ReadMe

TEAM:

- AQEEL KHAN ST10165886
- DYLAN FLASHMAN ST10083498
- TEWAHIDO TEFERA ST10083156
- RIAZ VALLI MAHOMMED ST10034485

Links:

Demonstration video: https://youtu.be/KM9-O6egfwQ

GitHub Repo: https://github.com/ST10083156/QuizCore_Part_3.git

API Link: https://github.com/ST10083156/OPSC_API.git

QuizCore

QuizCore is a dynamic and engaging quiz application designed to make learning fun, competitive, and accessible for all users. With an emphasis on customizable quizzes, leaderboards, and user-friendly features, QuizCore aims to combine the best elements of popular quiz platforms while offering a personalized and flexible experience.

We envisioned creating QuizCore as a way to blend education with interactive game elements, catering to both students and casual learners. By enabling users to create their own quizzes, compete with others, and track their progress, QuizCore offers a holistic and enjoyable approach to learning.

Purpose of the App

The primary goal behind **QuizCore** is to offer users a powerful tool for both learning and teaching, while also encouraging interaction through quizzes. Whether you're an educator designing quizzes for your class, or a student wanting to test your knowledge in a specific subject, QuizCore gives you the flexibility to do so, while also fostering competition and engagement through features like leaderboards and scores.

Key Features

1. **Custom Quiz Creation**
   - **Purpose**: Allow users to create personalized quizzes, tailoring content to their specific learning or teaching needs.
   - **Benefit**: This feature empowers users to design quizzes suited to their curriculum or area of interest, offering a unique learning experience.
2. **Leaderboards**

- **Purpose**: To introduce an element of competition and excitement to the quiz experience.
- **Benefit**: With the leaderboard, users can see how they rank compared to others, driving motivation and adding a competitive, game-like edge to learning.

3. **Score Display**
   - **Purpose**: Provide instant feedback to users after they complete a quiz.
   - **Benefit**: Users can view their performance immediately, giving them insight into their strengths and areas for improvement.
4. **Quiz Selection by Category**
   - **Purpose**: Allow users to browse and select quizzes from specific categories of interest.
   - **Benefit**: This feature enables targeted learning by helping users focus on quizzes relevant to their current area of study or interest.
5. **Settings Page**
   - **Theme Customization**: Users can change the theme of the app for a personalized interface.
   - **Difficulty Levels**: Adjust the difficulty of quizzes to match the user's learning pace.
   - **Language Selection**: Users can switch between a limited number of languages, making the app more accessible to non-English speakers.
6. **User Authentication (Google SSO)**
   - **Purpose**: Provide a seamless and secure login process.
   - **Benefit**: Users must log in via Google SSO to access the app, which ensures a more personalized experience by storing their quizzes, scores, and settings. If they don't have an account, they can easily create one.

New Features Implementation

1. **Biometric Face Recognition and Authentication**
   - **Purpose:** Adds an extra layer of security and ease of access.
   - **Benefit:** Allows users to access the app securely using biometric data, with traditional login as a backup.
2. **Push Notifications**
   - **Purpose:** Keeps users informed about quiz updates, new challenges, and performance alerts.
   - **Benefit:** Ensures users remain engaged and informed, enhancing the interactive experience.
3. **Additional Language Options**
   - **Languages Supported:** English, Spanish, and French.
   - **Benefit:** Expands the app's accessibility to non-English speakers, making it more inclusive.

4 Friends Feature

- **Purpose:** Allows users to add friends and send/receive friend requests.
- **Benefit:** Creates a social aspect, allowing users to engage with friends, compete on leaderboards, and track each other's progress.

- **Offline Mode**

  - **Purpose:** Enables users to access the app and complete quizzes in guest mode without an internet connection.
  - **Benefit:** Provides flexibility for users who may not always have internet access, broadening the app's usability.

Design Considerations

The design of **QuizCore** was shaped by the need for simplicity and user engagement. We wanted to ensure that users of all ages and technical expertise could navigate the app with ease, while also delivering rich functionality. Our choices reflect this:

- **User Interface**: The app's layout is intuitive, featuring easily accessible menus for creating quizzes, selecting categories, and changing settings.
- **Accessibility**: With the language and theme customization options, QuizCore is designed to be adaptable for different users.
- **Learning-Oriented**: From the leaderboard to score displays, every feature is tailored to enhance the user's learning experience in a fun, interactive way.

Technologies Used

- **Android Studio**: The entire app was developed using Android Studio, a powerful IDE for mobile app development.
- **Kotlin**: Kotlin was the primary programming language used for building the app, allowing for efficient, clean, and modern Android app development.
- **API Integration**: We incorporated external APIs to fetch quiz data dynamically, allowing users access to a broad range of topics.

GitHub and GitHub Actions

GitHub played a crucial role in the development of **QuizCore**, particularly for version control and collaboration. We utilized **GitHub Actions** to automate our testing pipeline, ensuring that every push to the repository triggered automated tests to verify the stability of our codebase.

GitHub Actions Workflow:

- **Automated Testing**: GitHub Actions was used to run unit tests and UI tests on our app to catch bugs early in the development process.
- **Continuous Integration**: Every change that is pushed to the repository triggers the automated workflow, keeping our development process efficient and reliable.

Unit Testing Details

- Unit testing is crucial to ensure the reliability of the app and to identify any potential issues early. In **QuizCore**, we have implemented automated testing using GitHub Actions to maintain code quality and verify functionality.
- **GitHub Actions**: The project integrates continuous integration (CI) through GitHub Actions, where tests are automatically run whenever new changes are pushed to the repository.

- **Types of Tests**: We have implemented unit tests that cover the following areas:
- **Quiz Creation**: Tests to ensure that quizzes are created with the correct format, questions, and categories.
- **Score Calculation**: Verifying that scores are correctly calculated and displayed after quizzes.
- **Leaderboard Updates**: Ensuring that the leaderboard updates in real-time based on user performance.
  This testing approach ensures that critical components of the app remain stable as new features are introduced or bugs are fixed. Automated testing helps us catch issues early, promoting a more robust and reliable learning experience for users.

API Interaction

In **QuizCore**, the integration of an API plays a pivotal role in enhancing the app's functionality and ensuring a seamless user experience. The API serves as the backbone for fetching quiz content, managing user data, and providing real-time updates to features like the leaderboard and score tracking.

By leveraging a RESTful API architecture, **QuizCore** communicates efficiently between the client (the mobile app) and the server. This interaction allows the app to dynamically retrieve quiz questions based on user preferences, such as category selection and difficulty level. Whether the user is creating a custom quiz or selecting a pre-made quiz from a specific category, the API ensures that all data is fetched, processed, and displayed in real-time

The API also handles user authentication, particularly through Google SSO, ensuring a secure and streamlined login experience. This secure API interaction is vital for managing user accounts, quiz history, and performance data, all while maintaining the privacy and integrity of user information.

In addition, the leaderboard feature, which tracks user performance across quizzes, is constantly updated by the API to reflect the most recent results. This real-time feedback system, made possible by the API, ensures a competitive and engaging learning experience for users. Overall, the integration of the API brings flexibility, scalability, and a responsive user experience to **QuizCore**, enabling the app to meet the dynamic needs of its users.

How to Install and Use

To install and run the app locally:

1. Clone the repository: git clone : https://github.com/ST10083156/QuizCore_Part_3.git
2. Open the project in **Android Studio**.
3. Sync Gradle files and build the project.
4. Run the app on an emulator or a physical device.

Usage

QuizCore is primarily focused on enhancing learning through interactive quizzes. Whether you're creating a quiz, competing on the leaderboard, or selecting quizzes by category,

QuizCore is designed to make learning both fun and effective. The app is perfect for classroom learning, self-study, or just casual knowledge testing.

Future Features

We are dedicated to continuously enhancing QuizCore, with plans to introduce even more features in future updates. Stay tuned for more exciting functionalities that will further enrich the user experience!

License

This project is licensed under the MIT License

We hope you enjoy using **QuizCore** and find it an invaluable tool for learning and teaching alike. Stay tuned for more updates!