A Thesis/Project/Dissertation Report

on

Secure Messenger using flutter

Submitted in partial fulfillment of the

requirement for the award of the degree of

B. Tech

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of

Mr. P. Raja Kumar
Assistant Professor

Submitted By

Nikhil Rana (18SCSE1010455)
Ayush Sinha (18SCSE1010441)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING / DEPARTMENT OF COMPUTERAPPLICATION

GALGOTIAS UNIVERSITY, GREATER NOIDA

INDIA

12/2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"secure messenger using flutter"** in partial fulfillment of the requirements for the award of the B. Tech CSE-submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Name… Designation, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Nikhil Rana (18SCSE1010455)
Ayush Sinha (18SCSE1010441)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

P. Raja Kumar
Assistant Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Nikhil Rana (18SCSE1010455) and Ayush Sinha (18SCSE1010441)

has been held on 17/12/2021 and his/her work is recommended for the award of B. Tech CSE

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: December 2021

Place: Greater Noida

Abstract

Paragraph-1 Existing Problem- 5-8 Lines

Paragraph-2 Proposed Solution- 6-8 Lines

Paragraph-3 Tools and Technology Used- 3-5 lines

Paragraph -4 Results and output- 3-5 lines

Paragraph-5 Conclusion and Future Scope – 2-4 Lines

Table of Contents

List of Table

List of Figures

Acronyms

B.Tech.	Bachelor of Technology
M.Tech.	Master of Technology
BCA	Bachelor of Computer Applications
MCA	Master of Computer Applications
B.Sc. (CS)	Bachelor of Science in Computer Science
M.Sc. (CS)	Master of Science in Computer Science
SCSE	School of Computing Science and Engineering

Abstract

Communication through internet is becoming vital these days. An online communication allows the users to communicate with other people in a fast and convenient way.

Instant messaging can be considered as a platform to maintain communication. The main objective of this paper is to present a software application for the launching of a real time communication between operators/users. The system developed on android and ios will enable the users to communicate with another users through text messages with the help of internet. The system requires both the device to be connected via internet.

Secure messenger is a messenger application that allows communication in a secured manner. A secured private messenger is needed to ensure data transmission between sender and recipient more secure. This app presents a secure Messenger architecture.

Technology and tools used in the project are flutter, dart, firebase and cryptography for encrypting and decrypting messages.

This Project provides a real time conversation which is time saving and hence is more effective than email. Unlike the old concept of one to one communication app also provides the facility of group or team discussions at a place.

Introduction

In the real world the communication plays a very vital role. People have been communicating with each other through various applications or mediums. In the beginning people communicated with each other using letters or other sources, as these mediums could take much time to deliver the content.

Cell phones are another medium of communication but the drawback is for any limited or small message which need to be passed to another user then phone call is not an ideal way. The developers then looked to implement a text-based communication which would allow an in instant communication service. Instant messaging is an internet service that allows people to communicate with each other in real time through an instant messaging software. Unlike e-mail, instant messaging allows messages from one person to appear right away on the other person's mobile screen.

Android is an operating system for mobiles which was developed by google. This operating system allows the applications to be used on mobiles. As it was developed by google, android users can develop mobile applications and can be sold through android application stores such as play store.

Firebase is a NoSQL database which make use of sockets which allows the users to store and retrieve the data from the

database. It is an internet based instant messaging application which provides the user to communicate with other users in a fast and convenient way. Both the devices must have an active internet connection for the communication.

In addition, here’s what else the current state of chat app market has to reveal:

- Around 2.3 billion users use one of the messaging apps every day but Facebook alone has 2.5 billion monthly users worldwide whereas Instagram has been followed by 1 billion users.
- Monthly active users of the top 3 chat apps including WhatsApp, Facebook Messenger and WeChat have around 4 billion active monthly users.
- According to the survey, the popular chatting apps like Facebook Messenger and WhatsApp are having 73% of users from 18–29 age group, whereas 66% of users are from 30–59 and 43% are 60+.
- The forecast from the report reveals that the number of chat app (WhatsApp) users in the US will jump from 22 million in 2018 to 25.6 million in 2021.

Literature Survey

This part will discuss some theories and the literature reviews (related to digital technologies) incorporated in this study.

- Mobile Learning and Instant Messaging Development

Over the years, mobile learning and instant messaging are one of the fastest growing technologies which are used by numerous individuals, companies and educational institutions. Educational institutions like colleges and universities have discovered the ability of instant messaging tools to be useful tools within collaborative learning environment, and the fact that the tools allow learning to occur anytime and anywhere.

- Chatting Application

Chatting is a feature or a program on the Internet to communicate directly with internet users who are online together. This communication can be in a form of text (text chat) or voice (voice chat). Chatting is not only popular with teenagers or young people, but, it has also spread to older people. With using chat, we can freely talk about anything ranging from friendships, jobs, lessons in school, courses, even to the personal matters.

- Android

Android is an OS software for Smartphone. Android is an open

source project called Android Open Source Project (AOSP). Google uses this project as the basis for creating their own Android version. As an OS, the task of Android is to be a translator between the user and the device.

- Firebase

Firebase is BaaS (Backend as a Service) own by Google. Firebase offers solution to facilitate Mobile Apps Developer job. With Firebase, apps developer can focus to develop an application without worrying about backend issues.

Objective of Project on Online Chat Application:

The main objective of the Project on Online Chat Application is to manage the details of Chat Profile, Chat User, Chat History, Group Chat, Smilies Chat. It manages all the information about Chat Profile, Multi Chat, Smilies Chat, Chat Profile.

The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Chat Profile, Chat User, Multi Chat, Chat History and when the user will type a message the receiver will be able to see that message while sender is typing it . It tracks all the details about the Chat History, Group Chat, Smilies Chat.

Functionalities provided by Online Chat Application are as follows:

- Provides the searching facilities based on various factors. Such as Chat Profile, Chat History, Group Chat, Smilies Chat
- Online Chat Application also manage the Multi Chat details online for Group Chat details, Smilies Chat details, Chat Profile.
- It tracks all the information of Chat User, Multi Chat, Group Chat etc
- Manage the information of Chat User
- Shows the information and description of the Chat Profile, Chat History
- To increase efficiency of managing the Chat Profile, Chat User
- It deals with monitoring the information and transactions of Group Chat.
- Manage the information of Chat Profile.

Chat User Module :

The main objective for developing this module is provide all the functionality realted to chat user. It tracks all the information and details of the chat user. We have developed all type of CRUD (Create, Read, Update and Delete) operations of the chat user. This is a role based module where admin can perform each and every operations on data but the chat user will be able to view only his/her data, so access level restrictions has also been implemented on the project.

Features of Chat User Module:

- Admin can add new chat user
- Admin can see the list of chat user details
- Only admin can edit and update the record of the chat user
- Admin will be able to delete the records of the chat user
- All chat user forms are validated on client side using JavaScript
- chat user will be able to see his details
- chat user will be able to update his details

Functionality performed by Project Online Chat Application:

These are the functionality performed by Project.

- Login For Admin
- Forgot password for Admin
- Edit Profile For Admin
- Change Password For Admin
- Logout Functionality
- Dashboard for Admin user

Manage Chat User

- Adding New Chat User
- Edit the Exiting Chat User
- View details of the Chat User
- Listing of all Chat User

Manage Chat Profile

- Adding New Chat Profile
- Edit the Exiting Chat Profile
- View details of the Chat Profile
- Listing of all Chat Profile

Manage Chat History

- Adding New Chat History
- Edit the Exiting Chat History
- View details of the Chat History
- Listing of all Chat History

PROBLEM FORMULATION

Some people have a habit of writing very long messages or in formal messages we send whole message at once. So, while the other user will write the message on his/her chatbox it will be visible to the other user in light color so they wont have to read the whole text at once and they can read the text while the other user is typing it.

Tools used for implementation

- Flutter
- Dart
- Firebase
- Cryptography

Supported Operating System

WINDOWS

- Microsoft® Windows® 8/7/Vista (32 or 64-bit)
- i3 Processor Based Computer or higher
- 2 GB RAM minimum, 4 GB RAM recommended
- 400 MB hard disk space
- At least 1 GB for Android SDK, emulator system images, and caches
- Java Development Kit (JDK) 7
- Optional for accelerated emulator: Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality.

ANDROID

- Android OS 4.0 or above
- 1.2 Quad core Processor or higher
- 512MB RAM
- 300MB Disk space

Mac

We can also easily configured this project on Mac operating system.

Few Mobile chat Applications

This sections compactly introduce numerous of popular converse operations in the mobile request according to security and sequestration enterprises**.**

Viber:

Viber Media developed Viber software for instant messaging and Voice over IP (VoIP) handsets. Users can send photos, videos, and audio media messages in addition to instant messaging.Viber has recently backed up the quitting encryption of its network company, but the best in individual and corporate conversations where all participants are using the latest version of Viber 6.0 for Android, iOS or windows 10. meanwhile, in the Viber iOS app for iPhone and iPad, email attachment includes photos and animations that can be sent as a percentage of iOS Extension does not help to encrypt. Viber had privacy issues that include posting a friend without his or her understanding or putting him or her in a collection without his or her consent. In addition, the local garage is unsafe. It is not an open source that makes it difficult to explore.

WhatsApp:

WhatsApp, one of the most well-known messaging apps, has now enabled end-to-end encryption for its 1 billion users across all platforms. WhatsApp makes use of a portion of the Open Whisper security protocol, which provides a security-verification code that can be shared with a contact to guarantee that the chat is encrypted . It's difficult to agree in WhatsApp software because it's not open source, making it impossible to confirm the working procedure and match them with the work of the encryption protocol that was introduced of analysable password. Provide

suitable education about what are the safeguard they have to take to avoid cyber-crimes. Also, sometimes actions to be taken simply in case if they're victim. A network administrator adopted the provisions and policies which consists via the Network security.

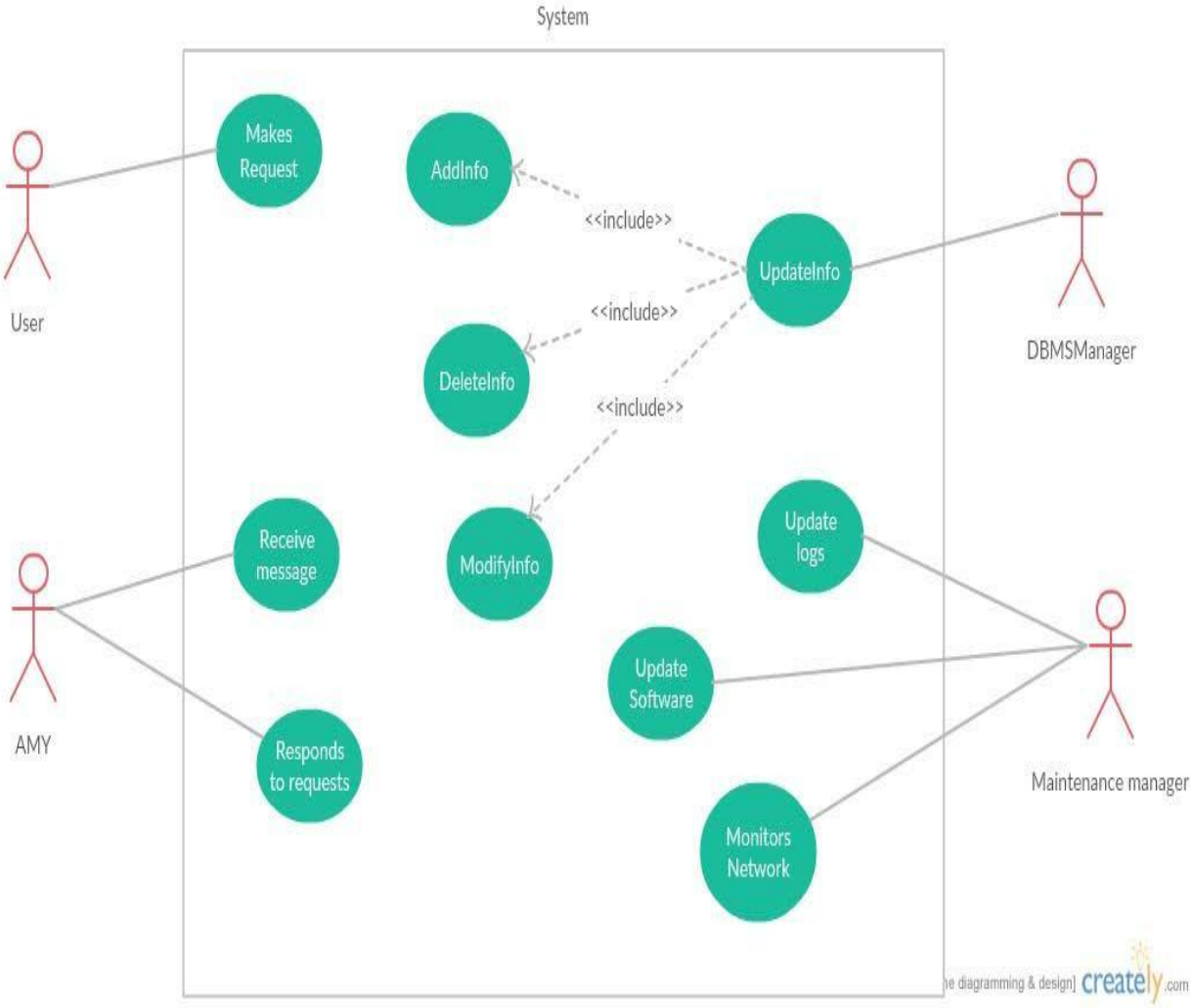
Telegram:

Telegram is a free instant messaging application that allows users to communicate messages, photos, videos, stickers, and files . Telegram has two communications modes: regular chat and secret chat. Normal chat is a client-server-based cloud-based messaging platform that does not provide stop-to-stop encryption, stores all communications on its servers, and synchronises with all consumer devices . In addition, the adjoining garage is not encrypted by default. End-to-end encryption is provided by secret conversation, which is patron-patron. Unlike regular chat messages, communications transmitted in a secret chat can only be accessed by the device that initiated the anonymous conversation, and a tool that has been accepted as an anonymous chat tool cannot be accessed by other devices.

Messages sent to anonymous discussions are self-destructive and can be erased at any time. Telegram employs the MTProto encryption system and has been chastised for its subscription support for Telegram, Viber, and WhatsApp. The Signaling System 7 (SS7) protocol is used to send SMS messages. The danger is in SS7. The attackers accessed the victims' accounts through SMS message using the SS7 protocol . Thanks to Telegram cloud-based, the attacker uses it and takes full control of the victim's account and can save it for protection if he has to start partial verification.

Facebook Messenger

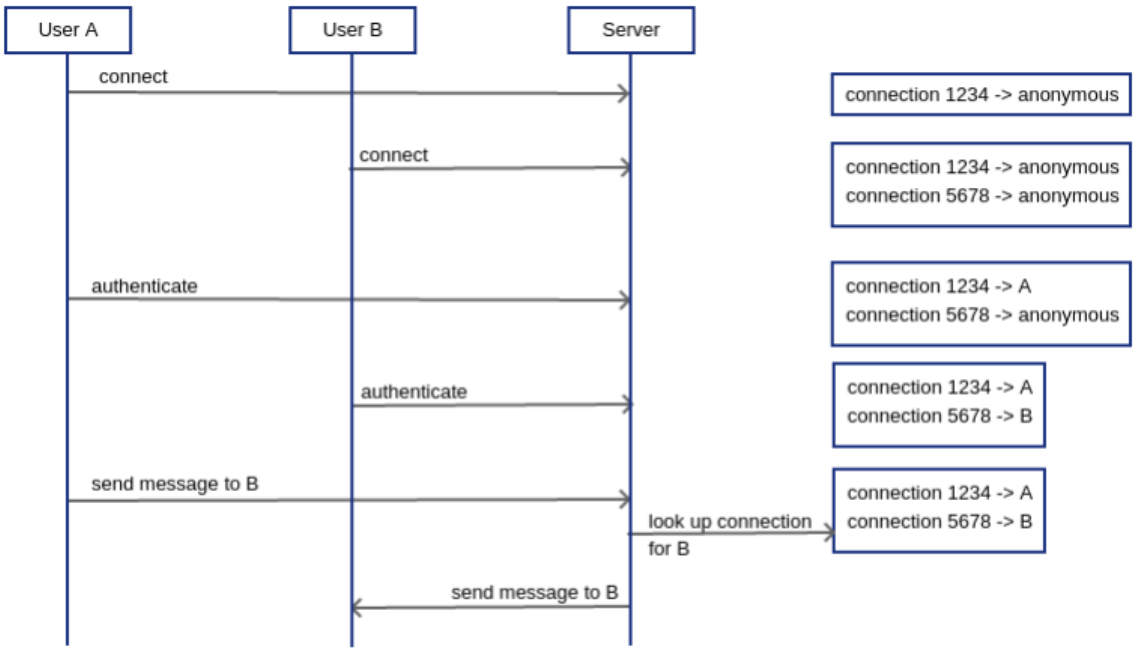
Facebook Messenger is a popular chat app for both Android and iOS devices. There are two message channels available: normal chat and private chat.Normal chat not only encrypts around secure secure communications using TLS, and stores all messages in its servers. Private chats have the same view of a private Telegram chat.


System
Makes Request
AddInfo
<<include>>
UpdateInfo
<<include>>
DeleteInfo
<<include>>
ModifyInfo
User
DBMSManager
Receive message
Responds to requests
AMY
Update logs
Update Software
Monitors Network
Maintenance manager
creately.com

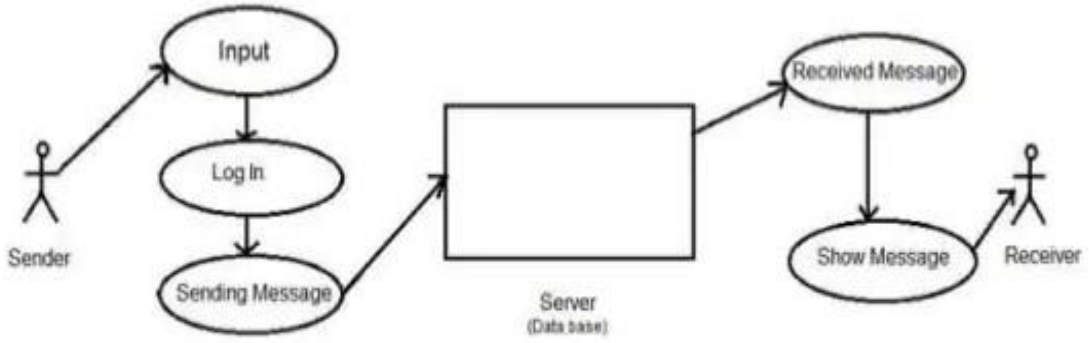
4 Major Steps to Develop Chat App in Flutter With Firebase

- Firebase Authorization: Sign in, Sign up or Sing Out
- Firebase Firestore for Installing Plugins: Upload, Remove and Retrieve Data Stored in Cloud Firestore
- Creating the Layout of the Chat App Screen
- Final Linking of the Flutter Chat App with Firebase

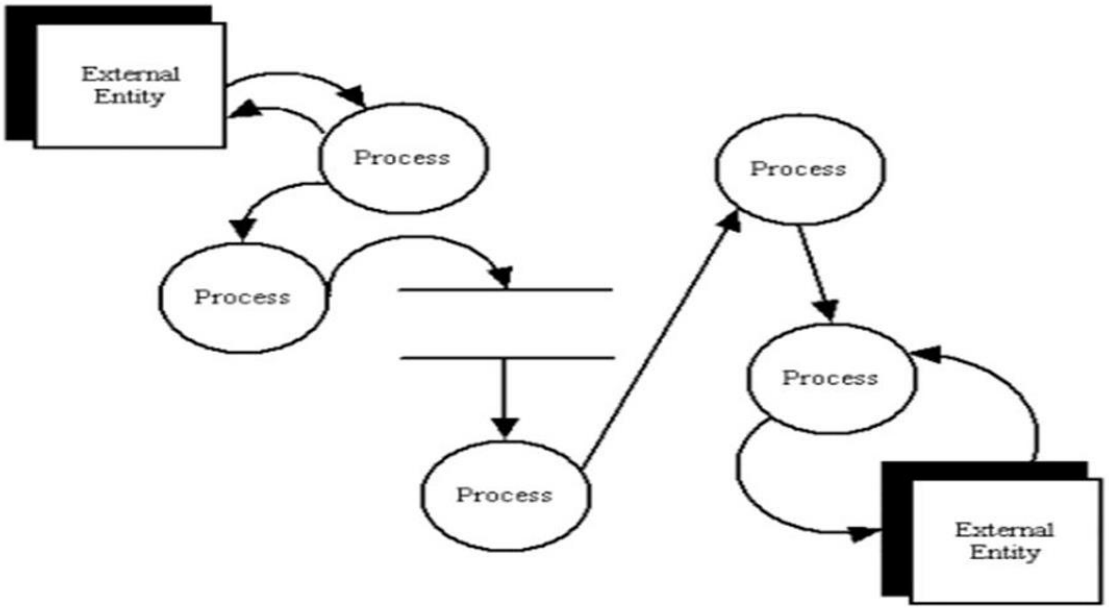
Process of chat system

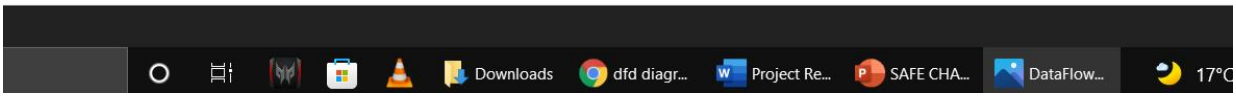


Data Flow Diagram



Data Flow Diagram





Future Scope:

With the knowledge I have gained by developing this application, I am confident that in the future I can make the application more effectively by adding this services.

- Extending this application by providing Authorisation service.
- Creating Database and maintaining users.
- Increasing the effectiveness of the application by providing Voice Chat.
- Extending it to app development Support.

Conclusion

The main objective of the project is to develop a Secure Chat Application. I had taken a wide range of literature review in order to achieve all the tasks, where I came to know about some of the products that are existing in the market. I made a detailed research in that path to cover the loop holes that existing systems are facing and to eradicate them in our application. In the process of research I came to know about the latest technologies and different algorithms.

People used to wait for their friend to send the whole message so they can read but now they can read it while they are typing so that they can save their time. As a result, hopefully the product will be successfully developed in terms of extendability , portability, and maintainability and tested in order to meet all requirements that are authentication, Integrity which are specified as the few basic concepts for the secure communication over a network.

Implementation of Project Modules

Code –

```
import 'dart:ui';

import 'package:firebase_core/firebase_core.dart';
import 'package:flash_chat/screens/chat_screen.dart';
import 'package:flash_chat/screens/login_screen.dart';
import 'package:flash_chat/screens/registration_screen.dart';
import 'package:flash_chat/screens/welcome_screen.dart';
import 'package:flash_chat/utilities/constants.dart';
import 'package:flutter/material.dart';

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp();
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
```

```
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Flash Chat',
      theme: ThemeData().copyWith(
          primaryColor: kBlueColor,
          textTheme: TextTheme(
            bodyText2: TextStyle(color: Colors.black),
          )),
      initialRoute: '/',
      routes: {
        '/': (context) => WelcomeScreen(),
        '/login_screen': (context) => LoginScreen(),
        '/registration_screen': (context) => RegistrationScreen(),
        '/chat_screen': (context) => ChatScreen(),
      },
    );
  }
}
```

Constants.dart code-

```
import 'package:flutter/material.dart';

const kBlueColor = Color(0xFF054BA6);
```

```
const kLightBlueColor = Color(0xFF0477BF);
const kPurpleColor = Color(0xFF5B61B9);
const kWhiteColor = Color(0xFFF7F7F8);

const kMessageTextFeildDecoration = InputDecoration(
  hintText: 'Type your message here...',
  hintStyle: TextStyle(
    color: Colors.grey,
  ),
  contentPadding: EdgeInsets.symmetric(
    vertical: 10.0,
    horizontal: 20.0,
  ),
);

const kMessageContainsDecoration = BoxDecoration(
  borderRadius: BorderRadius.all(
    Radius.circular(32.0),
  ),
);

const kTextFeildDecoration = InputDecoration(
  hintText: 'Enter a value',
  hintStyle: TextStyle(
```

```
  color: Colors.grey,
),
contentPadding: EdgeInsets.symmetric(
  vertical: 10.0,
  horizontal: 20.0,
),
border: OutlineInputBorder(
  borderRadius: BorderRadius.all(
    Radius.circular(32.0),
  ),
),
enabledBorder: OutlineInputBorder(
  borderSide: BorderSide(
    color: kBlueColor,
    width: 1.0,
  ),
  borderRadius: BorderRadius.all(
    Radius.circular(32.0),
  ),
),
focusedBorder: OutlineInputBorder(
  borderSide: BorderSide(
    color: kBlueColor,
    width: 2.0,
```

```
    ),
    borderRadius: BorderRadius.all(
      Radius.circular(32.0),
    ),
  ),
  prefixIcon: Icon(
    Icons.email,
    color: Colors.grey,
  ),
);
```

Welcome Screen.dart code-

```
import 'package:animated_text_kit/animated_text_kit.dart';
import 'package:flash_chat/components/round_button.dart';
import 'package:flash_chat/utilities/constants.dart';
import 'package:flutter/material.dart';

class WelcomeScreen extends StatefulWidget {
  @override
  _WelcomeScreenState createState() => _WelcomeScreenState();
}

class _WelcomeScreenState extends State<WelcomeScreen>
    with SingleTickerProviderStateMixin {
  AnimationController controller;
  Animation animation;

  @override
  void initState() {
    super.initState();

    controller = AnimationController(
      duration: Duration(
```

```
      seconds: 1,
    ),
    vsync: this,
  );

  animation = ColorTween(begin: Colors.blueGrey, end: Colors.white)
      .animate(controller);

  controller.forward();

  controller.addListener(() {
    setState(() {});
  });
}

@override
void dispose() {
  controller.dispose();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  return SafeArea(
```

```
child: Scaffold(
  backgroundColor: animation.value,
  body: Padding(
    padding: const EdgeInsets.symmetric(horizontal: 24.0),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      crossAxisAlignment: CrossAxisAlignment.stretch,
      children: [
        Flexible(
          child: Row(
            children: [
              Hero(
                tag: 'logo',
                child: Container(
                  child: Image.asset(
                    'images/logo.png',
                    height: 60.0,
                    fit: BoxFit.cover,
                  ),
                ),
              ),
              TypewriterAnimatedTextKit(
                text: ['Flash Chat'],
                speed: Duration(milliseconds: 300),
```

```
        textStyle: TextStyle(
          color: Colors.black,
          fontSize: 45.0,
          fontWeight: FontWeight.w900,
        ),
      ),
    ],
  ),
),
SizedBox(
  height: 48.0,
),
RoundedButton(
  buttonText: 'Log In',
  buttonColor: kLightBlueColor,
  onpress: () {
    Navigator.pushNamed(context, '/login_screen');
  },
),
RoundedButton(
  buttonColor: kBlueColor,
  buttonText: 'Register',
  onpress: () {
    Navigator.pushNamed(context, '/registration_screen');
```

```
            },
          ),
        ],
      ),
    ),
  ),
);
}
}
```

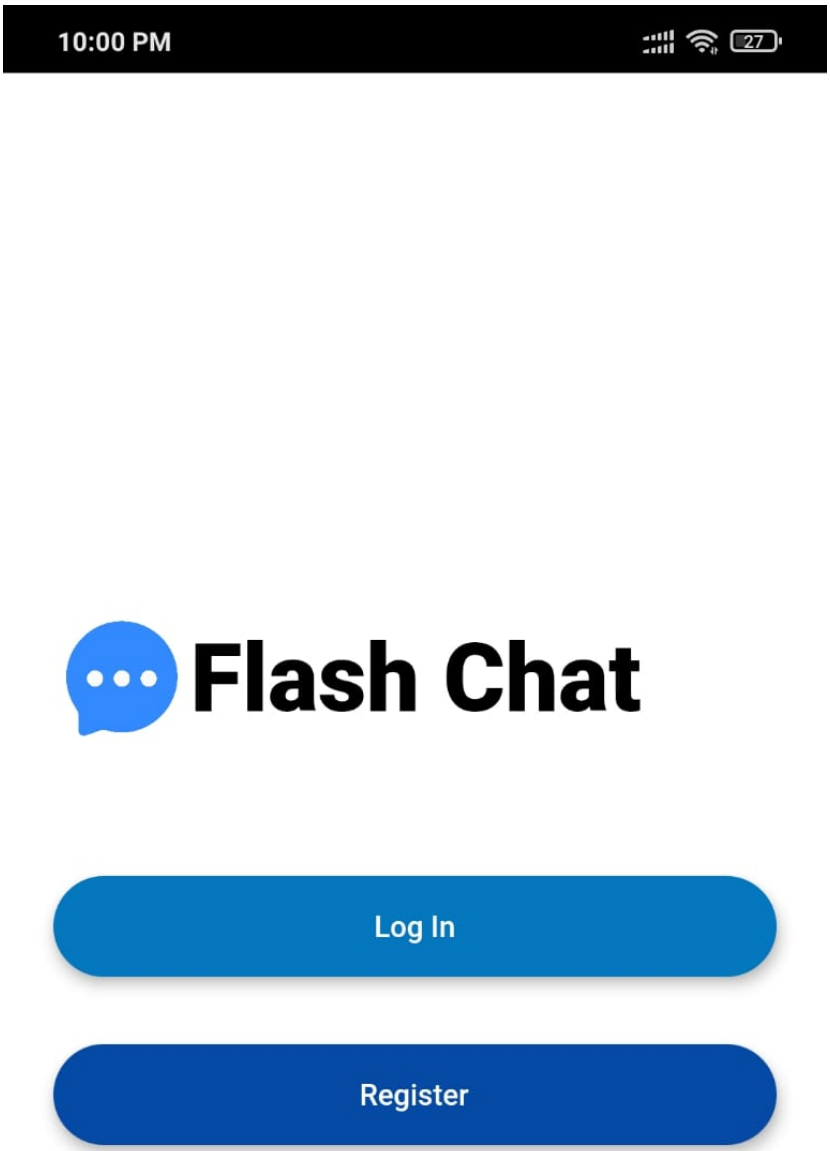
Pubspec.yaml code-

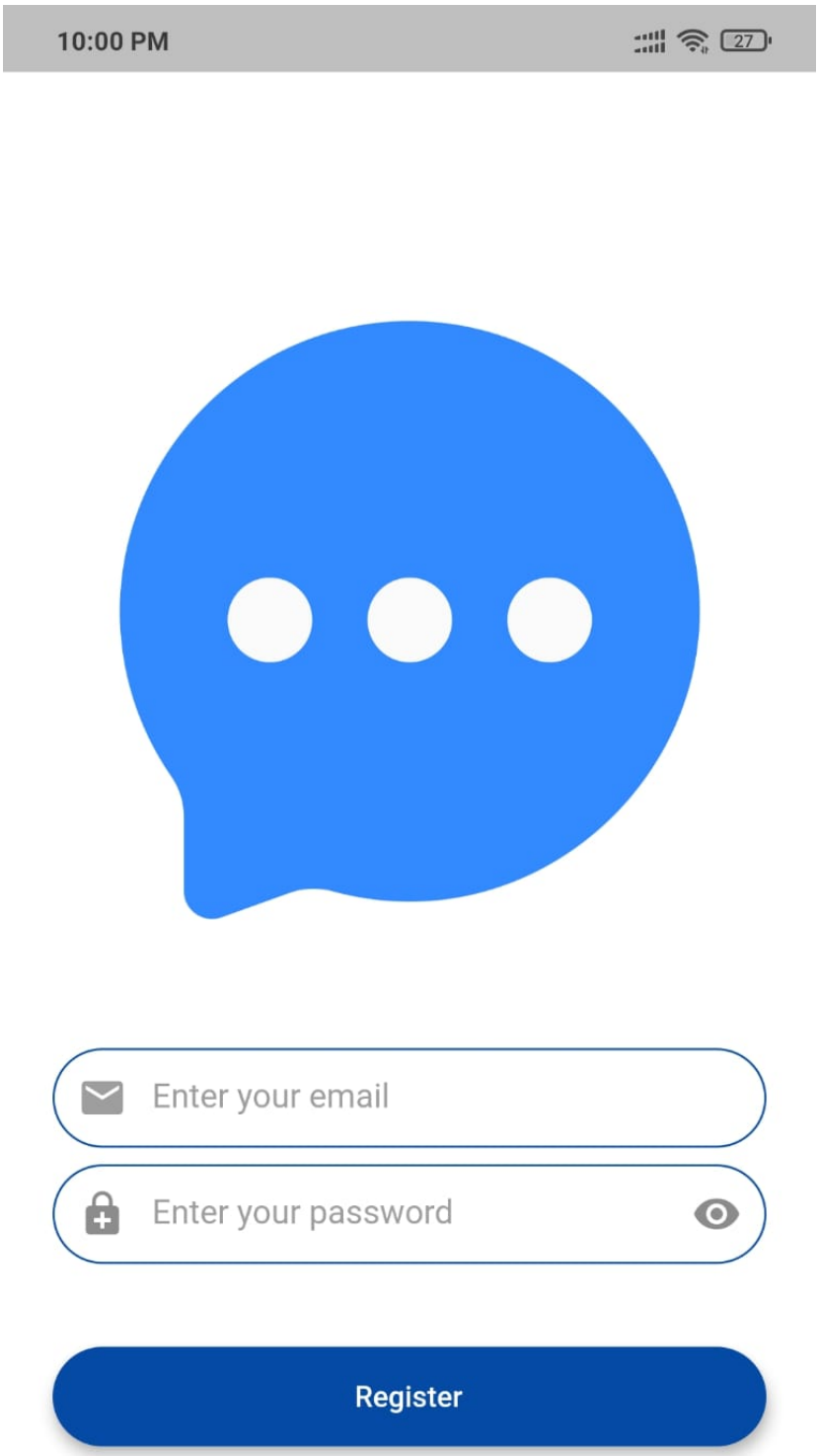
```
name: flash_chat

description: A new Flutter project.


# The following line prevents the package from being accidentally published to

# pub.dev using `pub publish`. This is preferred for private packages.

publish_to: 'none' # Remove this line if you wish to publish to pub.dev


# The following defines the version and build number for your application.

# A version number is three numbers separated by dots, like 1.2.43

# followed by an optional build number separated by a +.

# Both the version and the builder number may be overridden in flutter

# build by specifying --build-name and --build-number, respectively.

# In Android, build-name is used as versionName while build-number used as versionCode.

# Read more about Android versioning at https://developer.android.com/studio/publish/versioning

# In iOS, build-name is used as CFBundleShortVersionString while build-number used as CFBundleVersion.

# Read more about iOS versioning at

# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html
```

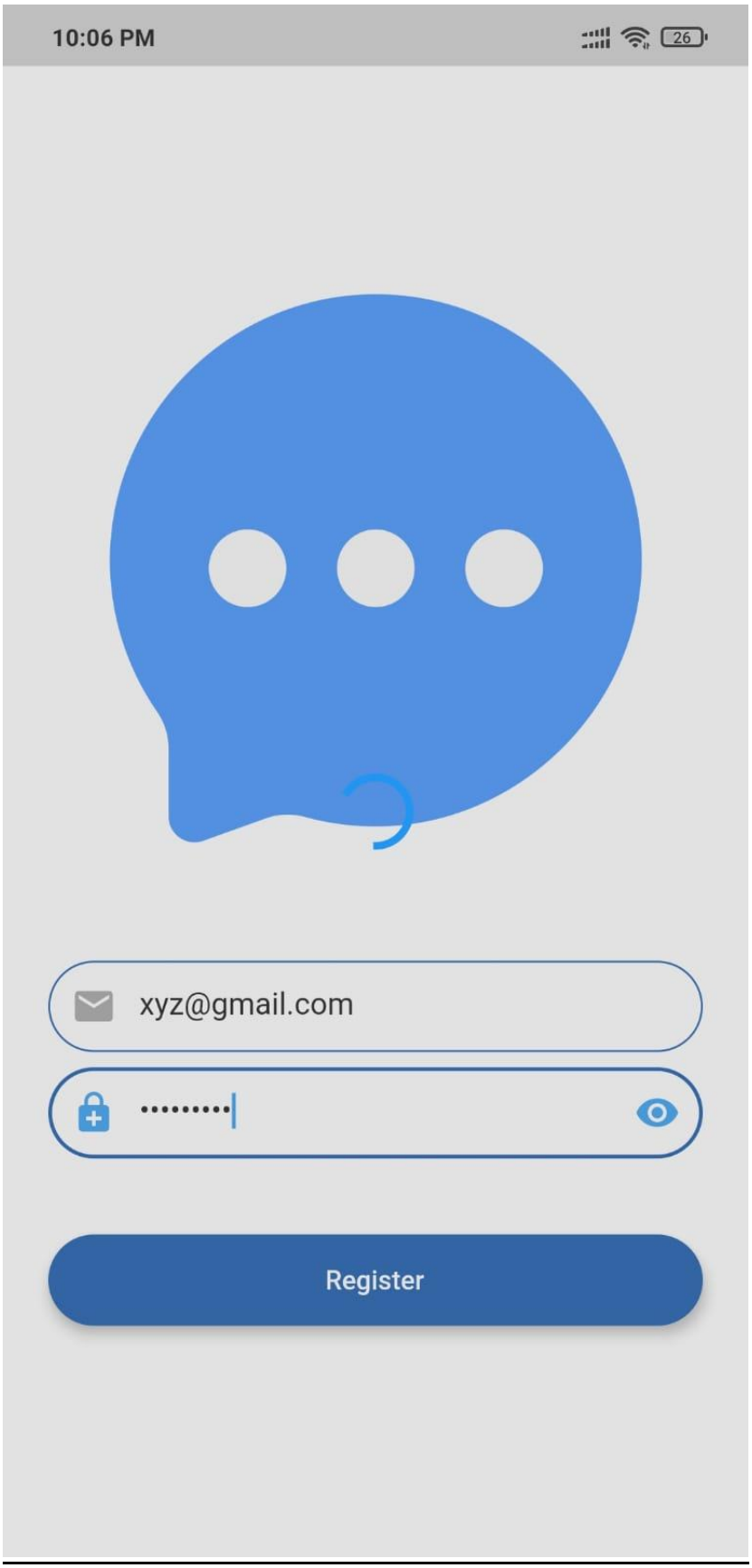
```
version: 1.0.0+1

environment:
  sdk: ">=2.7.0 <3.0.0"

dependencies:
  flutter:
    sdk: flutter
  animated_text_kit: ^3.1.0
  cloud_firestore: ^0.16.0
  cupertino_icons: ^1.0.0
  firebase_auth: ^0.20.0+1
  firebase_core: ^0.7.0
  modal_progress_hud: ^0.1.3

dev_dependencies:
  flutter_test:
    sdk: flutter

# For information on the generic Dart part of this file, see the
# following page: https://dart.dev/tools/pub/pubspec
# The following section is specific to Flutter.
flutter:
```

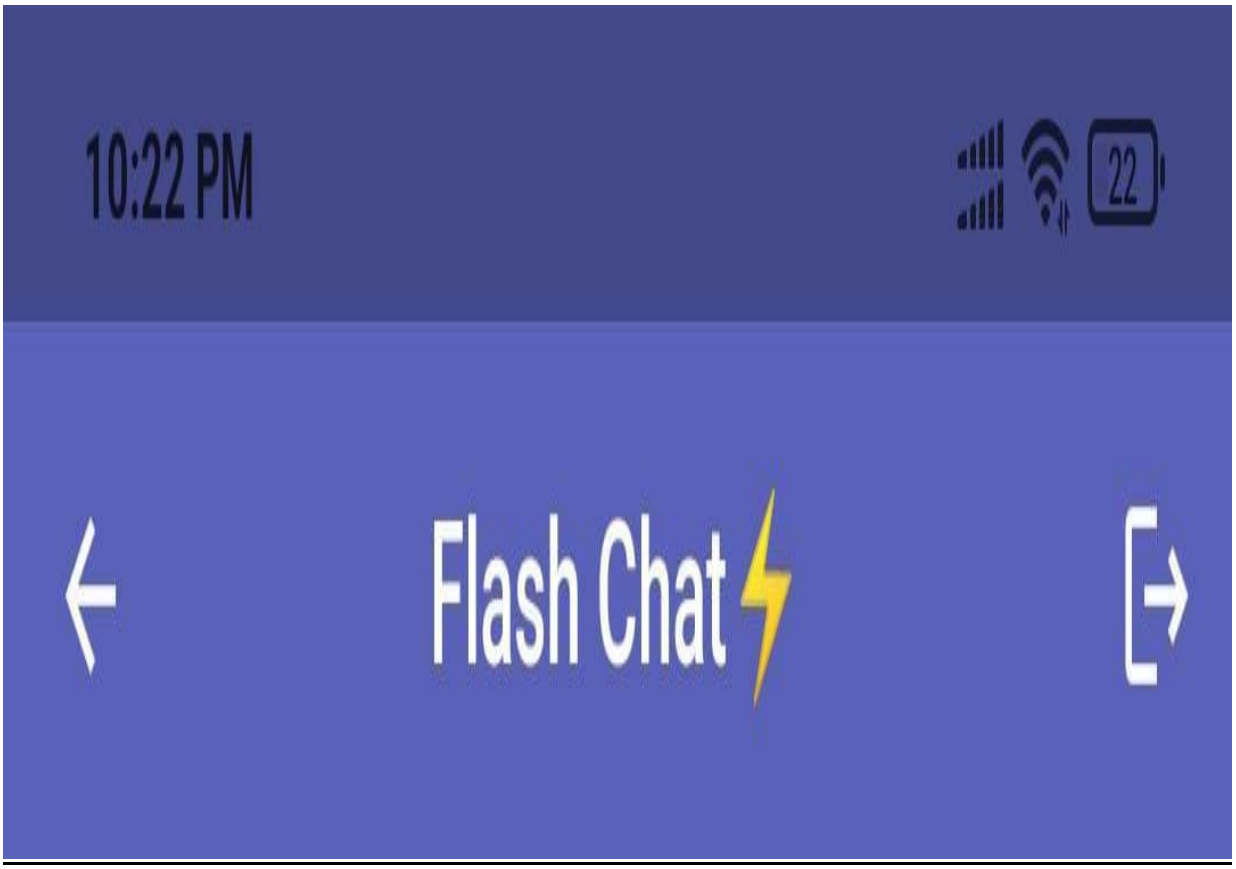
```
# The following line ensures that the Material Icons font is
# included with your application, so that you can use the icons in
# the material Icons class.
uses-material-design: true

# To add assets to your application, add an assets section, like this:
assets:
  - images/
#   - images/a_dot_ham.jpeg
# An image asset can refer to one or more resolution-specific "variants", see
# https://flutter.dev/assets-and-images/#resolution-aware.
# For details regarding adding assets from package dependencies, see
# https://flutter.dev/assets-and-images/#from-packages
# To add custom fonts to your application, add a fonts section here,
# in this "flutter" section. Each entry in this list should have a
# "family" key with the font family name, and a "fonts" key with a
# list giving the asset and other descriptors for the font. For
# example:
# fonts:
#   - family: Schyler
#     fonts:
#       - asset: fonts/Schyler-Regular.ttf
#       - asset: fonts/Schyler-Italic.ttf
#         style: italic
```

```
#   - family: Trajan Pro
#     fonts:
#       - asset: fonts/TrajanPro.ttf
#       - asset: fonts/TrajanPro_Bold.ttf
#         weight: 700
#
# For details regarding fonts from package dependencies,
# see https://flutter.dev/custom-fonts/#from-packages
```


10:00 PM
27
Flash Chat
Log In
Register


10:00 PM
27
Enter your email
Enter your password
Register


10:06 PM
26
xyz@gmail.com
Register


10:22 PM
22
Flash Chat

Type your message here...

Reference

[1] Chitra.(2013, September 26).What is the advantages and disadvantages of Whatsapp and Wechat apps [Msg 2].Message post to http://www.indiastudychannel.com/experts/33915-What-advantage-disadvantage-WhatsApp-WeChat-apps.aspx

[2] Mj1111983.(2013, September 19).Wechat send free text, image, video & audio messages.People & Blog.Product retrieved from http://www.youtube.com/watch?v=7K-DONyxhzg

[3] Mslgroup.(dicember, 2013).*We Chat About Wechat* . Retrieved from http://mslgroup.com/insights/2013/we-chat-about-wechat.aspx

[4] Wechat.(2014).[Blog].Retrieved from http://www.wechat.com/en/

[5] Wikipedia.(n.d.).Retrieved September 21, 2014 from the Wikipedia The free Encyclopedia : http://en.wikipedia.org/wiki/ChatApp