










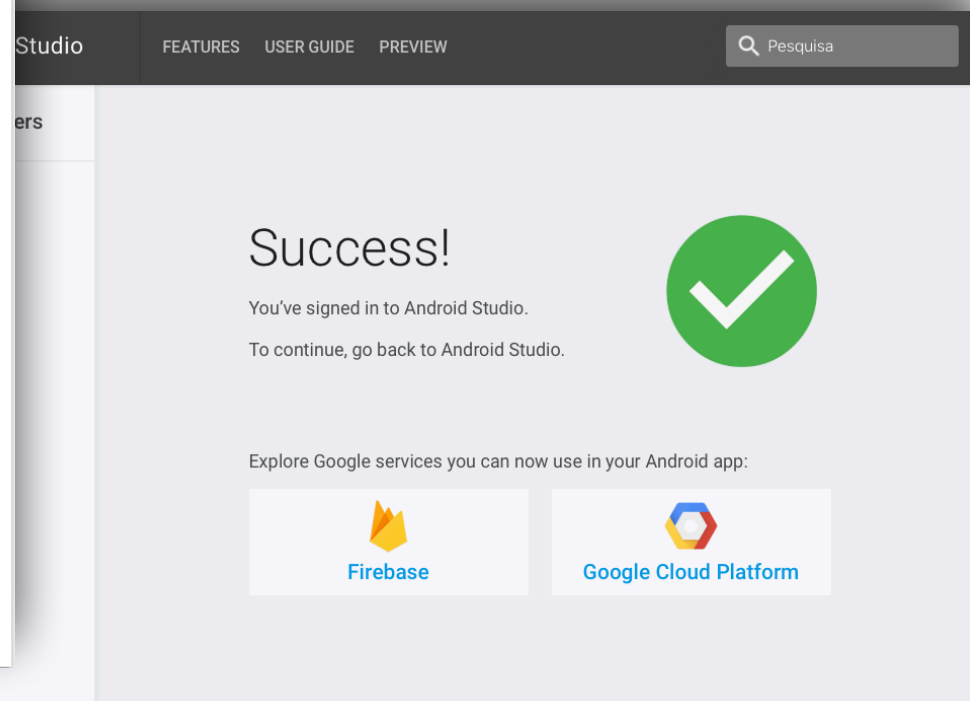
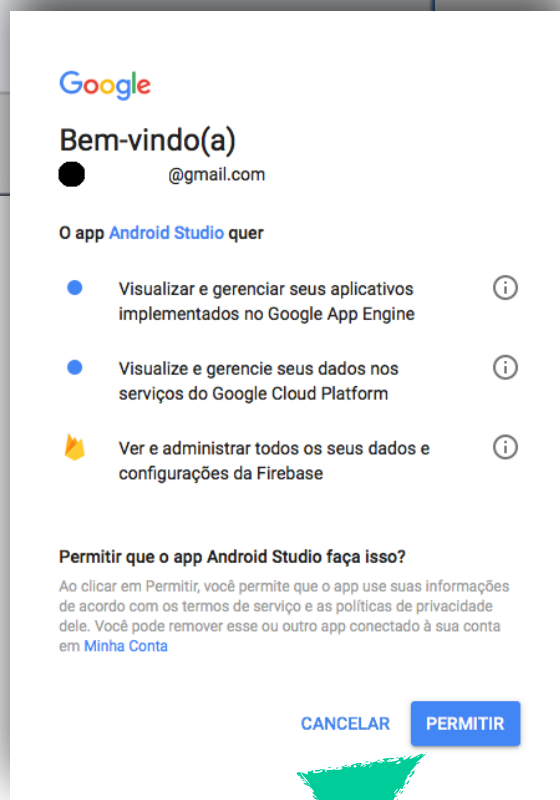
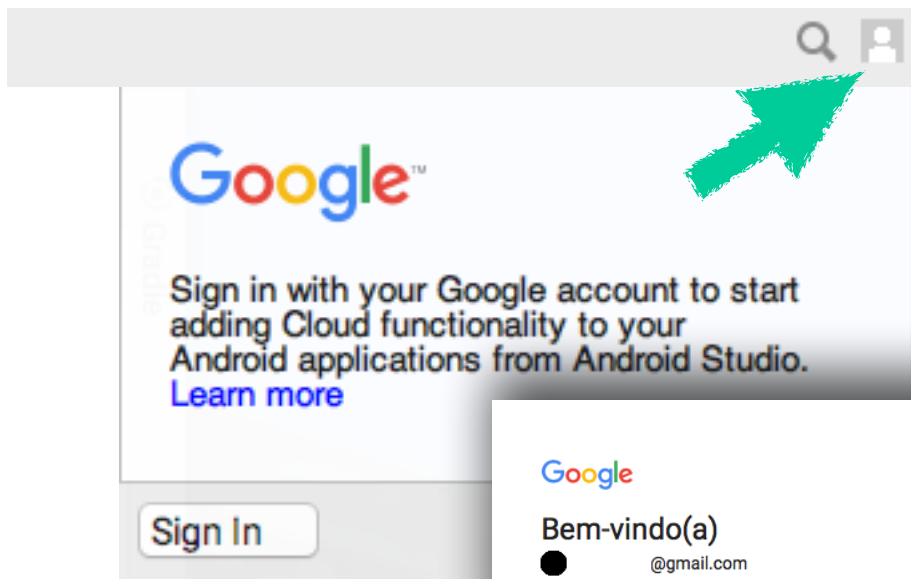
## Firestore Authentication

Com o objetivo de facilitar o desenvolvimento de mecanismos de autenticação para aplicações o *Firebase Authentication* oferece uma plataforma completa possibilitando a autenticação de aplicações com E-Mail/Senha, via Google, Twitter, Facebook, Github, etc

Provedores de login	
Provedor	Status
 E-mail/senha	Desativado
 Smartphone	Desativado
 Google	Desativado
 Facebook	Desativado
 Twitter	Desativado
 GitHub	Desativado
 Anônimo	Desativado

# Configurando o Ambiente

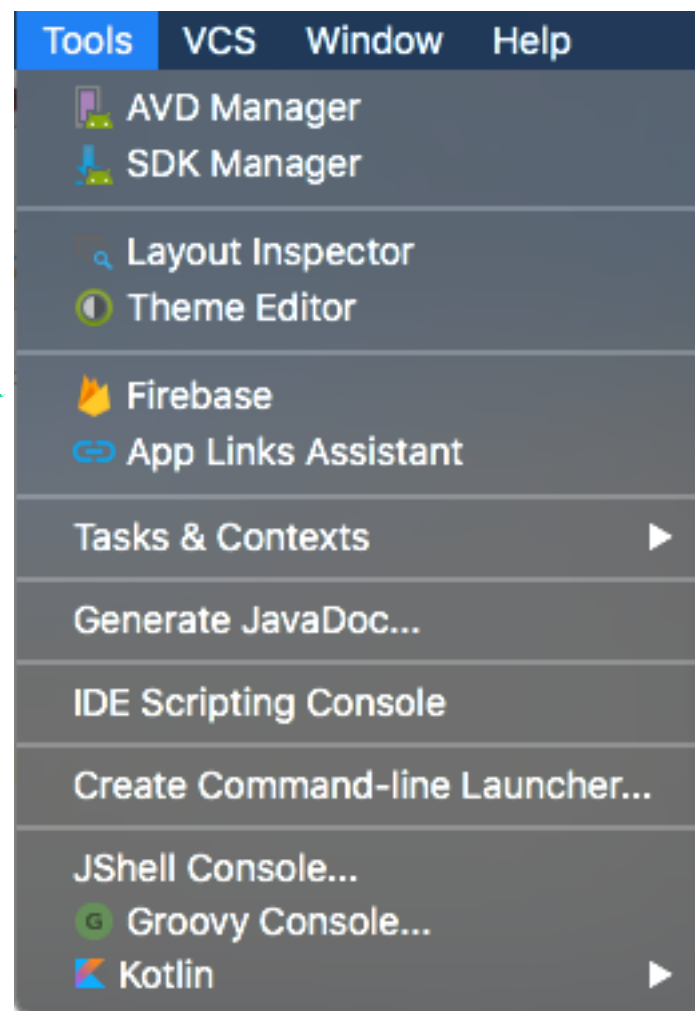
Antes de iniciar, faça o login no Google através do Android Studio





# Configurando o Ambiente

Para que uma aplicação Android possa fazer autenticação no Firebase, é necessário que seja definida uma identificação para a aplicação, desta forma será possível habilitar os tipos de autenticação e controlar a lista dos usuarios.

Esta configuração pode ser efetuada a partir do Android Studio no menu **Tools -> Firebase**













# Configurando o Ambiente


Assistant  



Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app. [Learn more](#)




- ▶  **Analytics**  
Measure user activity and engagement with free, easy, and unlimited analytics. [More info](#)
- ▶  **Cloud Messaging**  
Deliver and receive messages and notifications reliably across cloud and device. [More info](#)
- ▶  **Authentication**  
Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)
- ▶  **Realtime Database**  
Store and sync data in realtime across all connected clients. [More info](#)
- ▶  **Storage**  
Store and retrieve large files like images, audio, and video without writing server-side code. [More info](#)
- ▶  **Remote Config**  
Customize and experiment with app behavior using cloud-based configuration parameters. [More info](#)
- ▶  **Test Lab**  
Test your apps against a wide range of physical devices hosted in Google's cloud. [More info](#)
- ▶  **Crash Reporting**  
Get actionable insights and reports on app crashes, ANRs or other errors. [More info](#)
- ▶  **App Indexing**  
Get your app content into Google Search. [More info](#)
- ▶  **Dynamic Links**  
Create web URLs that can be shared to drive app installs and deep-linked into relevant content of your app. [More info](#)

Em seguida aparecerá o painel lateral do *Assistente*, selecione **Authetication**

Assistant  



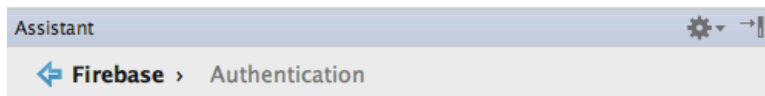
Firebase gives you the tools and infrastructure from Google to help you develop, grow and earn money from your app. [Learn more](#)

- ▶  **Analytics**  
Measure user activity and engagement with free, easy, and unlimited analytics. [More info](#)
- ▶  **Cloud Messaging**  
Deliver and receive messages and notifications reliably across cloud and device. [More info](#)
- ▼  **Authentication**  
Sign in and manage users with ease, accepting emails, Google Sign-In, Facebook and other login providers. [More info](#)
  - ▶ [Email and password authentication](#)

E clique no link **Email and password authentication**



# Configurando o Ambiente



## Email and password authentication

You can use Firebase Authentication to let your users sign in with their email addresses and passwords, and to manage your app's password-based accounts. This tutorial helps you set up an email and password system and then access information about the user.

[Launch in browser](#)

### 1 Connect your app to Firebase

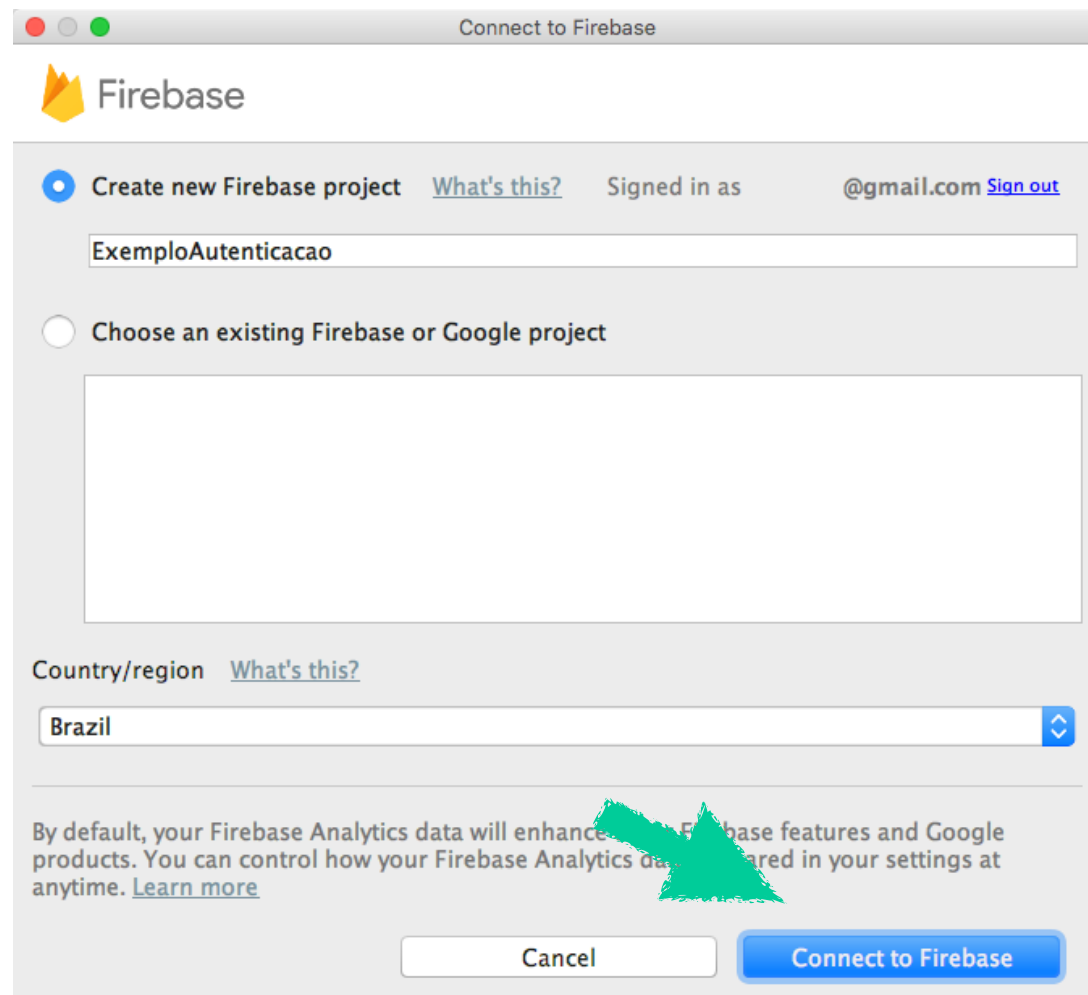
Connect to Firebase

### 2 Add Firebase Authentication to your app


Add Firebase Authentication to your app

To use an authentication provider, you need to enable it in the [Firebase console](#). Go to the Sign-in Method page in the Firebase Authentication section to enable Email/Password sign-in and any other identity providers you want for your app.

Clique no botão **Connect to Firebase** para apresentar o diálogo de conexão



Connect to Firebase

 Firebase

☒ Create new Firebase project [What's this?](#) Signed in as @gmail.com [Sign out](#)

ExemploAutenticacao

☐ Choose an existing Firebase or Google project

Country/region [What's this?](#)

Brazil

By default, your Firebase Analytics data will enhance your Firebase features and Google products. You can control how your Firebase Analytics data is shared in your settings at anytime. [Learn more](#)

Cancel Connect to Firebase

E em seguida clique no botão **Connect to Firebase** deste diálogo para conectar a aplicação ao Firebase

# Configurando o Ambiente

Assistant

← Firebase > Authentication

## Email and password authentication

You can use Firebase Authentication to let your users sign in with their email addresses and passwords, and to manage your app's password-based accounts. This tutorial helps you set up an email and password system and then access information about the user.

[Launch in browser](#)

- 1 Connect your app to Firebase
  - ✓ Connected
- 2 Add Firebase Authentication to your app

Add Firebase Authentication to your app

To use an authentication provider, you need to enable it in the [Firebase console](#). Go to the Sign-in Method page in the Firebase Authentication section to enable Email/Password sign-in and any other identity providers you want for your app.

Agora adicione a configuração do Firebase na aplicação clicando no botão **Add Firebase Authentication to your app**

E em seguida clique no botão **Connect to Firebase** deste diálogo para conectar a aplicação ao Firebase

Add Firebase Authentication to your app

Performing this action will make the following changes to your project.

**build.gradle (project-level)**

Add Firebase Gradle buildscript dependency  
classpath 'com.google.gms:google-services:3.1.1'

**app/build.gradle**

Add Firebase plugin for Gradle  
apply plugin: 'com.google.gms:google-services'

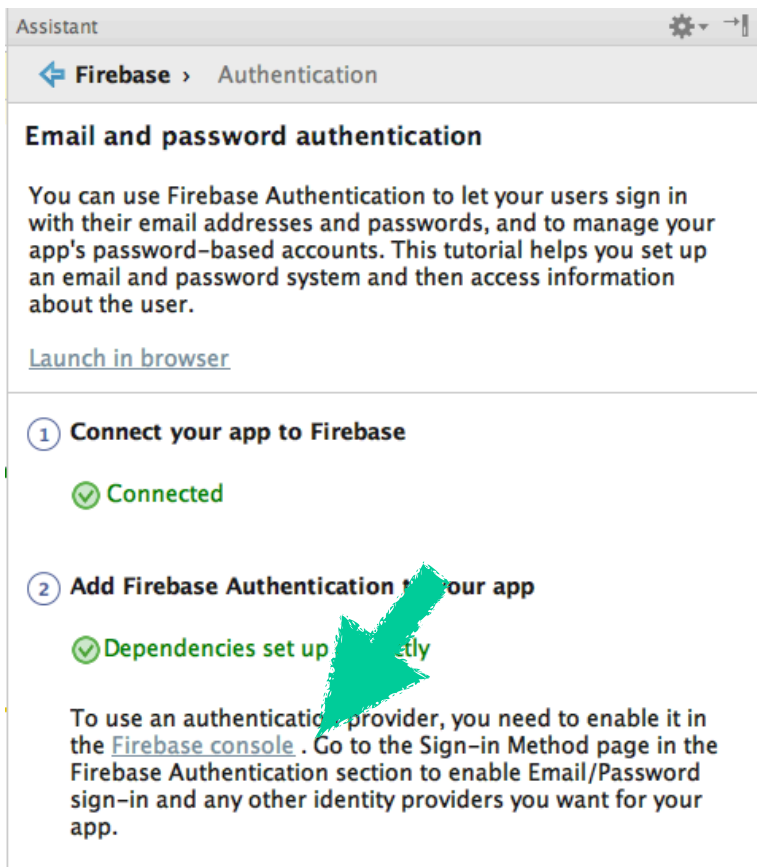
**build.gradle** will include these new dependencies:  
compile 'com.google.firebase:firebase-auth:11.6.0'

This will also enable the firebase-core library which includes Firebase Analytics [learn more](#)

Cancel Accept Changes

# Configurando o Ambiente

Após a configuração da aplicação  
clique no link **Firestore console**



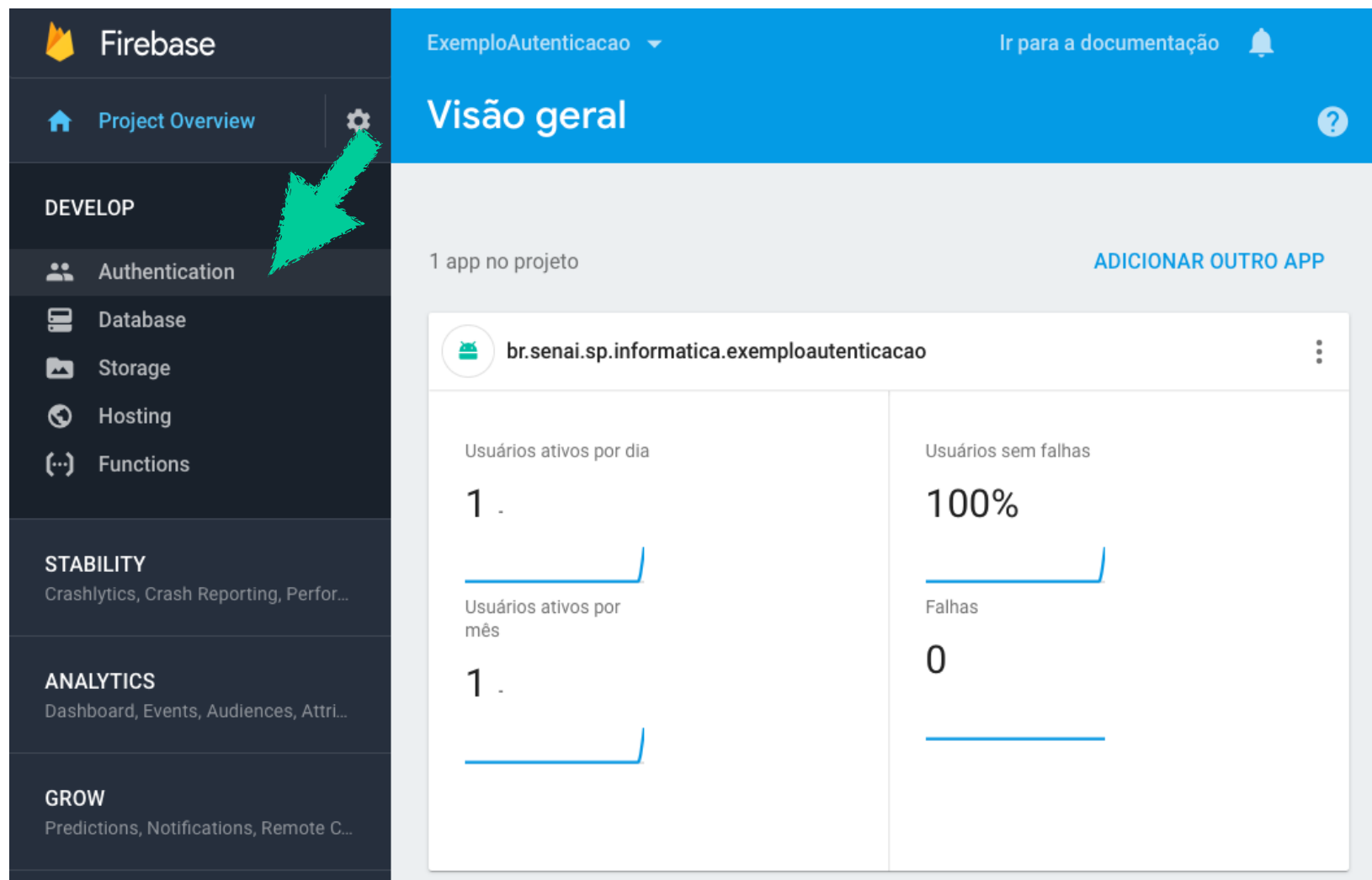
Na console do Firestore  
selecione a sua Aplicação





# Configurando o Ambiente

Na console do Firebase selecione **Authentication** na coluna a esquerda da tela.



The screenshot shows the Firebase console interface. On the left sidebar, the 'Authentication' option is highlighted with a green arrow. The main content area displays the 'Visão geral' (Overview) page for the project 'ExemploAutenticacao'. It shows a summary of the app and key authentication metrics.

**Project Overview**

ExemploAutenticacao ▾ Ir para a documentação 🔔

## Visão geral ?

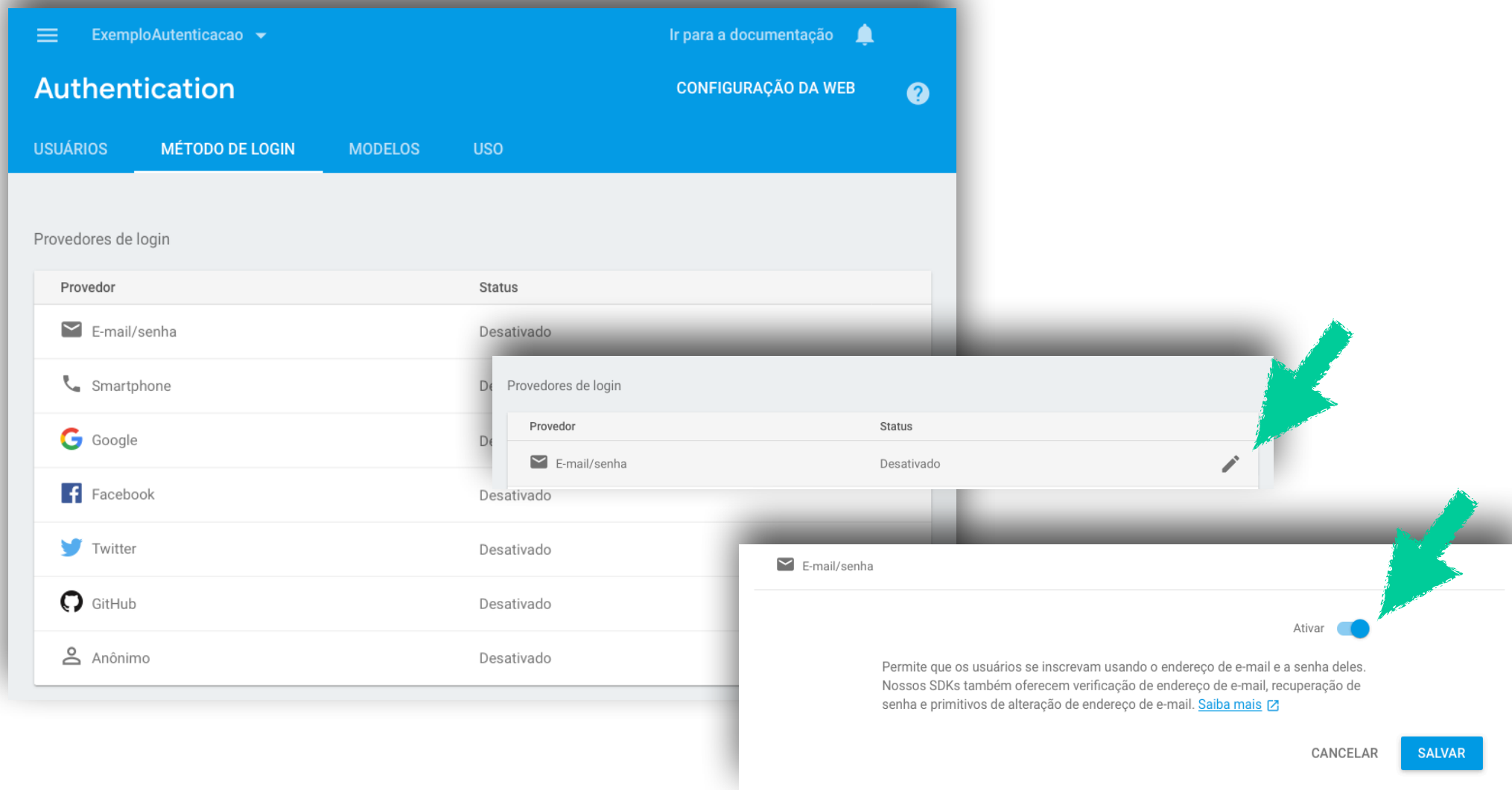
1 app no projeto [ADICIONAR OUTRO APP](#)

**br.senai.sp.informatica.exemploautenticacao** ⋮

Metric	Value
Usuários ativos por dia	1
Usuários ativos por mês	1
Usuários sem falhas	100%
Falhas	0

# Configurando o Ambiente

Habilite o provedor **E-mail/senha** nos *métodos de login*



ExemploAutenticacao ▾ Ir para a documentação 🔔

## Authentication

CONFIGURAÇÃO DA WEB ?

USUÁRIOS MÉTODO DE LOGIN MODELOS USO

Provedores de login

Provedor	Status
✉ E-mail/senha	Desativado
☎ Smartphone	Desativado
🌐 Google	Desativado
📘 Facebook	Desativado
🐦 Twitter	Desativado
🏠 GitHub	Desativado
👤 Anônimo	Desativado

Provedores de login

Provedor	Status
✉ E-mail/senha	Desativado

E-mail/senha

Ativar ☒

Permite que os usuários se inscrevam usando o endereço de e-mail e a senha deles. Nossos SDKs também oferecem verificação de endereço de e-mail, recuperação de senha e primitivos de alteração de endereço de e-mail. [Saiba mais](#)

CANCELAR SALVAR

# A Aplicação

Para demonstrar como autenticar foi criada uma aplicação simples que recebe o **email** e a **senha**, caso tenha sucesso na autenticação o **ID** do usuário será apresentado na tela.

O cadastro de uma nova conta utiliza os mesmos campos e deve ser selecionado o botão **Nova Conta**.

Para o teste de login deve ser selecionado o botão **Login**.

Se a aplicação for finalizada sem ser utilizado o **Logout** ao abrir novamente a aplicação, esta apresentará o **ID**



The screenshot displays a mobile application interface for authentication. At the top, a blue header bar contains the title "Exemplo de Autenticação" and status icons for signal, battery, and time (8:00). Below the header, there are two input fields: "E-Mail" and "Senha". At the bottom of the screen, there are three buttons: "LOGIN", "LOGOUT", and "NOVA CONTA".

No método **onCreate** deveremos inicializar o recurso de autenticação do *Firebase*

```
private FirebaseAuth mAuth;  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    edEmail = findViewById(R.id.edEmail);  
    edSenha = findViewById(R.id.edSenha);  
    edMensagem = findViewById(R.id.edMensagem);  
  
    mAuth = FirebaseAuth.getInstance();  
}
```



O botão **NOVA CONTA** permite demonstrar como criar uma nova conta de acesso no *Firebase*.

```
public void novoLoginClick(View view) {  
    String email = edEmail.getText().toString();  
    String senha = edSenha.getText().toString();  
  
    showProgressDialog();  
  
    mAuth.createUserWithEmailAndPassword(email, senha)  
        .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {  
            @Override  
            public void onComplete(@NonNull Task<AuthResult> task) {  
                if (task.isSuccessful()) {  
                    FirebaseUser user = mAuth.getCurrentUser();  
                    updateUI(user);  
                } else {  
                    Toast.makeText( context: MainActivity.this, text: "Falha na Autenticação.",  
                                   Toast.LENGTH_SHORT).show();  
                    updateUI( currentUser: null);  
                }  
  
                hideProgressDialog();  
            }  
        });  
}
```

O botão **LOGIN** permite demonstrar como conectar a uma conta existente no *Firebase*.

```
public void loginClick(View view) {  
    String email = edEmail.getText().toString();  
    String senha = edSenha.getText().toString();  
  
    showProgressDialog();  
  
    mAuth.signInWithEmailAndPassword(email, senha)  
        .addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {  
            @Override  
            public void onComplete(@NonNull Task<AuthResult> task) {  
                if (task.isSuccessful()) {  
                    FirebaseUser user = mAuth.getCurrentUser();  
                    updateUI(user);  
                } else {  
                    Toast.makeText( context: MainActivity.this, text: "Authentication failed.",  
                                   Toast.LENGTH_SHORT).show();  
                    updateUI( currentUser: null);  
                }  
  
                hideProgressDialog();  
            }  
        });  
}
```

O botão **LOGOUT** permite demonstrar como desconectar de uma conta do *Firebase*.

```
public void logoutClick(View view) {  
    mAuth.signOut();  
    updateUser( currentUser: null);  
}
```

Se a aplicação for finalizada sem que seja executado o método **signOut()** na próxima vez que a aplicação for carregada, a autenticação não será necessária.



Exemplo de Autenticação

E-Mail

---

Senha

---

---

LOGINLOGOUTNOVA CONTA

FIM