

Setting Up Facebook Authentication in Firebase for a Test Game

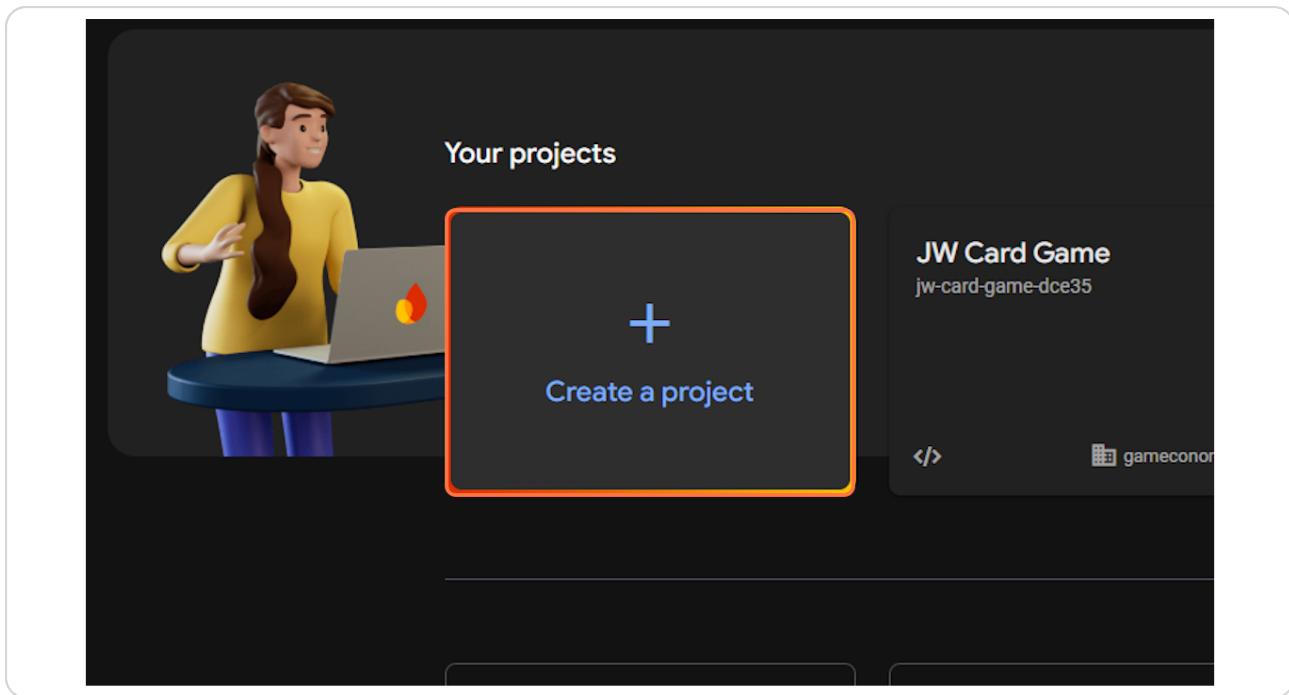
61 Steps

[View most recent version on Tango.us](#) 

Created by
Chester Lok

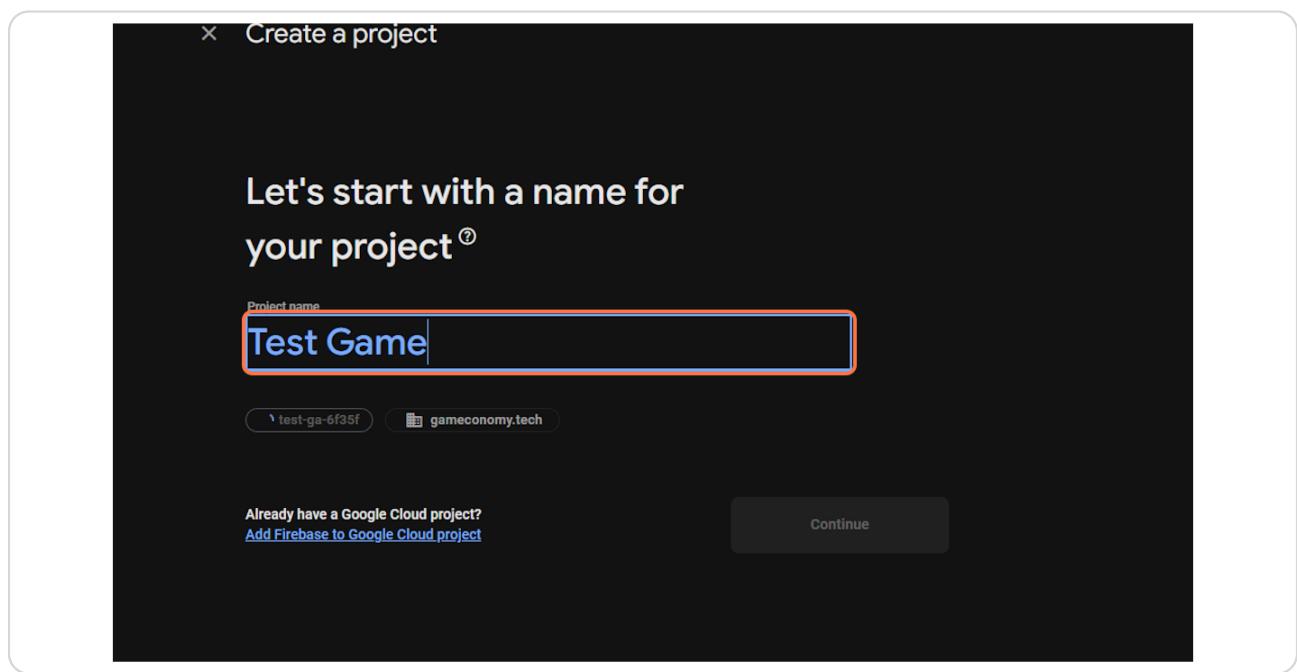
Creation Date
Jan 02, 2025

Last Updated
Jan 02, 2025

STEP 1**Click on add...**

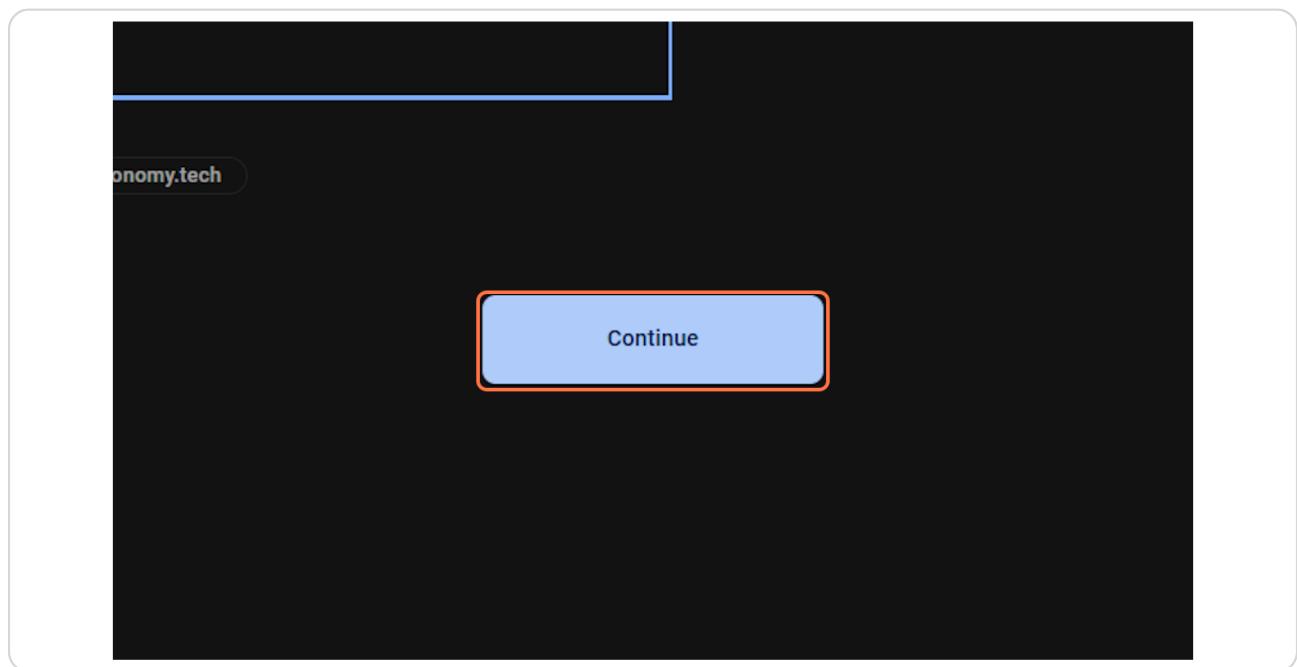
STEP 2

Type "Test Game"



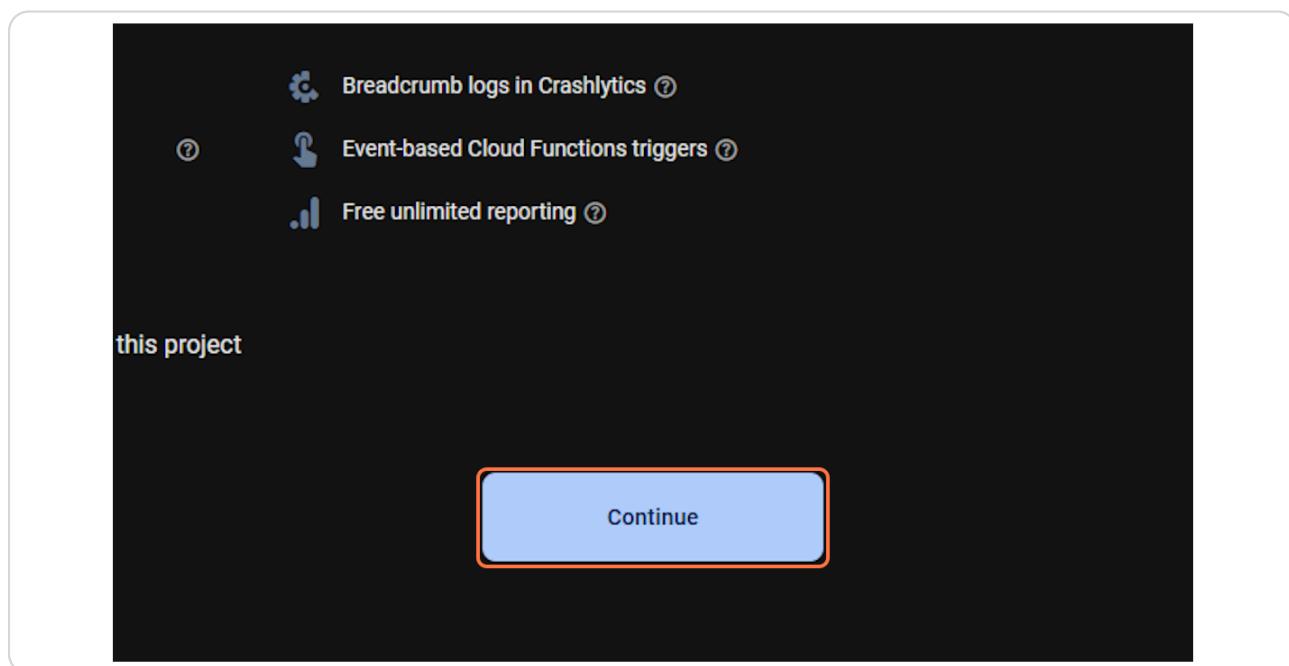
STEP 3

Click on Continue



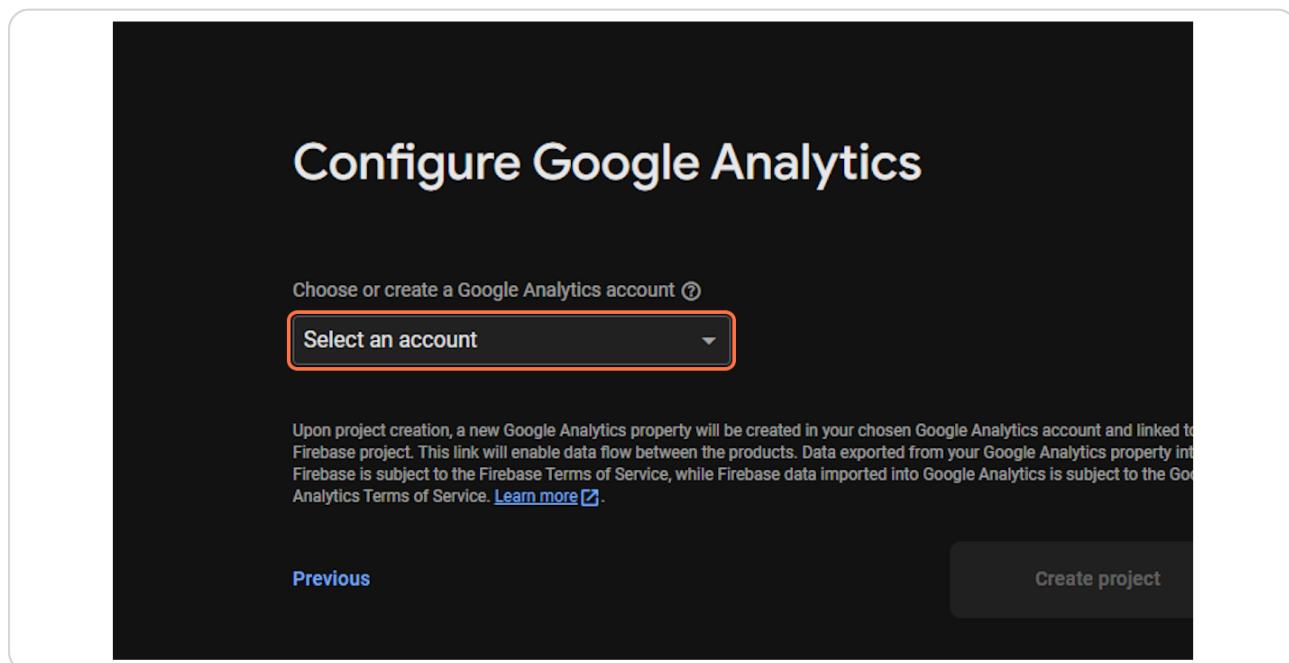
STEP 4

Click on Continue



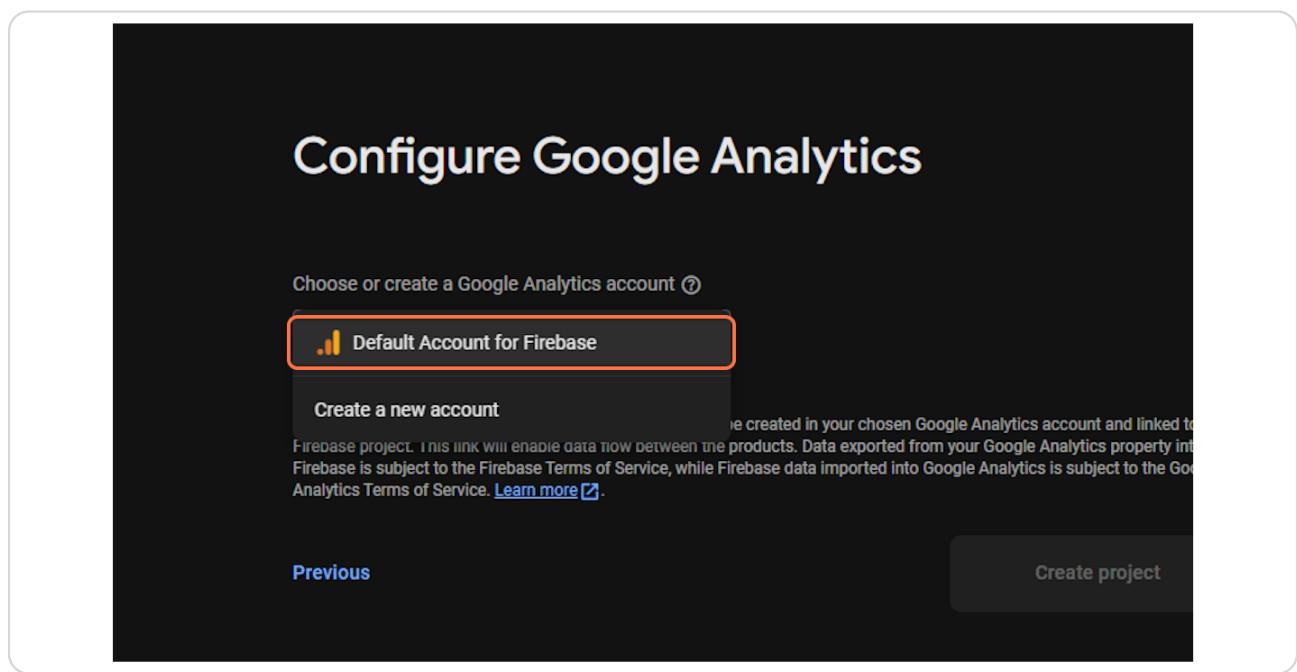
STEP 5

Click on Select an account...



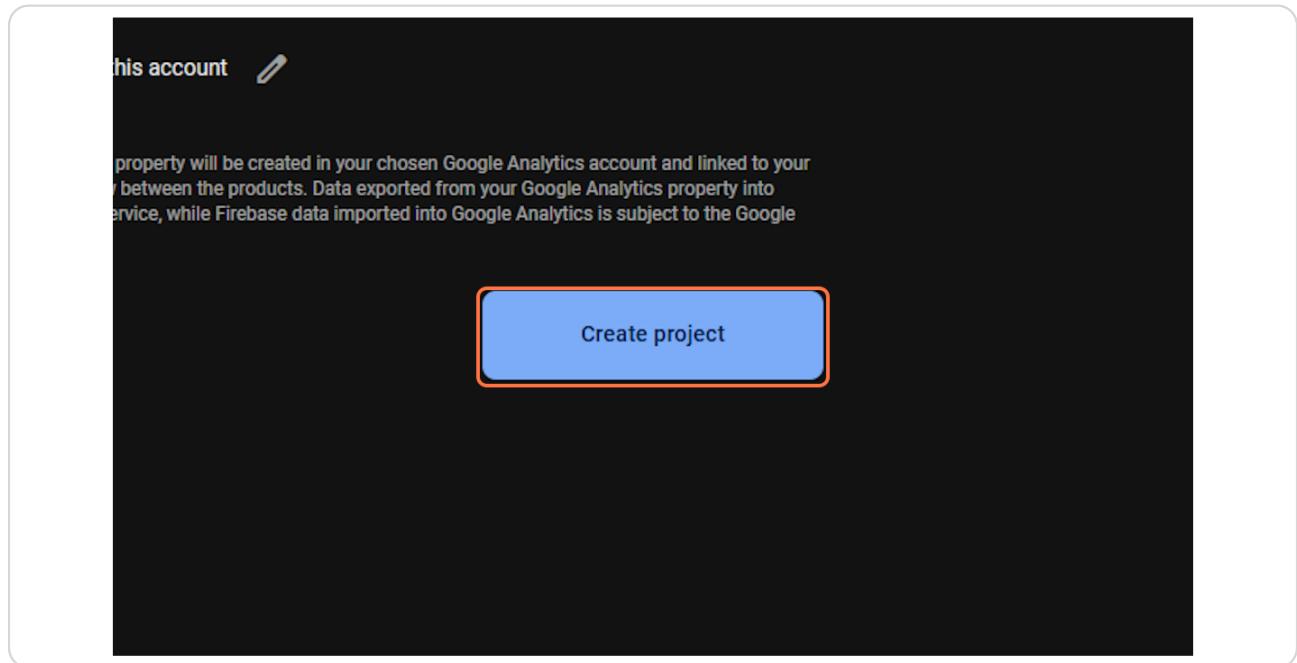
STEP 6

Click on Default Account for Firebase



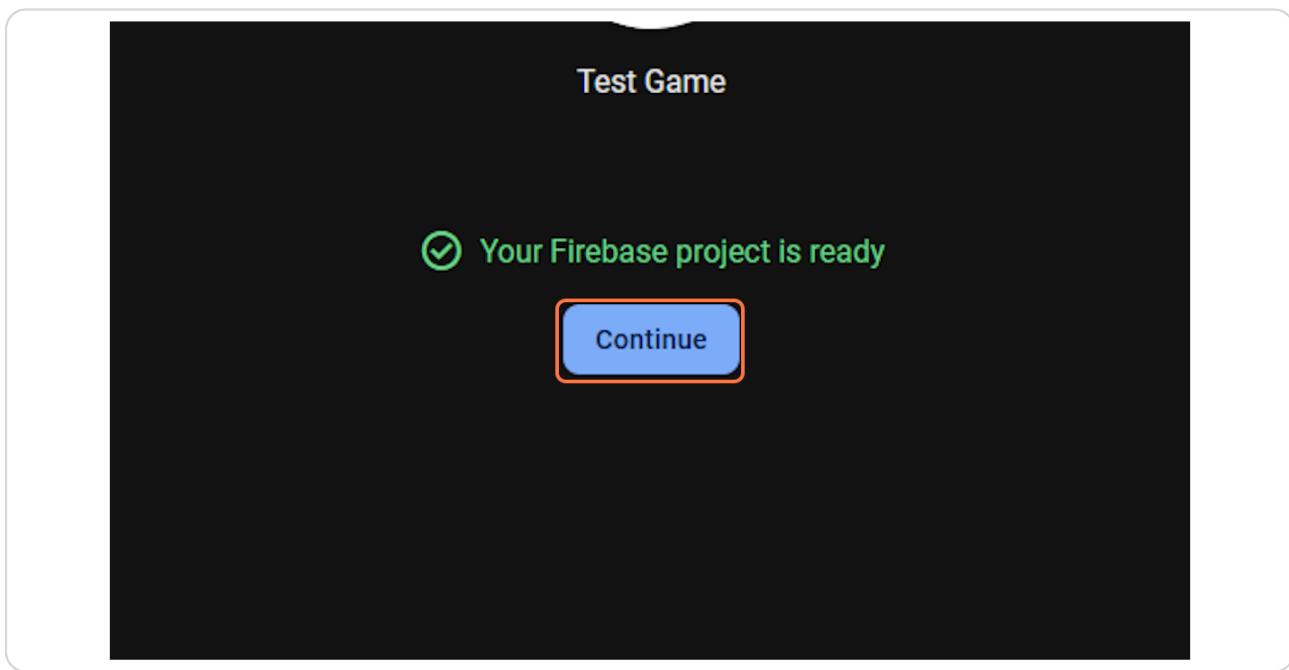
STEP 7

Click on Create project



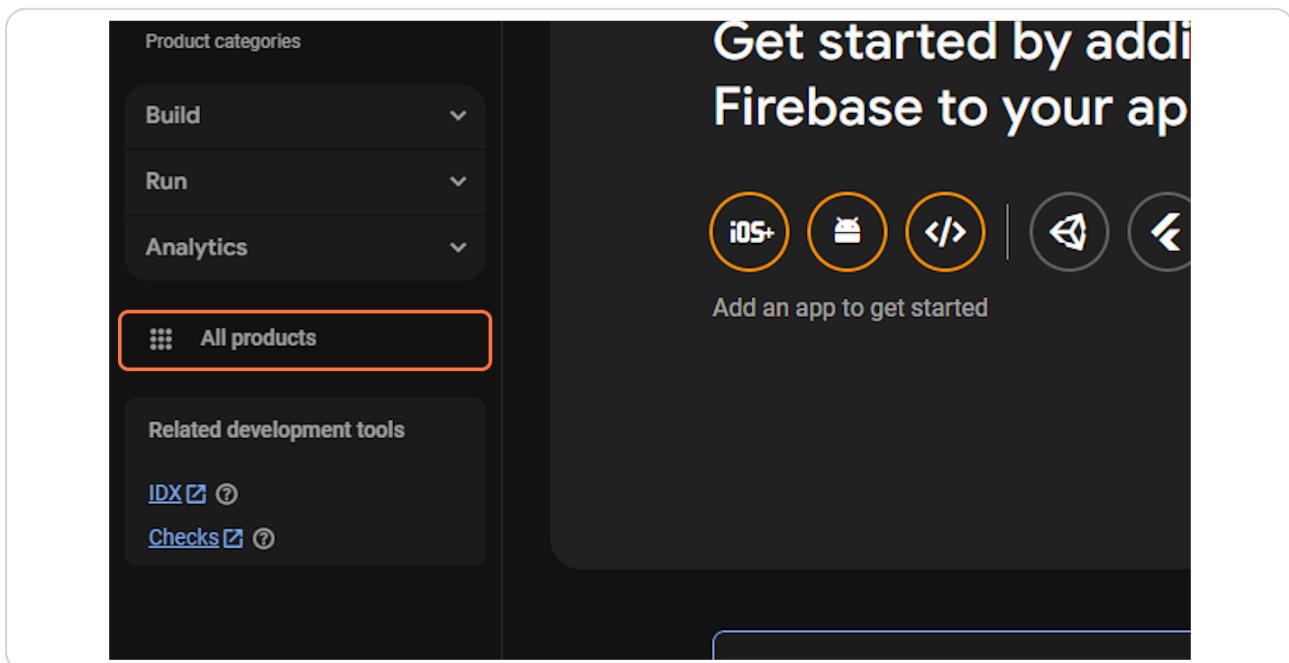
STEP 8

Click on Continue



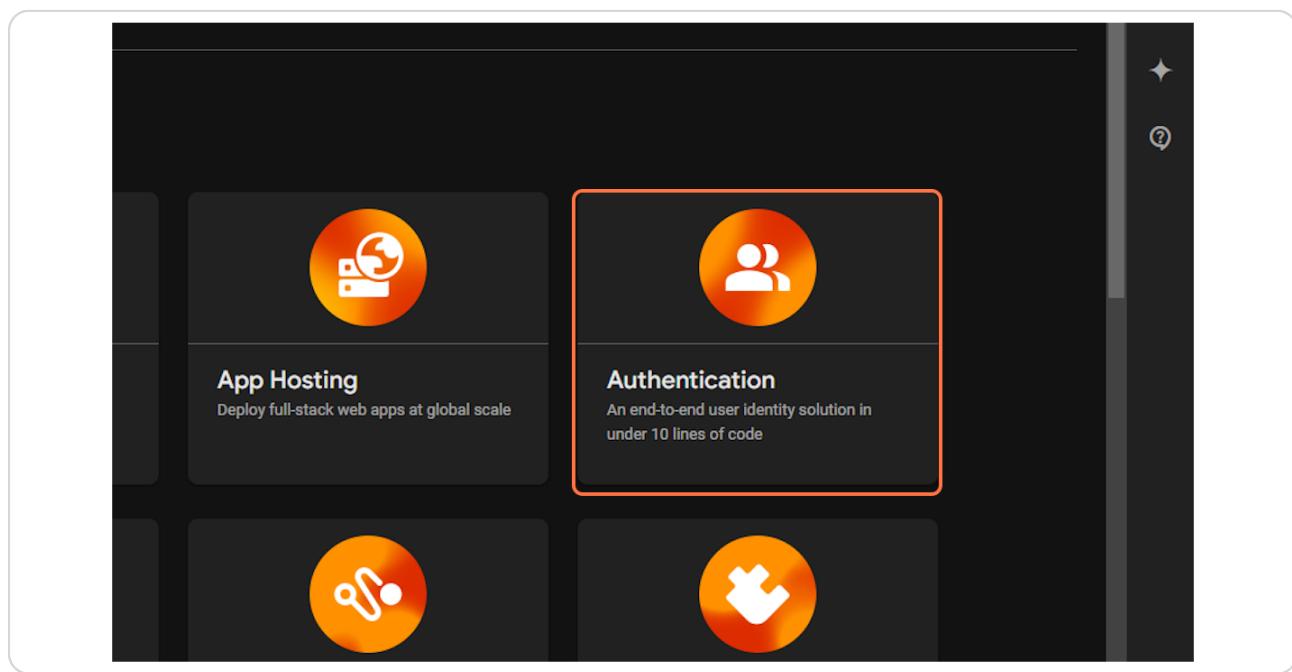
STEP 9

Click on apps...



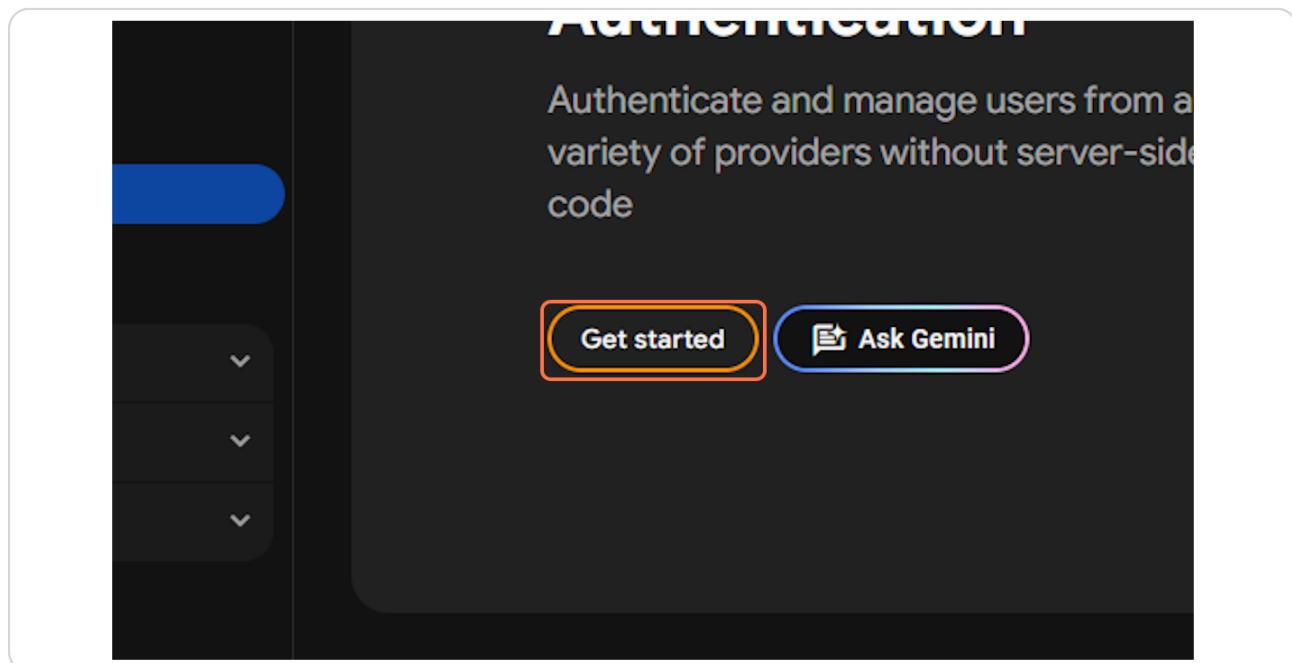
STEP 10

Click on Authentication...



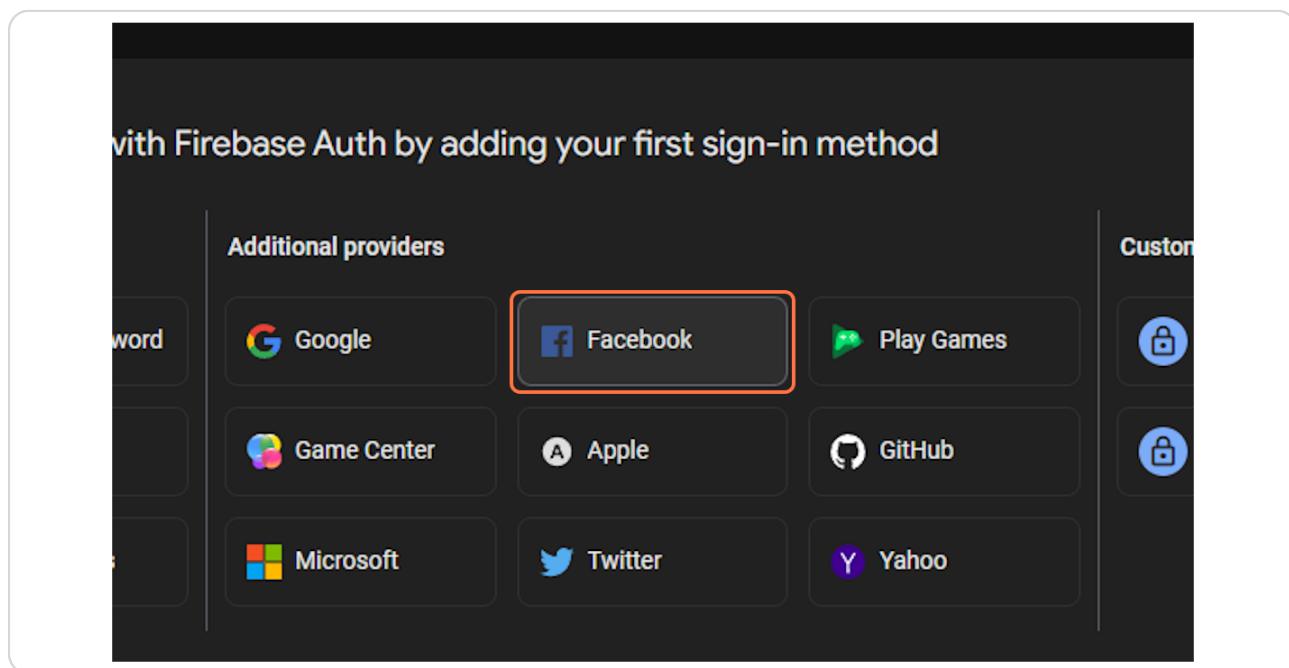
STEP 11

Click on Get started



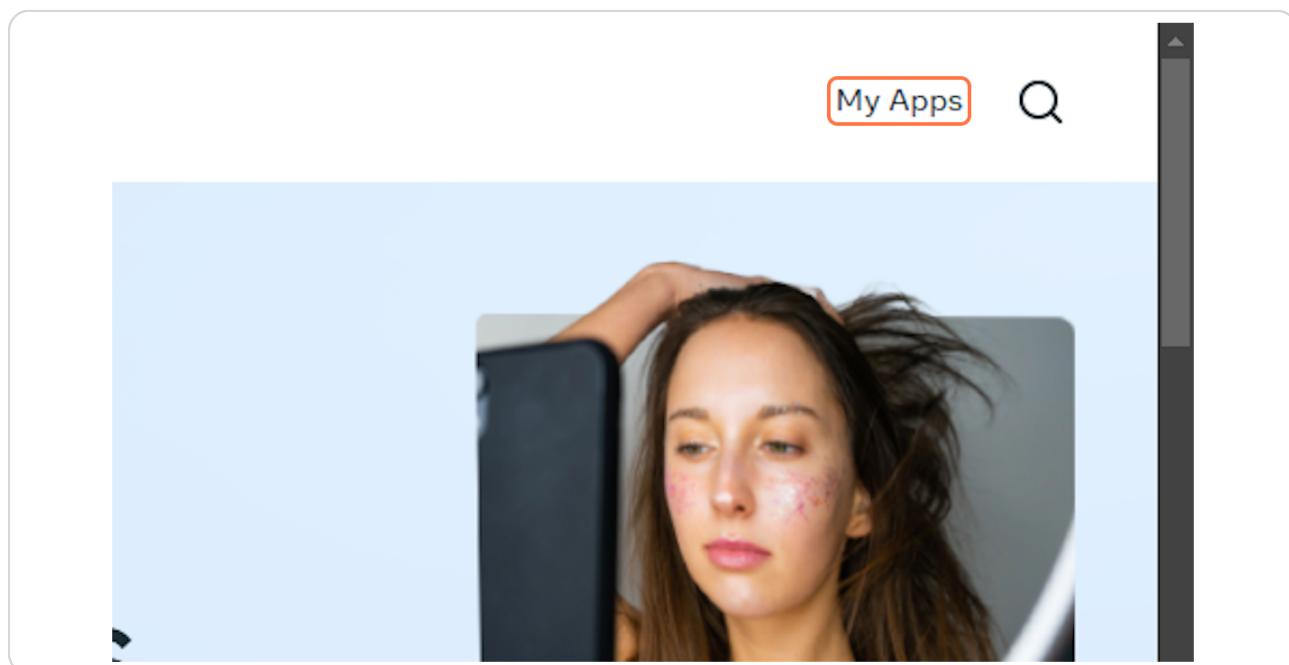
STEP 12

Click on Facebook



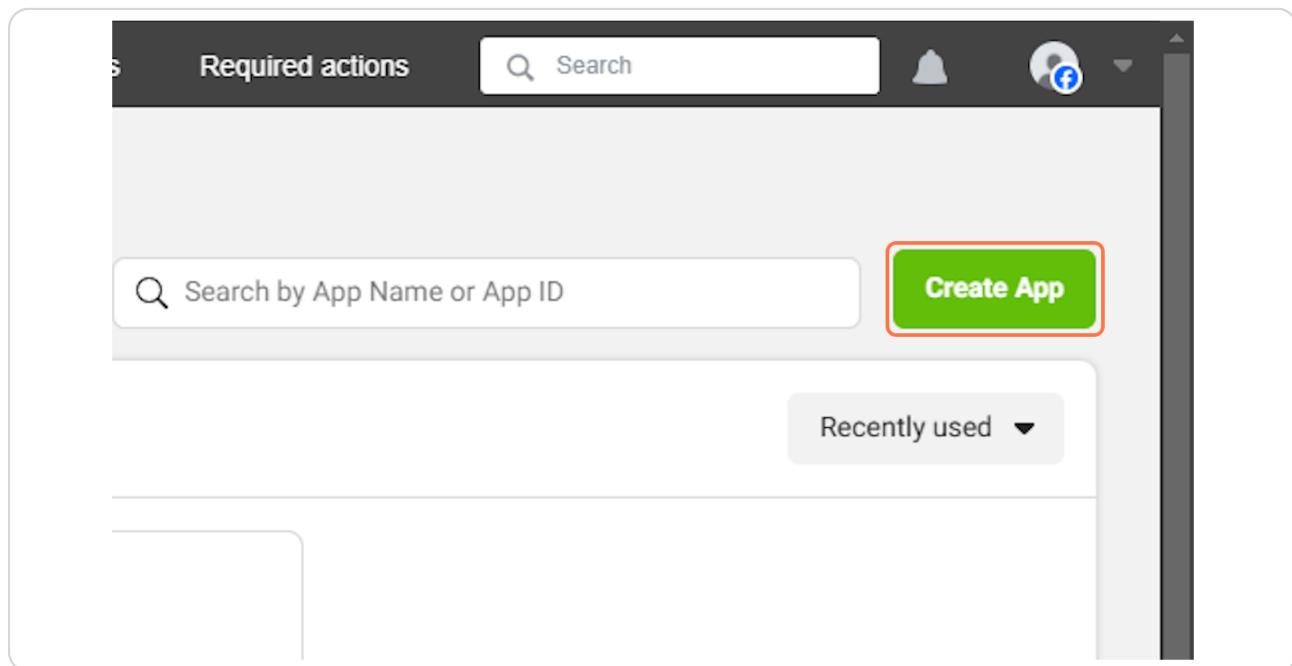
STEP 13

Click on My Apps



STEP 14

Click on Create App



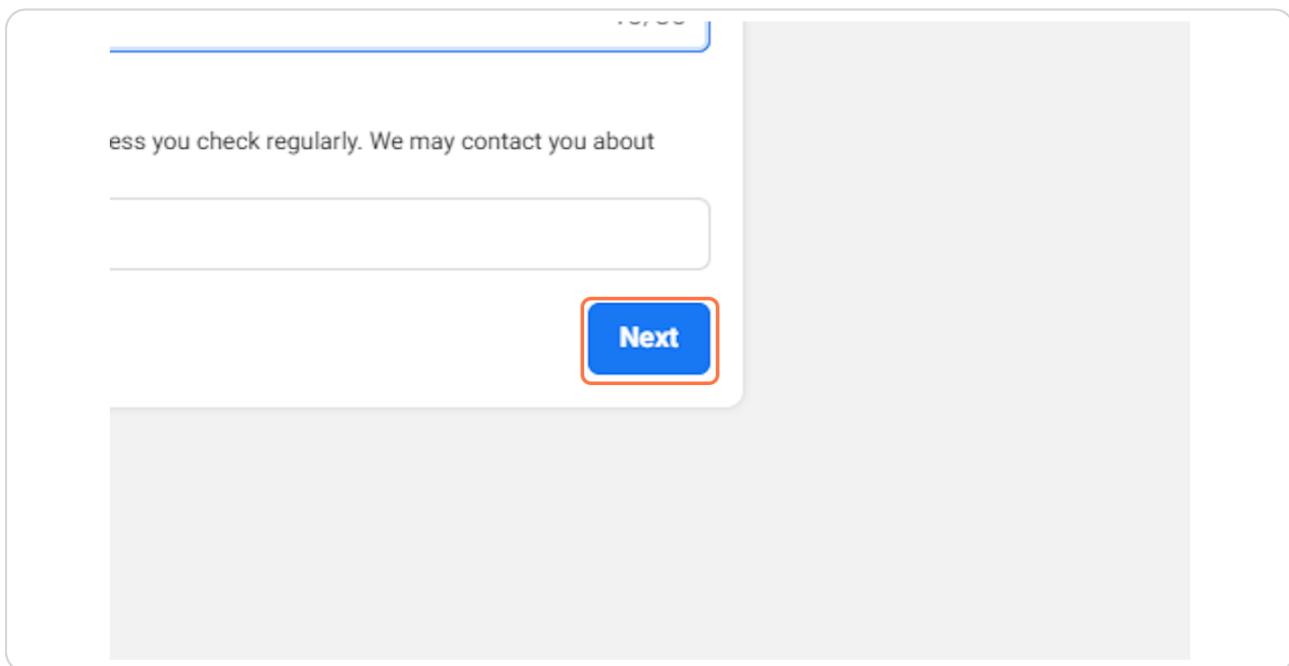
STEP 15

Type "Sample App"

A screenshot of a "Create App" form. The top navigation bar includes links for "Docs", "Tools", "Support", "Apps", and "Required actions", along with a search bar. On the left, a sidebar has tabs for "App details" (which is selected and highlighted in blue), "Use cases", "Business", and "Finalize". The main form area starts with a section for "App name", which contains the instruction "This is the app name that will show on your My Apps page and associated with your app ID. You can change the name later in Settings." Below this is an input field containing the text "Sample App", with a character count of "10/30" shown at the end. The next section is "App contact email", with the instruction "This is the email address we'll use to contact you about your app. Make sure it is an address you check regularly. We may contact you about policies, app restrictions or recovery if your app is deleted or compromised." Below this is an input field containing the email "chester.lok@hashcodestudio.com". At the bottom of the form are two buttons: "Cancel" on the left and "Next" on the right.

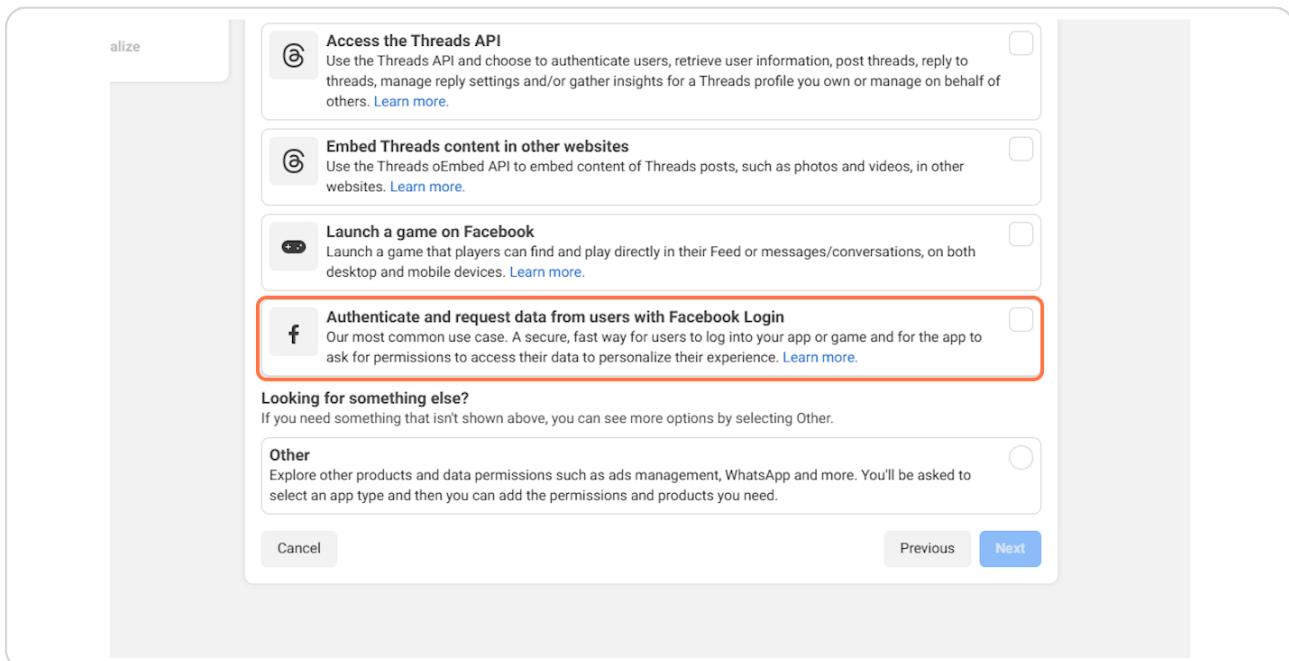
STEP 16

Click on Next



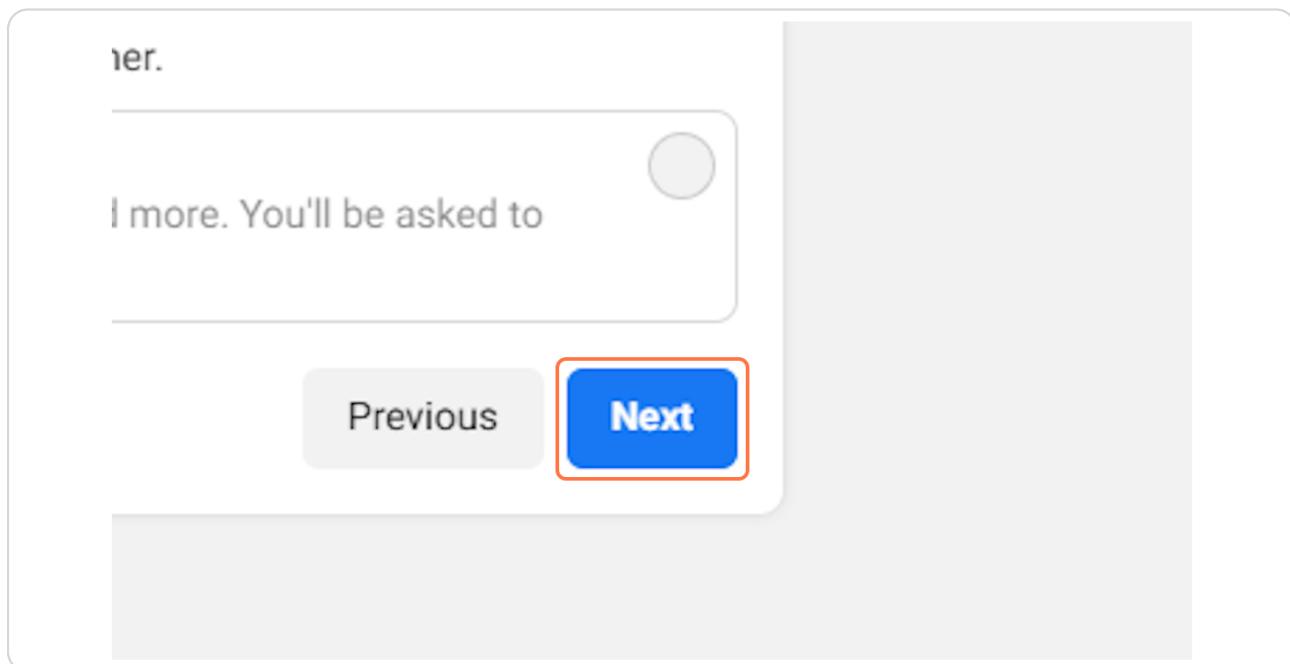
STEP 17

Click on Authenticate and request data from users with Facebook Login...



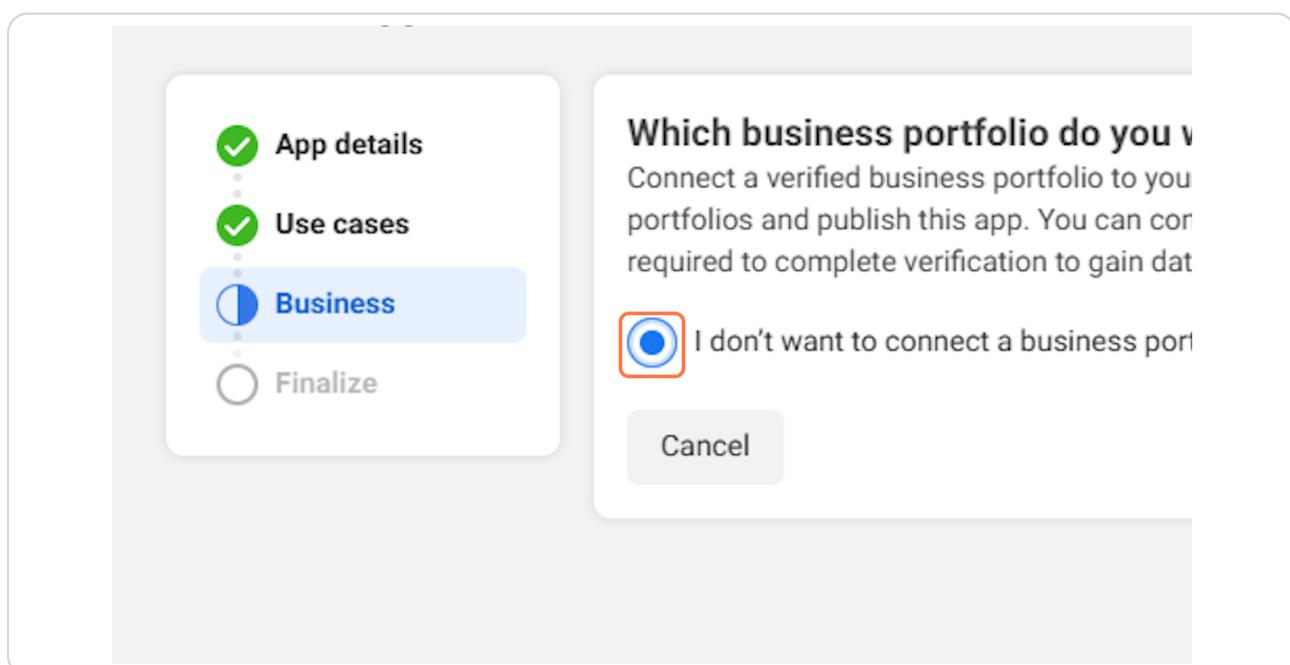
STEP 18

Click on Next



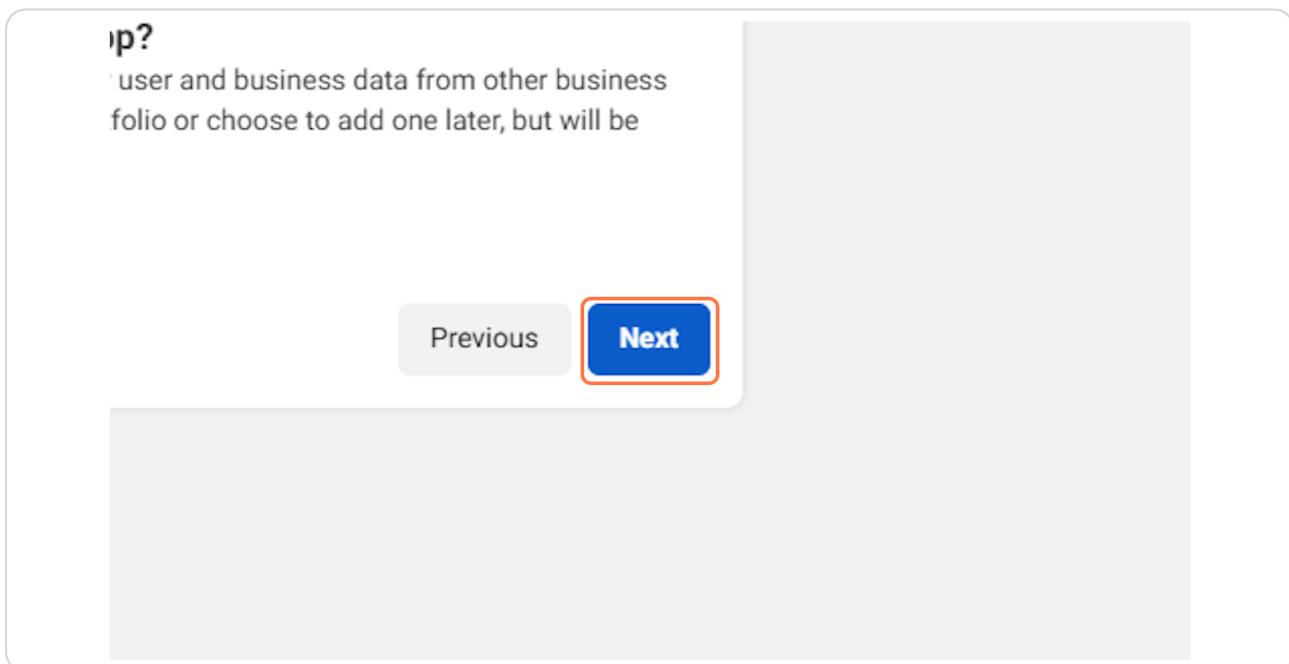
STEP 19

Select I don't want to connect a business portfolio yet.



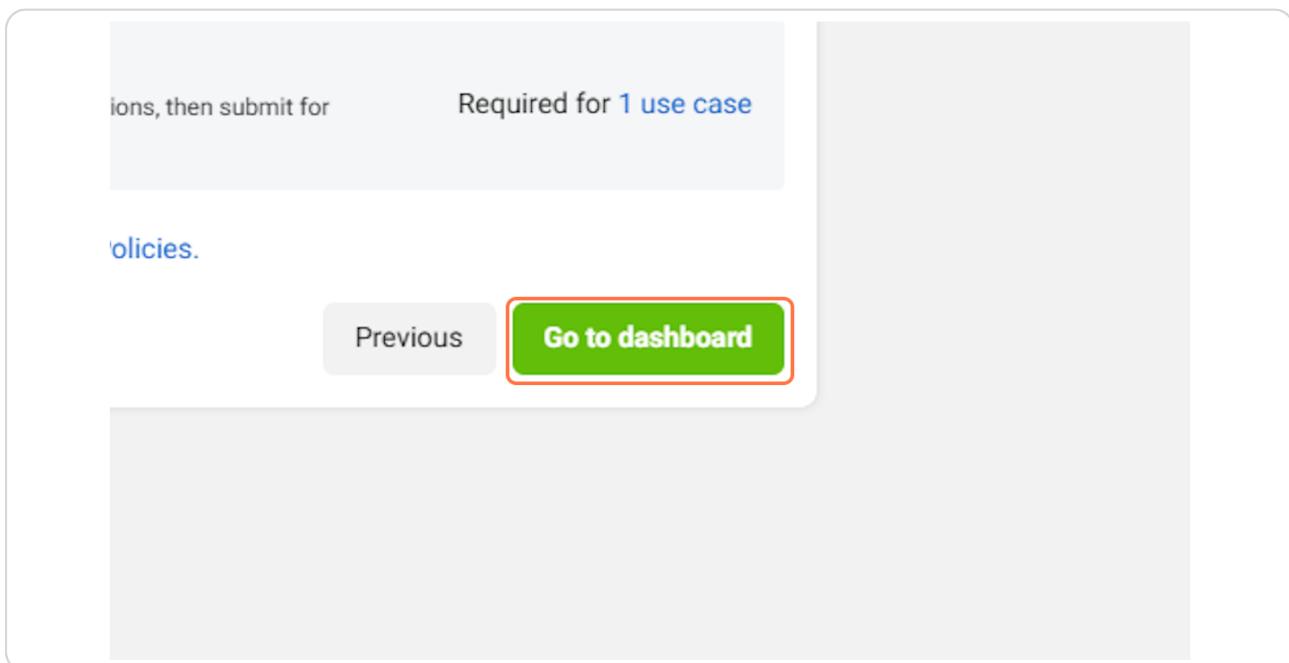
STEP 20

Click on Next



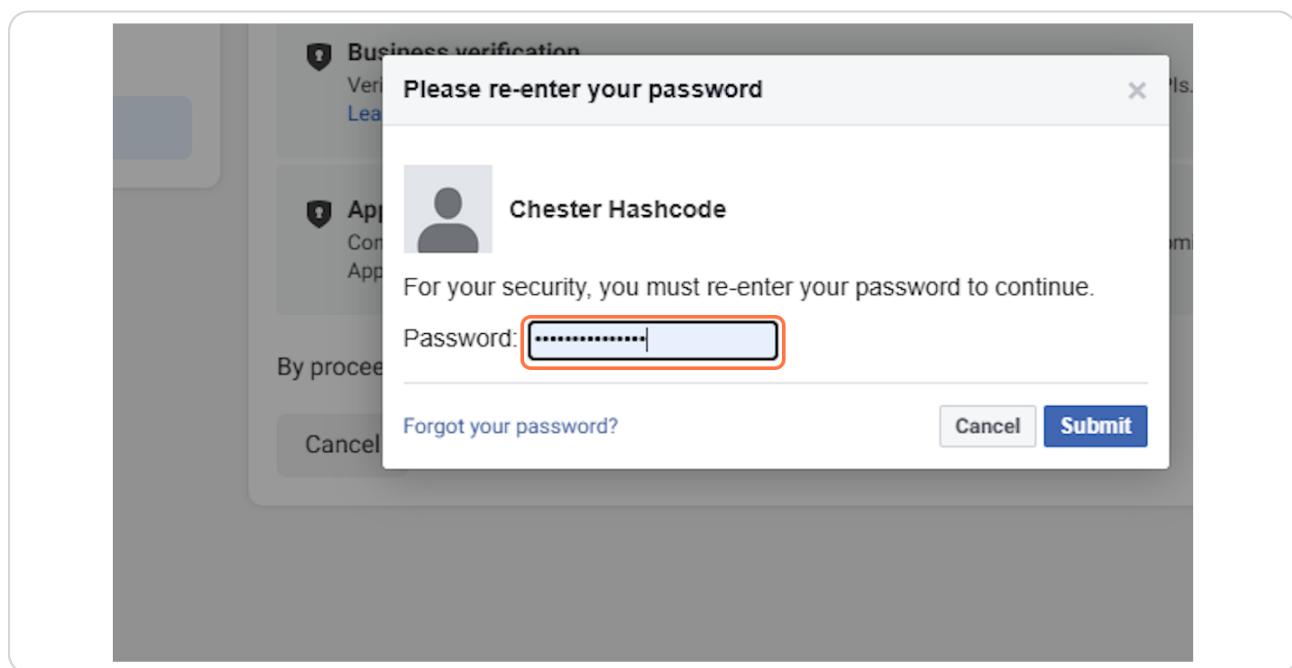
STEP 21

Click on Go to dashboard



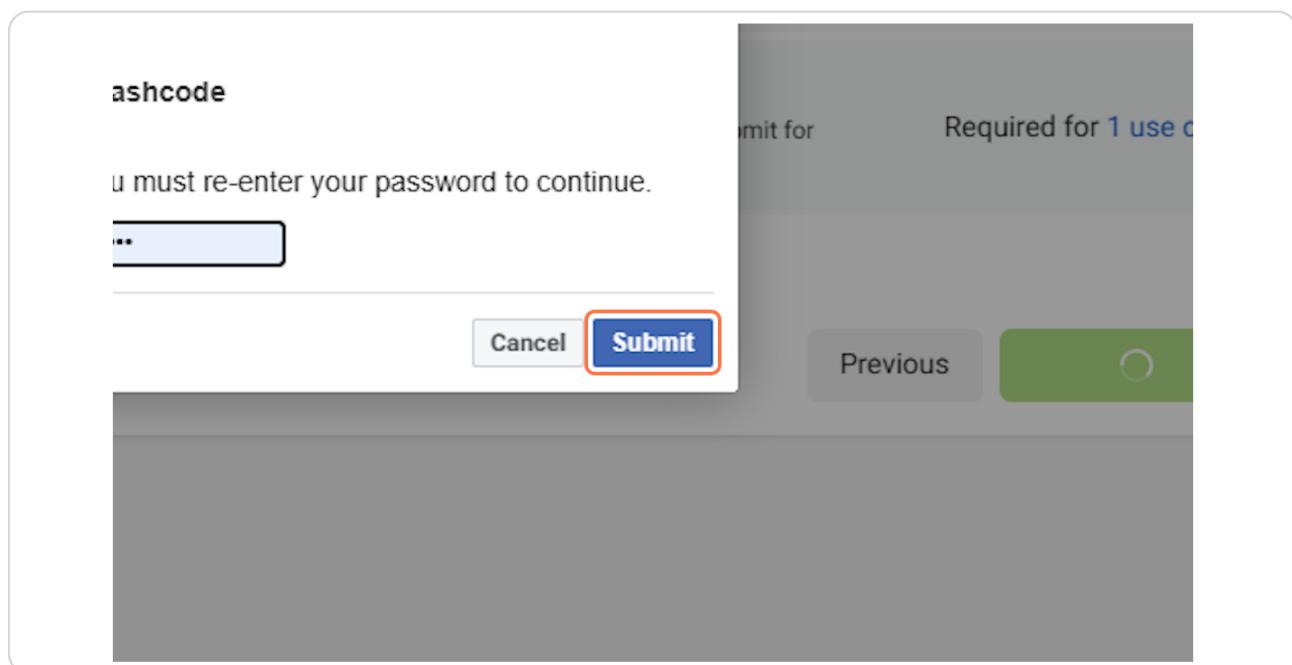
STEP 22

Type password



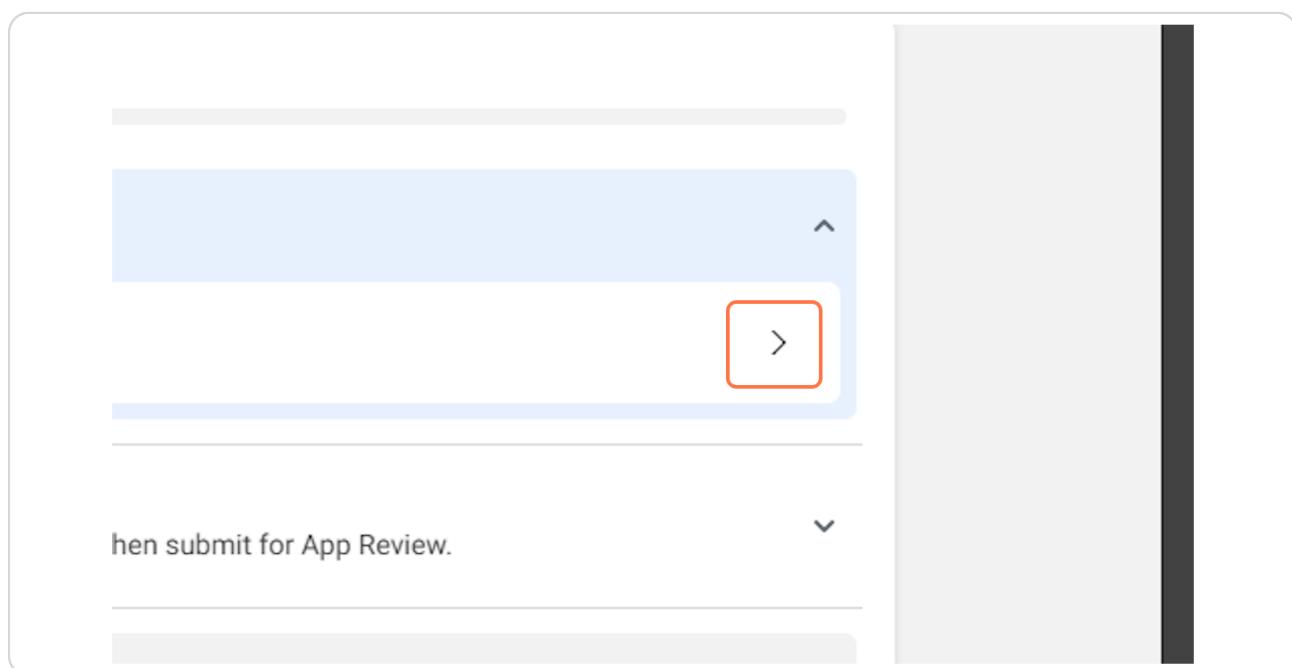
STEP 23

Click on Submit



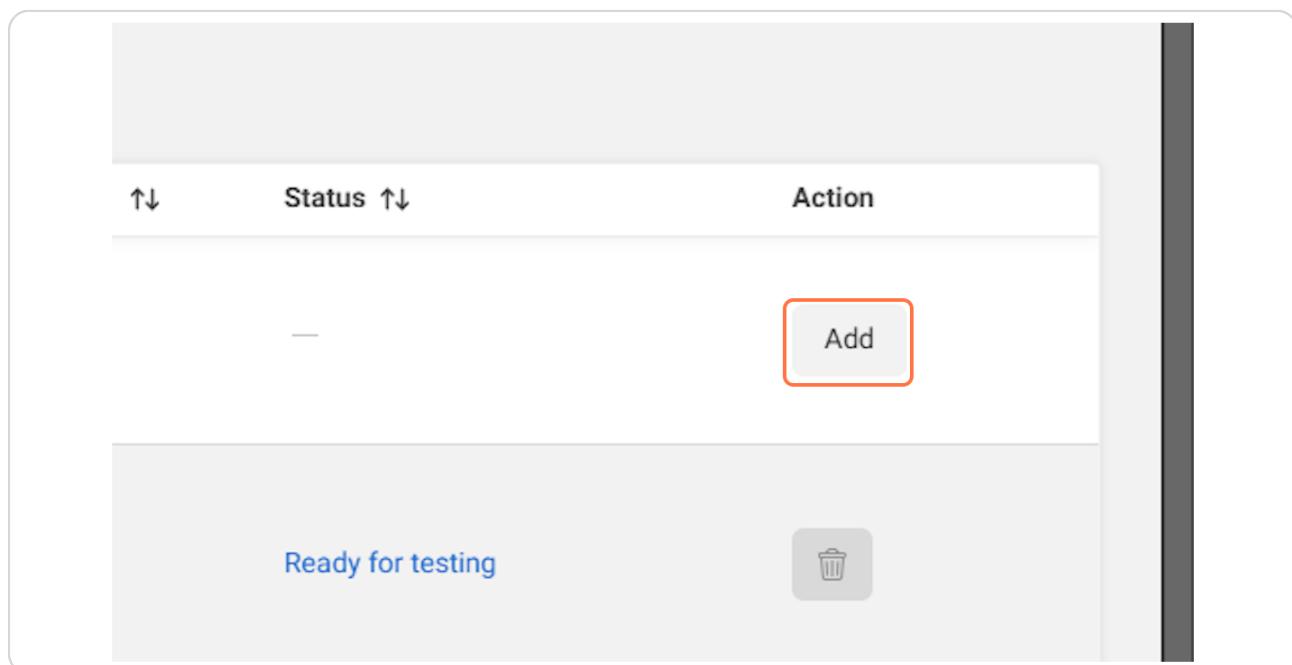
STEP 24

Click on



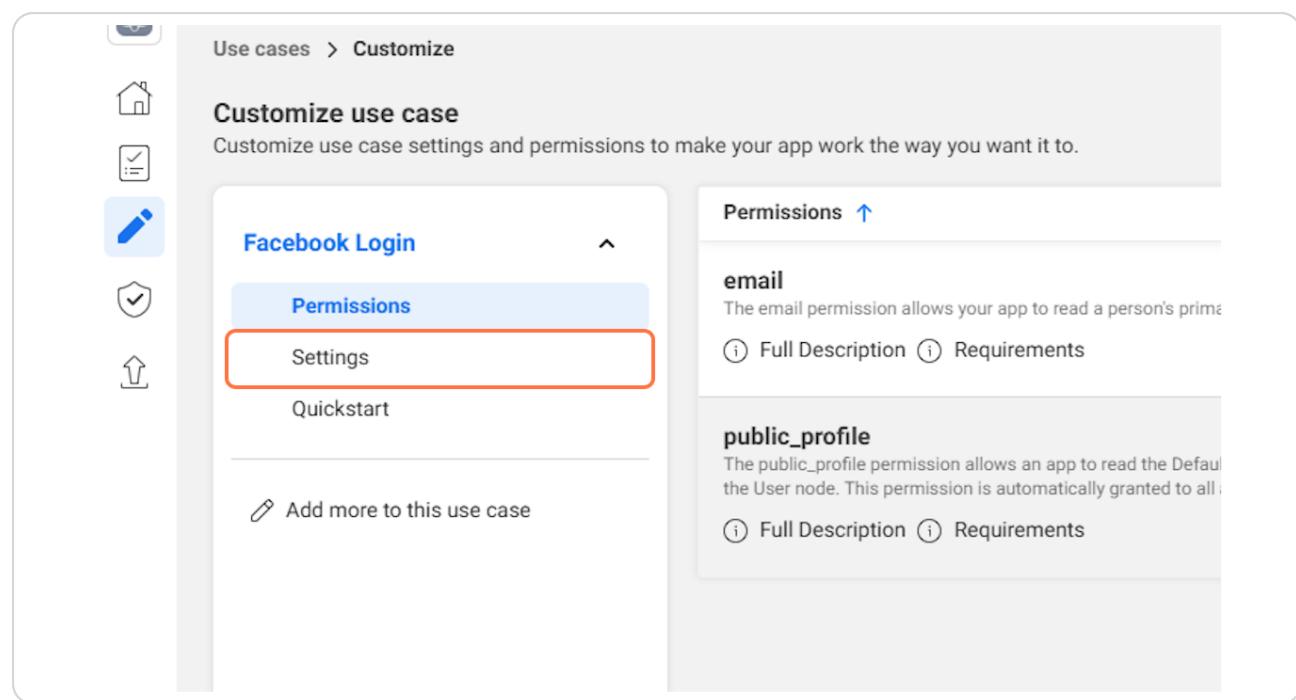
STEP 25

Click on Add



STEP 26

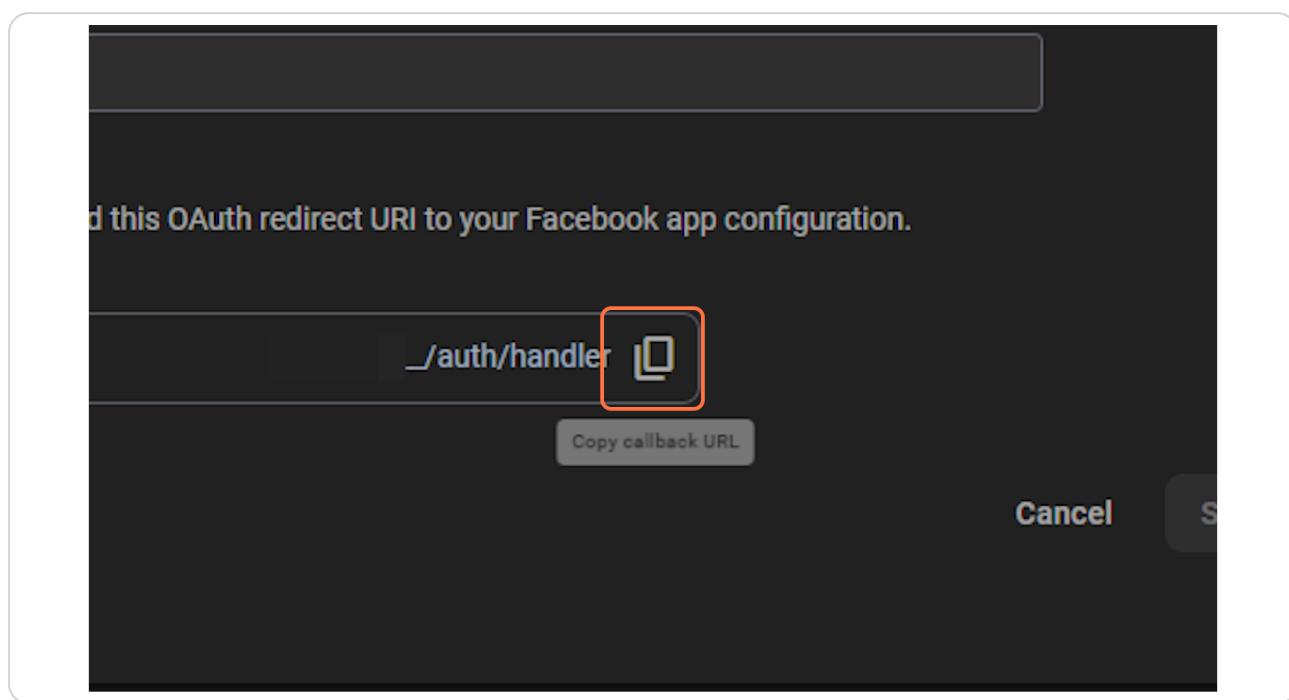
Click on Settings



G Test Game – Authentication – Sign-in method –... 1 Step ↗

STEP 27

Click on content_copy



 Sample App - Use cases - Meta for Developers 8 Steps 

STEP 28

Paste "https://*****.firebaseapp.com/_/auth/handler" into input

When on, prompts people to enter their Facebook password in order to log in on the web. [?]

Yes No Use Strict Mode for redirect URIs Only allow redirects that exactly match the Valid OAuth Red

Valid OAuth Redirect URIs
A manually specified redirect_uri used with Login on the web must exactly m This list is also used by the JavaScript SDK for in-app browsers that suppres

Login from Devices
Enables the OAuth client login flow for devices like a smart TV [?]

Allowed Domains for the JavaScript SDK
Login and signed-in functionality of the JavaScript SDK will only be available

STEP 29

Click on Facebook Login...

Client OAuth login Yes Enables the standard OAuth client token flow. Secure your application and prevent abuse by locking down which token redirect URIs are allowed with the options below. Disable globally if not used. [?]

Web OAuth login Yes Enables web-based Client OAuth Login. [?]

Enforce HTTPS Yes Enforce the use of HTTPS for Redirect URIs and the JavaScript SDK. Strongly recommended. [?]

Force Web OAuth reauthentication No When on, prompts people to enter their Facebook password in order to log in on the web. [?]

Embedded Browser OAuth Login No Enable webview Redirect URIs for Client OAuth Login. [?]

Use Strict Mode for redirect URIs Yes Only allow redirects that exactly match the Valid OAuth Redirect URIs. Strongly recommended. [?]

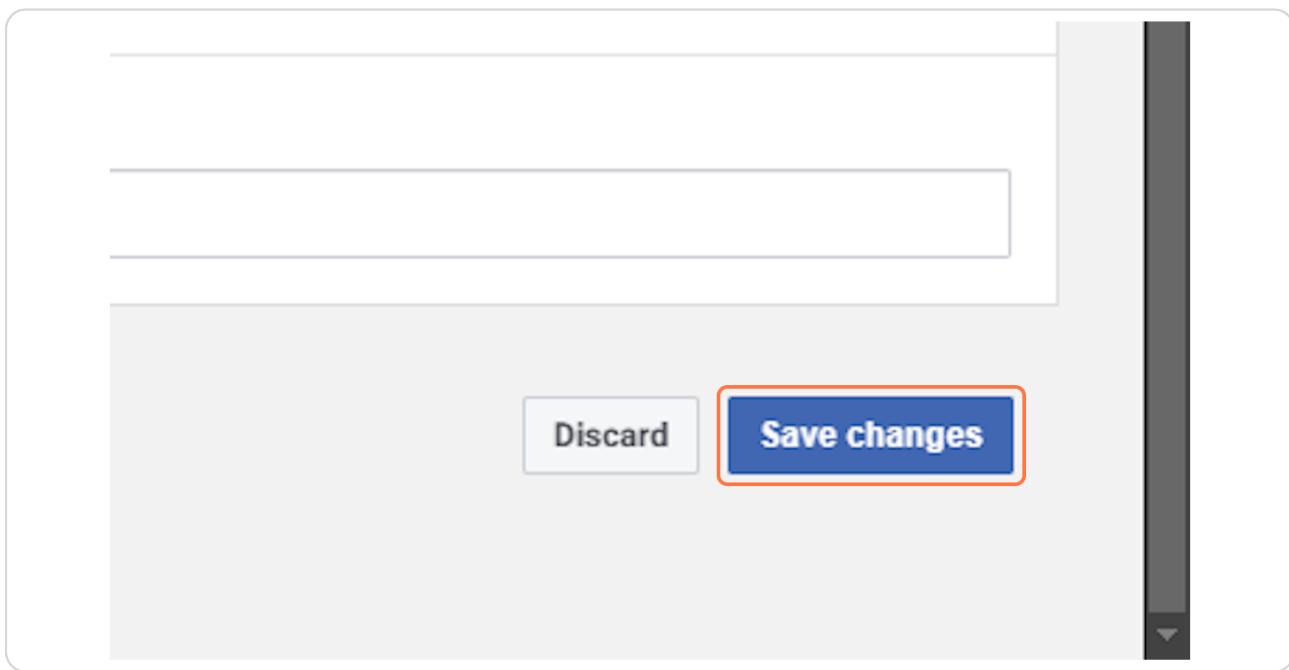
Valid OAuth Redirect URIs
A manually specified redirect_uri used with Login on the web must exactly match one of the URIs listed here. This list is also used by the JavaScript SDK for in-app browsers that suppress popups. [?]

Allowed Domains for the JavaScript SDK
Login and signed-in functionality of the JavaScript SDK will only be available on these domains. [?]

Deauthorize
Deauthorize callback URL
What should we ping when a user deauthorizes your app?

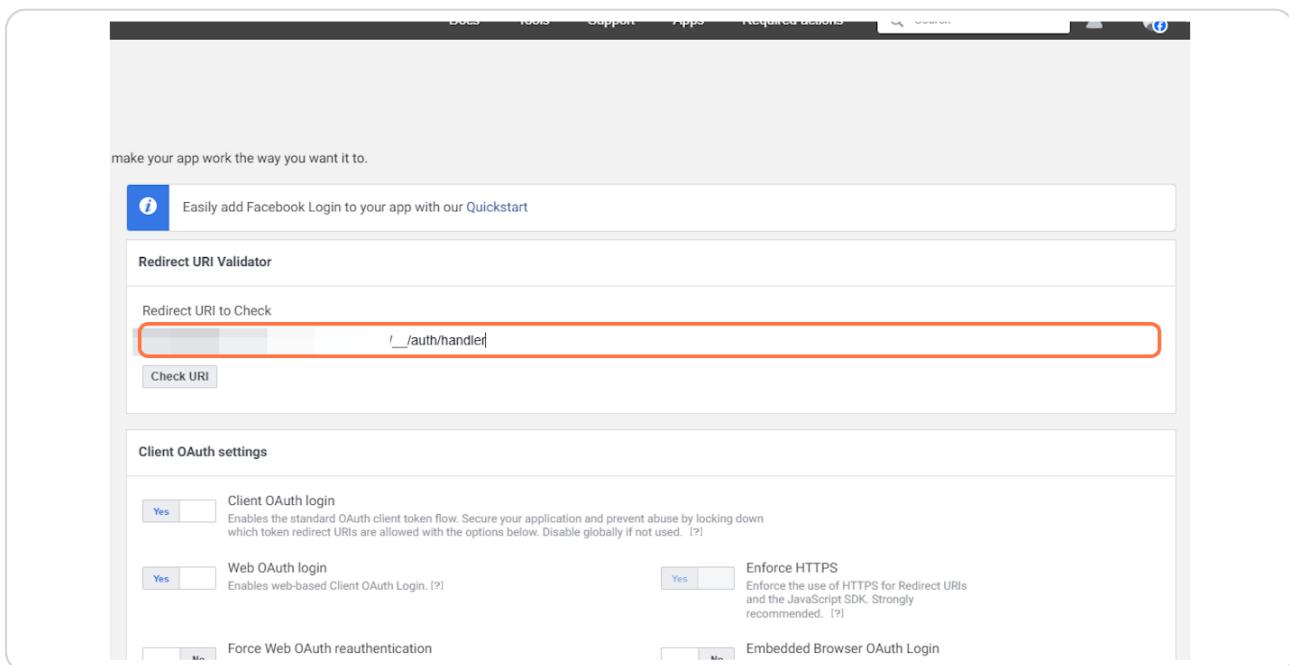
STEP 30

Click on Save changes



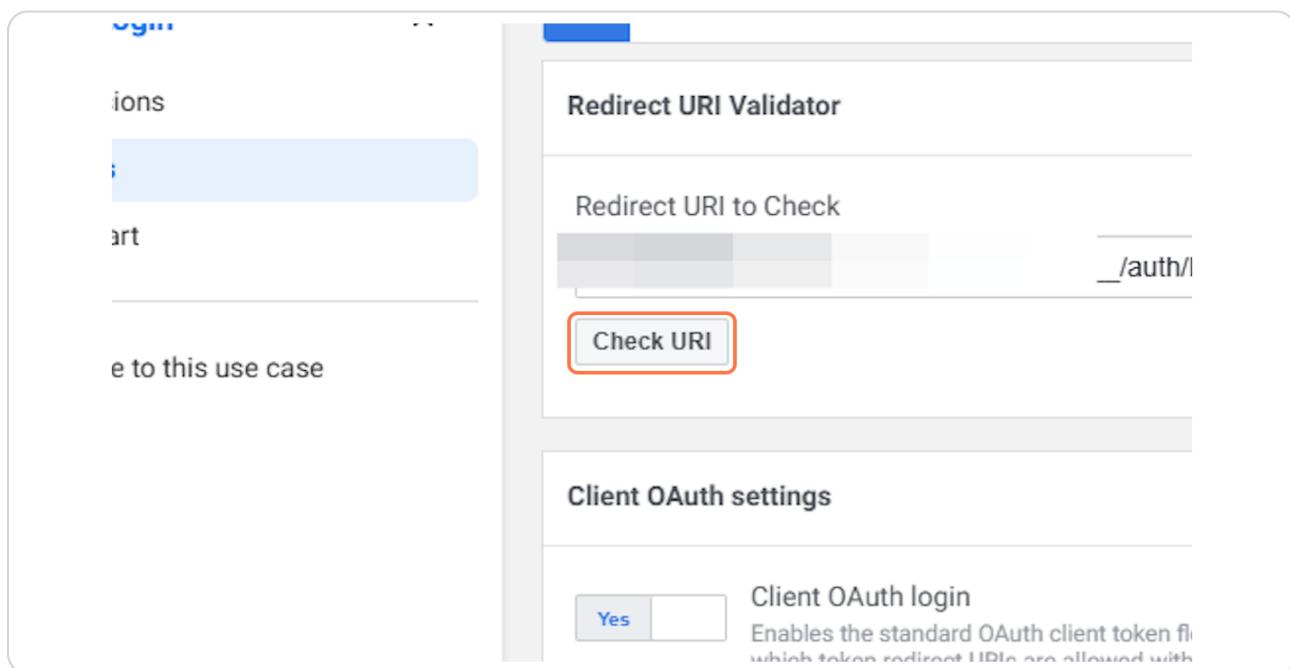
STEP 31

Paste "https://*****.firebaseapp.com/__/auth/handler" into input



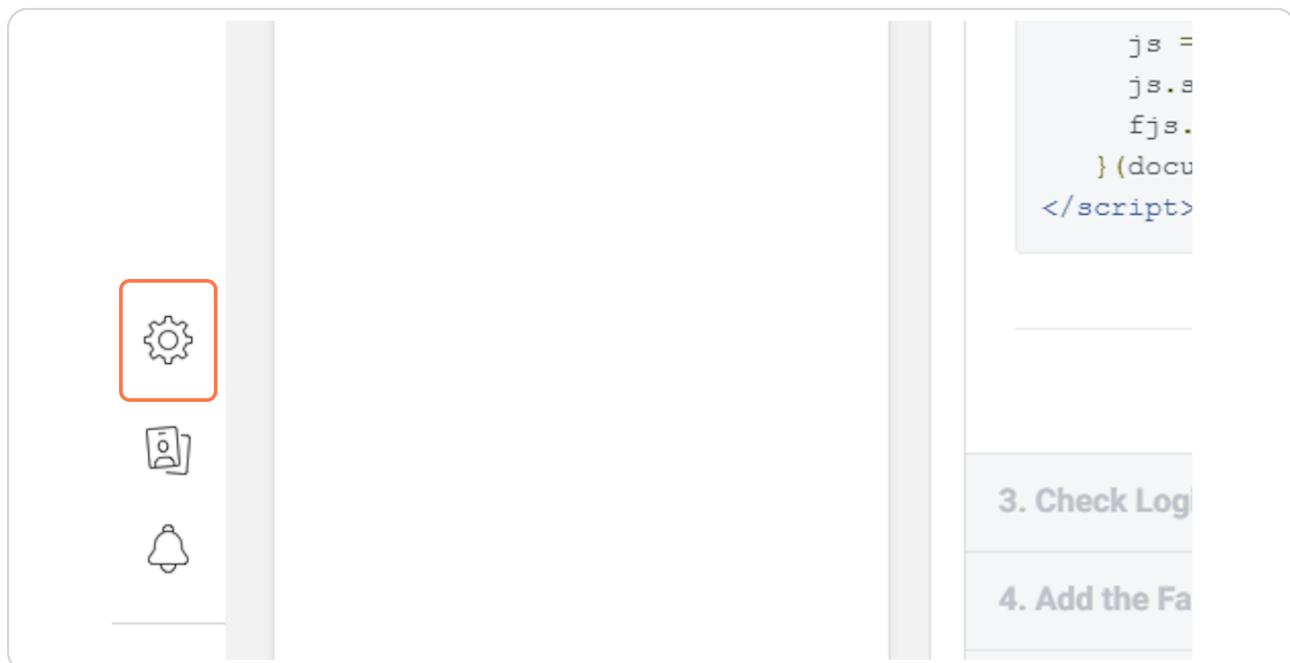
STEP 32

Click on Check URI



STEP 33

Click on App settings...



STEP 34

Click on Basic

The screenshot shows the Facebook App Settings interface. On the left, there's a sidebar with 'App settings' at the top, followed by 'Basic' (which is highlighted with a red border), 'Advanced', 'App roles' (with a dropdown arrow), and 'Alert Inbox'. Below the sidebar are three icons: a grid icon, a question mark icon, and a clipboard icon. To the right of the sidebar, there's a large text area containing a script snippet:

```
if (a.getElementById(id)) {return}
js = d.createElement(s); js.id = id;
js.src = "https://connect.facebook.net";
fjs.parentNode.insertBefore(js, document, 'script', 'facebook-j');
</script>
```

On the far right, there are three numbered steps: '3. Check Login Status', '4. Add the Facebook Login Button', and '5. Next Steps'.

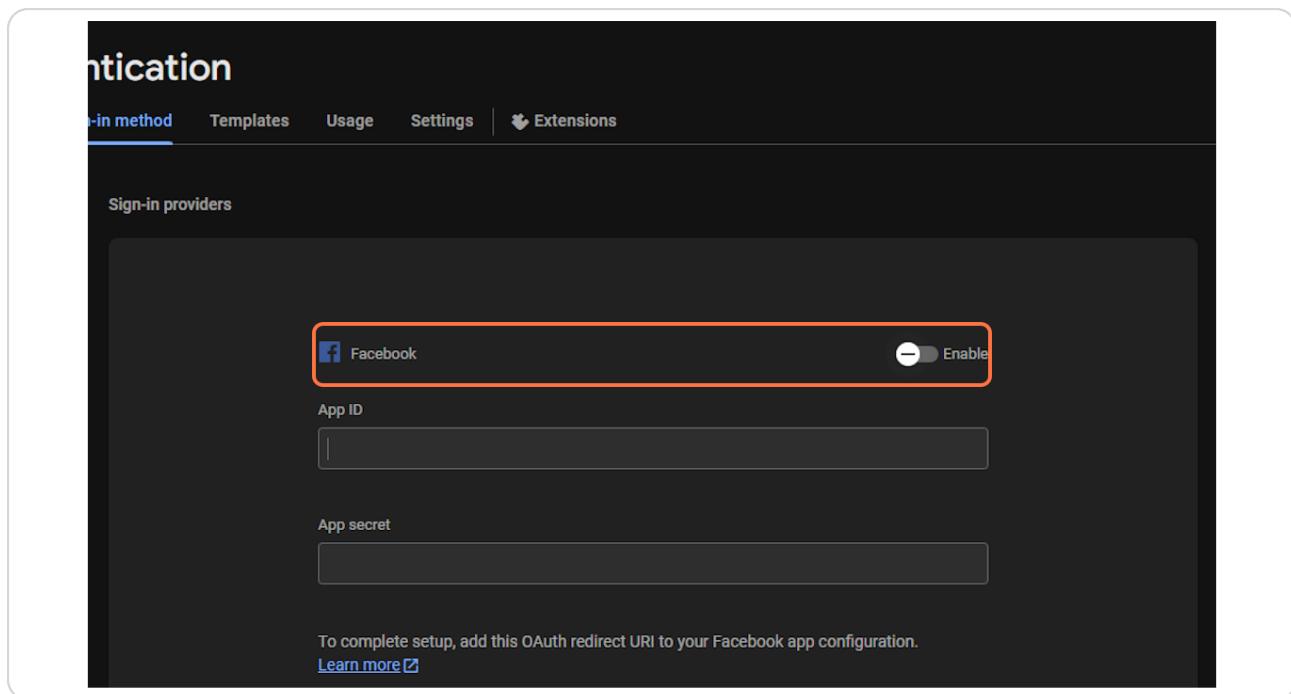
STEP 35

Copy input titled "App ID"

The screenshot shows the Facebook App Submission form. At the top, there's a red warning box with a white exclamation mark icon and the text 'Currently Ineligible for Submission' followed by 'Your submission is missing data in the following fields:'. A bulleted list includes 'App icon (1024 x 1024)', 'Privacy Policy URL', 'User data deletion', and 'Category'. Below this, there are several input fields: 'App ID' (with a value starting with '1' and a red border), 'App secret' (with a redacted value), 'Display name' ('Sample App'), 'Namespace' (empty), 'App domains' (empty), 'Contact email' ('chester.lok@hashcodestudio'), 'Privacy Policy URL' (with a placeholder 'Privacy policy for Login dialog and app details'), and 'Terms of Service URL' (with a placeholder 'Terms of Service for Login dia').

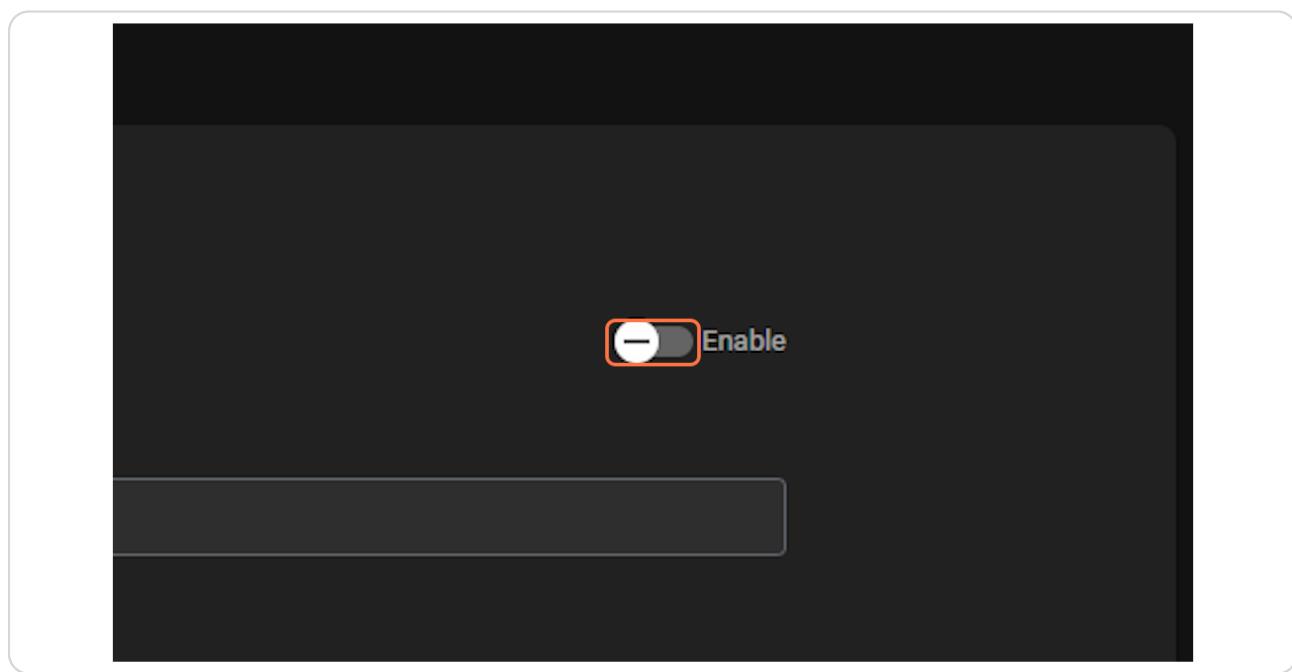
STEP 36

Click on Facebook...



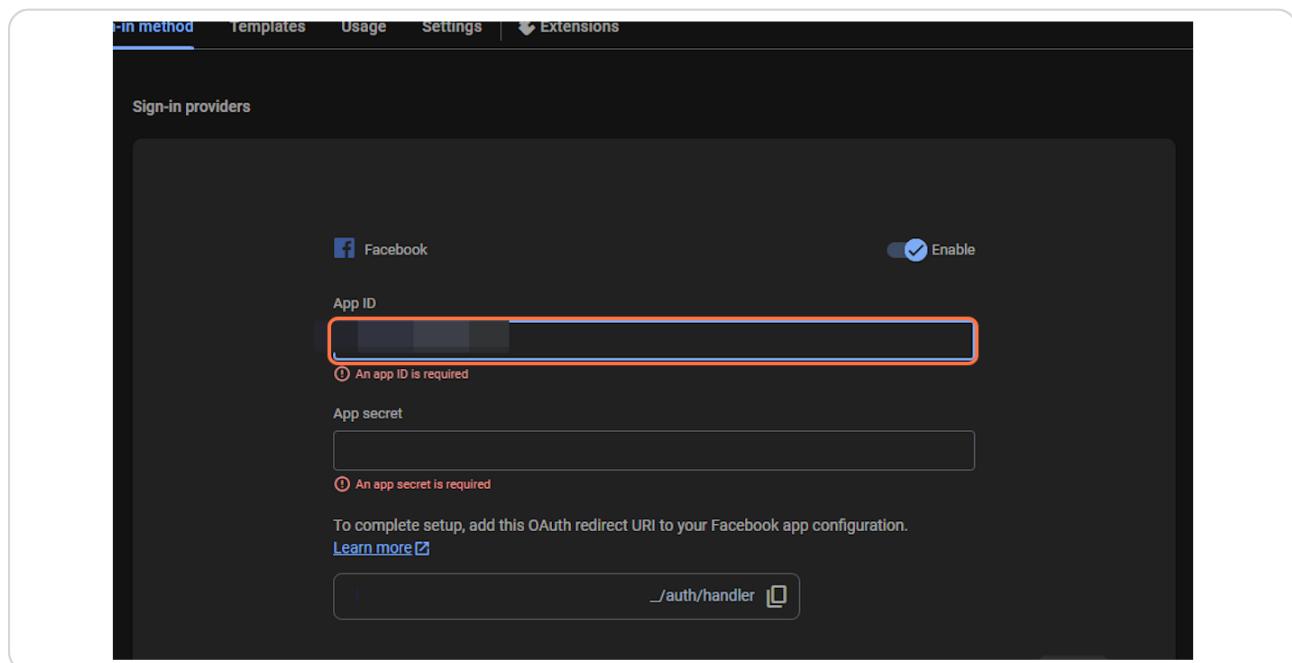
STEP 37

Click on Enable Facebook sign-in



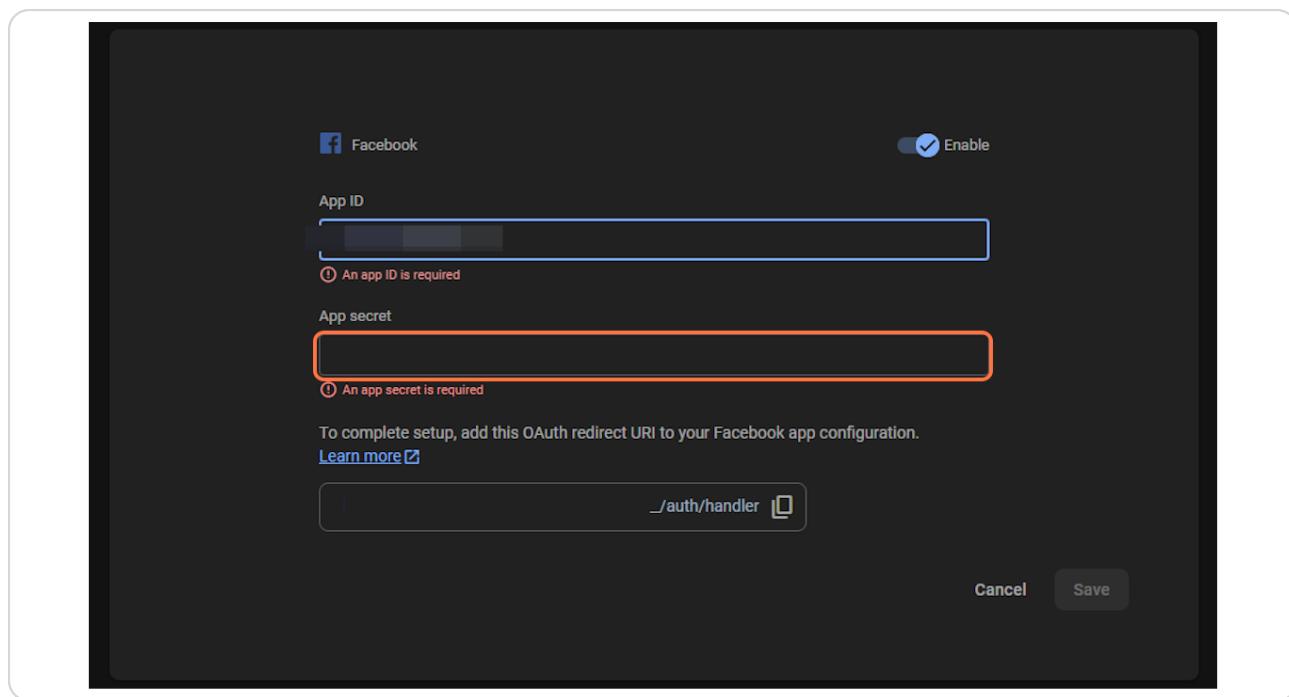
STEP 38

Paste "App ID" into input



STEP 39

Click on App secret



 Sample App - App settings - Meta for Develop... 2 Steps 

STEP 40

Click on App secret...

The screenshot shows a form with the following fields:

- Ids:** A large input field.
- App secret:** An input field containing a redacted password (represented by blue dots). This field is highlighted with an orange border.
- Show:** A button next to the App secret field.
- Namespace:** Two input fields.
- Contact email:** An input field containing "chester.lok@hashcodestudio.com".

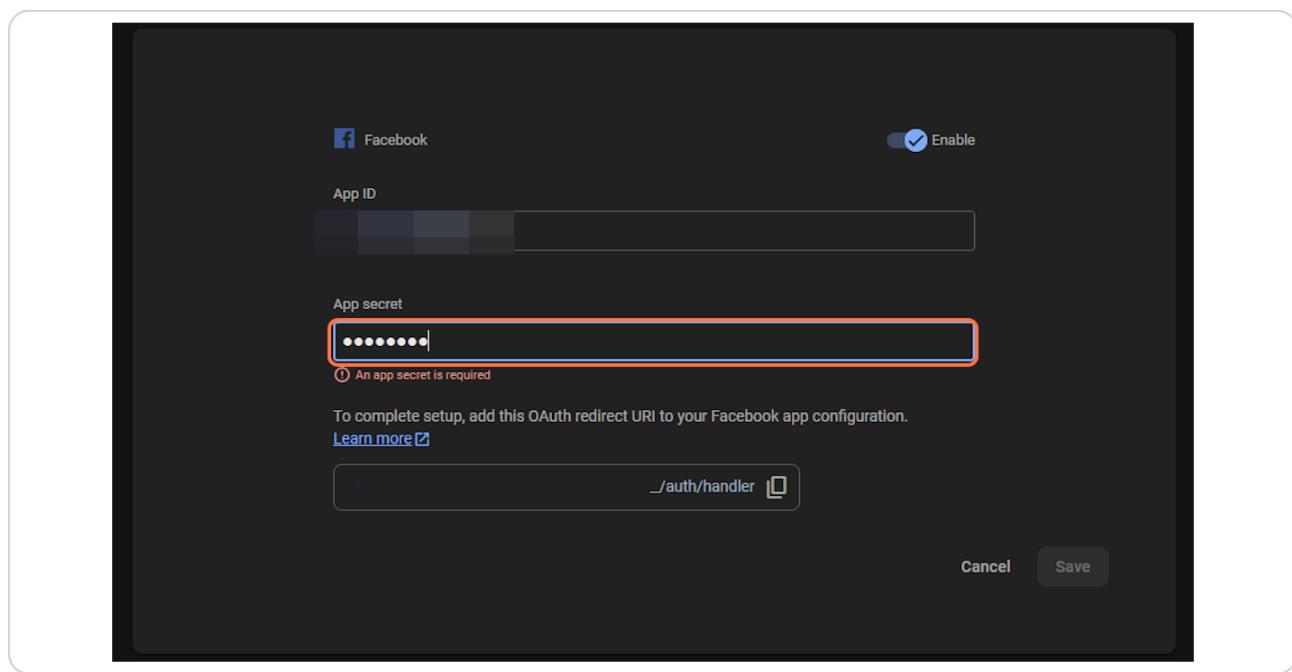
STEP 41

Click Show and Copy the App Secret

 **Test Game – Authentication – Sign-in method...** 18 Steps 

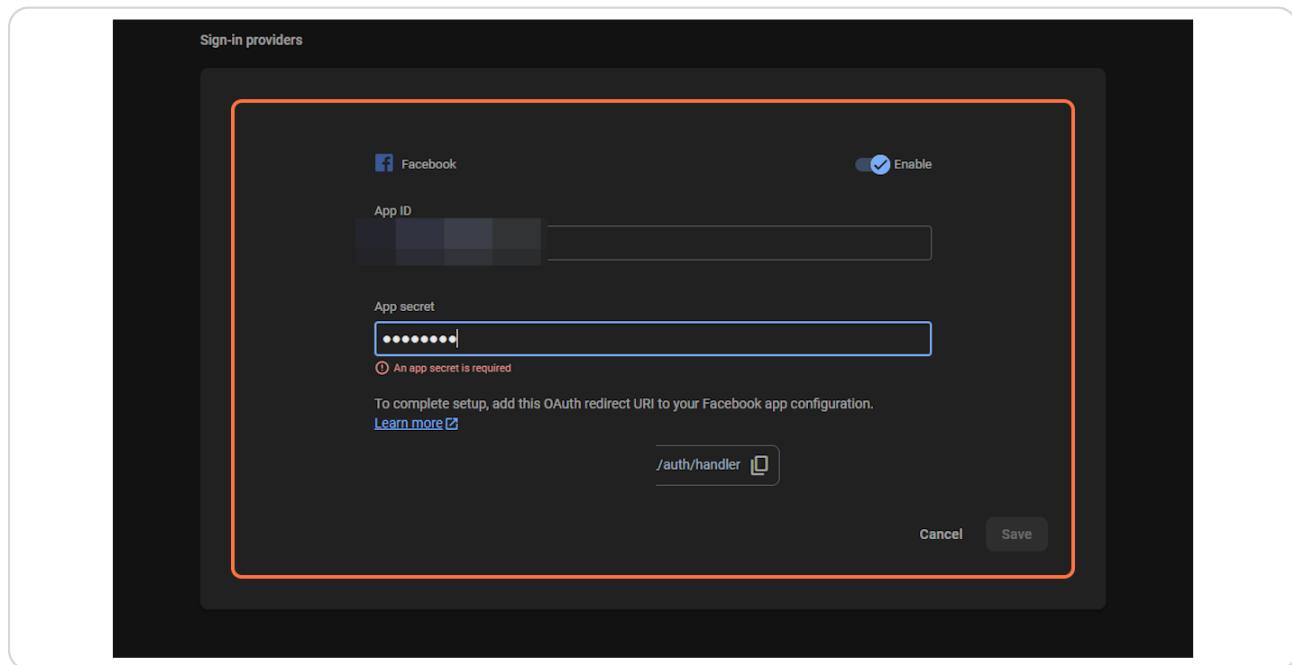
STEP 42

Paste "App secret" into input



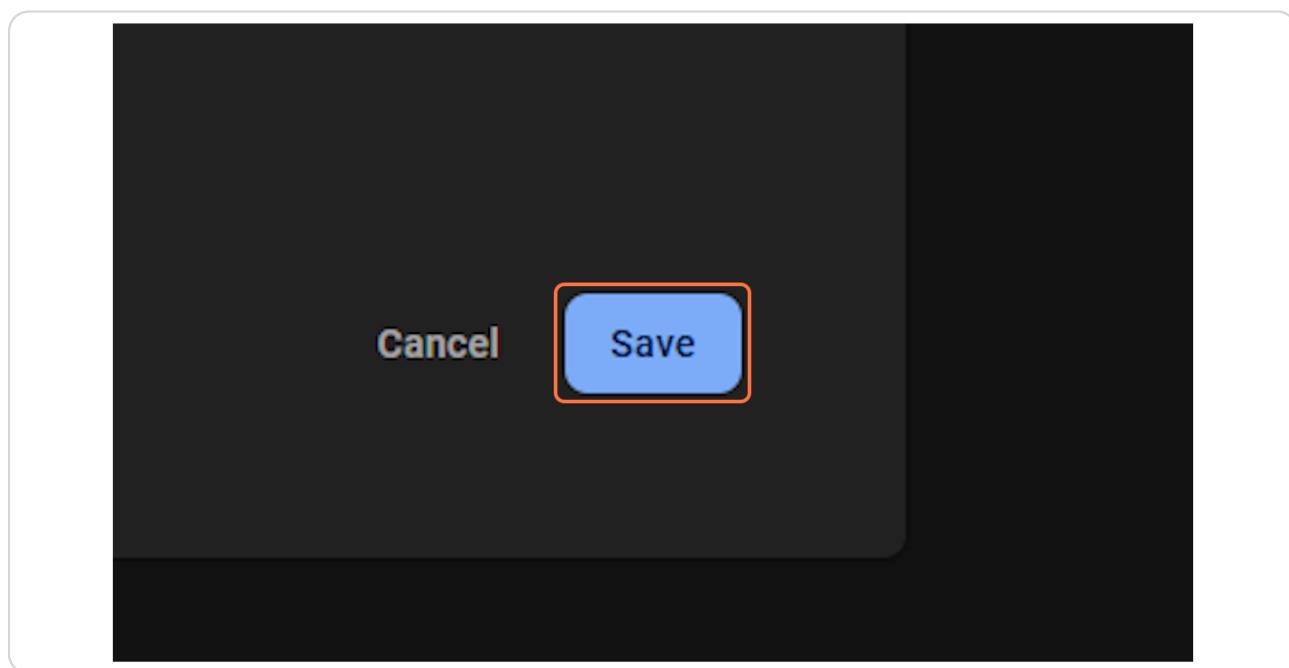
STEP 43

Click anywhere



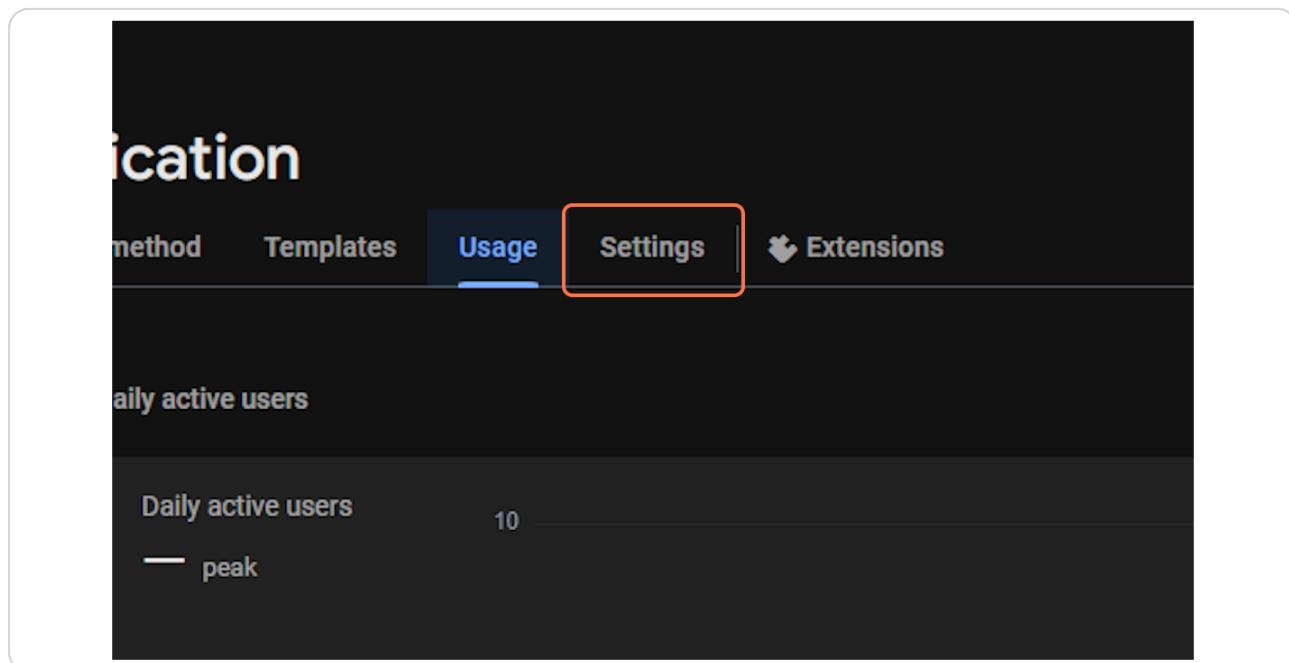
STEP 44

Click on Save



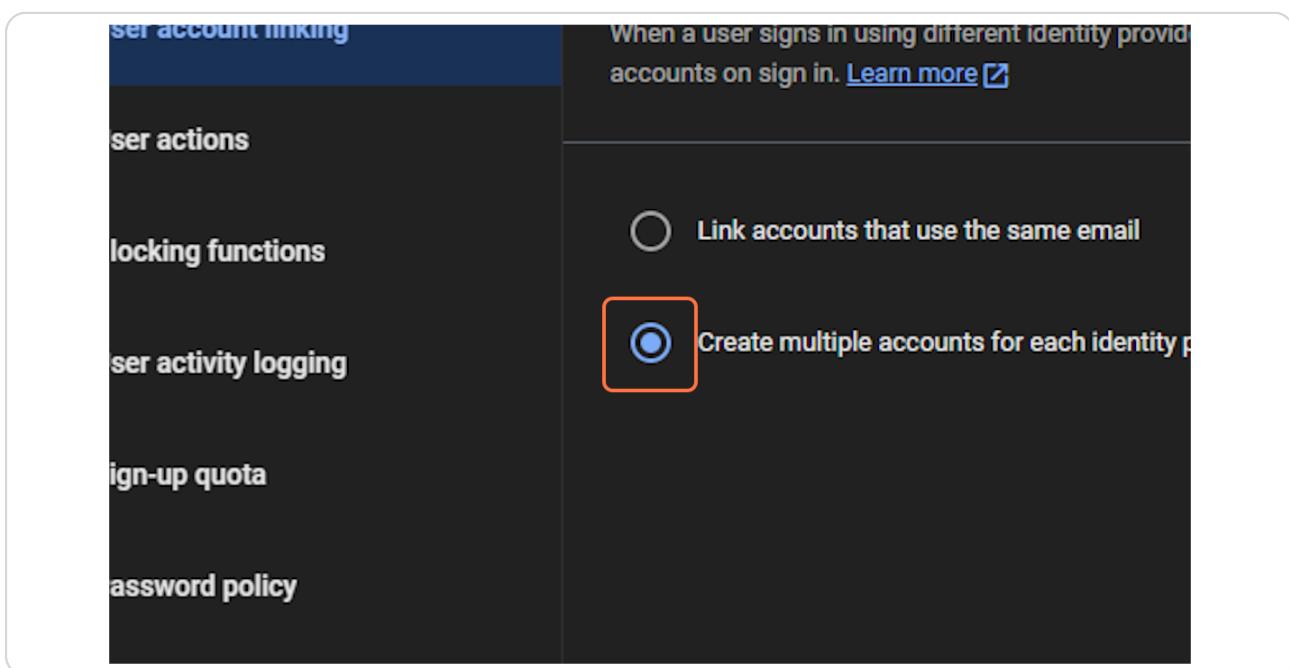
STEP 45

Click on Settings



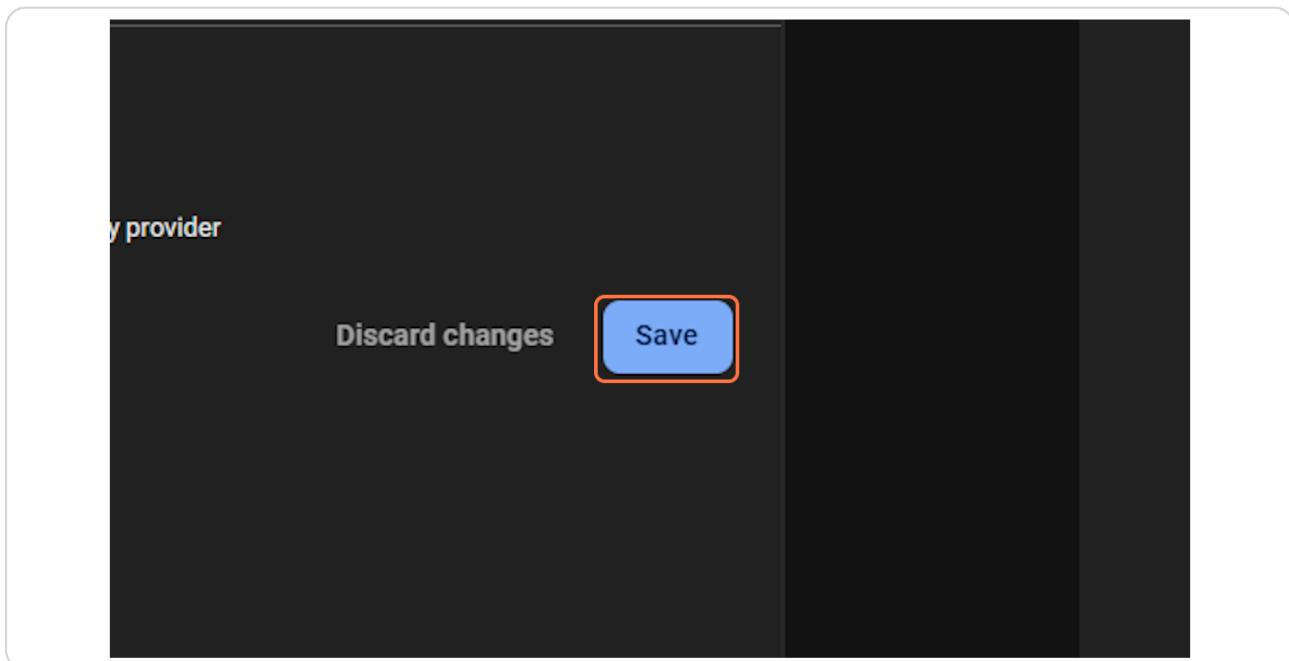
STEP 46

Select Create multiple accounts for each identity provider



STEP 47

Click on Save



STEP 48

Click on gmp_nav20_settings

The screenshot shows the Firebase console's Authentication interface. On the left, there's a sidebar with links like Project Overview, Generative AI, Build with Gemini, Genkit (NEW), Project shortcuts, Authentication (which is highlighted in blue), and App Check. The main area is titled 'Authentication' and has tabs for Users, Sign-in method (which is underlined in blue), and Templates. Below these tabs, there's a section for 'Sign-in providers' with a 'Provider' button.

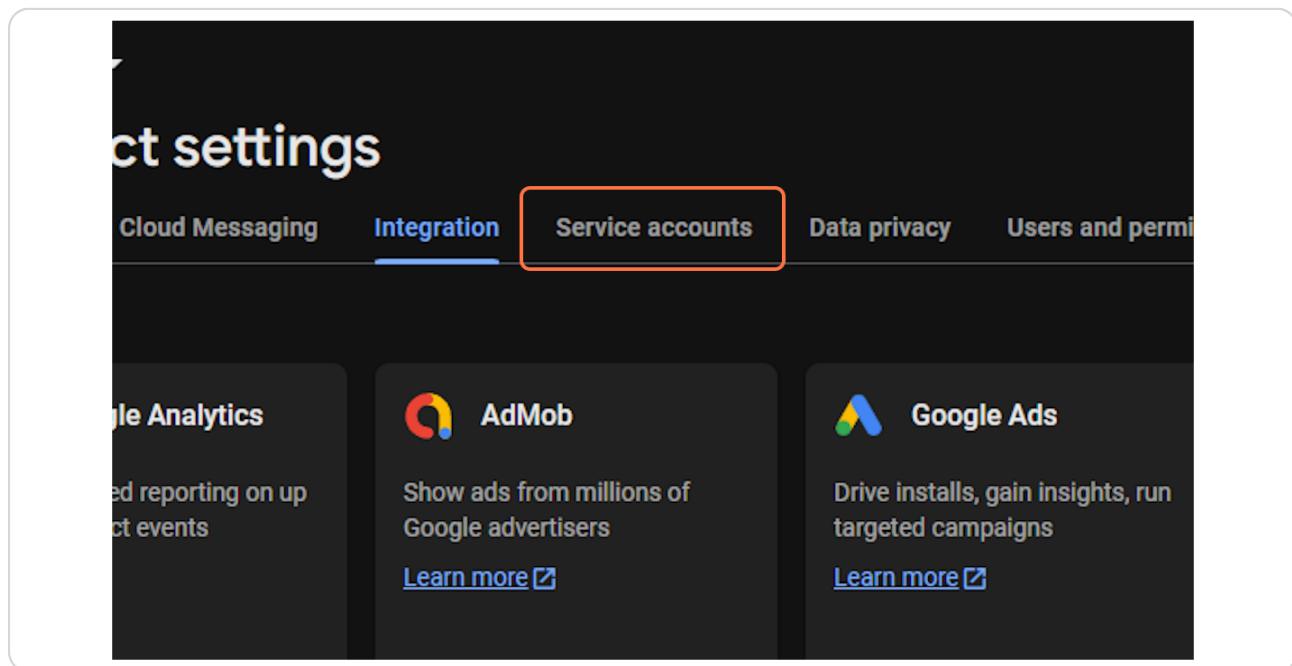
STEP 49

Click on Project settings

The screenshot shows the Firebase console's Project Overview screen. The sidebar includes Project Overview, Generative AI, Build with Gemini, Genkit (NEW), Project shortcuts, Authentication (highlighted in blue), App Check, and Product categories. A dropdown menu is open over the Authentication tab, showing options: Project settings (which is highlighted with a red box), Users and permissions, and Usage and billing. The main area displays the 'Authentication' section with tabs for Test Game, Users, Sign-in method, and Templates. Below these tabs, there's a 'Sign-in providers' section with a 'Provider' button and a 'Facebook' entry.

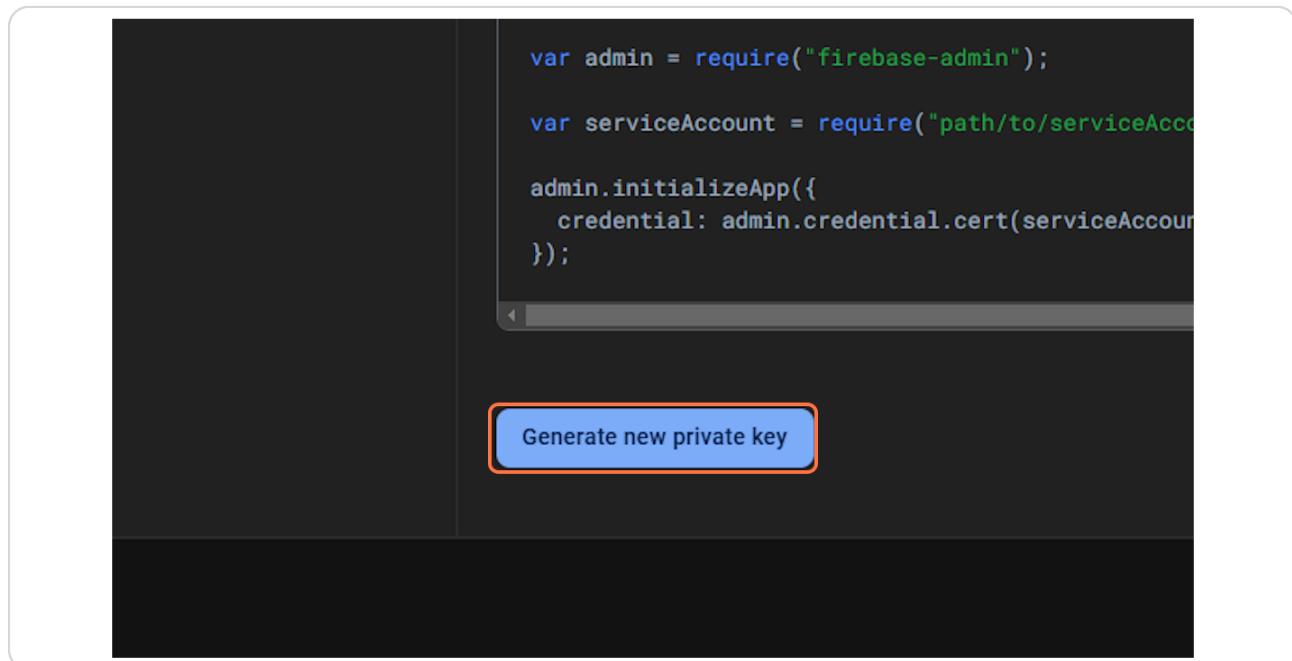
STEP 50

Click on Service accounts



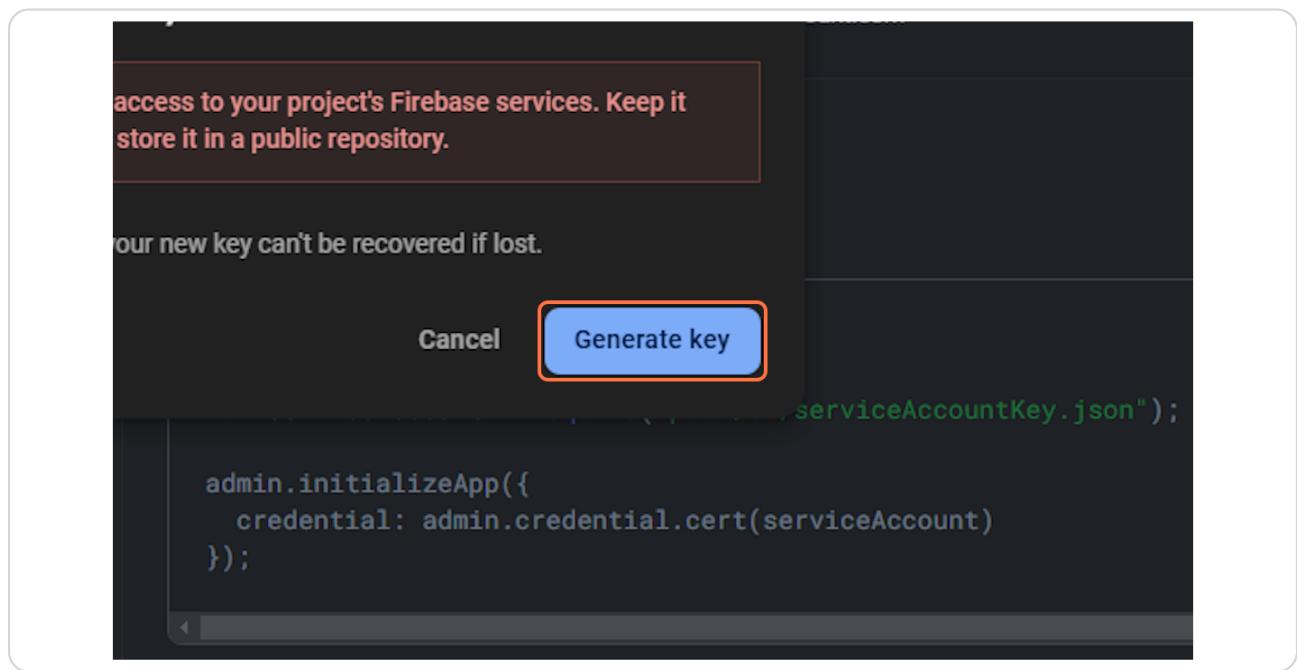
STEP 51

Click on Generate new private key



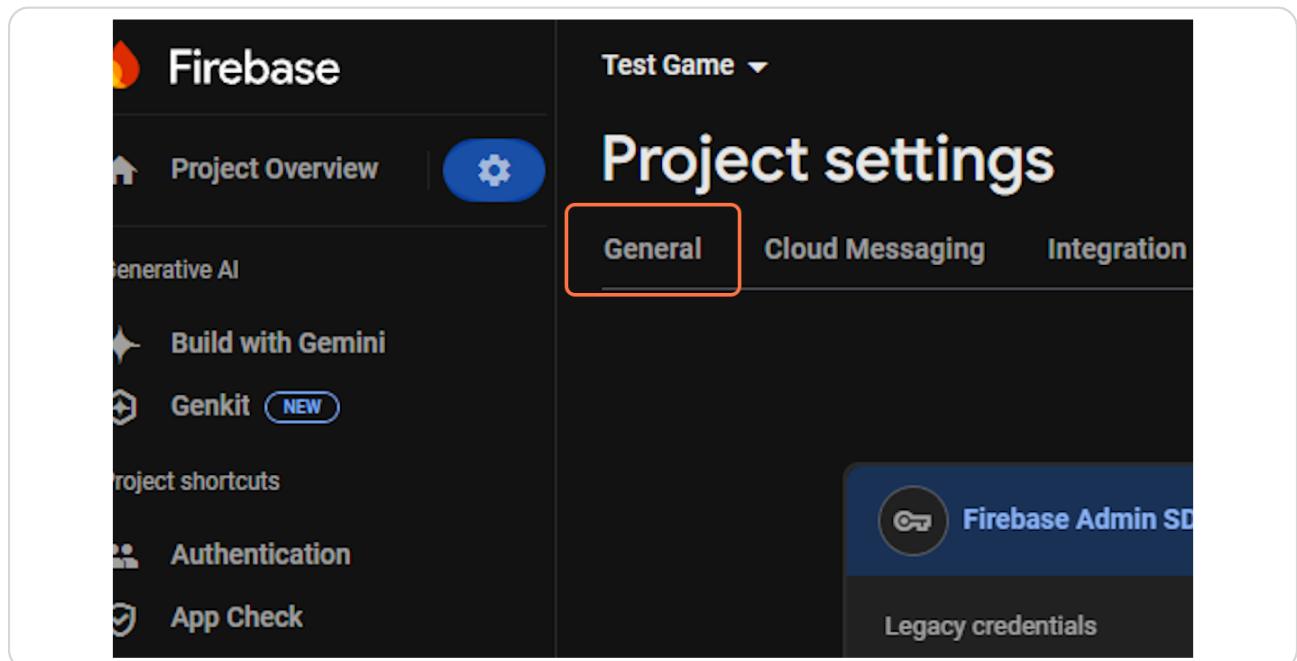
STEP 52

Click on Generate key and paste the json to the backend



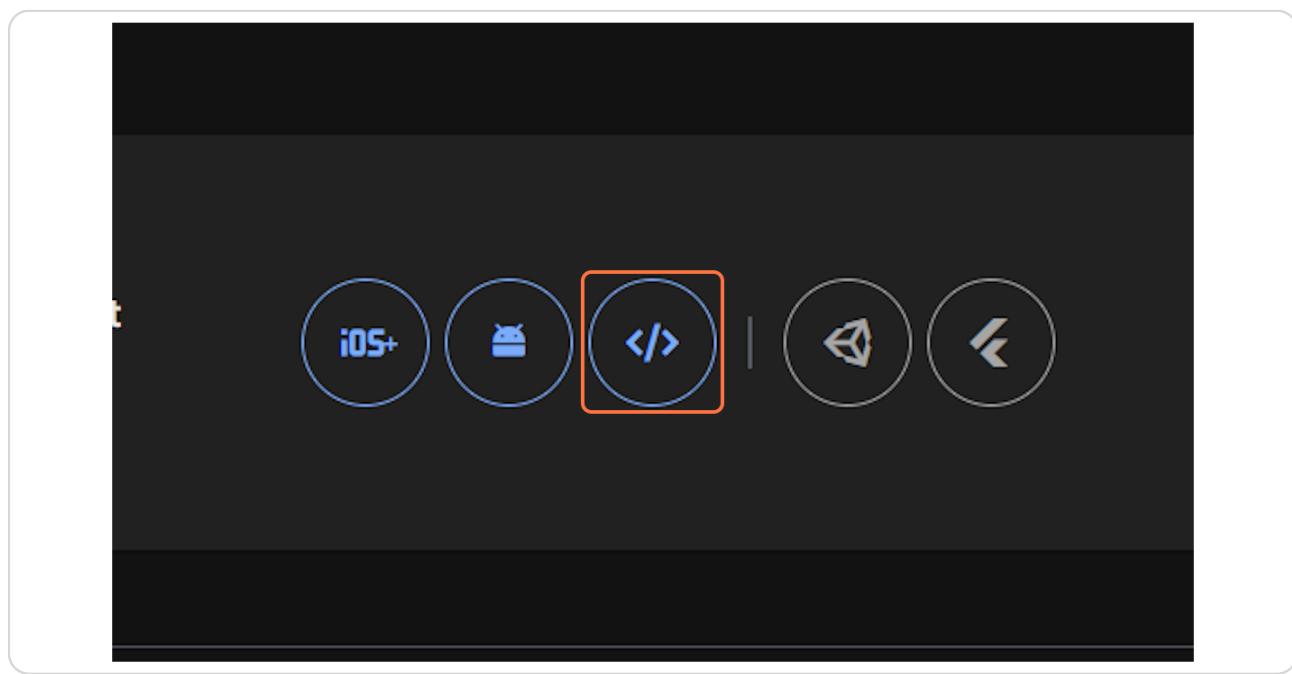
STEP 53

Click on General



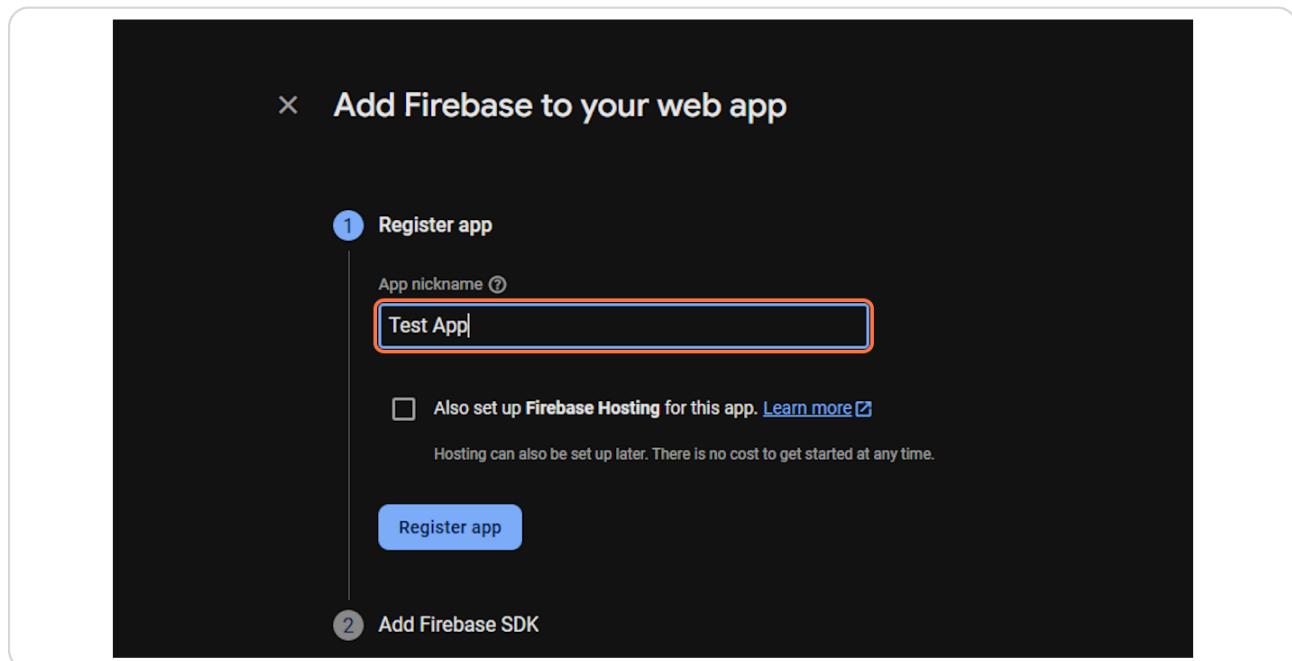
STEP 54

Click on plat_web



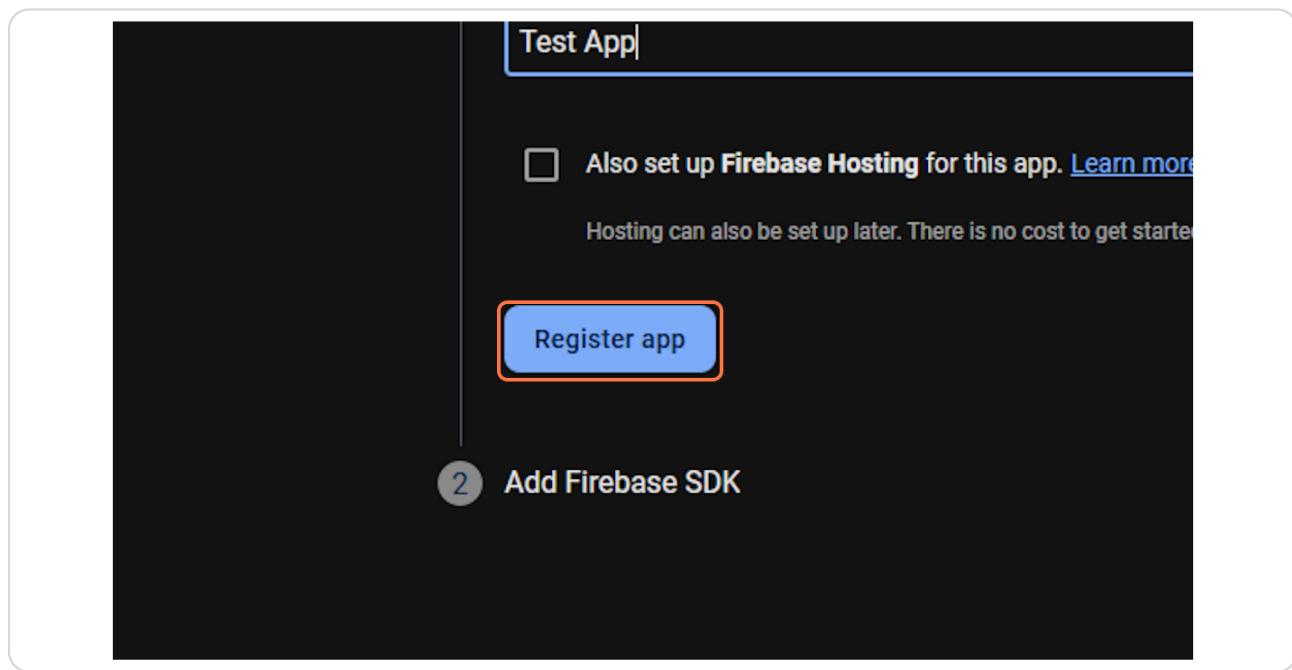
STEP 55

Type "Test App"



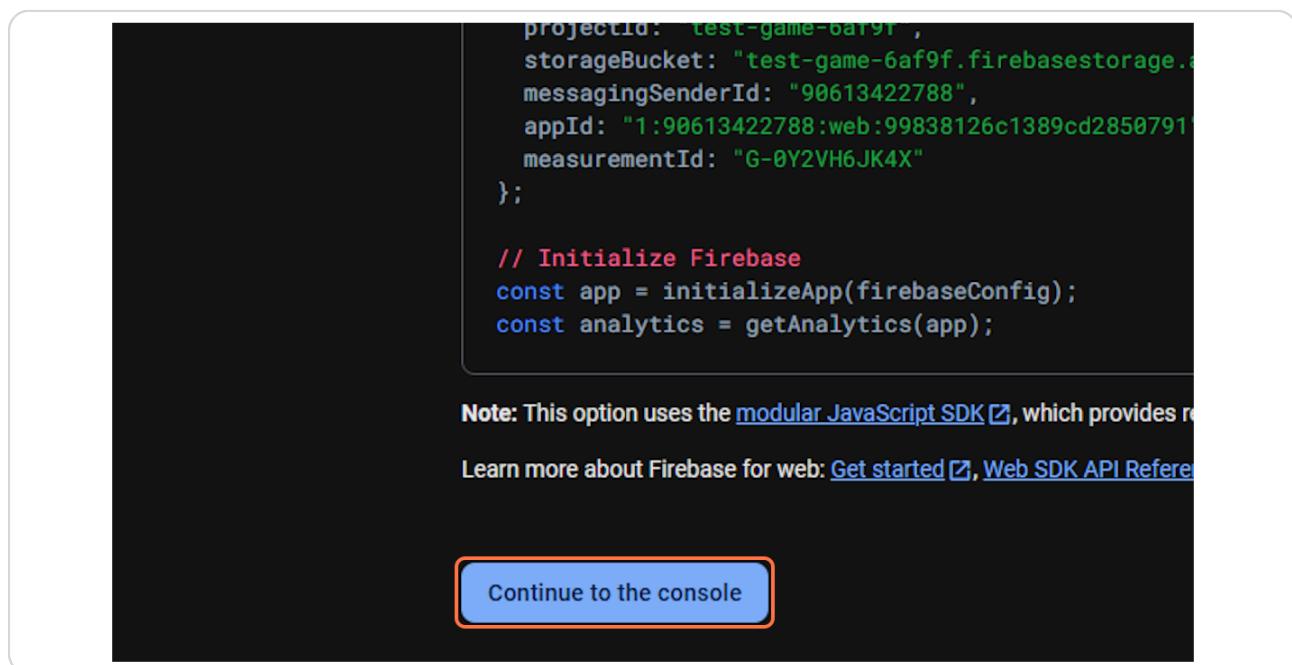
STEP 56

Click on Register app



STEP 57

Click on Continue to the console



STEP 58

Select Config

The screenshot shows a dark-themed interface for selecting an SDK setup method. At the top, there are three options: 'npm', 'CDN', and 'Config'. The 'Config' option is highlighted with a blue border and a blue circular icon. Below this, text instructions say 'Get the snippet for your app's Firebase config object.' followed by a link 'Learn more'. A code snippet is provided for the Firebase JS SDK v7.20.0 and later:

```
// For Firebase JS SDK v7.20.0 and later, measurement
const firebaseConfig = {
```

STEP 59

copy each of the fields and paste into .env at front-end

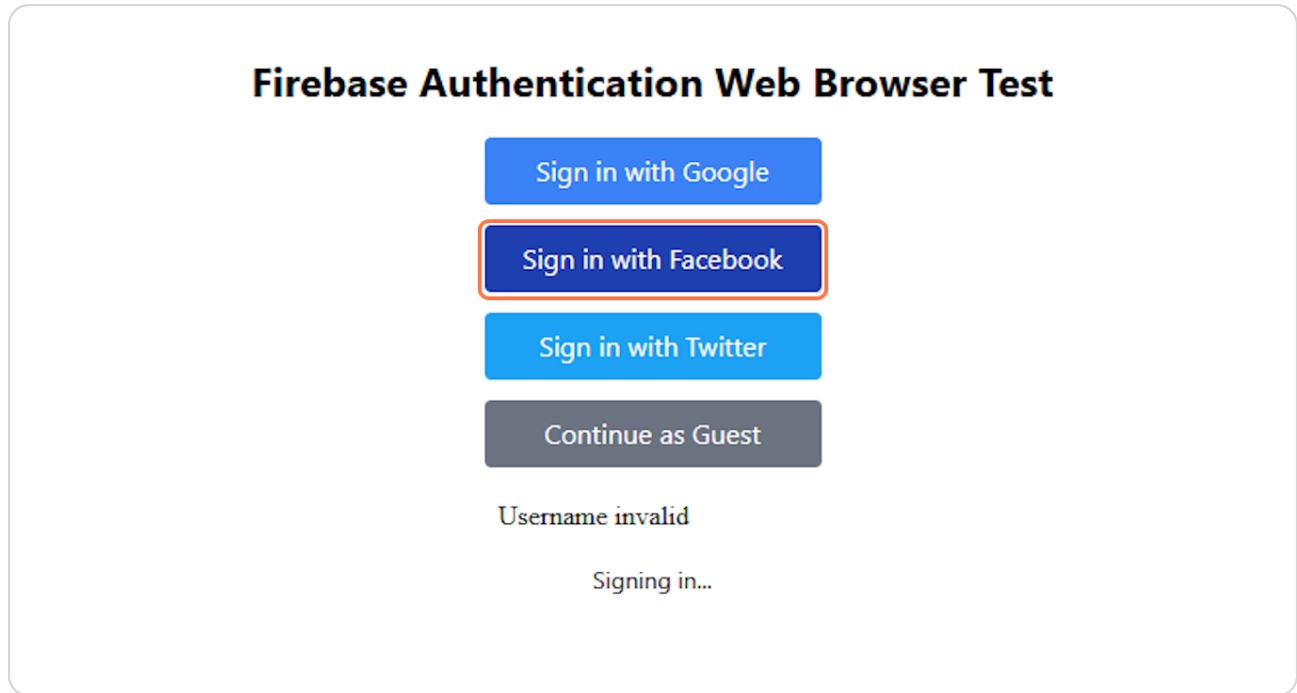
The screenshot shows a dark-themed page for the 'modular SDK'. It features a large button with a copy icon (a white 'C' inside a blue square) outlined in orange. Below the button, text reads 'Take a look at the [modular SDK](#)'. At the bottom of the page, there are links to 'API Reference' and 'Samples'.

Run front end and backend using npm run start

1 Step 

STEP 60

Click on Sign in with Facebook

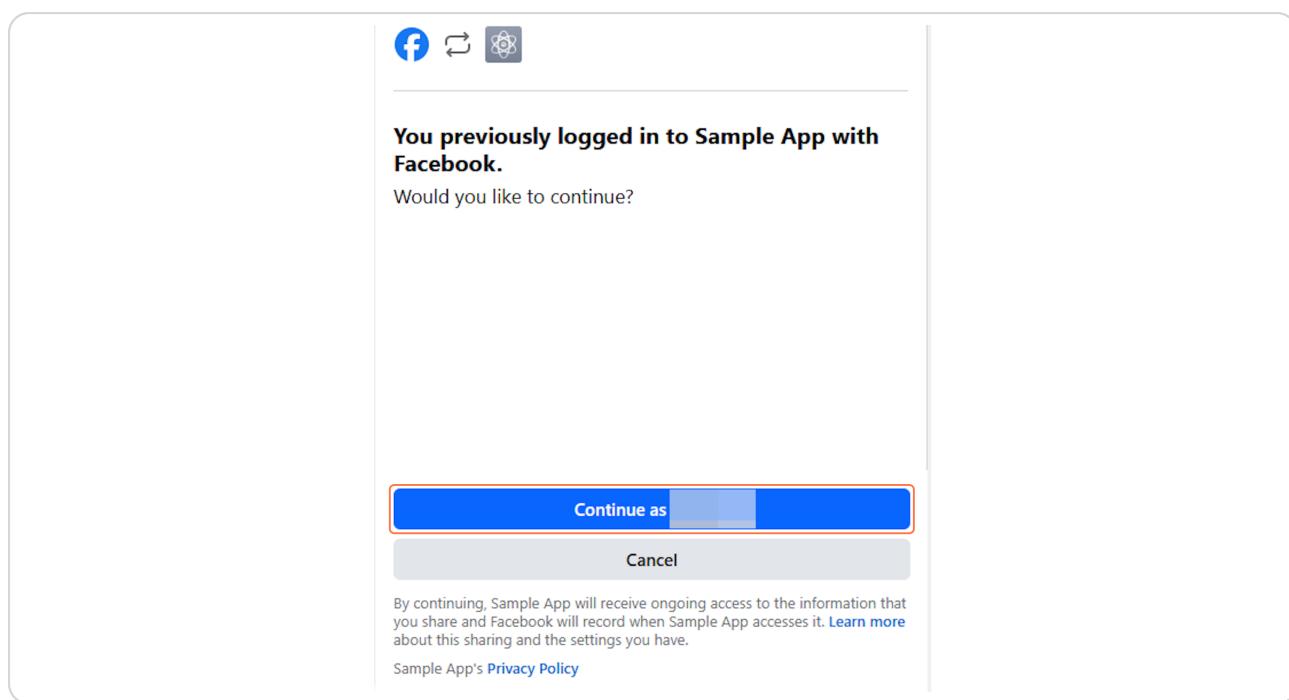


 Facebook

1 Step 

STEP 61

Click on Continue as *Your User Name



Tango

Never miss a step again. Visit Tango.us