SAFI INSTITUTE OF ADVANCED STUDY (SIAS)

(Affiliated to University of Calicut)

Vazhayoor East, Malappuram– 673633

Department of Computer Science and Applications

Project Report on

Location-Based
Anonymously Chatting App

Done by

Naswih Abdurahiman	**(Reg No. SIAUSCS011)**
Muzna Fathima KT	**(Reg No. SIAUSCS001)**
Sanath Habil KP	**(Reg No. SIAUSCS035)**
Safna	**(Reg No. SIAUSCS023)**

Under the Guidance of
Arshad P T

2020-2023

SAFI INSTITUTE OF ADVANCED STUDY (SIAS)

(Affiliated to University of Calicut)

Vazhayoor East, Malappuram– 673633

Department of Computer Science and Applications

CERTIFICATE

Certified that this is the bonafide report on the project work entitled

Location-Based
Anonymously Chatting App

Done by

Naswih Abdurahiman	**(Reg No. SIAUSCS011)**
Muzna Fathima KT	**(Reg No. SIAUSCS001)**
Sanath Habil KP	**(Reg No. SIAUSCS035)**
Safna	**(Reg No. SIAUSCS023)**

During the academic year 2020-2023 in partial fulfilment of the requirement for the award of degree

Bachelor of Computer Science
Of
University of Calicut, Malappuram Dist., Kerala

Internal Guide   **External Examiner**

Internal Examiner   **Head of the Department**

SAFI INSTITUTE OF ADVANCED STUDY

Affiliated to the University of Calicut. Recognised by UGC 2
(f)Accredited by NAAC with **'A++ Grade'** (3.54)
Rasiya Nagar, Vazhayur East P.O., Ramanattukara
Via,Malappuram District - 673 633, Kerala
Ph: 0483- 2880000 (100 Lines)
🌐www.sias.edu.in ✉ mail@sias.edu.in

TO WHOMSOEVER IT MAY CONCERN

This is to certify that the following students, **NASWIH ABDURAHIMAN (SIAUSCS011), MUZNA FATHIMA KT (SIAUSCS001), SANATH HABIL KP (SIAUSCS035), SAFNA (SIAUSCS023)** of **SAFI INSTITUTE OF ADVANCED STUDY** has successfully completed their project entitled "LOCATION-BASED ANONYMOUSLY CHATTING APP" in Flutter technology . The duration of the project was from NOVEMBER 2022 to MARCH 2023.

During this period, we found them hardworking and punctual. We wish them all the very best in their future endeavors.

Principal

Acknowledgement

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning about the people who made it possible, whose constant guidance and encouragement has crowned the efforts with success.

First and foremost, we sincerely thank the 'God Almighty' for his grace for the successful and timely completion of the project.

We express my sincere gratitude to **Prof. E. P. Imbichikoya**, our Principal and **Mr. Abdul Samad C.**, Head of the Department, Computer Applications providing the facilities and for all the encouragement and support.

We owe special thanks to **Mr. Arshad P.T.**, my project guide for his immense support encouragement, guidance and valuable suggestions that has contributed a lot in making this project what it is today.

We express my sincere thanks to other staff members in the department of Computer Applications who have extended their helping hand whenever needed.

Finally, we would like to acknowledge the heartfelt efforts, comments, criticisms co-operation and tremendous support given to us by our dear friends during the preparation of the project and also during the presentation without whose support this work would have been all the more difficult to accomplish.

Naswih Abdurahiman
Muzna Fathima KT
Sanath Habil KP
Safna

CONTENTS   **Page No**

1. INTRODUCTION

1. INTRODUCTION

1.1 Overview of the system

Technology has changed the way we communicate, but the concept of anonymous chatting remains mostly unexplored. Our location-based anonymously chatting app aims to change that by providing a platform where users can anonymously chat with others nearby. The app uses location-based services to connect users with other anonymous users in their area. It will allow users to share their thoughts, ideas, and experiences with others nearby without the fear of being judged or exposed. The app is designed to provide a secure and anonymous platform for users to communicate with each other. It will not require users to provide any personal information, and all communications will be encrypted to ensure user privacy. The app will have a simple and intuitive user interface that will allow users to easily navigate and find other anonymous users nearby. Chat rooms will be created by the developers and based on popular topics or interests. Users can join the chat rooms and participate in discussions. The app will use natural language processing and sentiment analysis to filter out any inappropriate content and ensure a safe and respectful chat environment. In summary, our location-based anonymously chatting app will provide a unique platform for users to anonymously chat with others nearby, without the fear of being judged or exposed. The app will use location-based services, encryption, and natural language processing to provide a secure and respectful chat environment for users. The app will include chat rooms created by the developers based on popular topics or interests, and users can participate in discussions. The app will also include a review mining module to analyze user feedback and improve the app's functionality.

2. SYSTEM ENVIRONMENT

2. SYSTEM ENVIRONMENT

System/Software environment is the term commonly used to refer to support an application. A software environment for a particular application could include the operating system, the database system, specific development tools or compiler.

2.1 Software configuration

Operating system: Windows 7 or above

Technology: Flutter

Language: Dart

IDE: Android Studio

Framework: Flutter

Back End: Firebase

Database: Firestore

2.1.1 Flutter

Flutter is an open-source UI software development kit created by Google .it is used to develop cross platform applications for Android, iOS, Mac, Windows, Google Fuchsia, and the web from a single codebase. The first version of Flutter was known as codename "sky" and ran on the Android operating system.

In 2015 Google unveiled Flutter, a new SDK based on Dart language as the next platform for Android development, and in 2017 May, an alpha version of it (0.0.6) was released to the public for the first time. In December 2018 Flutter 1.0 was released and made available so that developers could begin using the SDK to make app creation easier.

2.1.2 Dart

Dart is a programming language designed for client development, such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications. Dart is an object oriented, class-based, garbage-collected language with C-style syntax.it can compile to either native code or JavaScript. It supports interfaces, abstract classes, reified generics, and type inference.

2.1.3 Android Studio

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains IntelliJ IDEA software and designed specifically for Android development.

2.1.4 Firebase

Firebase is a cloud-based platform for mobile and web app development that offers a range of tools and services such as real-time databases, cloud storage, authentication, analytics, hosting, and more. It is a serverless platform, meaning developers do not need to manage the infrastructure. Firebase is popular due to its ease of use, scalability, and reliability.

2.1.5 Firestore

Firestore is a NoSQL document database offered as a part of Firebase. It is a fully-managed, serverless database designed to store and query large amounts of data in real-time. Firestore uses a flexible data model that can handle complex, hierarchical data structures and provides support for atomic operations and transactions. It also offers real-time updates and offline synchronization, making it ideal for mobile and web applications.

2.2 Hardware configuration

Processor: i3 or above.

System Bus: 32Bit or 64Bit

RAM: 4 GB or Above

HDD: 500 GB or Above

Mobile: Android supported mobile phone Software Requirements

3. SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

In analysis, a detailed study of these operations performed by a system and their relations within and outside of the system is done. During analysis, data are collected on available files, decision points and transaction handled by present system. Once analysis is completed, the analyst has firm understanding of what is to be done. Based on analysis, a cost /benefit analysis is considered. Here the cost for the system is calculated on the basis of analysis .The solutions are given as proposal. The proposal is then weighed with existing system analytically and best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on the user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

3.1 Existing system

The existing system for a location-based anonymously chatting app is a collection of different apps and platforms that offer similar features. Some of the most popular apps in this category include Whisper, Yik Yak, and Omegle.

Whisper is a social networking app that allows users to share their thoughts, secrets, and emotions anonymously with others in their local area. The app uses the user's location data to connect them with other users nearby, and users can choose to chat with others one-on-one or join public chat rooms.

Yik Yak was another popular location-based anonymous social networking app that allowed users to post messages and interact with others in their local area. However, the app was shut down in 2017 due to issues with cyberbullying and harassment.

Omegle is a web-based chat platform that connects users with strangers from around the world for anonymous text or video chat. While the app is not location-based, it allows users to remain anonymous and chat with others without revealing their identities.

Overall, the existing system for a location-based anonymously chatting app is no longer available for users to download and use. Additionally, some of the existing apps that offer similar features may not be user-friendly due to having too many additional features. These extra features may make it difficult for users to navigate and fully utilize the app's intended purpose. This can result in a poor user experience and potentially deter users from using the app altogether. Therefore, there is a need for a more streamlined and user-friendly location-based anonymously chatting app.

3.2 Proposed system

The proposed system for the location-based anonymously chatting app aims to address the drawbacks of the existing system and provide a more user-friendly platform for anonymous chatting between users in the same location. The proposed system will have a simple and intuitive user interface that allows users to quickly and easily connect with others in their local area. Users will be able to set their location and browse nearby chat rooms or create their own. The system will allow for anonymous chatting, allowing users to communicate freely without revealing their real identities. To ensure the safety and security of users, the proposed system will implement strict moderation and reporting features. Users will be able to report any inappropriate behavior or content, and moderators will be able to review and take appropriate action to maintain a safe and respectful community.

Additionally, the proposed system will incorporate machine learning and natural language processing algorithms to improve the chat experience for users. These algorithms will help identify and remove spam messages, detect and flag potentially offensive language, and provide personalized recommendations for users based on their chat history and preferences.

Overall, the proposed system for the location-based anonymously chatting app will provide a more user-friendly platform for anonymous chatting between users in the same location, while also prioritizing safety, security, and a positive user experience.

The proposed system contains following modules,

User

- Login

- Create and manage profile

- Search for nearby users

- Start a chat with nearby users anonymously

- Create and join chat rooms

- Send and receive messages

- Report inappropriate behavior or content

- View own chat history

Administrator

- Login

- View and manage reported content

- View and manage user accounts

- View statistics and analytics

- Manage the application settings

3.3 Feasibility study

All projects are feasible when given unlimited resources and infinite time. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. An estimate is made of whether the identified user needs may be satisfied using current software and hardware technologies. The study will decide if the proposed system will be cost effective from the business point of view and if it can be developed in the given existing budgetary constraints. The feasibility study should be relatively cheap and quick. The result should inform the decision of whether to go ahead with a more detailed analysis.

The feasibility study is done in these phases

- Operational feasibility
- Behavioural feasibility
- Technical feasibility
- Economic feasibility.

3.3.1 Technical Feasibility

Technical feasibility centres on the existing system (hardware, software etc) and to what extent it can support the proposed addition. This system must be evaluated from technical viewpoint first. The assessment of this feasibility must be based on the outline design of the system requirements in the terms of inputs, outputs program procedure. The minimum

requirement for this proposed system is a system, an internet connectivity and proper input or output devices. The existing system has all these requirements. So the proposed system is said to be technically feasible.

3.3.2 Behavioural Feasibility

People are inherently resistant to change and computers have known to facilitate change. An estimate should be made of how strong the reaction the user staff is likely to have towards the development of the system. A behavioural feasibility study is conducted among the top staffs using direct interviews and questionnaire.

3.3.3 Economic Feasibility

Economic Feasibility is the most frequently used method for evaluating the effectiveness of the proposed system. If the benefits of the proposed system outweigh the cost, then the decision is made to design and implement the system.

- The cost of hardware and software is affordable.

- High increase in the amount of profit earned by going global.

- Easy and cheap maintenance of the system possible.

3.3.4 Operational Feasibility

A system that has operational feasibility is the one that will be used effectively after it has been developed. If users have difficulty with a new system, it will not produce the expected benefits. If the system meets the requirements of the administrator we can say that the system is operationally feasible. The proposed system will be beneficial only if it can be turned into a system which will meet the requirements of the store when it is developed and installed, and there is sufficient support from the users. The proposed system will really benefit the organization. Then proposed system will improve the total performance. Hence, the proposed system is operationally feasible.

3.4 Project planning

Software project management is a art and science of planning and leading software projects.it is a sub discipline of project management in which software project are planned, monitored and controlled. To manage new development efforts, companies applied proven project management methods, but project schedules slipped during test runs, especially when confusion occurred in the grey zone between the user specification and the delivered software. To be able to avoid the problems, software project managements methods focused on matching user requirements to deliver products, by using waterfall model, science then, analysis of software project management failures has shown that the following the most common cause are:

- Unrealistic or unarticulated project goal
- Inaccurate estimates of needed resources
- Badly defined system requirements
- Poor reporting of the project's status
- Unmanaged risks
- Use of immature technology
- Inability to handle the project complexity
- Poor project management

3.5 Project scheduling

Project scheduling is a tool that communicates what work need to be performed, which resources of an organization will performed the work and the timeframes in which that work needs to be performed the project schedule should reflect all of the work associated with delivering the project on time without a full and complete schedule, the project manager will be unable to communicate the complete effort, in terms of cost and resourced8, necessary to deliver the project. The whole project is scheduled for four months period. Touch based digital ordering system is successfully completed within this time duration.
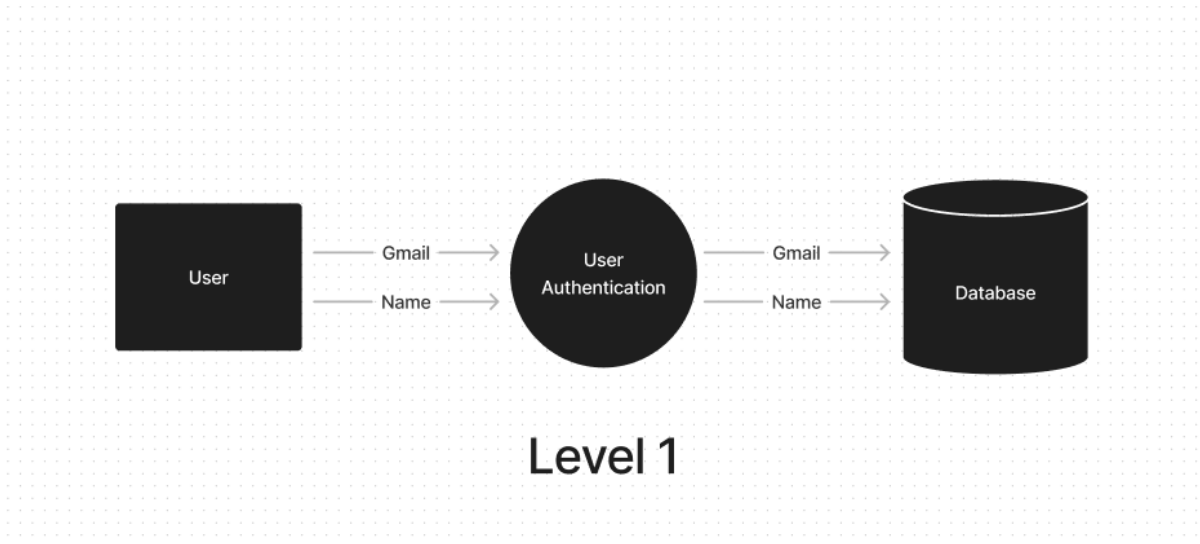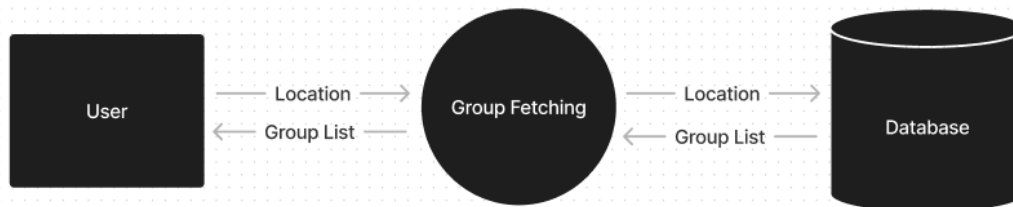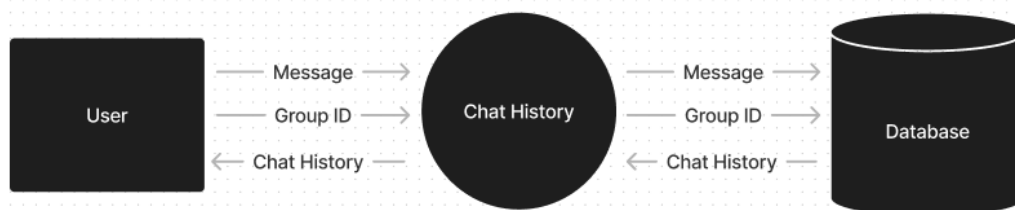
3.6 Data flow diagram

Level 0
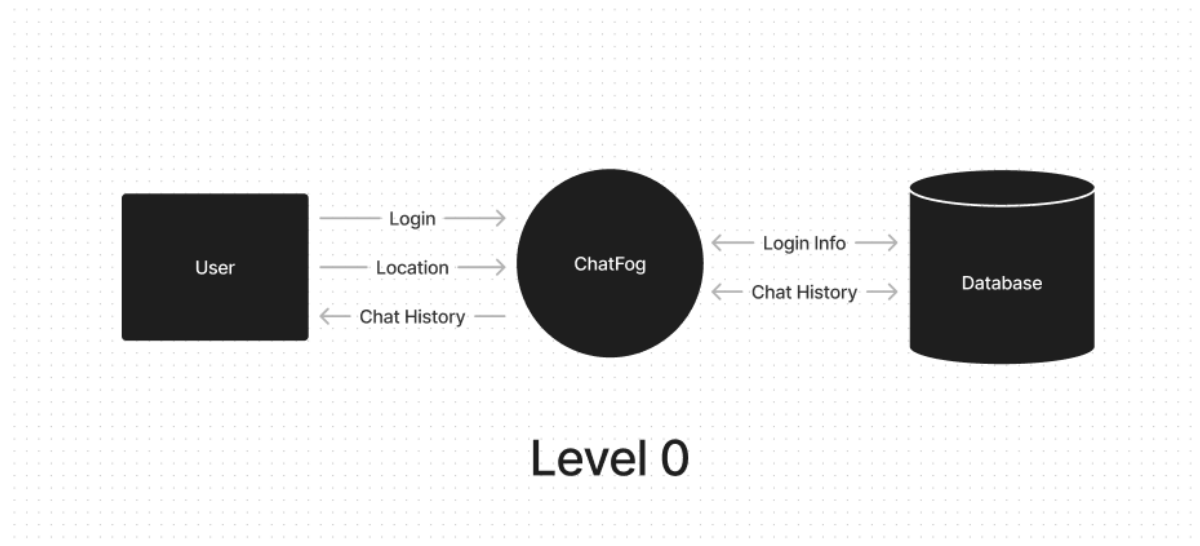
Level 1

Level 2

Level 3

4. SYSTEM DESIGN

4. SYSTEM DESIGN

4.1 Input design

Input design is the process of converting user-originated inputs to a computer based format. The goal of designing input data is to make data entry easy, logical and free from errors as far as possible.

The system input is made up of new incoming queries and the current set of indexes, which is initialized to be the suggested indexes from the output of the initial index selection algorithm. In this system the user's data are collected from various ways, such as through textbox and list boxes.

The system will consist of a mobile based graphical user interface (GUI) with all functions accessible by a virtual keyboard and hand gestures. Once logged on to the system the interface will consists of a single screen.

When the application is initiated, the user can login as either as a hotel or a customer. After this, the login screen; User needs to supply user id and password. Based on the user, the system will be able to determine which functionalities that are accessible to the user are.

After successful login of the user, the user will be led to the next activities which enable them to use the application's facilities. The application reads the security options for the user from the live database, and based on them, options will be enabled to the user

4.2 Output design

Output generally refers to the result and information for developing the system. For many end-users, output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application. Computer output is the most important and direct source of information to the user. In this system the output is reported to the administrator as a set of queries, which gives the detailed information about the previous actions. The immediate feedback for the user actions also provided through the output design Output design produces the hardcopy regarding the information requested or displays the output in a predefined format. It is the direct source of information to the end user. Efficient and intelligible outputs improve the system's relationships with the users and help in decision and design making. The nature of processing and procedure related to the system were

classified and gives the output results. While designing the output, the type of the output, concern format, frequency responses have been taken into consideration.

4.3 Process design

Software design is both a process and a model. This project there are 5 modules. First process is registration, after registration customer can login to the system. After login customer can view and order menu item, request home delivery, request dine-in, view and pay bills, add feedback, add complaint and view reply and add rating.

4.4 Database design

1. Chat Room

Purpose: To store chat room details

Primary key: id

Fields	Type	Size	Description
Id	Int	11	Manager id
Name	varchar	50	Manager name

2. User

Purpose: to store user details

Primary key: mail

Fields	Type	Size	Description
Mail	varchar	50	user mail
Name	varchar	50	user name

5. TESTING AND IMPLEMENTATION

5. TESTING AND IMPLEMENTATION

Implementation phase is the phase, which involves the process of converting a new system design into an operational one. It is the key stage in achieving a successful new system. Implementation is the stage if the project, where the theoretical design is turned into a working system. At this stage the main workload, the greatest up heal and the major impact on existing practices shift to user department. If the implementation stage is not planned and controlled carefully, it can cause chaos. The implementation stage is a system project in its own right. It involves careful planning, investigation of the current system and its constraints on the implementation, design methods to achieve the changeover procedures, and evaluation of change over methods. The implementation plan consists of the following steps,

- Testing the developed system with the sample data.

- Detection and correction of errors.

- Making necessary changes in the system.

- Training and involvement of user personnel.

Software testing is critical element of software quality assurance and represent the ultimate review of the specification, design and coding. System testing makes a logical assumption that all the part of the system is correct; the goal will be successfully achieved.

5.1 Testing Techniques

- Unit testing
- Integration testing
- Validation Testing
- Recovery testing
- Source code testing
- Output testing

5.1.1 Unit Testing

In this different test modules are tested against the specification of the modules. Unit testing was done for the verification of the code produced during the coding phase and to test the internal logic or modules. It refers to the verification of the single program module in

installed environment. Unit testing is performed on each program to ensure that the program has been built according to the program specifications and performs the desired actions.

5.1.2 Integration Testing

In this project the modules are integrated properly, the emphasis being and testing interfaces between modules .internal and external interfaces are tested as each module is incorporated into the structure . This test is designed to uncover errors associated with local are global data structures are conducted. It is also designed to verify performance levels established during software design are conducted Thus all these modules are combined, verified and the information about the items is properly carried on to the next module and then it is checked.

5.1.3 Validation Testing

At the culmination of integration testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected, and a final series of software tests validation testing may begin. Validation can be defined in many ways but a simple definition is that validation succeeds when software function in a manner that can be reasonably expected by the customer.

5.1.4 Recovery Testing

It is the activity of testing how well an application is able to recover from crashes, hardware failures and other similar problems. Recovery testing is the forced failure of the software in a variety of ways to verify that recovery is properly performed.

5.1.5 Source code Testing

This examines the logic of the system. If we are getting the output that is required by the user, then we can say that the logic is perfect

5.1.6 Output Testing

After performing the validation testing, the next step is output testing of Advanced ATM application system. Since no system would be termed as useful until it produce the required output in the specified format.

6. CONCLUSION

6. CONCLUSION

The proposed project provides maximum user interaction and flexibility. The testing of the whole system is completed with the data and the outputs generated are according to the requirement. The efficiency of the new system is found to be better than the existing system in all respects. The system is able to maintain the details in an effective manner. The programming techniques used in the design of the system provide a scope for further enhancement in the future. The various reports generated by the system are also useful.

7. SCOPE FOR FTURE DEVELOPMENT

7. SCOPE FOR FUTURE DEVELOPMENT

The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Changing the existing module or adding new module can append improvements.

These below modules are kept as future enhancement for implementation

- Integration with social media platforms

- Improved user interface and user experience

- Enhanced security and privacy features

- Gamification and rewards system

New modules can be added into existing system with less effort. The software can be given new futures and can be added thereby improving the efficiency of entire system. Moreover, in future the presented system can be upgraded to an Android system.

8. APPENDIX

    I.   System source code

   II.   GUI screen shots

  III.   System minimum Requirement

1. Coding

```dart
import 'package:chatfog/ui/login.dart';

import 'package:firebase_core/firebase_core.dart';

import 'package:flutter/material.dart';

import 'package:chatfog/ui/auth_service.dart';


void main() async {

  WidgetsFlutterBinding.ensureInitialized();

  await Firebase.initializeApp();

  runApp(const MyApp());

}


class MyApp extends StatelessWidget {

  const MyApp({super.key});


  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      theme: ThemeData(useMaterial3: true),
```

```
      home: AuthService().handleAuthState(),

    );

  }

}


import 'package:chatfog/ui/home.dart';

import 'package:chatfog/ui/login.dart';

import "package:firebase_auth/firebase_auth.dart";

import 'package:flutter/cupertino.dart';

import 'package:google_sign_in/google_sign_in.dart';


class AuthService {

  handleAuthState() {

    return StreamBuilder(

      stream: FirebaseAuth.instance.authStateChanges(),

      builder: (BuildContext context, snapshot) {

        if (snapshot.hasData) {

          return HomePage();

        } else {

          return const LoginPage();

        }

      });

  }
```

```
signInWithGoogle() async {

  // Trigger the authentication flow

  final GoogleSignInAccount? googleUser =

      await GoogleSignIn(scopes: <String>["email"]).signIn();



  // Obtain the auth details from the request

  final GoogleSignInAuthentication googleAuth =

      await googleUser!.authentication;



  // Create a new credential

  final credential = GoogleAuthProvider.credential(

    accessToken: googleAuth.accessToken,

    idToken: googleAuth.idToken,

  );



  // Once signed in, return the UserCredential

  return await FirebaseAuth.instance.signInWithCredential(credential);

}



signOut() {

  FirebaseAuth.instance.signOut();

}

}
```

```dart
import 'package:chatfog/ui/home.dart';

import 'package:flutter/material.dart';

import 'package:flutter/src/widgets/framework.dart';

import 'package:flutter/src/widgets/placeholder.dart';


class Chat extends StatelessWidget {
  const Chat({super.key});


  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Group(
          rad: 20,
        ),
      ),
    );
  }
}



import 'package:chatfog/ui/chat.dart';

import 'package:flutter/material.dart';

import 'package:flutter/src/widgets/framework.dart';
```

```dart
import 'package:flutter/src/widgets/placeholder.dart';

import 'package:firebase_auth/firebase_auth.dart';


class HomePage extends StatelessWidget {
  const HomePage({super.key});


  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.black,
      appBar: AppBar(
        backgroundColor: Colors.black,
        centerTitle: false,
        title: Text(
          // 'ChatFog',
          FirebaseAuth.instance.currentUser!.displayName!,
          style: const TextStyle(
            color: Colors.white, fontSize: 30, fontWeight: FontWeight.bold),
        ),
      ),
      body: Column(
        children: [
          Row(children: [
            Column(
```

```dart
          children: [
            Container(
              margin: const EdgeInsets.only(top: 20, left: 20),
              height: 80,
              width: 80,
              decoration: BoxDecoration(
                color: Colors.white,
                borderRadius: BorderRadius.circular(50)),
              child: const Placeholder(),
            ),
            const Padding(
              padding: EdgeInsets.only(left: 20, top: 10),
              child: Text(
                'Scan',
                style: TextStyle(fontSize: 13, color: Colors.white),
              ),
            )
          ],
        ),
        SizedBox(
          height: 130,
          width: MediaQuery.of(context).size.width - 100,
          child: ListView.builder(
            scrollDirection: Axis.horizontal,
```

```
      physics: const BouncingScrollPhysics(),

      itemBuilder: (context, index) {

        return Column(

          mainAxisAlignment: MainAxisAlignment.center,

          crossAxisAlignment: CrossAxisAlignment.center,

          children: const [

            Myprofile(),

            Padding(

              padding: EdgeInsets.only(left: 20, top: 10),

              child: Text(

                'Name',

                style: TextStyle(fontSize: 13, color: Colors.white),

              ),

            )

          ],

        );

      },

      itemCount: 5,

    ),

  )

]),

const SizedBox(height: 15),

Container(

  width: MediaQuery.of(context).size.width * 1,
```

```
height: MediaQuery.of(context).size.height - 245,

padding: const EdgeInsets.only(top: 10),

decoration: const BoxDecoration(

 color: Colors.white,

 borderRadius: BorderRadius.only(

  topLeft: Radius.circular(30),

  topRight: Radius.circular(30),

 ),

),

child: ListView.separated(

 separatorBuilder: (context, index) {

  return const Divider(

   height: 30,

   thickness: 3,

   indent: 20,

   endIndent: 20,

   color: Color(0xffEFF0F1),

  );

 },

 itemBuilder: (context, index) {

  return InkWell(

   onTap: () {

    Navigator.push(context,

     MaterialPageRoute(builder: (context) => const Chat()));
```

```
            },

          child: Container(

            margin: const EdgeInsets.only(left: 20, right: 20),

            height: 80,

            width: MediaQuery.of(context).size.width - 40,

            decoration: const BoxDecoration(

              color: Colors.white,

            ),

            child: const Group(rad: 30,),

          ),

        );

      },

      itemCount: 10,

    ),

  )

],

),

);

}

}


class Group extends StatelessWidget {

  const Group({super.key, required this.rad});
```

```dart
  final double rad;

  @override

  Widget build(

   BuildContext context,

  ) {

   return Row(

    children: [

     CircleAvatar(

       backgroundColor: Colors.white,

       radius: rad,

       child: const Placeholder(),

     ),

     const SizedBox(width: 20),

     const Text(

       'Name',

       style: TextStyle(

         fontSize: 20, color: Colors.black, fontWeight: FontWeight.bold),

     )

    ],

   );

  }

}


class Myprofile extends StatelessWidget {
```

```
  const Myprofile({
    super.key,
  });


  @override
  Widget build(BuildContext context) {
   return const CircleAvatar(
      radius: 40,
      backgroundColor: Color(0xffFFCB45),
      child: CircleAvatar(
        radius: 37,
        backgroundColor: Colors.black,
        child: CircleAvatar(
          radius: 35,
          backgroundColor: Colors.white,
          backgroundImage: NetworkImage('https://picsum.photos/200/300')),
      ));
  }
}


import 'package:chatfog/ui/auth_service.dart';

import 'package:chatfog/ui/home.dart';

import 'package:flutter/material.dart';

import 'package:flutter/src/widgets/framework.dart';
```

```dart
import 'package:flutter/src/widgets/placeholder.dart';

class LoginPage extends StatelessWidget {
  const LoginPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: Column(children: [
        Container(
          height: MediaQuery.of(context).size.height * 0.5,
          width: MediaQuery.of(context).size.width * 1,
          color: const Color(0xffFFCB45),
          child: Image.asset(
            'assets/image/home.png',
            width: MediaQuery.of(context).size.width * 1,
            alignment: Alignment.bottomCenter,
          ),
        ),
        Container(
          height: MediaQuery.of(context).size.height * 0.5,
          width: MediaQuery.of(context).size.width * 1,
          color: Colors.black,
```

```
child: Column(

  crossAxisAlignment: CrossAxisAlignment.start,

  children: [

    Container(

      padding:

        const EdgeInsets.only(top: 50, left: 25, right: 25),

      child: const Text(

        'Welcome to ChatFog',

        textAlign: TextAlign.left,

        style: TextStyle(

          color: Color(0xffFFCB45),

          fontSize: 30,

          fontWeight: FontWeight.bold),

      ),

    ),

    Container(

      margin: const EdgeInsets.only(top: 20),

      padding: const EdgeInsets.symmetric(horizontal: 25),

      child: const Text(

        'Connect with people around you anonymously and safely',

        style: TextStyle(

          color: Colors.white,

          fontSize: 26,

          fontWeight: FontWeight.bold),
```

```
        ),
      ),
      SizedBox(height: 40),
      Center(
        child: Container(
          margin: const EdgeInsets.only(top: 20),
          width: MediaQuery.of(context).size.width - 100,
          alignment: Alignment.center,
          child: ElevatedButton(
            onPressed: () {
              AuthService().signInWithGoogle();
              Navigator.pushReplacement(
                context,
                MaterialPageRoute(
                    builder: (context) => const HomePage()));
            },
            style: ElevatedButton.styleFrom(
                backgroundColor: const Color(0xffFFCB45),
                // padding: const EdgeInsets.symmetric(
                //     horizontal: 20, vertical: 20),
                textStyle: const TextStyle(
                    fontSize: 18, fontWeight: FontWeight.bold)),
            child: Row(
              mainAxisAlignment: MainAxisAlignment.spaceAround,
```

```
          children: [

            Image.asset(

              'assets/image/google.png',

              width: 30,

            ),

            const Text('Continue with Google'),

          ],

        ),

      ),

    ),

   ],

  ))

 ]));

 }

}
```
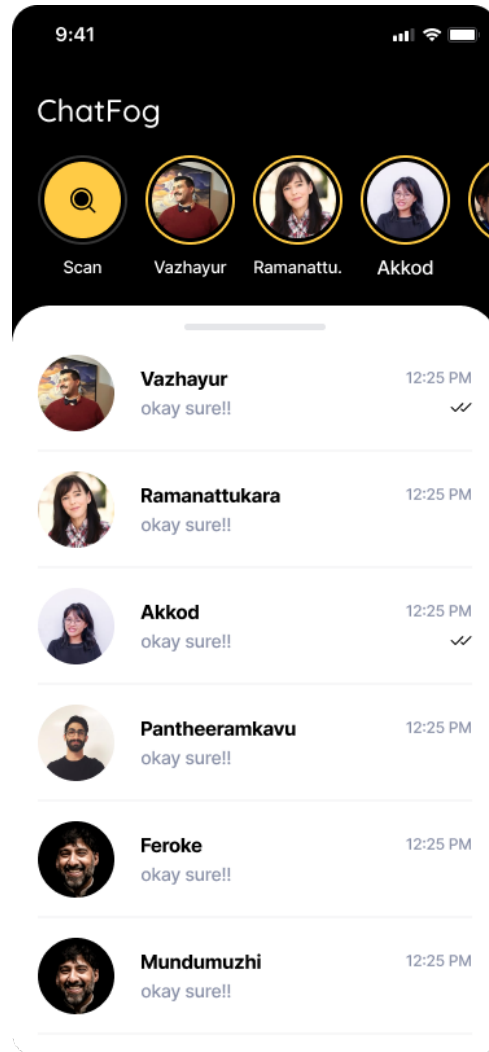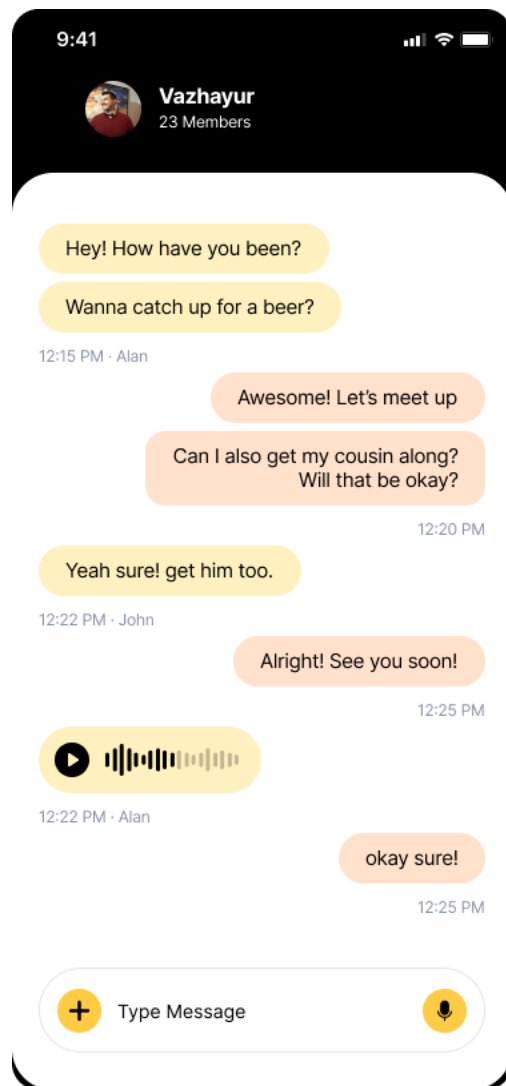
1. GUI Screen shots

Registration and Login   Home

Chat

2. System minimal Requirement

Software configuration

Operating system: Windows 7 or above

Technology: Flutter

Language: Dart

IDE: Android Studio

Framework: Flutter, Node.js expresses

Back End: Node.js

Database: MySQL

Hardware configuration

Processor: i3 or above.

System Bus: 32Bit or 64Bit

RAM: 4 GB or Above

HDD: 500 GB or Above

Mobile: Android supported mobile phone Software Requirements

9. REFERENCES

9. REFERENCES

1. Website References

1. https://flutter.dev/
2. https://firebase.google.com/docs
3. https://www.londonappbrewery.com/