MINI PROJECT

SYNOPSIS

ON

WhatsApp Clone for Android and IOS

Using Flutter

Department of Computer Science & Applications

SUBMITTED TO: -

Mr. Bhanu Kapoor

(Technical Trainer)

SUBMITTED BY: -

Shruti (201500676)

Vishnu Pratap Singh Chauhan (201500803)

Rohit Kumar (201500586)

CONTENTS

Acknowledgement

It gives us a great sense of pleasure to present the synopsis of the B. Tech Mini Project undertaken during B. Tech III Year. This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe special debt of gratitude to

Mr. Bhanu Kapoor, Technical Trainer, for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work.

His sincerity, thoroughness and perseverance has been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies. We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

Shruti (201500676)
Vishnu Pratap Singh Chauhan (201500803)
Rohit Kumar (201500586)

OVERVIEW

What is Flutter?

Flutter is Google's portable UI toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

Google introduced Flutter in May 2017. Flutter is an open-source UI software development kit used to create applications for both Android and iOS. Instead of writing different code for each operating system, it uses the same code base to create a cross-platform mobile application. Dart is the programming language used to create Flutter applications.

What kind of apps can I build with Flutter?

Flutter is designed to support mobile apps that run on both Android and iOS, as well as interactive apps that you want to run on your web pages or on the desktop.

Apps that need to deliver highly branded designs are particularly well suited for Flutter. However, you can also create pixel-perfect experiences that match the Android and iOS design languages with Flutter.

Flutter's package ecosystem supports a wide variety of hardware (such as camera, GPS, network, and storage) and services (such as payments, cloud storage, authentication, and ads).

Features of Flutter

Some features of Flutter include:

- Hot Reload
- Cross-Platform development
- Widget library
- Native Performance
- Open-Source

Hot Reload:

Hot reload makes it possible to see the changes in the code instantly reflected on the UI. This speeds up the process to work on the outlook of the application; moreover, it enables developers to correct errors that save cost and effort.

Cross-platform Development:

Flutter enables developers to write code that works on different platforms. Two different applications can use the same codebase. In addition to sharing the UI code, the UI itself is also shareable. This makes maintenance of the single codebase much easier as opposed to different codes for different platforms.

Widget Library

Everything in Flutter is defined as a widget. A widget can be a color, padding, or menu. Flutter is capable of creating complex widgets that can be customized according to the requirement of the application. Built-in widgets are also available for usage. Some examples include Cupertino pack and Material Design, which have sets of widgets that provide a glitch-free user experience.

Native Performance

In Flutter, platform-specific widgets are provided for Google Fuchsia, Android, and iOS. These widgets can be integrated into the Flutter application to make use of the different platform-dependent functionalities. Existing Java, Swift, and Objective-C codes can be used to utilize native features such as camera and geolocation. Therefore, Flutter can easily incorporate third-party integrations and APIs.

Open Source

Google introduced Flutter as an open-source platform. Various design options can be explored by developers to create Flutter applications. User-friendly applications can be created with Material Design and Cupertino widgets. Flutter Form is a community of Flutter enthusiasts who come together to answer questions related to Flutter and discuss it. Flutter is free of cost and has detailed documentation and communities available online.

INTRODUCTION TO APP

Our project is to clone very famous social media app "WhatsApp"

WhatsApp is a free cross-platform messaging service. It lets users of iPhone and Android smartphones and Mac and Windows PC call and exchange text, photo, audio and video messages with others across the globe for free, regardless of the recipient's device.

Our Aim is to cover all Main features of chatting App like Phone Number Authentication, One to one & Group chatting with contacts only - includes text, image, gif, video, audio (with recording), emoji sharing and image and video caching, status visible to contacts only and disappears after 24 hours, video calling (one-one and group), online/offline status, Message Seen Feature, Automatic Scrolling on New Message, Replying to Messages and much more!

IMPLEMENTATION STEPS

1. Landing Screen UI
2. Login Screen UI
3. Flutter Firebase Phone Authentication
4. Adding Riverpod
5. Auth Controller
6. About Riverpod
7. Resolving iOS App Error
8. OTP Screen UI
9. Verifying OTP
10. User Information Screen UI
11. Saving User Data To Firestore
12. Persisting Auth State
13. Displaying Contacts in User's Phone
14. Displaying Name and online/offline status
15. Modifying Bottom Chat Field UI
16. Sending Text Message
17. Displaying Chat Contacts
18. Displaying Messages
19. Automatic Scrolling on New Message
20. Changing Online/Offline Status
21. Sending Image Message
22. Displaying Cached Image
23. Sending Video Message
24. Sending Emojis
25. Sharing GIFs
26. Recording & Sharing Audio in Flutter App
27. Playing Audio in Flutter App
28. Replying to Messages
29. Seen Feature
30. Uploading Statuses/Stories
31. Displaying Statuses/Stories
32. Creating Groups
33. Displaying Groups
34. Fetching Group Chats & Chatting in Group
35. Managing Agora Dashboard & Credentials

36. Video Calling
37. Disconnecting Call
38. Group Calling

SOFTWARE REQUIREMENTS

- Flutter SDK
- Dart SDK
- Android Studio/Visual Studio Code
- Windows 10/ 64-bit

HARDWARE REQUIREMENT

- Android Emulator (Android Version Lollipop and above)
- 4 GB RAM Required in Desktop

REFERENCES/WEBSITE

- www.google.com
- https://docs.flutter.dev/
- https://dart.dev/guides
- https://developers.google.com/fonts/docs/material_icons
- https://appicon.co/
- www.youtube.com

FACULTY GUIDELINES:

Mr. Bhanu Kapoor (Technical Trainer in GLA University)

GitHub Repository Link:

https://github.com/shru-ty/wa_clone.git