

**Department of Computer Science**

**Faculty of Management and Technological Sciences**

**Gujarat Vidyapith, Ahmedabad - 380009**

**M.C.A. SEM-3 304: Project Report**

**Quantarks Chat Application**

**Submitted By**

**Sahil Lalani (212308043)**

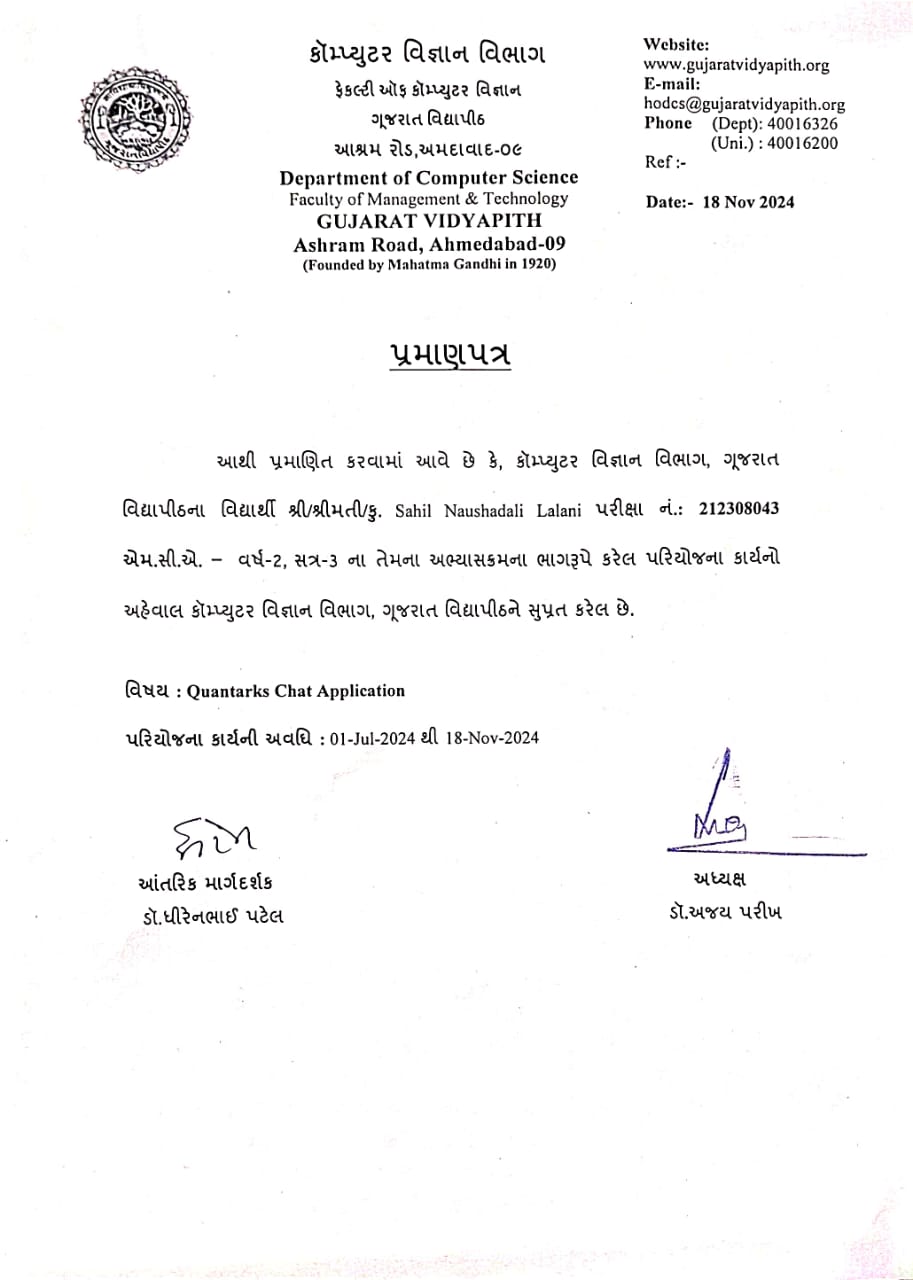
**Guide**

| **Dr. Dhiren Patel** |
| --- |
| **Professor** |
| **Department of Computer Science** |
| **Gujarat Vidyapith, Ahmedabad** |
| **Year 2024-25** |

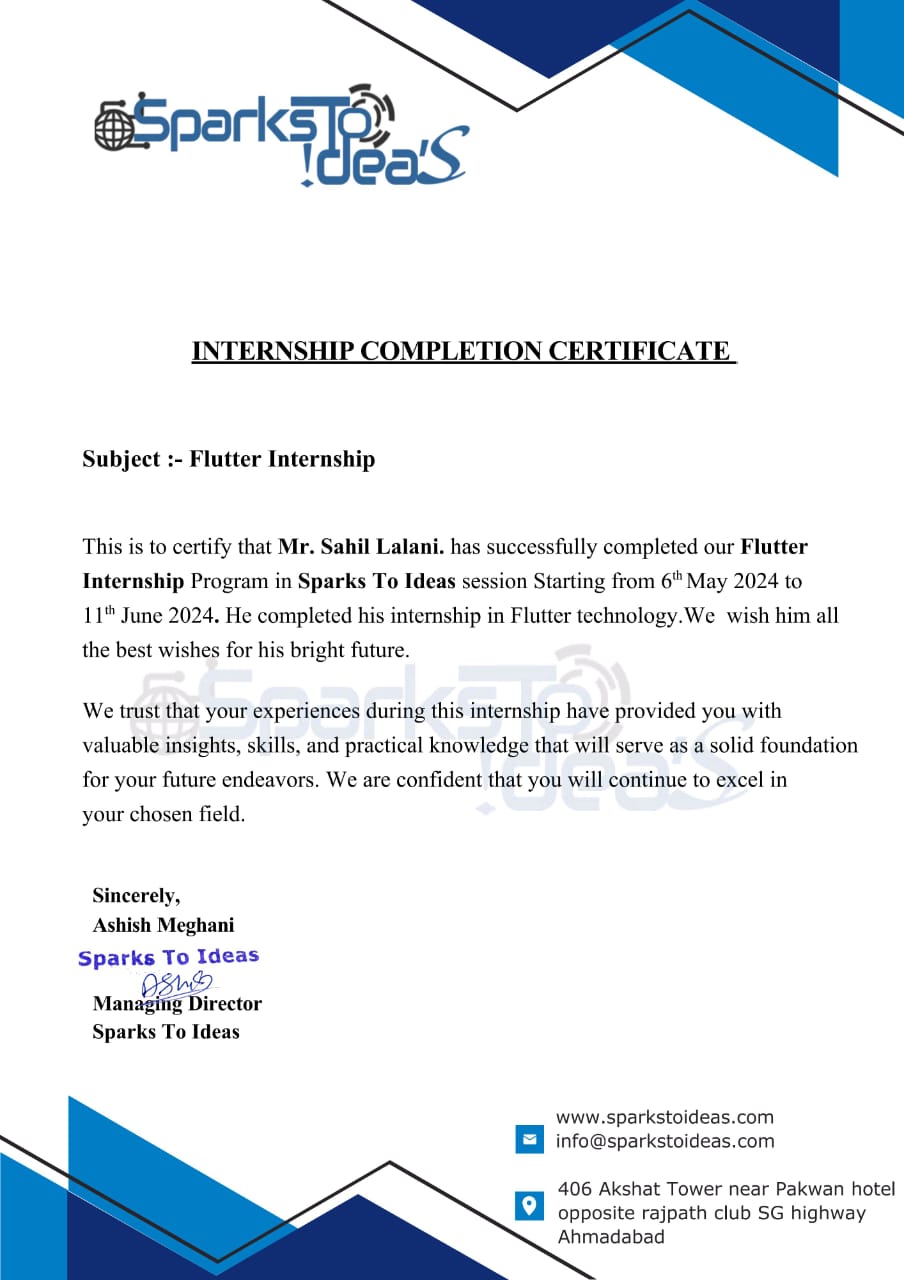




**Certificate From College**



**Certificate from Company**



INDEX

[1. About Gujarat Vidyapith 1](#_Toc184390276)

[2. Introduction 2](#_Toc184390277)

[3. Acknowledgement 3](#_Toc184390278)

[4. Project Overview 4](#_Toc184390279)

[**4.1** **Project Description** 5](#_Toc184390280)

[**4.2** **Objective of the Project** 6](#_Toc184390281)

[**4.3** **Scope of the Project** 7](#_Toc184390282)

[5. Development Environment – Software and Hardware 8](#_Toc184390283)

[6. Diagrams 9](#_Toc184390284)

[7. Data Dictionary 14](#_Toc184390285)

[ **Collection: chats** 14](#_Toc184390286)

[ **Collection: users** 14](#_Toc184390287)

[ **Collection: feedback** 14](#_Toc184390288)

[8. Screenshots 17](#_Toc184390289)

[9. Limitations of the Project 30](#_Toc184390290)

[10. Future Scope 31](#_Toc184390291)

[11 . Conclusion 32](#_Toc184390292)

[12 . References 33](#_Toc184390293)

# About Gujarat Vidyapith

**Gujarat Vidyapith** is a university located in Ahmedabad, Gujarat, India. It was founded in 1920 by Mahatma Gandhi, the leader of the Indian independence movement, and granted deemed university status in 1963. Gujarat Vidyapith is funded by the U. G. C. (University Grants Commission) for higher education programs. It was started as Rashtriya Vidyapith (National Institute of University Education) and was the wake of the Non-cooperative Movement. Mahatma Gandhi remained its life-long Kulpati (Chancellor). The Institute imparts higher education with an integrated system of education teaching from the Nursery to the Doctorate level.

The main objective is to prepare workers of character, ability, culture and dedication for the conduct of movements connected with the regeneration of country in accordance with the ideals given by Mahatma Gandhi. The guiding principles on which the integrated system of education, from Nursery & Basic Schools to Higher Secondary and from Bachelors to Master, M. Phil. & Doctorate level degree in various disciplines, is evolved include :-

|  |  |
| --- | --- |
| * Adherence to truth and non-violence * Participation in productive work with a sense of dignity of labour * Acceptance of equality of religions * Priority for the need of villages dwellers in all curricula and * Use of mother-tongue as a medium of instructions. |  |

Accordingly, the education system of Gujarat Vidyapith include regular participation in community work, residential life, social service, community prayers, simple and self reliant living, study tours and field studies, Hand spinning and training in Craft Work. The program of education has been linked with the national needs of rural uplift with a view to promote education conductive to the grass - root levels of the society.

# Introduction

The **Quantarks** **Real-Time Chat Application** was developed during my summer internship as a comprehensive solution to facilitate seamless, instant communication across multiple platforms. With the increasing demand for reliable and scalable communication tools, this project was designed to deliver real-time messaging, secure user authentication, and a responsive user experience using modern technologies like **Flutter** and **Firebase**.

Flutter, known for its cross-platform development capabilities, enabled the creation of a consistent and visually appealing user interface for Android, iOS, web, and desktop users. The use of Firebase provided robust backend support, including user authentication, real-time database functionality, cloud storage, and scalability. By integrating these technologies, the project ensures that users can connect effortlessly, irrespective of their device or platform.

The application’s core features include one-on-one real-time messaging, a user-friendly search and filter system for finding users by name or city, and media sharing capabilities for enhanced interaction. The chat interface is designed to be intuitive, offering smooth navigation and consistent performance. Additionally, the app implements Firebase Authentication to provide secure login and registration, while ensuring data privacy and session management.

One of the unique aspects of this project is its focus on user experience and accessibility. The application features a clean, responsive design that adapts to different screen sizes and devices. User profiles display online/offline status, and a sidebar navigation system allows easy access to key functionalities such as profile editing, password management, and logout options.

Data security and privacy were given top priority during development. Firebase's built-in security rules and encryption mechanisms ensure that user data and messages are protected from unauthorized access. The scalable infrastructure of Firebase further guarantees that the application can handle a growing number of users and messages without compromising on performance.

The project also lays a foundation for future enhancements, such as group chats, video sharing, and push notifications. These additions will make the application more versatile and capable of meeting diverse communication needs. Performance optimization measures, such as efficient Firebase database operations and lightweight app design, have also been considered for future implementation to support lower-end devices.

This application is not only a demonstration of modern development practices but also a practical example of how Flutter and Firebase can be used together to create real-world solutions. Through this project, I gained valuable experience in building scalable, secure, and user-friendly applications, while addressing challenges related to cross-platform compatibility, real-time communication, and data security.

# Acknowledgement

I am pleased to present my project, **Quantarks Chat Application**, developed as part of the MCA (Master of Computer Applications) Year-2, Semester-3 curriculum at the Department of Computer Science, Gujarat Vidyapith. This project represents a significant milestone in my academic journey and has provided me with invaluable insights into the field of real-time communication and application development.

The successful completion of this project would not have been possible without the guidance, support, and encouragement of several individuals and organizations.

First and foremost, I express my profound gratitude to **Dr. Dhiren Patel**, my esteemed guide, for his invaluable mentorship throughout the development process. His expert guidance, constructive feedback, and unwavering support played a pivotal role in shaping this project. Dr. Patel’s vast knowledge and experience in the field were instrumental in overcoming challenges and achieving the desired outcomes.

I am deeply thankful to **SparksToIdeas** for providing me with a platform to enhance my technical knowledge and skills. The training sessions and resources offered by SparksToIdeas significantly contributed to the practical implementation of the project.

I would like to specially acknowledge the efforts and guidance of my mentors at **SparksToIdeas**—**Mr. Parth Solanki**, **Mr. Kamal Yadav**, and **Ms. Shivani Rathod**. Their encouragement, technical expertise, and practical insights were critical in successfully developing this application. They generously shared their time and knowledge, which greatly enhanced my understanding of real-world software development practices.

Additionally, I extend my heartfelt appreciation to the **Department of Computer Science**, Gujarat Vidyapith, for fostering an environment of learning and innovation. The support and resources provided by the department enabled me to approach this project with confidence and dedication.

Lastly, I am grateful to all those who directly or indirectly contributed to the success of this project. Whether through technical advice, motivation, or moral support, their contributions have been invaluable.

The development of the **Quantarks Chat Application** has been a challenging yet rewarding experience, and I am thankful to everyone who helped me make this project a reality.

**Sahil Lalani**

# Project Overview

The **Quantarks Chat Application** is an innovative, real-time messaging platform developed to provide users with seamless communication capabilities across multiple devices and operating systems. The application is designed to bridge the gap between users by offering a modern, easy-to-use interface combined with powerful backend features that ensure secure and instant messaging. Built with **Flutter** for the frontend and **Firebase** for the backend, the Quantarks Chat Application is a cross-platform solution that enables users to chat, share media, and stay connected no matter the device or platform they are using.

The core of the application is its **cross-platform compatibility**. By using **Flutter**, the app is developed with a single codebase that works across Android, iOS, web, and desktop platforms. This ensures that users experience consistent and optimized performance no matter what device they are on. Whether on a mobile phone, tablet, or desktop, users can interact with the app in the same smooth and intuitive way, contributing to a unified user experience.

At the heart of the application's functionality lies **real-time messaging**, which is powered by **Firebase**. Firebase’s **Realtime Database** and **Cloud Firestore** allow the app to sync messages instantly across users, ensuring no delays in communication. Users can send text messages, images, and other media files, all of which are instantly available to the other participants in a conversation. Firebase’s backend services handle the heavy lifting of managing users, storing chat histories, and syncing data, allowing the app to scale efficiently and reliably.

The **user authentication system** integrated via **Firebase Authentication** provides secure and seamless login and registration processes. Users can authenticate using email/password combinations, ensuring that only authorized users can access their accounts. The app's security measures are bolstered by the use of Firebase's built-in security features, which safeguard user data and ensure the privacy of communication within the platform.

In addition to basic messaging, the app includes advanced features such as **image sharing**, where users can send and receive photos within chat threads. Images are uploaded to **Firebase Cloud Storage**, providing a fast and reliable medium for storing and retrieving media content. Another standout feature is **real-time presence updates**, allowing users to see whether their contacts are online or offline in real-time, enhancing social interaction.

The application also provides **search and filtering functionalities**, which allow users to easily find other users based on criteria like name or location. This feature ensures that users can connect with others in their network, further enhancing the social experience and encouraging engagement within the app.

Overall, the **Quantarks Chat Application** offers a comprehensive solution for real-time communication, combining an intuitive user interface with powerful backend services. Its scalability, cross-platform capabilities, and seamless user experience make it a versatile and future-proof messaging application. Looking ahead, the app will evolve with features like **push notifications**, **group chats**, and **end-to-end encryption**, all of which will contribute to a more robust, secure, and engaging user experience.

## **Project Description**

The **Quantarks Chat Application** is an innovative real-time messaging platform that allows users to communicate seamlessly across multiple platforms, such as Android, iOS, Web, and Desktop. This chat application is developed using Flutter for the frontend and Firebase for the backend, offering a complete cross-platform solution for users to exchange messages in real time.

At its core, the Quantarks Chat Application is designed to enable smooth, secure, and fast messaging. The app allows users to send and receive messages instantly, with all conversations being stored and synchronized in real time via Firebase’s Realtime Database or Cloud Firestore. Users can engage in one-on-one chats and share multimedia content like images, which are stored securely in Firebase Cloud Storage.

The user authentication process is securely managed using Firebase Authentication, which handles login, registration, and password recovery. The app ensures users’ data privacy and security by leveraging Firebase’s built-in security rules and by encrypting messages for end-to-end protection.

The app’s cross-platform design ensures that users can access their conversations across different devices and platforms with consistent performance and UI. The app includes several features such as search and filter options, real-time presence updates (showing whether users are online or offline), and a user-friendly interface that enhances the overall user experience.

This project is designed to grow and expand with additional functionalities such as group chats, push notifications, and media sharing, making it a scalable solution that can accommodate more users and more complex interactions over time.

The Quantarks Chat Application also focuses on providing seamless synchronization across all platforms. Whether a user switches from a mobile device to a web browser or vice versa, the app ensures that all messages are synchronized in real time, giving users a continuous and uninterrupted experience. Firebase's cloud infrastructure ensures that conversations are not lost, even in cases of device failure or switching. This synchronization, combined with the app's responsive design, provides an intuitive and unified experience for all users, regardless of the device they are using.

## **Objective of the Project**

 **Real-Time Communication**  
To implement real-time messaging between users using **Firebase's Realtime Database** or **Cloud Firestore**, ensuring instant message delivery and seamless updates within the chat interface.

 **Cross-Platform Compatibility**  
To create a seamless user experience on **Android, iOS, Web, and Desktop platforms** by using **Flutter**, providing a consistent, optimized chat interface across different devices.

 **User Authentication with Firebase**  
To integrate **Firebase Authentication** for secure user login and registration. This will include options like email/password authentication, with future plans for phone and third-party logins such as Google and Facebook.

 **Message Management and Storage**  
To store and manage messages in **Firebase Firestore** or **Realtime Database**, allowing users to access chat history and synchronize messages in real-time. Additionally, users will be able to share images through **Firebase Cloud Storage**.

 **Responsive User Interface (UI)**  
To develop a clean and intuitive user interface that works seamlessly on all devices, ensuring a user-friendly chat experience. The UI will include essential features such as a message input box, chat history, and user profile management.

 **User Status**  
To show users' **online/offline status** in real time, allowing them to know the availability of their contacts.

 **Search and Filter**  
To enable users to search for others by name or city, with the ability to filter the displayed users by city.

 **Data Security and Privacy**  
To utilize **Firebase’s security rules** to protect user data and implement **encryption** to ensure the privacy and security of messages exchanged between users.

 **Scalability**  
To design the backend to handle a growing number of users and messages efficiently by leveraging **Firebase’s scalable infrastructure**.

## **Scope of the Project**

The **Quantarks Chat Application** is designed to serve as a cross-platform messaging solution with an emphasis on **real-time communication** and **data security**. The scope of this project includes the development of a functional chat application with the following key areas:

1. **Cross-Platform Development**  
   The application will support multiple platforms, including **Android**, **iOS**, **Web**, and **Desktop**, making it accessible to a broad user base. **Flutter** ensures a unified codebase for all platforms, reducing development time and maintaining consistency across devices.
2. **User Authentication and Data Management**  
   The app will include secure **user authentication** features using **Firebase Authentication**, allowing users to securely log in, register, and manage their accounts. User data, including chat history and images, will be stored in **Firebase Firestore** and **Firebase Cloud Storage**, providing real-time synchronization across all platforms.
3. **Real-Time Messaging**  
   Real-time messaging functionality will be at the core of the application, ensuring that messages are delivered instantly to all participants in a conversation. The use of **Firebase's Realtime Database** or **Cloud Firestore** allows for seamless communication without delay.
4. **User Interface and Experience**  
   The app will feature a **responsive user interface (UI)**, designed to work smoothly across different screen sizes and devices. The interface will include essential features like a message input box, user profile, settings, and real-time chat updates.
5. **Media Sharing**  
   Users will be able to share **images** in their conversations, with the images being stored securely in **Firebase Cloud Storage**. Support for additional media formats like videos or documents is planned as part of future objectives.
6. **Future Enhancements**  
   The project has been designed with scalability in mind, ensuring that it can accommodate increasing numbers of users. Future objectives include adding **push notifications**, **group chats**, **end-to-end encryption**, and **performance optimization** to improve the overall functionality and user experience.
7. **Security and Privacy**  
   The application will implement **data security measures** to protect user information, including encrypted messaging and secure data storage, in line with best practices for privacy and security.

Overall, the **Quantarks Chat Application** is a scalable, cross-platform messaging app that provides real-time communication, robust backend support, and user-friendly features, while ensuring a secure and engaging experience for all users.

# Development Environment – Software and Hardware

**Development Environment**

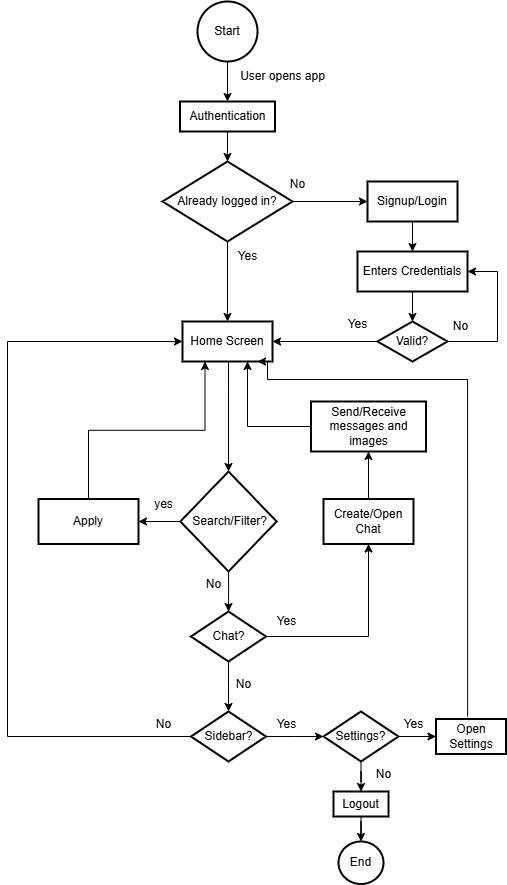
| **Category** | **Software/Tools** | **Version/Details** |
| --- | --- | --- |
| **Frontend Development** | Flutter SDK | Version: 3.0.0 or above |
|  | Dart | Version: 2.10 or above |
|  | Android Studio | Version: Latest stable release |
| **Backend Development** | Firebase (Firebase Authentication, Firestore, Firebase Storage) | Firebase SDK for Flutter, Latest Version |
|  | Firebase CLI | Latest Version |
| **Operating System** | Windows 11 | Latest stable version of Windows 11 |
| **Web Development** | Google Chrome or any modern browser for testing and debugging | Latest stable version |
| **Version Control** | Git | Version: 2.x and above |
| **Database** | Firebase Realtime Database or Firestore | Firebase Hosting & Firestore SDK |
| **Deployment** | Firebase Console (for deployment and management) | Web-based console, Latest Version |

**Hardware/Resources**

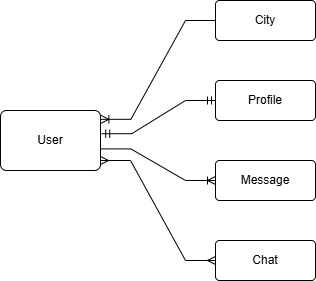
The Quantarks Chat Application was developed on a system with an **Intel Core i3 processor**, **12 GB of RAM**, and a **Solid State Drive (SSD)**. The system runs on **Windows 11**, providing a stable environment for development with efficient performance for Android Studio and other tools.

# Diagrams

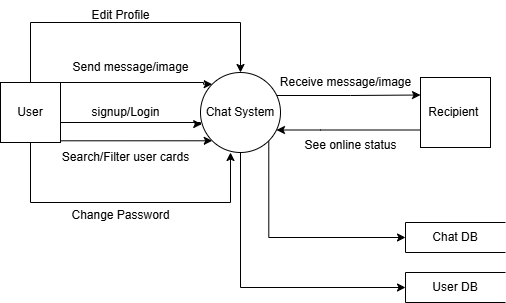
* **System flow chart**



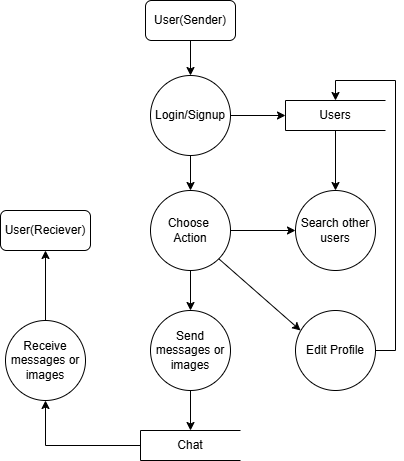
* **ER Diagram**



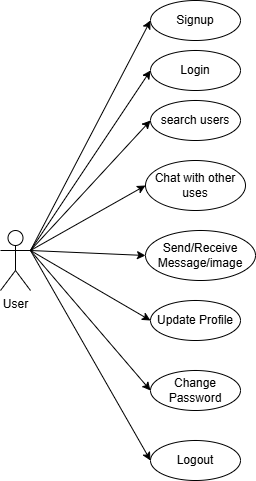
* **Data flow diagram(DFD) - Level 0**

****

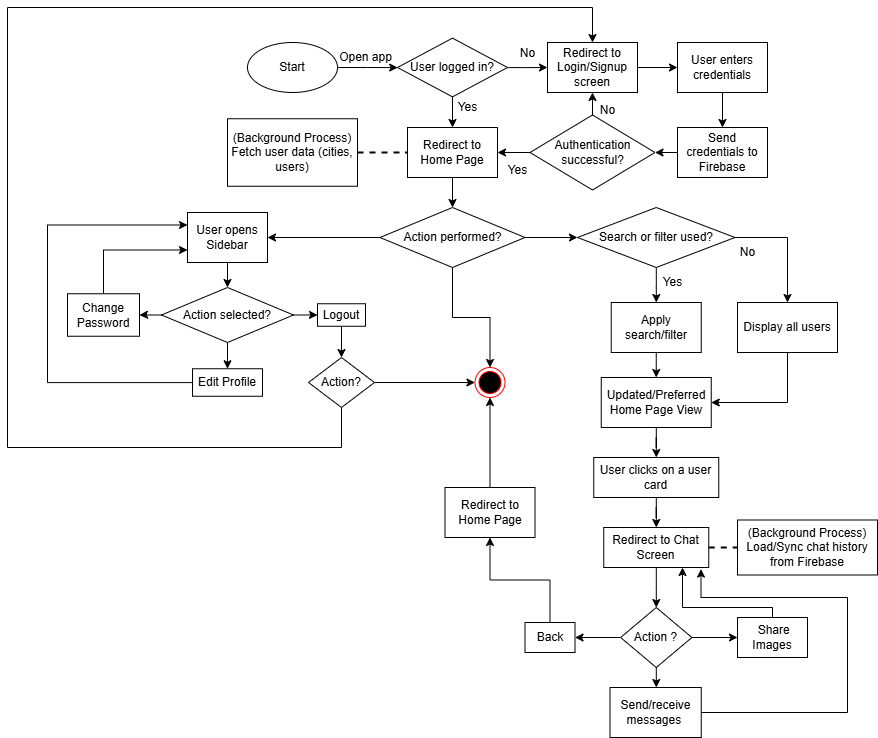
* **DFD(Level 1)**



* **Usecase Diagram**



* **Activity Diagram**



# Data Dictionary

### **Collection: chats**

| **Field Name** | **Data Type** | **Description** |
| --- | --- | --- |
| createdAt | timestamp | The timestamp indicating when the chat was created. |
| users | array of strings | Array containing the user IDs of the participants in the chat. |

#### **Sub-Collection: messages**

| **Field Name** | **Data Type** | **Description** |
| --- | --- | --- |
| recipientId | string | The user ID of the recipient of the message. |
| senderId | string | The user ID of the sender of the message. |
| text | string | The content of the message sent in the chat. |
| timestamp | timestamp | The timestamp indicating when the message was sent. |
| type | string | The type of message (e.g., text, image, video). |

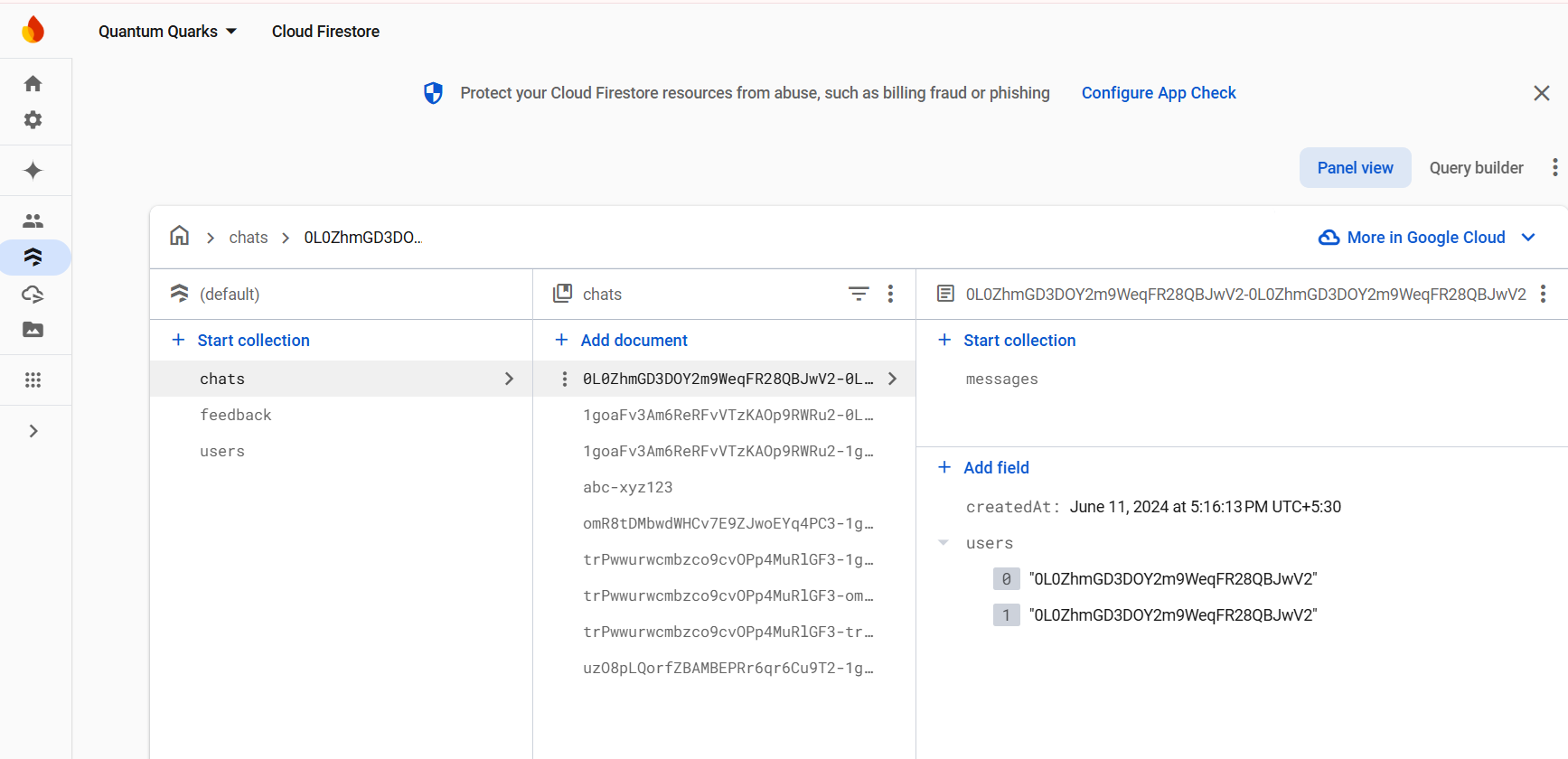
### **Collection: users**

| **Field Name** | **Data Type** | **Description** |
| --- | --- | --- |
| email | string | The email address of the user. |
| Name | string | The name of the user. |
| City | string | The city where the user resides. |
| Status | string | The user status(online/offline) |

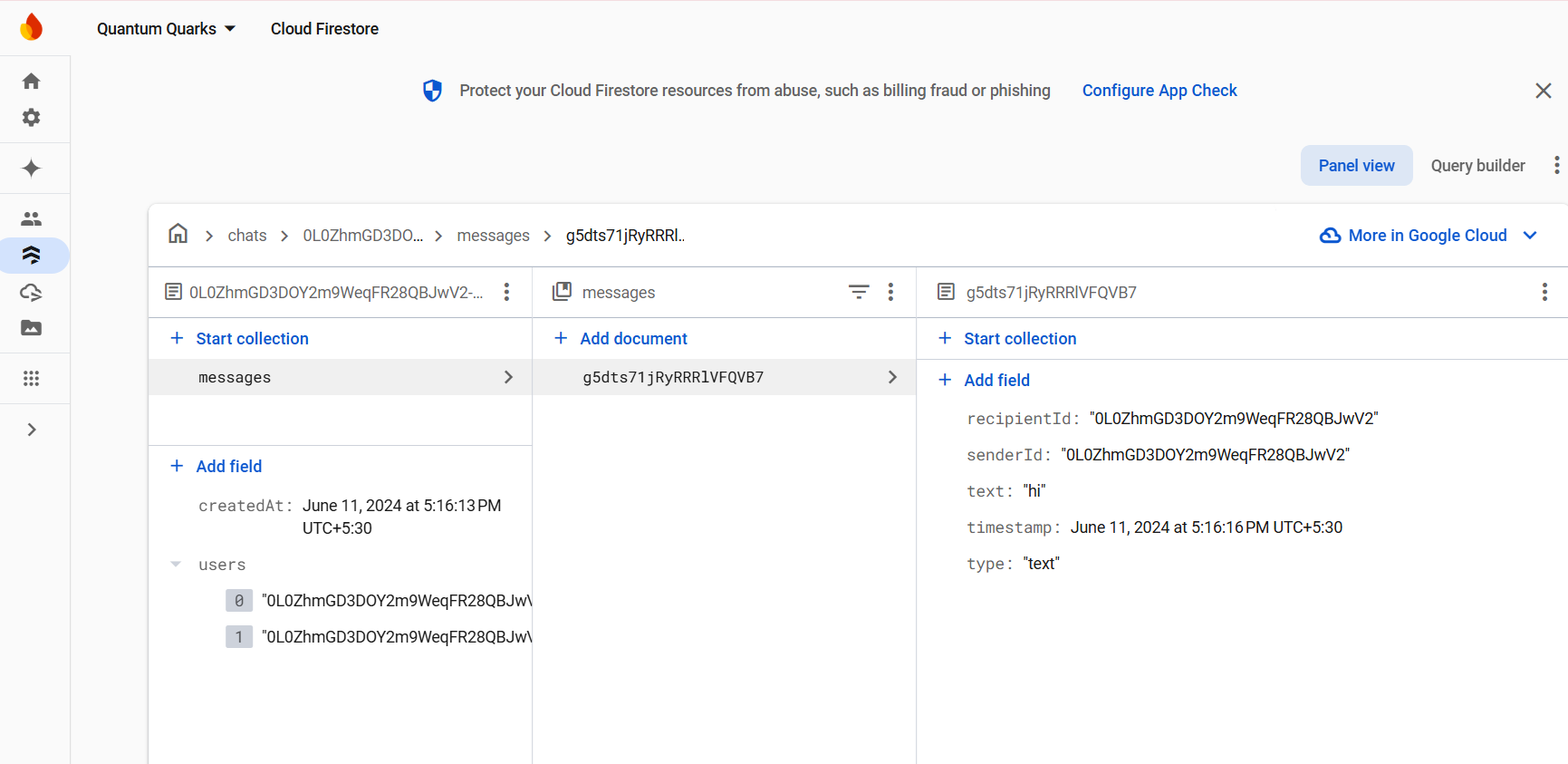
### **Collection: feedback**

| **Field Name** | **Data Type** | **Description** |
| --- | --- | --- |
| email | string | The email address of the user who submitted the feedback. |
| Review | string | The review or feedback content provided by the user. |

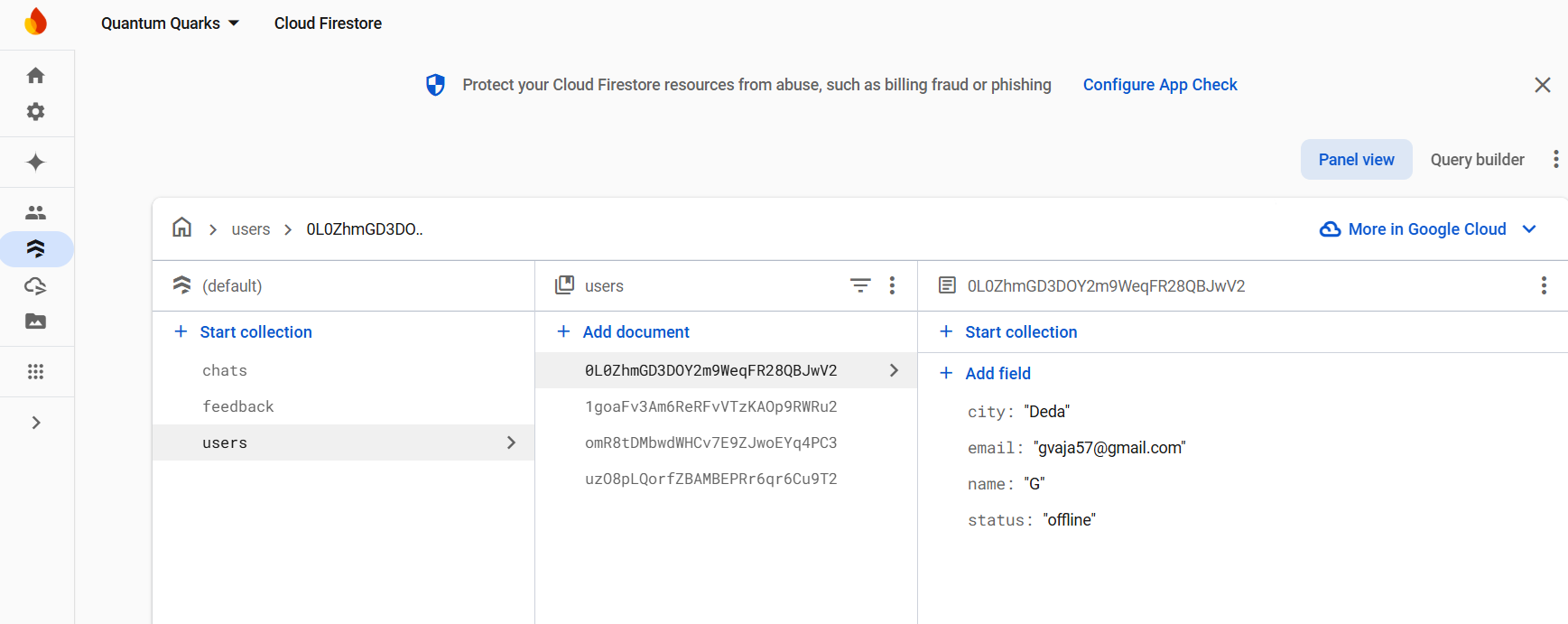
* **Chats Collection**



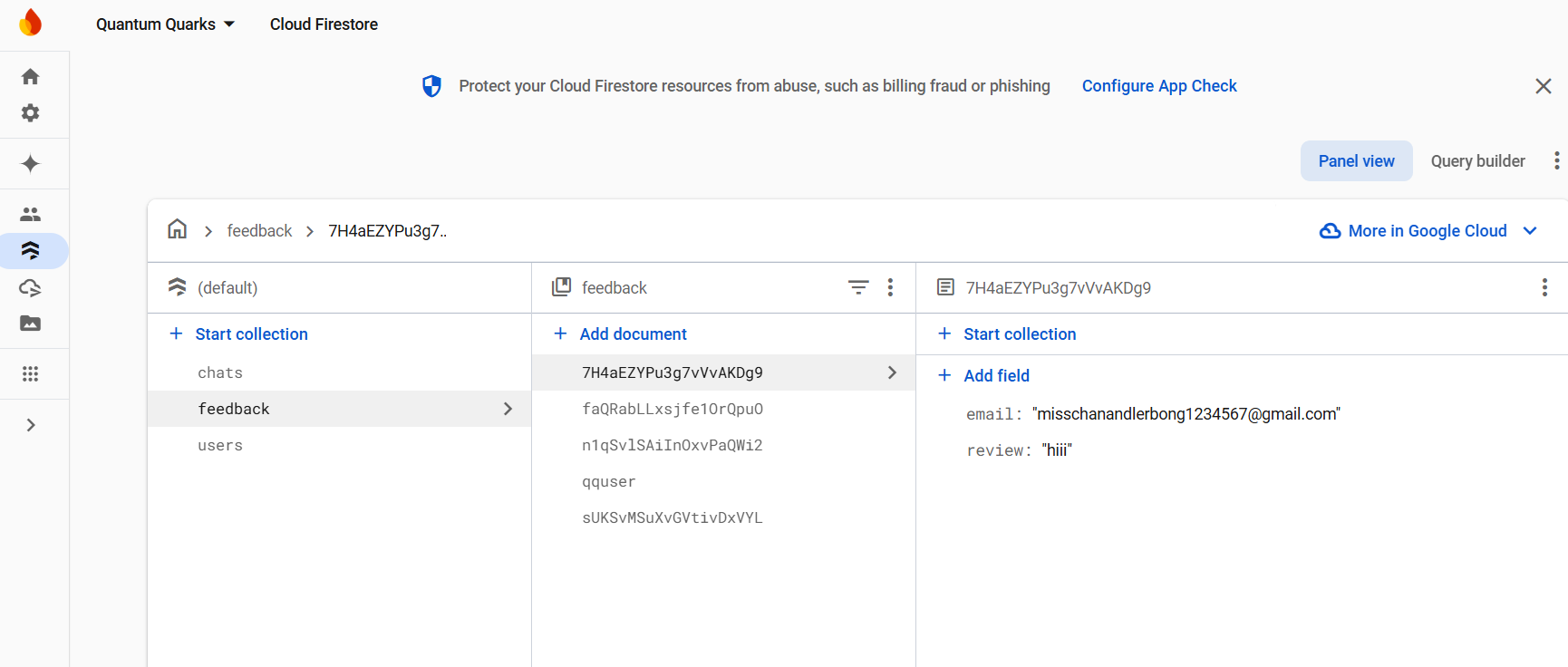
* **Chats(Messages) Collection**



* **Users Collection**

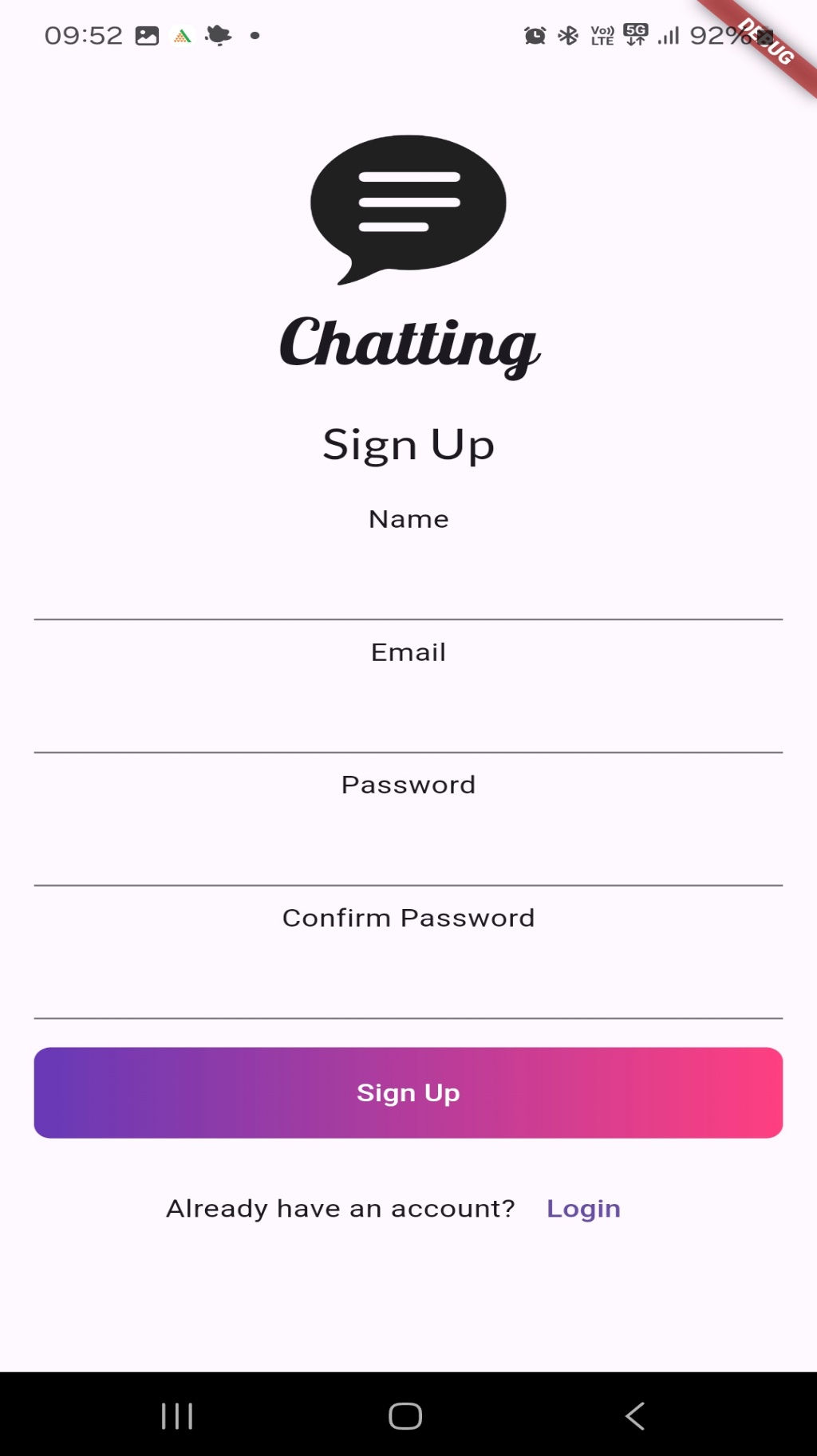


* **Feedback Collection**

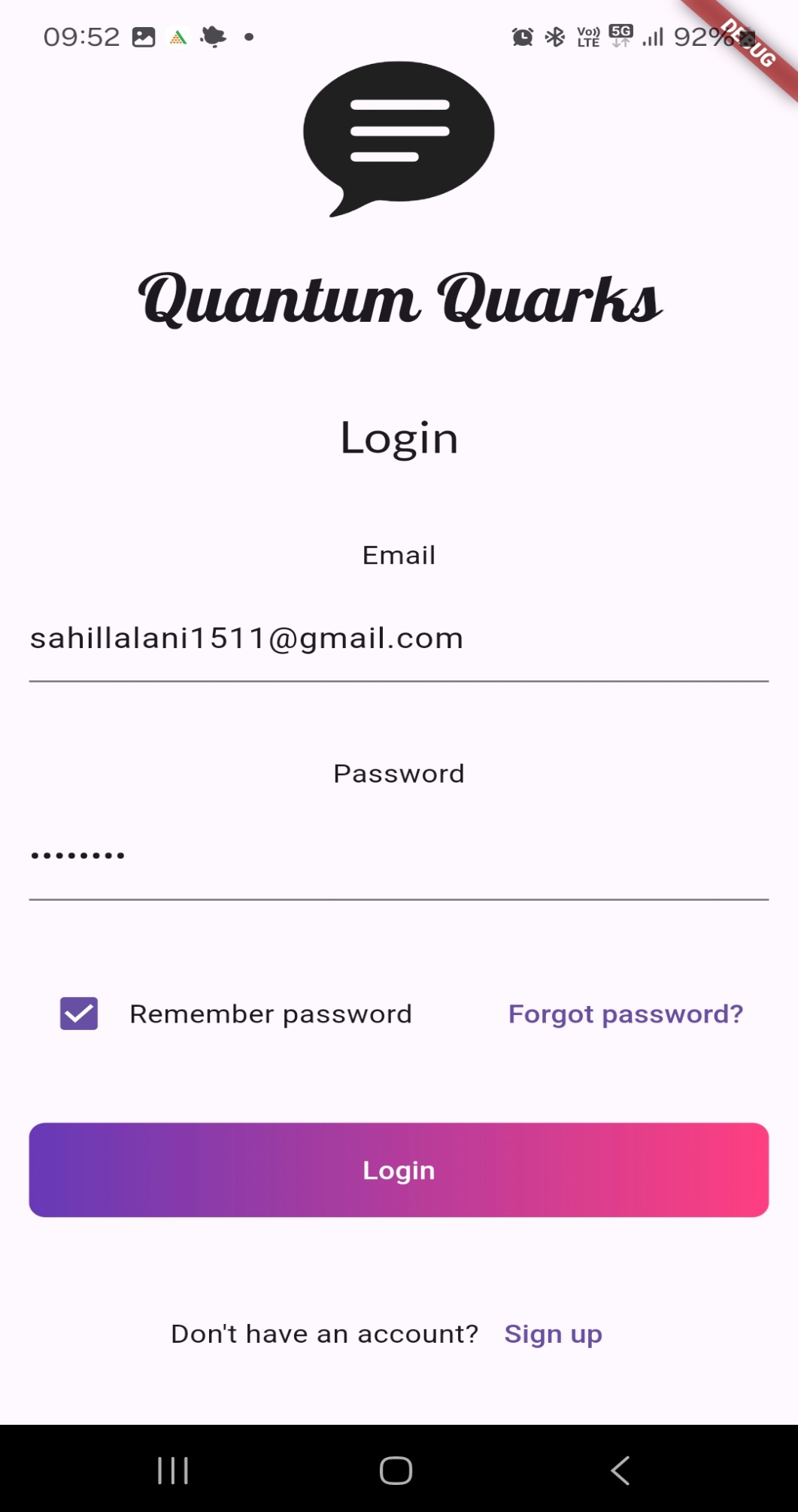


# Screenshots

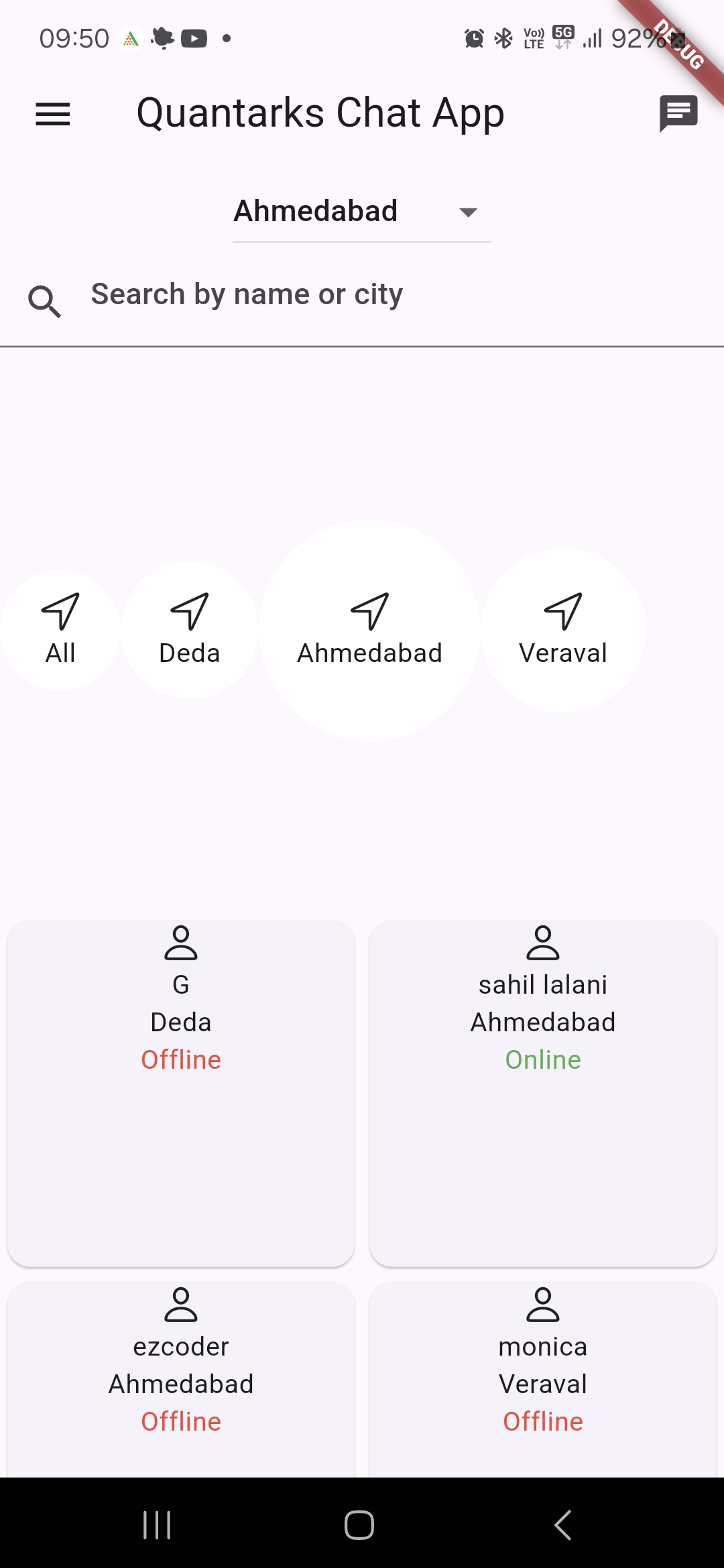
* **Signup Screen**

****

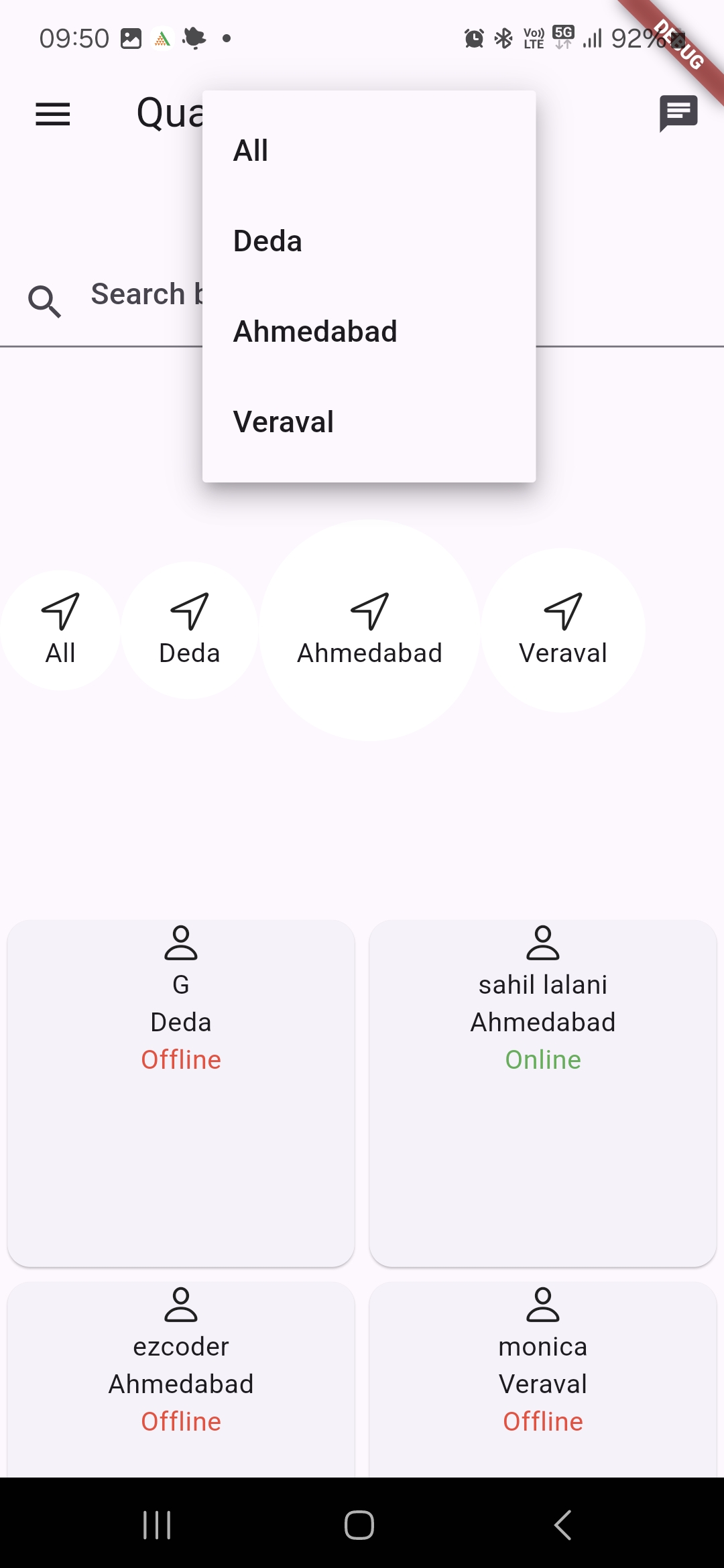
* **Login Screen**

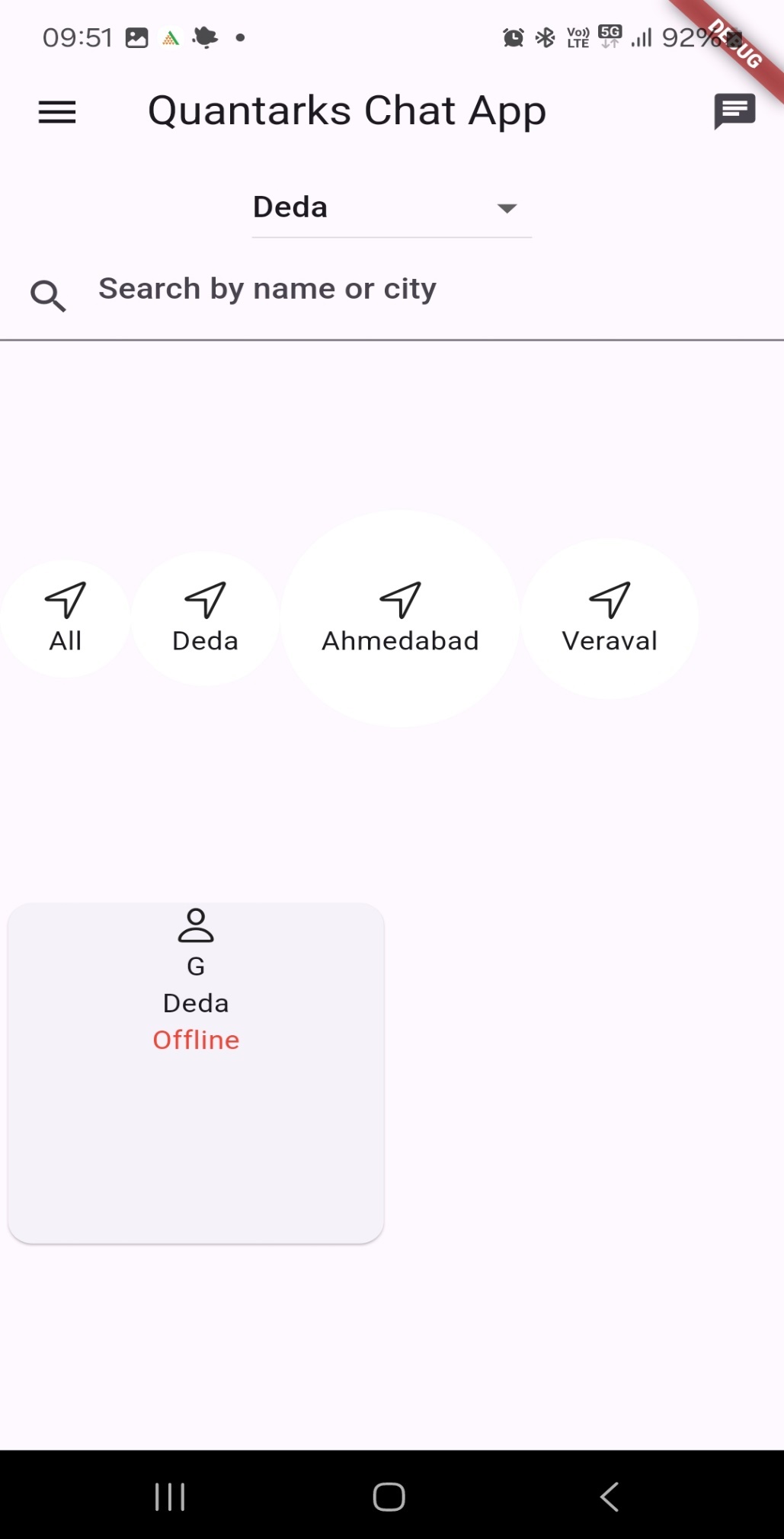


* **Home Screen**

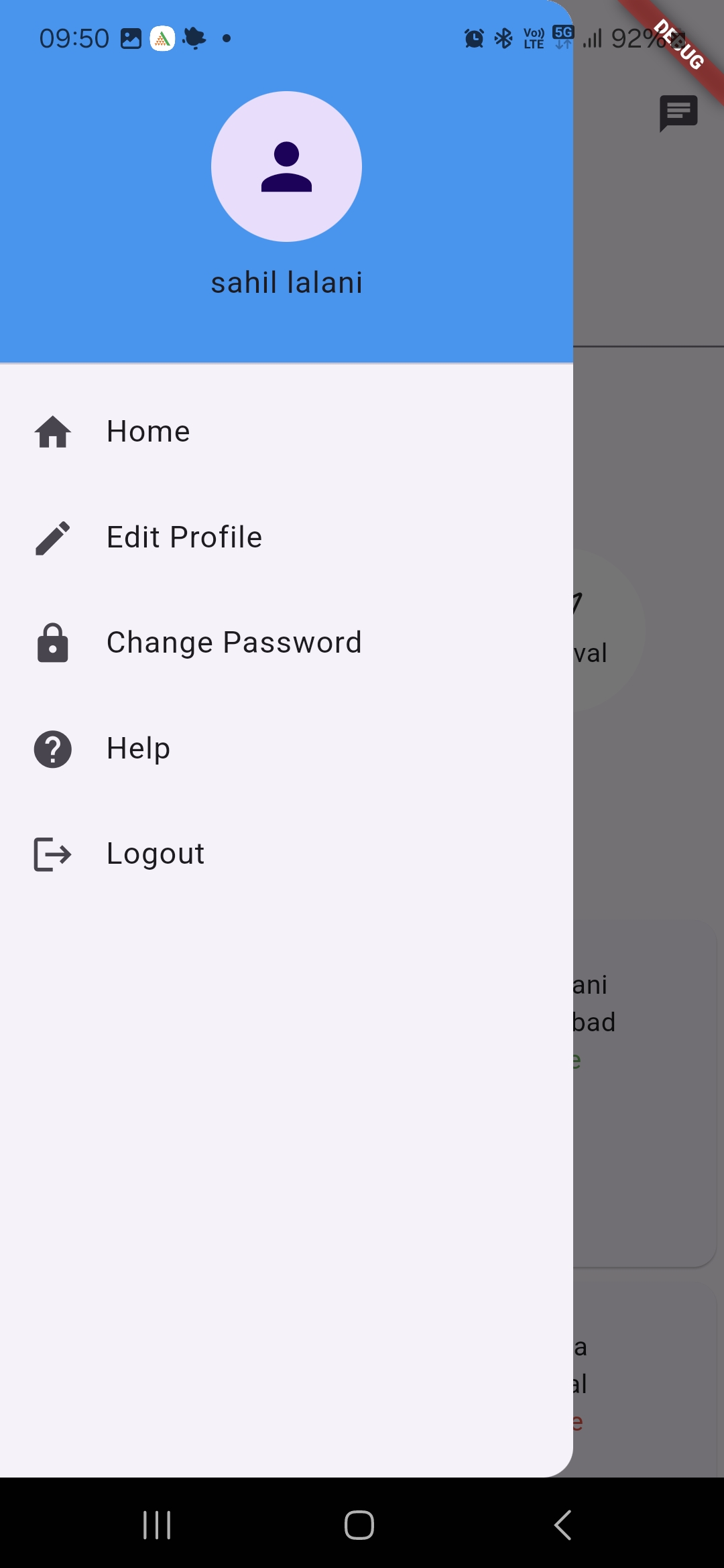


* **Search and Filter of Home screen**

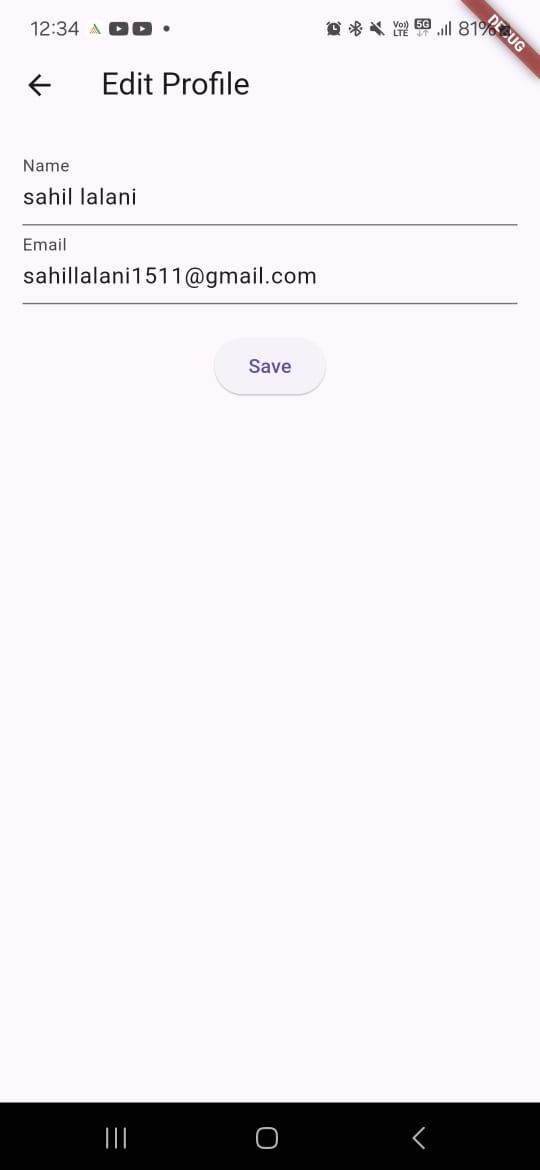


****

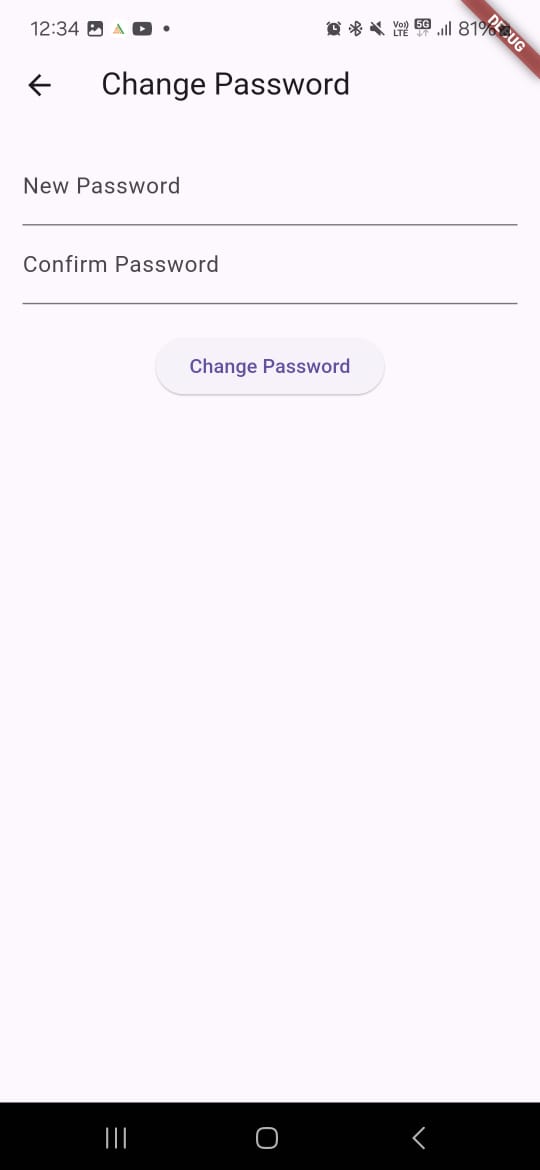
* **Home Screen Drawer**



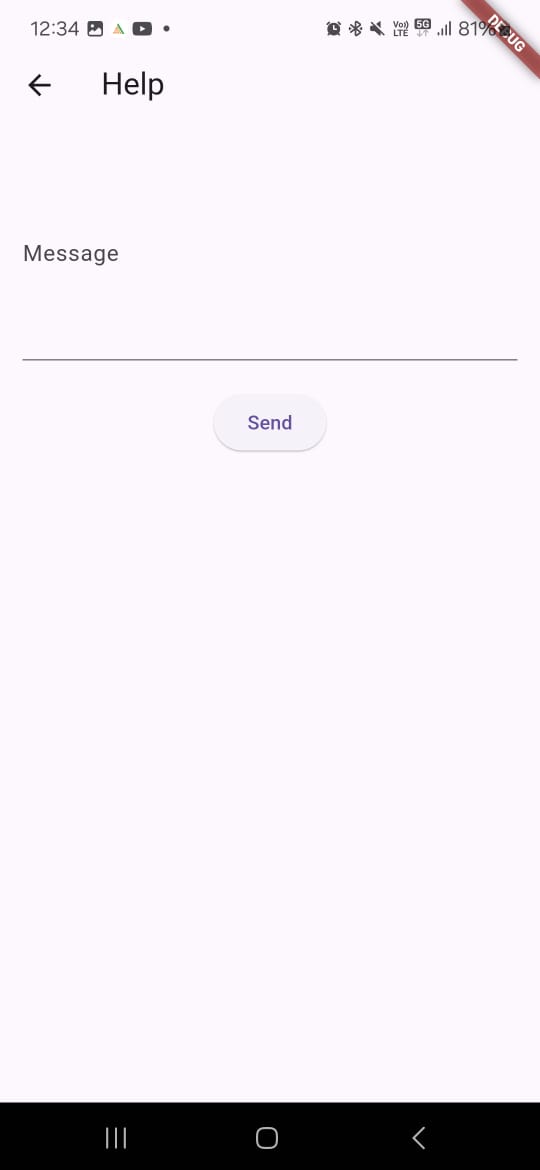
* **Edit Profile Screen**

****

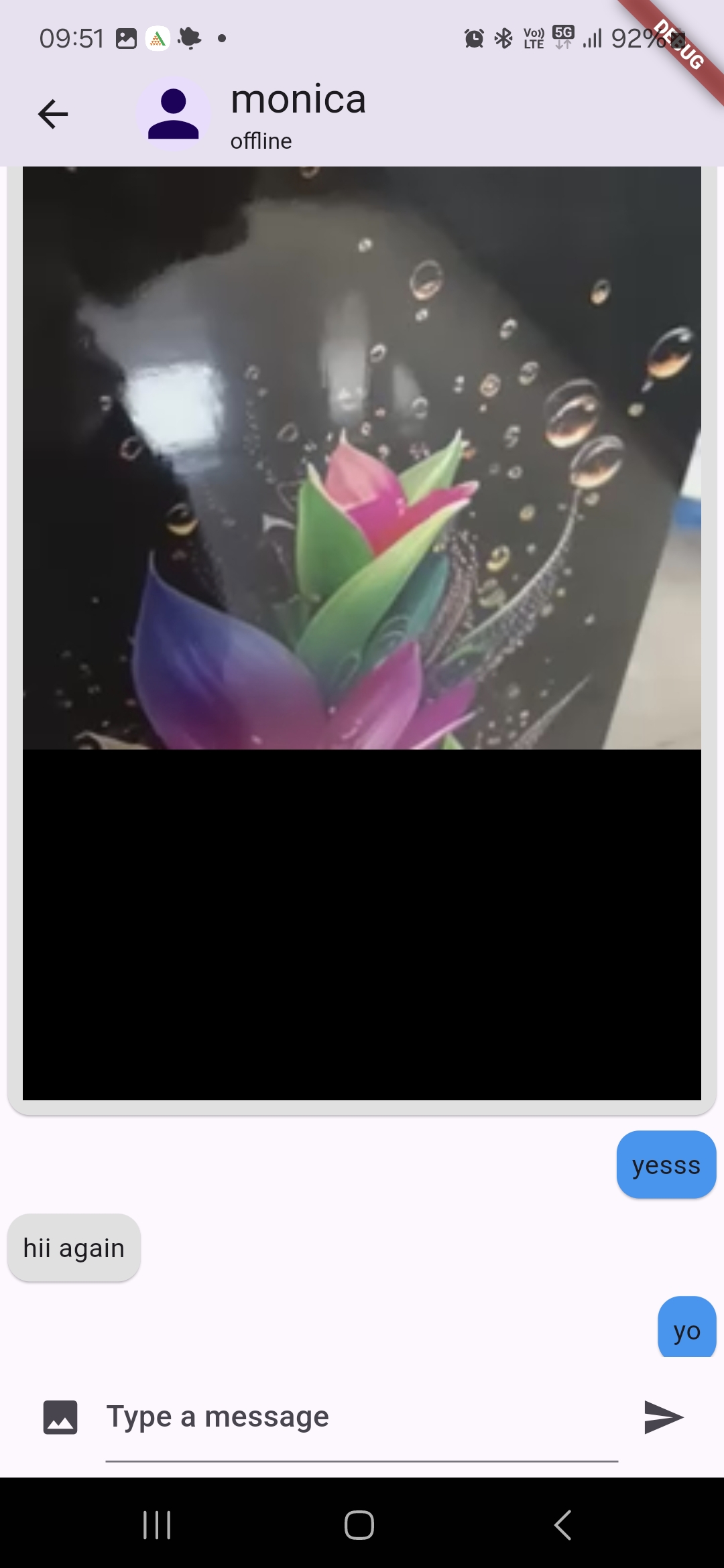
* **Change Password Screen**

****

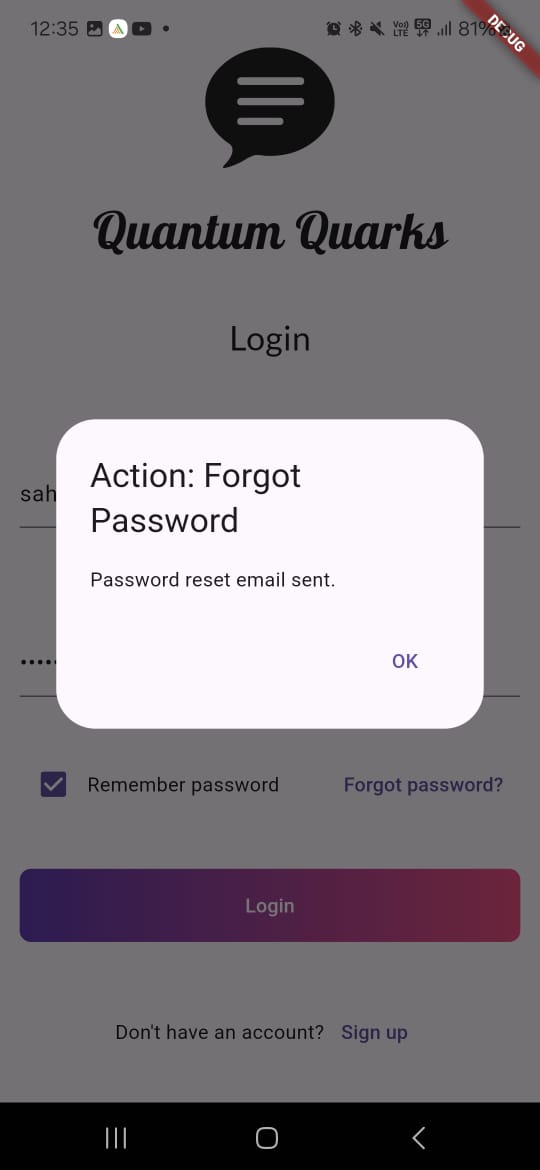
* **Help/Feedback Screen**

****

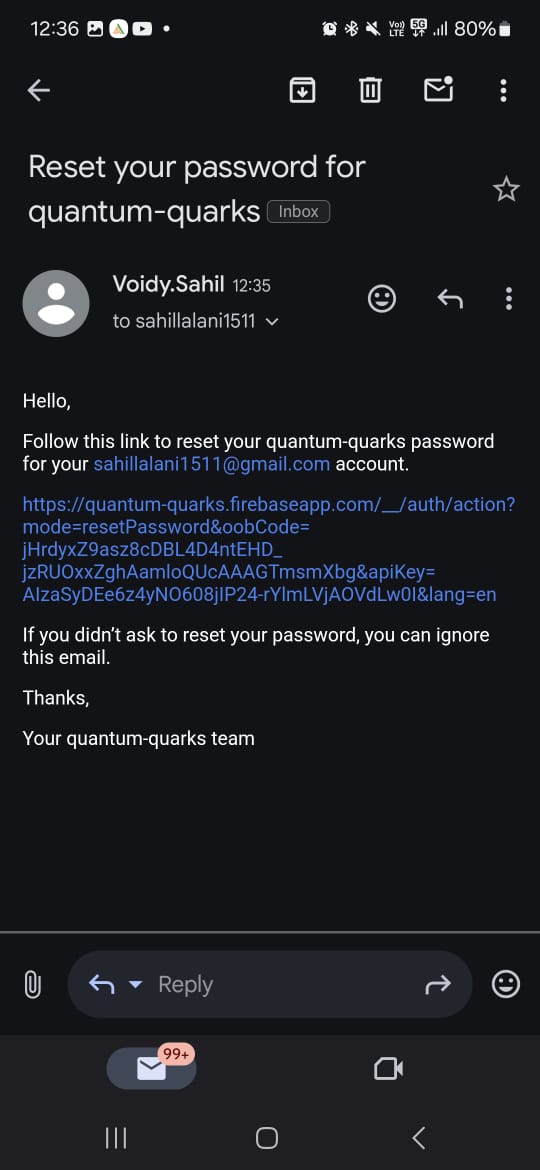
* **Chat Screen**

****

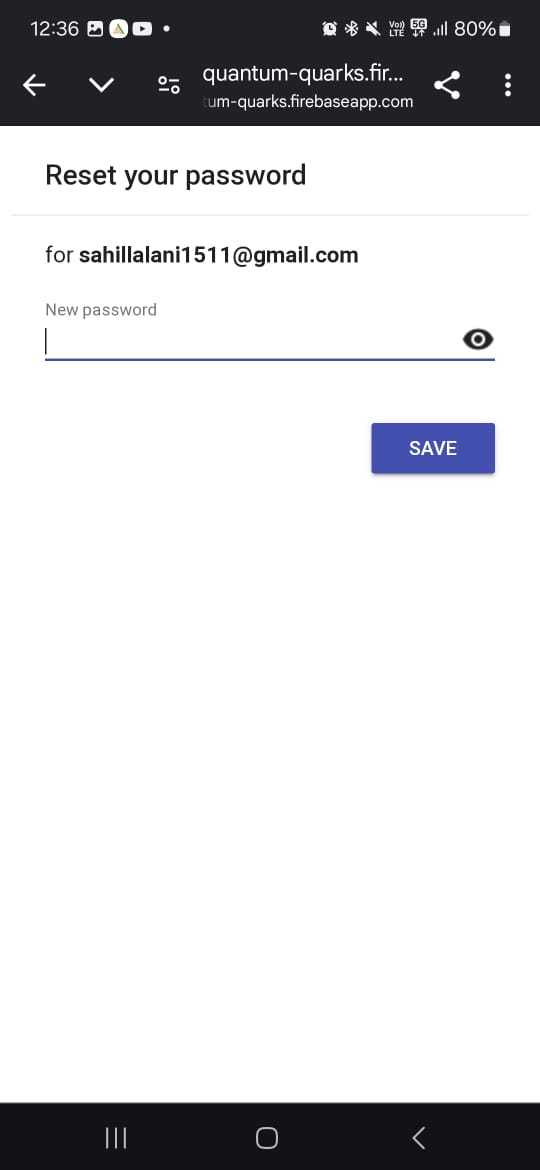
* **Forget Password Option(Login Screen)**

****

* **Email Link to Reset Password**

****

* **Reset Password Page(in browser)**

****

# Limitations of the Project

 **Lack of Notification System**:  
The system currently does not support push notifications or real-time alerts for new messages. This limitation can lead to delayed communication, especially if users are not actively monitoring the app.

 **Scalability Challenges**:  
With an increasing number of users and messages, the system may face performance issues. Efficient database management and optimization techniques will be crucial to prevent delays and ensure smooth operation.

 **Limited Media Support**:  
The system does not allow multimedia file sharing such as images, videos, or documents. This restricts the type of communication users can engage in, which is a common feature in modern chat applications.

 **Absence of End-to-End Encryption**:  
While the system ensures basic security, it lacks advanced end-to-end encryption protocols, making it less suitable for users concerned about privacy.

 **No Offline Functionality**:  
Users cannot compose or view messages without an internet connection. This can be a significant limitation for users in areas with intermittent or limited connectivity.

 **Limited Personalization**:  
Users are currently unable to set profile images, create groups, or manage group chats. This restricts personalization and collaborative features, which are often expected in chat systems.

# Future Scope

 **Integration of Real-Time Notifications**:  
Introducing push notifications to notify users of new messages instantly. This feature will enhance communication efficiency and improve the overall user experience.

 **Enhanced Security Features**:  
Incorporating end-to-end encryption to ensure user privacy and secure message transmission. Additional security measures like two-factor authentication can also bolster account protection.

 **Support for Multimedia Communication**:  
Upgrading the system to support the exchange of media files, such as images, videos, and documents, which will diversify the communication options available to users.

 **AI-Based Chat Assistance**:  
Adding features like chatbots for automated responses and customer support. AI-driven message suggestions and assistance can further improve user engagement and functionality.

 **Cross-Platform Compatibility**:  
Expanding the application to work seamlessly on web and desktop platforms, allowing users to access the system from various devices.

 **Scalability Improvements**:  
Implementing a microservices architecture and leveraging cloud-based solutions to handle increased user demand and data volume. This will ensure system reliability and performance as the user base grows.

 **Offline Messaging**:  
Developing offline functionality to enable users to draft messages without an active internet connection. Messages can be queued and sent once the user reconnects, enhancing usability in areas with limited connectivity.

 **Group Chat Functionality**:  
Adding support for group creation and management to enable collaborative conversations. This includes features like group messaging, member management, and assigning group roles.

 **Profile Image Customization**:  
Allowing users to set and update profile images for better personalization and a more interactive user experience.

# . Conclusion

The chat system serves as a reliable platform for facilitating basic user-to-user communication by offering core functionalities such as:

* **User Authentication**: Ensures only authorized users access the platform.
* **Message Sending and Retrieval**: Enables seamless text-based communication between users.

While the system successfully addresses essential communication requirements, it also presents opportunities for improvement. By addressing its current limitations, such as the lack of real-time notifications, end-to-end encryption, and multimedia support, the platform has the potential to transform into a feature-rich, user-friendly, and secure communication solution.

**Key Areas for Growth:**

1. **Enhanced Security**:  
   Adding features like encryption and advanced authentication mechanisms to build user trust and secure data transmission.
2. **Real-Time Functionality**:  
   Incorporating real-time notifications and live updates to improve responsiveness and engagement.
3. **User Personalization**:  
   Offering features like profile customization, group creation, and advanced settings for a more tailored user experience.
4. **Scalability and Performance**:  
   Leveraging modern technologies such as cloud computing and microservices architecture to handle increasing demand efficiently.
5. **Cross-Device Access**:  
   Expanding the system's reach by enabling compatibility across multiple platforms like desktop, web, and mobile devices.

**Vision for the Future:**

By focusing on these enhancements, the chat system can evolve into a competitive communication platform capable of meeting dynamic user demands. It will not only support diverse communication modes (text, multimedia, group chats) but also ensure a high level of security, reliability, and scalability, positioning it as a preferred choice in the market.

This progression reflects a commitment to innovation and adaptability, ensuring the system remains relevant in an ever-changing digital landscape.

# . References

* [**https://flutter.dev/**](https://flutter.dev/)
* [**https://dart.dev/**](https://dart.dev/)
* [**https://firebase.google.com/docs/**](https://firebase.google.com/docs/)
* [**https://www.youtube.com/c/flutterdev/videos**](https://www.youtube.com/c/flutterdev/videos)