**Report: KnowHub (Developing a Simple Quiz App with Firebase in Flutter)**

**Developed By :** Zehra Khuwaja (20SW065) , Fiza Khuwaja (20SW063)

**Assigned By:** Maam Mariam (MAD)

**1. Introduction**

This report outlines the development of a simple quiz app using the Flutter framework with Firebase integration. The app will consist of multiple screens, including a loading page, quiz pages, correct/wrong answer feedback pages, a final score page, and authentication screens for login and signup. Firebase Realtime Database will be used to store and retrieve quiz data, and Firebase Authentication will provide user authentication capabilities

**2. App Architecture**

The quiz app with Firebase integration will follow a similar architecture as previously described, with the addition of Firebase components:

1. Loading Page: Displayed while the app loads quiz data from Firebase.
2. Quiz Pages: Display quiz questions and answer choices to the user.
3. Correct/Wrong Answer Pages: Provide feedback on the user's answers.
4. Score Page: Display the final score.
5. Authentication Screens: Login and signup pages with validation and Firebase Authentication.

**3. Functionality**

The app will have the following functionality:

**The app will have the following functionality:**

* **Loading Page:**

Display a loading indicator.

Fetch quiz data from Firebase Realtime Database.

Navigate to the first quiz question when data is loaded.

* **Firebase Integration:**

Set up Firebase in the Flutter app, including authentication (login and signup) and database configuration.

Store quiz questions, answer choices, correct answers, and user scores in the Firebase Realtime Database.

* **Quiz Pages:**

Retrieve quiz questions and answer choices from Firebase.

Allow the user to select an answer.

* **Correct/Wrong Answer Pages:**

Display feedback indicating whether the user's answer was correct or wrong.

Show the correct answer if the user's answer was wrong.

Keep track of the user's score in Firebase.

* **Score Page:**

Retrieve the user's final score from Firebase.

Display the user's final score out of the total number of questions.

Allow the user to restart the quiz or exit the app.

* **Authentication Screens:**

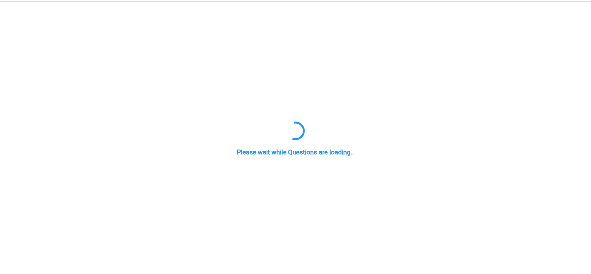
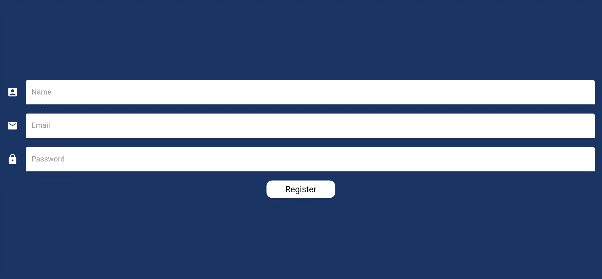
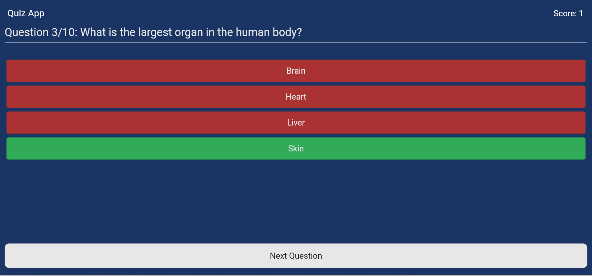
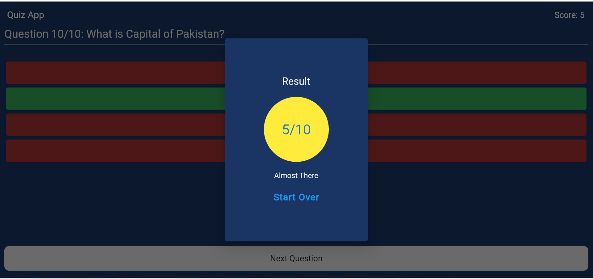
Implement user authentication using Firebase Authentication.

Create login and signup pages with validation for user credentials.

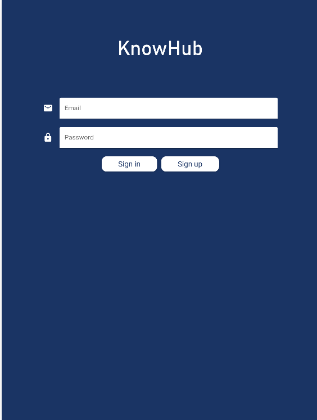
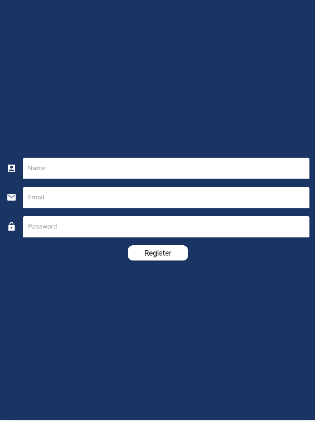
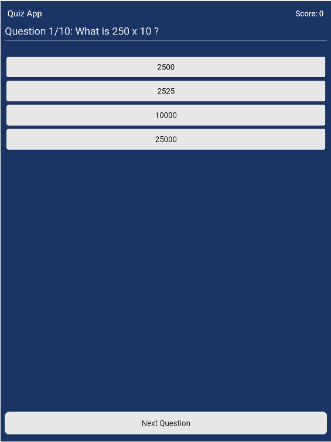
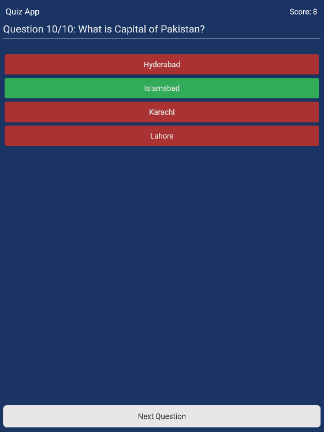
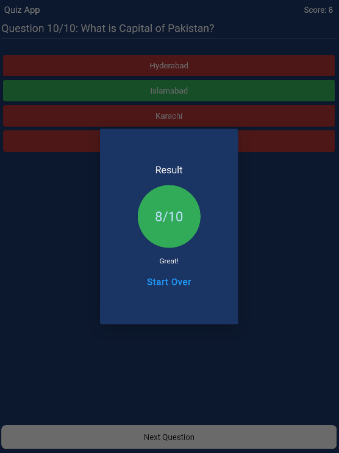
Provide a signout button on the result page to log the user out.

**4.ScreenShots**

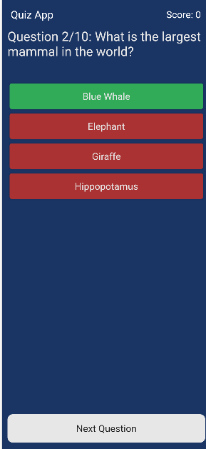
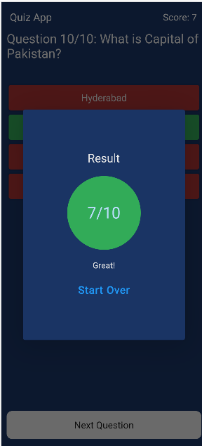
1. **DeskTop View**

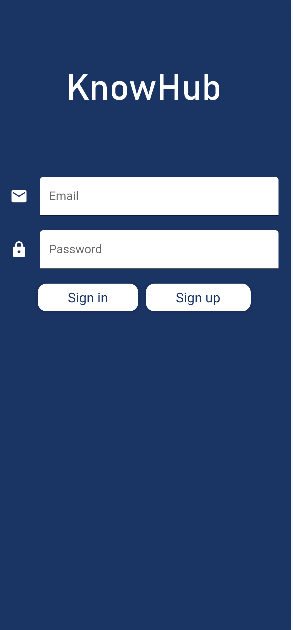
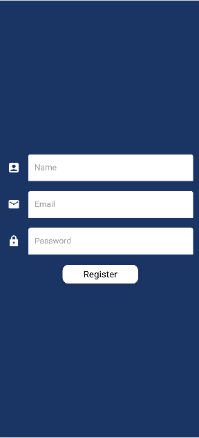
****** **

1. **Ipad View**

**    **

1. **Mobile View**

**  **

** **

**5.Responsiveness**

The app's user interface is designed to be responsive, adapting its layout and content to different screen sizes and orientations. Key strategies for achieving responsiveness include using MediaQuery, layout widgets like Container, Row, Column, and Expanded, media queries for conditional styling, and thorough testing on various devices.

**6. App Development**

The app is developed using the Flutter framework, Dart programming language, and Firebase for data storage, retrieval, and authentication. Key Firebase components include Firebase Realtime Database and Firebase Authentication.

**7. Conclusion**

In conclusion, this report outlines the development of a feature-rich quiz app in Flutter with Firebase integration. The app uses Firebase Realtime Database to store and retrieve quiz data, Firebase Authentication for secure user login and signup, and provides an interactive and engaging experience for users while testing their knowledge on various topics.

**8.Future Enhancements**

In the future, the app can be enhanced with additional features such as user profile management, the ability to create and share custom quizzes, a leaderboard to compare scores with other users, and more.

**9.GitHub Repository Link:**

**10. References**

* Flutter documentation: https://flutter.dev/
* Dart documentation: https://dart.dev/
* Firebase documentation: https://firebase.google.com/docs
* FlutterFire (Firebase plugins for Flutter): https://pub.dev/packages?q=flutterfire