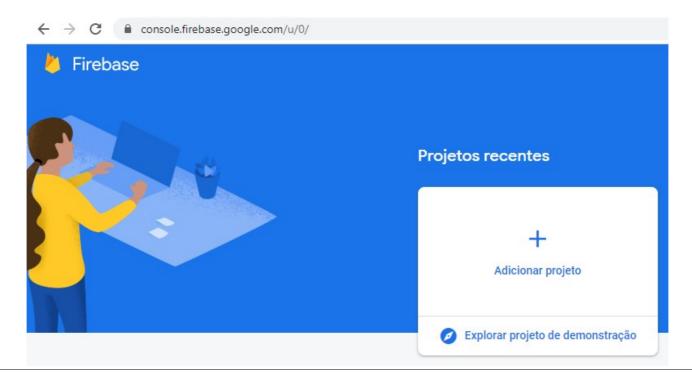
Firebase Authentication



Criando um projeto

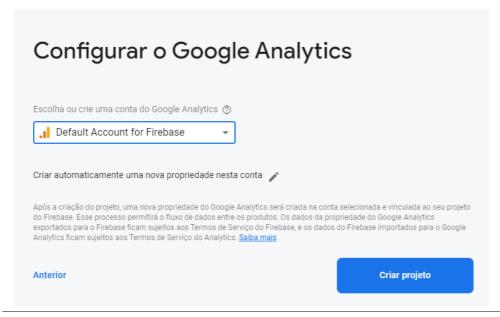
 O primeiro passo para usar qualquer serviço do Firebase é criar um projeto em: https://console.firebase.google.com/

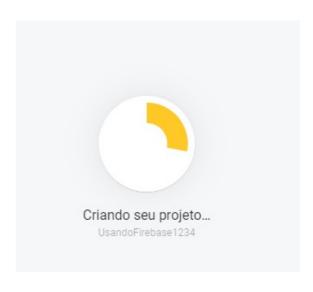


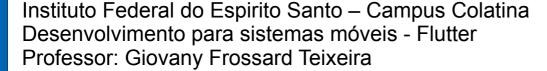


Criando um projeto

- Na tela seguinte você deverá informar o nome do seu projeto e clicar em "Continue"
- É recomendável manter o Google Analytics.
- Depois selecione a opção Default Account for Firebase e "Criar projeto".

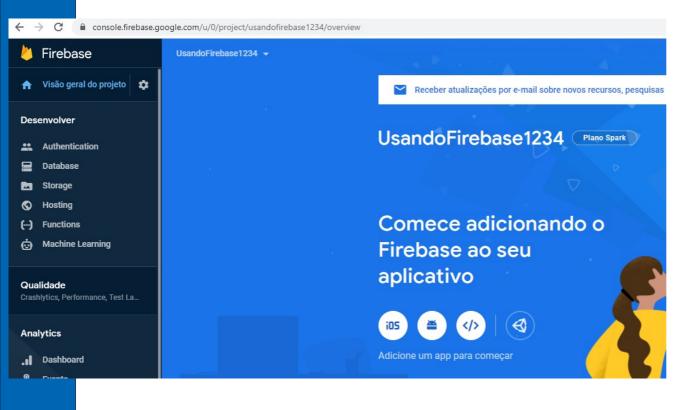






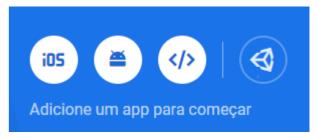


Criando um projeto



- Se tudo deu certo você deverá ver uma tela como ao lado.
- É importante notar que ainda não temos nenhuma configuração para Android ou iOS. Isso ainda precisa ser feito

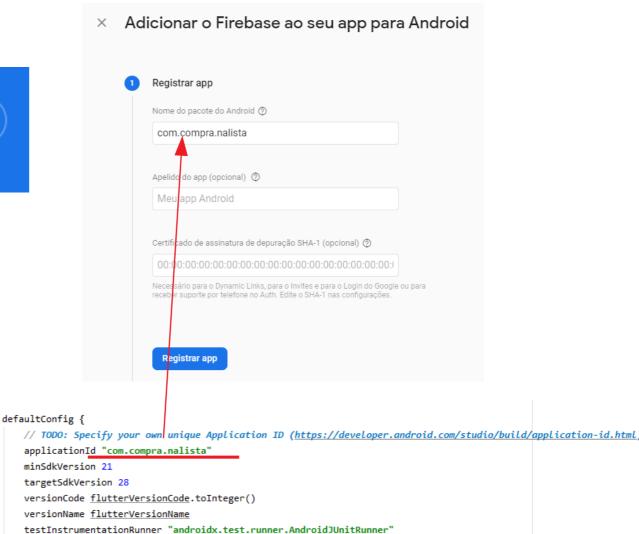






Configuração Android







// TODO: Specify your own unique Application ID (https://developer.android.com/studio/build/application-id.html). applicationId "com.compra.nalista" minSdkVersion 21 targetSdkVersion 28 versionCode flutterVersionCode.toInteger() versionName flutterVersionName testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner" multiDexEnabled true





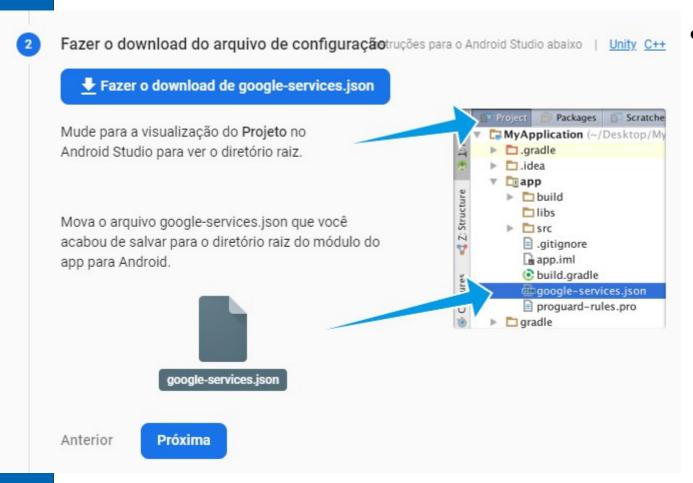
O SHA-1 pode ser obtido no terminal usando os seguintes comandos:

Linux ou Mac: keytool -list -v -keystore ~/.android/debug.keystore \ -alias androiddebugkey -storepass android -keypass android

Windows: keytool -list -v -keystore "C:\Users\<seu_usuario>\.android\debug.keystore" -alias androiddebugkey -storepass android -keypass android



Configuração Android



 Agora deve ser feito o download do arquivo de configuração gerado e ele deve ser colocado no local correto. Conforme a imagem que aparece na página:



```
Adicionar o SDK do Firebase
                                                    Instruções para Gradle | Unity C++
O plug-in dos serviços do Google para Gradle ♂ carrega o arquivo google-services.json
cujo download você acabou de fazer. Modifique seus arquivos build gradle para usar o plug-in.
build.gradle no nível do projeto (<project>/build.gradle):
  buildscript {
     repositories -
       // Check that you have the following line (if not, add it):
       google() // Google's Maven repository
     dependencies {
      // Add this line
      classpath 'com.google.gms:google-services:4.3.3'
  allprojects {
     repositories {
      // Check that you have the following line (if not, add it):
      google() // Google's Maven repository
build.gradle no nível do app (ct>/<app-module>/build.gradle):
  apply plugin: 'com.android.application'
  // Add this line
   apply plugin: 'com.google.gms.google-services'
  dependencies {
     // add the Firebase SDK for Google Analytics
```

implementation 'com.google.firebase:firebase-analytics:17.2.2'

// https://firebase.google.com/docs/android/setup#available-librarie

// add SDKs for any other desired Firebase products

Configuração Android

- Agora modifique os arquivos build.gradle conforme as instruções.
- O Google até coloca botões para copiar os códigos necessários.



Configuração Android

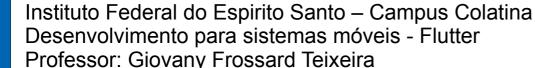
No passo 4 clique em Pular Etapa.





Ao selecionar as configurações do app Android, informe o e-mail de suporte

Este será o endereço de e-mail apresentado aos seus usuários no momento da autenticação com o Google. Ele pode ser alterado para o e-mail que você usou no login ou um e-mail de um Grupo do Google gerenciado por você.





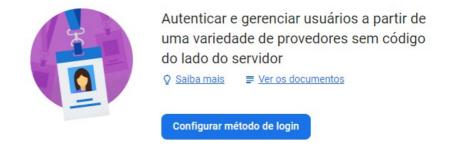
Firebase Authentication

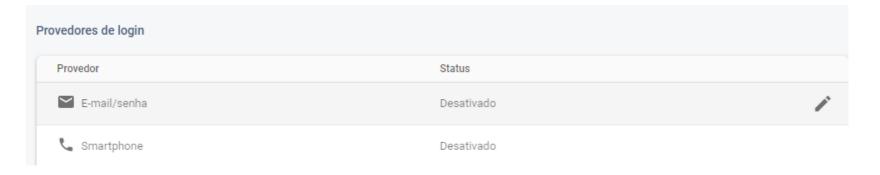
- Para fazer autenticação precisaremos usar o plugin firebase_auth.
- O processo de instalação é basicamente o mesmo de outros plugins que já instalamos e está documentado em:
 - https://pub.dev/packages/firebase_auth#-installing-tab-
- Também é importante ler a aba Readme para, se necessário, acrescentar algumas dependências nos arquivos de build.gradle.



Configurar o método de login

- Voltando ao console do Firebase, selecione seu projeto e no menu lateral Authentication.
- Selecione "Configurar método de login". No nosso caso, utilizaremos Email/senha.

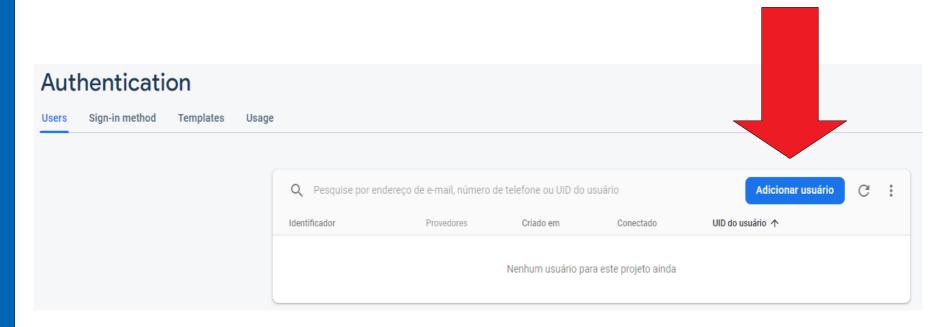






Cadastro de usuários

 É possível cadastrar alguns usuários no Firebase Authentication (isso também pode ser feito via codificação no aplicativo).

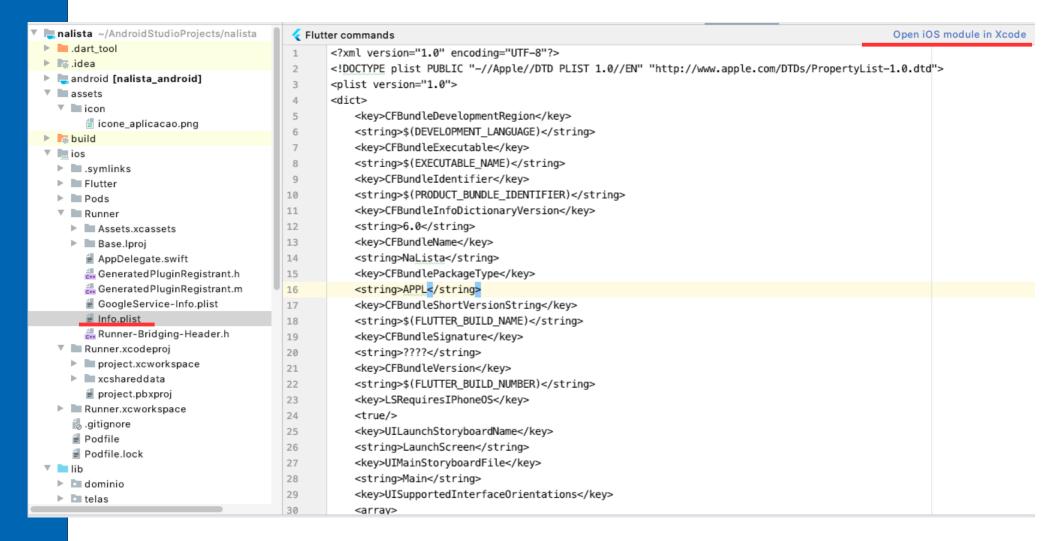




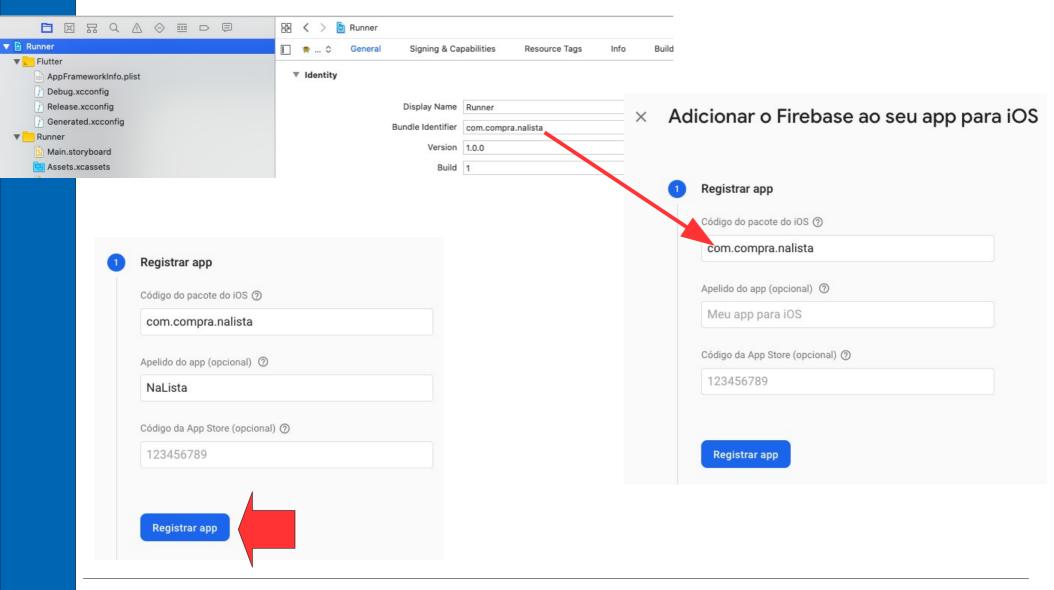
- O primeiro passo é ir no Console do Firebase e na opção Visão geral do Projeto.
- Em seguida clicar em "Adicionar App" e escolher a opção iOS.



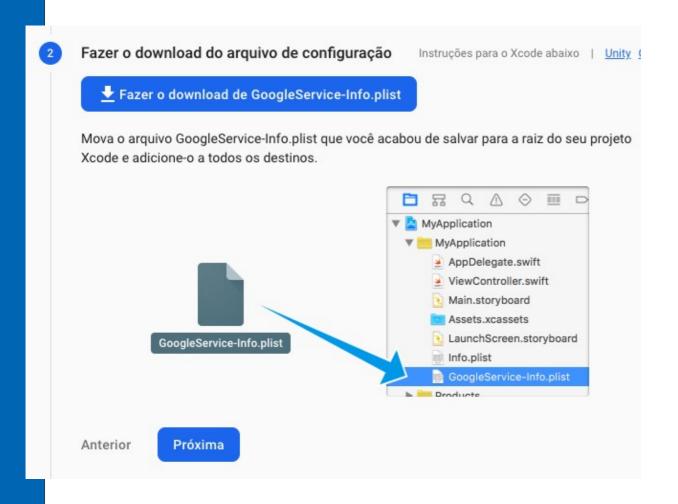




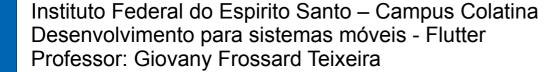








 Na sequência devemos baixar o arquivo gerado e colocá-lo no local indicado pela imagem (deve-se inserir o arquivo via XCode para que ele seja reconhecido no projeto).



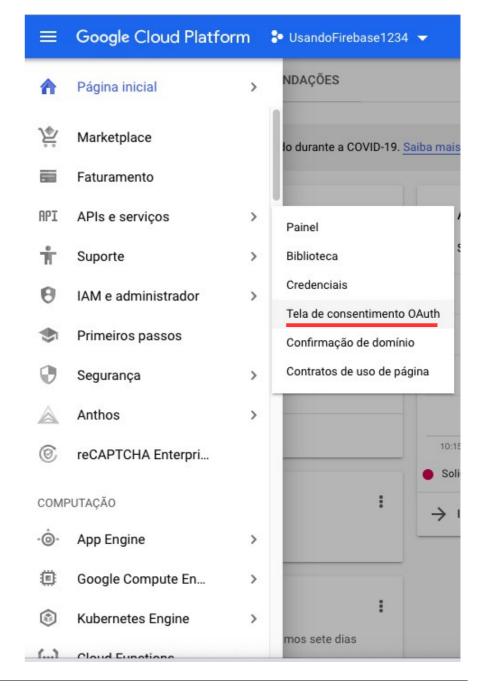


- O passo 3 pode ser ignorado pois o Flutter fará as instalações necessárias quando for rodar o projeto.
- Os passos 4 e 5 também podem ser ignorados.
 No passo 5 deve-se clicar em Pular Etapa.
- Por fim, verifique em Visão geral do Projeto→E-mail para suporte está preenchido com um e-mail válido.



Configuração Google Cloud

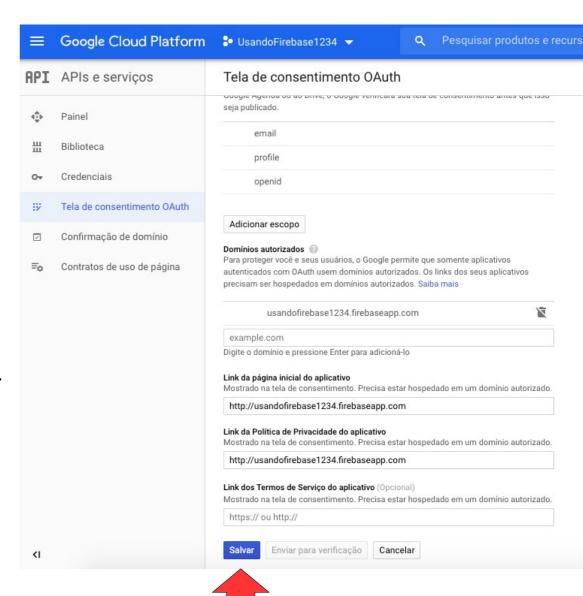
- Se o projeto não funcionar vá em https://console.cloud.google.com/.
- Todo projeto criado no Firebase cria automaticamente um projeto no Cloud do Google.
- Clique então na opção da imagem ao lado:
- Na tela seguinte peça para Editar o Aplicativo.





Configuração Google Cloud

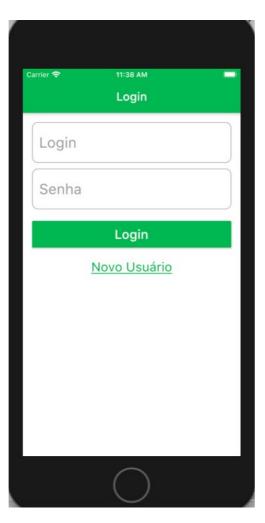
- Basicamente precisamos preencher a página inicial do aplicativo e a política de privacidade.
- Os links colocados não precisam ser verdadeiros.
- Faça como na imagem ao lado.
- Note que apenas copiei o Domínio Autorizado para os campos que precisamos preencher.

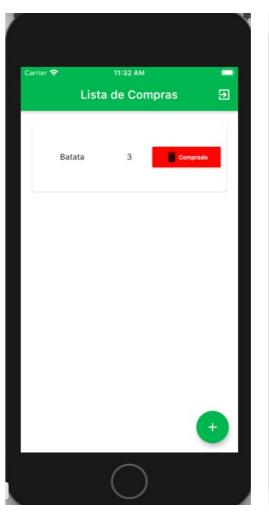




Telas da aplicação iOS











Agora vamos para o código ver como logar, deslogar e cadastrar um novo usuário



Dúvidas?



