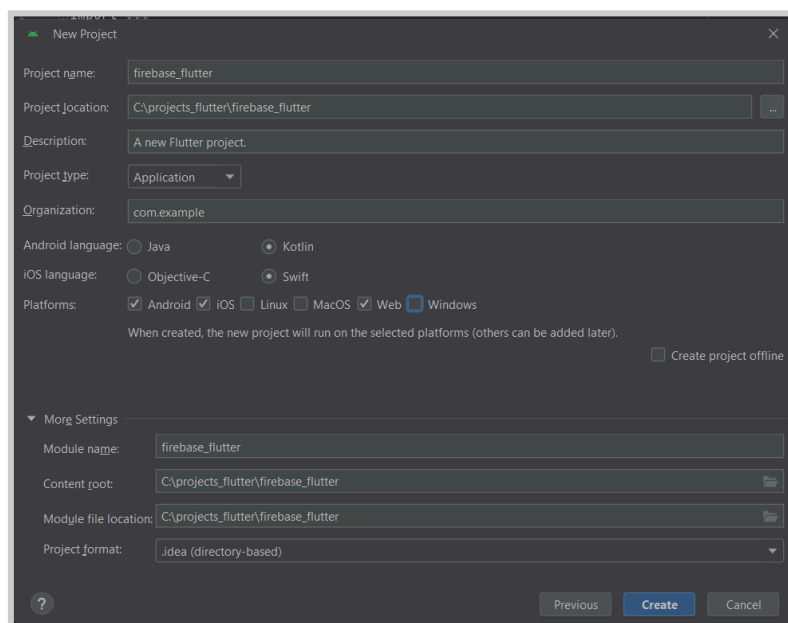


CONFIGURAÇÃO FIREBASE COM FLUTTER

Neste tutorial, você aprenderá a configurar o Firebase no Flutter e a implementar um botão de login utilizando o Google como método de autenticação. Vamos começar!

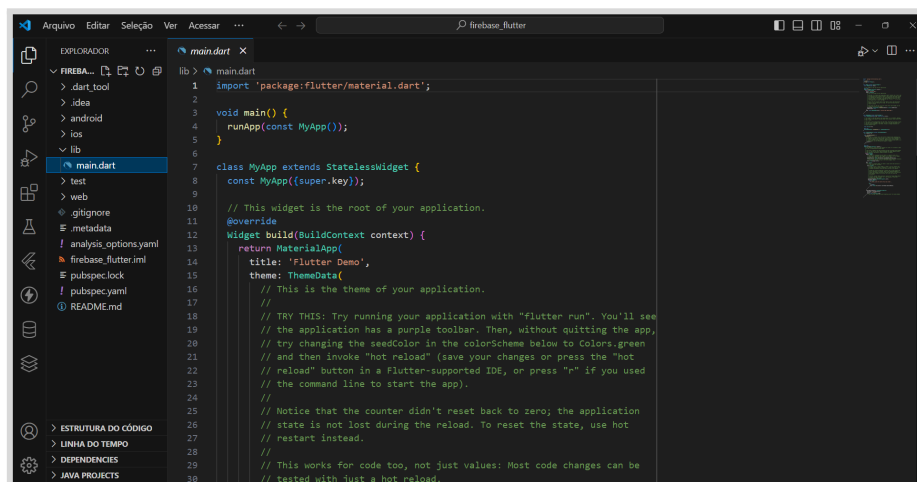
1. Criação do projeto Flutter

Para começar, vamos criar um novo projeto Flutter. Abra o Android Studio e siga os passos abaixo para criar um novo projeto, conforme as especificações mostradas na imagem a seguir.



1.1 Abrir o aplicativo no Visual Studio Code

O próximo passo é abrir o aplicativo no VSCode. Para fazer isso, abra o VSCode e siga o caminho **Arquivo > Abrir Pasta**. Em seguida, selecione a pasta onde você criou o projeto Flutter.



2. Configuração do Firebase

Para configurar o Firebase, é necessário adicionar algumas dependências essenciais para garantir o bom funcionamento da integração com o Firebase:

- [firebase_core](#)
- [firebase_auth](#)
- [sign_in_button](#)

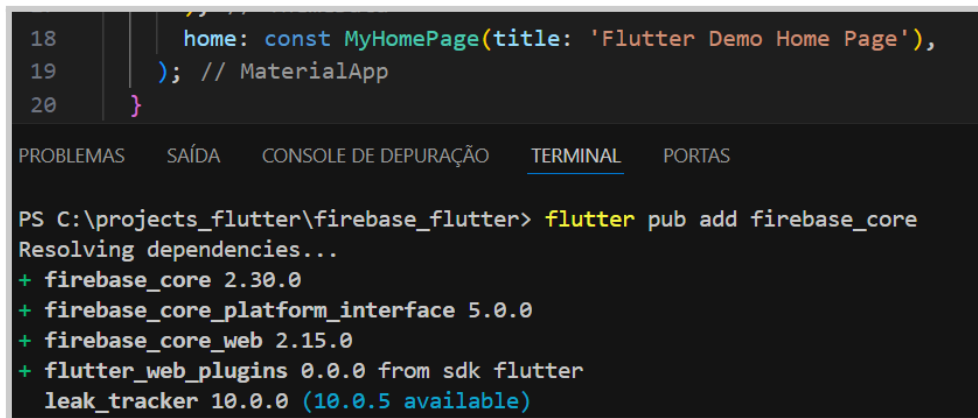
2.1 Adicionando as dependências do Firebase no projeto Flutter

Abra um novo terminal no próprio VSCode e digite os seguintes comandos para adicionar as dependências necessárias:

```
flutter pub add firebase_core
```

```
flutter pub add firebase_auth
```

```
flutter pub add sign_in_button
```

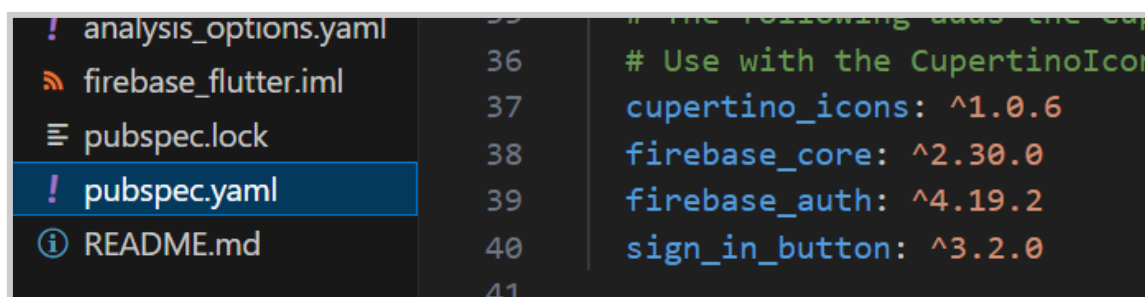


```
18     home: const MyHomePage(title: 'Flutter Demo Home Page'),
19   ); // MaterialApp
20 }
```

PROBLEMAS SAÍDA CONSOLE DE DEPURAÇÃO **TERMINAL** PORTAS

```
PS C:\projects_flutter\firebase_flutter> flutter pub add firebase_core
Resolving dependencies...
+ firebase_core 2.30.0
+ firebase_core_platform_interface 5.0.0
+ firebase_core_web 2.15.0
+ flutter_web_plugins 0.0.0 from sdk flutter
+ leak_tracker 10.0.0 (10.0.5 available)
```

Podemos verificar no arquivo “**Pubspec.yaml**” e garantir que todas as dependências foram instaladas, conforme exibido na imagem abaixo:



```
! analysis_options.yaml 35
🔥 firebase_flutter.iml 36
☰ pubspec.lock 37
! pubspec.yaml 38
📘 README.md 40
```

```
36 # Use with the CupertinoIcons
37 cupertino_icons: ^1.0.6
38 firebase_core: ^2.30.0
39 firebase_auth: ^4.19.2
40 sign_in_button: ^3.2.0
41
```

2.2 Acessando o console do Firebase

Em seguida, acesse o console do Firebase para criar um novo banco de dados para a nossa aplicação. Você pode acessá-lo [clicando aqui](#).

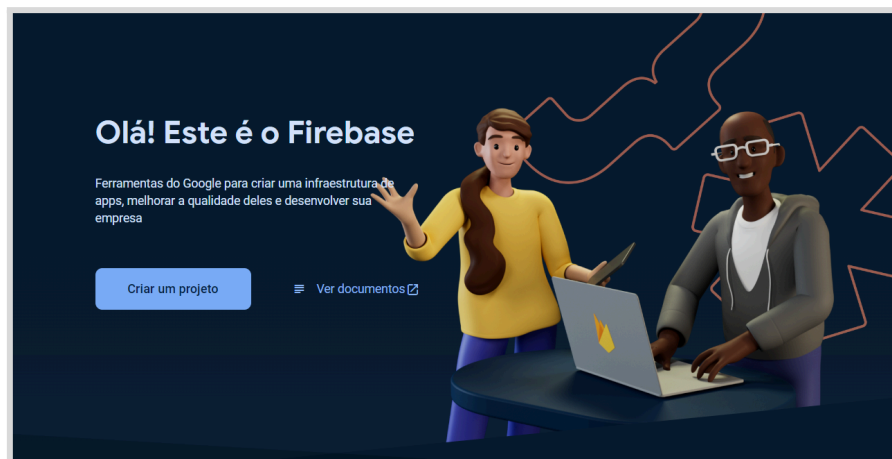
Para prosseguir, faça login com sua conta Google. Por favor, certifique-se de usar uma conta Google regular, e não uma conta de aluno.



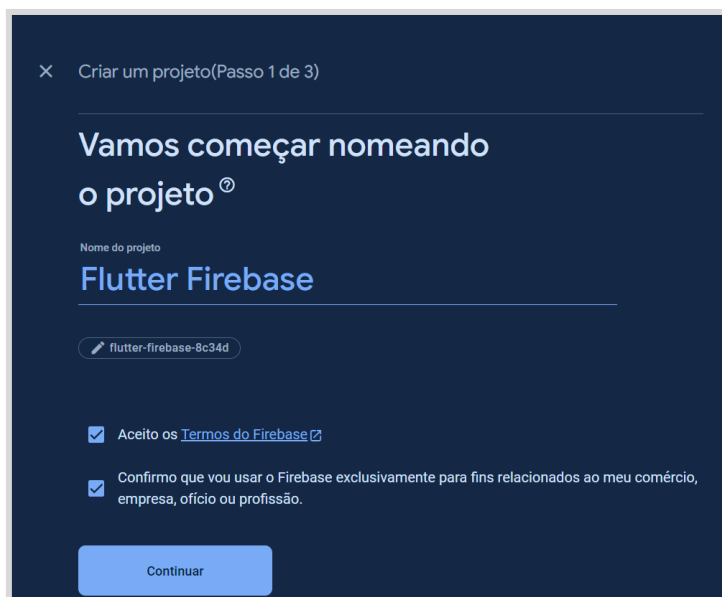
The image shows the Google login page. At the top left is the Google logo. Below it, the text "Fazer login" is displayed, followed by "Use sua Conta do Google". To the right, there is a text input field labeled "E-mail ou telefone". Below this field is a link that says "Esqueceu seu e-mail?". Further down, a message states: "Não está no seu computador? Use o modo visitante para fazer login com privacidade. Saiba mais sobre como usar o Modo visitante". At the bottom right, there are two buttons: "Criar conta" and "Avançar". At the very bottom, there is a language selector showing "Português (Brasil)" and links for "Ajuda", "Privacidade", and "Termos".

2.3 Criando um novo projeto no Firebase

Após fazer login no Firebase, clique no botão **"Criar um projeto"** para iniciar a criação de um novo projeto.



O primeiro passo é definir um nome para o projeto e selecionar as duas opções de confirmação disponíveis (semelhante ao apresentado na imagem). Na tela seguinte deixar desmarcado a opção de google analytics.

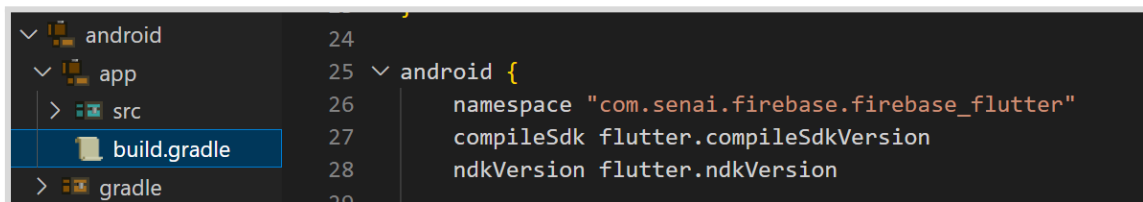


The image shows the Firebase project creation screen. At the top, there is a close button (X) and the text "Criar um projeto(Passo 1 de 3)". The main heading is "Vamos começar nomeando o projeto®". Below this, there is a label "Nome do projeto" and a text input field containing "Flutter Firebase". Underneath the input field, there is a button with a pencil icon and the text "flutter-firebase-8c34d". Below this, there are two checkboxes, both of which are checked. The first checkbox is labeled "Aceito os Termos do Firebase" with an external link icon. The second checkbox is labeled "Confirmo que vou usar o Firebase exclusivamente para fins relacionados ao meu comércio, empresa, ofício ou profissão.". At the bottom, there is a button labeled "Continuar".

No segundo passo, precisamos configurar o Firebase para Android. Para isso, acesse o console do Firebase e clique no ícone do Android.



Ao clicar no ícone do Android, uma tela aparecerá para registrar o aplicativo. No primeiro campo, insira o nome do pacote Android. Você pode encontrar esse nome no projeto, no VSCode, abrindo o arquivo “**build.gradle**” localizado no caminho: **android\app\build.gradle**.



Em seguida, insira o nome do aplicativo. Deve ficar semelhante à imagem abaixo:

[illegible]

Uma etapa crucial no processo de registro do aplicativo é a geração do certificado de assinatura, que é essencial para as funcionalidades de login do Firebase no aplicativo.

Para gerar essa chave, siga os passos abaixo:

1. Realize a primeira execução do app.
2. Abra o terminal (cmd) diretamente no seguinte caminho:

```
C:\Program Files\Java\jdk1.8.0_74\jre\bin
```

3. Utilize a ferramenta keytool, disponível no Java.
4. Com o terminal aberto, copie o caminho do arquivo debug do seu projeto. Por exemplo:

```
C:\flutter_projects\firebase_flutter\build\app\outputs\apk\debug\app-debug.apk
```

5. Digite o seguinte comando no terminal (cmd):

```
keytool -printcert -jarfile C:\flutter_projects\firebase_flutter\build\app\outputs\apk\debug\app-debug.apk
```

Este comando irá gerar o certificado de assinatura necessário para a configuração do Firebase.

```
C:\Program Files\Java\jdk1.8.0_74\jre\bin>keytool -printcert -jarfile C:\flutter_projects\firebase_flutter\build\app\outputs\apk\debug\app-debug.apk
Signatário #1:

Assinatura:

Proprietário: C=US, O=Android, CN=Android Debug
Emissor: C=US, O=Android, CN=Android Debug
Número de série: 1
Válido de: Mon Mar 18 17:10:53 GMT-03:00 2024 a: Wed Mar 11 17:10:53 GMT-03:00 2054
Fingerprints do certificado:
MD5: 86:48:07:62:D9:BE:7B:71:22:1E:22:2F:17:E6:14:06
SHA1: C1:26:D2:5E:3B:D6:27:B1:9D:84:6C:C8:D3:5E:57:85:FC:FB:8F:7F
SHA256: 53:E8:9D:A3:59:02:DD:23:3F:27:E3:5C:04:8E:CD:E1:A8:28:30:88:50:C4:9A:94:5A:A3:7F:EA:0D:20:EF:D7
Nome do algoritmo de assinatura: SHA1withRSA
Versão: 1
```

Com a chave SHA1 gerada, copie-a e cole na configuração do seu app no Firebase. Em seguida, clique no botão "Registrar App" para concluir o processo de registro.

1

Registrar app

Nome do pacote do Android ?

com.senai.firebase.firebase_flutter

Apelido do app (opcional) ?

Firebase Flutter

Certificado de assinatura de depuração SHA-1 (opcional) ?

5E:3B:D6:27:B1:9D:84:6C:C8:D3:5E:57:85:FC:FB:8F:7F

Necessário para o Dynamic Links, o Login do Google ou o suporte por telefone no Auth. Edite os certificados SHA-1 nas configurações.

Registrar app

Agora, faça o download do arquivo de configuração do Firebase e cole-o na pasta especificada, conforme mostrado na imagem abaixo:

✓ Registrar app

Nome do pacote Android: com.senai.firebase.firebase_flutter, apelido do app: Firebase Flutter

2

Faça o download e adicione o arquivo de configuração para o Android Studio abaixo | [Unity](#) [C++](#)

↓ Fazer o download de google-services.json

Mude para a visualização **Projeto** no Android Studio para acessar o diretório raiz.

Mova o arquivo `google-services.json` salvo para o diretório raiz do módulo (nível do app).

google-services.json

Próxima

Project

MyApplication [My Application]

.gradle

.idea

app

libs

src

.gitignore

build.gradle.kts

google-services.json

proguard-rules.pro

gradle

Salvar como

C:\flutter_projects\android\app

Pesquisar em app

Organizar Nova pasta

Este Computador

Desktop

Documentos

Downloads

Imagens

Músicas

Objetos 3D

Videos

Disco Local (C:)

Rede

Nome	Data de modificação
src	18/03/2024 16:20

Nome: google-services

Tipo: Arquivo JSON

Ocultar pastas

Salvar Cancelar

Project

MyApplication [My Application]

.gradle

.idea

app

libs

src

.gitignore

build.gradle.kts

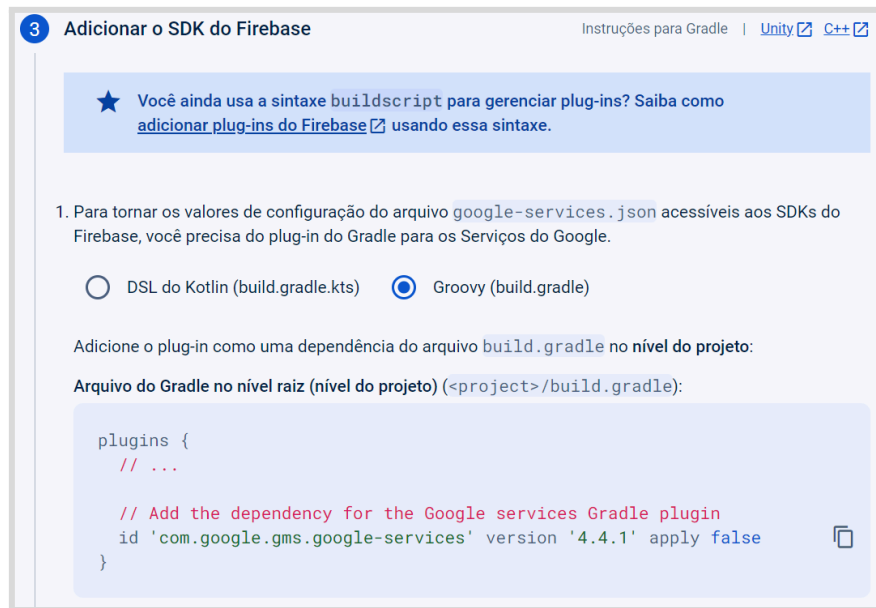
google-services.json

proguard-rules.pro

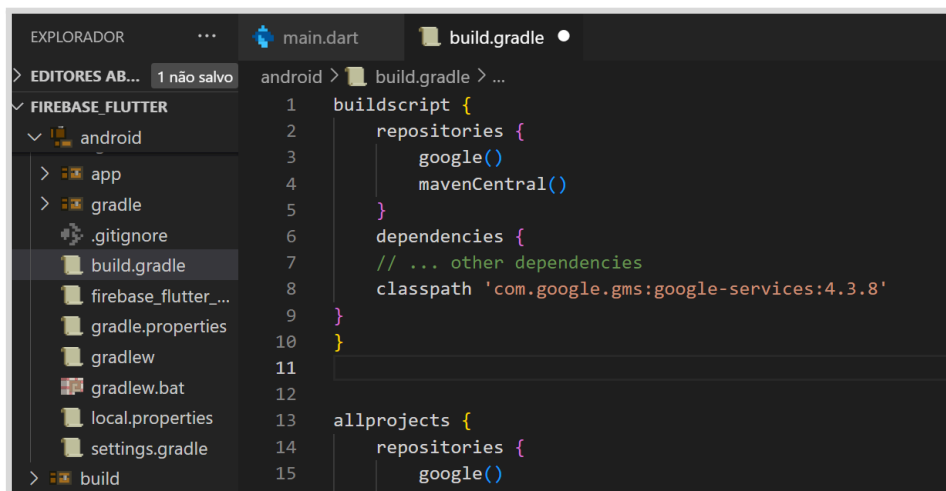
gradle

2.3 Adicionando o SDK do Fibase ao projeto

Para integrar o SDK do Firebase ao projeto Flutter, é necessário adicionar algumas linhas de código nos arquivos pertinentes.



Abra o arquivo build.gradle, localizado em “**android/build.gradle**”, e adicione o trecho de código “**buildscript**”, da linha 1 até a linha 10, conforme mostrado na imagem abaixo:



Outro arquivo “**build.gradle**” deve ser modificado. No entanto, é importante prestar muita atenção, pois **não é o mesmo que foi modificado anteriormente**. Este arquivo está localizado em outra pasta, no caminho “**android/app/build.gradle**”. Adicione o trecho de código necessário conforme mostrado na imagem abaixo:

2. Em seguida, no arquivo `build.gradle` do módulo (nível do app), adicione o plug-in `google-services` e todos os SDKs do Firebase que você quer usar no aplicativo:

Arquivo do Gradle do módulo (nível do app) (<project>/<app-module>/build.gradle):

```
plugins {  
    id 'com.android.application'  
    // Add the Google services Gradle plugin  
    id 'com.google.gms.google-services'  
    ...  
}  
  
dependencies {  
    // Import the Firebase BoM  
    implementation platform('com.google.firebase:firebase-bom:32.8.1')  
  
    // TODO: Add the dependencies for Firebase products you want to use  
    // When using the BoM, don't specify versions in Firebase dependencies  
    // https://firebase.google.com/docs/android/setup#available-libraries  
}
```

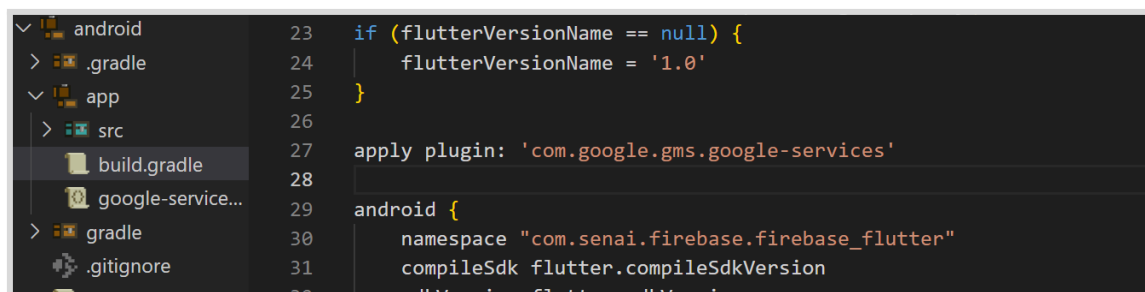
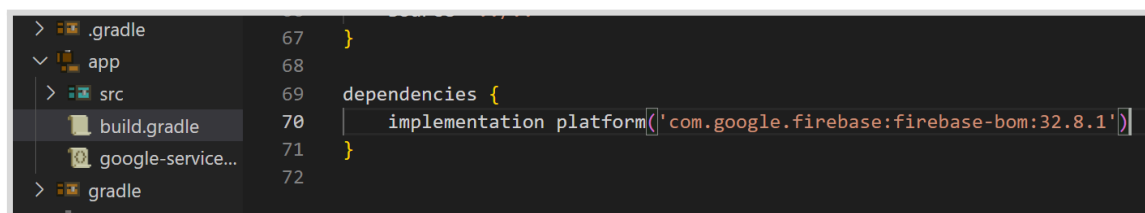
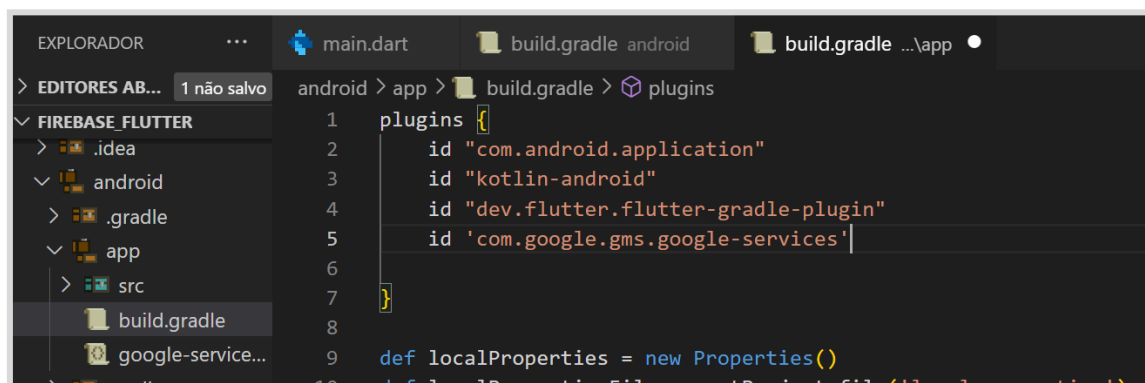
Ao usar a BoM do Firebase para Android, o app sempre vai utilizar versões da biblioteca do Firebase compatíveis.

[Saiba mais](#)

3. Depois de adicionar o plug-in e os SDKs desejados, sincronize seu projeto do Android com os arquivos do Gradle.

[Anterior](#)

[Próxima](#)



Após concluir essa configuração, clique em **"Continuar para o Console"** para prosseguir:



3. Configuração da CLI do Firebase

Para baixar e configurar a cli do firebase, acesse o [link](#)

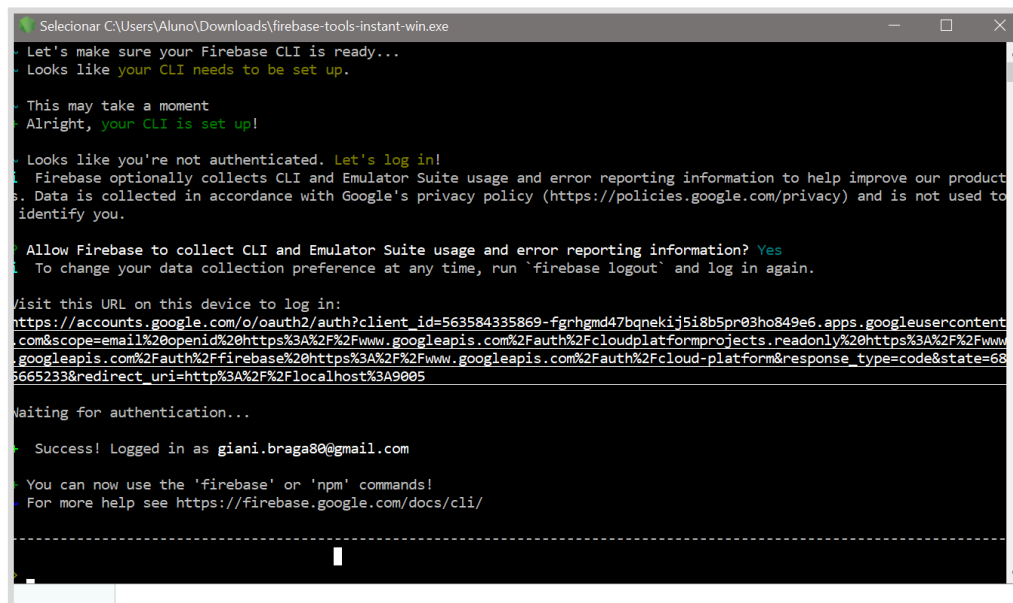
3.1 baixar a ferramenta CLI Firebase:

Clique no link e faça o download da ferramenta: [cli_firebase](#)

Caso apareça uma mensagem de bloqueio, clique em **"Mais informações"** e, em seguida, selecione **"Executar mesmo assim"** para prosseguir com a instalação da ferramenta.



Ao executar o programa, será aberta uma nova guia contendo o processo de instalação. Durante esse processo, será necessário fazer login no Firebase. Uma nova guia será aberta, onde você deverá inserir o seu login do Google para continuar:

A screenshot of a Windows command prompt window titled "Selecionar C:\Users\Aluno\Downloads\firebase-tools-instant-win.exe". The window shows the Firebase CLI installation process. It starts with a message "Let's make sure your Firebase CLI is ready..." and "Looks like your CLI needs to be set up." followed by "This may take a moment" and "Alright, your CLI is set up!". Then it says "Looks like you're not authenticated. Let's log in!". It provides a long URL for authentication and asks "Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes". After logging in as "giani.braga80@gmail.com", it says "Success! Logged in as giani.braga80@gmail.com" and "You can now use the 'firebase' or 'npm' commands!".

```
Selecionar C:\Users\Aluno\Downloads\firebase-tools-instant-win.exe
Let's make sure your Firebase CLI is ready...
Looks like your CLI needs to be set up.

This may take a moment
Alright, your CLI is set up!

Looks like you're not authenticated. Let's log in!
Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our product.
Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to
identify you.

Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes
To change your data collection preference at any time, run 'firebase logout' and log in again.

Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335869-fgrhgm47bqnekij5i8b5pr03ho849e6.apps.googleusercontent.com&scope=email%20openid%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloudplatformprojects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ffirebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform&response_type=code&state=685665233&redirect_uri=http%3A%2F%2Flocalhost%3A9005

Waiting for authentication...

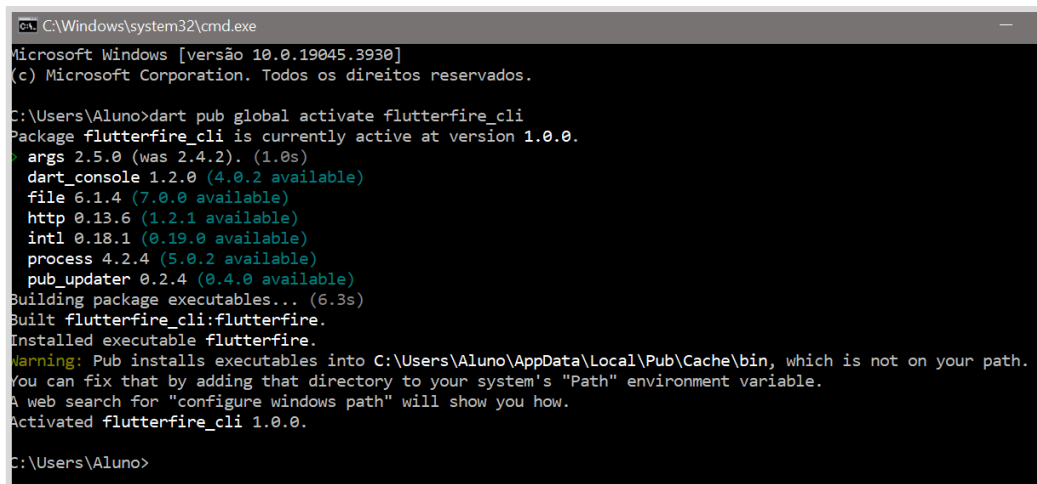
Success! Logged in as giani.braga80@gmail.com

You can now use the 'firebase' or 'npm' commands!
For more help see https://firebase.google.com/docs/cli/
```

Se aparecer uma mensagem de “**Sucess!**”, significa que a instalação foi bem-sucedida.

Após a instalação, abra um novo prompt de comando e digite o seguinte comando:

dart pub global activate flutterfire_cli

A screenshot of a Windows command prompt window titled "C:\Windows\system32\cmd.exe". It shows the execution of the command "dart pub global activate flutterfire_cli". The output indicates that the package "flutterfire_cli" is currently active at version 1.0.0. It lists the versions of various dependencies: args 2.5.0 (was 2.4.2), dart_console 1.2.0 (4.0.2 available), file 6.1.4 (7.0.0 available), http 0.13.6 (1.2.1 available), intl 0.18.1 (0.19.0 available), process 4.2.4 (5.0.2 available), and pub_updater 0.2.4 (0.4.0 available). It then says "Building package executables..." and "Built flutterfire_cli:flutterfire.". A warning message states: "Warning: Pub installs executables into C:\Users\Aluno\AppData\Local\Pub\Cache\bin, which is not on your path. You can fix that by adding that directory to your system's 'Path' environment variable. A web search for 'configure windows path' will show you how." Finally, it says "Activated flutterfire_cli 1.0.0.".

```
C:\Windows\system32\cmd.exe
Microsoft Windows [versão 10.0.19045.3930]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\Aluno>dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 1.0.0.
args 2.5.0 (was 2.4.2). (1.0s)
dart_console 1.2.0 (4.0.2 available)
file 6.1.4 (7.0.0 available)
http 0.13.6 (1.2.1 available)
intl 0.18.1 (0.19.0 available)
process 4.2.4 (5.0.2 available)
pub_updater 0.2.4 (0.4.0 available)
Building package executables... (6.3s)
Built flutterfire_cli:flutterfire.
Installed executable flutterfire.
Warning: Pub installs executables into C:\Users\Aluno\AppData\Local\Pub\Cache\bin, which is not on your path.
You can fix that by adding that directory to your system's "Path" environment variable.
A web search for "configure windows path" will show you how.
Activated flutterfire_cli 1.0.0.

C:\Users\Aluno>
```

3.2 Adicionando as variáveis de ambiente

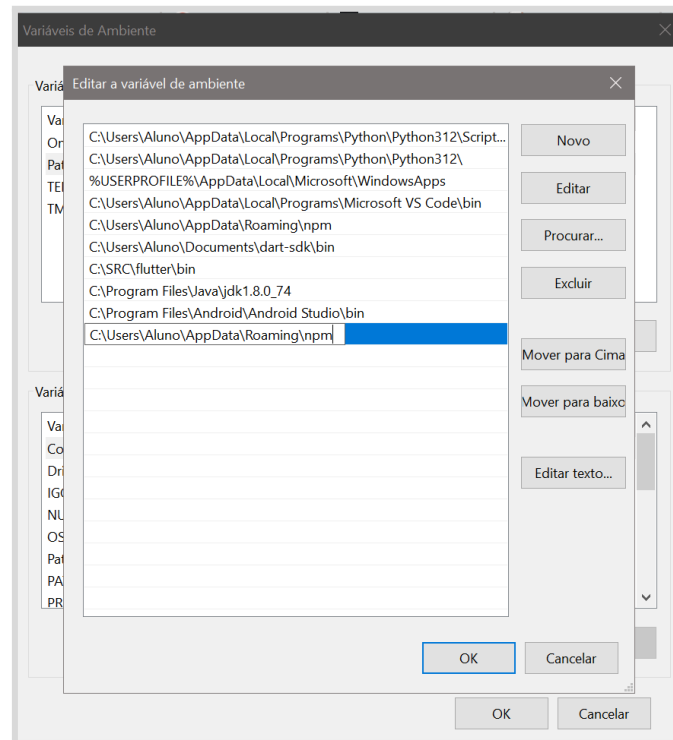
Em seguida, você deve adicionar o Firebase ao PATH para poder utilizá-lo em qualquer lugar.

1. Primeiro, é necessário copiar o caminho específico onde foi instalado a cli do firebase:

C:\Users\Aluno\AppData\Roaming\npm

C:\Users\Aluno\AppData\Local\Pub\Cache\bin

2. Em segundo, abrir as variáveis de ambientes e adicionar o caminho copiado ao path



3.3 Configurando a conexão entre o Firebase e o Projeto Flutter

Agora, abra o Visual Studio (se estiver aberto, feche e abra novamente) e estabeleça a conexão entre o Firebase e o aplicativo.

No terminal do Visual Studio, digite o seguinte comando:

flutterfire configure

Ao executar o comando **“flutterfire configure”**, serão exibidas as opções contendo todos os projetos existentes no Firebase. Selecione o projeto que foi criado no início deste tutorial:

```
PROBLEMAS  SAÍDA  CONSOLE DE DEPURAÇÃO  TERMINAL  PORTAS

i Found 10 Firebase projects.
? Select a Firebase project to configure your Flutter application with >
  appcompras-8644b (AppCompras)
  appcontatosfirebase-7fb49 (AppContatosFireBase)
  appfluttercontatos-854f4 (AppFlutterContatos)
  apppetshop-c7ce3 (AppPetShop)
  chatflutter-7d31b (chatflutter)
> fir-flutter-tutorial-be30e (Firebase Flutter Tutorial)
  flutter-firebase-45bd1 (Flutter Firebase)
  loja-virtual-8c4e0 (loja-virtual)
  lojavirtual-9b79f (LojaVirtual)
  petshopapp-44635 (PetshopApp)
<create a new project>
```

Após selecionar o projeto, escolha a plataforma que você deseja conectar. Deixe selecionadas as opções “**android**”, “**ios**” e “**web**”. Em seguida, pressione “**Enter**” na opção “**android**”, conforme mostrado na imagem abaixo:

```
i Found 10 Firebase projects.
✓ Select a Firebase project to configure your Flutter application with · fir-flutter-tutorial-be30e (Flutter Tutorial)
? Which platforms should your configuration support (use arrow keys & space to select)? >
✓ android
✓ ios
  macos
✓ web
  windows
```

Se tudo ocorrer conforme o esperado, uma mensagem de “Successfully” deve ser exibida, semelhante à imagem abaixo:

```
PROBLEMAS  SAÍDA  CONSOLE DE DEPURAÇÃO  TERMINAL  PORTAS
i Registered a new Firebase ios app on Firebase project fir-flutter-tutorial-be30e.
i Firebase web app firebase_flutter (web) is not registered on Firebase project fir-flutter-tutorial-be30e.
i Registered a new Firebase web app on Firebase project fir-flutter-tutorial-be30e.

Firebase configuration file lib\firebase_options.dart generated successfully with the following Firebase apps:

Platform  Firebase App Id
web        1:490969390128:web:5ede5065e62f9c2e8a61f5
android    1:490969390128:android:f55b418f869c34338a61f5
ios        1:490969390128:ios:cf069a205ba280a18a61f5

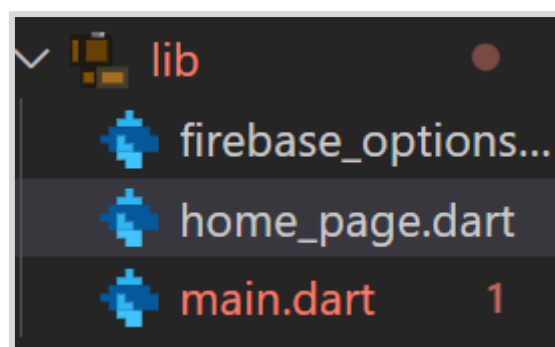
Learn more about using this file and next steps from the documentation:
> https://firebase.google.com/docs/flutter/setup
PS C:\flutter_projects\firebase_flutter>
```

Com isso, a configuração do Firebase está finalizada. A seguir, para fins de teste, vamos criar um botão de login com o Google

4 DESENVOLVENDO O APLICATIVO

Acesse o arquivo principal “**main.dart**” e apague toda a parte da classe “**MyHomePage**”. Além disso, modifique a inicialização do aplicativo, adicionando um `await` no código, pois será necessário aguardar um retorno da API do Firebase. O código pode ser consultado no GitHub: [main.dart](#)

A seguir, vamos criar um novo arquivo chamado “**home_page.dart**”. Este arquivo conterá a nossa classe com o botão para realizar o login no Google. O código pode ser consultado no GitHub: [home_page.dart](#)



As telas do aplicativo devem ficar semelhantes às exibidas abaixo:

