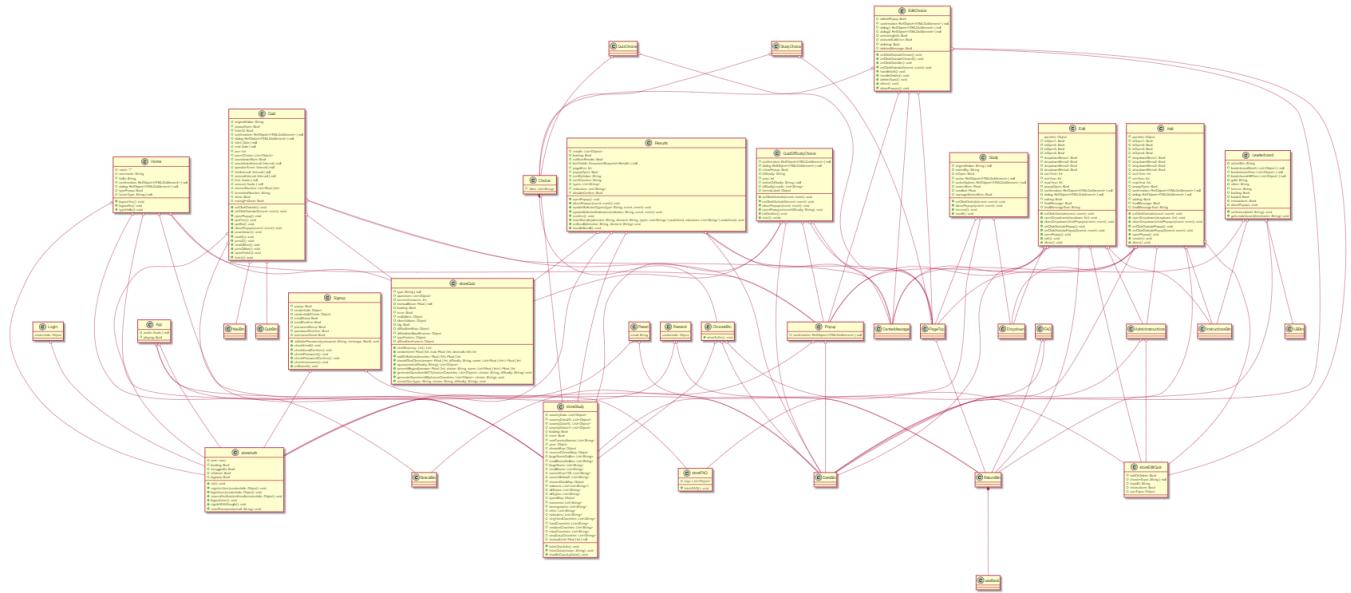
(Created by author)

The brand color of the website is #ffa550. This is because the orange color is associated with playfulness, happiness, having fun, and being simply joyful. Sporcle uses a similar color on their website as well. Other elements of the website have also been created so as to match the fun vibe one would expect from a quiz app. The buttons are rounded, there are fun images and icons, as well as animations, and even some music. The font used is Poppins, since it also matches the vibe. The language is mostly simple, even though the quiz topics and contents are not easy. Econ Quiz aims to satisfy the needs of an intelligent audience that is interested in economics, but it wants to do so in a fun manner. Other colors that are frequently used in the app include “white gray” (#e6e6e6) and “black gray” (#1a1a1a). One actual grey color (#808080) is also used. Three variations of black are used for different purposes. Button background often equals rgba(0, 0, 0, 0.45), form background equals rgba(0, 0, 0, 0.7), and popup background equals rgba(0, 0, 0, 0.95). Backup fonts are Arial and sans-serif. Buttons scale and change color when clicked on, in a way that one would most likely expect. Twenty-five websites served as an inspiration to come

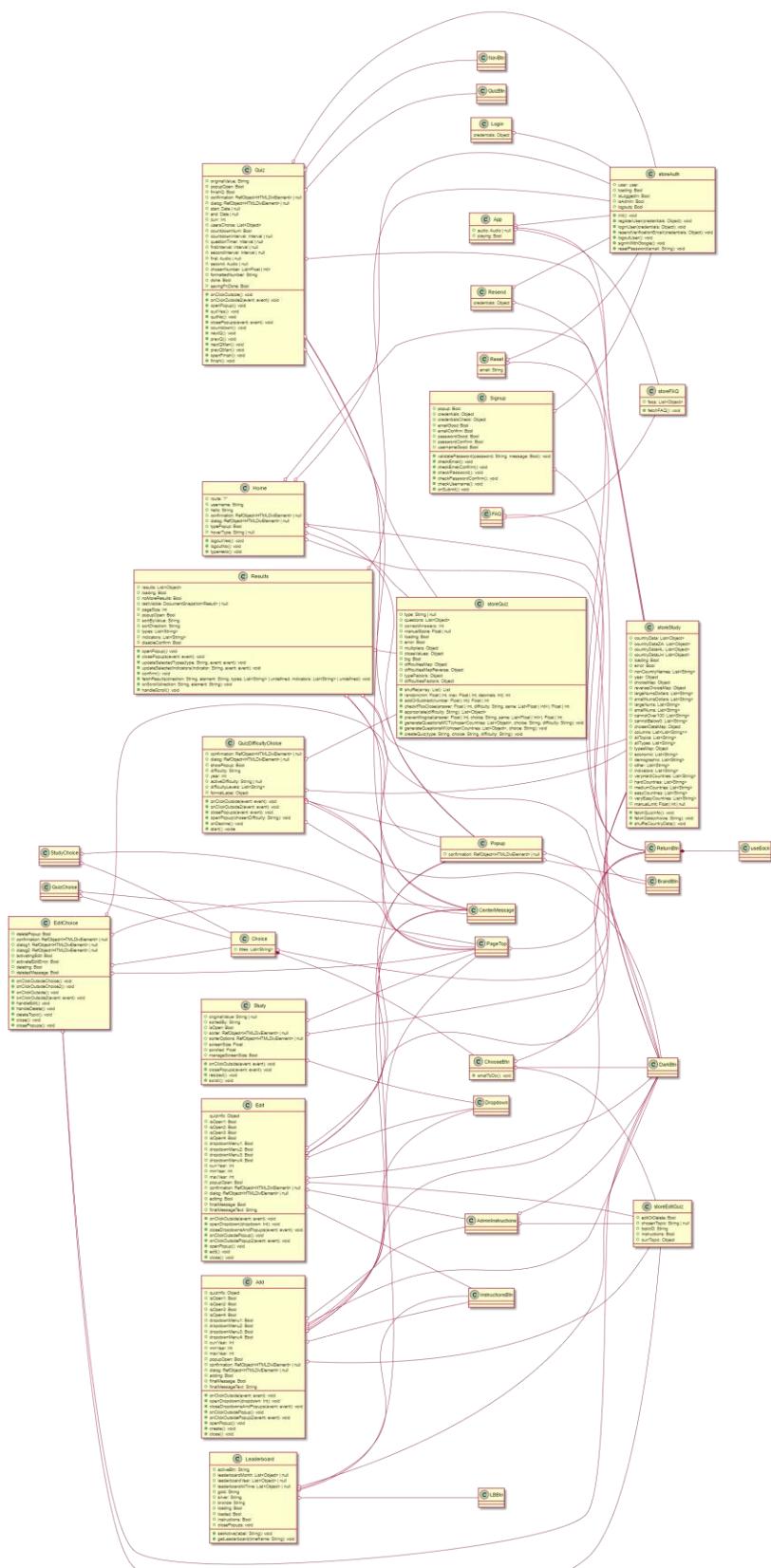
up with design ideas, but the ultimate outcome is a somewhat unique but also familiar-looking fun quiz website for intelligent economics nerds.

4. IMPLEMENTATION

The app consists of 16 views (excluding App.vue), 14 components, one composable and 5 pinia stores. Assets include one gif, three icons, 25 images and three sounds.



(Created by author)



(Created by author)

The class UML diagram above shows how the views, components, composable, and pinia stores relate to each other. The diagram is shown both horizontally and vertically. Classes also include the variables and functions declared globally in the views, composable and stores they represent, except for the variables used to reference the Vue router, or pinia stores. Hooks are also not included. Indeed, the diagram is a bit difficult to read. It can be recreated using the PlantUML extension in VSCode or VSCodium. The extension requires Java and GraphViz to be installed as well. The code used to create this diagram:

```
@startuml
` left to right direction

` views
class App {
    +audio: Audio | null
    +playing: Bool
}

class Home {
    +route: "/"
    +logoutYes(): void
    +logoutNo(): void
    +username: String
    +hello: String
    +typeHello(): void
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog: RefObject<HTMLDivElement> | null
    +typePopup: Bool
    +hoverType: String | null
}

class Results {
    +results: List<Object>
    +loading: Bool
    +noMoreResults: Bool
    +lastVisible: DocumentSnapshot<Result> | null
    +pageSize: Int
```

```

+popupOpen: Bool
+openPopup(): void
+closePopups(event: event): void
+sortByValue: String
+sortDirection: String
+types: List<String>
+indicators: List<String>
+updateSelectedTypes(type: String, event: event): void
+updateSelectedIndicators(indicator: String, event: event): void
+disableConfirm: Bool
+confirm(): void
+fetchResults(direction: String, element: String, types: List<String> | undefined,
indicators: List<String> | undefined): void
+onScroll(direction: String, element: String): void
+handleScroll(): void
}

class Leaderboard {
    +activeBtn: String
    +setActive(label: String): void
    +leaderboardMonth: List<Object> | null
    +leaderboardYear: List<Object> | null
    +leaderboardAllTime: List<Object> | null
    +gold: String
    +silver: String
    +bronze: String
    +loading: Bool
    +loaded: Bool
    +getLeaderboard(timeframe: String): void
    +instructions: Bool
    +closePopups: void
}

class StudyChoice

class Study {
    +originalValue: String | null
    +sortedBy: String
    +isOpen: Bool
    +sorter: RefObject<HTMLDivElement> | null
    +sorterOptions: RefObject<HTMLDivElement> | null
    +onClickOutside(event: event): void
    +closePopups(event: event): void
    +screenSize: Float
    +scrolled: Float
}

```

```
+resized(): void
+scroll(): void
+manageScreenSize: Bool
}

class FAQ

class Signup {
    +popup: Bool
    +validatePassword(password: String, message: Bool): void
    +credentials: Object
    +credentialsCheck: Object
    +emailGood: Bool
    +checkEmail(): void
    +emailConfirm: Bool
    +checkEmailConfirm(): void
    +passwordGood: Bool
    +checkPassword(): void
    +passwordConfirm: Bool
    +checkPasswordConfirm(): void
    +usernameGood: Bool
    +checkUsername(): void
    +onSubmit(): void
}

class Login {
    credentials: Object
}

class Resend {
    credentials: Object
}

class Reset {
    email: String
}

class QuizChoice

class QuizDifficultyChoice {
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog: RefObject<HTMLDivElement> | null
    +onClickOutside(event: event): void
    +onClickOutside2(event: event): void
    +closePopups(event: event): void
```

```
+showPopup: Bool
+difficulty: String
+year: Int
+openPopup(chosenDifficulty: String): void
+onDecline(): void
+start(): void
+activeDifficulty: String | null
+difficultyLevels: List<String>
+formatLabel: Object
}

class Quiz {
    +originalValue: String
    +popupOpen: Bool
    +finishQ: Bool
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog: RefObject<HTMLDivElement> | null
    +onClickOutside(): void
    +onClickOutside2(event: event): void
    +openPopup(): void
    +quitYes(): void
    +quitNo(): void
    +closePopups(event: event): void
    +start: Date | null
    +end: Date | null
    +curr: Int
    +usersChoice: List<Object>
    +countdownNum: Bool
    +countdownInterval: Interval | null
    +countdown(): void
    +questionTimer: Interval | null
    +firstInterval: Interval | null
    +secondInterval: Interval | null
    +first: Audio | null
    +second: Audio | null
    +nextQ(): void
    +prevQ(): void
    +chosenNumber: List<Float | Int>
    +formattedNumber: String
    +nextQMan(): void
    +prevQMan(): void
    +done: Bool
    +openFinish(): void
    +savingFnDone: Bool
    +finish(): void
}
```

```
}

class Add {
    +quizInfo: Object
    +isOpen1: Bool
    +isOpen2: Bool
    +isOpen3: Bool
    +isOpen4: Bool
    +dropdownMenu1: Bool
    +dropdownMenu2: Bool
    +dropdownMenu3: Bool
    +dropdownMenu4: Bool
    +onClickOutside(event: event): void
    +openDropdown(dropdown: Int): void
    +closeDropdownsAndPopups(event: event): void
    +currYear: Int
    +minYear: Int
    +maxYear: Int
    +popupOpen: Bool
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog: RefObject<HTMLDivElement> | null
    +onClickOutsidePopup(): void
    +onClickOutsidePopup2(event: event): void
    +openPopup(): void
    +adding: Bool
    +create(): void
    +close(): void
    +finalMessage: Bool
    +finalMessageText: String
}

class EditChoice {
    +deletePopup: Bool
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog1: RefObject<HTMLDivElement> | null
    +dialog2: RefObject<HTMLDivElement> | null
    +onClickOutsideChoice(): void
    +onClickOutsideChoice2(): void
    +onClickOutside(): void
    +onClickOutside2(event: event): void
    +activatingEdit: Bool
    +activateEditError: Bool
    +handleEdit(): void
    +handleDelete(): void
    +deleting: Bool
}
```

```
+deletedMessage: Bool
+deleteTopic(): void
+close(): void
+closePopups(): void
}

class Edit {
    quizInfo: Object
    +isOpen1: Bool
    +isOpen2: Bool
    +isOpen3: Bool
    +isOpen4: Bool
    +dropdownMenu1: Bool
    +dropdownMenu2: Bool
    +dropdownMenu3: Bool
    +dropdownMenu4: Bool
    +onClickOutside(event: event): void
    +openDropdown(dropdown: Int): void
    +closeDropdownsAndPopups(event: event): void
    +currYear: Int
    +minYear: Int
    +maxYear: Int
    +popupOpen: Bool
    +confirmation: RefObject<HTMLDivElement> | null
    +dialog: RefObject<HTMLDivElement> | null
    +onClickOutsidePopup(): void
    +onClickOutsidePopup2(event: event): void
    +openPopup(): void
    +editing: Bool
    +edit(): void
    +close(): void
    +finalMessage: Bool
    +finalMessageText: String
}

' stores
class storeAuth {
    +user: user
    +loading: Bool
    +isLoggedIn: Bool
    +isAdmin: Bool
    +logoutq: Bool
    +init(): void
    +registerUser(credentials: Object): void
    +loginUser(credentials: Object): void
```

```
+resendVerificationEmail(credentials: Object): void
+logoutUser(): void
+signInWithGoogle(): void
+resetPassword(email: String): void
}

class storeEditQuiz {
    +editOrDelete: Bool
    +chosenTopic: String | null
    +topicID: String
    +instructions: Bool
    +currTopic: Object
}

class storeFAQ {
    +faqs: List<Object>
    +fetchFAQ(): void
}

class storeQuiz {
    +type: String | null
    +questions: List<Object>
    +correctAnswers: Int
    +manualScore: Float | null
    +loading: Bool
    +error: Bool
    +multipliers: Object
    +closeValues: Object
    +big: Bool
    +difficultiesMap: Object
    +difficultiesMapReverse: Object
    +typeFactors: Object
    +difficultiesFactors: Object
    +shuffle(array: List): List
    +random(min: Float | Int, max: Float | Int, decimals: Int): Int
    +addOrSubtract(number: Float | Int): Float | Int
    +checkIfTooClose(answer: Float | Int, difficulty: String, same: List<Float | Int>): Float | Int
    +appropriate(difficulty: String): List<Object>
    +preventIllegal(answer: Float | Int, choice: String, same: List<Float | Int>): Float | Int
    +generateQuestionsMCT(chosenCountries: List<Object>, choice: String, difficulty: String): void
    +generateQuestionsMI(chosenCountries: List<Object>, choice: String): void
    +createQuiz(type: String, choice: String, difficulty: String): void
```

```
}

class storeStudy {
    +countryData: List<Object>
    +countryDataZA: List<Object>
    +countryDataHL: List<Object>
    +countryDataLH: List<Object>
    +loading: Bool
    +error: Bool
    +nonCountryNames: List<String>
    +year: Object
    +choiceMap: Object
    +reverseChoiceMap: Object
    +largeNumsDollars: List<String>
    +smallINumsDollars: List<String>
    +largeNums: List<String>
    +smallINums: List<String>
    +cannotOver100: List<String>
    +cannotBelow0: List<String>
    +chosenDataMap: Object
    +columns: List<List<String>>
    +allTopics: List<String>
    +allTypes: List<String>
    +typesMap: Object
    +economic: List<String>
    +demographic: List<String>
    +other: List<String>
    +indicators: List<String>
    +veryHardCountries: List<String>
    +hardCountries: List<String>
    +mediumCountries: List<String>
    +easyCountries: List<String>
    +veryEasyCountries: List<String>
    +manualLimit: Float | Int | null
    +fetchQuizInfo(): void
    +fetchData(choice: String): void
    +shuffleCountryData(): void
}

' composable
class useBack

' components
class AdminInstructions
```

```
class BrandBtn

class CenterMessage

class Choice {
    +titles: List<String>
}

class ChooseBtn {
    +whatToDo(): void
}

class DarkBtn

class Dropdown

class InstructionsBtn

class LBBtn

class NavBtn

class PageTop

class Popup {
    +confirmation: RefObject<HTMLDivElement> | null
}

class QuizBtn

class ReturnBtn

' relations
App o-- storeAuth
App o-- storeStudy
App o-- storeFAQ

Add o-- ReturnBtn
Add o-- Dropdown
Add o-- DarkBtn
Add o-- Popup
Add o-- CenterMessage
Add o-- AdminInstructions
Add o-- InstructionsBtn
Add o-- storeEditQuiz
```

```
Edit o-- ReturnBtn
Edit o-- Dropdown
Edit o-- DarkBtn
Edit o-- Popup
Edit o-- CenterMessage
Edit o-- AdminInstructions
Edit o-- InstructionsBtn
Edit o-- storeEditQuiz

EditChoice o-- PageTop
EditChoice o-- Choice
EditChoice o-- DarkBtn
EditChoice o-- Popup
EditChoice o-- CenterMessage
EditChoice o-- storeEditQuiz

FAQ o-- storeFAQ
FAQ o-- ReturnBtn

Home o-- BrandBtn
Home o-- DarkBtn
Home o-- Popup
Home o-- storeAuth
Home o-- storeQuiz
Home o-- storeStudy

Leaderboard o-- PageTop
Leaderboard o-- LBBtn
Leaderboard o-- InstructionsBtn
Leaderboard o-- DarkBtn

Login o-- storeAuth

Quiz o-- storeQuiz
Quiz o-- QuizBtn
Quiz o-- NavBtn
Quiz o-- Popup
Quiz o-- storeStudy
Quiz o-- CenterMessage
Quiz o-- storeAuth

QuizChoice o-- Choice
QuizChoice o-- PageTop
```

```
QuizDifficultyChoice o-- PageTop
QuizDifficultyChoice o-- storeStudy
QuizDifficultyChoice o-- DarkBtn
QuizDifficultyChoice o-- Popup
QuizDifficultyChoice o-- storeQuiz
QuizDifficultyChoice o-- CenterMessage

Resend o-- ReturnBtn
Resend o-- storeAuth

Reset o-- ReturnBtn
Reset o-- storeAuth

Results o-- storeAuth
Results o-- CenterMessage
Results o-- PageTop
Results o-- DarkBtn
Results o-- storeQuiz
Results o-- storeStudy

Signup o-- DarkBtn
Signup o-- storeAuth

Study o-- PageTop
Study o-- storeStudy
Study o-- Dropdown

StudyChoice o-- Choice
StudyChoice o-- PageTop

AdminInstructions o-- storeEditQuiz
AdminInstructions o-- DarkBtn

Choice *-- ChooseBtn
Choice o-- storeStudy

ChooseBtn o-- DarkBtn
ChooseBtn o-- storeStudy
ChooseBtn o-- storeEditQuiz

PageTop o-- ReturnBtn

Popup o-- DarkBtn
Popup o-- BrandBtn
```

```
ReturnBtn *-- useBack  
@enduml
```

The relations are pretty clear from these lines of ‘code’, as well as which functions and variables can be found in each class, what type they are, and what parameters the functions expect, and of what type. All of the views, components, composable, and pinia stores will be listed below.

The list of all views: App, Add, Edit, EditChoice, FAQ, Home, Leaderboard, Login, Quiz, QuizChoice, QuizDifficultyChoice, Resend, Reset, Results, Signup, Study, StudyChoice.

The list of all components: AdminInstructions, BrandBtn, CenterMessage, Choice, ChooseBtn, DarkBtn, Dropdown, InstructionsBtn, LBBtn, NavBtn, PageTop, Popup, QuizBtn, ReturnBtn.

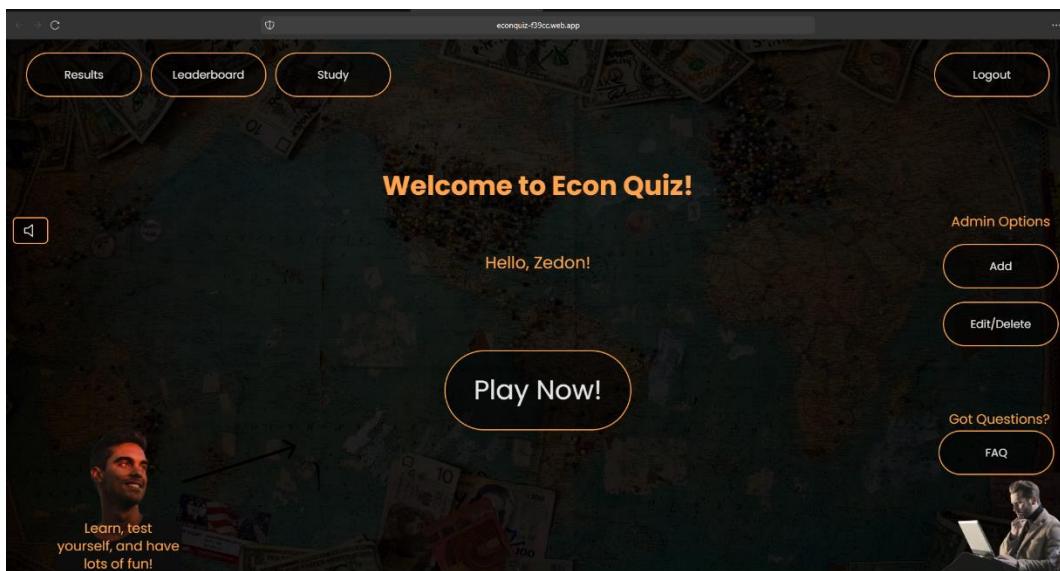
The composable: useBack.

The list of all pinia stores: storeAuth, storeEditQuiz, storeFAQ, storeQuiz, storeStudy.

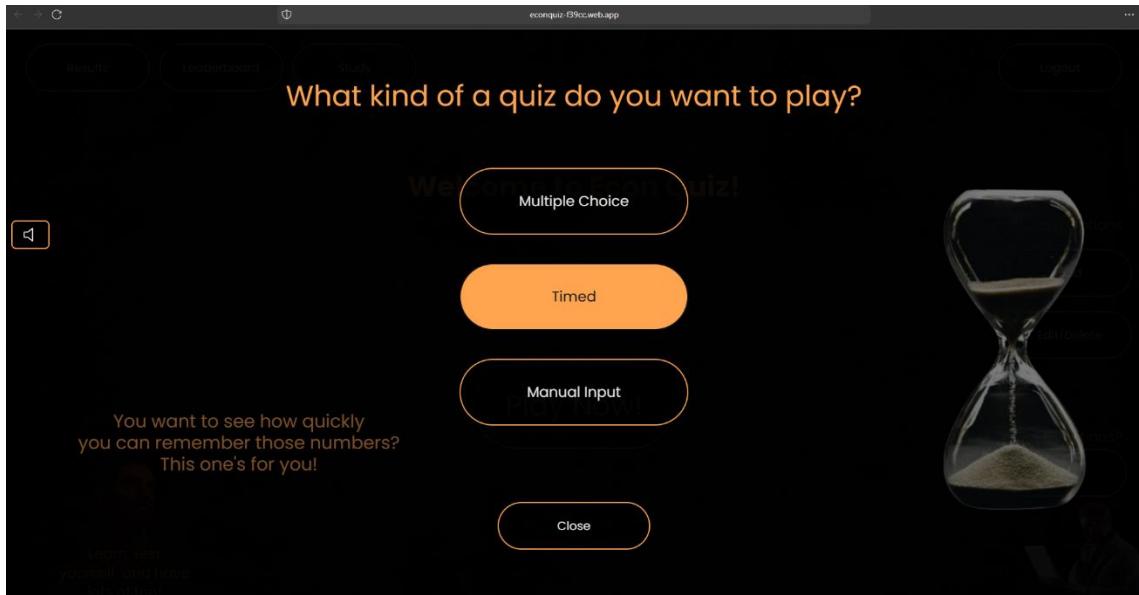
5. USER INSTRUCTIONS

This chapter will show screenshots from the app to portray what the user sees while using it.

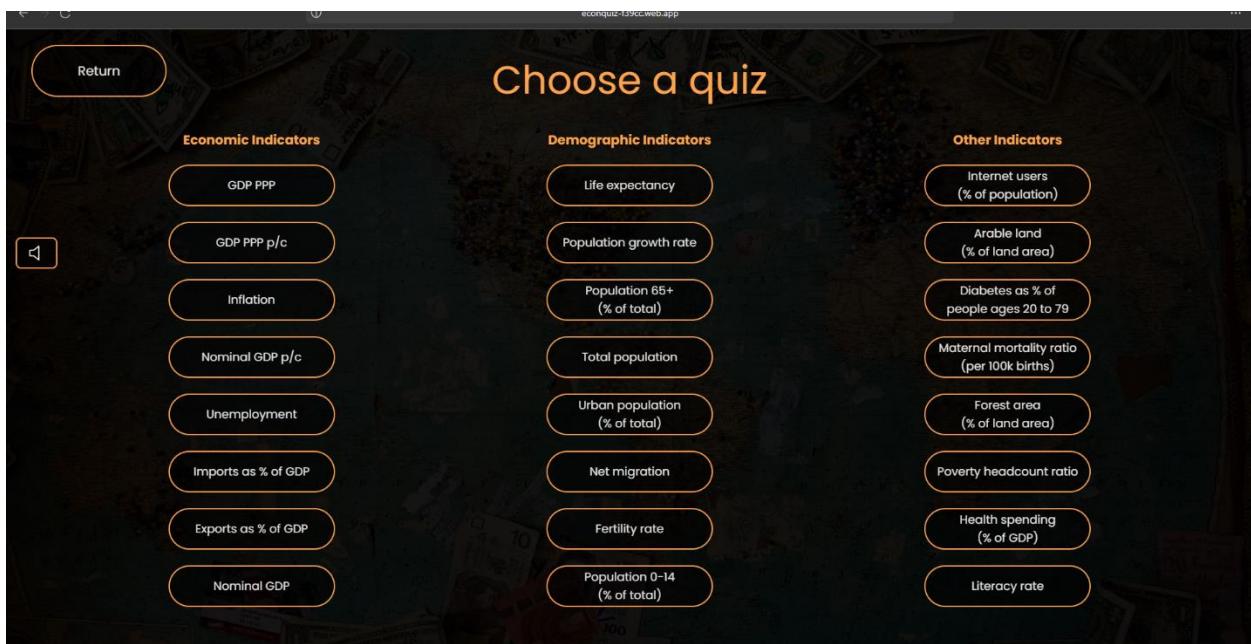
If the user decides to play a timed quiz about Nominal GDP p/c on hard difficulty, their experience on a desktop computer will go something like this.



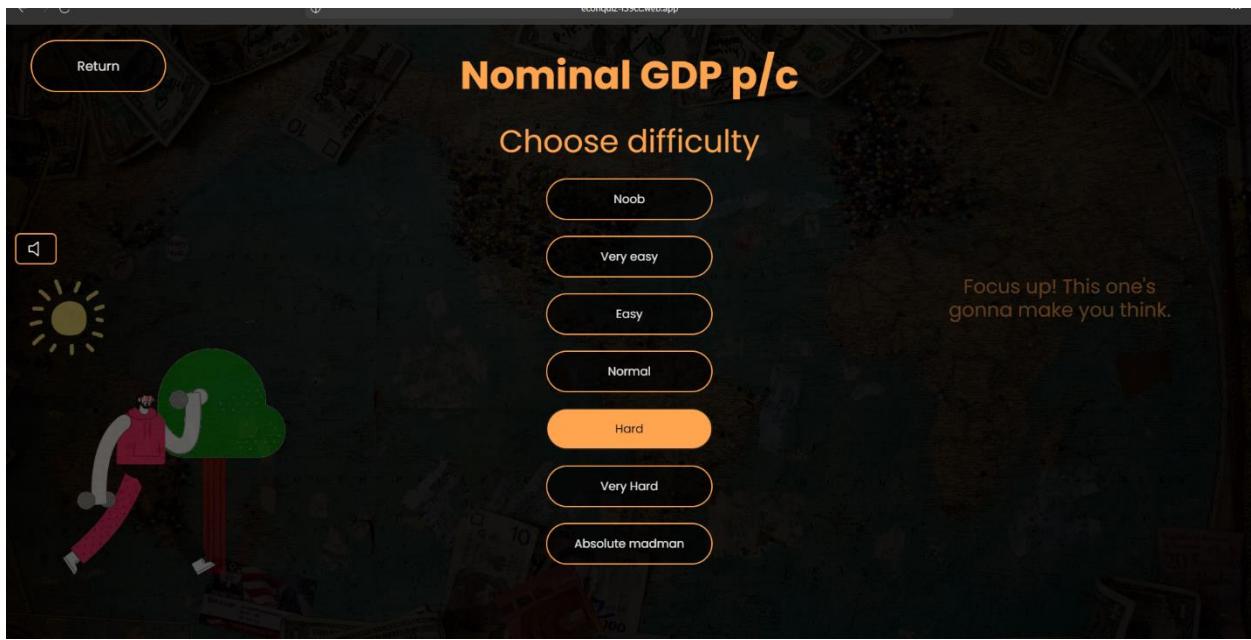
(Created by author)



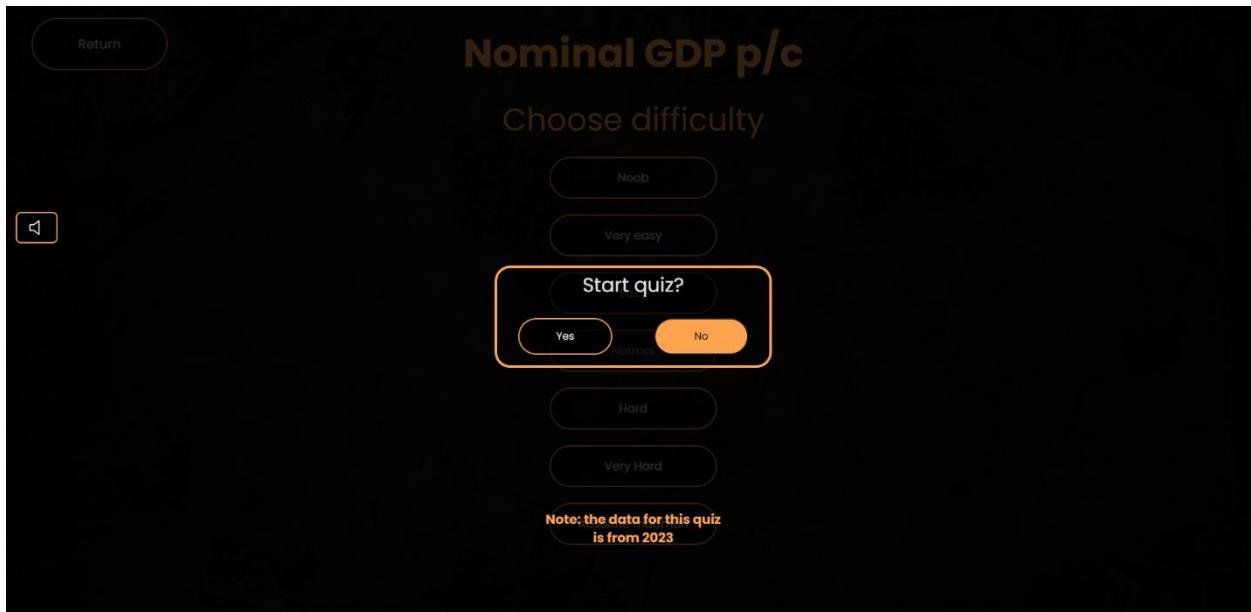
(Created by author)



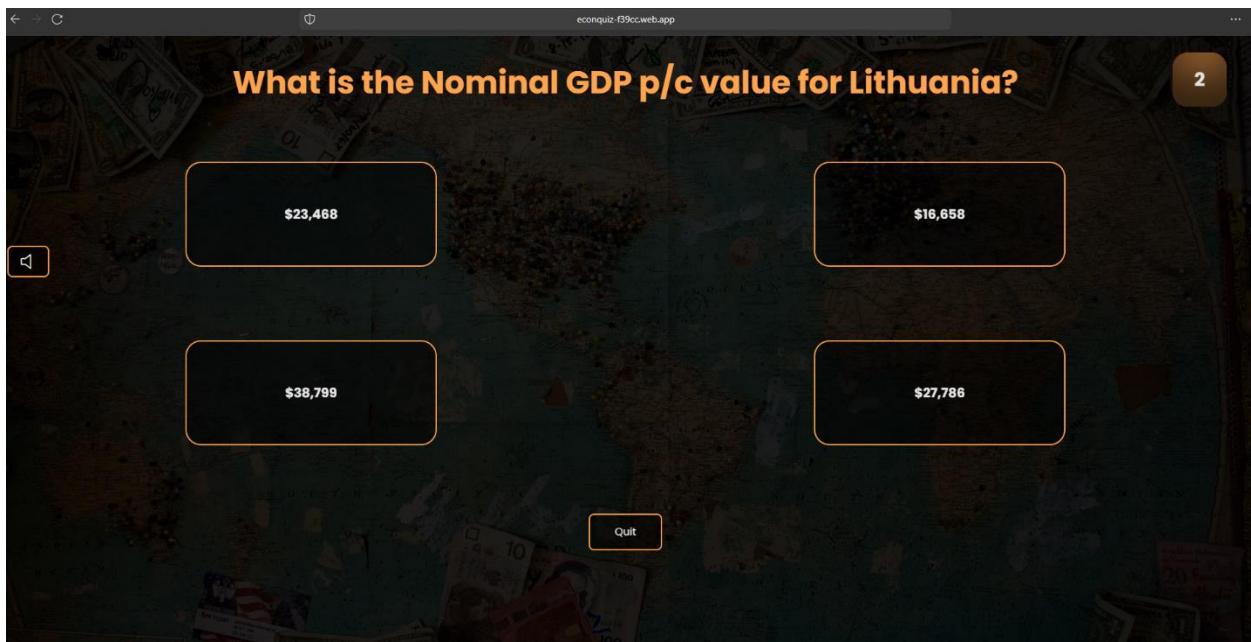
(Created by author)



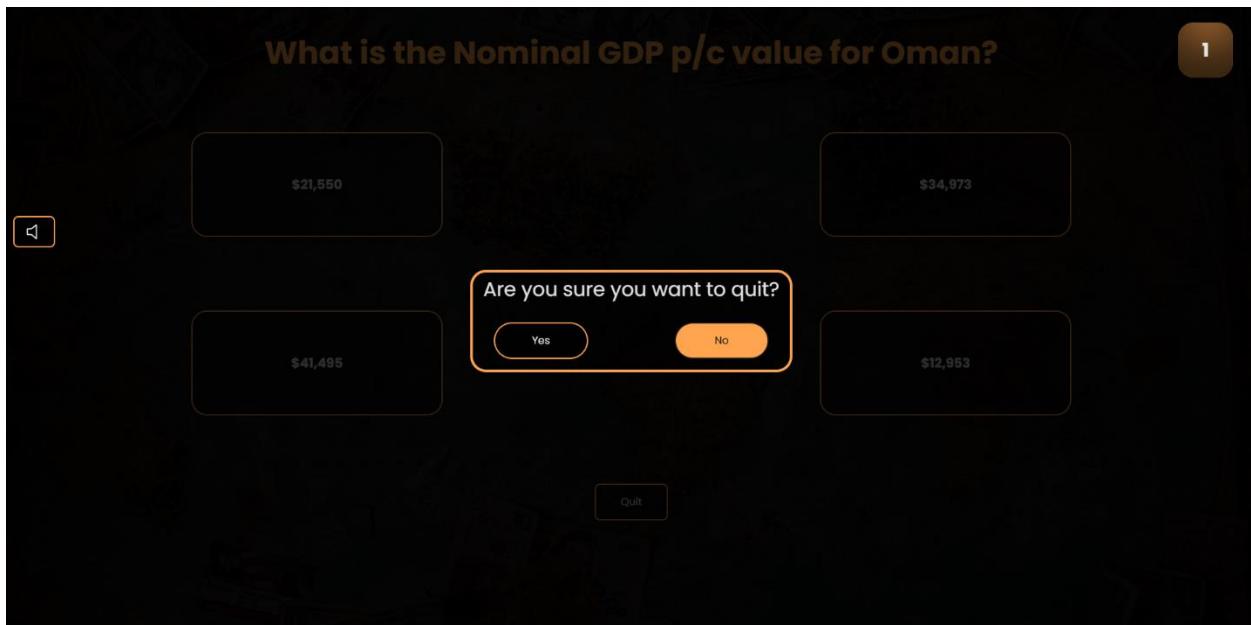
(Created by author)



(Created by author)

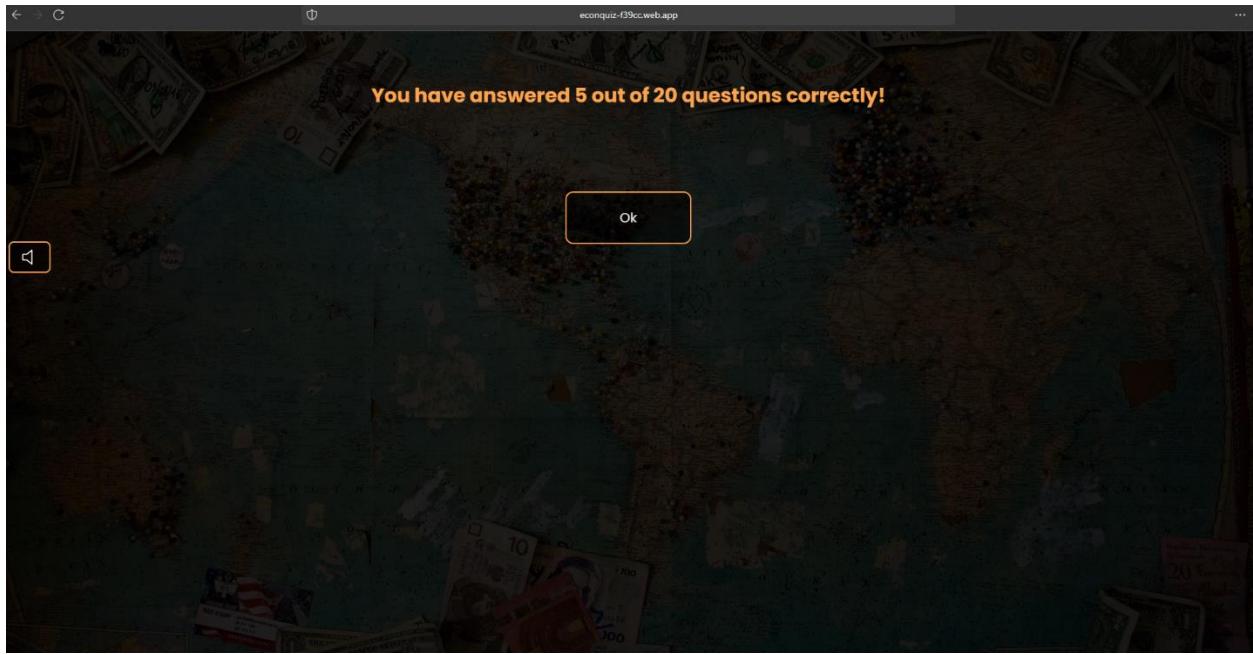


(Created by author)



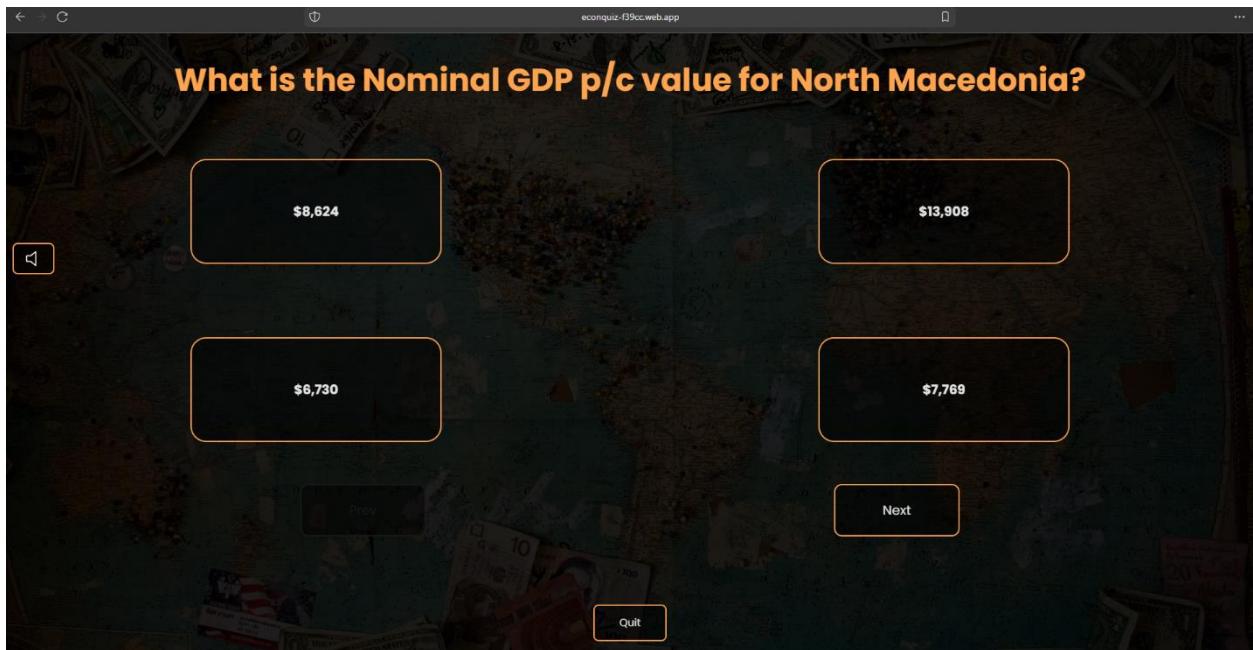
(Created by author)

Assuming the user does not quit...

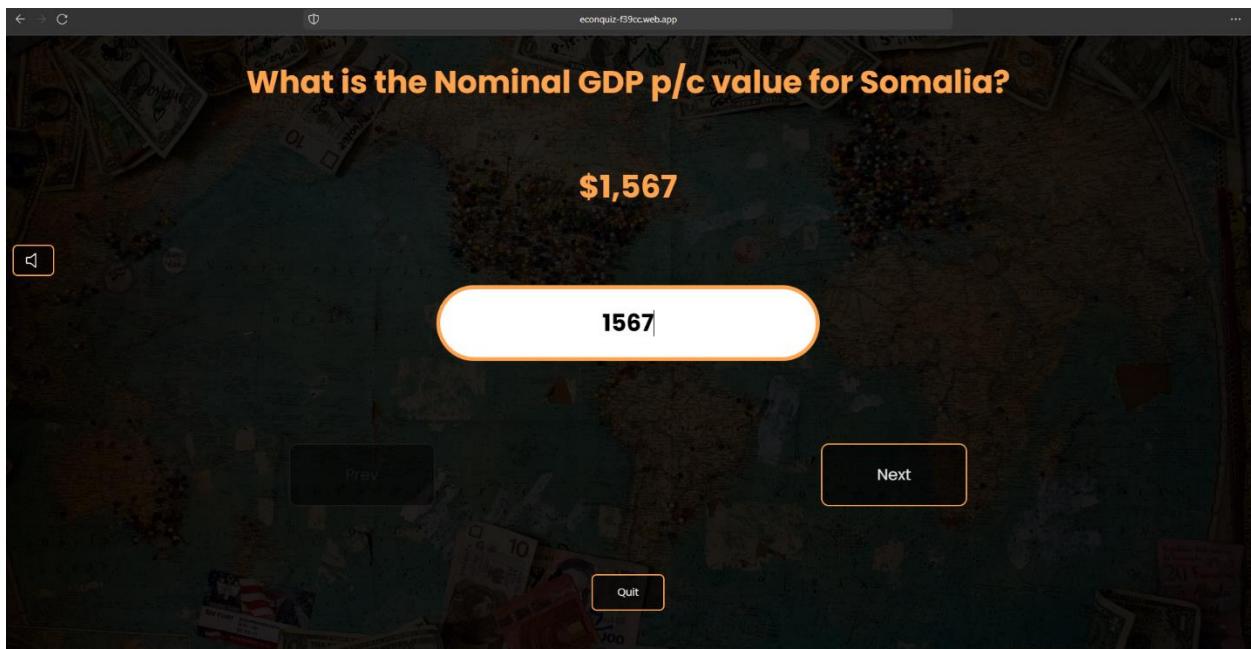


(Created by author)

The UI for multiple choice and manual input quizzes is slightly different:



(Created by author)



(Created by author)

FAQ section appears like this:

A screenshot of the FAQ section of the quiz app. The background is the same dark money collage. The word "FAQ" is centered at the top in yellow. On the left, there is a "Return" button. On the right, there is a small portrait of a man in a suit. The FAQ section contains several questions and answers:

- Q: Why is there no "Contact us"? I have something to say!**
A: Because I tried creating a new email and Google didn't let me, lol. I don't want to leave my personal email here. I apologize for any inconvenience that may have caused.
- Q: Can I add some topics?**
A: Only admins can add, edit and delete topics. This is done to prevent abuse of those options.
- Q: Why is my leadership score different from my score?**
A: Your score represents what % of the questions you got right in that particular quiz. In the case of manual input quizzes it shows how close you were to the right numbers. Leadership score takes that score and "rewards" you with more points for harder quizzes. The exact calculation for this can be found in the leadership section by clicking on the question mark in the top right corner.
- Q: How are the quizzes created?**
A: The quizzes are created using an algorithm that adjusts the difficulty according to which one of the seven difficulties you choose. Apologies if some difficulties seem too easy or too hard relative to the other ones. This is a one man project and it is kinda hard to create perfectly balanced quizzes. They should be balanced enough though.
- Q: How do I know that the provided data is accurate?**
A: That's the neat part. You don't! Just kidding! :) I get this data from the World Bank database, so it is as accurate as they are. Also, keep in mind that it takes them some time to calculate and update the data, so for the first couple of months of any given year, the data is probably from the year before previous. This is not something I can impact.
- Q: Why did you put a quiz about X, but not Y?**
A: The most popular/relevant topics from the World Bank database are included already (e.g. GDP Nominal, Total).

(Created by author)

Viewing past results will go like this:

The screenshot shows a dark-themed interface titled "Results". At the top right is a "Sort and Filter" button. On the left is a "Return" button and a speaker icon. The main area displays two quiz results in cards:

Type:	Timed
Topic:	Nominal GDP p/c
Difficulty:	Hard
Score:	25%
Time spent:	1m 40s
Leaderboard score:	71
Taken on:	10.3.2025, 15:20

Type:	Timed
Topic:	Life expectancy
Difficulty:	Very hard
Score:	50%
Time spent:	1m 40s
Leaderboard score:	105

(Created by author)

The screenshot shows a dark-themed interface titled "Results". At the top right is a "Sort and Filter" button. On the left is a "Return" button and a speaker icon. The main area is divided into sections:

Sort by

- Score
- Difficulty
- Time spent
- Taken on
- Leaderboard score

Order

- Descending
- Ascending

Filter

Type

- Multiple Choice
- Timed
- Manual Input

Indicator

- Economic
- Demographic
- Other

Confirm choices

(Created by author)

Viewing the leaderboard:

The screenshot shows a dark-themed leaderboard interface. At the top center is the word "Leaderboard". Below it are three buttons: "Past Month", "Past Year", and "All Time", with "All Time" being highlighted in orange. On the left side, there are navigation buttons: a back arrow and a question mark icon. The main area displays three user entries: "Zedon:" with a score of "271", "Sebastijan:" with a score of "197", and "Mirusasa:" with a score of "125". The background features a faint image of various banknotes.

(Created by author)

The screenshot shows an informational page titled "How are scores calculated?". It contains several paragraphs of text explaining the scoring system. At the top right is a button labeled "All Time". The text includes: "Scores take into account the type of quiz, difficulty and score of that particular quiz. That score is multiplied with two factors that reward good results on harder quizzes.", "Timed and Manual Input quizzes give you 1.3x score. This represents the first factor.", "The second factor comes from difficulty. Very easy difficulty gives you 1.3x score, easy gives 1.6x, normal gives 1.9x, hard gives 2.2x, very hard gives 2.5x, and absolute madman gives 2.8x.", "The highest possible score is $100 \times 1.3 \times 2.8 = 364$. The lowest possible score is 0.", "Life expectancy scores are "nerfed" because they are relatively easier than the rest, so they are multiplied by 0.65.", and "Only one score per user can appear on the leaderboard, so your highest score for the chosen period (month/year/all-time) is displayed.".

(Created by author)

Study section:

The screenshot shows a dark-themed web application interface. At the top center, the title "Choose a Topic to Study" is displayed in a large, bold, orange font. Below the title, there are three main sections arranged horizontally: "Economic Indicators", "Demographic Indicators", and "Other Indicators".

- Economic Indicators:** Contains nine items: GDP PPP, GDP PPP p/c, Inflation, Nominal GDP p/c, Unemployment, Imports as % of GDP, Exports as % of GDP, and Nominal GDP.
- Demographic Indicators:** Contains nine items: Life expectancy, Population growth rate, Population 65+ (% of total), Total population, Urban population (% of total), Net migration, Fertility rate, and Population 0-14 (% of total).
- Other Indicators:** Contains nine items: Internet users (% of population), Arable land (% of land area), Diabetes as % of people ages 20 to 79, Maternal mortality ratio (per 100k births), Forest area (% of land area), Poverty headcount ratio, Health spending (% of GDP), and Literacy rate.

A "Return" button is located in the top left corner, and a volume icon is in the top right corner. The background features a faint image of money bills.

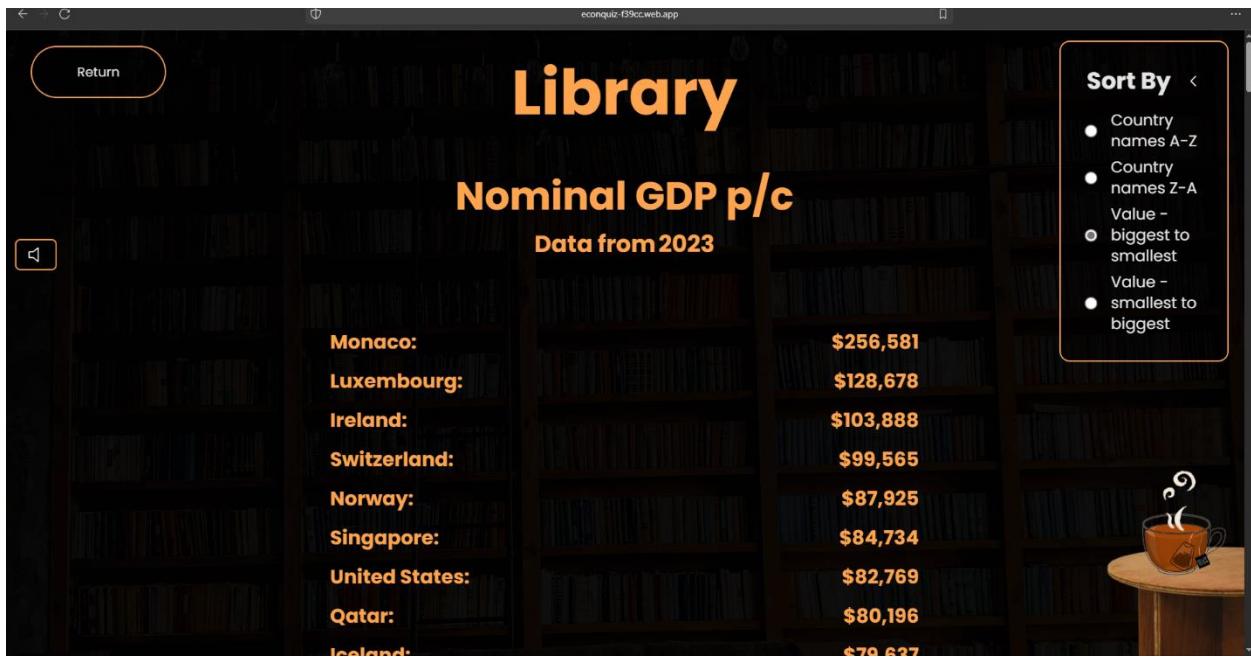
(Created by author)

The screenshot shows a dark-themed web application interface. At the top center, the title "Library" is displayed in a large, bold, orange font. To the right of the title, there is a "Sort By" dropdown menu. Below the title, the subtitle "Nominal GDP p/c" is centered, followed by "Data from 2023".

A volume icon is in the top left corner. On the right side, there is a small illustration of a teacup on a saucer with a question mark above it.

Country	Nominal GDP p/c (2023)
Afghanistan	\$416
Albania	\$8,575
Algeria	\$5,364
Andorra	\$46,812
Angola	\$2,308
Antigua and Barbuda	\$21,787
Argentina	\$14,187
Armenia	\$8,053
Australia	\$64,821

(Created by author)



(Created by author)

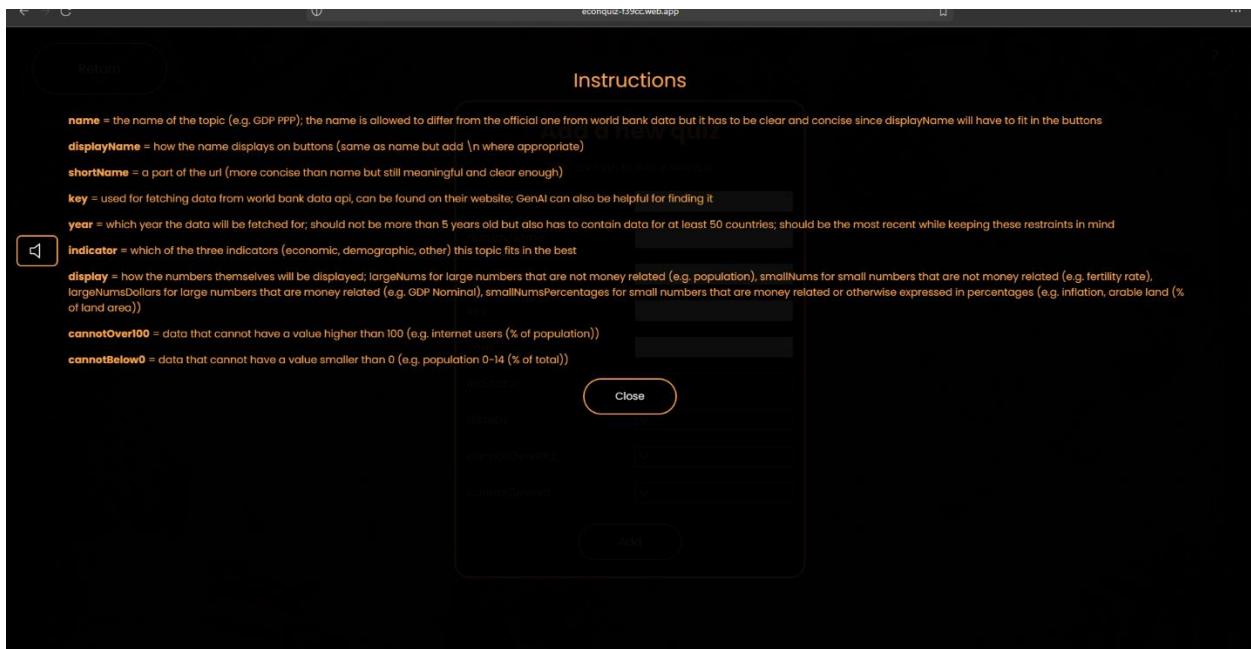
Adding a new topic:

The screenshot shows a mobile application interface with a background of US dollar bills. In the center, there is a modal dialog titled "Add a new quiz". The dialog contains a sub-instruction "Fill in the form to add a new quiz" and several input fields:

Field	Type
name:	Text input
displayName:	Text input
shortName:	Text input
key:	Text input
year:	Text input
indicator:	Dropdown
display:	Dropdown
cannotOver00:	Dropdown
cannotBelow0:	Dropdown

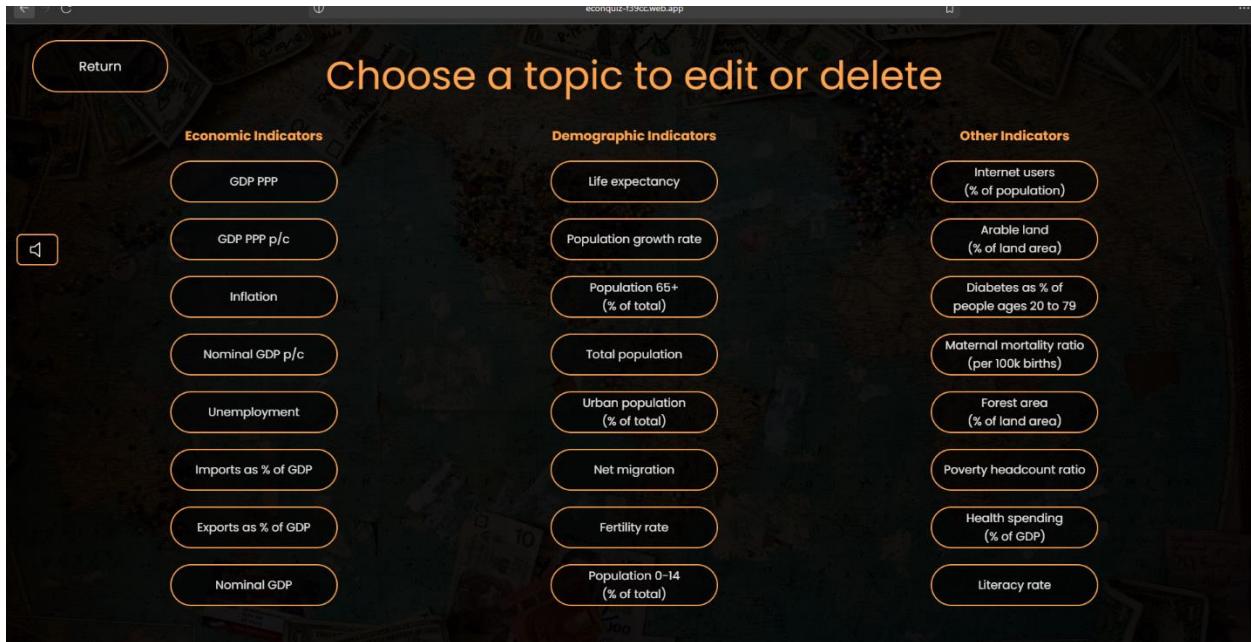
At the bottom of the dialog is an "Add" button.

(Created by author)

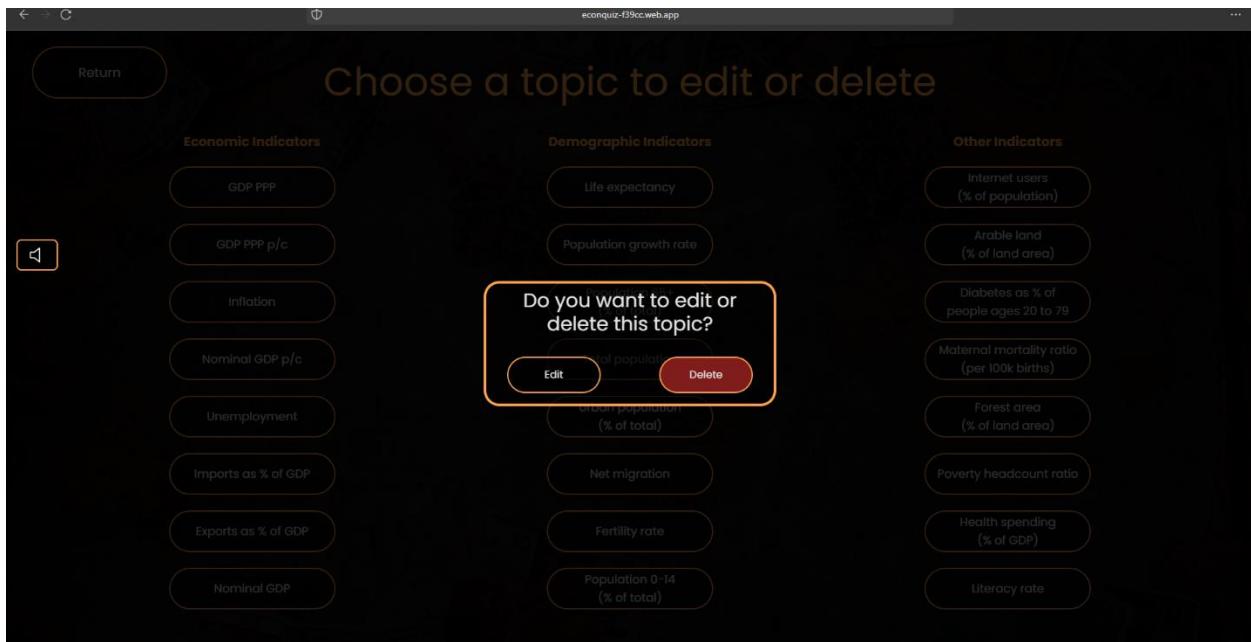


(Created by author)

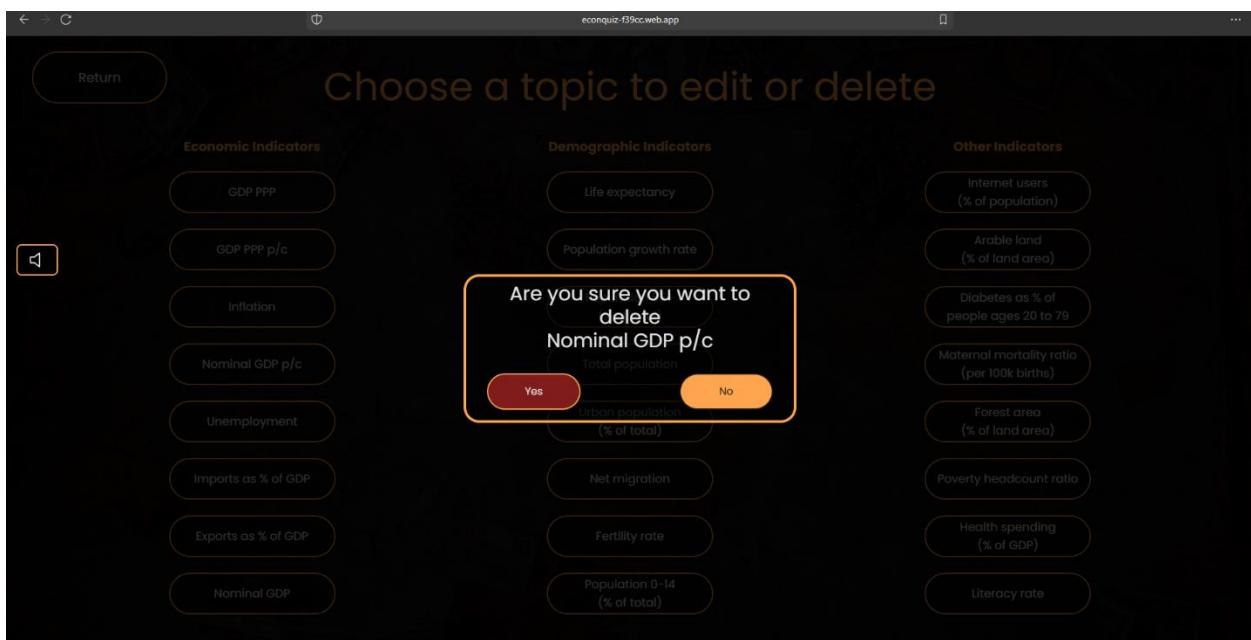
Editing and deleting:



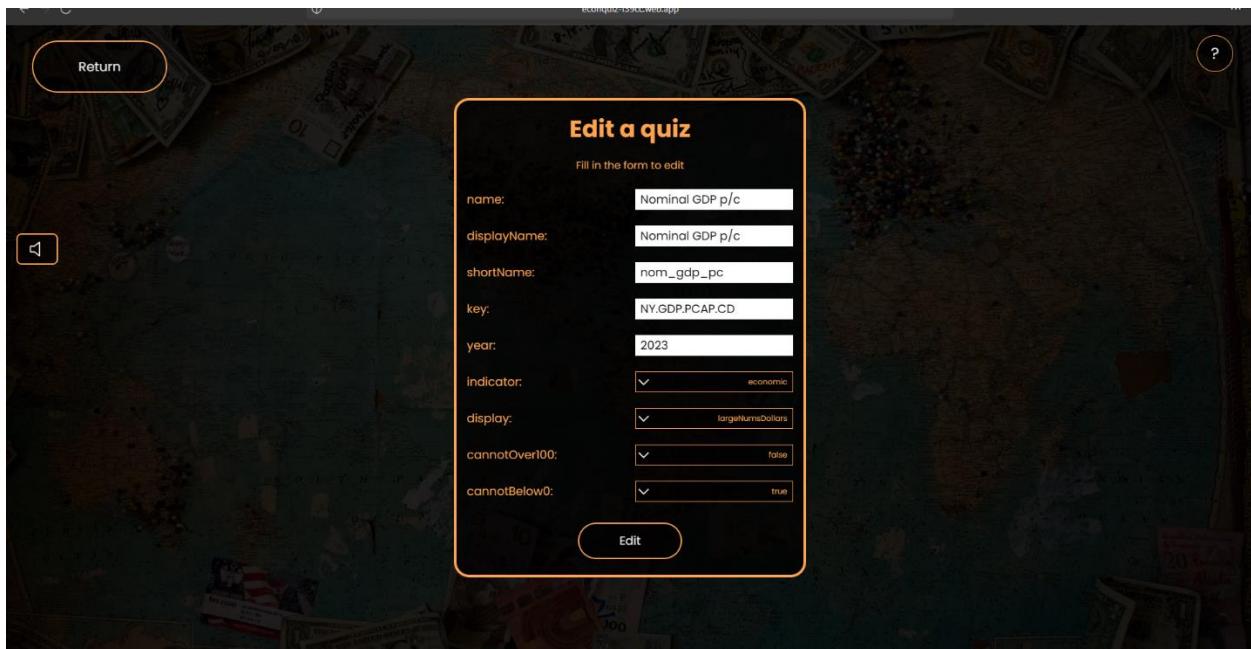
(Created by author)



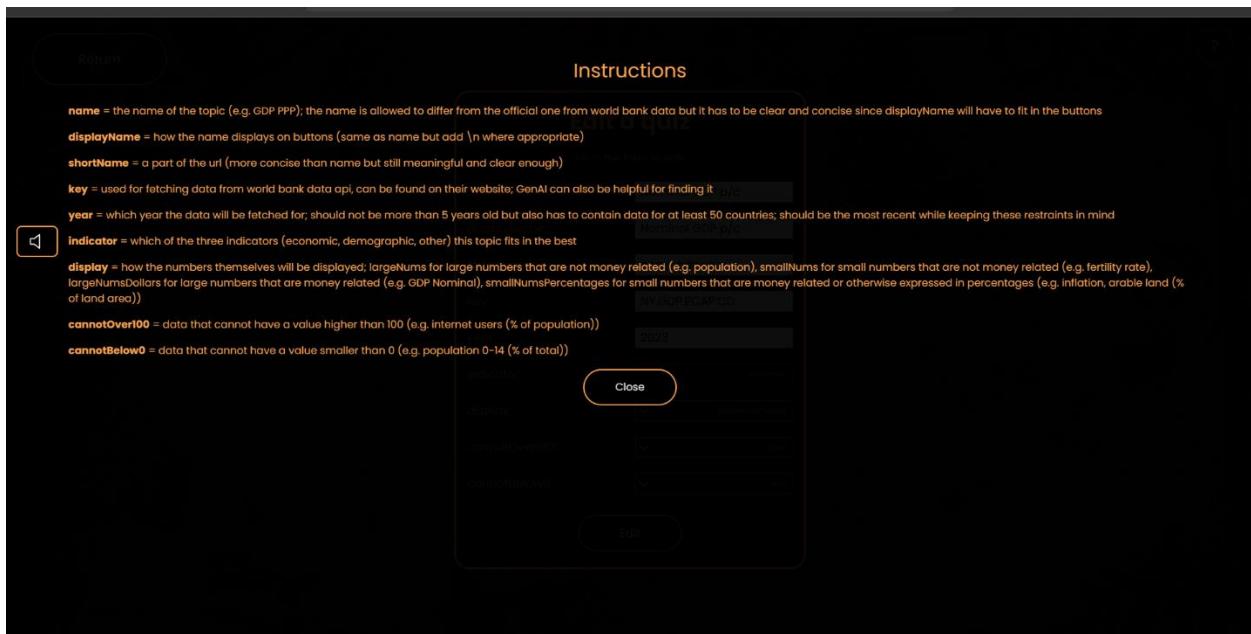
(Created by author)



(Created by author)

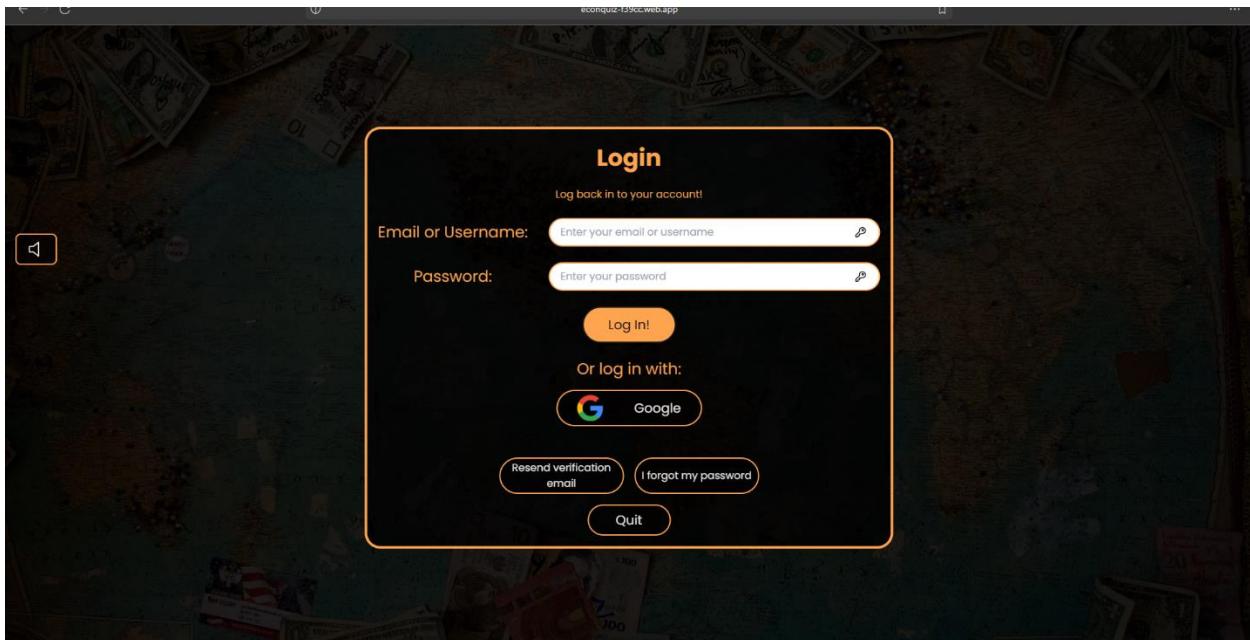


(Created by author)

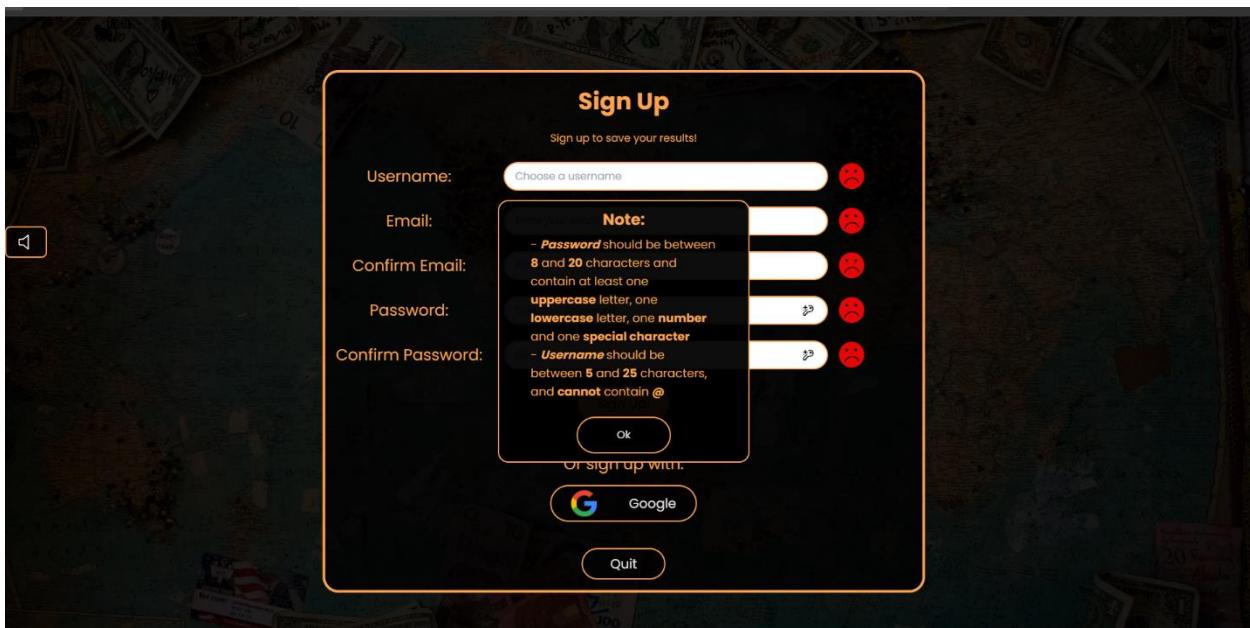


(Created by author)

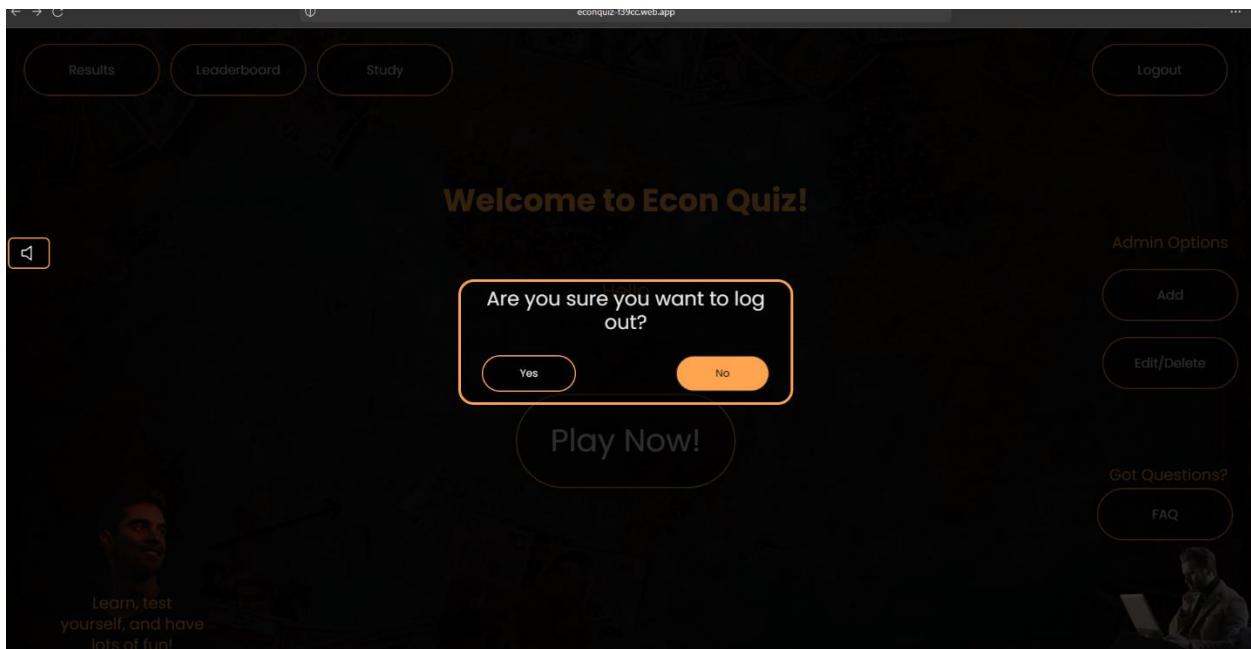
Authentication (login, signup, logout, resending verification email, reseting password):



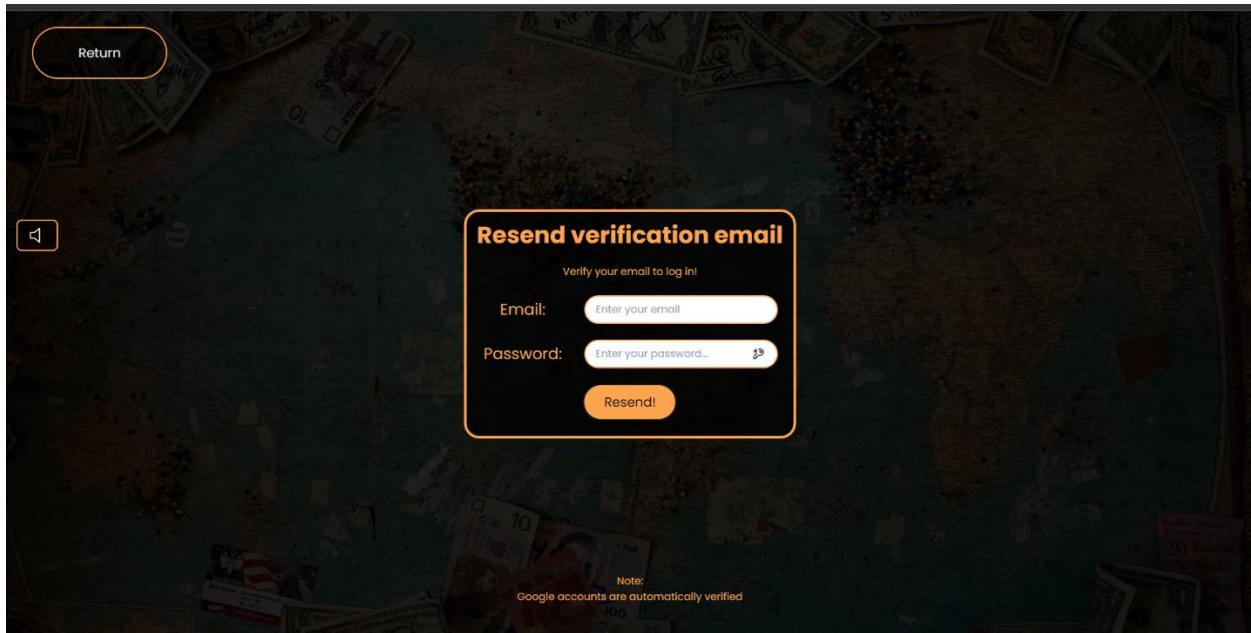
(Created by author)



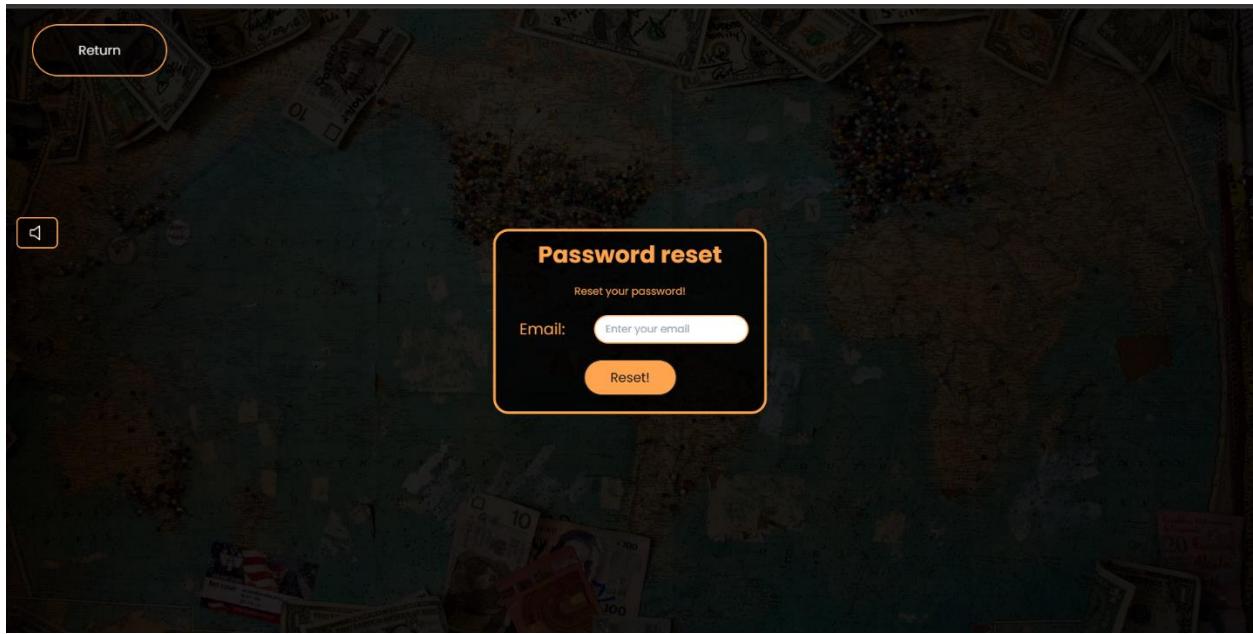
(Created by author)



(Created by author)

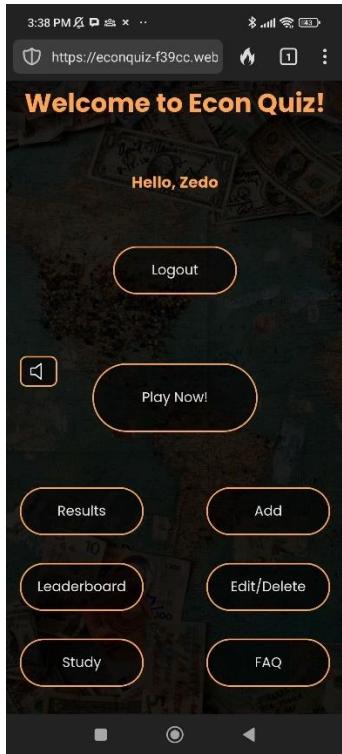


(Created by author)

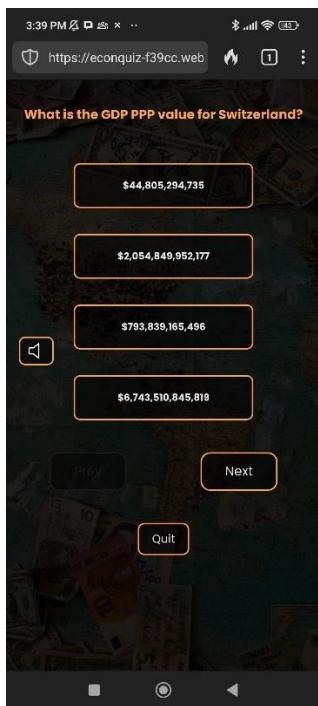


(Created by author)

Screenshots of some of the more relevant parts of the app on a mobile phone:



(Created by author)



(Created by author)



(Created by author)



(Created by author)



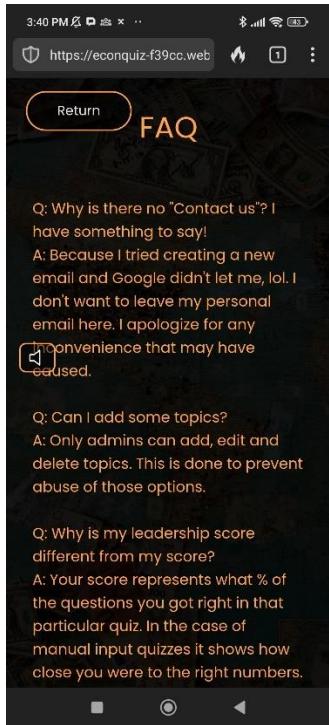
(Created by author)



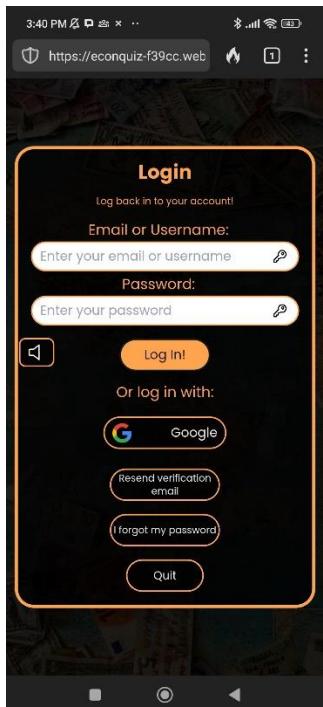
(Created by author)



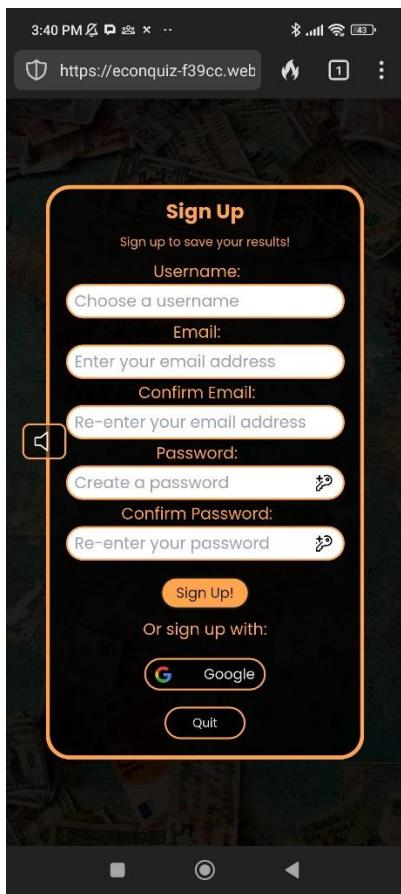
(Created by author)



(Created by author)



(Created by author)



(Created by author)

LITERATURE

1. Fastrich, G.M., Kerr, T., Castel, A.D. and Murayama, K. (2018). The role of interest in memory for trivia questions: An investigation with a large-scale database. *Motivation Science*, [online] 4(3), pp.227–250. doi:<https://doi.org/10.1037/mot0000087>.
2. McGillivray, S., Murayama, K. and Castel, A.D. (2015). Thirst for Knowledge: The Effects of Curiosity and Interest on Memory in Younger and Older Adults. *Psychology and aging*, [online] 30(4), pp.835–841. doi:<https://doi.org/10.1037/a0039801>.
3. Saifuddin, M., Kalsum, U.H., Hakimi, M., Mariani, M.D., Shima, S. and Airuddin, A. (2024). Descriptive Analytics Approach to Determine Popularity of Games Played on Android Platform. *International Journal of Intelligent Systems and Applications in Engineering*. [online] Available at: https://www.researchgate.net/profile/Mariani-Mohd-Dahlan/publication/380669270_Descriptive_Analytics_Approach_to_Determine_Popularity_of_Games_Played_on_Android_Platform/links/664891270b0d28457440f56d/Descriptive-Analytics-Approach-to-Determine-Popularity-of-Games-Played-on-Android-Platform.pdf [Accessed 6 Apr. 2025].
4. Salsabila, N. and Fi'aunillah, T.K. (2023). Students' Perceptions on the Use of Online Quiz Platforms in English Literature Department UINSA. [online] (682). Available at: <https://proceedings.uinsa.ac.id/index.php/iconfahum/article/view/1375> [Accessed 6 Apr. 2025].
5. Wang, A.I. and Tahir, R. (2020). The effect of using Kahoot! for learning – a literature review. *Computers & Education*, [online] 149, p.103818. doi:<https://doi.org/10.1016/j.compedu.2020.103818>.