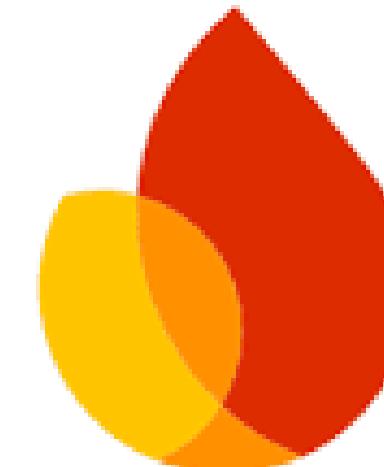


# Firebase Authentication

Trình bày: Nguyễn Trọng Nhân - Lê Duy Hoàng



# Firebase

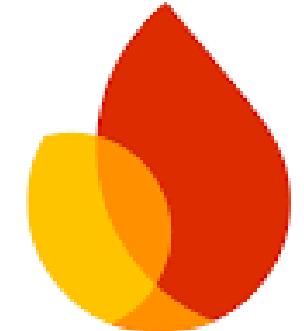
## BẢNG PHÂN CÔNG CÔNG VIỆC TRONG NHÓM

STT	HỌ VÀ TÊN	NHIỆM VỤ	KHỐI LƯỢNG
01	Nguyễn Trọng Nhân	<ul style="list-style-type: none"><li>• User management và profile</li><li>• Security best practices</li></ul>	50%
02	Lê Duy Hoàng	<ul style="list-style-type: none"><li>• Email/password authentication</li><li>• Social login (Google, Facebook)</li></ul>	50%



# NỘI DUNG

- 1. Đặt vấn đề**
- 2. Thiết lập firebase**
- 3. Xác thực bằng Email/Password**
- 4. Đăng nhập bằng Google, Facebook**
- 5. User management và profile**
- 6. Security best practices**



Firebase



# 1. Đặt vấn đề

Trong phát triển ứng dụng, việc quản lý người dùng, dữ liệu và bảo mật thường phức tạp và tốn thời gian. Firebase cung cấp giải pháp toàn diện để giải quyết những vấn đề này:

- Xác thực người dùng nhanh chóng
- Quản lý dữ liệu thời gian thực và đồng bộ giữa thiết bị.
- Bảo mật, mã hóa dữ liệu và quản lý quyền truy cập.
- Tiết kiệm thời gian phát triển, dễ mở rộng và tích hợp.

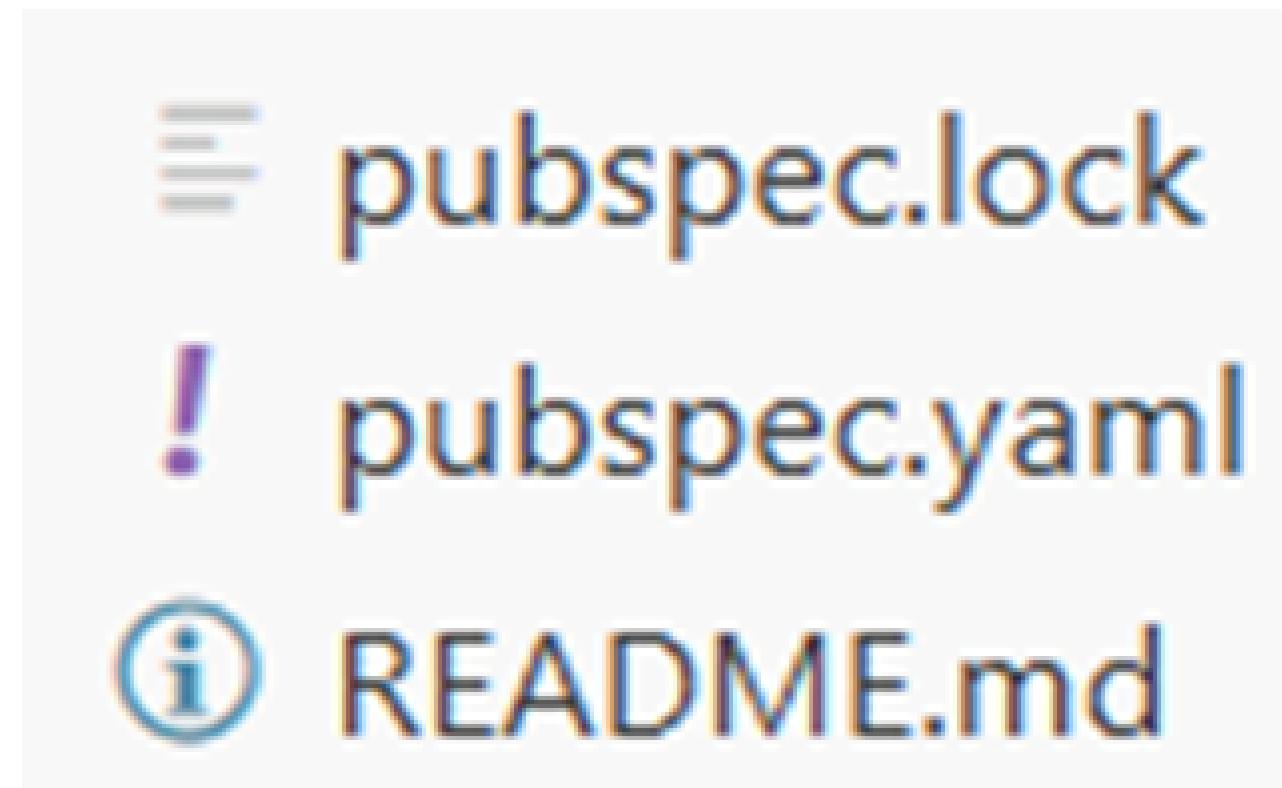


Firebase



## 2. Thiết lập firebase

Bước 1: Thêm các package vào phần dependencies của file pubspec.yaml



```
dependencies:  
  flutter:  
    sdk: flutter  
  firebase_core: ^3.0.0  
  firebase_auth: ^5.0.0  
  google_sign_in: ^6.0.0  
  flutter_facebook_auth: ^7.0.0
```

Ấn tổ hợp Ctrl+S hoặc chạy lệnh “flutter pub get”



Firebase



## 2. Thiết lập firebase

**Chạy lệnh “flutter pub outdated” để kiểm tra phiên bản của các package**

Package Name	Current	Upgradable	Resolvable	Latest
<b>direct dependencies:</b>				
firebase_auth	*5.7.0	*5.7.0	6.1.2	6.1.2
firebase_core	*3.15.2	*3.15.2	4.2.1	4.2.1
google_sign_in	*6.3.0	*6.3.0	7.2.0	7.2.0



Firebase



## 2. Thiết lập firebase

**Chạy lệnh “flutter pub outdated” để kiểm tra phiên bản của các package**

Package Name	Current	Upgradable	Resolvable	Latest
<b>direct dependencies:</b>				
firebase_auth	*5.7.0	*5.7.0	6.1.2	6.1.2
firebase_core	*3.15.2	*3.15.2	4.2.1	4.2.1
google_sign_in	*6.3.0	*6.3.0	7.2.0	7.2.0



Firebase



## 2. Thiết lập firebase

**Chạy lệnh “flutter pub outdated” để kiểm tra phiên bản của các package**

Package Name	Current	Upgradable	Resolvable	Latest
<b>direct dependencies:</b>				
firebase_auth	*5.7.0	*5.7.0	6.1.2	6.1.2
firebase_core	*3.15.2	*3.15.2	4.2.1	4.2.1
google_sign_in	*6.3.0	*6.3.0	7.2.0	7.2.0

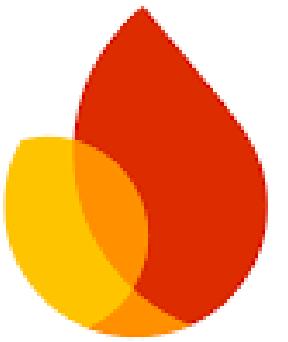


Firebase



## 2. Thiết lập firebase

**Bước 2: Chạy lệnh trong terminal “`dart pub global activate flutterfire_cli`”**  
→ Dùng để cài công cụ dòng lệnh của Firebase dành riêng cho Flutter, tên là **FlutterFire CLI**, giúp tự động cấu hình Firebase cho project Flutter, thay vì phải làm thủ công.



Firebase



## 2. Thiết lập firebase

**Bước 2:**

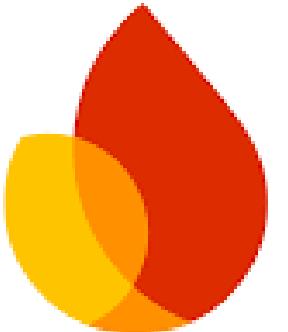
**1. Chạy lệnh trên terminal: `npm install -g firebase-tools`**

→ Cài Firebase CLI dùng cho tất cả các dự án Firebase

**2. Chạy lệnh trong terminal “`dart pub global activate flutterfire_cli`”**

Cài FlutterFire CLI, chuyên dụng cho Flutter. Giúp liên kết dự án

Flutter với Firebase, tự động tạo file cấu hình `firebase_options.dart`.



Firebase

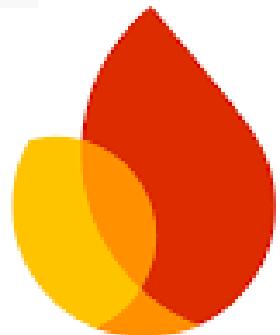


## 2. Thiết lập firebase

Có thể kiểm tra phiên bản của firebase và flutterfire bằng lệnh “firebase --version” và “flutterfire –version”

- PS D:\FlutterProject\Demoweek2\demoweek3> firebase --version  
14.24.0

- PS D:\FlutterProject\Demoweek2\demoweek3> flutterfire --version  
1.3.1

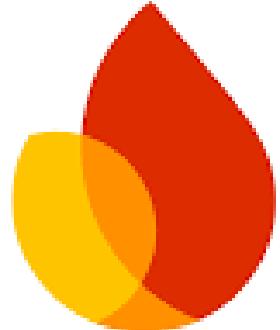
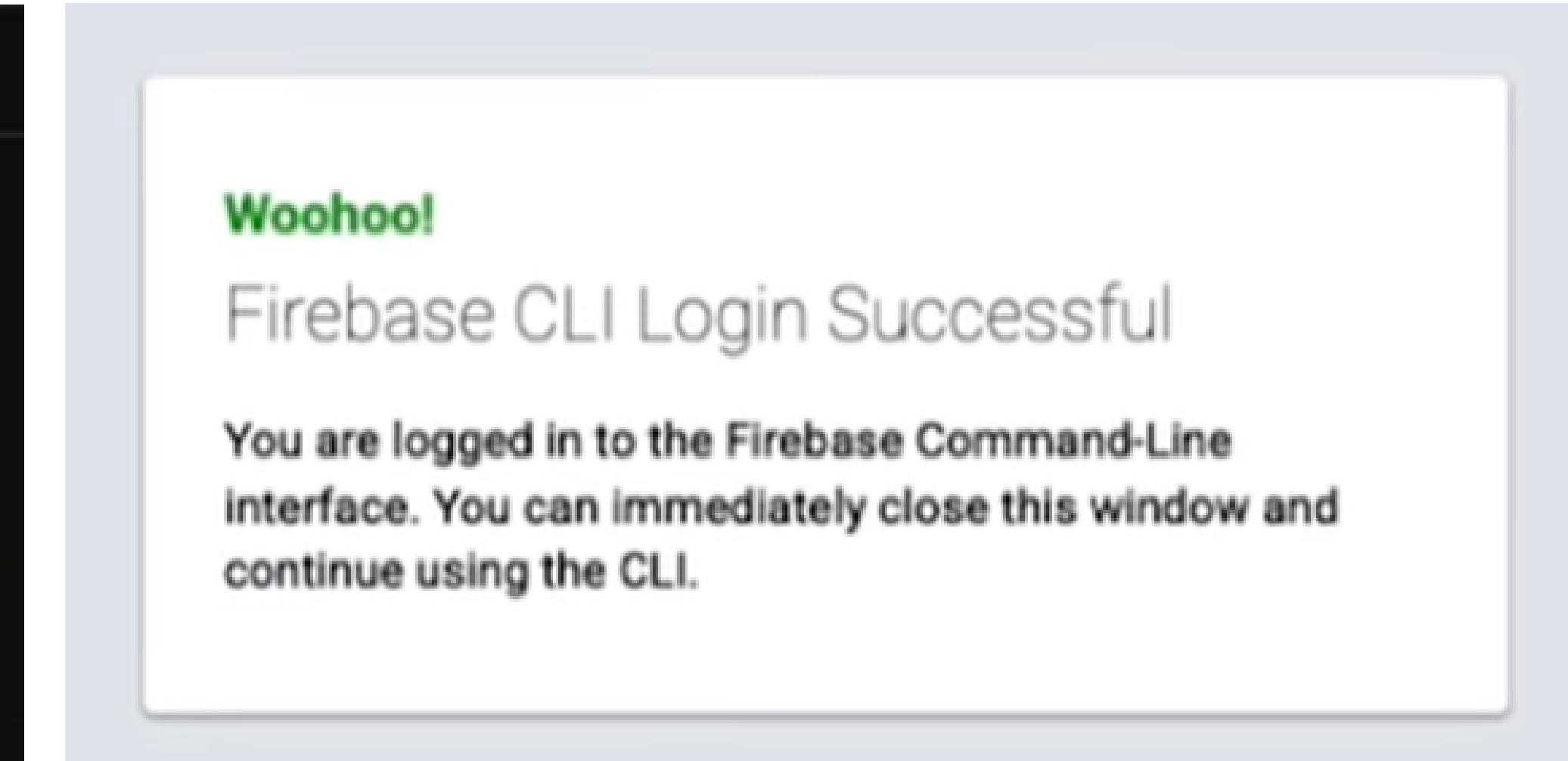
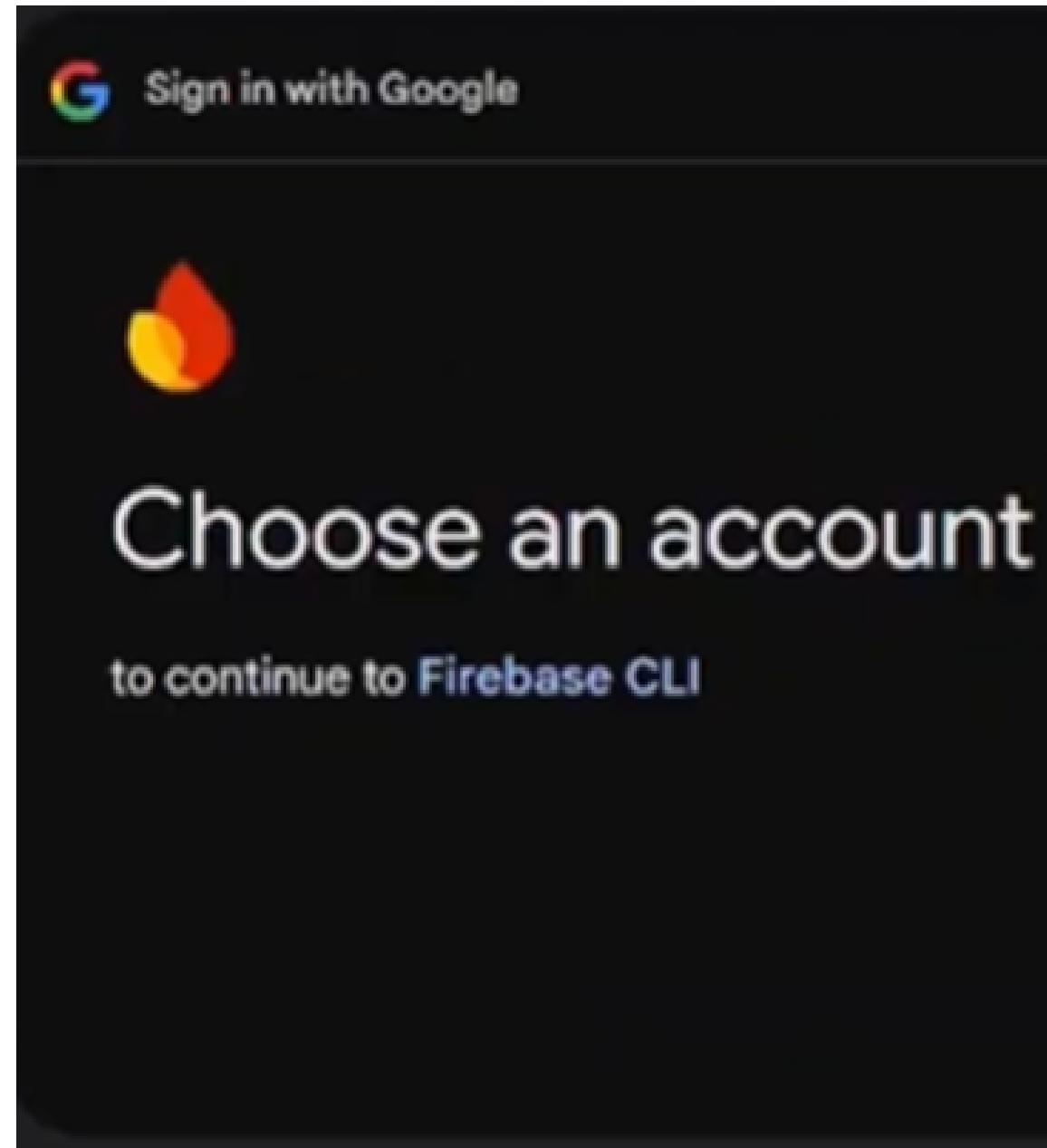


Firebase



## 2. Thiết lập firebase

Bước 3: Chạy lệnh “firebase login” để đăng nhập trên firebase



Firebase

## 2. Thiết lập firebase

### Bước 4: Chạy lệnh “flutterfire configure”

→ Tự động liên kết Flutter và Firebase + tạo file cấu hình sẵn sàng cho khởi tạo ứng dụng.

```
● PS D:\FlutterProject\Demoweek2> cd demoweek3
● PS D:\FlutterProject\Demoweek2\demoweek3> flutterfire configure
  i Found 2 Firebase projects.
  ✓ Select a Firebase project to configure your Flutter application with · fir-auth-demo-e8a01 (firebase-auth-demo)
  ✓ Which platforms should your configuration support (use arrow keys & space to select)? · android, ios, macos, web, windows
  i Firebase android app com.example.demoweek3 is not registered on Firebase project fir-auth-demo-e8a01.
  i Registered a new Firebase android app on Firebase project fir-auth-demo-e8a01.
  i Firebase ios app com.example.demoweek3 is not registered on Firebase project fir-auth-demo-e8a01.
  i Registered a new Firebase ios app on Firebase project fir-auth-demo-e8a01.
  i Firebase macos app com.example.demoweek3 registered.
  i Firebase web app demoweek3 (web) is not registered on Firebase project fir-auth-demo-e8a01.
  i Registered a new Firebase web app on Firebase project fir-auth-demo-e8a01.
  i Firebase windows app demoweek3 (windows) is not registered on Firebase project fir-auth-demo-e8a01.
  i Registered a new Firebase windows app on Firebase project fir-auth-demo-e8a01.

Firebase configuration file lib.firebaseio_options.dart generated successfully with the following Firebase apps:
```

Platform	Firebase App Id
web	1:87411107443:web:ed598ff8dfa8ee876ec06
android	1:87411107443:android:469e592446e5955d76ec06
ios	1:87411107443:ios:684186e808b83f7b76ec06
macos	1:87411107443:ios:684186e808b83f7b76ec06
windows	1:87411107443:web:e4ce68bc59076d5776ec06

Learn more about using this file and next steps from the documentation:

> <https://firebase.google.com/docs/flutter/setup>

○ PS D:\FlutterProject\Demoweek2\demoweek3>



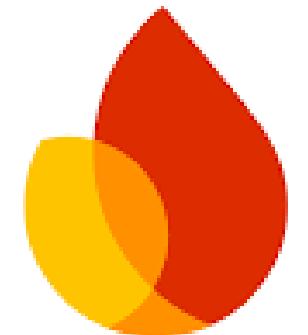
Firebase



## 2. Thiết lập firebase

**File `firebase_options.dart` được sinh ra:**

```
// File generated by FlutterFire CLI.  
// ignore_for_file: type=lint  
import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;  
import 'package:flutter/foundation.dart'  
    ||| show defaultTargetPlatform, kIsWeb, TargetPlatform;
```



Firebase

## 2. Thiết lập firebase

```
class DefaultFirebaseOptions {  
    static FirebaseOptions get currentPlatform {  
        if (kIsWeb) {  
            return web;  
        }  
        switch (defaultTargetPlatform) {  
            case TargetPlatform.android:  
                return android;  
            case TargetPlatform.iOS:  
                return ios;  
            case TargetPlatform.macOS:  
                return macos;  
            case TargetPlatform.windows:  
                return windows;  
            case TargetPlatform.linux:  
                throw UnsupportedError(  
                    'DefaultFirebaseOptions have not been configured for linux - '  
                    'you can reconfigure this by running the FlutterFire CLI again.',  
                );  
            default:  
                throw UnsupportedError(  
                    'DefaultFirebaseOptions are not supported for this platform.',  
                );  
        }  
    }  
}
```



Firebase

# 2. Thiết lập firebase

```
static const FirebaseOptions web = FirebaseOptions(  
    apiKey: 'AIzaSyADMo3F1PY_bbEiyu5gvw-4DmlWviaRfmlo',  
    appId: '1:87411107443:web:ed598ff8dfea8ee876ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    authDomain: 'fir-auth-demo-e8a01.firebaseio.com',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    measurementId: 'G-C42KT69LQR',  
);  
  
static const FirebaseOptions android = FirebaseOptions(  
    apiKey: 'AIzaSyDZ-a1P7Y7P1o9bxjGtD3Cnkpw7wnXwTX0',  
    appId: '1:87411107443:android:469e592446e5955d76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
);  
  
static const FirebaseOptions ios = FirebaseOptions(  
    apiKey: 'AIzaSyDBu3KBs94JqujKXaCJh3R2lu-1kQ_6-aQ',  
    appId: '1:87411107443:ios:684186e808b83f7b76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    iosBundleId: 'com.example.demoweek3',  
);
```

```
static const FirebaseOptions macos = FirebaseOptions(  
    apiKey: 'AIzaSyDBu3KBs94JqujKXaCJh3R2lu-1kQ_6-aQ',  
    appId: '1:87411107443:ios:684186e808b83f7b76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    iosBundleId: 'com.example.demoweek3',  
);  
  
static const FirebaseOptions windows = FirebaseOptions(  
    apiKey: 'AIzaSyADMo3F1PY_bbEiyu5gvw-4DmlWviaRfmlo',  
    appId: '1:87411107443:web:e4ce68bc59076d5776ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    authDomain: 'fir-auth-demo-e8a01.firebaseio.com',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    measurementId: 'G-RV320FRHK5',  
);  
}
```



Firebase

# 2. Thiết lập firebase

```
static const FirebaseOptions web = FirebaseOptions(  
    apiKey: 'AIzaSyADMo3F1PY_bbEiyu5gvw-4DmlWviaRfmlo',  
    appId: '1:87411107443:web:ed598ff8dfea8ee876ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    authDomain: 'fir-auth-demo-e8a01.firebaseio.com',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    measurementId: 'G-C42KT69LQR',  
);  
  
static const FirebaseOptions android = FirebaseOptions(  
    apiKey: 'AIzaSyDZ-a1P7Y7P1o9bxjGtD3Cnkpw7wnXwTX0',  
    appId: '1:87411107443:android:469e592446e5955d76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
);  
  
static const FirebaseOptions ios = FirebaseOptions(  
    apiKey: 'AIzaSyDBu3KBs94JqujKXaCJh3R2lu-1kQ_6-aQ',  
    appId: '1:87411107443:ios:684186e808b83f7b76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    iosBundleId: 'com.example.demoweek3',  
);
```

```
static const FirebaseOptions macos = FirebaseOptions(  
    apiKey: 'AIzaSyDBu3KBs94JqujKXaCJh3R2lu-1kQ_6-aQ',  
    appId: '1:87411107443:ios:684186e808b83f7b76ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    iosBundleId: 'com.example.demoweek3',  
);  
  
static const FirebaseOptions windows = FirebaseOptions(  
    apiKey: 'AIzaSyADMo3F1PY_bbEiyu5gvw-4DmlWviaRfmlo',  
    appId: '1:87411107443:web:e4ce68bc59076d5776ec06',  
    messagingSenderId: '87411107443',  
    projectId: 'fir-auth-demo-e8a01',  
    authDomain: 'fir-auth-demo-e8a01.firebaseio.com',  
    storageBucket: 'fir-auth-demo-e8a01.firebaseiostorage.app',  
    measurementId: 'G-RV320FRHK5',  
);  
}
```



Firebase



## 2. Thiết lập firebase

Giao diện ban đầu trên Firebase khi đã kết nối với Flutter

The screenshot shows the Firebase Project Overview interface. On the left, there's a sidebar with a navigation bar at the top containing a home icon, 'Project Overview' (which is highlighted in blue), and a gear icon. Below this are sections for 'Project shortcuts', 'Authentication' (with a user icon), 'Product categories', and dropdown menus for 'Build', 'Run', 'Analytics', and 'AI'. The main content area has a header 'firebase-auth-demo ▾' and a title 'firebase-auth-demo' with a 'Spark plan' button. It displays '2 apps visible (max 3)' and '4 apps in project'. A table lists four apps: 'demoweek3 (android) com.example.demoweek3', 'demoweek3 (ios) com.example.demoweek3', 'demoweek3 (web) Web App', and 'demoweek3 (windows) Web App'. Each app entry includes an 'eye' icon, a device-specific icon (Android, iOS, browser, or Windows), the app name, its package name, and a gear icon for settings. At the bottom, a note states: 'Note: all apps are included in project-level metrics below, but only selected apps above are broken out'. A 'No data' message is also present.



Firebase



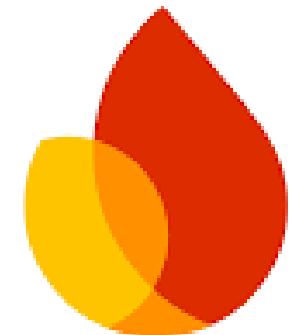
### 3. Xác thực bằng Email/Password

Bước 1: Trong file main.dart import các package và khởi tạo Firebase

```
import 'package:flutter/material.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'firebase_options.dart';
import 'login_page.dart';
import 'home_page.dart';
```

Run | Debug | Profile

```
void main() async {
    WidgetsFlutterBinding.ensureInitialized();
    await Firebase.initializeApp(
        options: DefaultFirebaseOptions.currentPlatform,
    );
    runApp(const MyApp());
}
```



Firebase



# 3. Xác thực bằng Email/Password

Bước 2: Vào Firebase chọn Build Authentication Sign-in method.  
Chọn Add new provider “Email/Password”

The screenshot shows the Firebase console interface for managing authentication providers. On the left, there's a sidebar with 'Project Overview' and 'Authentication' selected. The main area is titled 'Authentication' and has tabs for 'Users', 'Sign-in method' (which is active), 'Templates', 'Usage', 'Settings', and 'Extensions'. A sub-section titled 'Sign-in providers' lists several providers: 'Email/Password' (selected), 'Google', 'Facebook', 'Play Games', 'Phone', 'Game Center', 'Apple', 'GitHub', 'Anonymous', 'Microsoft', 'Twitter', and 'Yahoo'. A blue button labeled 'Add new provider' is visible at the top right of this list. At the bottom, there are sections for 'Custom providers' (OpenID Connect, SAML) and a note that says 'Email/Password Enabled'.

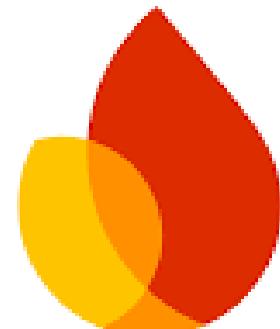




# 3. Xác thực bằng Email/Password

Bước 2: Vào Firebase chọn Build Authentication Sign-in method.  
Chọn Add new provider “Email/Password”

The screenshot shows the Firebase console interface for a project named "firebase-auth-demo". The left sidebar has sections for Project Overview, Authentication (which is selected), Product categories (Build, Run, Analytics, AI), and Related development tools (Firebase Studio). The main content area is titled "Authentication" and has tabs for Users, Sign-in method (which is active), Templates, Usage, Settings, and Extensions. Below this, there's a "Sign-in providers" section with a table header "Provider" and "Status". A blue button "Add new provider" is visible. A modal window titled "Select a sign-in provider (Step 1 of 2)" is overlaid. It contains three columns: "Native providers" (Email/Password, Phone, Anonymous), "Additional providers" (Google, Facebook, Play Games, Game Center, Apple, GitHub, Microsoft, Twitter, Yahoo), and "Custom providers" (OpenID Connect, SAML). At the bottom of the modal, there are "Email/Password" and "Enabled" buttons.



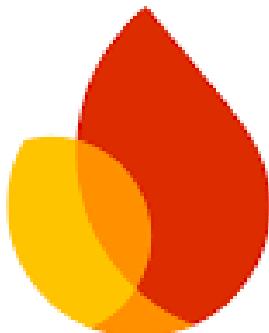


# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:google_sign_in/google_sign_in.dart';
import 'package:flutter_facebook_auth/flutter_facebook_auth.dart';
import 'package:logger/logger.dart';
import 'package:flutter/foundation.dart' show kIsWeb;

class AuthService {
  final FirebaseAuth _auth = FirebaseAuth.instance;
  final Logger _logger = Logger();
```

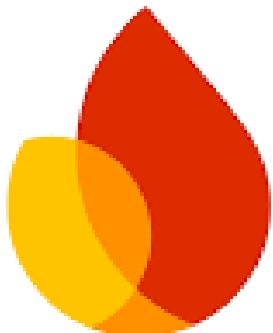




# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
Future<User?> registerWithEmail(String email, String password) async {
  try {
    UserCredential userCredential =
        await _auth.createUserWithEmailAndPassword(
      email: email,
      password: password,
    );
    return userCredential.user;
  } on FirebaseAuthException catch (e) {
    _logger.e('Email Registration Error: ${e.message}');
    return null;
  } catch (e) {
    _logger.e('Unexpected Error: $e');
    return null;
  }
}
```





# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
Future<User?> signInWithEmailAndPassword(String email, String password) async {
  try {
    UserCredential userCredential =
        await _auth.signInWithEmailAndPassword(
      email: email,
      password: password,
    );
    return userCredential.user;
  } on FirebaseAuthException catch (e) {
    _logger.e('Email Sign In Error: ${e.message}');
    return null;
  } catch (e) {
    _logger.e('Unexpected Error: $e');
    return null;
  }
}
```





# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
Future<User?> signInWithEmailAndPassword(String email, String password) async {
  try {
    UserCredential userCredential =
        await _auth.signInWithEmailAndPassword(
      email: email,
      password: password,
    );
    return userCredential.user;
  } on FirebaseAuthException catch (e) {
    _logger.e('Email Sign In Error: ${e.message}');
    return null;
  } catch (e) {
    _logger.e('Unexpected Error: $e');
    return null;
  }
}
```

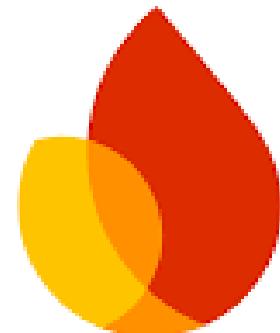




# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
Future<User?> signInWithEmailAndPassword(String email, String password) async {
  try {
    UserCredential userCredential =
        await _auth.signInWithEmailAndPassword(
      email: email,
      password: password,
    );
    return userCredential.user;
  } on FirebaseAuthException catch (e) {
    _logger.e('Email Sign In Error: ${e.message}');
    return null;
  } catch (e) {
    _logger.e('Unexpected Error: $e');
    return null;
  }
}
```

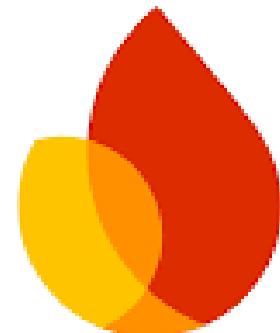




# 3. Xác thực bằng Email/Password

## Bước 3: Xây dựng file auth\_service.dart

```
Future<User?> signInWithEmailAndPassword(String email, String password) async {
  try {
    UserCredential userCredential =
        await _auth.signInWithEmailAndPassword(
      email: email,
      password: password,
    );
    return userCredential.user;
  } on FirebaseAuthException catch (e) {
    _logger.e('Email Sign In Error: ${e.message}');
    return null;
  } catch (e) {
    _logger.e('Unexpected Error: $e');
    return null;
  }
}
```





# 3. Xác thực bằng Email/Password

Demo kết quả: Giao diện đăng ký

Firebase Auth Demo    localhost:50694

## Đăng Ký

### Tạo Tài Khoản Mới

Sử dụng Email & Mật khẩu hoặc Đăng nhập xã hội

Địa chỉ Email —

Mật khẩu —

**ĐĂNG KÝ**

[Đã có tài khoản? Đăng nhập](#)

HOẶC

[Đăng nhập với Google](#)

[Đăng nhập với Facebook](#)



Firebase



# 3. Xác thực bằng Email/Password

Demo kết quả: tạo một tài khoản mới, sẽ được lưu vào phần Authentication users

firebase-auth-demo ▾

## Authentication

Users Sign-in method Templates Usage Settings Extensions

**i** The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.

Identifier	Providers	Created ↓	Signed In	User UID
bihoang0044@gmail.co...		Nov 9, 2025	Nov 9, 2025	y6PgpzHzfmRoz9MLV5U2cdC...
Idhoang170104@gmail....		Nov 8, 2025	Nov 8, 2025	YPwtBdPyekhFTJIBB2mKhX9...
admin1@gmail.com		Nov 8, 2025	Nov 8, 2025	usNcgV6VeRMb01qeNwZ6g6...
admin@gmail.com		Nov 6, 2025	Nov 10, 2025	cCFqUWRHOPaPvHA21lypwy...

Rows per page: 50 < 1 - 4 of 4 >



Firebase



# 3. Xác thực bằng Email/Password

Demo kết quả: Giao diện đăng nhập

Đăng Nhập

Chào mừng trở lại

Sử dụng Email & Mật khẩu hoặc Đăng nhập xã hội

Địa chỉ Email

Mật khẩu

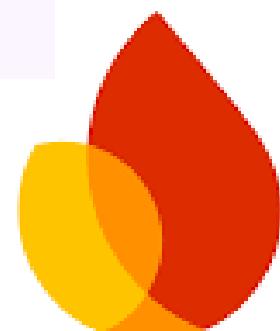
**ĐĂNG NHẬP**

Chưa có tài khoản? [Đăng ký ngay](#)

HOẶC

Đăng nhập với Google

Đăng nhập với Facebook

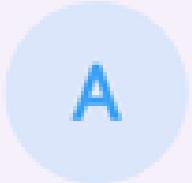




# 3. Xác thực bằng Email/Password

Demo kết quả: Sau khi đăng nhập thành công

Trang Chủ Logout



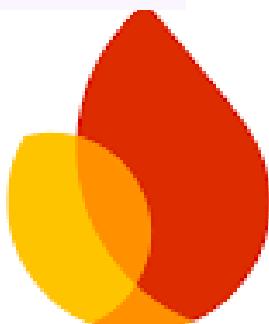
Xin chào, admin@gmail.com!  
admin@gmail.com

---

ID Người dùng (UID)  
cCFqUWRHOPaPvHA21lypwykSX893

Các Chức Năng Chính:

- [!\[\]\(f92dd4da798b4dcd89fc51d3c718915d\_img.jpg\) Quản lý Dữ liệu >](#)
- [!\[\]\(6c67751db710087d3e7dab80cdf55154\_img.jpg\) Cài đặt Tài khoản >](#)



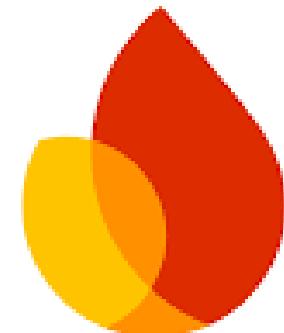


# 4. Đăng nhập bằng Google, Facebook

## Đối với Google

**Bước 1: Vào Firebase chọn Build Authentication Sign-in method.  
Chọn Add new provider “Google”**

The screenshot shows the 'Add new provider' dialog for Google. At the top, there is a 'Google' logo and an 'Enable' toggle switch which is turned on. Below the toggle, a note states: 'Important: To enable Google sign-in for your Android apps, you must provide the SHA-1 release fingerprint for each app (go to Project Settings > Your apps section)'. A modal window is open, prompting the user to 'Update the project-level setting below to continue'. It contains two fields: 'Public-facing name for project' with the value 'project-87411107443' and 'Support email for project' with the value 'bihoang0044@gmail.com'. There are also sections for 'Safelist client IDs from external projects (optional)' and 'Web SDK configuration'. At the bottom of the dialog are 'Delete provider', 'Cancel', and 'Save' buttons.





# 4. Đăng nhập bằng Google, Facebook

## Bước 2: Bật Enable cho People API

The screenshot shows the Google Cloud Platform's API Services interface. The top navigation bar includes the Google Cloud logo, the project name "firebase-auth-demo", a search bar, and various navigation icons. The main menu on the left is set to "APIs & Services". The "Enabled APIs & services" section is selected, showing the "People API" listed. The "People API" details page is displayed on the right, featuring a blue user icon, the API name "People API", a description about profile and contact access, and the note "By Google Enterprise API". Below this, there are columns for "Service name" (people.googleapis.com), "Type" (Public API), "Status" (Enabled), "Documentation" (with a "Learn more" link), and "Explore" (with a "Try in API Explorer" link). A vertical sidebar on the right contains links for "Library", "Credentials", "OAuth consent screen", and "Page usage agreements".

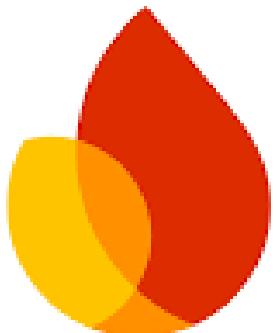




## 4. Đăng nhập bằng Google, Facebook

Bước 3: Thêm Web client ID được cấp bởi Firebase vào file web/index.html

```
<!--Google-->  
<meta name="google-signin-client_id" content="87411107443-dv0scbsioue4qbnbvds4lvv102ntpauh.apps.googleusercontent.com">
```





# 4. Đăng nhập bằng Google, Facebook

## Bước 4: Xây dựng file auth\_service.dart

```
Future<User?> signInWithGoogle() async {
  try {
    final GoogleSignInAccount? googleUser = await GoogleSignIn(
      clientId: '87411107443-dv0scbsioue4qbnbvds4lvv102ntpauh.apps.googleusercontent.com',
    ).signIn();
    if (googleUser == null) return null;

    final GoogleSignInAuthentication googleAuth =
      await googleUser.authentication;

    final credential = GoogleAuthProvider.credential(
      accessToken: googleAuth.accessToken,
      idToken: googleAuth.idToken,
    );

    UserCredential userCredential =
      await _auth.signInWithCredential(credential);
    return userCredential.user;
  } catch (e) {
    _logger.e('Google Sign In Error: $e');
    return null;
  }
}
```





# 4. Đăng nhập bằng Google, Facebook

## Demo kết quả: Xác thực đăng nhập

The image shows two side-by-side screenshots of a web browser window. The left screenshot displays the Google sign-in page with a blue header bar at the top. The right screenshot shows the two-step verification screen.

**Left Screenshot (Google Sign-In Page):**

- Header: "Đăng nhập - Tài khoản Google - Google Chrome".
- URL: "accounts.google.com/v3/signin/identifier?oparams=%253Fenable\_gr..."
- Title: "Đăng nhập bằng Google".
- Text: "Đăng nhập".
- Text: "Tiếp tục tới project-87411107443".
- Input Field: "Email hoặc số điện thoại".
- Text: "Bạn quên địa chỉ email?".
- Buttons: "Tạo tài khoản" and "Tiếp theo".
- Footer: "Tiếng Việt", "Trợ giúp", "Quyền riêng tư", "Điều khoản".

**Right Screenshot (Two-Step Verification Screen):**

- Header: "Xác minh 2 bước Để giữ an toàn cho tài khoản của bạn, Google...".
- URL: "accounts.google.com/v3/signin/challenge/dp?TL=ANzgctS95AX5HZSK...".
- Title: "Đăng nhập bằng Google".
- Text: "Xác minh 2 bước".
- Text: "Để giữ an toàn cho tài khoản của bạn, Google muốn đảm bảo rằng bạn chính là người đang cố đăng nhập".
- Text: "bihoang0044@gmail.com".
- Large Number: "16".
- Text: "Kiểm tra Redmi Note 10 5G của bạn".
- Text: "Google đã gửi thông báo đến Redmi Note 10 5G của bạn. Nhấn vào Có trên thông báo, rồi nhấn vào 16 trên điện thoại để xác minh danh tính của bạn.". A blue checkmark icon is next to the word "Có".
- Checkboxes: One checkbox is checked with the text "Không hỏi lại trên thiết bị này".





# 4. Đăng nhập bằng Google, Facebook

Demo kết quả: Đăng nhập thành công

Trang Chủ

ĐANG XỬ LÝ

H

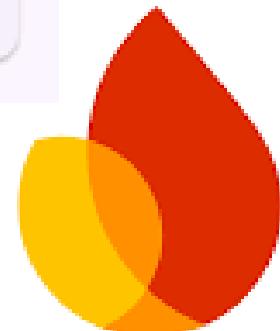
Xin chào, Hoàng Duy!

bihuong0044@gmail.com

ID Người dùng (UID)  
y6PgpzHzfmRoz9MLV5U2cdCVbwI

Các Chức Năng Chính:

- Quản lý Dữ liệu >
- Cài đặt Tài khoản >



Firebase



# 4. Đăng nhập bằng Google, Facebook

Demo kết quả: Lưu lại đăng nhập trên Firebase

The screenshot shows the Firebase console interface for the project "firebase-auth-demo". The left sidebar has sections for Project Overview, Authentication (which is selected and highlighted in blue), Product categories, Build, Run, Analytics, and AI. A "Related development tools" section includes a link to "Firebase Studio". At the bottom are links for "Spark" and "Upgrade". The main content area is titled "Authentication" and shows a table of users. The table has columns for Identifier, Providers, Created, Signed In, and User UID. There are four users listed:

Identifier	Providers	Created	Signed In	User UID
bihoang0044@gmail.co...	G	Nov 9, 2025	Nov 10, 2025	y6PgpzHzfmRoz9MLV5U2cdC...
ldhoang170104@gmail....	G	Nov 8, 2025	Nov 8, 2025	YPwtBdPyekhFTJIBB2mKhX9...
admin1@gmail.com		Nov 8, 2025	Nov 8, 2025	usNcgV6VeRMb01qeNwZ6g6...
admin@gmail.com		Nov 6, 2025	Nov 10, 2025	cCFqUWRHOPaPvHA21lypwy...

A message at the top right of the table area says: "The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps." Below the table are buttons for "Add user", a trash icon, and a more options icon. At the bottom are pagination controls for "Rows per page: 50" and "1 – 4 of 4".



Firebase



# 4. Đăng nhập bằng Google, Facebook

## Đối với Facebook

**Bước 1: Tạo một ứng dụng quản lý “Xác thực và yêu cầu dữ liệu từ người dùng qua phương thức Đăng nhập bằng Facebook”**

Meta

Tài liệu Công cụ Hỗ trợ Ứng dụng Hành động cần làm Q Search

Auth\_demo

Bảng điều khiển

Hành động cần thực hiện

Trường hợp sử dụng

Xét duyệt

Đăng

Dã hủy đăng

Cài đặt ứng dụng

Vai trò trong ứng dụng

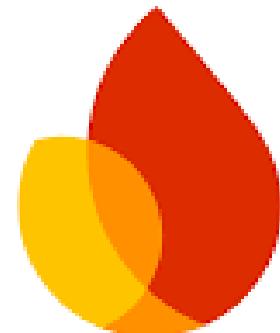
Hộp thư thông báo

Bảng điều khiển

Tùy chỉnh ứng dụng và các yêu cầu

- Tùy chỉnh trường hợp sử dụng Xác thực và yêu cầu dữ liệu từ người dùng qua phương thức Đăng nhập bằng Facebook >
- Xem lại và hoàn tất các yêu cầu thử nghiệm >
- Xác minh doanh nghiệp >
- Xét duyệt ứng dụng >
- Kiểm tra để đảm bảo rằng bạn đã hoàn tất mọi yêu cầu rồi phát hành ứng dụng. >

Thêm trường hợp sử dụng



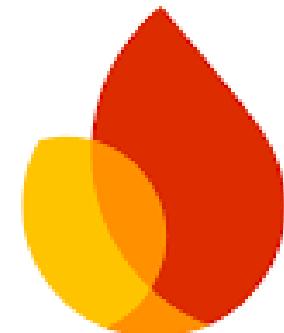


## 4. Đăng nhập bằng Google, Facebook

Bước 2: Vào Firebase chọn Build Authentication Sign-in method.

Chọn Add new provider “Facebook”, cập nhật App ID và App Secret

The screenshot shows the 'Facebook' provider configuration in the Firebase Authentication console. The 'Enable' switch is turned on. The 'App ID' field contains '1384689683085449'. The 'App secret' field contains '940d8003758ba9429238e75c45b12f9d'. Below these fields, a note says 'To complete set up, add this OAuth redirect URI to your Facebook app configuration.' followed by a 'Learn more' link. A copyable URL 'https://fir-auth-demo-e8a01.firebaseio.com/\_/auth/handler' is provided with a copy icon. At the bottom, there are 'Delete provider', 'Cancel', and 'Save' buttons.



Firebase



# 4. Đăng nhập bằng Google, Facebook

## Bước 3: Cập nhật URL của Firebase vào Meta Developer

### URI chuyển hướng OAuth hợp lệ

redirect\_uri được chỉ định theo cách thủ công để dùng với tính năng Đăng nhập trên web phải khớp chính xác với một trong những URI tại đây. Danh sách này cũng được dùng với JavaScript SDK cho trình duyệt trong ứng dụng để chặn cửa sổ bật lên. [?]

[https://fir-auth-demo-e8a01.firebaseioapp.com/\\_/auth/handler](https://fir-auth-demo-e8a01.firebaseioapp.com/_/auth/handler) X

Sao chép vào bảng nhớ tạm

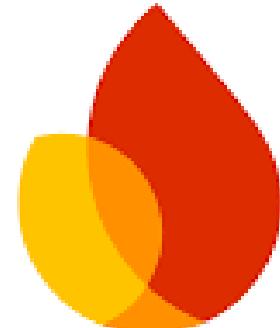
### 1. Cho chúng tôi biết về trang web của bạn

Cho chúng tôi biết URL của trang web của bạn là gì.

URL trang web

<http://localhost/>

Save



Firebase



# 4. Đăng nhập bằng Google, Facebook

## Bước 4: Thêm App ID vào file web/index.html

```
<!-- Facebook SDK -->
<script>
    // Tạo promise để đợi FB SDK load xong
    window.fbLoaded = new Promise(resolve) => {
        window.fbAsyncInit = function() {
            FB.init({
                appId          : '1384689683085449',
                cookie         : true,
                xfbml          : true,
                version        : 'v18.0'
            });
            console.log('✓ Facebook SDK loaded successfully');
            resolve();
        };
    });

```





## 4. Đăng nhập bằng Google, Facebook

### Bước 4: Thêm App ID vào file web/index.html

```
// Load Facebook SDK
(function(d, s, id){
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) {return;}
  js = d.createElement(s); js.id = id;
  js.src = "https://connect.facebook.net/en_US/sdk.js";
  fjs.parentNode.insertBefore(js, fjs);
}(document, 'script', 'facebook-jssdk'));
</script>
```





# 4. Đăng nhập bằng Google, Facebook

## Bước 5: Xây dựng file auth\_service.dart

```
Future<User?> signInWithFacebook() async {
  try {
    if (kIsWeb) {
      // CHO WEB: Dùng Firebase signInWithPopup
      _logger.i('Using Web Facebook login with Firebase popup...');
      FacebookAuthProvider facebookProvider = FacebookAuthProvider();
      facebookProvider.addScope('email');
      facebookProvider.addScope('public_profile');

      UserCredential userCredential =
          await _auth.signInWithPopup(facebookProvider);
      _logger.i('Facebook login success: ${userCredential.user?.email}');
      return userCredential.user;
    }
  } catch (e) {
    _logger.e('Error during Facebook login: $e');
  }
}
```





# 4. Đăng nhập bằng Google, Facebook

## Bước 5: Xây dựng file auth\_service.dart

```
    } else {
      _logger.i('Using Mobile Facebook login...');

      final LoginResult result = await FacebookAuth.instance.login(
        permissions: ['email', 'public_profile'],
      );

      if (result.status == LoginStatus.success) {
        final accessToken = result.accessToken?.tokenString;

        if (accessToken == null || accessToken.isEmpty) {
          _logger.e('Facebook access token is null or empty');
          return null;
        }

        final OAuthCredential credential =
          FacebookAuthProvider.credential(accessToken);

        UserCredential userCredential =
          await _auth.signInWithCredential(credential);
      }
    }
  }
}
```





# 4. Đăng nhập bằng Google, Facebook

## Bước 5: Xây dựng file auth\_service.dart

```
        _logger.i('Facebook login success: ${userCredential.user?.email}');
    return userCredential.user;
} else if (result.status == LoginStatus.cancelled) {
    _logger.w('Facebook login cancelled by user.');
    return null;
} else {
    _logger.e('Facebook login failed: ${result.message}');
    return null;
}
} catch (e) {
    _logger.e('Facebook Sign In Error: $e');
    return null;
}
}
```





# 4. Đăng nhập bằng Google, Facebook

## Demo kết quả

Đăng nhập Facebook - Google Chrome  
facebook.com/login.php?skip\_api\_login=1&api\_key=13846896830...

facebook Tạo tài khoản mới

Đăng nhập Facebook

Email hoặc số điện thoại

Mật khẩu

**Đăng nhập**

Bạn quên tài khoản ư? · Đăng ký Facebook  
Lúc khác

Facebook - Google Chrome  
facebook.com/auth\_platform/afad/?apc=Adk9ZyeJ2ApNluOd...

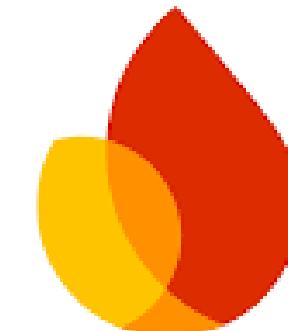
facebook

Duy Hoàng · Facebook  
**Kiểm tra thông báo của bạn**

Chúng tôi đã gửi thông báo đến Máy tính Windows, Xiaomi Redmi Note 10 5G và tài khoản Instagram của bạn.

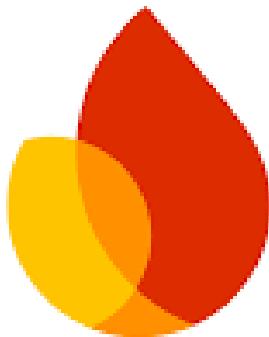
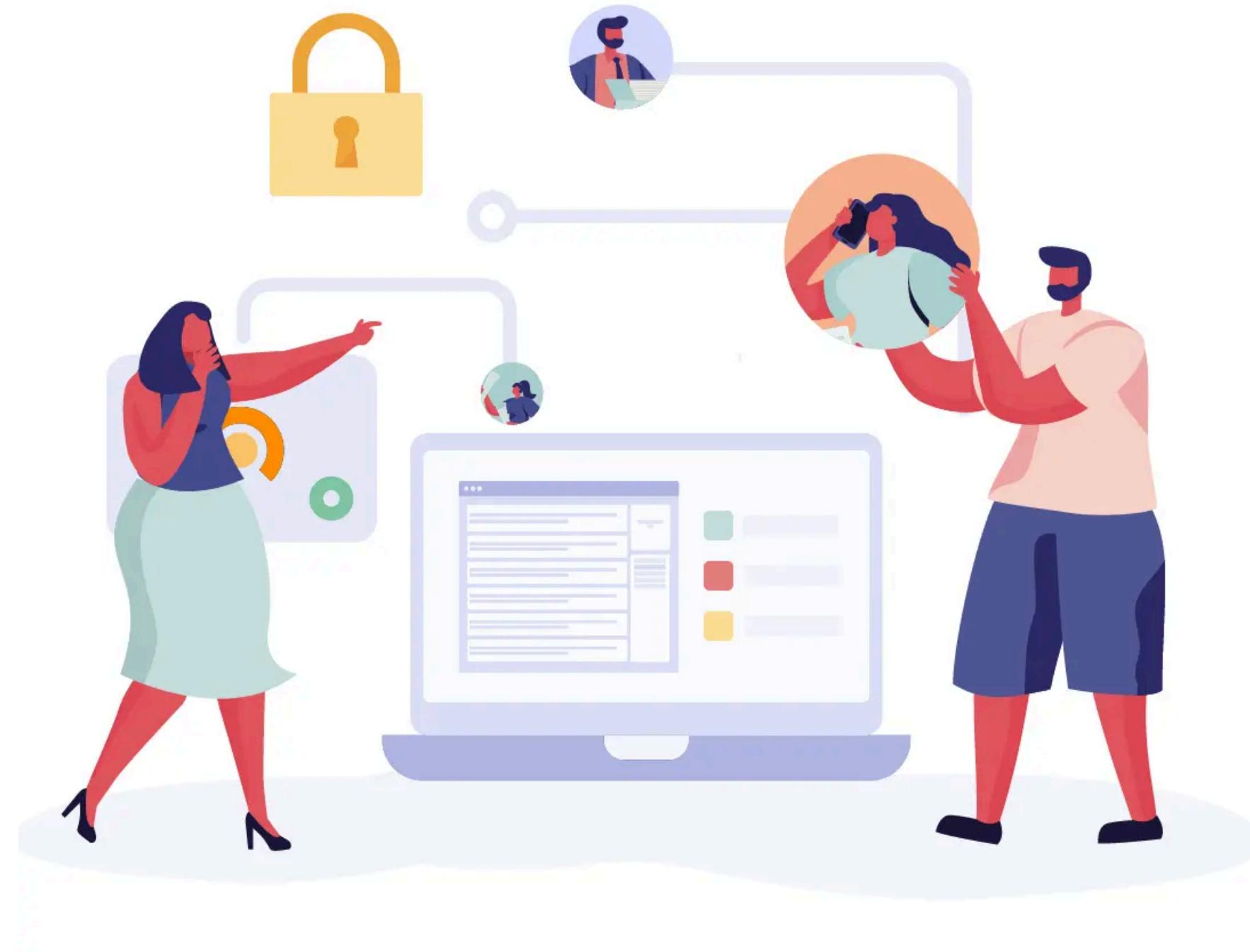
Hãy xem thông báo tại đó rồi phê duyệt lần đăng nhập để tiếp tục.

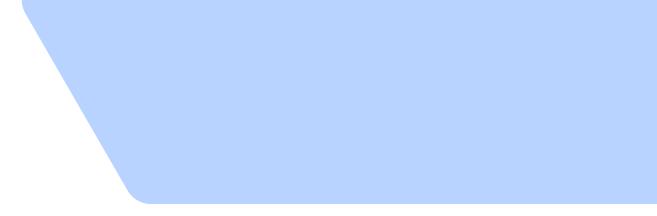
Đang chờ phê duyệt  
Có thể vài phút nữa bạn mới nhận được thông báo.





# 5. User management và profile



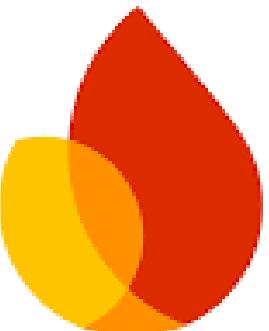


## 5.1. Khái niệm và tầm quan trọng

Quản lý người dùng là tập hợp các quy trình, công nghệ và chính sách nhằm kiểm soát, theo dõi và bảo vệ thông tin người dùng trong một hệ thống ứng dụng.

Bao gồm các hoạt động chính:

- Xác thực
- Ủy quyền
- Quản lý hồ sơ
- Quản lý phiên
- Bảo mật thông tin

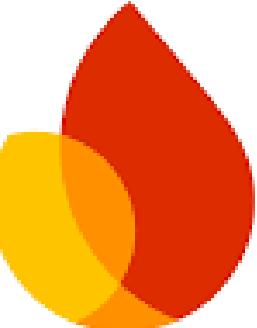




## 5.1. Khái niệm và tầm quan trọng

Hiện nay, các vi phạm dữ liệu có liên quan đến quản lý người dùng kém hiệu quả ngày càng nhiều, cho thấy tầm quan trọng của việc xây dựng hệ thống quản lý người dùng vững chắc:

- Bảo mật và An toàn dữ liệu
- Nâng cao trải nghiệm Người dùng
- Tuân thủ Pháp luật
- Phân tích và Tối ưu hóa dữ liệu người dùng



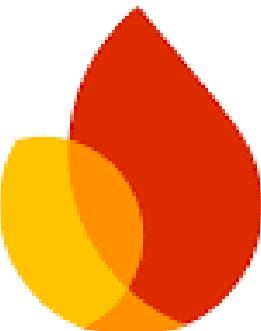


## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Các cách khởi tạo người dùng mới:

- Gọi phương thức `createUserWithEmailAndPassword()`.

```
await FirebaseAuth.instance.createUserWithEmailAndPassword(  
    email: _emailController.text.trim(),  
    password: _passwordController.text,  
);
```

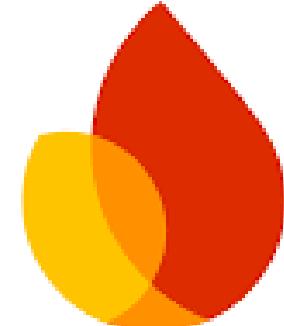
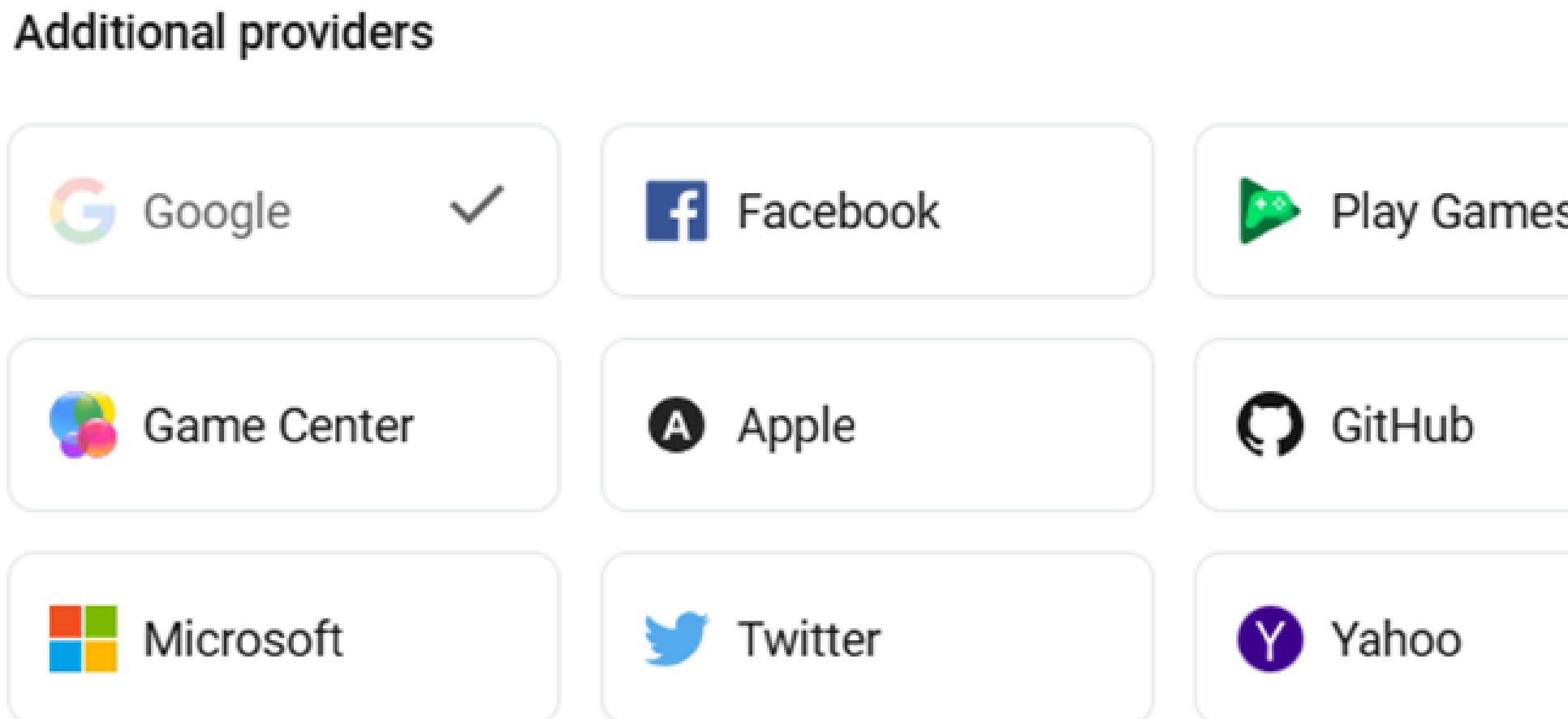




## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Các cách khởi tạo người dùng mới:

- Đăng nhập bằng nhà cung cấp danh tính liên kết mà Flutter cung cấp.



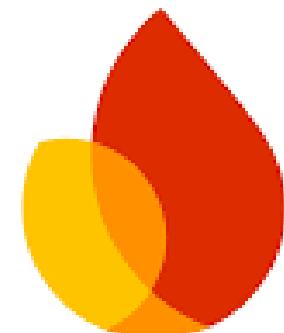


## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Các cách khởi tạo người dùng mới:

- Tạo người dùng trực tiếp từ phần Authentication của Firebase console, tại trang Users.

The screenshot shows the 'Authentication' section of the Firebase console, specifically the 'Users' tab. At the top, there's a navigation bar with tabs: 'Users' (which is selected), 'Sign-in method', 'Templates', 'Usage', 'Settings', and 'Extensions'. Below the navigation bar, a message box contains a warning about authentication features stopping soon due to Firebase Dynamic Links shutdown. The main area has a search bar and buttons for 'Add user' and 'Import'. A table lists user data with columns: Identifier, Providers, Created (with a dropdown arrow), Signed In, and User UID. A modal window titled 'Add an Email/Password user' is open at the bottom, containing fields for 'Email' and 'Password', both with blue outlines indicating they are active. There are 'Cancel' and 'Add user' buttons at the bottom of the modal.



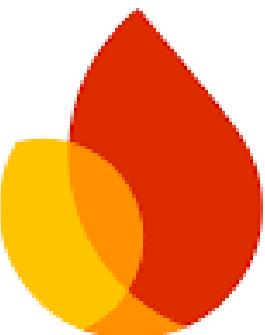
Firebase



## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

**Lấy thông tin hồ sơ người dùng:** Để lấy thông tin hồ sơ người dùng, sử dụng các thuộc tính của đối tượng User. 2 cách lấy thông tin người dùng chính bao gồm:

- Sử dụng thuộc tính `currentUser`.
- Lấy thông tin hồ sơ theo nhà cung cấp (Google, Facebook, ...).

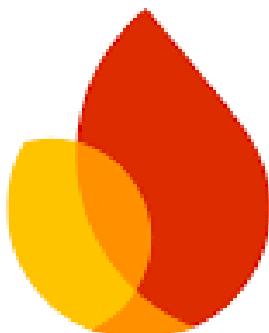




## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

**Thuộc tính currentUser:** Nếu bạn chắc chắn người dùng đang đăng nhập, có thể truy cập User từ thuộc tính currentUser như: uid, email, Metadata (thời gian tạo & đăng nhập gần nhất), ...

```
if (FirebaseAuth.instance.currentUser != null) {  
    print(FirebaseAuth.instance.currentUser?.uid);  
}
```

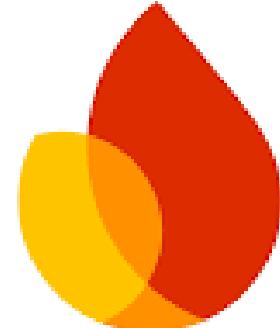




## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

**Lấy thông tin hồ sơ theo nhà cung cấp: Ví dụ khi xác thực thành công  
với Google, Google sẽ trả về 2 mã là IdToken và accessToken:**

```
GoogleAuthProvider.credential(  
    idToken: googleAuth.idToken,  
    accessToken: googleAuth.accessToken,  
)
```





## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Sau đó, Firebase sẽ liên hệ với máy chủ của Google để kiểm tra xem mã idToken có hợp lệ hay không với phương thức:

`FirebaseAuth.instance.signInWithCredential(credential).`

Nếu hợp lệ, Firebase sẽ tự động đồng bộ thông tin vào đối tượng `FirebaseAuth.instance.currentUser`.





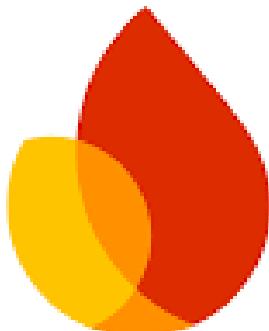
## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Cập nhật hồ sơ người dùng:

```
await user?.updateDisplayName("Jane Q. User");  
await user?.updatePhotoURL("https://example.com/jane-q-user/profile.jpg");
```

Cập nhật email:

```
await user?.updateEmail("janeq@example.com");
```



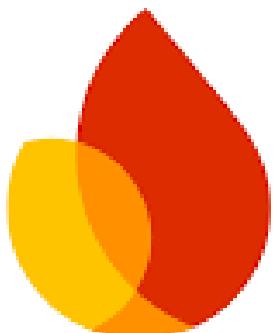


## 5.2. Quản lý dữ liệu người dùng với Firebase Authentication:

Gửi email xác minh:

```
await user?.sendEmailVerification();
```

Ngoài ra, Flutter còn hỗ trợ các phương thức cập nhật thông tin người dùng khác như: cập nhật mật khẩu, gửi email đặt mật khẩu, xóa người dùng, ...

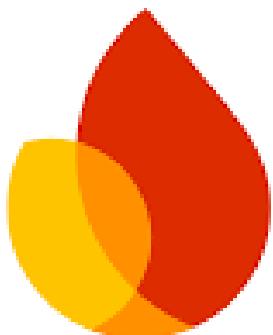




## 5.3. Quản lý dữ liệu người dùng với Firestore Database:

**Firestore Database** trong Flutter là một dịch vụ cơ sở dữ liệu thời gian thực, tích hợp trong Firebase và có thể sử dụng trong ứng dụng Flutter thông qua package `cloud_firestore`.

Firestore lưu trữ dữ liệu dưới dạng document trong các collection, cho phép quản lý dữ liệu có cấu trúc linh hoạt với các kiểu dữ liệu như chuỗi, số, boolean, mảng, bản đồ (map) và thời gian.





## 5.3. Quản lý dữ liệu người dùng với Firestore Database:

Tạo một lớp UserProfile để biểu diễn thông tin người dùng, gồm các trường cơ bản như uid, email, displayName và các thông tin bổ sung.

Lớp này cung cấp các phương thức thao tác với Firestore, bao gồm

- `getUserProfile`
- `updateProfile`
- `getUserProfileStream`





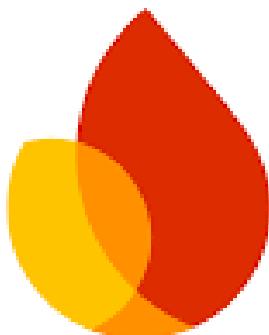
## 5.4. Demo App Flutter:

### Phần Demo App:

The screenshot displays a Flutter application interface. At the top, it says "Trang chủ". In the top right corner, there is a user icon and a "DEBUG" indicator. The main area features a large circular profile picture with a white letter "T" in the center. Below the picture, the name "Trọng Nhân Nguyễn" is displayed in bold, followed by the email address "trongnhan030904@gmail.com" in a smaller box. A large white card-like overlay contains the following information:

- i Thông tin tài khoản**
- User ID: ojZMRLU1ajeNBrF56XB1tQt7ZbW2
- Email đã xác thực: **Đã xác thực**
- Phương thức đăng nhập: Google
- Ngày tạo tài khoản: 8 Tháng 11, 2025

At the bottom of the screen are two buttons: a blue one labeled "Xem Profile Đầy Đủ" and a red one labeled "Đăng xuất".



Firebase



# 5.4. Demo App Flutter:

## Phần Demo App:

The screenshot shows a Flutter application interface. At the top left is a back arrow icon and the text "Profile". At the top right is a "DEBUG" indicator. The main area features a large circular placeholder for a profile picture with a white letter "T" inside. Below it is the name "Trọng Nhân Nguyễn" and an email address "trongnhan030904@gmail.com" with a green verified badge. A sidebar on the left contains sections for "Thông tin cá nhân" (Personal information) with fields for "Giới thiệu" (Introduction) showing "DUT, 22KTMT1", "Số điện thoại" (Phone number) showing "12345678910", "Địa chỉ" (Address) showing "8 Hà Văn Tính", and "Ngày sinh" (Date of birth) showing "10/11/2003".

The screenshot shows a detailed view of account information. It starts with a header "Thông tin tài khoản" (Account information) with an info icon. Below it is a "User ID" section with a fingerprint icon and the value "ojZMRLU1ajeN...". The next section is "Phương thức đăng nhập" (Login method) with a blue arrow icon and the value "Google". Following that is "Trạng thái email" (Email status) with a green checkmark icon and the value "Đã xác thực" (Verified). The next section is "Ngày tạo tài khoản" (Account creation date) with a calendar icon and the value "09/11/2025". The final section is "Đăng nhập lần cuối" (Last login) with a clock icon and the value "10/11/2025 18:36". At the bottom is a red button with a delete icon and the text "Xóa tài khoản" (Delete account).



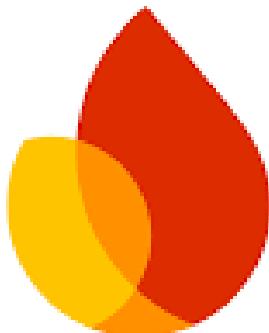
## 5.4. Demo App Flutter:

Phần Demo App:

The screenshot shows a mobile application interface titled "Chỉnh sửa Profile" (Edit Profile) with a "DEBUG" indicator in the top right corner. The screen displays a circular placeholder for a profile picture with a letter "T" and a camera icon. Below it is a note: "Nhấn camera để thay đổi ảnh" (Press camera to change photo) and a warning: "⚠ Lưu ý: Dung lượng ảnh không quá 800KB" (Note: Photo size must not exceed 800KB). The form fields include:

- Họ và tên \*: Text input field containing "Trọng Nhân Nguyễn".
- Số điện thoại: Text input field containing "12345678910".
- Ngày sinh: Date input field showing "10/11/2003" with a calendar icon.
- Địa chỉ: Location input field showing "8 Hà Văn Tính".
- Giới thiệu: Text input field containing "DUT, 22KTMT1" with an info icon and character count "12/200".

A blue "Lưu thay đổi" (Save changes) button is at the bottom, and a "Hủy" (Cancel) button is below it.



Firebase



# 5.4. Demo App Flutter:

## Firebase Authentication

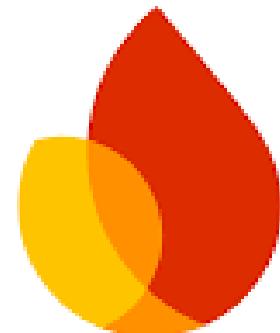
**Authentication**

Users Sign-in method Templates Usage Settings Extensions

**i** The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.

Identifier	Providers	Created ↓	Signed In	User UID
trongnhan03092004@g...		Nov 9, 2025	Nov 9, 2025	e3FsdDqJpkUSTPjok8mX5FD...
demo@gmail.com		Nov 9, 2025	Nov 10, 2025	J9LxuRQMU9V8w4MtMkk6IM...
trongnhan030904@gm...		Nov 8, 2025	Nov 9, 2025	ojZMRLU1ajeNBrF56XB1tQt7Z...
test@gmail.com		Nov 6, 2025	Nov 9, 2025	tqPQdGxOhJgSqTRZ3jQrYUXT...

Rows per page: 50 < 1 - 4 of 4 >



Firebase



# 5.4. Demo App Flutter:

## Firestore Database

Home > users > J9LxuRQMU9V8...

More in Google Cloud ▾

(default)	users	J9LxuRQMU9V8w4MtMkk6IMCaon22
+ Start collection	+ Add document	+ Start collection
users	J9LxuRQMU9V8w4MtMkk6IMCaon22	+ Add field
	e3FsdDqJpkUSTPjok8mX5FDTgYJ3	address: "8 Hà Văn Tính"
	ojZMRLU1ajeNBrF56XB1tQt7zbW2	bio: "Sinh viên DUT"
		createdAt: November 9, 2025 at 6:17:48 PM UTC+7
		dateOfBirth: September 3, 2004 at 12:00:00 AM UTC+7
		displayName: "Nguyễn Trọng Nhân"
		email: "demo@gmail.com"
		lastLoginAt: November 10, 2025 at 3:39:22 PM UTC+7
		loginProvider: "password"
		phoneNumber: "12345678910"
		photoURL: "data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAAD/2v
		uid: "J9LxuRQMU9V8w4MtMkk6IMCaon22"

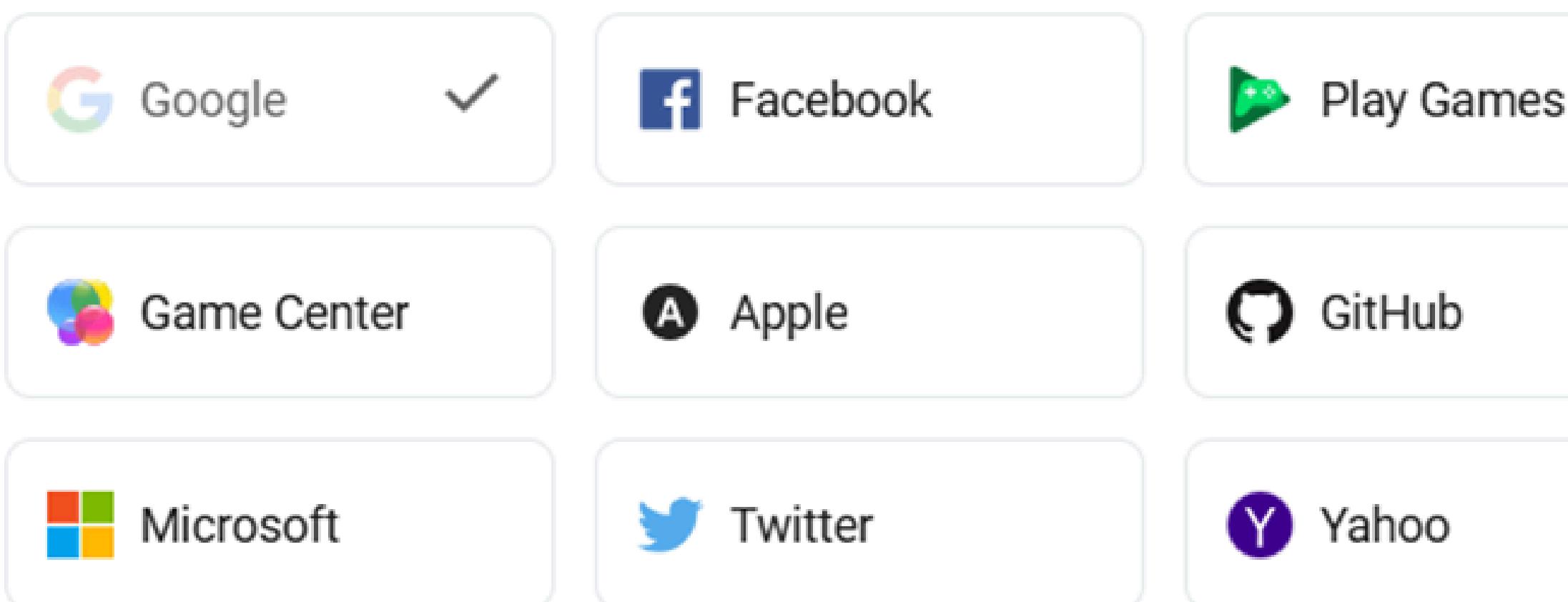


# 6. Security best practices:

## 6.1. Sử dụng các phương thức xác thực an toàn:

- **Ưu tiên OAuth provider thay vì chỉ dùng email/password**

Additional providers



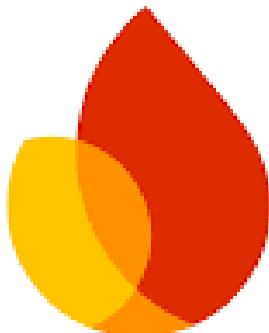


# 6. Security best practices:

## 6.1. Sử dụng các phương thức xác thực an toàn:

- Nếu dùng email/password, luôn bắt buộc mật khẩu mạnh

```
// Hàm kiểm tra mật khẩu mạnh
bool isStrongPassword(String password) {
    // Ít nhất 8 ký tự
    if (password.length < 8) return false;
    // Chữ hoa
    if (!RegExp(r'[A-Z]').hasMatch(password)) return false;
    // Chữ thường
    if (!RegExp(r'[a-z]').hasMatch(password)) return false;
    // Số
    if (!RegExp(r'[0-9]').hasMatch(password)) return false;
    // Ký tự đặc biệt
    if (!RegExp(r'[@#$%^&*(),.?":{}|<>]').hasMatch(password)) return false;
    return true;
}
```





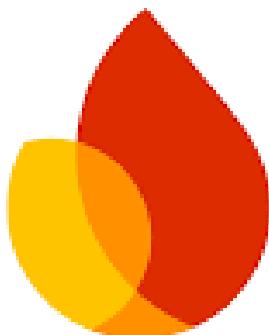
## 6. Security best practices:

**6.2. Xác minh email: Bật Email Verification để đảm bảo người dùng đã xác thực email trước khi truy cập các tính năng quan trọng.**

The screenshot shows a user profile in the Firebase Authentication console. The profile includes the following details:

- Thông tin tài khoản**: User ID: 9XbSWdhtyUXT...
- Phương thức đăng nhập**: Email & Mật khẩu
- Trạng thái email**: Chưa xác thực
- Ngày tạo tài khoản**: 10/11/2025
- Đăng nhập lần cuối**: 10/11/2025 17:08

At the bottom, there are two buttons: **Gửi email xác thực** (Send verification email) and **Xóa tài khoản** (Delete account).





# 6. Security best practices:

## 6.2. Xác minh email: Bật Email Verification để đảm bảo người dùng đã xác thực email trước khi truy cập các tính năng quan trọng.

Verify your email for project-226431785429 Hộp thư đến x

 noreply@lutter-authentication.firebaseio.com 17:09 (1 phút trước) ☆ 😊 ↶ :

đến tôi ▾

Hello,

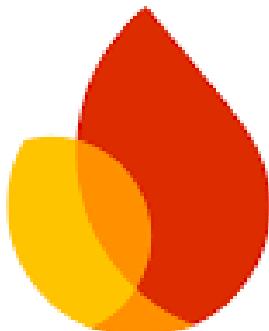
Follow this link to verify your email address.

[https://lutter-authentication.firebaseio.com/\\_auth/action?mode=verifyEmail&oobCode=2I1r7A8h6NvTdid8otV\\_6O0mASSGZVclfV\\_iDXzkxUAAAAGabT5cg&apiKey=AlzaSyBM5nM8IcpOXGvdAQSVIZ9mAtnN4Crsoo&lang=en](https://lutter-authentication.firebaseio.com/_auth/action?mode=verifyEmail&oobCode=2I1r7A8h6NvTdid8otV_6O0mASSGZVclfV_iDXzkxUAAAAGabT5cg&apiKey=AlzaSyBM5nM8IcpOXGvdAQSVIZ9mAtnN4Crsoo&lang=en)

If you didn't ask to verify this address, you can ignore this email.

Thanks,

Your project-226431785429 team





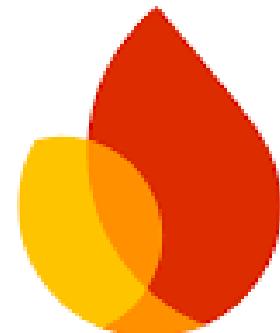
## 6. Security best practices:

**6.2. Xác minh email: Bật Email Verification để đảm bảo người dùng đã xác thực email trước khi truy cập các tính năng quan trọng.**

The screenshot shows a mobile application interface for managing a user account. The top bar has a light blue background with the text "Thông tin tài khoản" (Account Information) in white. Below this, there are five data items, each with an icon and text:

- User ID**: Represented by a purple circular icon with a fingerprint pattern. The text below it is "9XbSWdhtyUXT...".
- Phương thức đăng nhập** (Login Method): Represented by a blue circular icon with a right-pointing arrow. The text below it is "Email & Mật khẩu" (Email & Password).
- Trạng thái email** (Email Status): Represented by a green circular icon with a checkmark inside a shield. The text below it is "Đã xác thực" (Verified).
- Ngày tạo tài khoản** (Account Creation Date): Represented by a teal circular icon with a calendar. The text below it is "10/11/2025".
- Đăng nhập lần cuối** (Last Login): Represented by a light blue circular icon with a clock. The text below it is "10/11/2025 17:14".

At the bottom of the screen is a red rectangular button with a white outline and a white "X" icon, labeled "Xóa tài khoản" (Delete Account).

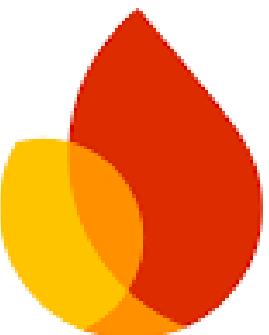




## 6. Security best practices:

**6.3. Bảo vệ chống spam: Điều chỉnh Firebase Authentication tự động giới hạn số lần thử đăng nhập thất bại và có thể tạm khóa trong thời gian ngắn nếu đăng nhập thất bại quá nhiều lần.**

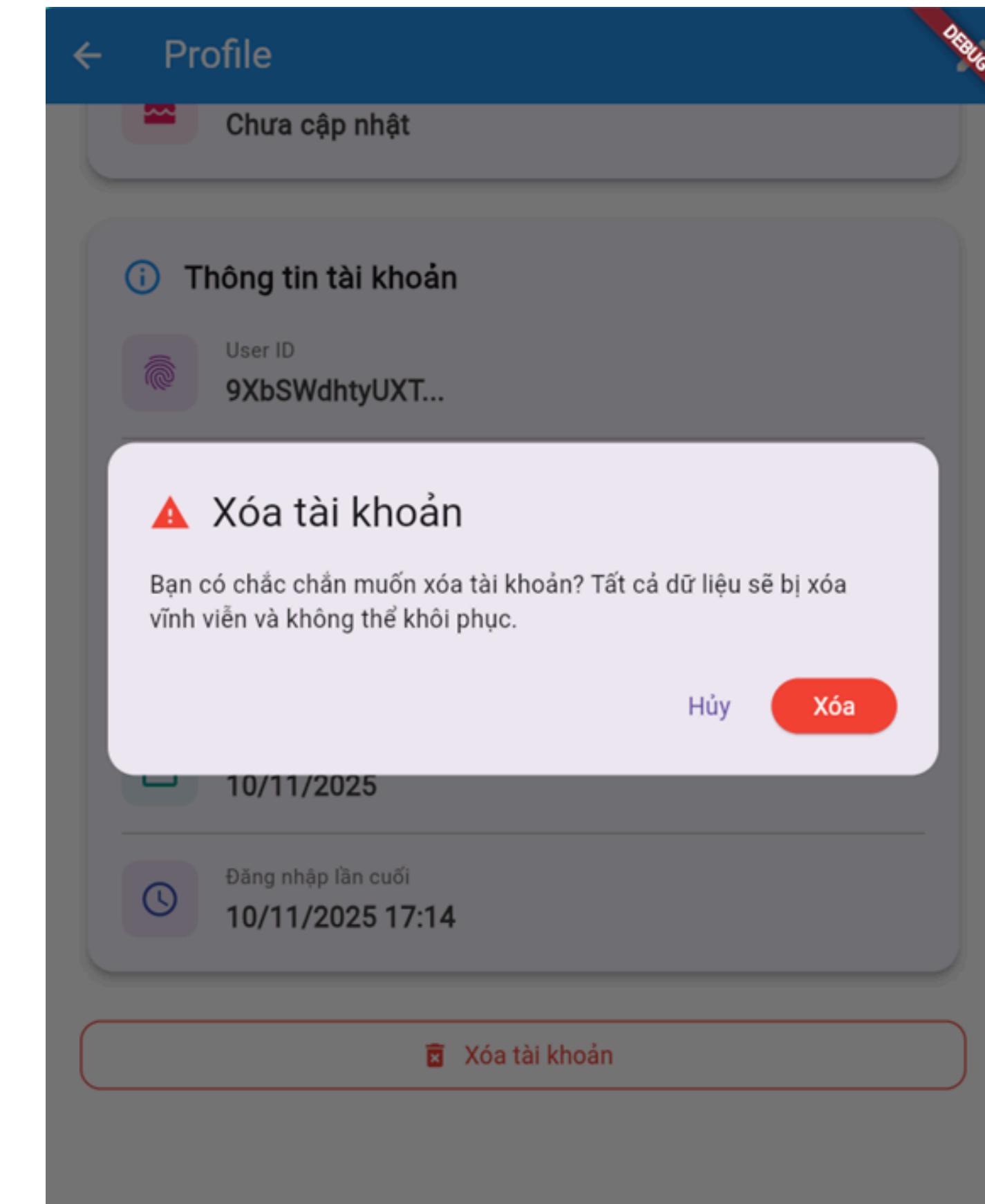
**6.4. Cập nhật SDK thường xuyên: Firebase liên tục cập nhật SDK với các bản vá bảo mật. Đảm bảo luôn dùng phiên bản mới nhất của Firebase Auth để tránh lỗ hổng bảo mật cũ.**





# 6. Security best practices:

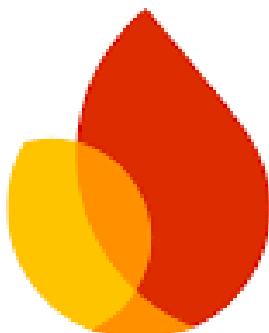
**6.5. Gửi cảnh báo cho người dùng khi thực hiện các chức năng quan trọng như đổi mật khẩu hay xóa tài khoản.**





# TÀI LIỆU THAM KHẢO

- [1] Nguyễn Duy Nhật Viễn, Slide bài giảng môn Lập trình đa nền tảng, Trường Đại học Bách Khoa Đà Nẵng, 2025.
- [2] Firebase, Thêm Firebase vào ứng dụng Flutter, Bắt đầu xác thực Firebase trên Flutter: Firebase Documentation, 2025.
- [3] Flutter Team, Flutter and Firebase resource: Flutter Documentation, 2025.
- [4] Flutter Fire, Social Authentication, Documentation 2025
- [5] Firebase, Cloud Firestore, Firebase Documentation, 2025.



**CẢM ƠN THẦY VÀ CÁC  
BẠN ĐÃ LẮNG NGHE**