

# UDB VIRTUAL

## EL SALVADOR

### UNIVERSIDAD DON BOSCO

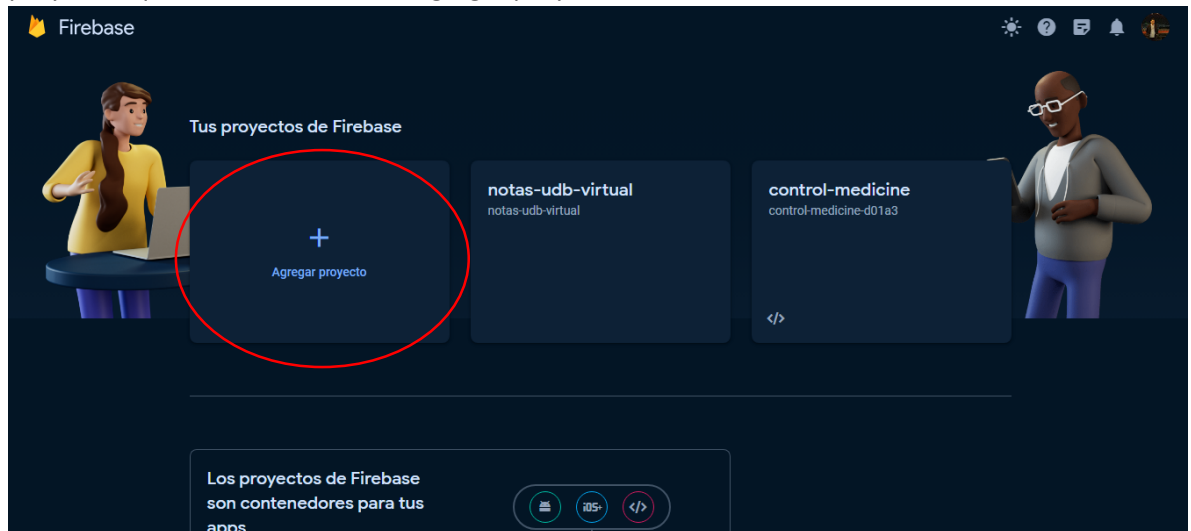
Alumnos: Campos Hernández, David Guillermo, CH220048

Artiga Mármol, Carlos Alfredo, AM221132

Materia: DPS

Tema: Foro2

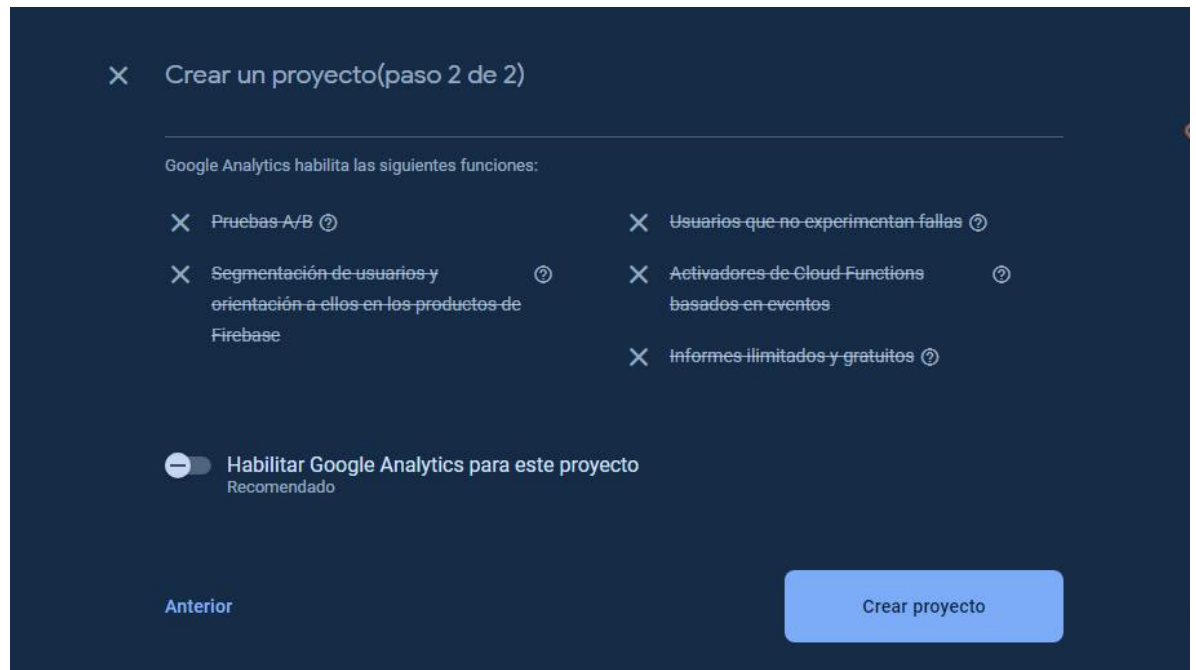
1. Primero hacemos login en el sitio de Firebase, y en la consola de inicio donde vemos lo proyectos que tenemos, daremos agregar proyecto



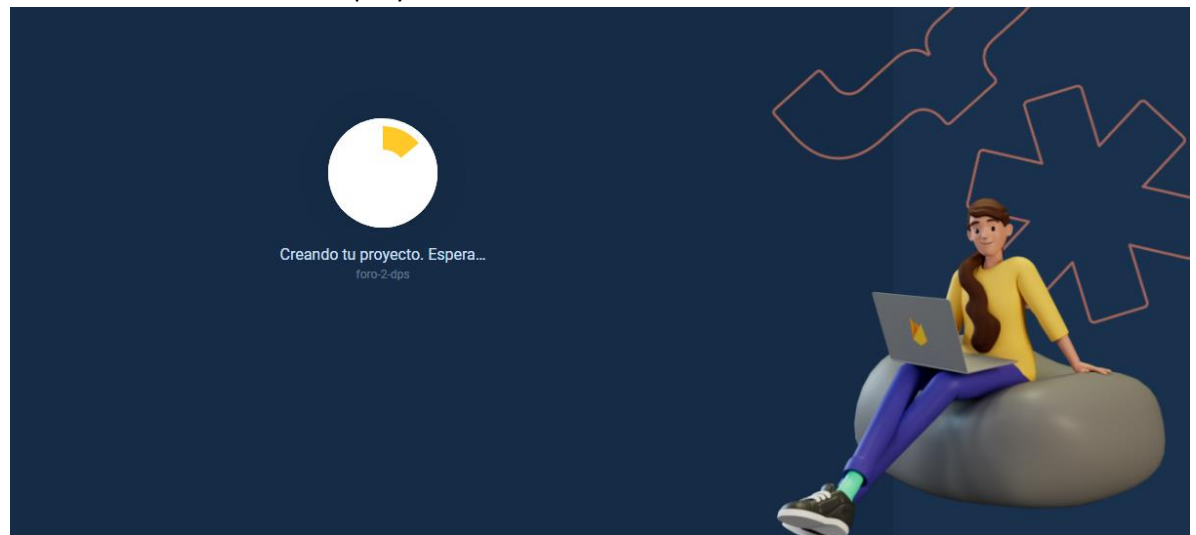
2. Le damos un nombre a nuestro proyecto



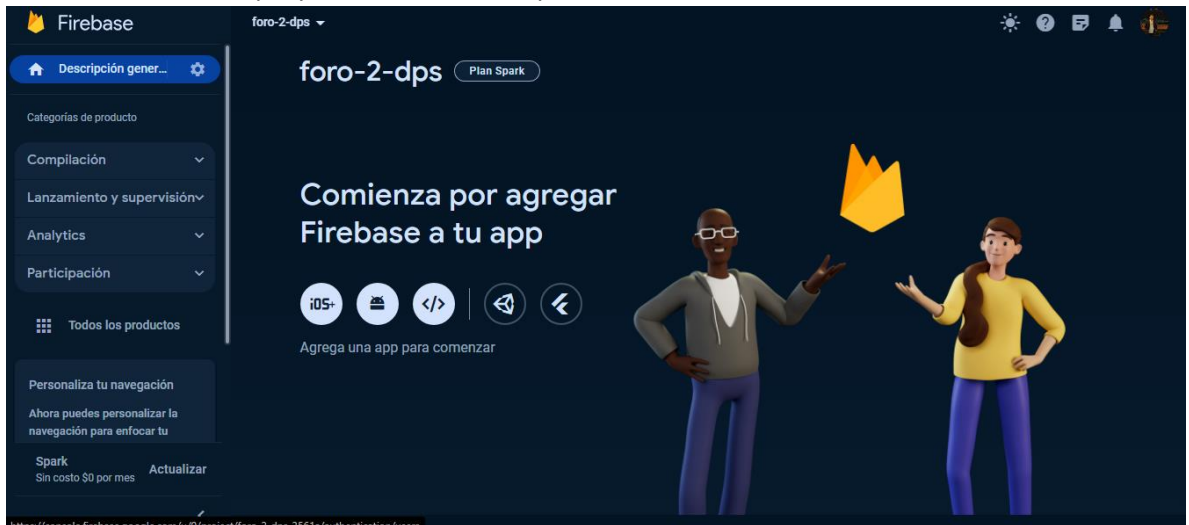
3. Nos preguntara si queremos activar google analytics, desactivamos ya que no necesitaremos analizar los datos de nuestros usuarios



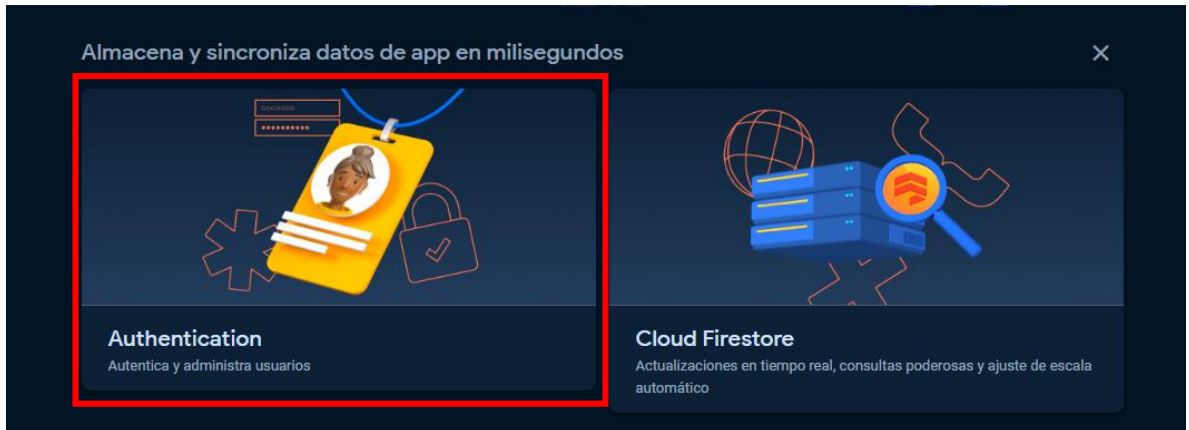
4. Procedemos a crear nuestro proyecto



5. Una vez creado este proyecto veremos esta pantalla

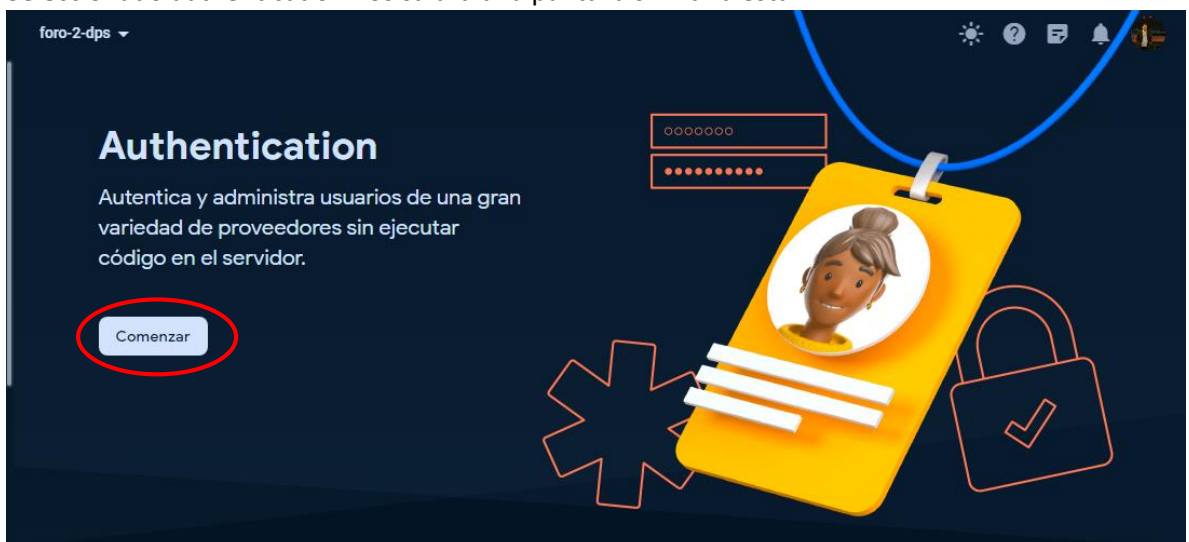


En la cual deberemos navegar un poco más para abajo hasta encontrar lo que usaremos en este día,



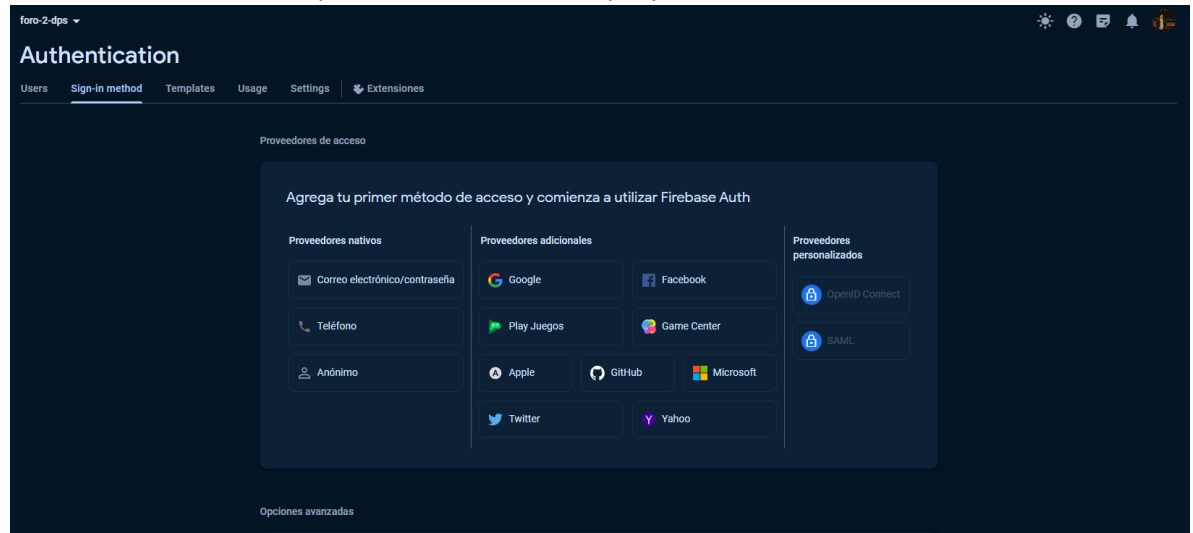
Authentication.

6. Seleccionado authentication nos saldrá una pantalla similar a esta



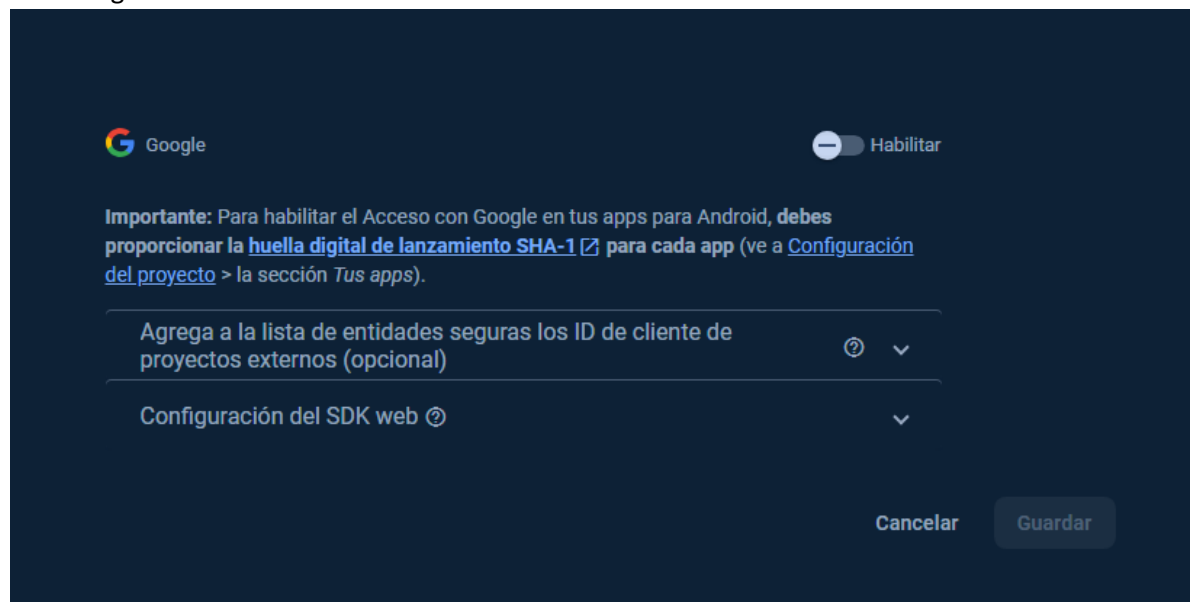
Donde seleccionaremos el botón de “Comenzar”

7. Cuando seleccionamos el botón de “Comenzar” nos dirigirá a una pantalla en donde nos muestran los diferentes tipos de autenticaciones que posee Firebase





Pero nosotros usaremos solo “Google” en esta ocasión.

8. Al seleccionar Google se nos despliega una nueva ventana con información sobre el acceso con Google




Aca deberemos habilitar la opción, y una vez habilitada veremos la configuración

 Google

 Habilitar

**Importante:** Para habilitar el Acceso con Google en tus apps para Android, debes proporcionar la [huella digital de lanzamiento SHA-1](#) para cada app (ve a [Configuración del proyecto](#) > la sección *Tus apps*).


 **Actualiza la siguiente configuración a nivel de proyecto para continuar**

Nombre público del proyecto ?

project-355026092135

Correo electrónico de asistencia del proyecto ?


No configurada


 Selecciona una dirección de correo electrónico.

Agrega a la lista de entidades seguras los ID de cliente de proyectos externos (opcional) ?


Configuración del SDK web ?

9. Ponemos nuestro correo electrónico

 Google

 Habilitar

**Importante:** Para habilitar el Acceso con Google en tus apps para Android, debes proporcionar la [huella digital de lanzamiento SHA-1](#) para cada app (ve a [Configuración del proyecto](#) > la sección *Tus apps*).

 **Actualiza la siguiente configuración a nivel de proyecto para continuar**

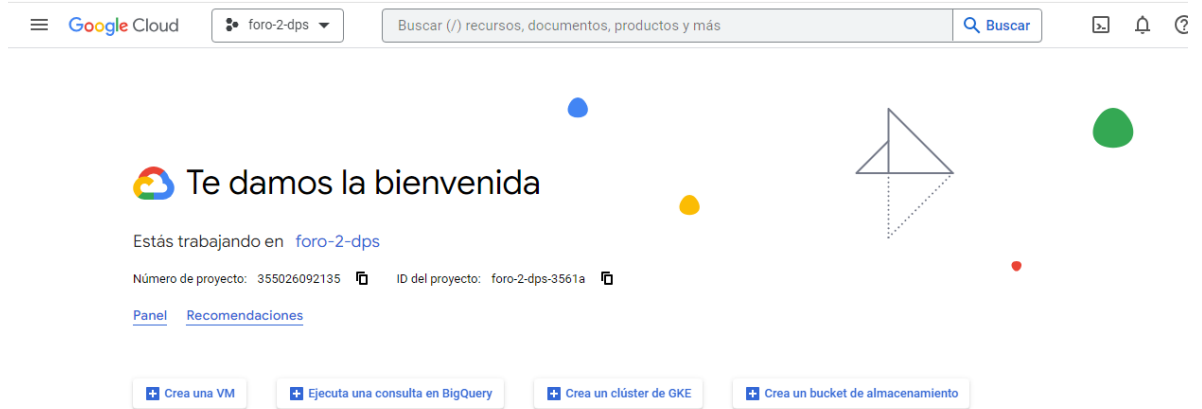
Nombre público del proyecto ?

project-355026092135

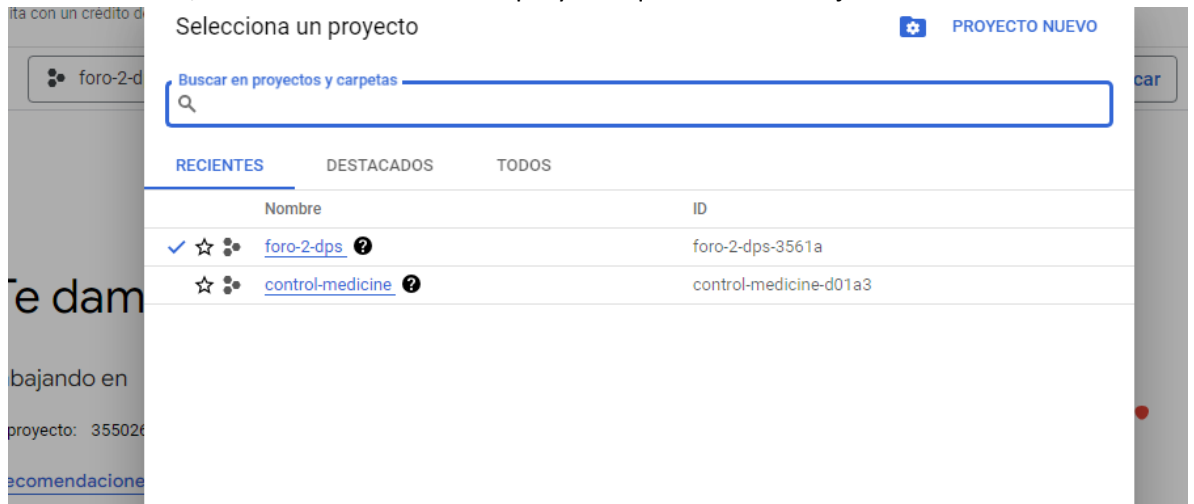
Correo electrónico de asistencia del proyecto ?

davidguillermo2004@gmail.com

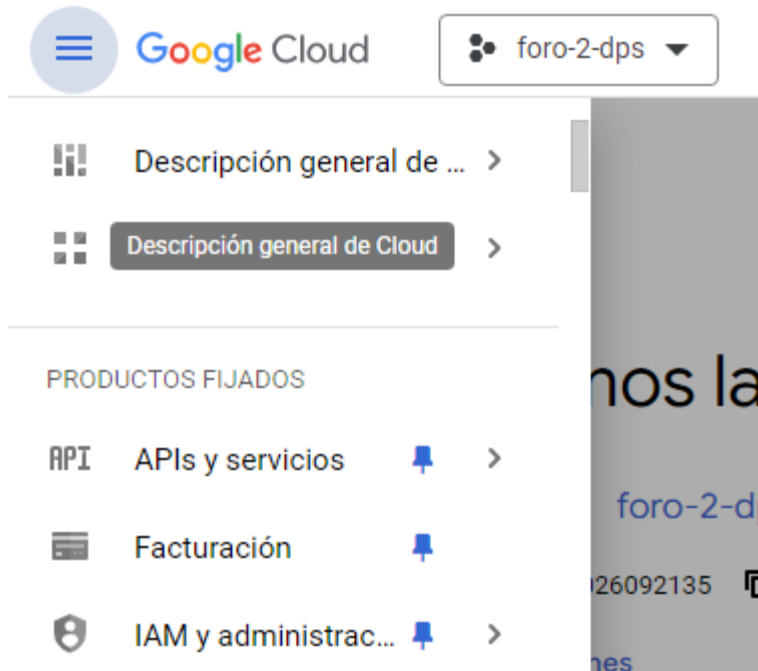
10. Luego vamos a Google Cloud, en donde deberemos entrar a la consola



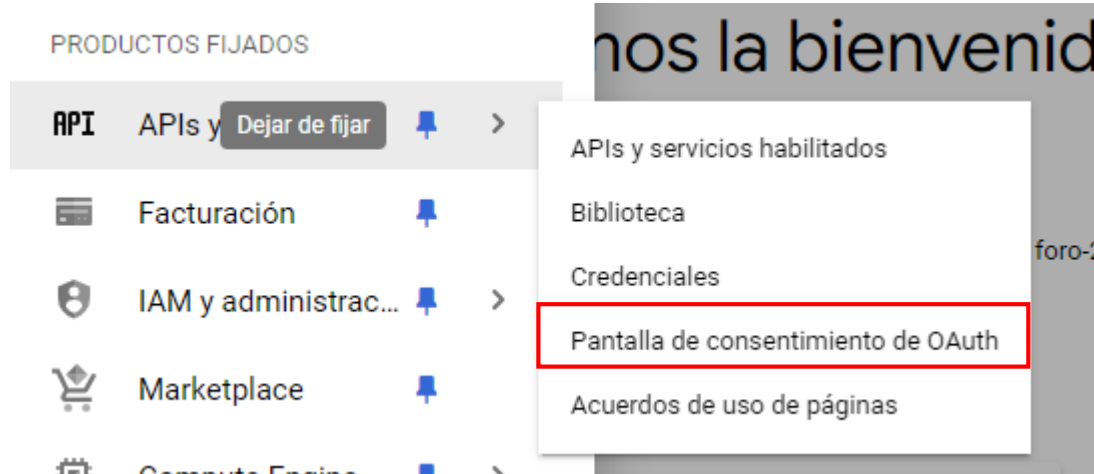
11. Una vez dentro, deberemos seleccionar el proyecto que estamos trabajando



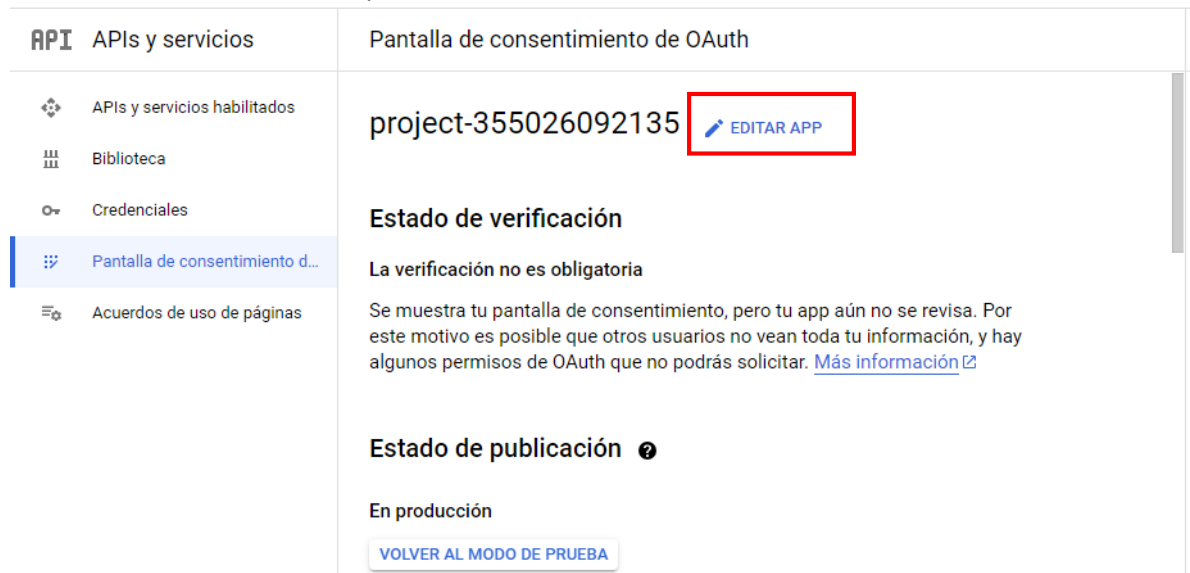
12. Luego de eso entramos al menú lateral



Seleccionamos APIs y servicios para poder seleccionar y configurar nuestra pantalla de consentimiento



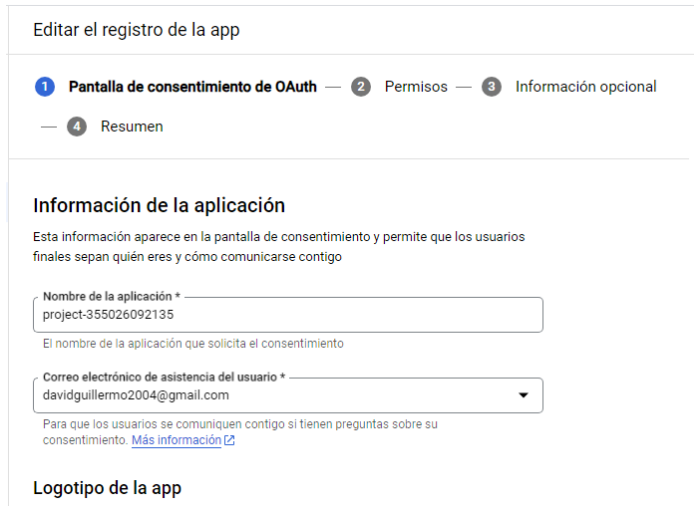
13. Una vez dentro veremos esta pantalla



En la cual deberemos seleccionar editar app



14. Como podremos observar en la siguiente imagen la pantalla de editar el registro de la app estará configurada en su mayoría



Editar el registro de la app

1 Pantalla de consentimiento de OAuth — 2 Permisos — 3 Información opcional — 4 Resumen

### Información de la aplicación

Esta información aparece en la pantalla de consentimiento y permite que los usuarios finales sepan quién eres y cómo comunicarse contigo

Nombre de la aplicación \*  
project-355026092135

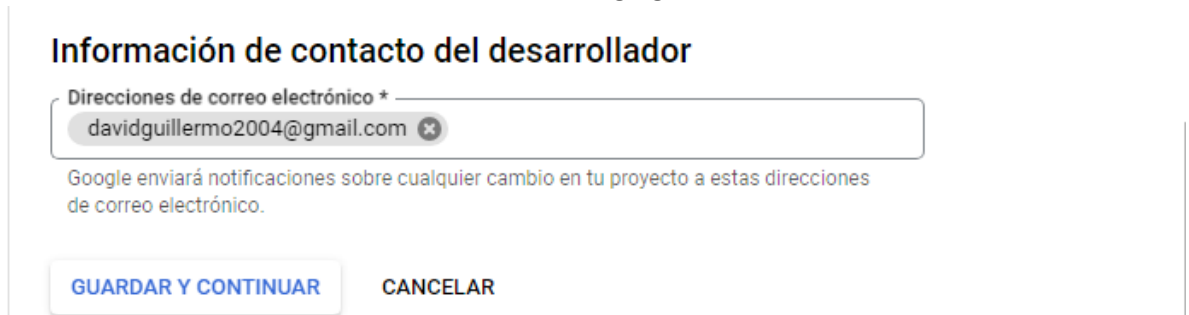
El nombre de la aplicación que solicita el consentimiento

Correo electrónico de asistencia del usuario \*  
davidguillermo2004@gmail.com

Para que los usuarios se comuniquen contigo si tienen preguntas sobre su consentimiento. [Más información](#)

Logotipo de la app

Entonces navegaremos hasta abajo hasta encontrar la sección de “Información de Contacto de desarrollador” en donde deberemos agregar nuestro correo



### Información de contacto del desarrollador

Direcciones de correo electrónico \*

davidguillermo2004@gmail.com x

Google enviará notificaciones sobre cualquier cambio en tu proyecto a estas direcciones de correo electrónico.

**GUARDAR Y CONTINUAR** CANCELAR

Todo esto es para así nosotros poder ver el correo, foto y la información del usuario que estará haciendo login con su Gmail de google. Damos en guardar y continuar para pasar al paso 2 de registro de app.

15. En la pantalla de permisos (paso 2)

## Editar el registro de la app

- ✓ Pantalla de consentimiento de OAuth — 2 **Permisos** — 3 Información opcional —  
4 Resumen

Los permisos representan lo que solicitas que los usuarios autoricen para la app y permiten que tu proyecto tenga acceso a tipos específicos de datos privados del usuario de sus Cuentas de Google. [Más información](#)

Si agregas permisos sensibles (🔒) o restringidos (🔒), como permisos para acceder a los contactos o correos electrónicos de los usuarios, tu app deberá pasar un proceso de verificación. [Más información](#)

AGREGAR O QUITAR PERMISOS

### Tus permisos no sensibles

Veremos que nos permitirá configurar que datos solicitar del usuario, por defecto trae solo para el correo, foto y su nombre, que es todo lo que necesitaremos, así que no configuraremos nada, damos guardar y continuar.

16. Luego en el paso 3 “información personal” no tocamos nada así que igual que el paso anterior avanzamos dando guardar y continuar

## Editar el registro de la app

- ✓ Pantalla de consentimiento de OAuth — ✓ Permisos — 3 **Información opcional** —  
4 Resumen

Puedes acelerar el proceso de verificación si proporcionas más detalles útiles sobre tu app a los revisores de Google.

### Información opcional

Comparte las direcciones de correo electrónico de cualquier contacto de Google que hayas tenido.

17. Luego veremos el Resumen de todo lo que hemos hecho, en el paso 4 “Resumen”, verificamos que aparezca nuestro correo configurado y damos guardar y cerrar

Editar el registro de la app

✓ Pantalla de consentimiento de OAuth — ✓ Permisos — ✓ Información opcional —

4 Resumen

---

**Pantalla de consentimiento de OAuth** [EDITAR](#)

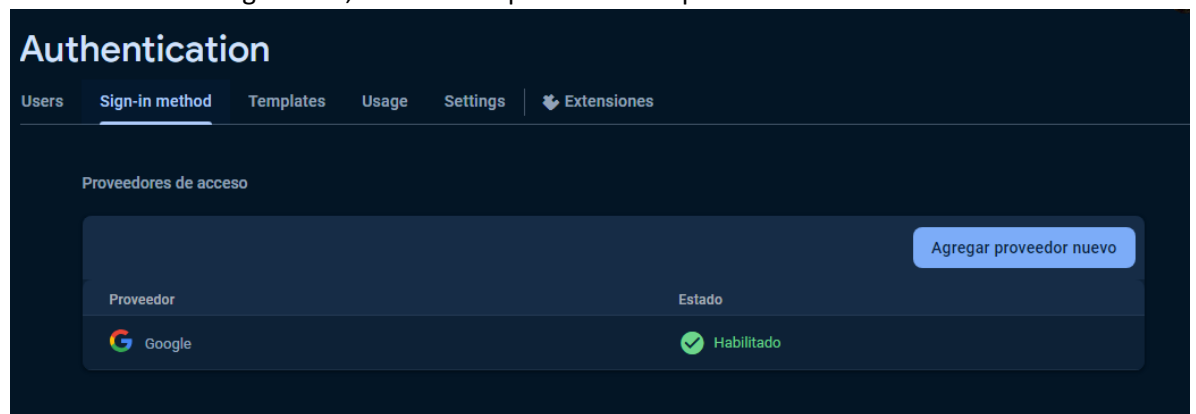
Tipo de usuario  
Externa

Nombre de la app  
project-355026092135

Correo electrónico de asistencia  
davidguillermo2004@gmail.com


Logotipo de la app  
No se proporcionó


18. Luego de eso volvemos a FireBase de nuevo, si todo se ha creado correctamente en FireBase no en GoogleCloud, deberíamos poder ver una pantalla así




Entonces presionamos en donde dice google ya que el paso siguiente es muy importante

**Importante:** Para habilitar el Acceso con Google en tus apps para Android, debes proporcionar la [huella digital de lanzamiento SHA-1](#) para cada app (ve a [Configuración del proyecto](#) > la sección *Tus apps*).



 Actualiza la siguiente configuración a nivel de proyecto para continuar



Nombre público del proyecto 

project-355026092135



Correo electrónico de asistencia del proyecto 

davidguillermo2004@gmail.com

Agrega a la lista de entidades seguras los ID de cliente de proyectos externos (opcional)  

Configuración del SDK web  

Al desplegarse esta pantalla deberemos configurar el SDK

Configuración del SDK web  

ID de cliente web

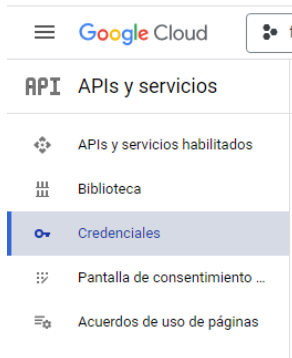
355026092135-uqc10cr1afeksvicrcjotoagias82vpl.apps.googleusercontent.com

Secreto de cliente web

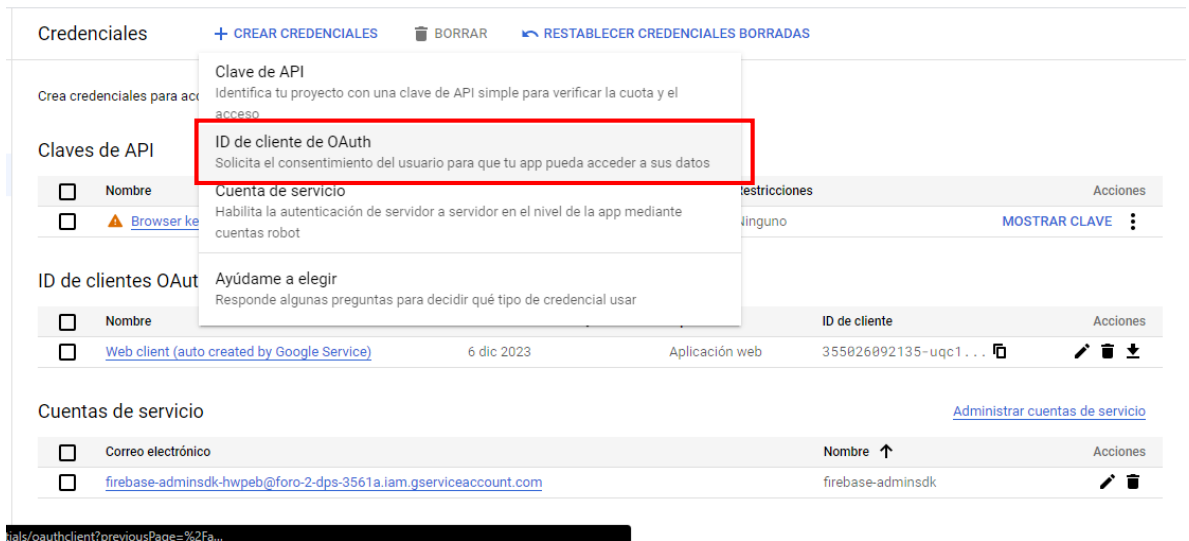
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

Entonces seleccionamos el ID del cliente web, esto es para conectar a nuestro proyecto de Firebase con google Cloud

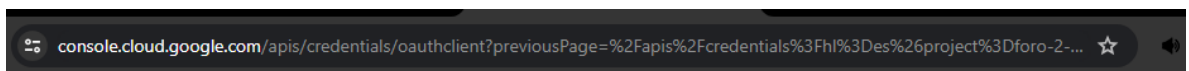
19. Entonces volvemos a Google Cloud y en la misma pestaña de APIs and Services, seleccionamos “Credenciales”



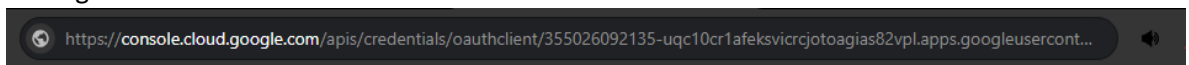
20. Entonces se nos desplegara una pantalla mas, en donde deberemos seleccionar la pestaña “+ Crear Credenciales” y seleccionamos ID de cliente de OAuth



21. Acá haremos algo muy curioso al dar clic e ID cliente de OAuth deberemos modificar nuestra URL usando el id de cliente web que nos trajimos de FireBase



Por algo similar a esto



Al dar enter habremos creado nuestro URI de FireBase valido, para poder manejar nuestra autenticación, así FireBase hará el request desde la aplicación con el siguiente URL

← ID de cliente para Aplicación web BORRAR

Para usar con solicitudes de un navegador

URI 1 \*

URI 2 \*

URI 3 \*

[+ AGREGAR URI](#)

URI de redireccionamiento autorizados ?

Para usar con solicitudes de un servidor web

URI 1 \*

Ya que, si no hacemos esto, al momento de hacer el request de autenticación nos dirá que no se encuentra el link o que no está permitido, entonces damos Guardar.

22. Y ahora necesitaremos generar un ID para nuestra app de Android (no haremos el de IOS) esto se hace desde nuestro código con nuestro IDE al momento de instalar las dependencias, entonces veremos en el app.json nos generara un package con una serie de números, usaremos eso:

```

},
"ios": {
  "supportsTablet": true
},
"android": {
  "adaptiveIcon": {
    "foregroundImage": "./assets/adaptive-icon.png",
    "backgroundColor": "#ffffff"
  },
  "package": "com.anonymous.LoginFireBase"
},
"web": {

```

23. Volvemos a nuestro Google Cloud, en APIs services, Credenciales y en crear credencial

Google Cloud foro-2-dps  Buscar

APIs y servicios

APIs y servicios habilitados

Biblioteca

**Credenciales**

Pantalla de consentimiento ...

Acuerdos de uso de páginas

Credenciales + CREAR CREDENCIALES BORRAR RESTABLECER CREDENCIALES BORRADAS

Crea credenciales para acceder a los recursos de la API

Claves de API

☐ Nombre

☐ Browser key

ID de clientes OAuth

☐ Nombre

☐ Cliente de Android

Clave de API

Identifica tu proyecto con una clave de API simple para verificar la cuota y el acceso

ID de cliente de OAuth

Solicita el consentimiento del usuario para que tu app pueda acceder a sus datos

Cuenta de servicio

Habilita la autenticación de servidor a servidor en el nivel de la app mediante cuentas robot

Ayúdame a elegir

Responde algunas preguntas para decidir qué tipo de credencial usar

ID de cliente

355826092135-fj8

Seleccionamos ID de cliente de nuevo,

← Crear ID de cliente de OAuth

Un ID de cliente se usa con el fin de identificar una sola app para los servidores de OAuth de Google. Si la app se ejecuta en varias plataformas, cada una necesitará su propio ID de cliente. Consulta [Configura OAuth 2.0](#) para obtener más información. [Obtén más información](#) sobre los tipos de clientes de OAuth.

Tipo de aplicación \*

- Aplicación web
- Android
- Extensión de Chrome
- IOS
- TVs y dispositivos de entrada limitada
- App de escritorio
- Plataforma universal de Windows (UWP)

Luego veremos esta pantalla, como haremos Android seleccionamos Android,

← Crear ID de cliente de OAuth

Un ID de cliente se usa con el fin de identificar una sola app para los servidores de OAuth de Google. Si la app se ejecuta en varias plataformas, cada una necesitará su propio ID de cliente. Consulta [Configura OAuth 2.0](#) para obtener más información. [Obtén más información](#) sobre los tipos de clientes de OAuth.

Tipo de aplicación \*

Android

Nombre \*

Cliente de Android 2

El nombre de tu cliente de OAuth 2.0. Este nombre solo se usa para identificar al cliente en la consola y no se mostrará a los usuarios finales.

Nombre del paquete \*

En tu archivo AndroidManifest.xml

Huella digital del certificado SHA-1 \*

El certificado de firma SHA-1 con huella digital restringe el uso de tus aplicaciones para Android. [Más información](#)

Recuerda el package que copiamos de nuestro archivo app.json, pues ese ira acá, luego para el certificado de Huella digital, usaremos nuestra cuenta de Expo, ejecutando “eas build:configure” (debemos tener cuenta previamente en eas, es decir expo app service) luego de eso configuraremos el tipo de app que vamos a construir (usando “eas credentials -p android”), en este caso es produccction, ya que queremos que funcione

```
→ navigation-test git:(main) ✗ eas credentials -p android

★ eas-cli@3.13.2 is now available.
To upgrade, run npm install -g eas-cli.
Proceeding with outdated version.

? Which build profile do you want to configure? > - Use arrow-keys.
Return to submit.
  development
  preview
> production
```

Key Alias	846fd27ebd34bdee2a48019fec30b54b
MD5 Fingerprint	53:D0:DA:F7:35:2E:D4:46:71:F0:8C:57:1F:1F:17:60
SHA1 Fingerprint	01:98:20:0B:71:CE:28:57:5D:3C:AA:24:22:D7:EA:93
:BD:D5:C4:9C	
SHA256 Fingerprint	B2:4B:42:8F:72:FA:90:93:30:54:1A:2F:90:CB:2F:EB
:16:6A:1E:4C:82:9B:0D:26:56:6E:58:4D:83:EF:DD:0A	
Updated	2 hours ago

Entonces obtenemos nuestra huella1 y la colocamos en nuestro espacio de google console

Nombre \*  
Cliente de Android 1

El nombre de tu cliente de OAuth 2.0. Este nombre solo se usa para identificar al cliente en la consola y no se mostrará a los usuarios finales.

Nombre del paquete \*  
com.anonymous.LoginFireBase

En tu archivo AndroidManifest.xml

Huella digital del certificado SHA-1 \*  
17:71:F5:0C:C7:76:42:07:B2:79:BE:F1:1F:D4:4A:4D:CC:FB:AE:25

El certificado de firma SHA-1 con huella digital restringe el uso de tus aplicaciones para Android. [Más información](#)

Use this command to get the fingerprint.


```
$ keytool -keystore path-to-debug-or-production-keystore -list
```


Una vez listo, guardamos y cerramos, ahora deberemos volver a nuestro FireBase Console



Google Habilitar


Importante: Para habilitar el Acceso con Google en tus apps para Android, debes proporcionar la [huella digital de lanzamiento SHA-1](#) para cada app (ve a [Configuración del proyecto](#) > la sección [Tus apps](#)).

Actualiza la siguiente configuración a nivel de proyecto para continuar

Nombre público del proyecto   
project-355026092135

Correo electrónico de asistencia del proyecto   
davidguillermo2004@gmail.com

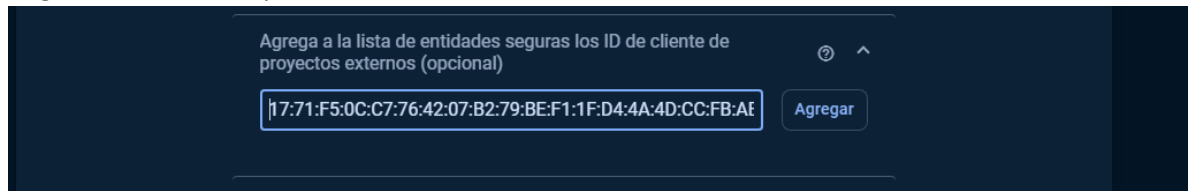
Agrega a la lista de entidades seguras los ID de cliente de proyectos externos (opcional)  

Configuración del SDK web 

Ya que ahora trabajaremos esta parte

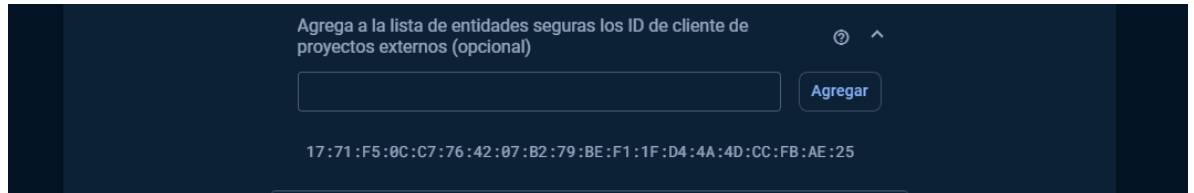


24. Pegamos nuestra ID que obtuvimos de Android en EAS



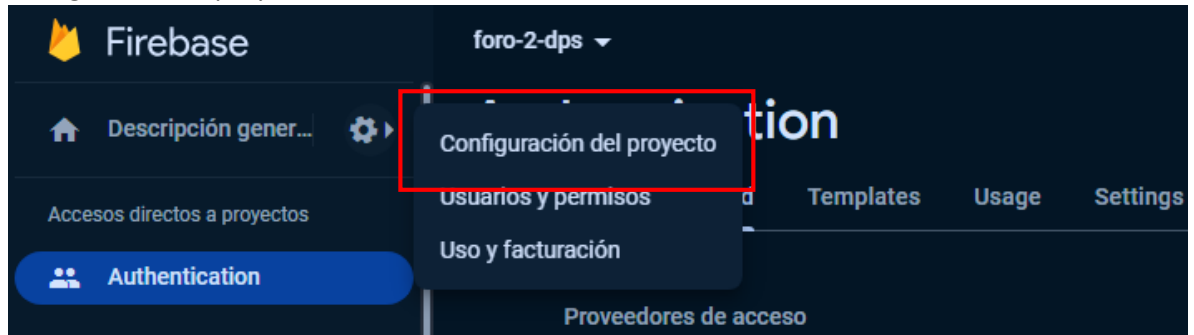
A screenshot of the EAS console interface. At the top, there is a header bar with the text "Agrega a la lista de entidades seguras los ID de cliente de proyectos externos (opcional)" and a question mark icon. Below the header, there is a text input field containing the client ID "17:71:F5:0C:C7:76:42:07:B2:79:BE:F1:1F:D4:4A:4D:CC:FB:AE". To the right of the input field is a button labeled "Agregar".

25. Y eso es todo del lado de nuestros servicios de nube, ahora nos encontramos conectados y deberemos crear la lógica en nuestra app

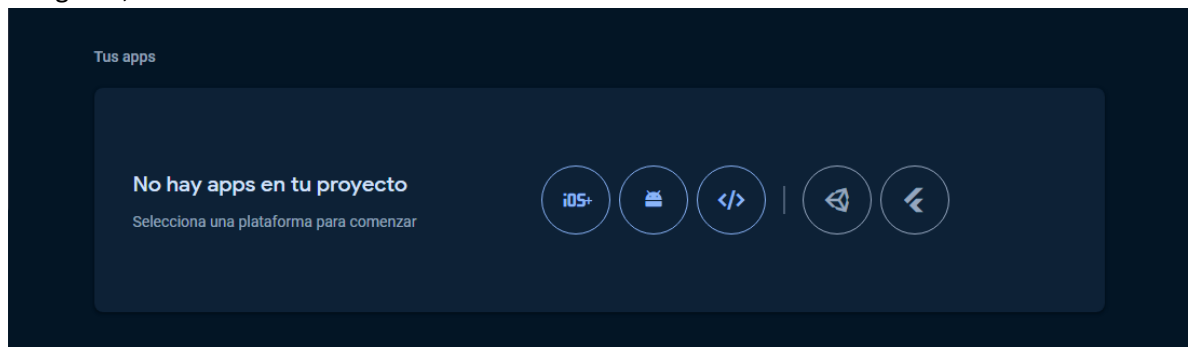


A screenshot of the EAS console interface, similar to the one above. The header bar contains the same text and icon. The text input field is currently empty. To the right of the input field is a button labeled "Agregar". Below the input field, the client ID "17:71:F5:0C:C7:76:42:07:B2:79:BE:F1:1F:D4:4A:4D:CC:FB:AE:25" is displayed.

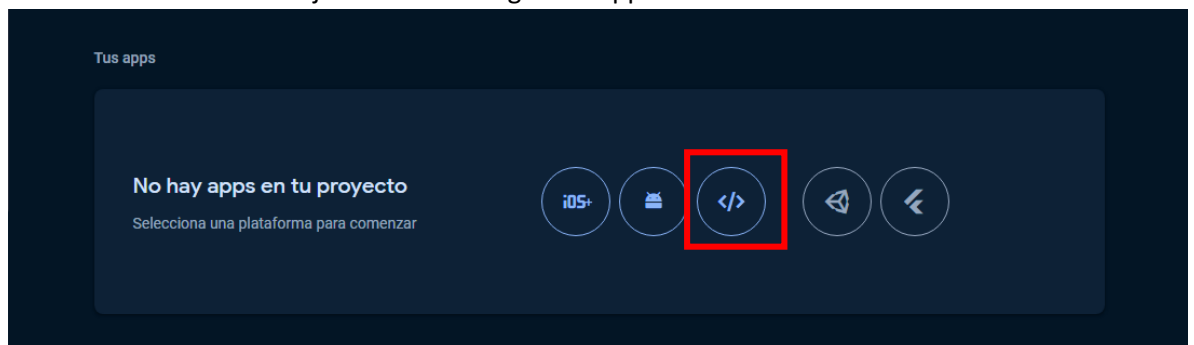
26. Mentiras, aun debemos conectar nuestra app con FireBase, asi que aca vamos de nuevo, nos dirigimos a nuestro proyecto de FireBase y nos dirigimos al menú lateral a configuración de proyectos



Al deslizarnos hasta la parte de abajo podremos ver que no poseemos ninguna app integrada, entonces aca comienza lo bueno



27. Entonces de las opciones de integracio que nos brindan, seleccionaremos web, ya que será mucho mas fácil de manejar en nuestra lógica de app



## 28. Registramos nuestra app con cualquier nombre no importa

[Ir a la documentación](#)

×

Agrega Firebase a tu aplicación web

1

Registrar app

Sobrenombre de la app ②

foro2dps-dc

☐

Además, configura **Firebase Hosting** para esta app. [Más información](#)

Hosting también se puede configurar más adelante. Puedes comenzar cuando quieras sin costo alguno.

Registrar app

2

Agrega el SDK de Firebase

Entonces obtenemos nuestro código sdk para integrarlo en nuestra app

2

Agrega el SDK de Firebase

☒ Usar npm

☐ Usar una etiqueta <script>

Si ya usas [npm](#) y un agrupador de módulos como [Webpack](#) o [Rollup](#), puedes ejecutar el siguiente comando para instalar la versión más reciente del SDK ([más información](#)):

\$ npm install firebase

Luego, inicializa Firebase y comienza a usar los SDK de los productos que quieres utilizar.

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
const firebaseConfig = {
  apiKey: "AIzaSyCuWMB2h8u-qPQnt-sjmNgQp-J2j1rlj0s",
  authDomain: "foro-2-dps-3561a.firebaseio.com",
  projectId: "foro-2-dps-3561a",
  storageBucket: "foro-2-dps-3561a.appspot.com",
  messagingSenderId: "355026092135",
  appId: "1:355026092135:web:4d7b2df31ab9121f2d54c2"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
```

**Nota:** Esta opción utiliza el [SDK de JavaScript modular](#), que proporciona un tamaño reducido del SDK.

Obtén más información sobre Firebase para la Web: [primeros pasos](#), [referencia de la API del SDK web](#) y [muestras](#)

Ir a la consola

