

Lab

- Setting up project and app
- Setting up Firebase Auth
- Setting up Google Sign In
- Setting up Image Picker
- Enable API in Google Cloud
- Running the Chat App

Setup project and app

- Create firebase project
- Connect your app to firebase project
- Upgrade to Blaze plan
- Create firestore
- Create missing folder

Create firebase project

× 建立專案

請先輸入專案名稱^①

專案名稱

chat-app

chat-app-20778

已有 Google Cloud 專案嗎?
[將 Firebase 新增到 Google Cloud 專案](#)

繼續

Firestore 專案的 AI 輔助功能

Gemini in Firebase 已整合至 Firebase 控制台，用以簡化開發程序。

- ✦ 與 Gemini 對話，輕鬆規劃及設計應用程式、排解問題，並取得專案專屬建議
- ⚙️ 在 Firebase Crashlytics 使用 AI 輔助，為 Apple 和 Android 應用程式偵錯及排解問題
- 📊 運用活動洞察資訊和建議，增進使用者與 Firebase 雲端通訊之間的互動
- 🔗 使用 Firebase Data Connect 以自然語言生成結構定義及探索資料

啟用 Gemini in Firebase
建議做法

免責事項：
Gemini in Firebase 仍在開發階段，可能會提供不正確的資訊。在應用程式使用 Gemini in Firebase 前，請先測試生成的程式碼。Gemini 可以與 Firebase 資料互動，並根據應用程式提供合適的回應，且可能會使用你的提示或資料進行訓練。
[進一步瞭解我們如何訓練模型](#) 使用 Gemini in Firebase 時，須遵守 [《Google 服務條款](#) 和 [《生成式 AI 使用限制政策](#) 》

上一部分

繼續

Google Analytics

Google Analytics 是免付費且無使用限制的數據分析解決方案，能讓你在 Firebase Crashlytics、雲端通訊、應用程式內通訊、遠端設定、A/B 測試和 Cloud Functions 中執行指定目標和查看報表等動作。

Google Analytics 可讓你運用下列功能：

- × A/B 測試^①
- × 以事件為基礎的 Cloud Functions 觸發條件^②
- × 所有 Firebase 產品中的使用者區隔和指定目標功能^③
- × 沒有使用限制的免費報表^④
- × Crashlytics 中的導覽標記記錄^⑤

啟用這項專案的 Google Analytics 功能
建議做法

上一部分

建立專案

chat-app Spark 方案

需要入門協助嗎？向 Gemini 說明你的專案

將 Firebase 新增至應用程式即可開始使用

請先新增應用程式

iOS+ Android </> <=> <=>

Connect your app to firebase project

1 準備工作區

如要開始設定，最簡單的方式就是使用 FlutterFire CLI。

繼續操作之前，請務必完成下列事項：

- 安裝 [Firebase CLI](#) 並登入 (執行 `firebase login`)
- 安裝 [Flutter SDK](#)
- 建立 Flutter 專案 (執行 `flutter create`)

1 Clone the project first

2 安裝並執行 FlutterFire CLI

在任何目錄中執行下列指令：

```
$ dart pub global activate flutterfire_cli
```

接著，在 Flutter 專案的根目錄中執行下列指令：

```
$ flutterfire configure --project=chat-app-20778
```

完成後，系統會自動透過 Firebase 註冊每個平台專屬的應用程式，並將 1:

```
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> firebase login
Already logged in as jerryfeng1103@gmail.com
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 1.2.0.
Downloading packages... (1.0s)
The package flutterfire_cli is already activated at newest available version.
To recompile executables, first run `dart pub global deactivate flutterfire_cli`.
Installed executable flutterfire.
Activated flutterfire_cli 1.2.0.
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> flutterfire configure --project=chat-app-20778
i Found 4 Firebase projects. Selecting project chat-app-20778.
? Which platforms should your configuration support (use arrow keys & space to select)? >
✓ android
✓ ios
  macos
✓ web
  windows
```

```
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> flutterfire configure --project=chat-app-20778
i Found 4 Firebase projects. Selecting project chat-app-20778.
✓ Which platforms should your configuration support (use arrow keys & space to select)? · android, ios, web
✓ Which Android application id (or package name) do you want to use for this configuration, e.g. 'com.example.app'? · com.example.app
i Firebase android app com.example.app is not registered on Firebase project chat-app-20778.
i Registered a new Firebase android app on Firebase project chat-app-20778.
✓ Which ios bundle id do you want to use for this configuration, e.g. 'com.example.app'? · com.example.app
i Firebase ios app com.example.app is not registered on Firebase project chat-app-20778.
i Registered a new Firebase ios app on Firebase project chat-app-20778.
i Firebase web app flutter_app (web) is not registered on Firebase project chat-app-20778.
i Registered a new Firebase web app on Firebase project chat-app-20778.

Firebase configuration file lib\firebase_options.dart generated successfully with the following information:

Platform  Firebase App Id
web       1:245817093486:web:06d16c45e7504535ffe57f
android   1:245817093486:android:3dc530d7ff1e2a6effe57f
ios       1:245817093486:ios:fc530408fa38f5acffe57f
```

6

Used com.example.flutter_app as package name

```
> functions
v lib
  > models
  > repositories
  > services
  > view_models
  > views
  ● firebase_options.dart 8, U
  ● main.dart 9+
  📄 firebase.json U
  ! flutter_native_splash.yaml
  ! pubspec.yaml
  ⓘ README.md
```

7

Check if there is firebase_option.dart

Upgrade to Blaze plan

1

Spark
免付費 (每月 \$0 美元) 升級

專案總覽

產品類別

建構

執行

數據分析

AI

所有產品

相關開發人員工具

Firestore Studio

Checks

2

3

免付費 (每月 \$0 美元)
Spark 方案

即付即用
Blaze 方案

付費服務的用量和配額受限制

目前方案

付費服務的用量和配額增加

可以完整使用免費服務，並享有一定的付費 Firebase 服務用量配額

包含免費服務，以及付費服務的免費配額。超出付費服務的用量配額之後，用多少付多少。隨時可以取消訂閱。

無法使用其他 Google Cloud 服務

經過擴充，可以使用其他 Google Cloud 服務

所有方案皆包含 Analytics、通知、當機回報和支援等服務

所有方案皆包含 Analytics、通知、當機回報和支援等服務

選取方案

3

已選取的方案

即付即用 Blaze 方案 查看完整方案詳情

選取 Cloud Billing 帳戶

建立 Cloud Billing 帳戶

可用的 Cloud Billing 帳戶 ↑ 操作數

PostOffice card

4

設定帳單預算

專案費用接近或超過這個金額時，傳送電子郵件給帳單帳戶的所有管理員

帳單帳戶：
PostOffice card 查看帳戶

預算金額 (TWD)
25

系統會在金額達到 \$12.50 (50%)、\$22.50 (90%) 和 \$25.00 (100%) 時傳送電子郵件給你，你稍後可以在 Google Cloud Console 中自訂金額。

5

6

從現在起，使用付費 Firebase 和 Google Cloud 服務時，超出免付費配額範圍外的部分會依用量計費

已選取的方案

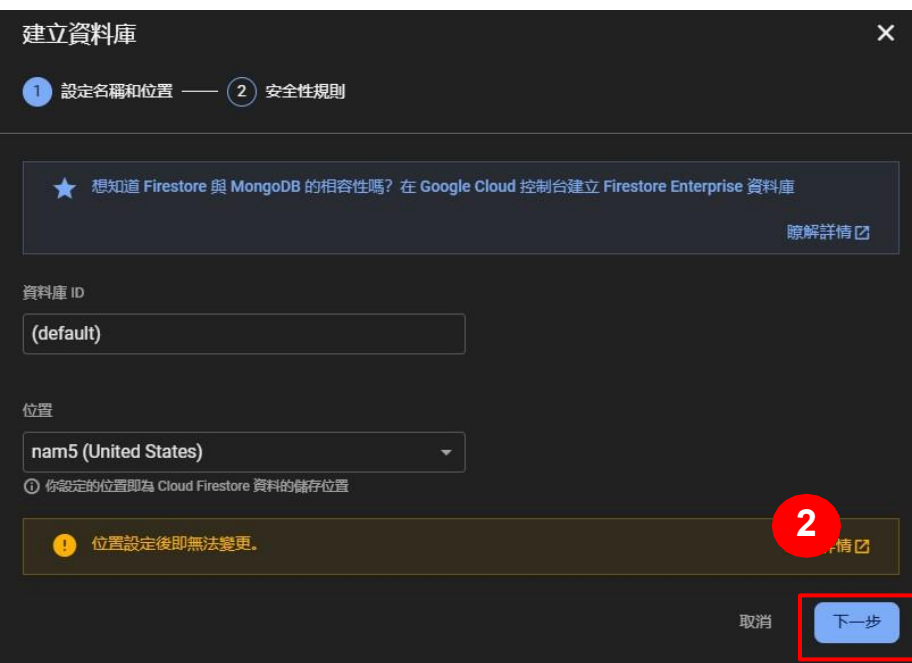
即付即用 Blaze 方案 查看完整方案詳情

帳單帳戶：
PostOffice card 查看帳戶

預算金額 (TWD)
25

取消 連結 Cloud Billing 帳戶

Create firestore



Create missing folder

```
Flutter-basic-dart-group-chat-app> flutter create --platforms=android .
```

(created)

1

```
Flutter-basic-dart-group-chat-app> flutter create --platforms=ios .
```

(created)
(created)
schemes\Runner.xcscheme (created)

2

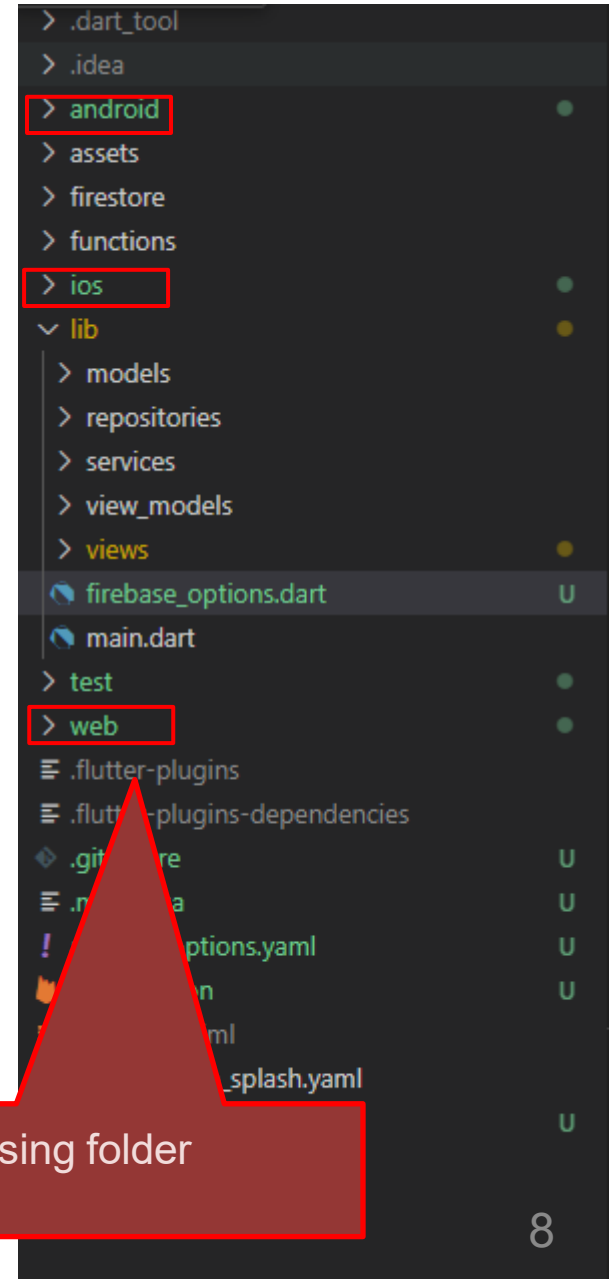
```
Flutter-basic-dart-group-chat-app> flutter create --platform web .
```

(created)

3

4

You should see the missing folder now



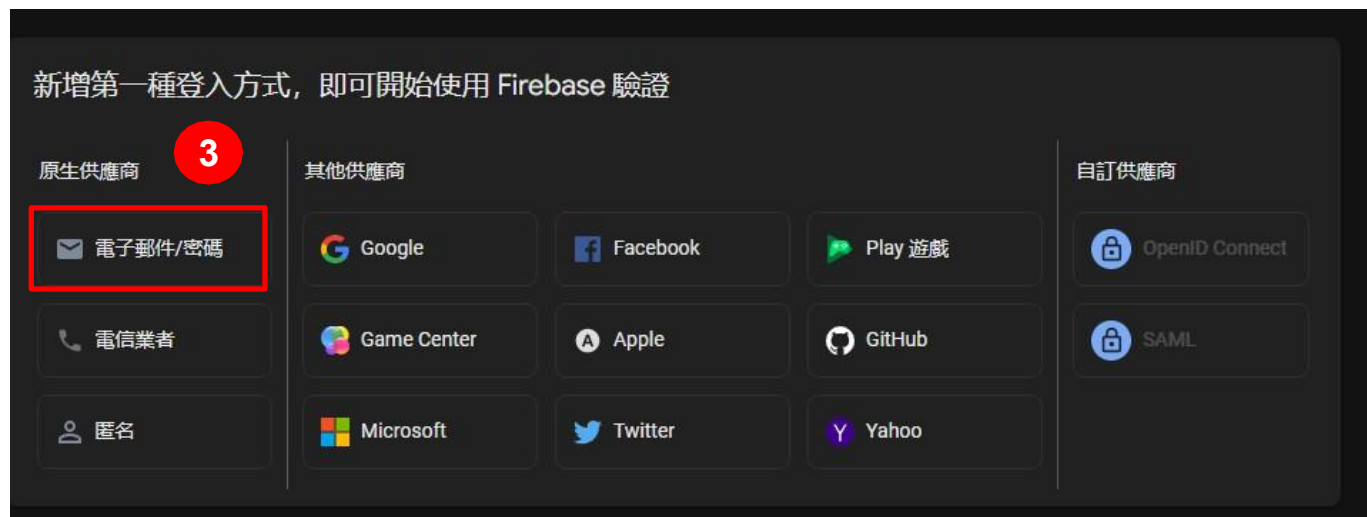
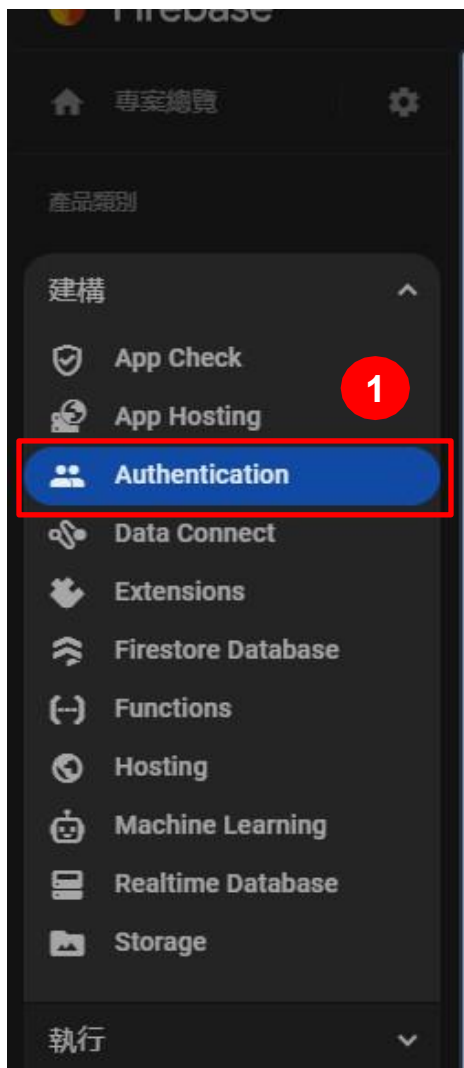
Lab

- Setting up project and app
- **Setting up Firebase Auth**
- Setting up Google Sign In
- Setting up Image Picker
- Enable API in Google Cloud
- Running the Chat App

Setting up Firebase Auth

- Enable Email sign-in
- Enable Google auth
 - Get you SHA-1 release fingerprint
 - Apply SHA-1 release fingerprint to your android app
 - Replace and Download service file to enable google sign-in

Enable email sign-in



4

✉ 電子郵件/密碼 啟用

可讓使用者以自己的電子郵件地址和密碼註冊。我們的 SDK 也提供電子郵件地址驗證、密碼救援和電子郵件地址變更等基本功能。[瞭解詳情](#)

電子郵件連結 (不需要密碼即可登入) 啟用

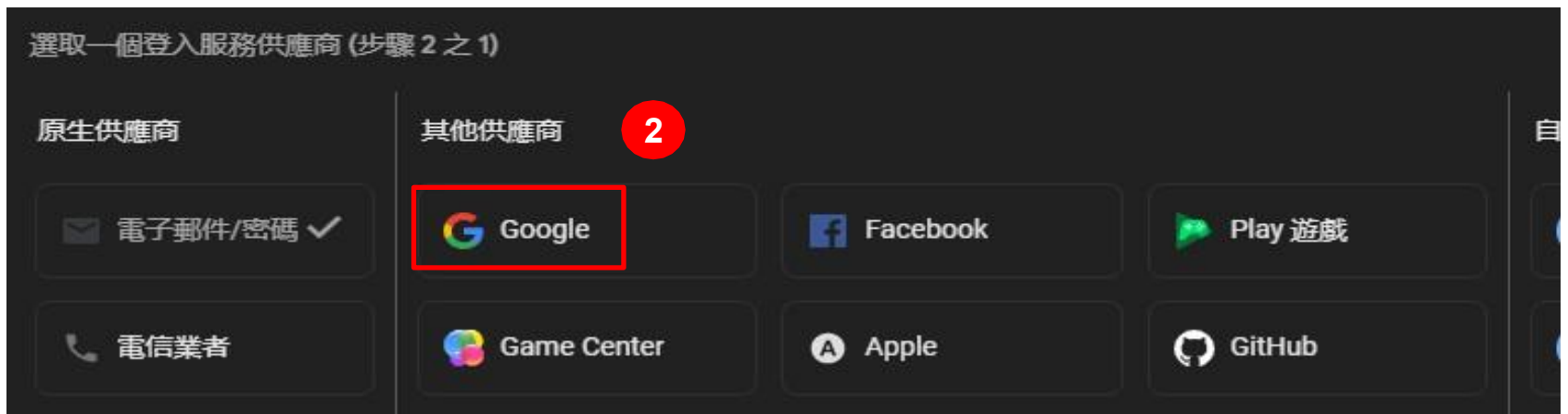
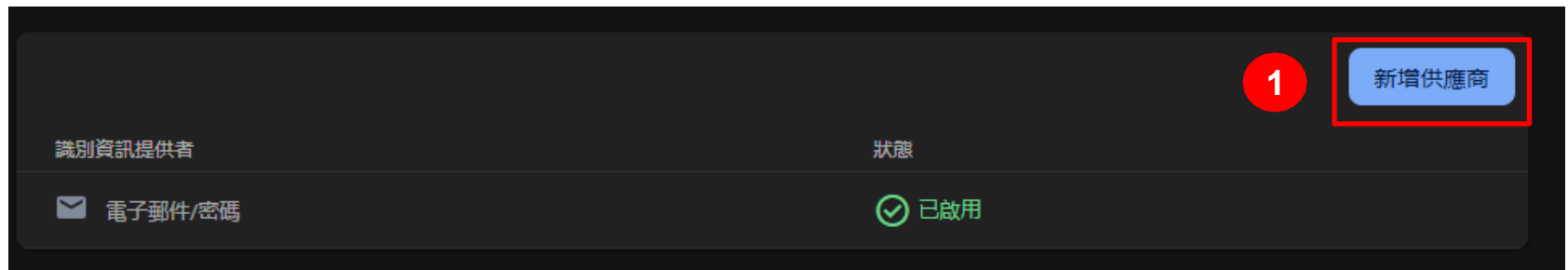
取消

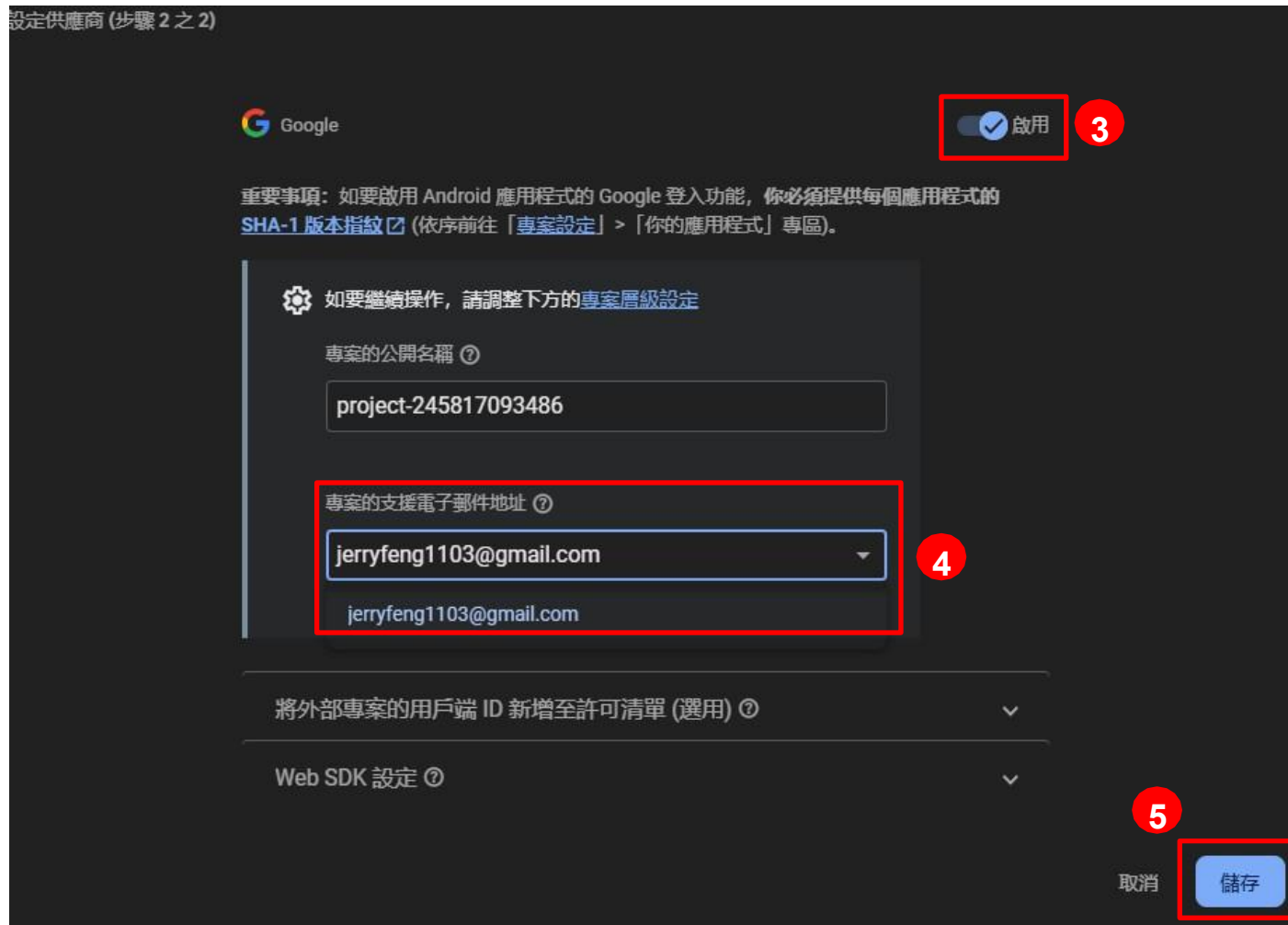
5

識別資訊提供者	狀態
✉ 電子郵件/密碼	<input checked="" type="checkbox"/> 已啟用

新增供應商

Enable google sign-in





You need [SHA-1 release fingerprint](#) for Android app to enable google sign-in

重要事項: 如要啟用 Android 應用程式的 Google 登入功能, 你必須提供每個應用程式的 [SHA-1 版本指紋](#) (依序前往 [專案設定] > [你的應用程式] 專區)。

Get your SHA-1 release fingerprint (Windows)

- Download Java x64 installer: (if you don't have java)
<https://www.oracle.com/tw/java/technologies/downloads/#jdk17-windows>
- Add C:\Program Files\Java\jdk-{version}\bin to System Path
In cmd: java -version for checking Java successfully installed. (if you don't have java)
- In cmd:
keytool -list -v -alias androiddebugkey -keystore "C:\Users\{your user name}\.android\debug.keystore"
Default Password: android
- It should print your fingerprint like this: (This is just an example)
Certificate fingerprint: SHA1:
DA:39:A3:EE:5E:6B:4B:0D:32:55:BF:EF:95:60:18:90:AF:D8:07:09
- See next page for example

Get your SHA-1 release fingerprint (Mac)

- Download Java 17 x64 DMG Installer: (if you don't have java)
<https://www.oracle.com/tw/java/technologies/downloads/#jdk17-mac>
- In terminal:
- `mkdir ~/.android`
- `keytool -genkey -v -keystore ~/.android/debug.keystore -storepass android -alias androiddebugkey -keypass android -keyalg RSA -keysize 2048 -validity 10000`
- Enter your name and skip other questions
- `keytool -list -v -alias androiddebugkey -keystore ~/.android/debug.keystore`
Default Password: android
- It should print your fingerprint like this: (This is just an example)
Certificate fingerprint: SHA1:
DA:39:A3:EE:5E:6B:4B:0D:32:55:BF:EF:95:60:18:90:AF:D8:07:09
- See next page for example

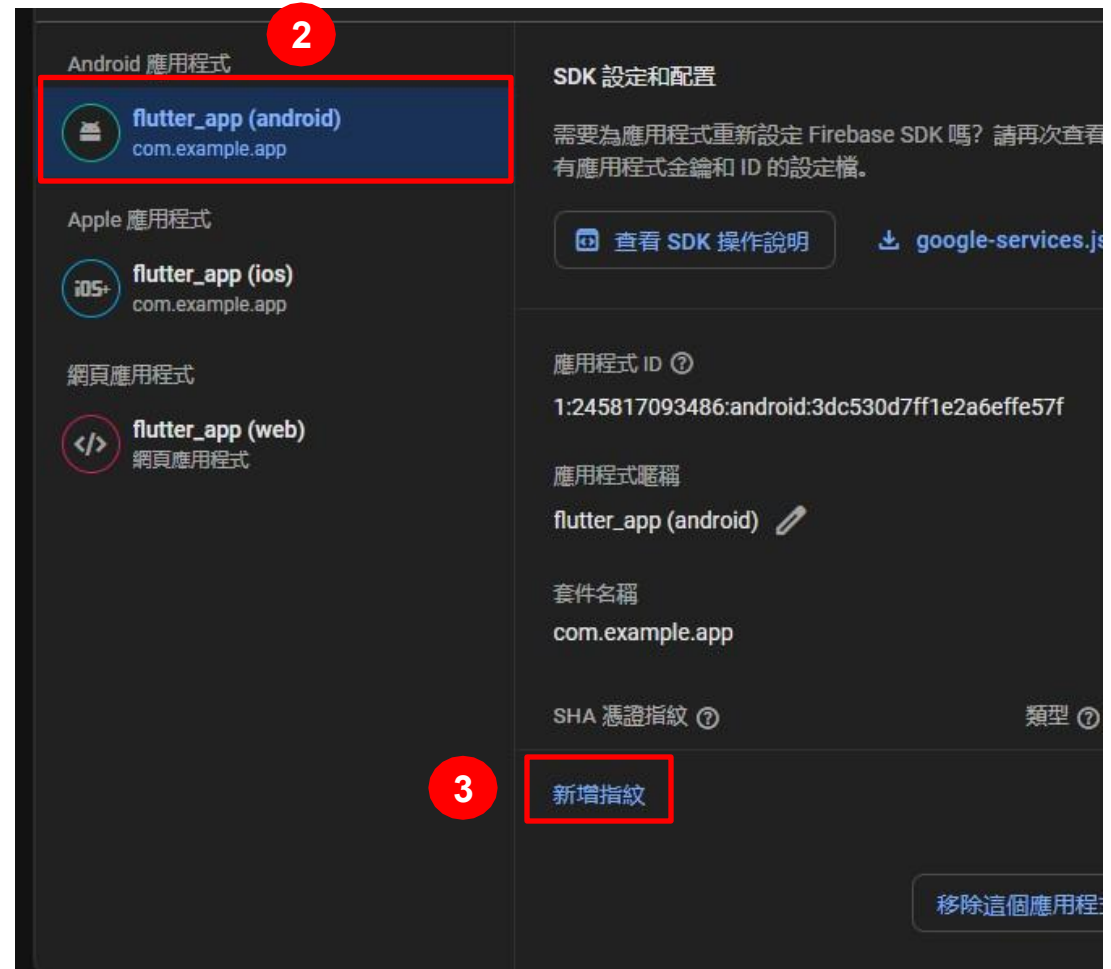
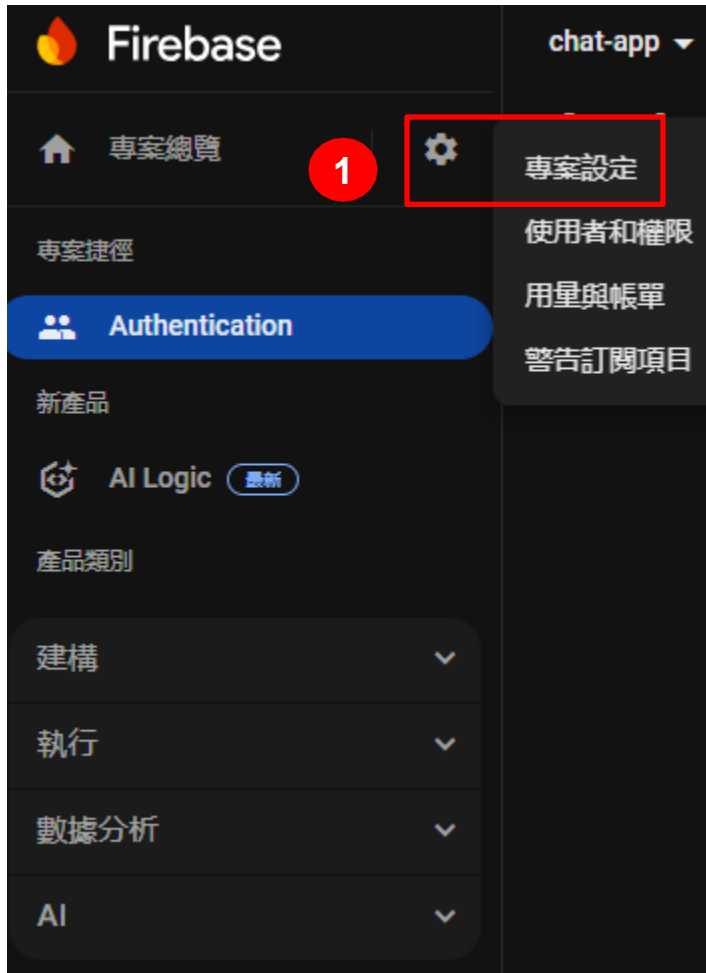
Example (window)

```
... 0 more
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> keytool -list -v -alias androiddebugkey -keystore C:\Users\jerry\.android\debug.keystore
輸入金鑰儲存庫密碼:

別名名稱: androiddebugkey
建立日期: 2024年7月16日
項目類型: PrivateKeyEntry
憑證鏈長度: 1
憑證 [1]:
擁有者: C=US, O=Android, CN=Android Debug
發行人: C=US, O=Android, CN=Android Debug
序號: 1
有效期自: Tue Jul 16 17:25:22 CST 2024 到: Thu Jul 09 17:25:22 CST 2054
憑證指紋:
SHA1: 13:7B:03:3E:95:B7:03:ED:2B:1C:3F:7D:11:54:E7:45:11:24:E1:70
SHA256: 26:63:6B:AE:FC:E4:09:E1:64:83:FF:E3:51:8B:F4:B0:15:DC:F7:9B:60:39:F6:DC:D6:AA:59:79:4D:E5:50:70
簽章演算法名稱: SHA1withRSA (低強度)
主體公開金鑰演算法: 2048 位元的 RSA 金鑰
版本: 1


Warning:
憑證 uses the SHA1withRSA signature algorithm which is considered a security risk. This algorithm will be disabled in a future update.
PS C:\Users\jerry\ss_spring_2025\lab-flutter-basic-dart-group-chat-app> □
```

Apply SHA-1 release fingerprint to you android app



 flutter_app (android)
com.example.app


Apple 應用程式


 flutter_app (ios)
com.example.app

網頁應用程式

 flutter_app (web)
網頁應用程式

需要為應用程式重新設定 Firebase SDK 嗎？請再次查看 SDK 設定的操作說明，或是直接下載含有應用程式金鑰和 ID 的設定檔。


 查看 SDK 操作說明

 google-services.json

應用程式 ID 

1:245817093486:android:3dc530d7ff1e2a6effe57f

應用程式暱稱

flutter_app (android) 


Paste the SHA-1 release fingerprint in
your terminal and save it

SHA 憑證指紋 

類型 

憑證指紋

13:7B:03:3E:95:B7:03:ED:2B:1C:3F:7D:11:54:E7:45:11:24:E1:7

 SHA1

SHA256

4

5

取消

儲存

Replace service file to enable google sign-in

- Download and Replace “google-services.json” files in “android/app” folder
- **Just Download** “GoogleService-info.plist”

Android 應用程式

flutter_app (android)
com.example.app

SDK 設定和配置

需要為應用程式重新設定 Firebase SDK 嗎？請再次查看 SDK 設定的操作說明，或是直接下載含有應用程式金鑰和 ID 的設定檔。

[查看 SDK 操作說明](#) [↓ google-services.json](#)

6

Android 應用程式

flutter_app (android)
com.example.app

Apple 應用程式

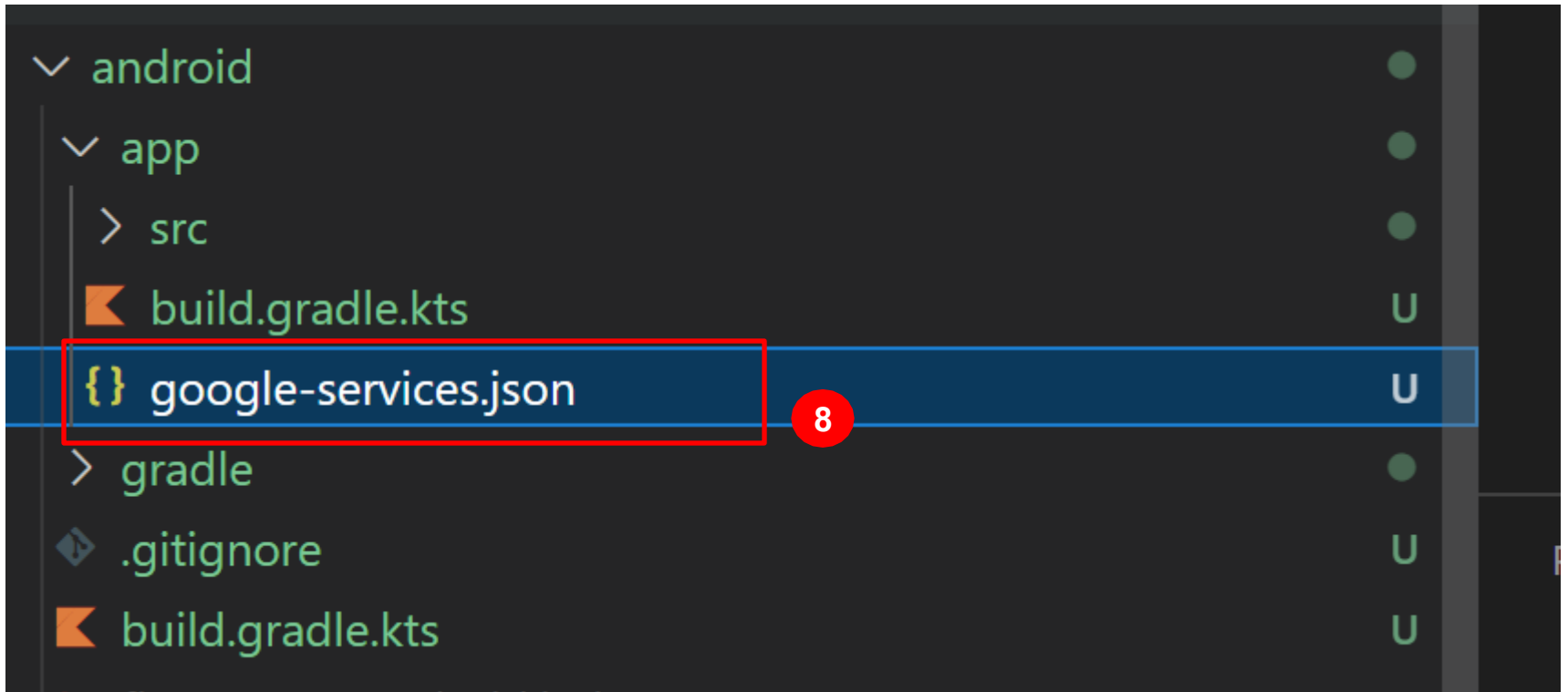
flutter_app (ios)
com.example.app

SDK 設定和配置

需要為應用程式重新設定 Firebase SDK 嗎？請再次查看 SDK 設定的操作說明，或是直接下載含有應用程式金鑰和 ID 的設定檔。

[查看 SDK 操作說明](#) [↓ GoogleService-Info.plist](#)

7



9. Finally run “flutterfire config” again

Lab

- Setting up project and app
- Setting up Firebase Auth
- **Setting up Google Sign In**
- Setting up Image Picker
- Enable API in Google Cloud
- Running the Chat App

Setting up Google Sign In

- Setting Google sign-in for ios
- Setting Google sign-in for web

Google sign-in for ios

- Please refer to the [official guide](#) in the **iOS Integration section** for instructions on setting up iOS
- Some note in the next page



Google sign-in for ios

Note:

- You only need to do step 3,4,6

Google sign-in for Web

- Please refer to the [official guide](#) in the **Web Integration section** for instructions on setting up iOS
- Some note in the next age



Google sign-in for Web

Note:

- You can skip the first step—it just walks you through a long process to create a Google Sign-In OAuth client ID.
- Your Google sign-in OAuth client ID is at Google Cloud Console > APIs & Services > Credentials page (make sure you are at the right project in Google Console)

<input type="checkbox"/>	名稱	建立日期 ↓	限制
<input checked="" type="checkbox"/>	iOS key (auto created by Firebase)	2025年5月21日	25 個 API ...
<input checked="" type="checkbox"/>	Android key (auto created by Firebase)	2025年5月21日	25 個 API ...
<input checked="" type="checkbox"/>	Browser key (auto created by Firebase)	2025年5月21日	25 個 API ...

OAuth 2.0 用戶端 ID

<input type="checkbox"/>	名稱	建立日期 ↓	類型	用戶端編號
<input type="checkbox"/>	Android client	2025年5月21日	Android	245817093486-1t4h...
<input type="checkbox"/>	iOS client for com.example.app (auto created by Google Service)	2025年5月21日	iOS	245817093486-a7ka...
<input type="checkbox"/>	Web client (auto created by Google Service)	2025年5月21日	網頁應用程式	245817093486-4trj...

- Last step is optional, but if you want to avoid any error, follow the instruction

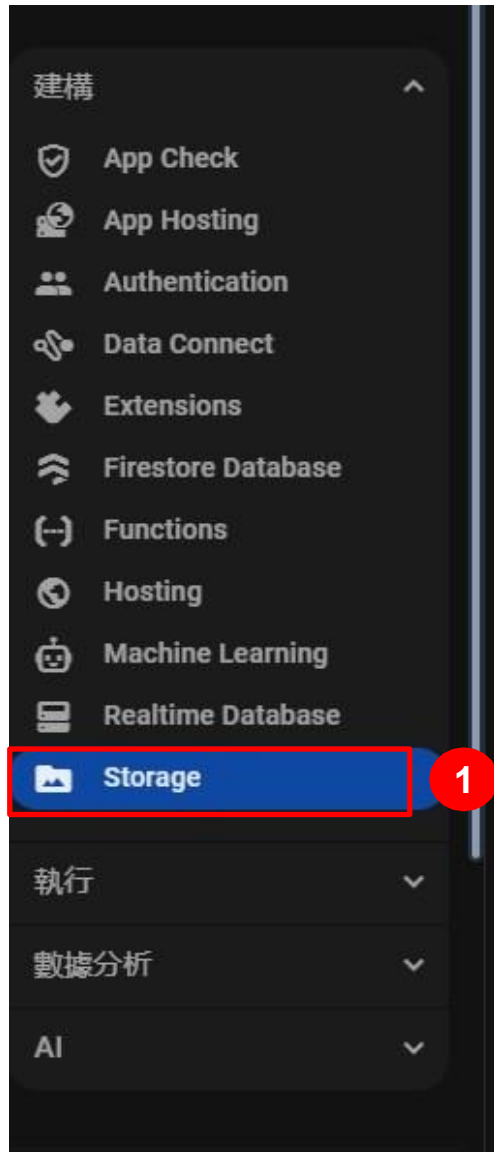
Lab

- Setting up project and app
- Setting up Firebase Auth
- Setting up Google Sign In
- **Setting up Image Picker**
- Enable API in Google Cloud
- Running the Chat App

Setting up Image picker

- Enable firebase Storage
- Setting up image picker

Enable firebase storage



設定預設 bucket

- 1 值區選項 ——— 2 安全性規則

bucket 參考檔案 儲存空間級別 ①

gs://chat-app-20778.firebaseio.com Regional

免付費位置

位置 ① 存取頻率 ①

US-CENTRAL1

Standard

所有位置

位置 ① 存取頻率 ①

US

標準

3

取消

繼續

設定預設 bucket

- ✓ 值區選項 ——— 2 安全性規則

定義資料結構之後，你必須撰寫規則來保護資料。

[瞭解詳情](#)

以正式版模式啟動

你的資料會預設為私人資料，系統只會按照你在安全性規則中指定的設定，將讀取/寫入權限授予用戶端。

4

以測試模式啟動

你的資料預設為公開資料，以便啟用快速設定功能。不過，你必須在 30 天內更新安全性規則，才能長期啟用用戶端讀取/寫入權限。

```
rules_version = '2';
service firebase.storage {
  match /b/{bucket}/o {
    match /{allPaths=**} {
      allow read, write: if
        request.time < timestamp.date(2025, 6, 20);
    }
  }
}
```

⚠ 套用測試模式的預設安全性規則之後，凡是可參照儲存空間值區的使用者，都能在未來 30 天內查看、編輯及刪除儲存空間值區中的所有資料

5

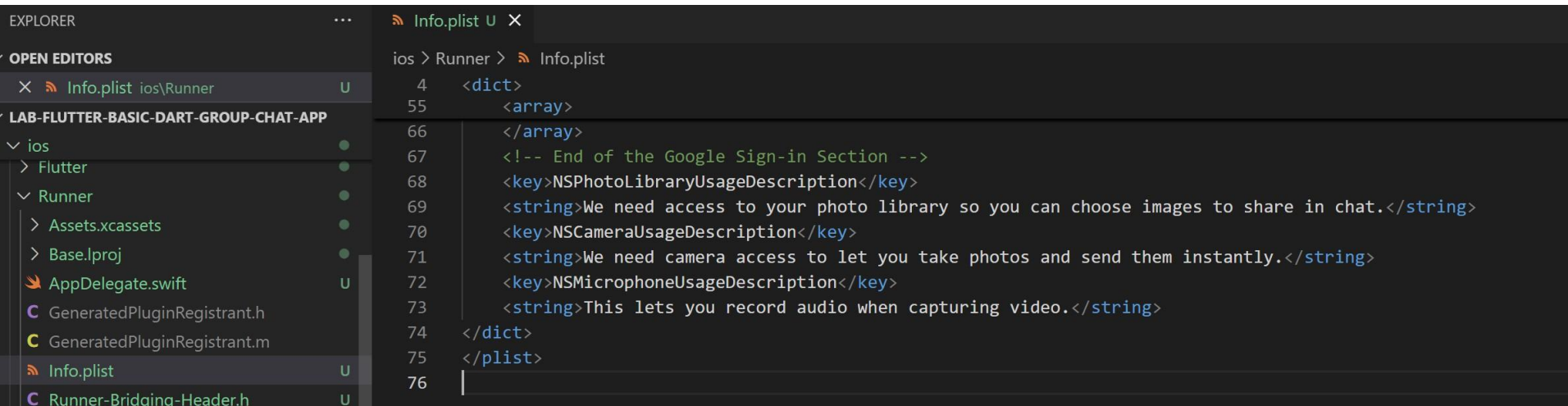
返回

1 建立

Setting up image picker

- Add to “ios/Runner/Info.plist”:

- `<key>NSPhotoLibraryUsageDescription</key>`
`<string>[Can be anything, just don't leave it blank.]</string>`
- `<key>NSCameraUsageDescription</key>`
`<string>...</string>`



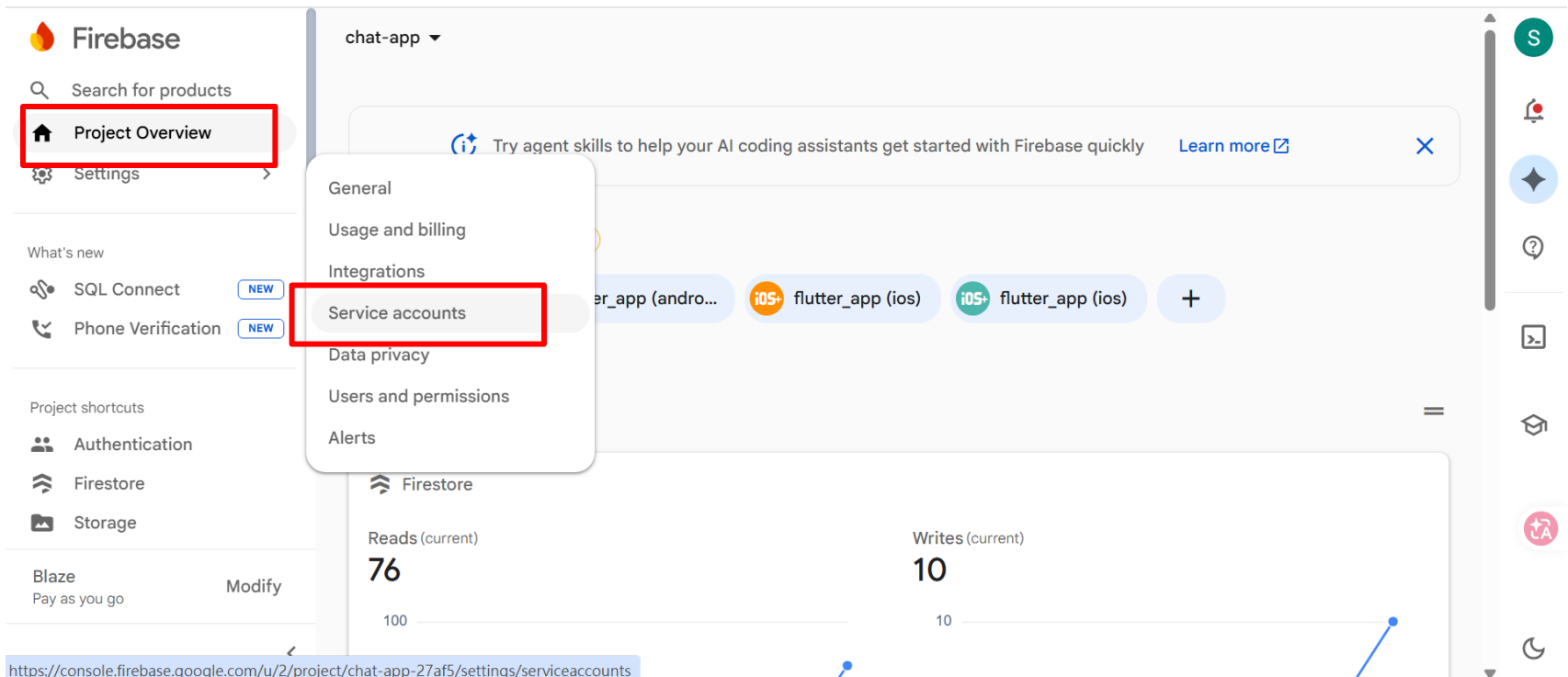
```
EXPLORER
...
Info.plist U X
ios > Runner > Info.plist
4 <dict>
55 <array>
66 </array>
67 <!-- End of the Google Sign-in Section -->
68 <key>NSPhotoLibraryUsageDescription</key>
69 <string>We need access to your photo library so you can choose images to share in chat.</string>
70 <key>NSCameraUsageDescription</key>
71 <string>We need camera access to let you take photos and send them instantly.</string>
72 <key>NSMicrophoneUsageDescription</key>
73 <string>This lets you record audio when capturing video.</string>
74 </dict>
75 </plist>
76
```

- Refer the [installation guide](#) for more details

Solve CORS Limitation

- You could refer to [this solution](#). The following steps are the same as [the link](#).

Step 1: Go to Project overview and click “Service accounts”



The screenshot shows the Firebase console interface for a project named 'chat-app'. The left sidebar contains navigation options: 'Project Overview' (highlighted with a red box), 'Settings', 'What's new' (including SQL Connect and Phone Verification), 'Project shortcuts' (Authentication, Firestore, Storage), and 'Blaze Pay as you go'. A dropdown menu is open over the 'Service accounts' option, which is also highlighted with a red box. The main content area shows a 'Firestore' section with 'Reads (current)' at 76 and 'Writes (current)' at 10. The URL at the bottom is <https://console.firebase.google.com/u/2/project/chat-app-27af5/settings/serviceaccounts>.

Solve CORS Limitation

Step 2: Click “Manage service account permissions”

The screenshot displays the Firebase console interface for a project named 'chat-app'. The left sidebar contains the 'Firebase' logo, a search bar, and navigation links for 'Project Overview' and 'Settings'. Below these are sections for 'What's new' (SQL Connect and Phone Verification), 'Project shortcuts' (Authentication, Firestore, Storage), and 'Blaze' (Pay as you go). The main content area is titled 'Project settings' and has tabs for 'General', 'Cloud Messaging', 'Integrations', 'Service accounts', 'Data privacy', 'Users and permissions', and 'Alerts'. The 'Service accounts' tab is selected. A red rectangular box highlights the link 'Manage service account permissions' in the top right corner of the main content area. A blue loading spinner is visible at the bottom center of the page.

Solve CORS Limitation

Step 3: Check your project name and go to cloud shell


The screenshot shows the Google Cloud IAM & Admin console. The project name 'chat-app' is highlighted in a red box in the top navigation bar. A red arrow points from this box to the 'Your project name' field in the 'Service accounts' page. Another red box highlights the 'Cloud shell' button in the top right corner, with a red arrow pointing to it from the search bar area. The main content area displays 'Service accounts for project "chat-app"' and a table of service accounts.

<input type="checkbox"/>	Email	Status	Name ↑	Description	Key ID	Key creation date	O	Actions
<input type="checkbox"/>	firebase-adminsdk-fbsvc@chat-app-27af5.iam.gserviceaccount.com	Enabled	firebase-adminsdk	Firebase Admin SDK Service Agent	No keys		11	

Solve CORS Limitation

Step 4: Paste these two commands to your terminal

- `echo '{"origin": ["*"], "method": ["GET"], "maxAgeSeconds": 3600}' > cors.json`
- `gsutil cors set cors.json gs://your-project.firebaseio.app`

 Gemini CLI is available in Cloud Shell terminal! Type `gemini` to try it. [Learn more](#)

```
Welcome to Cloud Shell! Type "help" to get started, or type "gemini" to try prompting with Gemini CLI.
Your Cloud Platform project in this session is set to chat-app-27af5.
Use `gcloud config set project [PROJECT_ID]` to change to a different project.
sphuang@cloudshell:~ (chat-app-27af5) $ echo '{"origin": ["*"], "method": ["GET"], "maxAgeSeconds": 3600}' > cors.json
sphuang@cloudshell:~ (chat-app-27af5) $ gsutil cors set cors.json gs://your-project.firebaseio.app
```

Step 5: Restart your app and you could see image

Lab

- Setting up project and app
- Setting up Firebase Auth
- Setting up Google Sign In
- Setting up Image Picker
- **Enable API in Google Cloud**
- Running the Chat App

Enable Google People API

- Go to Project Settings -> Click “View in Google Cloud”

The screenshot shows the Firebase Project Settings interface for a project named 'chat-app'. The left sidebar contains navigation options like 'Project Overview' and 'Settings'. The main content area is titled 'Project settings' and has tabs for 'General', 'Cloud Messaging', 'Integrations', 'Service accounts', 'Data privacy', 'Users and permissions', and 'Alerts'. The 'General' tab is active, showing project details such as Project name, Project ID, and Project number. A red box highlights the 'View in Google Cloud' button in the top right corner of the main content area.

Setting	Value
Project name	chat-app
Project ID	chat-app-27af5
Project number	93576281842
Environment	
This setting customizes your project for different stages of the app lifecycle	
Environment type	Unspecified
Public settings	
These settings control instances of your project shown to the public	
Public-facing name	project-93576281842

Enable Google People API

- Find APIs & Services -> Click “Enabled APIs & services”

The screenshot shows the Google Cloud console interface. On the left, a navigation menu is open, with the 'APIs & Services' option highlighted by a red box. Below this menu, a sub-menu is visible, also with 'Enabled APIs & services' highlighted by a red box. The main content area of the console shows a 'Welcome, Shih-Ping Huang' message and a 'Try Gemini Cloud Assist chat' panel. At the bottom, there are buttons for 'Create an agent', '+ Create a VM', '+ Run a query in BigQuery', and '+ Deploy an application'. The URL at the bottom of the browser window is <https://console.cloud.google.com/apis/dashboard?authuser=0&hl=en-US&project=chat-app-27af5>.

Enable Google People API

- Click “Enable APIs & services”

The screenshot shows the Google Cloud console interface. At the top, there's a search bar and navigation icons. The left sidebar is titled 'APIs & Services' and lists various options like 'Enabled APIs & services', 'Library', 'Credentials', 'OAuth consent screen', and 'Page usage agreements'. The main content area is titled 'APIs & Services' and features a prominent blue button with a red border that says '+ Enable APIs and services'. Below this, there are three monitoring charts for the date 'Jun 2':

- Traffic:** A line chart showing a sharp spike in traffic around 6:00 AM, with a scale of 0.05/s.
- Errors:** A line chart showing a small spike in errors around 6:00 AM, with a scale of 100%.
- Median latency:** A line chart showing a small spike in latency around 6:00 AM, with a scale of 1 millisecond and 0.5 milliseconds.

Enable Google People API

- Search “Google People API” and click it

The screenshot shows the Google Cloud API Library interface. At the top, there is a search bar containing the text "Google People API". Below the search bar, the results are displayed in a grid. The first result, "Google People API", is highlighted with a red box. This result includes a blue person icon, the title "Google People API", the subtitle "Google Enterprise API", and a description: "Provides access to information about profiles and contacts." To the left of the results is a filter sidebar with categories like "Visibility", "Category", and "Social".

API API Library

Navigation menu (/) / API Library / Browse

Google Cloud chat-app

Search: Google People API

API Library > "Google People API"

5 results

Filter Type to filter

- Visibility ^
- Public (5)
- Category ^
- DevOps (1)
- Google Enterprise APIs (1)
- Google Workspace (2)
- Other (2)
- Social (2)

Google People API
Google Enterprise API ⓘ
Provides access to information about profiles and contacts.

Google Forms API
Google
Reads and writes Google Forms and responses.

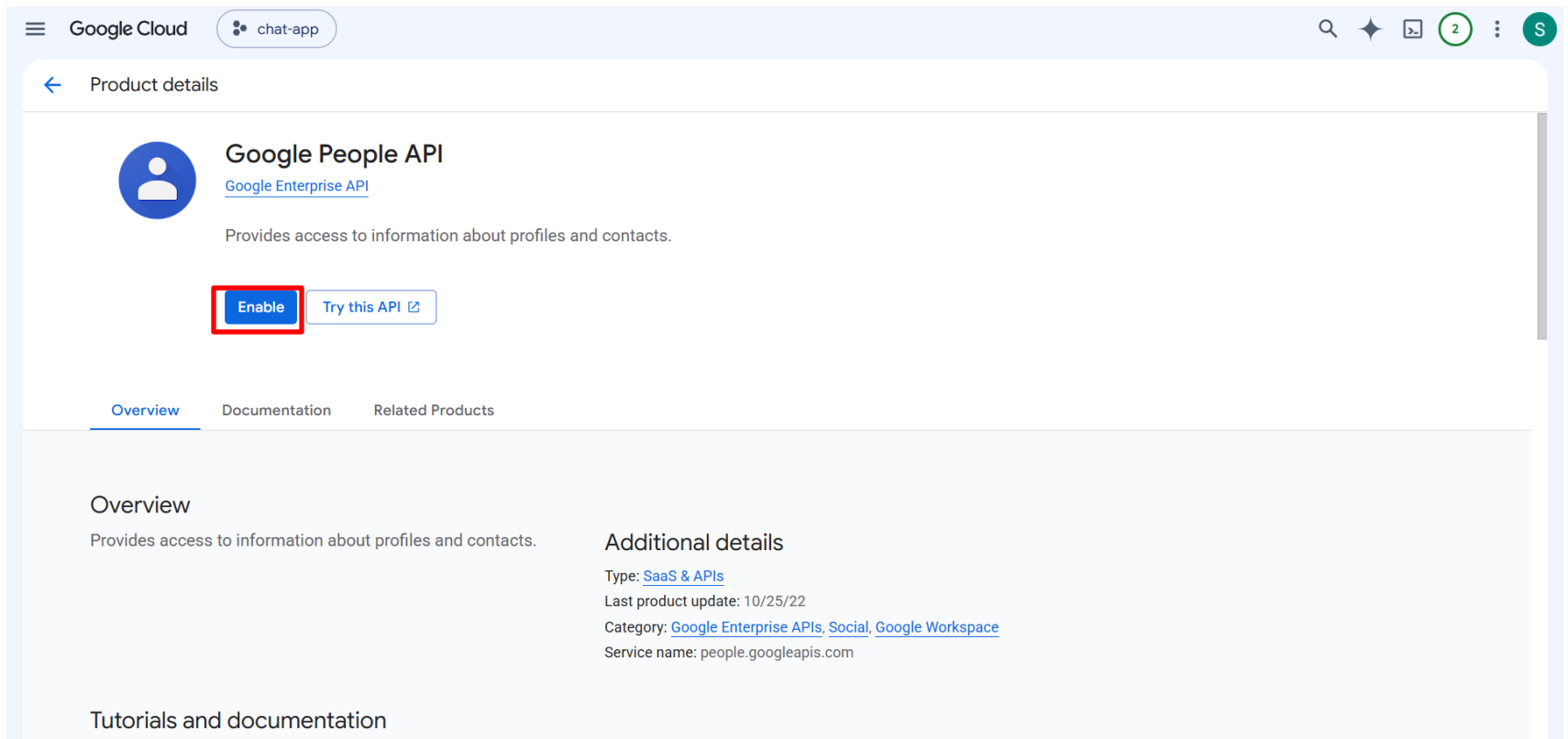
Contacts API
Google
The Contacts API lets you view, edit, and organize your contacts.

ARCore Cloud Anchor API (Legacy)
Google

Poly API
Google

Enable Google People API

- Finally, enable it



The screenshot shows the Google Cloud console interface for the Google People API. At the top, the navigation bar includes the Google Cloud logo, a search bar, and a 'chat-app' button. The main content area is titled 'Product details' and features a blue profile icon, the text 'Google People API', and a link to 'Google Enterprise API'. Below this, a description states: 'Provides access to information about profiles and contacts.' Two buttons are present: a blue 'Enable' button, which is highlighted with a red rectangular box, and a white 'Try this API' button with an external link icon. A horizontal menu below the buttons contains 'Overview', 'Documentation', and 'Related Products', with 'Overview' being the active tab. The 'Overview' section contains the same description as above. To the right, the 'Additional details' section lists: 'Type: SaaS & APIs', 'Last product update: 10/25/22', 'Category: Google Enterprise APIs, Social, Google Workspace', and 'Service name: people.googleapis.com'. At the bottom left, there is a link for 'Tutorials and documentation'.

Lab

- Setting up project and app
- Setting up Firebase Auth
- Setting up Google Sign In
- Setting up Image Picker
- Enable API in Google Cloud
- **Running the Chat App**

Running Chat App

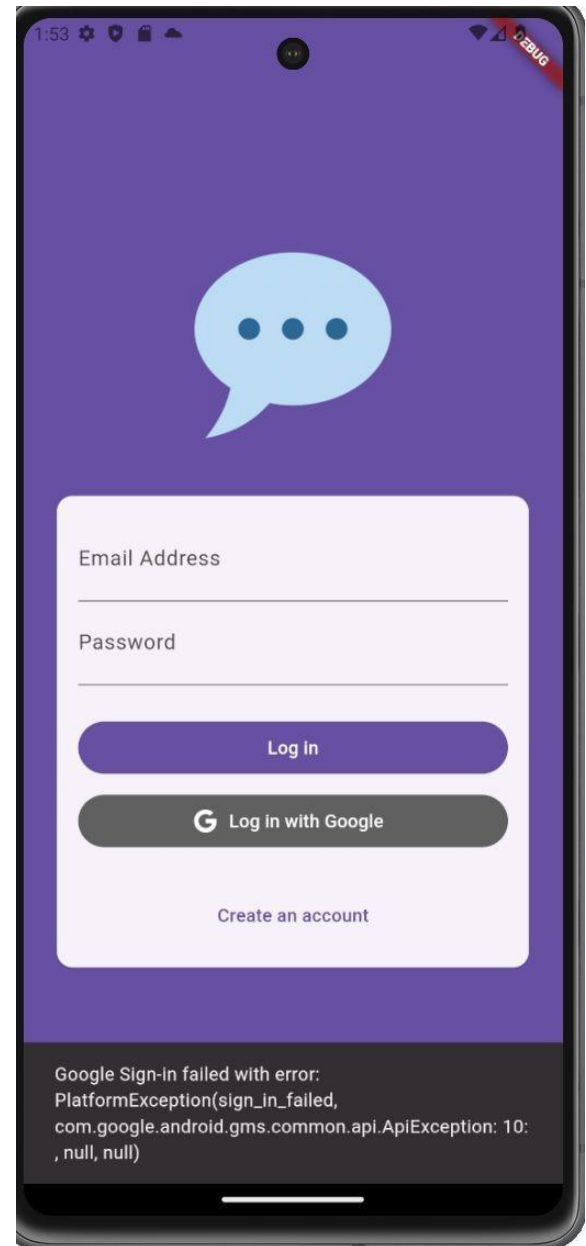
- Sign up using your email address
- Send some chat messages
- Log out, then log in with Google using same email address
 - Account linking will be triggered
- Check:
 - Image picker runs correctly, and selected file stored in Cloud Storage
 - User and Message docs created in Firestore
 - After account linking, your User doc should record two log-in methods

For android app

You will probably encounter this error when trying to use google sign in.

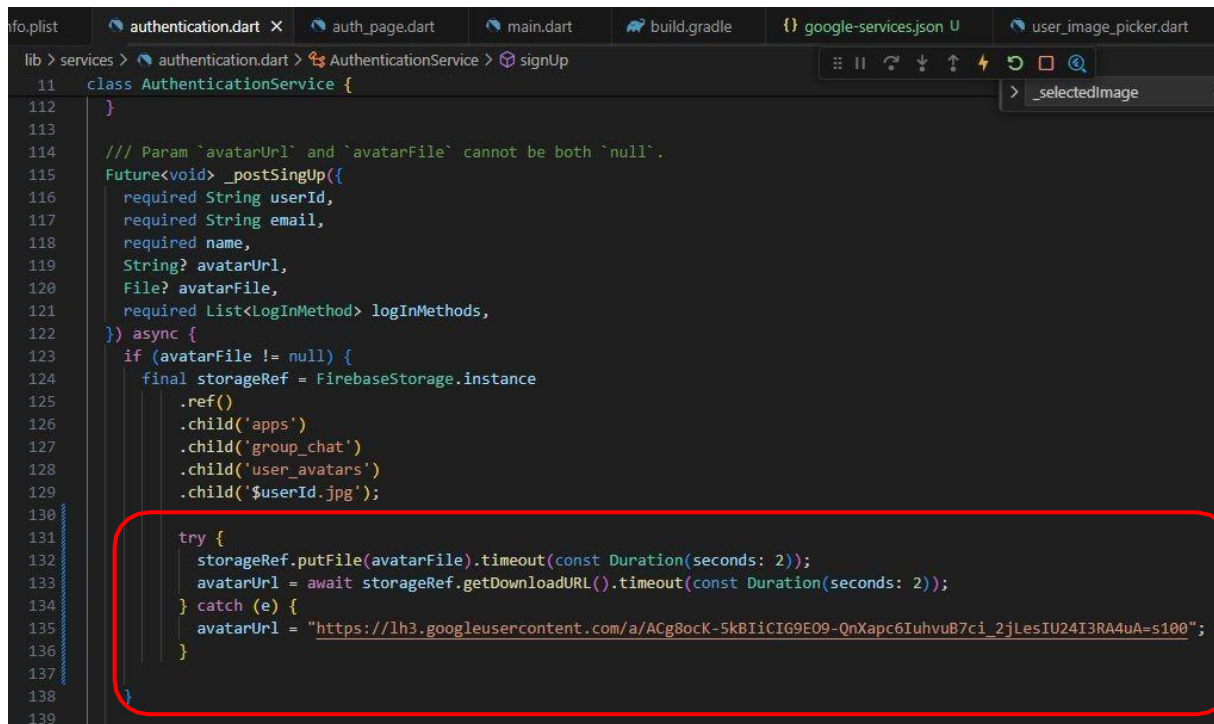
Please try following steps:

1. Go to `android/settings.gradle` add `id("com.google.gms.google-services") version "4.4.2" apply false`
2. Go to `android/app/build.gradle` add `id("com.google.gms.google-services")` in plugins



For web

Since image picker for web and app is different. There might be some error while getting avatarUrl. You can try this first:



```
lib > services > authentication.dart > AuthenticationService > signUp
11 class AuthenticationService {
112 }
113
114 /// Param `avatarUrl` and `avatarFile` cannot be both `null`.
115 Future<void> _postSingUp({
116   required String userId,
117   required String email,
118   required name,
119   String? avatarUrl,
120   File? avatarFile,
121   required List<LogInMethod> logInMethods,
122 }) async {
123   if (avatarFile != null) {
124     final storageRef = FirebaseStorage.instance
125       .ref()
126       .child('apps')
127       .child('group_chat')
128       .child('user_avatars')
129       .child('$userId.jpg');
130
131     try {
132       storageRef.putFile(avatarFile).timeout(const Duration(seconds: 2));
133       avatarUrl = await storageRef.getDownloadURL().timeout(const Duration(seconds: 2));
134     } catch (e) {
135       avatarUrl = "https://lh3.googleusercontent.com/a/ACg8ocK-5kBIiCI69E09-QnXapc6IuhvuB7ci_2jLesIU24I3RA4uA=s100";
136     }
137   }
138 }
139
```