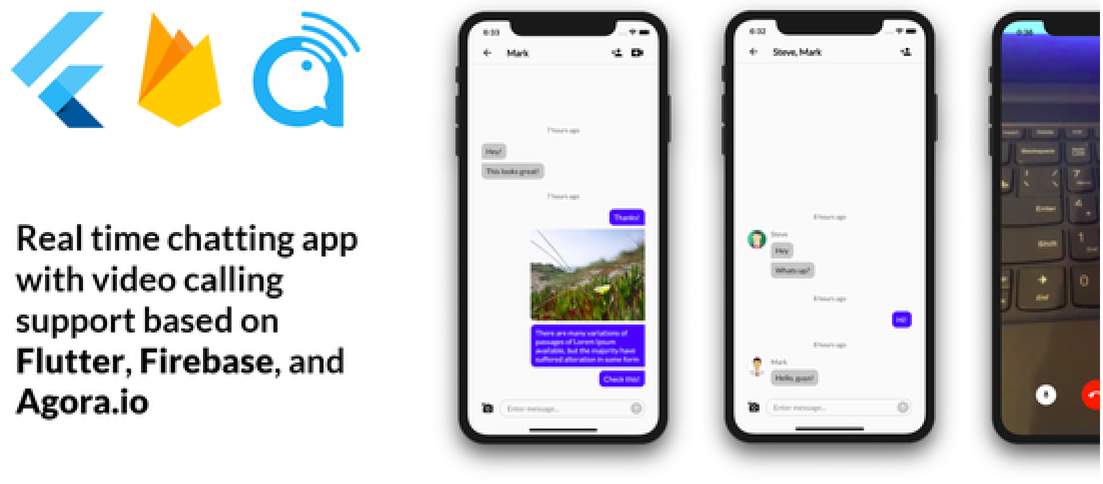
Flutter Firebase Chat Documentation

Flutter Firebase Chat is a real time chatting app with video calling support based on Flutter, Firebase, and Agora.io. You can run this app on both platforms: Android and iOS. Also you can easily customize and refine it for yourself, since it uses a BLoC pattern.


Real time chatting app with video calling support based on Flutter, Firebase, and Agora.io
Mark
Hey!
This looks great!
Thanks!
Check this!
Steve, Mark
Steve
Hey
Whats up?
Hi!
Mark
Hello, guys!

Main features

- One-to-one chatting
- Group chatting
- One-to-one video calling via Agora.io
- Image sharing
- Email authentication
- Implemented BLoC pattern

Flutter packages

- firebase_core
- firebase_auth
- cloud_firestore
- firebase_storage
- flutter_bloc
- agora_rtc_engine
- flutter_screenutil
- image_picker
- photo_view
- timeago
- email_validator
- adaptive_action_sheet
- flutter_slidable
- visibility_detector

- keyboard_dismisser
- permission_handler

Code Overview

The app is built using Flutter and uses Cloud Firestore as a database. The app also uses Agora.io to make one-to-one video calls and flutter_bloc in order to implement the BLoC pattern.

The app uses the following Project Structure:

Project Structure

```
...
   ├── models/              # This file contains the models used in the
project.
   ├── screens/             # This folder contains many different folders,
each of which corresponds to a different screen of the app.
   ├── services/            # This folder contains the services that
connect with the Cloud Firestore.
   ├── widgets/             # This folder contains the widgets which are
used in multiple different screens.
   ├── app_colors.dart      # This file contains the colors used in the
project.
   ├── app_constants.dart   # This file contains the constants used in the
project.
   └── app.dart             # This file contains the main StatelessWidget
(a MaterialApp wrapped in the necessary BlocProvider).
```

Also each screen folder contains the following files:

```
...
   ├── screen_bloc.dart     # This file contains BLoC implementation for
the current screen.
   ├── screen_event.dart    # This file contains BLoC's events for the
current screen.
   ├── screen_screen.dart   # This file contains the screen's internal
content.
   ├── screen_state.dart    # This file contains BLoC's states for the
current screen.
   └── screen.dart          # This file contains all exports for the
current screen.
```

Project Setup

In order to setup the project you need to follow 3 steps: setup Agora.io, setup Firebase, and setup your flutter project.

Agora.io setup

1. Create a developer account at https://www.agora.io/.
2. Create a project (using APP ID mode).
3. Copy the app ID and set the const agoraAppId in lib/src/app_constants.dart.

Firebase setup

1. Go to https://console.firebase.google.com and create a project.
2. Go to "Authentication/Sign-in method" and enable "Email/Password".
3. Go to "Firestore Database" and create a Cloud Firestore database.
4. Go to "Firestore Database/Rules" and publish this code:

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if request.auth.uid != null;
    }
  }
}
```

5. Go to "Storage/Rules" and publish this code:

```
rules_version = '2';
service firebase.storage {
  match /b/{bucket}/o {
    match /{allPaths=**} {
      allow read, write: if request.auth != null;
    }
  }
}
```

6. Go to "Project Settings", add an Android app to your project. Follow the assistant, and download the generated google-services.json file and place it inside android/app.
7. Add an iOS app to your project. Follow the assistant, download the generated GoogleService-Info.plist file. Do NOT follow the steps named "Add Firebase SDK" and "Add initialization code" in the Firebase assistant. Open ios/Runner.xcworkspace with Xcode, and within Xcode place the GoogleService-Info.plist file inside ios/Runner.

Flutter setup

1. Install package dependencies:

```
flutter pub get
```

2. Use one of these commands to build the project:

```
flutter build ipa
flutter build apk
flutter build appbundle
```