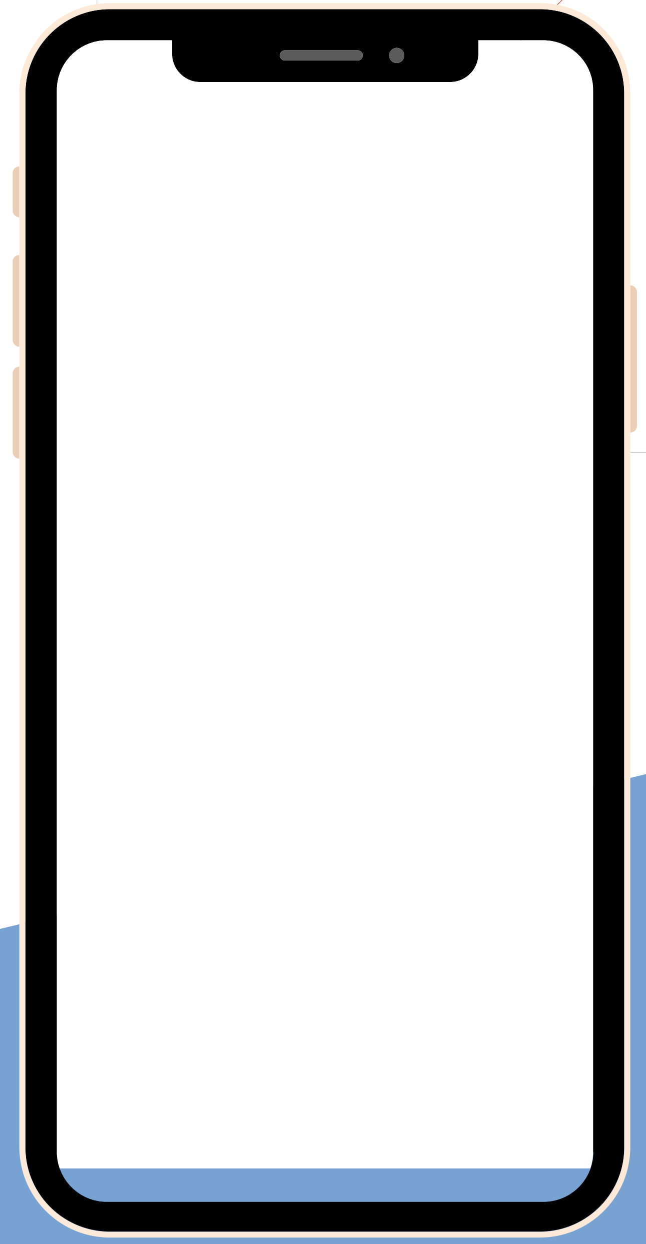
Facebook Login in Flutter App

Requirements:-

- Flutter should be installed on your system (latest version)
- Android Studio should be installed on your system.
- Must have a Facebook Account
- Must have a Facebook developer account.
- Must have a firebase account.
- Emulator or Android/iOS device for testing.



Before moving further it is assumed that all the requirements are fulfilled. So, without wasting any time lets move to our first step.

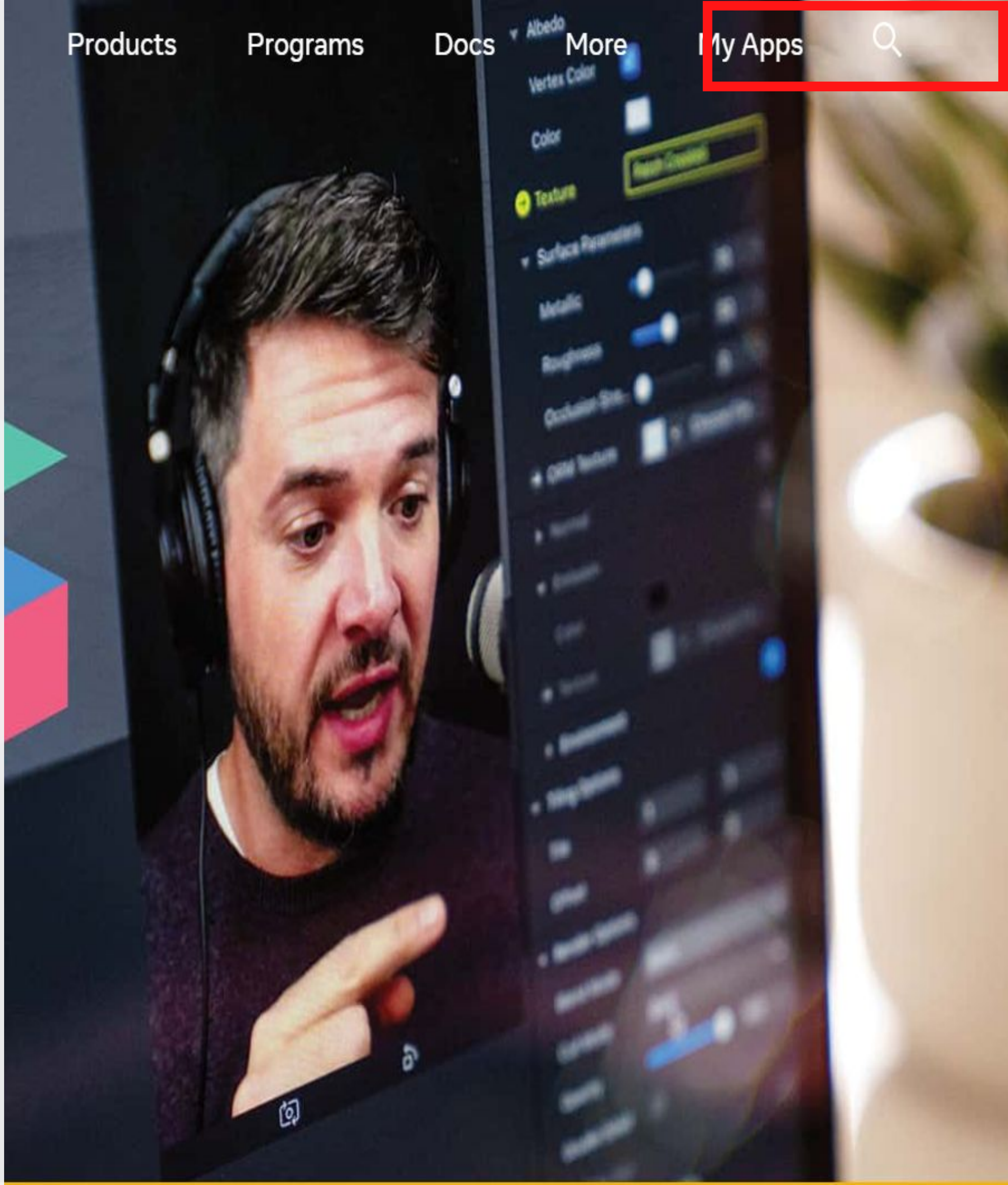
Step - 1

It is assumed that you have downloaded and extracted appcode on your system.

Open facebook developer dashboard.
you can also follow this link
- Facebook for Developers

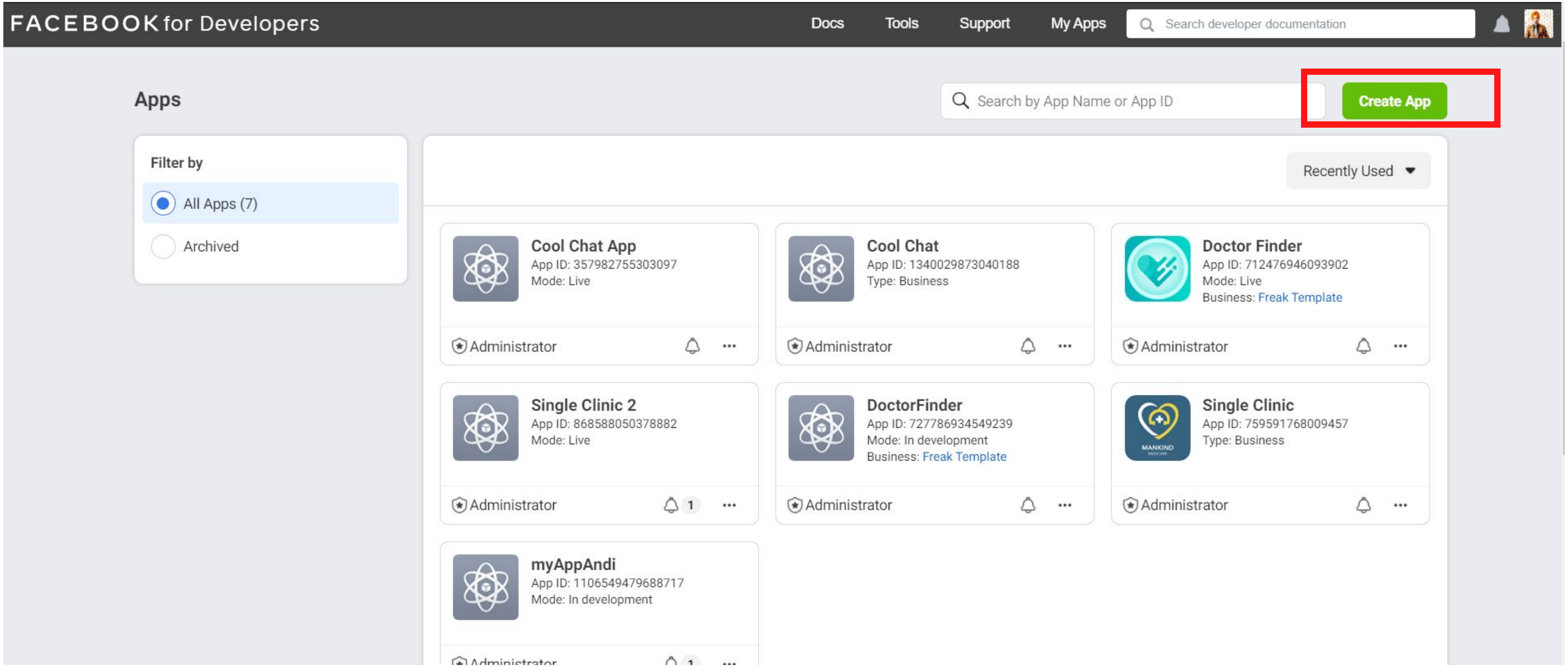
In order to proceed further you must have a facebook account, if you don't have one create a facebook account first

After that click on My Apps button in top right corner on facebook developer dashboard.



Step - 2

Click on create App.



Cool Chat App
App ID: 357982755303097
Mode: Live
Administrator

Cool Chat
App ID: 1340029873040188
Type: Business
Administrator

Doctor Finder
App ID: 712476946093902
Mode: Live
Business: Freak Template
Administrator

Single Clinic 2
App ID: 868588050378882
Mode: Live
Administrator 1

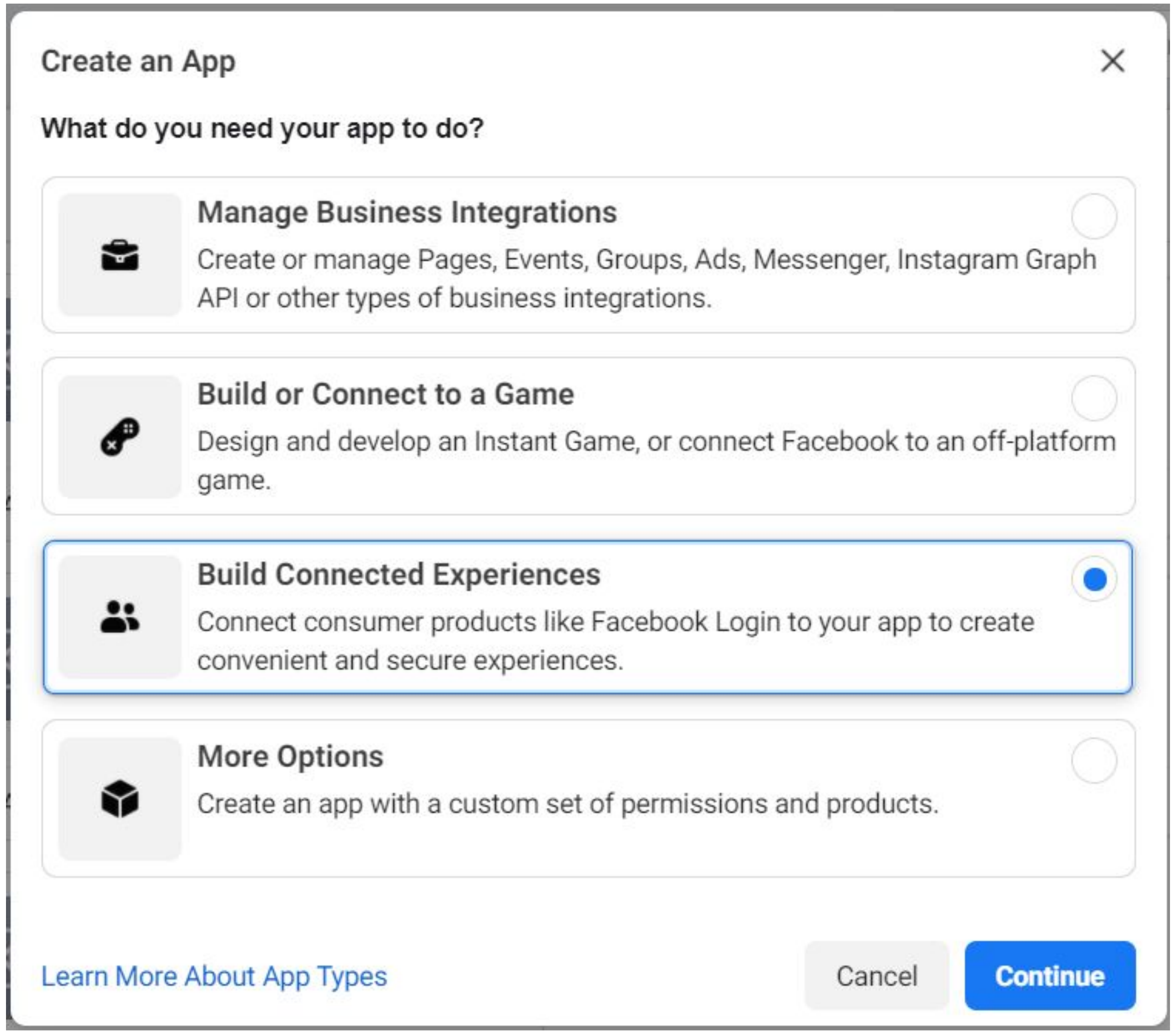
DoctorFinder
App ID: 727786934549239
Mode: In development
Business: Freak Template
Administrator

Single Clinic
App ID: 759591768009457
Type: Business
Administrator

myAppAndi
App ID: 1106549479688717
Mode: In development
Administrator 1

Step - 3

Select the type of for which you want to implement facebook login.



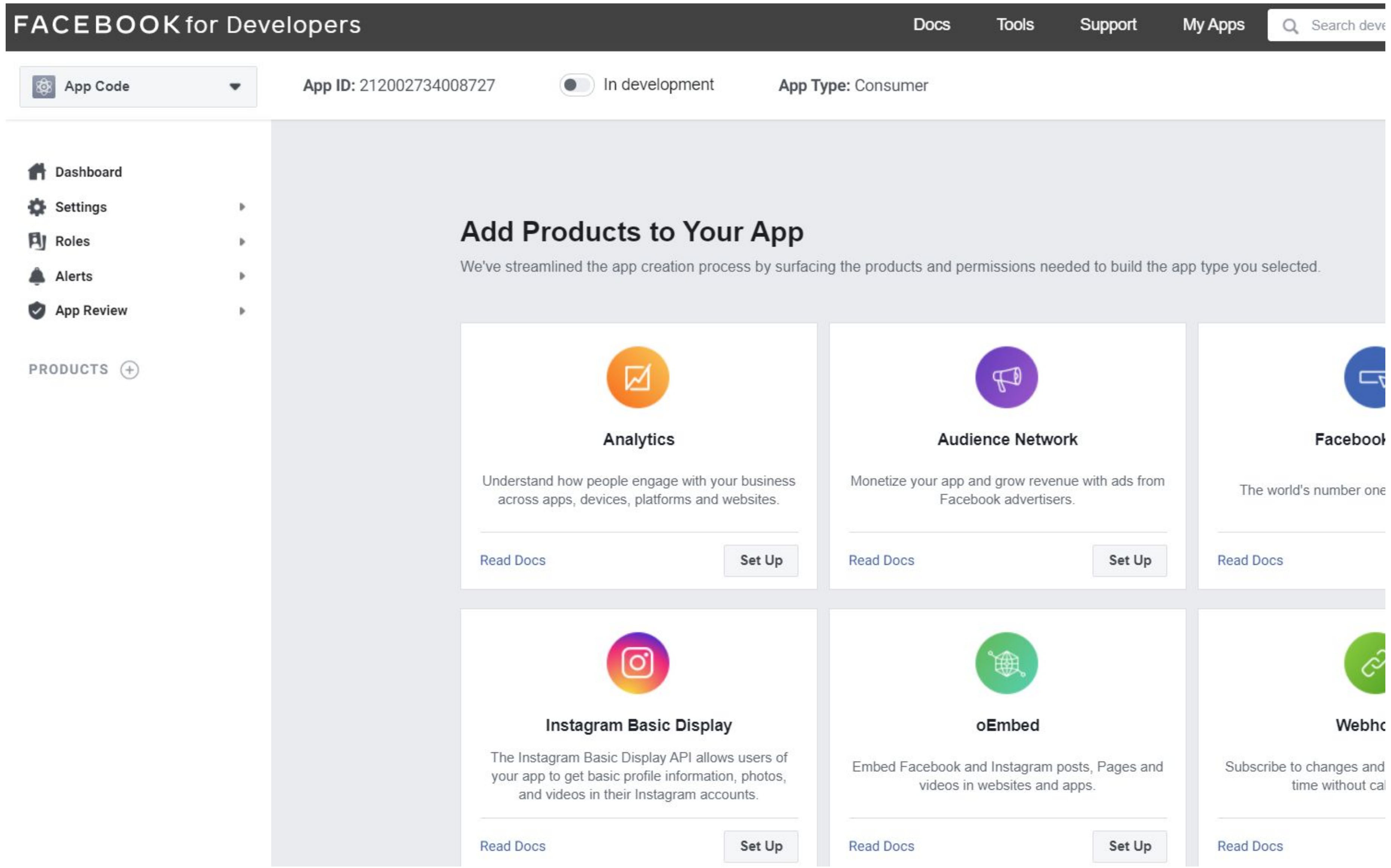
What do you need your app to do?

○ **Manage Business Integrations**
Create or manage Pages, Events, Groups, Ads, Messenger, Instagram Graph API or other types of business integrations.

○ **Build or Connect to a Game**
Design and develop an Instant Game, or connect Facebook to an off-platform game.

● **Build Connected Experiences**
Connect consumer products like Facebook Login to your app to create convenient and secure experiences.

○ **More Options**
Create an app with a custom set of permissions and products.



App ID: 212002734008727 In development **App Type:** Consumer

Add Products to Your App

We've streamlined the app creation process by surfacing the products and permissions needed to build the app type you selected.

Analytics
Understand how people engage with your business across apps, devices, platforms and websites.

Audience Network
Monetize your app and grow revenue with ads from Facebook advertisers.

Faceboo
The world's number one

Instagram Basic Display
The Instagram Basic Display API allows users of your app to get basic profile information, photos, and videos in their Instagram accounts.

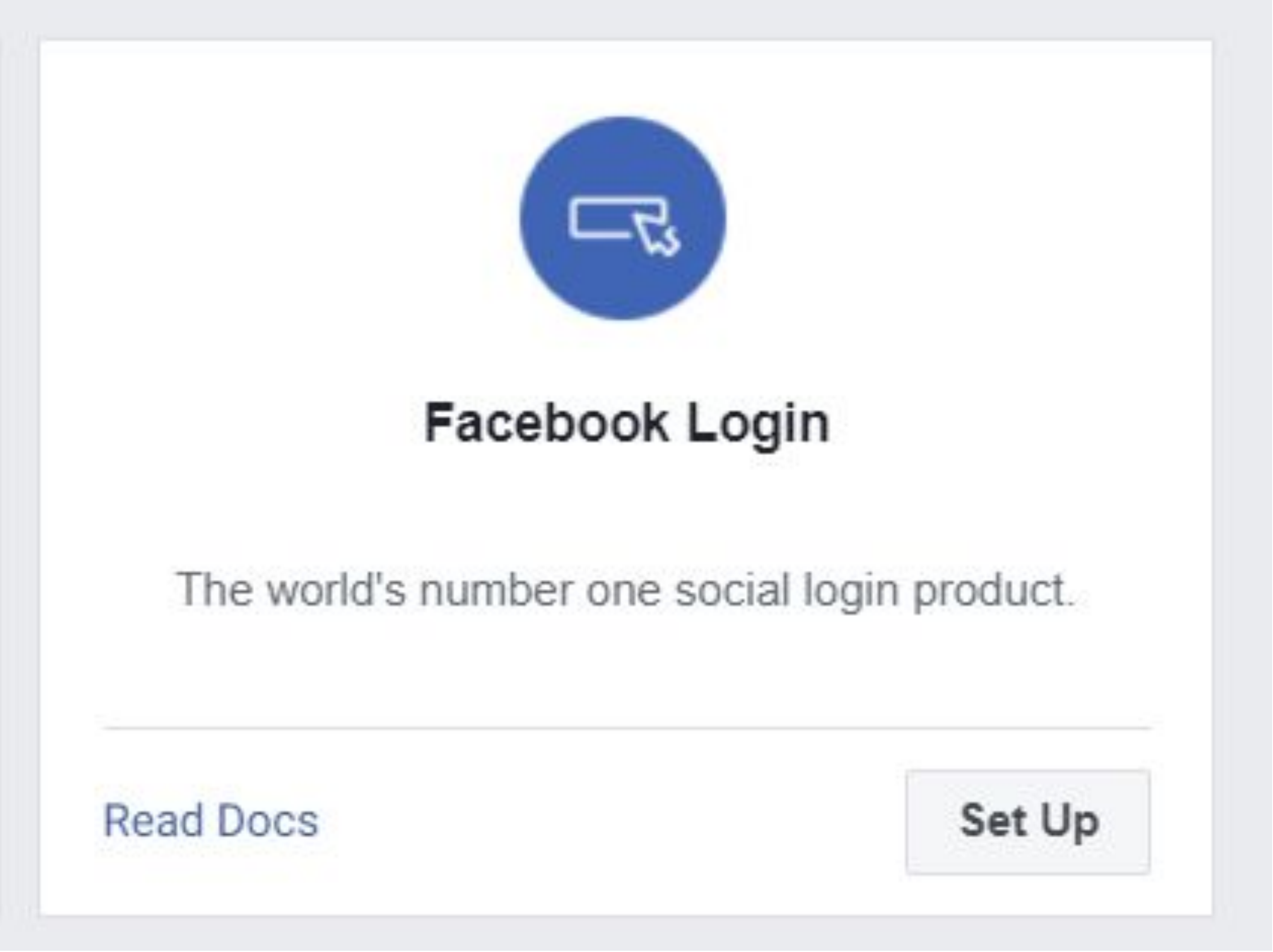
oEmbed
Embed Facebook and Instagram posts, Pages and videos in websites and apps.

Webh
Subscribe to changes and time without ca

Facebook dashboard will open in front of you displaying plenty of services that facebook offers.

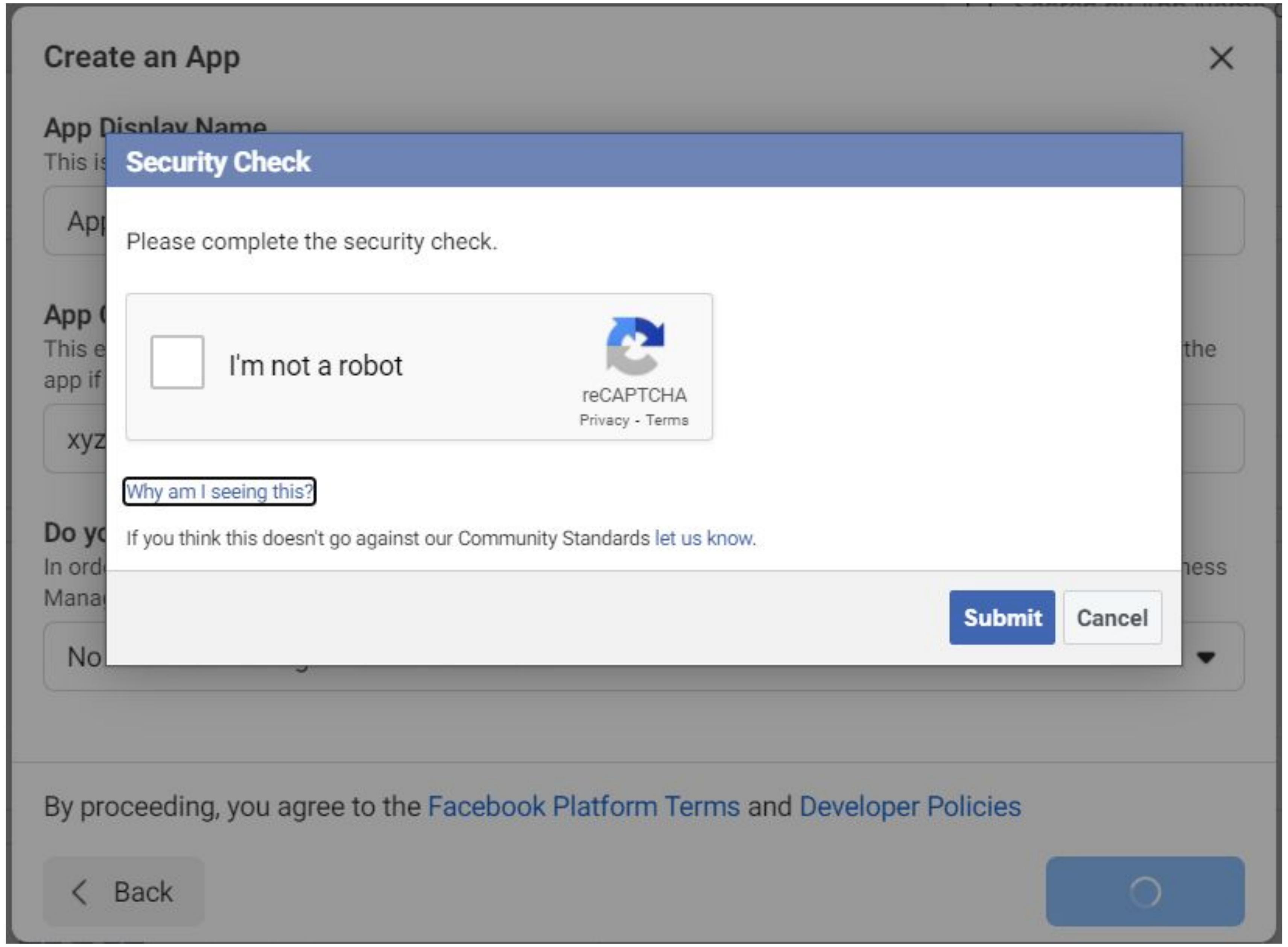
Step - 4

Our interested area is Facebook login, we will select Facebook login and continue. Click on "Set Up"



Facebook Login

The world's number one social login product.



Create an App

Security Check

Please complete the security check.

□ I'm not a robot

Why am I seeing this?

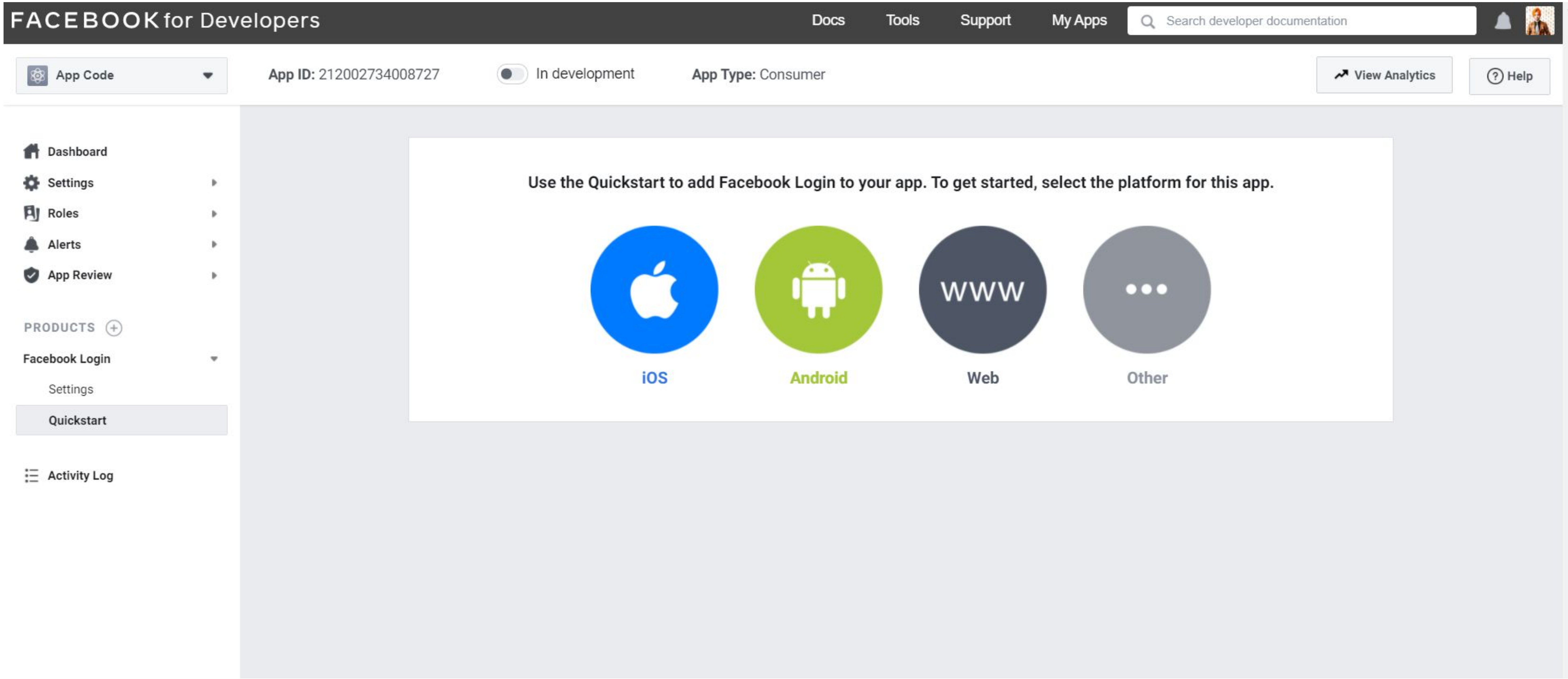
If you think this doesn't go against our Community Standards let us know.

By proceeding, you agree to the Facebook Platform Terms and Developer Policies

A verification dialog will appear in front of you. Mark it as checked and click on submit

Step - 5

Select type of app for which you want to setup facebook login.



App ID: 212002734008727

In development

App Type: Consumer

Use the Quickstart to add Facebook Login to your app. To get started, select the platform for this app.

Step - 6

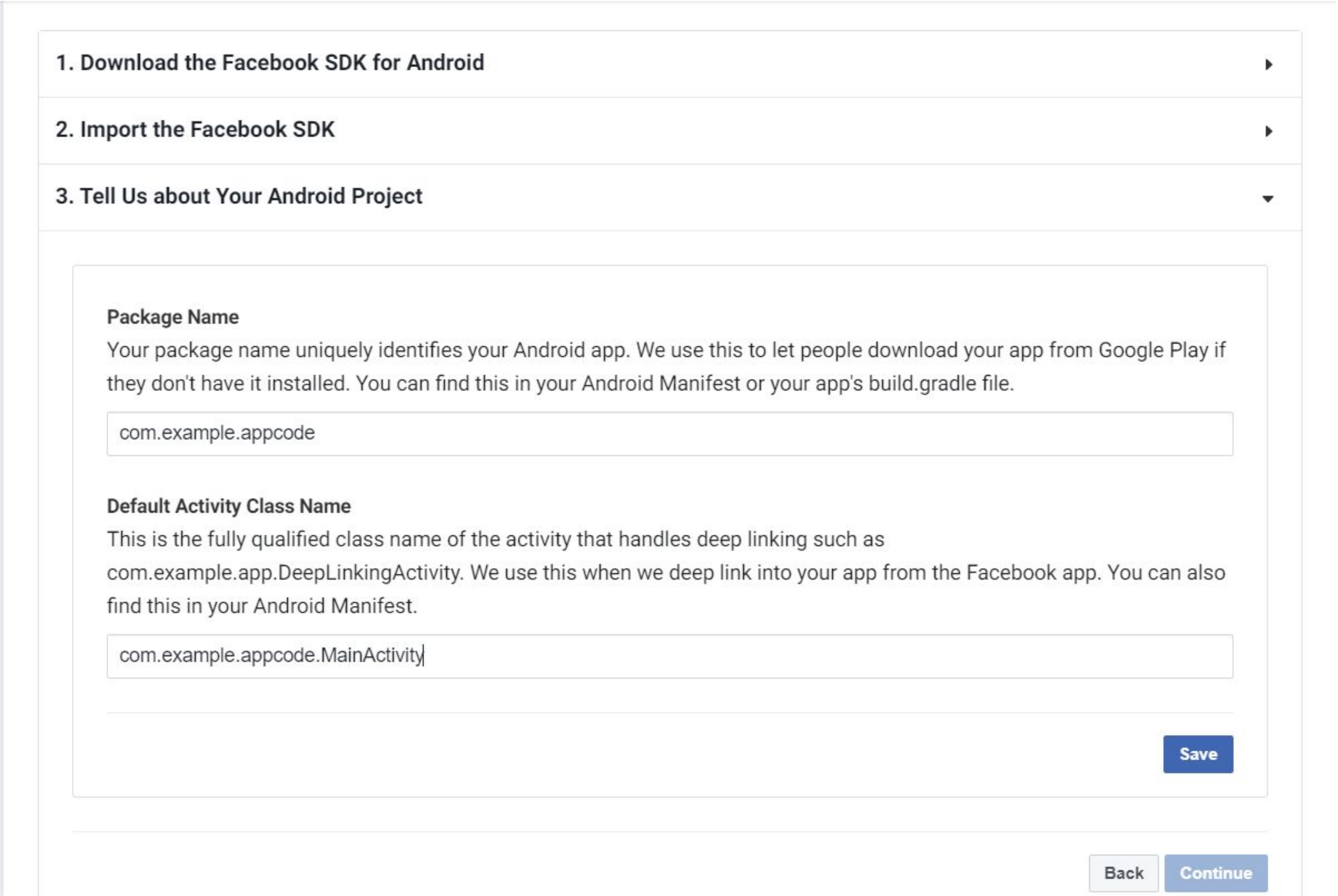
Steps for integration will appear in from of you, skip first two steps

In third step place package name of your application, as in our case it is:-

com.example.appcode

Default activity name will be pacakage name followed by .MainActivity:-

com.example.appcode.MainActivity



1. Download the Facebook SDK for Android

2. Import the Facebook SDK

3. Tell Us about Your Android Project

Package Name

Your package name uniquely identifies your Android app. We use this to let people download your app from Google Play if they don't have it installed. You can find this in your Android Manifest or your app's build.gradle file.

com.example.appcode

Default Activity Class Name

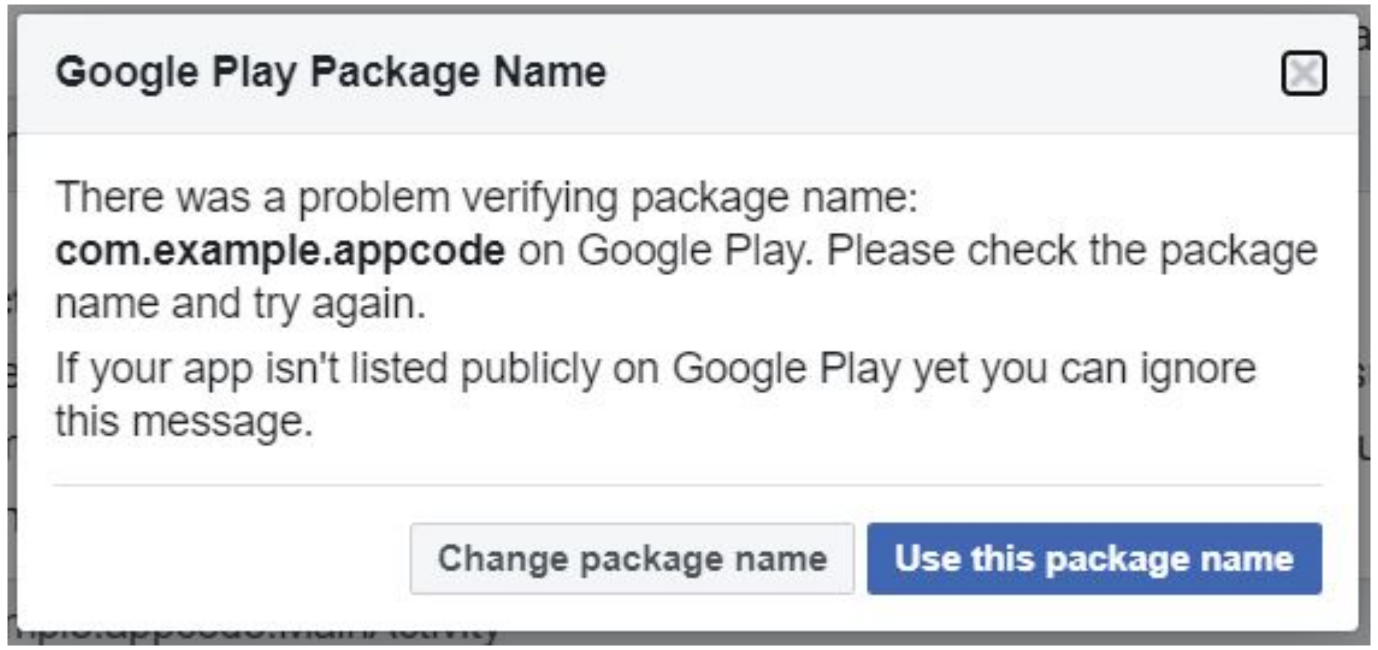
This is the fully qualified class name of the activity that handles deep linking such as com.example.app.DeepLinkingActivity. We use this when we deep link into your app from the Facebook app. You can also find this in your Android Manifest.

com.example.appcode.MainActivity

Step - 7

Click on save button an alert will appear in front of you, click on

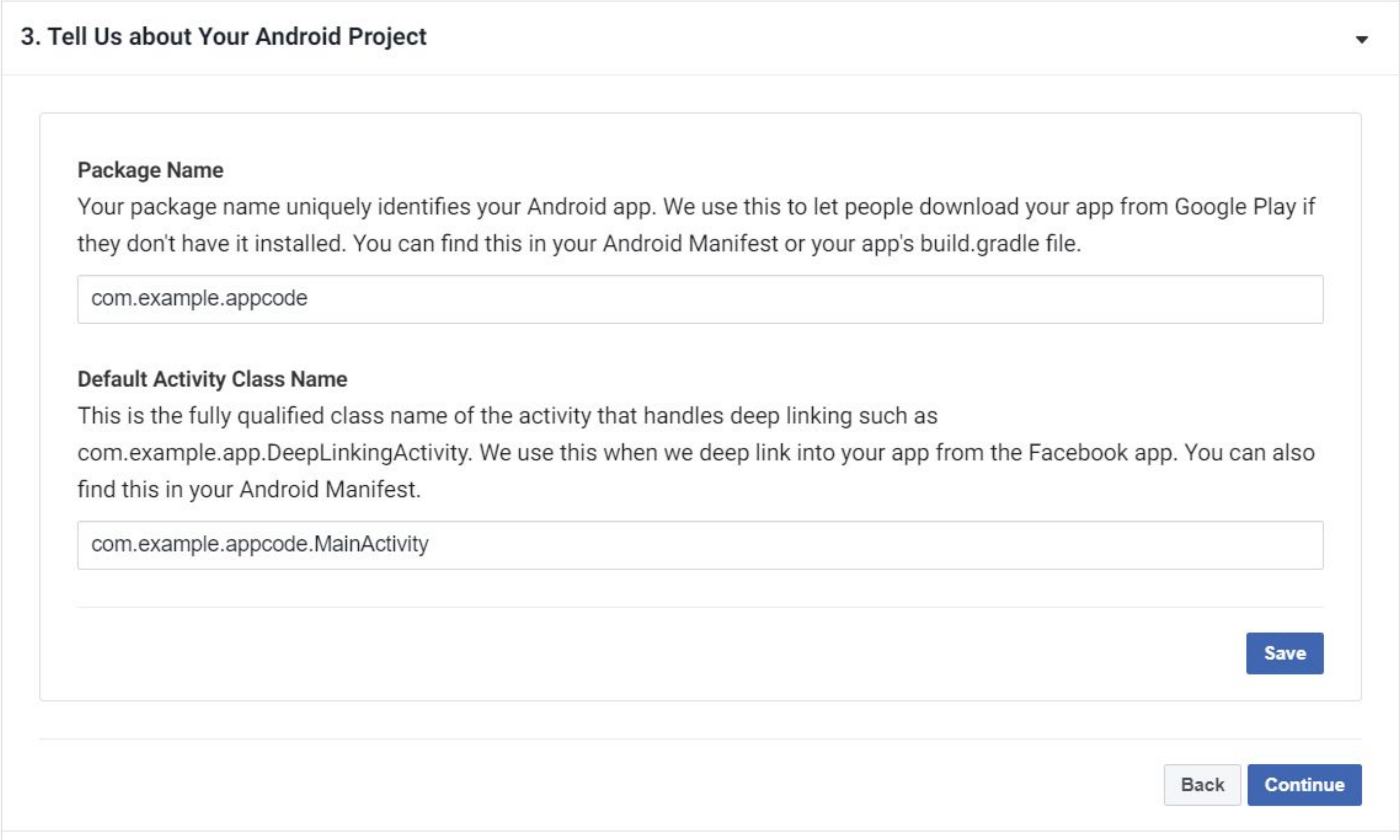
"Use this package name"



Google Play Package Name

There was a problem verifying package name: **com.example.appcode** on Google Play. Please check the package name and try again.

If your app isn't listed publicly on Google Play yet you can ignore this message.



3. Tell Us about Your Android Project

Package Name

Your package name uniquely identifies your Android app. We use this to let people download your app from Google Play if they don't have it installed. You can find this in your Android Manifest or your app's build.gradle file.

com.example.appcode

Default Activity Class Name

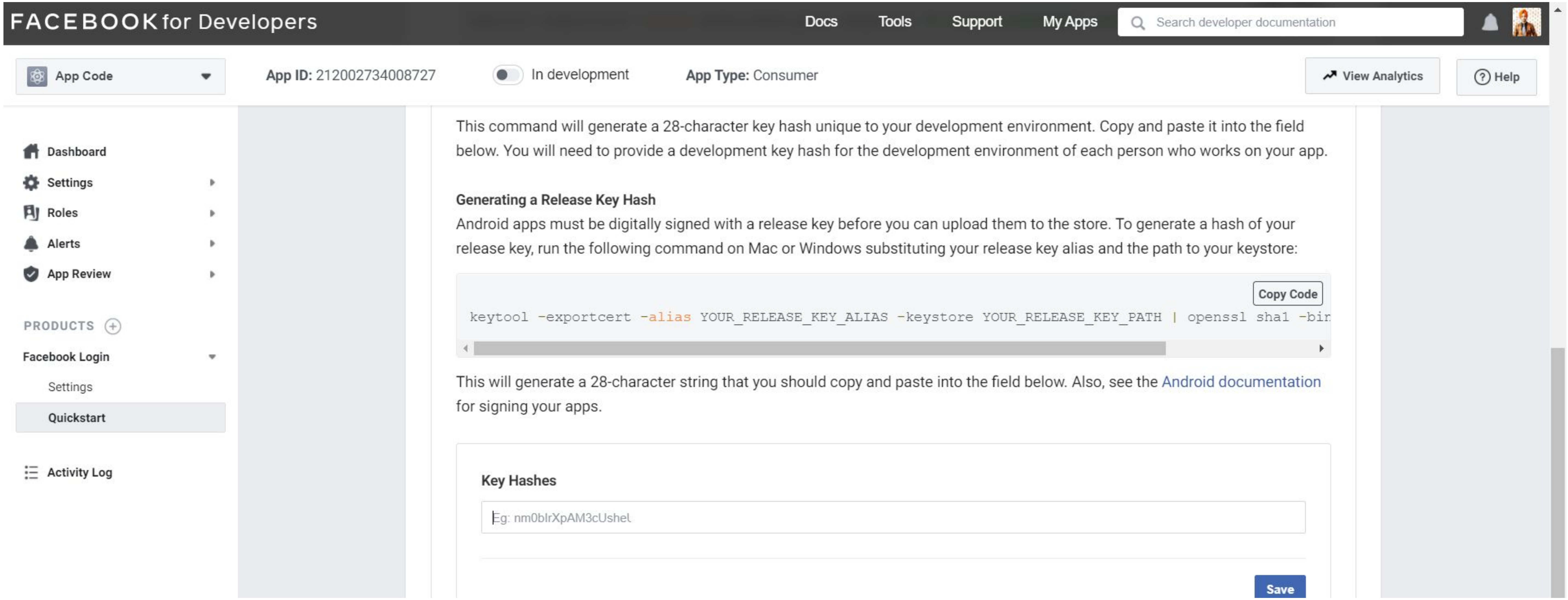
This is the fully qualified class name of the activity that handles deep linking such as com.example.app.DeepLinkingActivity. We use this when we deep link into your app from the Facebook app. You can also find this in your Android Manifest.

com.example.appcode.MainActivity

Now you can see that continue button is enabled click on "Continue" button and move toward next step

Step - 8

In this step we have to generate key hashes, For the same we need our project opened in android studio. it is assumed that your project is running and you also have .jsk file ready for your project.



App ID: 212002734008727

In development

App Type: Consumer

This command will generate a 28-character key hash unique to your development environment. Copy and paste it into the field below. You will need to provide a development key hash for the development environment of each person who works on your app.

Generating a Release Key Hash

Android apps must be digitally signed with a release key before you can upload them to the store. To generate a hash of your release key, run the following command on Mac or Windows substituting your release key alias and the path to your keystore:

```
keytool -exportcert -alias YOUR_RELEASE_KEY_ALIAS -keystore YOUR_RELEASE_KEY_PATH | openssl sha1 -bir
```

This will generate a 28-character string that you should copy and paste into the field below. Also, see the Android documentation for signing your apps.

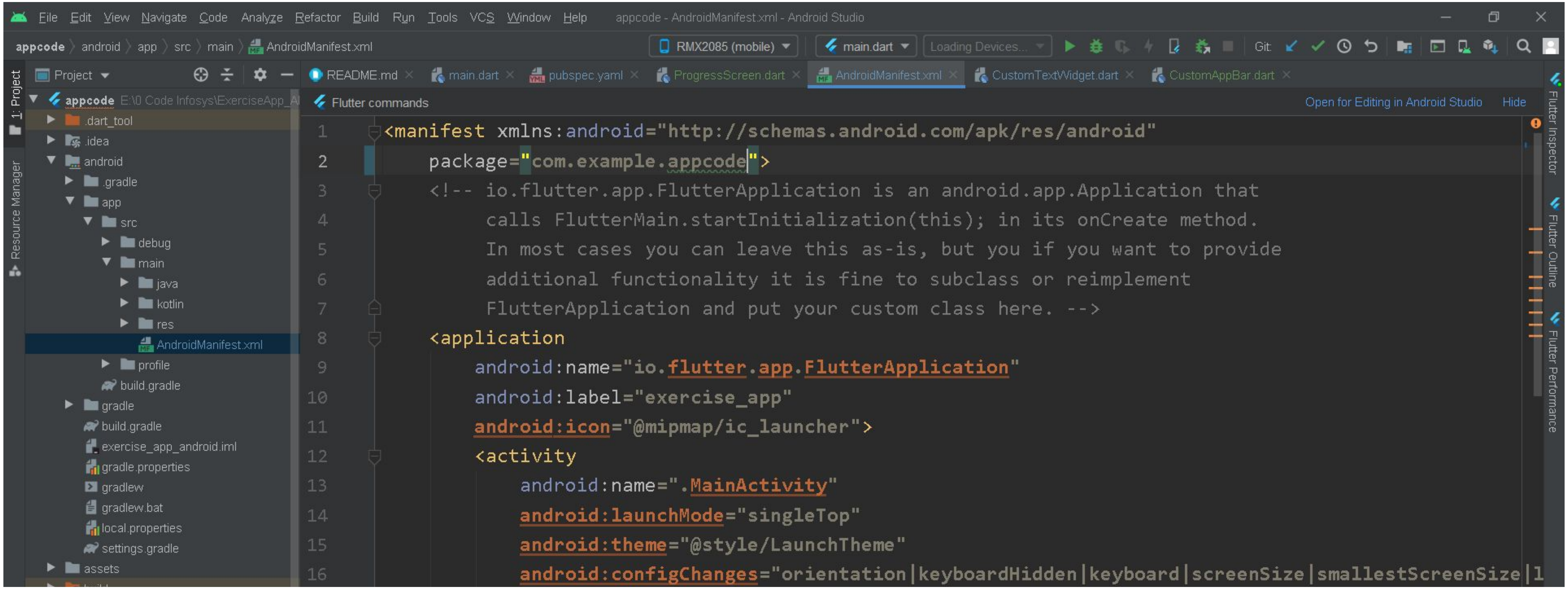
Key Hashes

Eg: nm0blrXpAM3cUshel

Step - 9

Navigate to AndroidManifest.xml file in your project

Root -> android -> app -> src -> main -> AndroidManifest.xml



```xml
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.appcode">
    <!-- io.flutter.app.FlutterApplication is an android.app.Application that
         calls FlutterMain.startInitialization(this); in its onCreate method.
         In most cases you can leave this as-is, but you if you want to provide
         additional functionality it is fine to subclass or reimplement
         FlutterApplication and put your custom class here. -->
    <application
        android:name="io.flutter.app.FlutterApplication"
        android:label="exercise_app"
        android:icon="@mipmap/ic_launcher">
        <activity
            android:name=".MainActivity"
            android:launchMode="singleTop"
            android:theme="@style/LaunchTheme"
            android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|l
```