



## LẬP TRÌNH ĐA NỀN TẢNG 2

BÀI 7: GIỚI THIỆU VỀ FIREBASE, ĐĂNG  
NHẬP EMAIL/PASSWOR VÀ GOOGLE

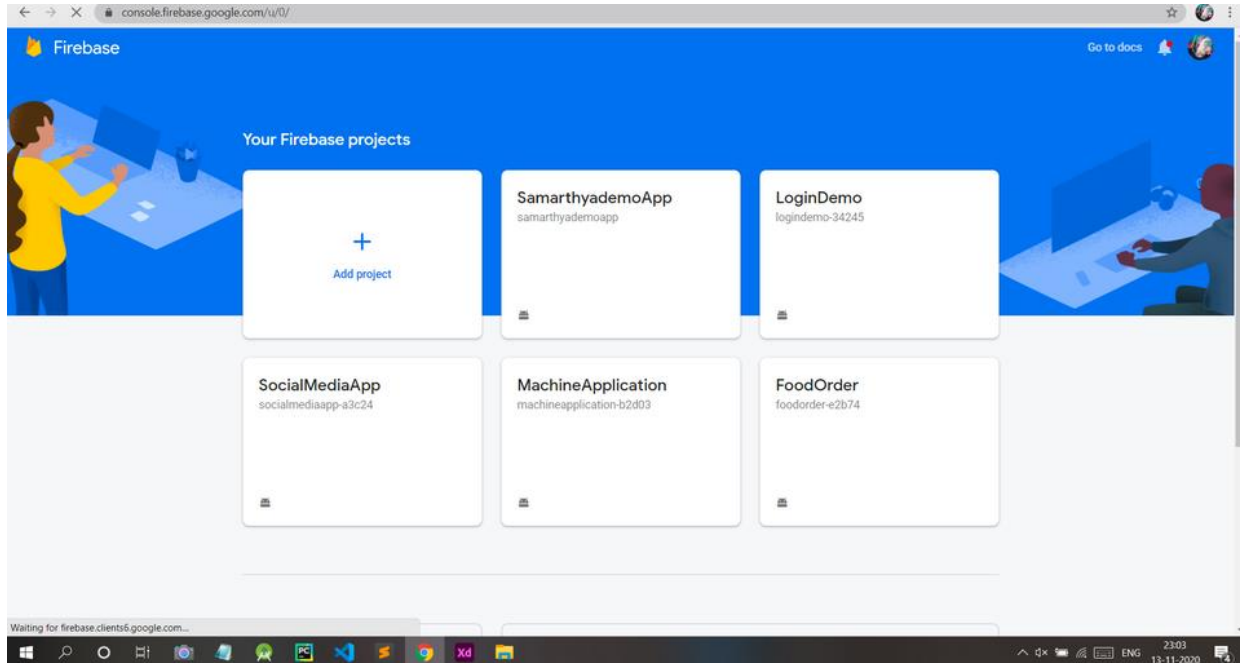
PHẦN 1: GIỚI THIỆU VỀ FIREBASE

- ☐ Giới thiệu về firebase
- ☐ Setup Firebase console

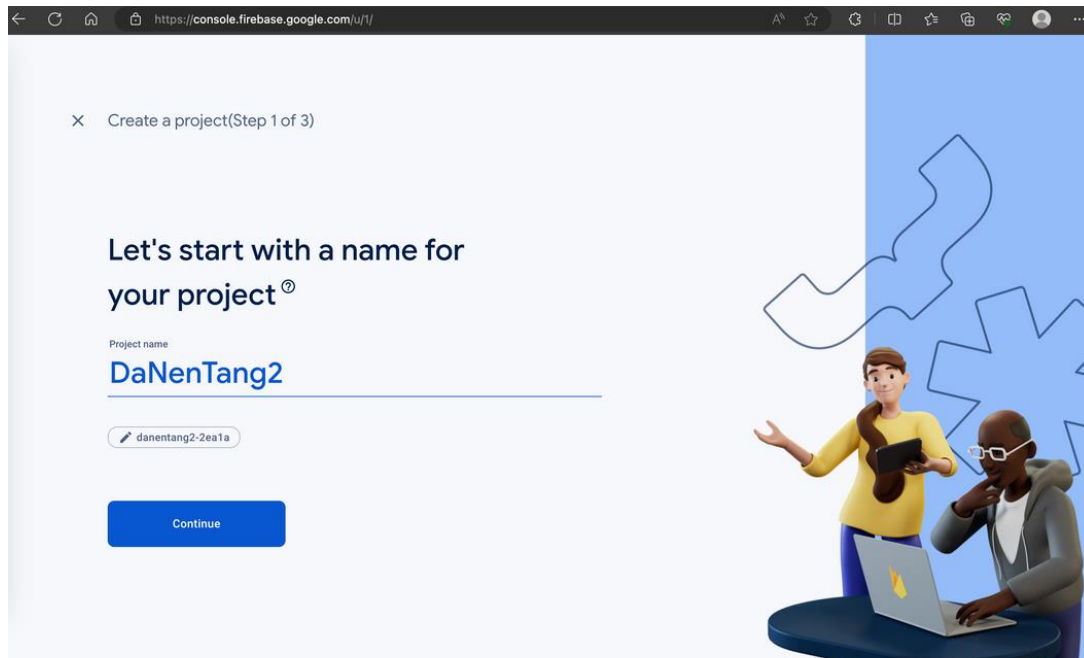
- Firebase là một nền tảng dịch vụ đám mây của Google cung cấp các tính năng cho việc phát triển ứng dụng di động và web. Firebase cung cấp cho các nhà phát triển các công cụ để phát triển và triển khai ứng dụng di động và web, cũng như quản lý người dùng, dữ liệu và thông tin đăng nhập.
- Các tính năng của Firebase bao gồm:
  - ❖ Authentication: Firebase cung cấp tính năng xác thực người dùng, giúp cho việc đăng ký và đăng nhập trở nên dễ dàng hơn. Điều này giúp cho các lập trình viên có thể tập trung vào việc phát triển tính năng thay vì việc quản lý người dùng.

- ❖ Realtime Database: Firebase cung cấp cơ sở dữ liệu thời gian thực, cho phép các ứng dụng có thể truy cập và cập nhật dữ liệu một cách nhanh chóng và liên tục. Điều này giúp cho các ứng dụng có thể đồng bộ hóa dữ liệu trên nhiều thiết bị cùng một lúc.
- ❖ Hosting: Firebase: cung cấp tính năng hosting cho các ứng dụng web, cho phép các lập trình viên có thể đăng ký và triển khai các ứng dụng web của mình trên Firebase một cách dễ dàng.
- ❖ Storage: Firebase cung cấp dịch vụ lưu trữ đám mây, cho phép lưu trữ các tệp tin như hình ảnh, video và âm thanh.

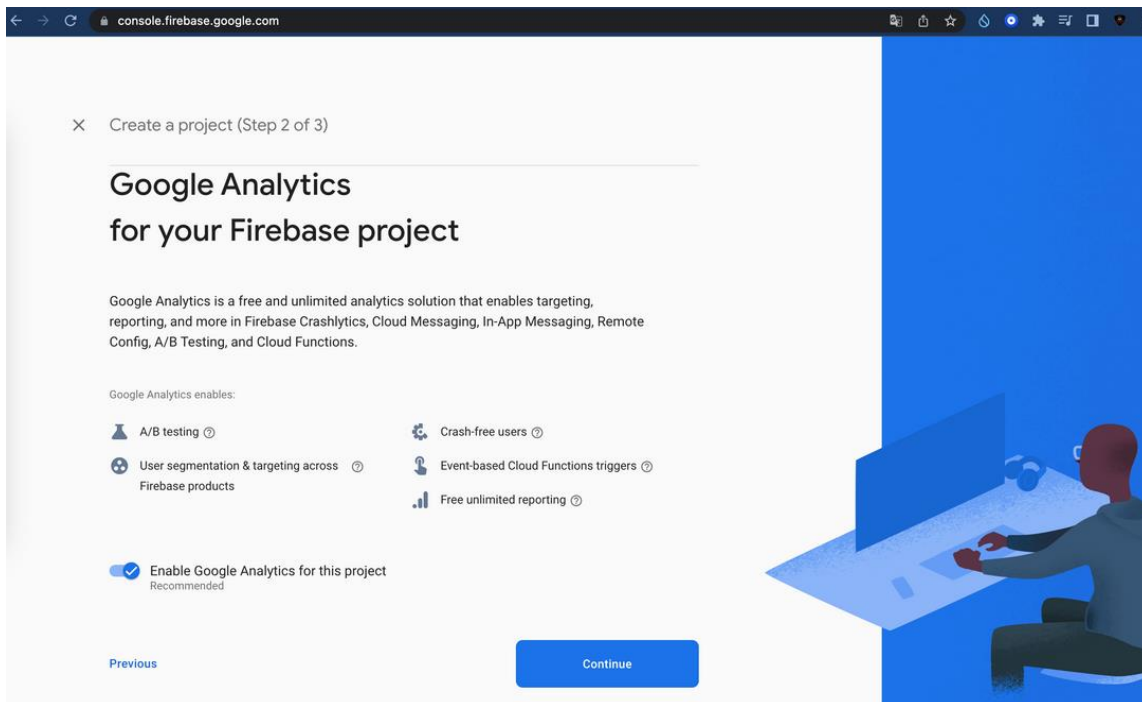
- ❖ Cloud Functions: Firebase cung cấp tính năng Cloud Functions, cho phép các lập trình viên viết các chức năng backend của ứng dụng của mình bằng các ngôn ngữ như JavaScript hoặc TypeScript.



## ❖ Bước 2: Đặt tên cho dự án của bạn và nhấn nút Continue

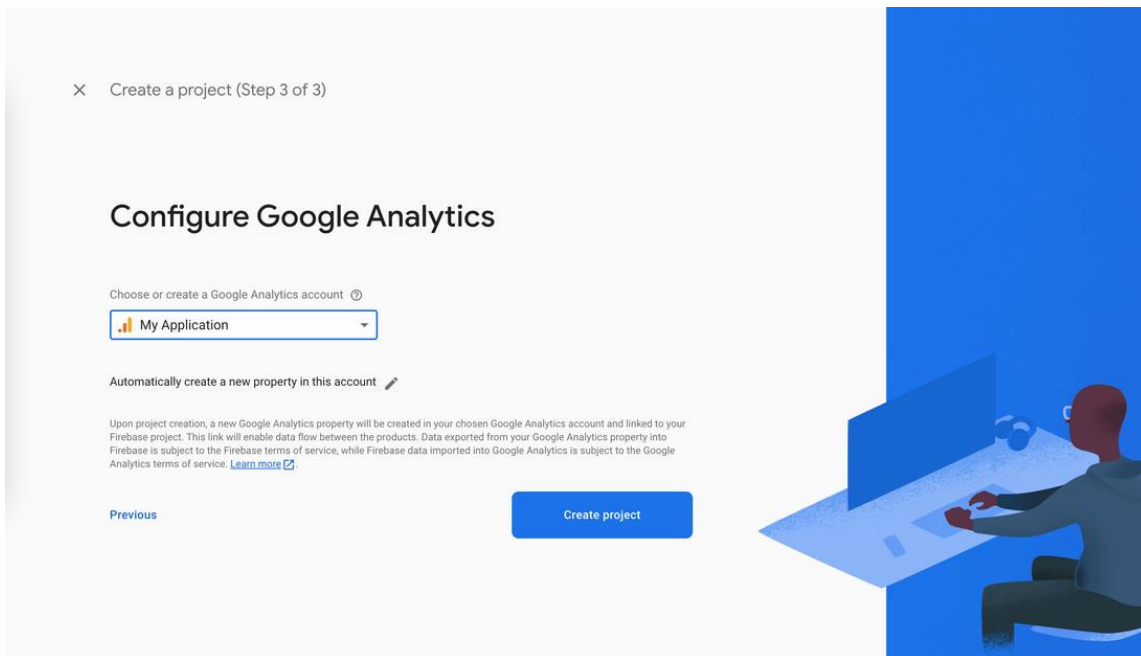


## ❖ Bước 3: Nhấn nút Continue

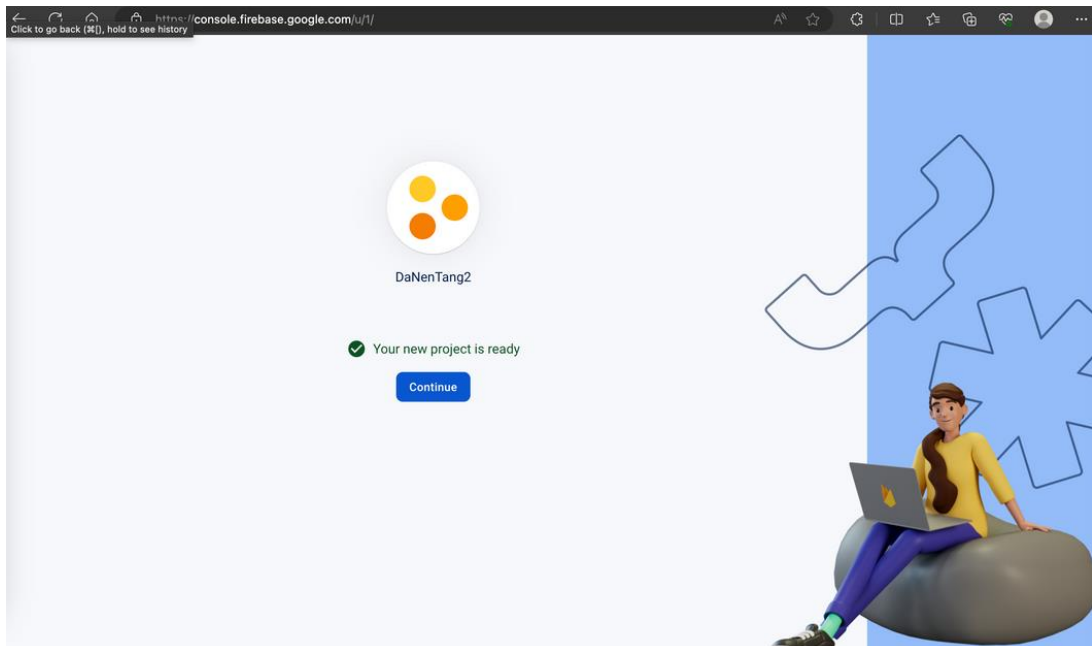




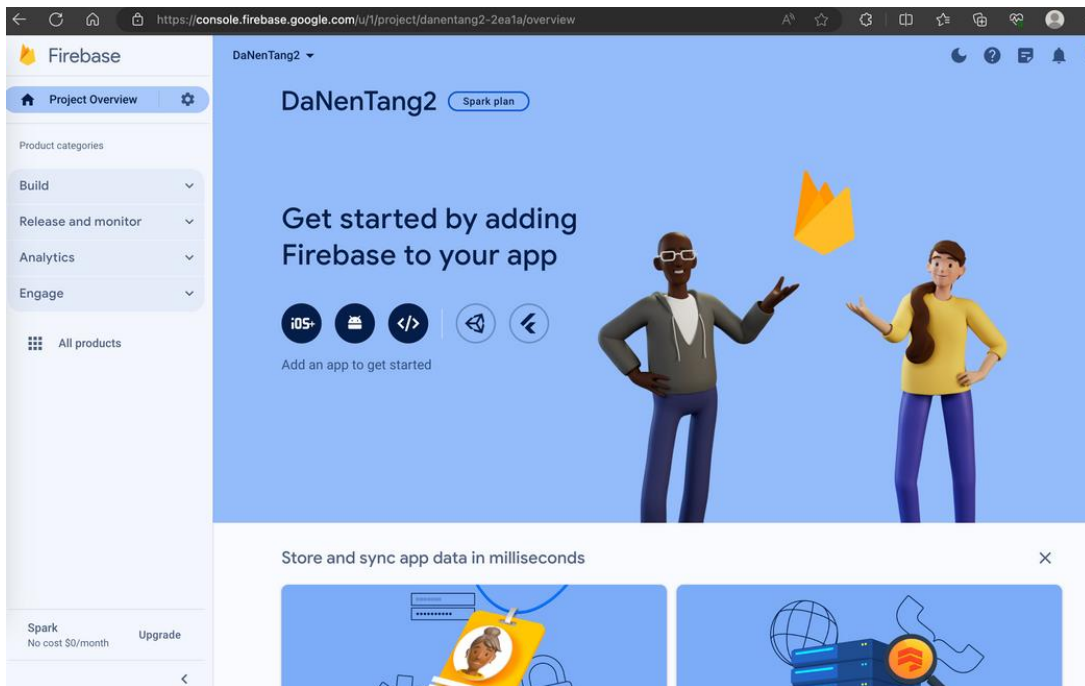
## ❖ Bước 4: Chọn bất kỳ tài khoản nào bạn muốn và nhấn nút Create project



❖ Bước 5: Bây giờ ứng dụng của bạn đã được tạo và tiếp tục làm thêm vài bước nữa nhé.



## ❖ Bước 6: Nhấn icon thứ 2, nút có biểu tượng Android



- ❖ Bước 7: Điền package name của project của bạn, ví dụ com.package\_name. Để lấy được certificate SHA-1 chạy lệnh cd android && ./gradlew signingReport

11

## ❖ Bước 8: Tải file google-services.json và lưu nó trong thư mục android/app.

× Add Firebase to your Android app

✓ Register app  
Android package name: com.danentang2

2 Download and then add config file [Instructions for Android Studio below](#) | [Unity](#) [C++](#)

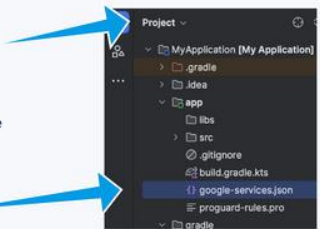
[Download google-services.json](#)

Switch to the **Project** view in Android Studio to see your project root directory.

Move your downloaded `google-services.json` file into your module (app-level) root directory.

`google-services.json`

[Next](#)



### ❖ Bước 9: Thêm firebase SDK .

Để làm cho SDK Firebase có thể truy cập các giá trị config google-services.json, bạn cần có plugin Gradle của các dịch vụ của Google.

Thêm trình cắm làm phụ thuộc vào tệp build.gradle của bạn:

Tệp Gradle nằm ở root (project-level)  
(<project>/build.gradle):

```
plugins {  
    // ...  
  
    id 'com.google.gms.google-services' version '4.4.0' apply false  
}
```

Sau đó, trong tệp build.gradle module (cấp ứng dụng), hãy thêm cả plugin dịch vụ của google và bất kỳ SDK Firebase nào mà bạn muốn sử dụng trong ứng dụng của mình:

Module (app-level) Gradle file(<project><app-module>/build.gradle):

```

plugins {
    id 'com.android.application'
    // Thêm vào Google services Gradle plugin
    id 'com.google.gms.google-services'
    ...
}
dependencies {
    // Import Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:32.6.0')
    // TODO: Thêm phần phụ thuộc cho các sản phẩm Firebase bạn muốn sử
    dụng
    // Khi sử dụng BoM, không chỉ định các phiên bản trong phần phụ thuộc
    Firebase

    implementation 'com.google.firebase:firebase-analytics'
}
    
```



- ❑ Để sử dụng Firebase trong React Native, bạn phải cài đặt package của firebase

```
npm install --save @react-native-firebase/app
```

Package `@react-native-firebase/app` phải được cài đặt trước khi sử dụng bất kỳ dịch vụ Firebase nào khác.

- Kết thúc bài này, các bạn đã được học cách setup một project trên Firebase Console và cài đặt thư viện @react-native-firebase/app. Ở bài sau, các bạn sẽ được học cách Authentication Firebase với React Native app.



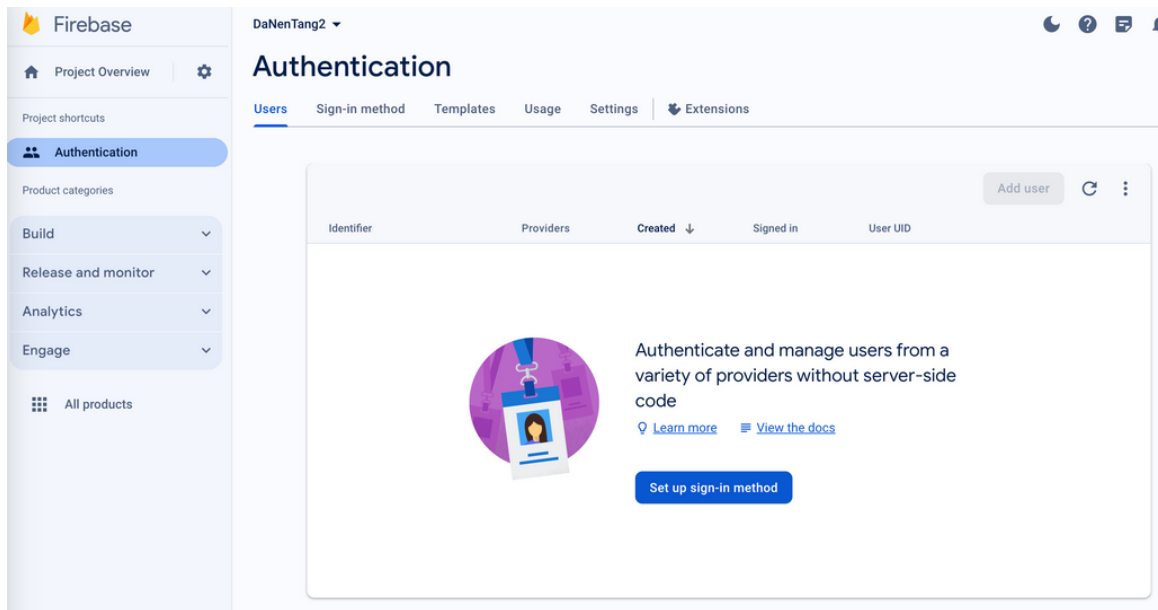
## LẬP TRÌNH ĐA NỀN TẢNG 2

BÀI 7: GIỚI THIỆU VỀ FIREBASE, ĐĂNG  
NHẬP EMAIL/PASSWOR VÀ GOOGLE

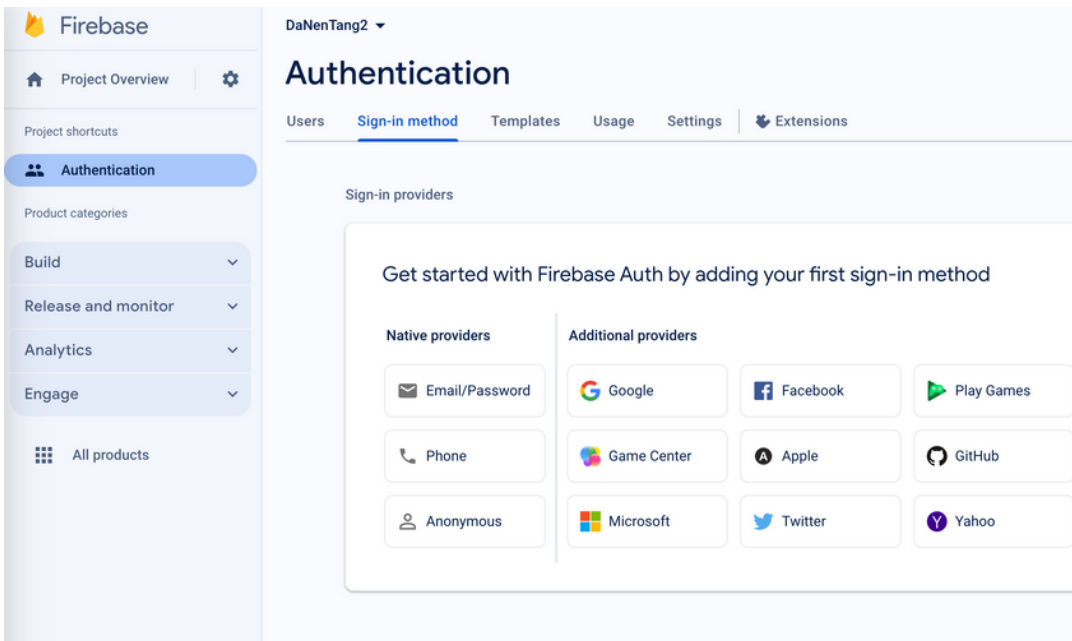
PHẦN 2: ?ĐĂNG NHẬP EMAIL/PASSWOR  
VÀ GOOGLE

- ☐ Ở bài này, chúng ta sẽ học cách xây dựng nút đăng nhập Google với firebase
- ☐ Config project thêm trên console firebase để có thể đăng nhập bằng Google
- ☐ Đăng nhập, đăng ký với tài khoản Email/Password.

## ❖ Bước 1: Vào menu Authentication



## ❖ Bước 2: Tại phần Sign-in-method chọn Google




The screenshot shows the Firebase Authentication console for the project 'DaNenTang2'. The 'Sign-in method' tab is selected, displaying a grid of authentication providers. The 'Native providers' column includes Email/Password, Phone, and Anonymous. The 'Additional providers' column includes Google, Facebook, Play Games, Game Center, Apple, GitHub, Microsoft, Twitter, and Yahoo. A message at the top of the main content area reads: 'Get started with Firebase Auth by adding your first sign-in method'.

- ❖ Bước 3: Nhấn Enable, sau đó chọn mail Project support email, đây là email dùng để hỗ trợ người dùng, cuối cùng nhấn Save.



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).


**Update the [project-level setting](#) below to continue**

Public-facing name for project ⓘ

Support email for project ⓘ

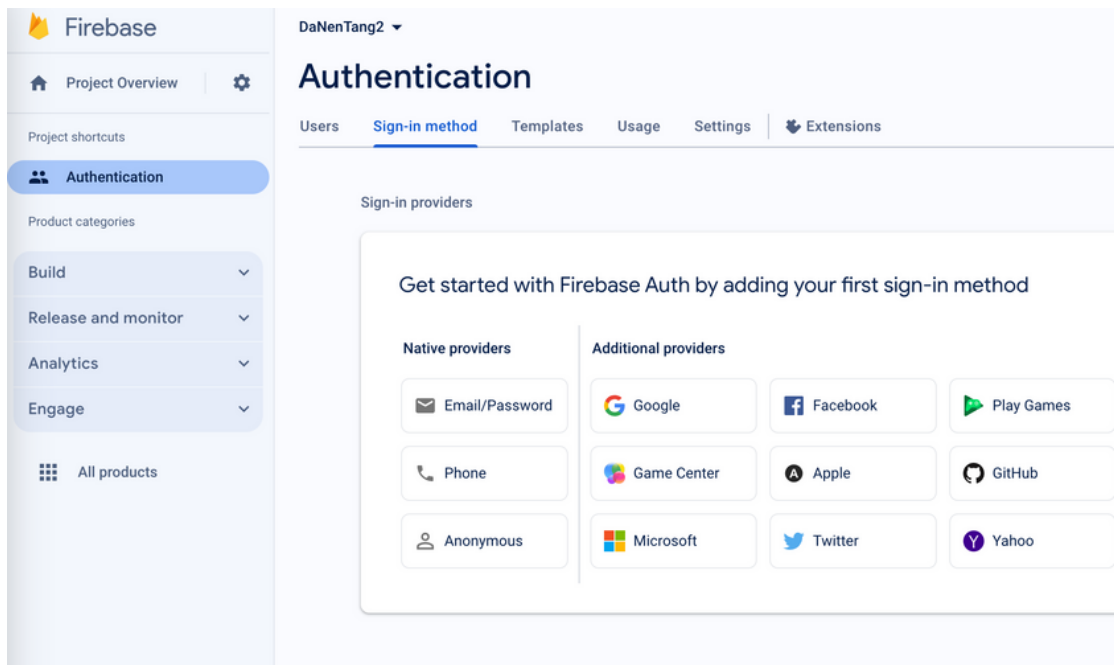
Whitelist client IDs from external projects (optional) ⓘ

Web SDK configuration ⓘ

Web client ID

Web client secret

## ❖ Bước 4: Chọn Email/Password



The screenshot shows the Firebase Authentication console for a project named 'DaNenTang2'. The left sidebar contains the 'Authentication' section, which is highlighted. The main content area shows the 'Sign-in method' tab, which lists 'Email/Password' as the first sign-in method. Below this, there are two columns of additional providers: 'Native providers' (Phone, Anonymous) and 'Additional providers' (Google, Facebook, Play Games, Game Center, Apple, GitHub, Microsoft, Twitter, Yahoo).

**Authentication**

Users **Sign-in method** Templates Usage Settings Extensions

Sign-in providers

Get started with Firebase Auth by adding your first sign-in method

Native providers	Additional providers
Email/Password	Google
Phone	Facebook
Anonymous	Play Games
	Game Center
	Apple
	GitHub
	Microsoft
	Twitter
	Yahoo



## ❖ Bước 5: Enable Email/Password sau đó nhấn nút Save

Provider Status

Configure provider (step 2 of 2)

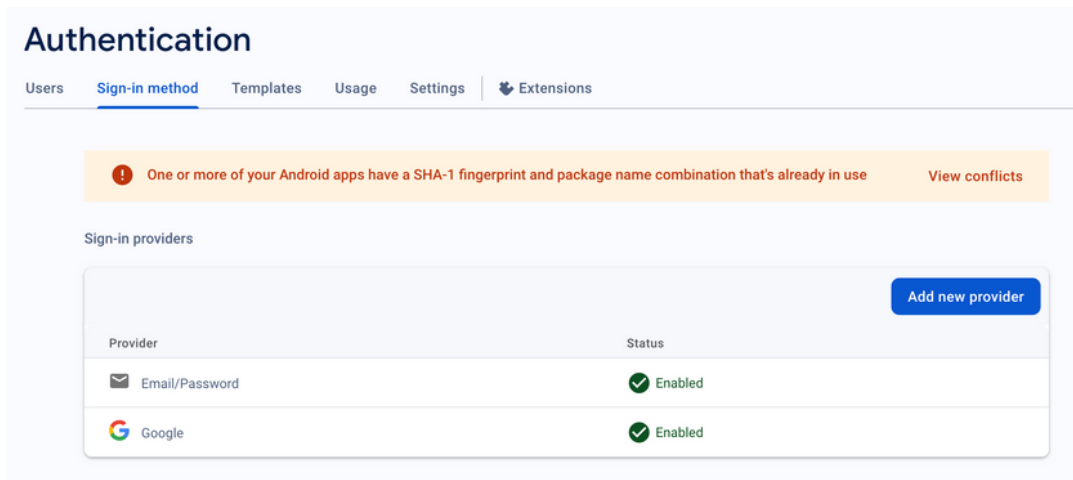
Email/Password ☒ Enable

Allow users to sign up using their email address and password. Our SDKs also provide email address verification, password recovery and email address change primitives. [Learn more](#)

Email link (passwordless sign-in) ☐ Enable

Cancel Save

- ❖ Hoàn tất, ứng dụng của bạn sẽ có giao diện trên Firebase console như sau:



- ☐ Tới bước này thì bạn đã setup mọi thứ thành công, tiếp theo chúng ta sẽ kết nối app với Firebase.
- ☐ Các bạn sẽ thực hiện tạo ứng dụng đăng nhập với Email/Password và đăng nhập bằng google

- ❑ Để có thể đăng nhập bằng tài khoản google bạn cần tải thêm thư viện của firebase theo hướng dẫn dưới đây:

Module này yêu cầu module `@react-native-firebase/app` đã được thiết lập và cài đặt

```
npm i @react-native-firebase/auth
```

## ☐ Giới thiệu về Authentication

Xác thực Firebase cung cấp các dịch vụ backend và SDK để sử dụng để xác thực người dùng với ứng dụng của bạn. Nó hỗ trợ xác thực bằng mật khẩu, số điện thoại, các nhà cung cấp danh tính liên kết phổ biến như Google, Facebook và Twitter, v.v.

Xác thực Firebase tích hợp chặt chẽ với các dịch vụ Firebase khác và nó tuân dụng các tiêu chuẩn ngành như OAuth 2.0 và OpenID Connect, vì vậy nó có thể dễ dàng tích hợp với phụ trợ tùy chỉnh của bạn.

## ☐ Lắng nghe trạng thái xác thực

Trong hầu hết các trường hợp sử dụng Authentication, bạn sẽ muốn biết liệu người dùng của bạn hiện đang đăng nhập hay đăng xuất khỏi ứng dụng của bạn. Mô-đun cung cấp một phương thức được gọi là `onAuthStateChanged` cho phép bạn đăng ký trạng thái xác thực hiện tại của người dùng và nhận một sự kiện bất cứ khi nào trạng thái đó thay đổi.

Điều quan trọng cần nhớ là trình nghe `onAuthStateChanged` không đồng bộ và sẽ kích hoạt trạng thái ban đầu sau khi kết nối với Firebase đã được thiết lập. Do đó, điều quan trọng là phải thiết lập trạng

thái 'initializing' chặn kết xuất ứng dụng chính của chúng tôi trong khi kết nối được thiết lập:

```
DaNenTang2 - GoogleLoginScreen.tsx  
1  const [initializing, setInitializing] = useState(true);  
2  const [user, setUser] = useState();
```

Tạo 2 state, state initializing được dùng để biết trạng thái đang kiểm tra trạng thái đăng nhập

Tiếp theo bạn import auth từ @react-native-firebase/auth. Sử dụng onAuthStateChanged để xác định trạng thái đăng nhập

```
DaNenTang2 - GoogleLoginScreen.tsx

1  function onAuthStateChanged(user) {
2    setUser(user);
3    if (initializing) {
4      setInitializing(false);
5    }
6  }
7
8  useEffect(() => {
9    const subscriber = auth().onAuthStateChanged(onAuthStateChanged);
10   return subscriber; // unsubscribe on unmount
11 }, []);
```



Dựa vào state chúng ta render ra giao diện phù hợp với trạng thái đăng nhập cho người dùng

```
DaNenTang2 - GoogleLoginScreen.tsx

1  if (initializing) {
2    return null;
3  }
4
5  if (!user) {
6    return (
7      <View>
8        <Text>Vui lòng đăng nhập</Text>
9      </View>
10   );
11 }
12 return (
13   <View>
14     <Text>Chào {user.email}</Text>
15   </View>
16 );
```

## Đăng nhập bằng Email/Mật khẩu

---

- ☐ Đăng nhập email/mật khẩu là một phương pháp phổ biến để người dùng đăng nhập trên các ứng dụng. Điều này yêu cầu người dùng cung cấp địa chỉ email và mật khẩu an toàn. Người dùng có thể đăng ký và đăng nhập bằng phương thức có tên `createUserWithEmailAndPassword` hoặc đăng nhập vào tài khoản hiện có bằng `signInWithEmailAndPassword`.
- ☐ Đảm bảo bạn đã bật đăng nhập "Email/Password" được bật trên Bảng điều khiển Firebase.

## Đăng nhập bằng Email/Mật khẩu

---

- ☐ Đăng nhập email/mật khẩu là một phương pháp phổ biến để người dùng đăng nhập trên các ứng dụng. Điều này yêu cầu người dùng cung cấp địa chỉ email và mật khẩu an toàn. Người dùng có thể đăng ký và đăng nhập bằng phương thức có tên `createUserWithEmailAndPassword` hoặc đăng nhập vào tài khoản hiện có bằng `signInWithEmailAndPassword`.
- ☐ Đảm bảo bạn đã bật đăng nhập "Email/Password" được bật trên Bảng điều khiển Firebase.

- Đầu tiên bạn viết hàm tạo tài khoản, bạn gọi `createUserWithEmailAndPassword` của Firebase để tạo tài khoản. Truyền vào 2 prop, prop đầu tiên là email bạn muốn tạo, prop thứ 2 là mật khẩu

```
DaNenTang2 - AuthenticationScreen.tsx

1  const onSignUpWithPassword = () => {
2    auth().createUserWithEmailAndPassword(
3      'jane.doe@example.com',
4      'SuperSecretPassword!',
5    );
6  };
```

- Để bắt được thái tạo tài khoản thành công hay chưa, sử dụng then và catch ngay sau `createUserWithEmailAndPassword`

```

DaNenTang2 - AuthenticationScreen.tsx

1  .then(() => {
2      console.log('Tài khoản đã được tạo và đăng nhập');
3  })
4  .catch(error => {
5      if (error.code === 'auth/email-already-in-use') {
6          console.log('Email đã tồn tại');
7      }
8
9      if (error.code === 'auth/invalid-email') {
10         console.log('Email của bạn không hợp lệ!');
11     }
12
13     console.error(error);
14 });
    
```

- Gọi hàm `signInWithEmailAndPassword` để đăng nhập tài khoản với email và password

```
DaNenTang2 - AuthenticationScreen.tsx

1  const onSignIpWithPassword = () => {
2    auth()
3      .signInWithEmailAndPassword(
4        'jane.doe@example.com',
5        'SuperSecretPassword!',
6      )
7      .then(() => {
8        console.log('Tài khoản đã đăng nhập');
9      })
10     .catch(error => {
11       console.error(error);
12     });
13  };
```

- ☐ Khi bạn đăng nhập thành công `onAuthStateChanged` sẽ cập nhật trạng thái đăng nhập của bạn

- ☐ Thư viện đăng nhập google cung cấp một trình bao bọc xung quanh thư viện đăng nhập chính thức của Google, cho phép bạn tạo thông tin đăng nhập và đăng nhập vào Firebase.
- ☐ Hầu hết các cấu hình đã được thiết lập khi sử dụng Đăng nhập Google bằng Firebase, tuy nhiên bạn cần đảm bảo khóa SHA1 của máy đã được định cấu hình để sử dụng với Android.



## ❑ Cài đặt thư viện

```
npm i @react-native-google-signin/google-signin
```

- ❑ Trước khi kích hoạt yêu cầu đăng nhập, bạn phải khởi tạo Google SDK bằng cách sử dụng bất kỳ required scopes và webClientId có thể tìm thấy trong tệp android/app/google-services.json dưới dạng thuộc tính client/oauth\_client/client\_id (id kết thúc bằng .apps.googleusercontent.com). Đảm bảo chọn client\_id có client\_type: 3

- Gọi hàm GoogleSignin từ package và config webClientId từ file google-service.json của bạn

```
DaNenTang2 - AuthenticationScreen.tsx
1  import {GoogleSignin} from '@react-native-google-signin/google-signin';
2
3  GoogleSignin.configure({
4    webClientId:
5      '182591633460-p38tuh1aeakf26gu9g1lnukt0adfasdf.apps.googleusercontent.com',
6  });
```

- Tiếp theo gọi hàm hasPlayServices để kiểm tra thiết bị có hỗ trợ Google Play không, sau đó gọi hàm signIn để lấy idToken. Cuối cùng gọi hàm signInWithCredential để đăng nhập lên Firebase

```

DaNenTang2 - AuthenticationScreen.tsx

1  async function onGoogleButtonPress() {
2    // Check if your device supports Google Play
3    await GoogleSignin.hasPlayServices({showPlayServicesUpdateDialog: true});
4
5    // Get the users ID token
6    const {idToken} = await GoogleSignin.signIn().catch(error =>
7      console.log('ERROR ', error),
8    );
9
10   // Create a Google credential with the token
11   const googleCredential = auth.GoogleAuthProvider.credential(idToken);
12
13   // Sign-in the user with the credential
14   return auth().signInWithCredential(googleCredential);
15 }

```

- Để đăng xuất khỏi tài khoản google, bạn gọi `GoogleSignin.signOut()`

- ☐ Ở chương này, các bạn đã được học cách đăng nhập, đăng ký tài khoản với Email/Password và đăng nhập với tài khoản google.

- ☐ Giới thiệu về firebase
- ☐ Setup Firebase console
- ☐ Ở bài này, chúng ta sẽ học cách xây dựng nút đăng nhập Google với firebase
- ☐ Config project thêm trên console firebase để có thể đăng nhập bằng Google
- ☐ Đăng nhập, đăng ký với tài khoản Email/Password.

# Kết thúc