

Técnico de Informática

Módulo 5

Desenvolvimento de App Mobile

Aula 12 / 15

Desenvolvimento de App Mobile

Novo projeto

Ioniccreator – form – android e IOS - firebase.

- Anexo o final do slide pg 58

Antes, vamos acessar o GitHub....

Preparar o diretório para envio do arquivo posteriormente.

Técnico de Informática

Desenvolvimento de App Mobile



ionic signup - plataforma do Ionic

Ionic link – para enviar os arquivos para o GitHub confirme a opção de conectar com a branch “master” de seu repositório em “Link to master branch only”. Pressione “Enter”.

Para fazer o primeiro commit, continue na pasta raiz de seu projeto. Lá na janela shell, adicione os arquivos desejados para o commit, digitando o comando abaixo:

git add — all

Seus arquivos foram adicionados ao pacote para fazer o commit. Agora digite o comando abaixo:

git commit -m “sua mensagem de commit”

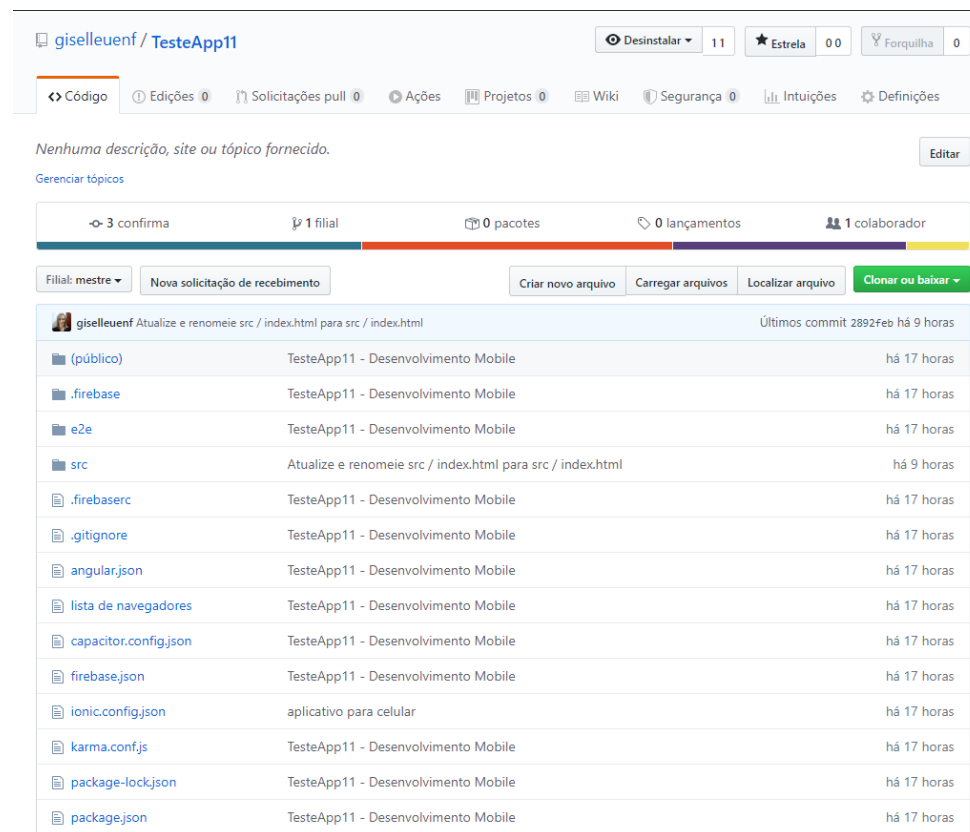
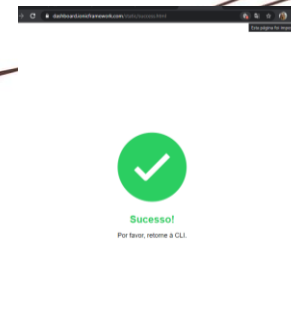
E finalmente o comando que envia o pacote comitado para o repositório remoto:

git push origin

ou se for a primeira vez, e der erro, digite os comandos:

git remote add origin <url do seu app>

git push — set-upstream origin master — force

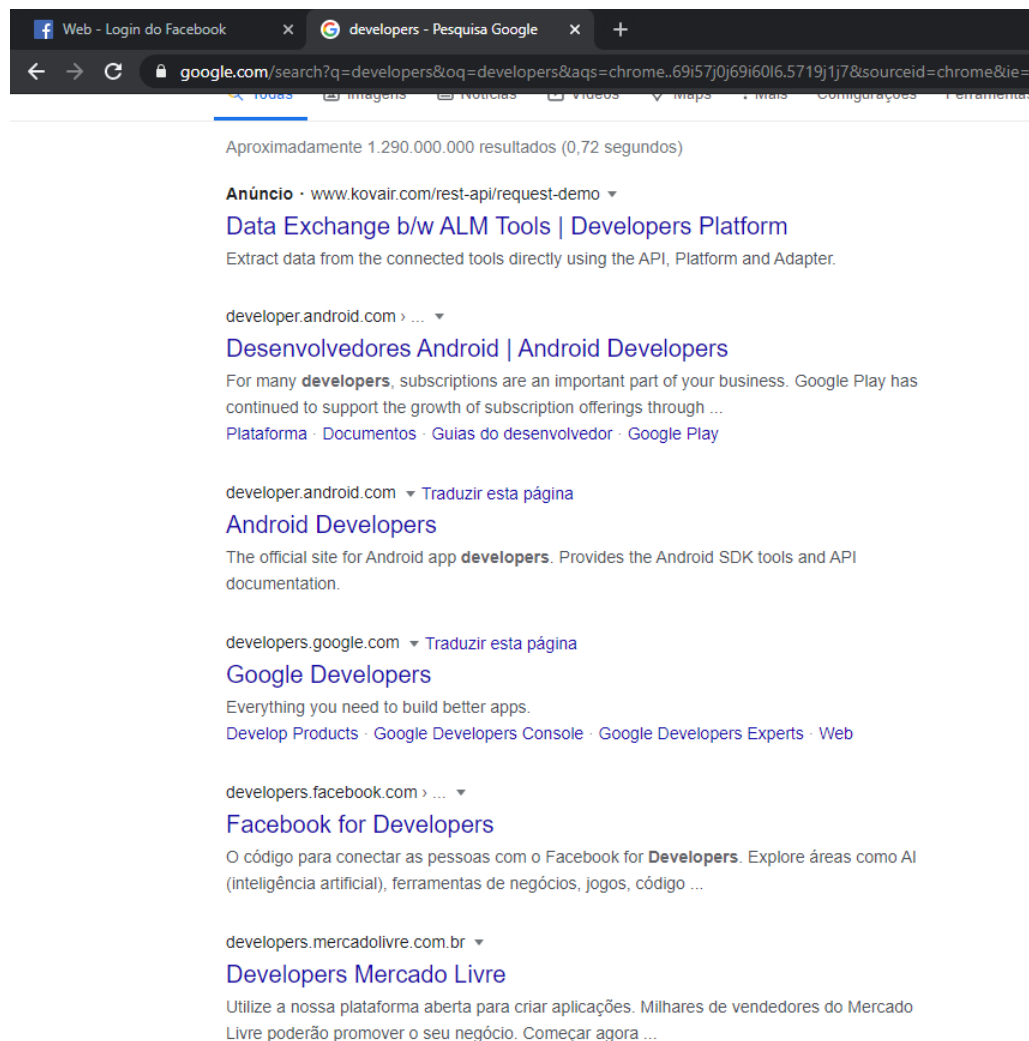


Técnico de Informática

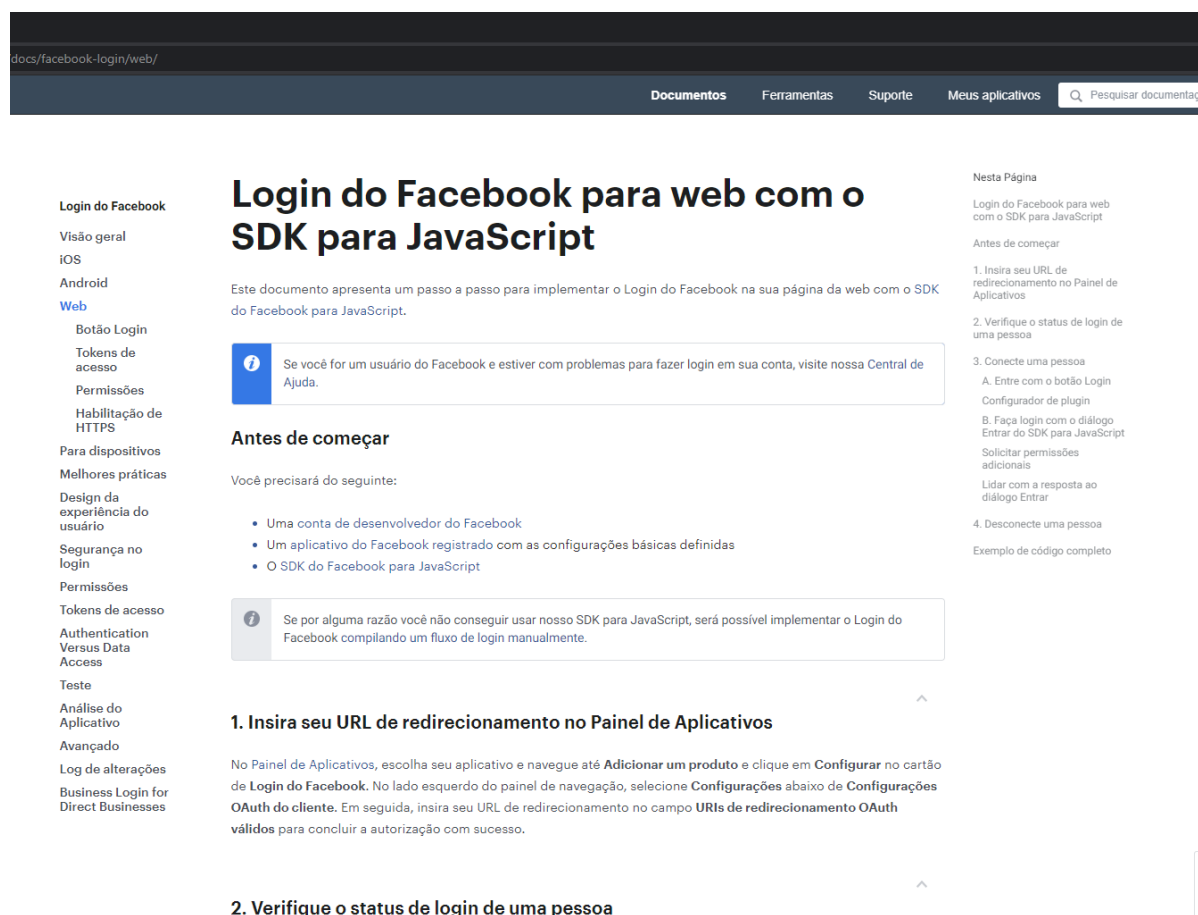
Desenvolvimento de App Mobile



Geralmente as plataformas tem
área de acesso aos
Desenvolvedores



<https://developers.facebook.com/docs/facebook-login/web/>



The screenshot shows the Facebook Login for Web SDK documentation page. The page has a dark header with the URL 'docs/facebook-login/web/' and navigation links: Documentos, Ferramentas, Suporte, Meus aplicativos, and a search bar. The main content area is titled 'Login do Facebook para web com o SDK para JavaScript'. It includes a sidebar with a table of contents, a main text area with a 'Antes de começar' section, and a right sidebar with a 'Nesta Página' section.

Table of Contents (Left Sidebar):

- Login do Facebook
- Visão geral
- iOS
- Android
- Web
 - Botão Login
 - Tokens de acesso
 - Permissões
 - Habilitação de HTTPS
- Para dispositivos
- Melhores práticas
- Design da experiência do usuário
- Segurança no login
- Permissões
- Tokens de acesso
- Authentication Versus Data Access
- Teste
- Análise do Aplicativo
- Avançado
- Log de alterações
- Business Login for Direct Businesses

Main Content:

Login do Facebook para web com o SDK para JavaScript

Este documento apresenta um passo a passo para implementar o Login do Facebook na sua página da web com o SDK do Facebook para JavaScript.

Antes de começar

Você precisará do seguinte:

- Uma conta de desenvolvedor do Facebook
- Um aplicativo do Facebook registrado com as configurações básicas definidas
- O SDK do Facebook para JavaScript

1. Insira seu URL de redirecionamento no Painel de Aplicativos

No Painel de Aplicativos, escolha seu aplicativo e navegue até **Adicionar um produto** e clique em **Configurar** no cartão de **Login do Facebook**. No lado esquerdo do painel de navegação, selecione **Configurações** abaixo de **Configurações OAuth do cliente**. Em seguida, insira seu URL de redirecionamento no campo **URIs de redirecionamento OAuth válidos** para concluir a autorização com sucesso.

2. Verifique o status de login de uma pessoa

Right Sidebar (Nesta Página):

- Login do Facebook para web com o SDK para JavaScript
- Antes de começar
- 1. Insira seu URL de redirecionamento no Painel de Aplicativos
- 2. Verifique o status de login de uma pessoa
- 3. Conecte uma pessoa
 - A. Entre com o botão Login Configurador de plugin
 - B. Faça login com o diálogo Entrar do SDK para JavaScript
 - Solicitar permissões adicionais
 - Lidar com a resposta ao diálogo Entrar
- 4. Desconecte uma pessoa
- Exemplo de código completo

Códigos de
acesso ao FB

Técnico de Informática

Desenvolvimento de App Mobile



← Detalhes de testeapp11-20191-s...



testeapp11-20191-senac

Ativo • Adição: 5 de mai de 2020

Informações que você está compartilhando com este aplicativo:

Nome e foto do perfil

OBRIGATÓRIO

Endereço de email

giselle@uenf.br



Salvar

Visibilidade do aplicativo

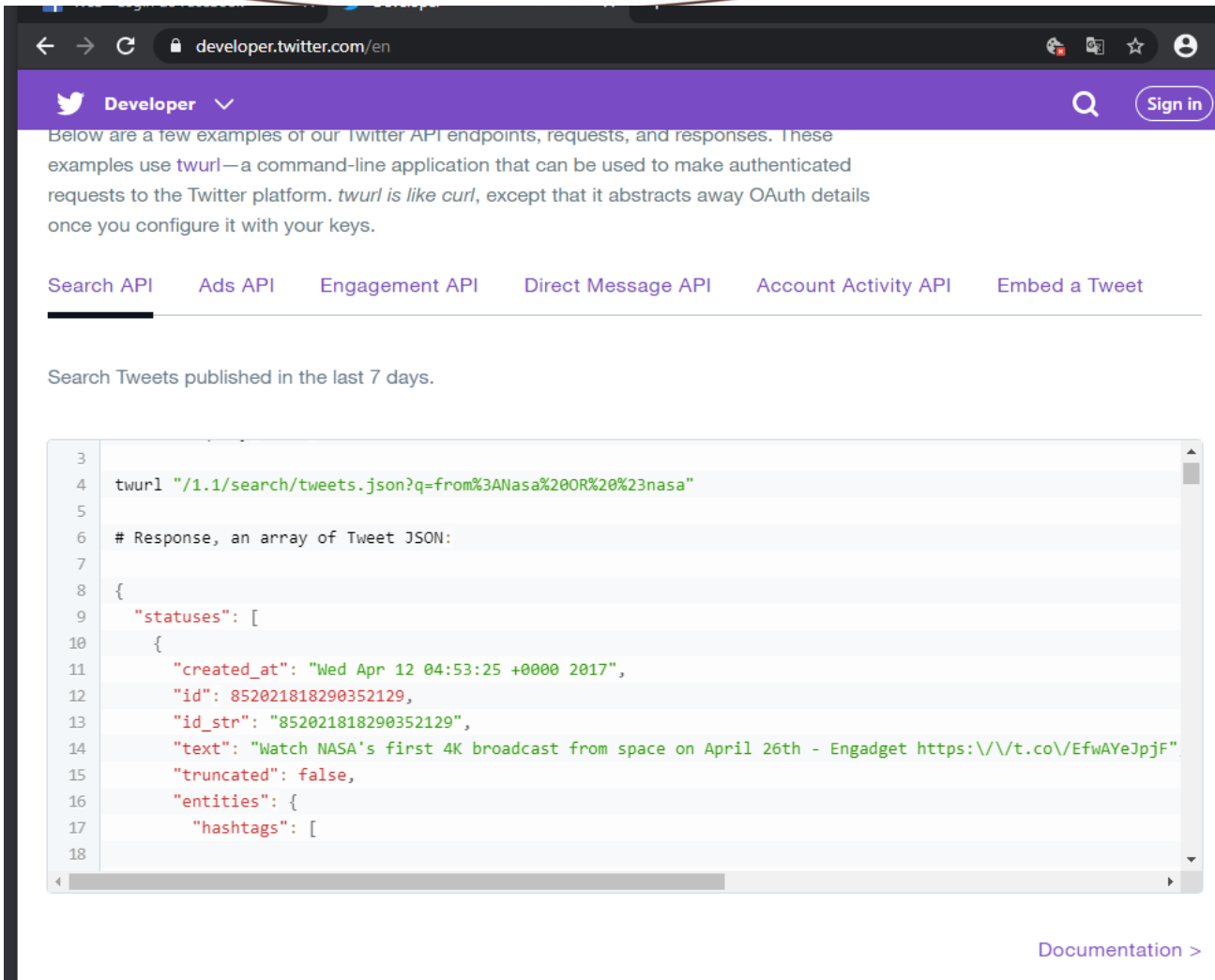
Somente eu >

Esta configuração controla quem no Facebook pode ver que você usa esse aplicativo.

Saiba mais:

Política de Privacidade





The screenshot shows the Twitter Developer API documentation page. The browser address bar displays `developer.twitter.com/en`. The page header includes the Twitter logo, the word "Developer", a search icon, and a "Sign in" button. The main content area begins with an introductory paragraph about API endpoints and the use of `twurl`. Below this is a horizontal navigation menu with links for "Search API", "Ads API", "Engagement API", "Direct Message API", "Account Activity API", and "Embed a Tweet". The "Search API" link is currently selected. The page content shows a search for tweets published in the last 7 days. A code block displays a `twurl` command and a sample JSON response for a search query. The JSON response includes fields for `created_at`, `id`, `id_str`, `text`, `truncated`, `entities`, and `hashtags`. At the bottom right of the page, there is a link labeled "Documentation >".

```
3  
4 twurl "/1.1/search/tweets.json?q=from%3ANasa%20OR%20%23nasa"  
5  
6 # Response, an array of Tweet JSON:  
7  
8 {  
9   "statuses": [  
10    {  
11      "created_at": "Wed Apr 12 04:53:25 +0000 2017",  
12      "id": 852021818290352129,  
13      "id_str": "852021818290352129",  
14      "text": "Watch NASA's first 4K broadcast from space on April 26th - Engadget https://t.co/EfwAYeJpJf",  
15      "truncated": false,  
16      "entities": {  
17        "hashtags": [  
18
```

Documentation >

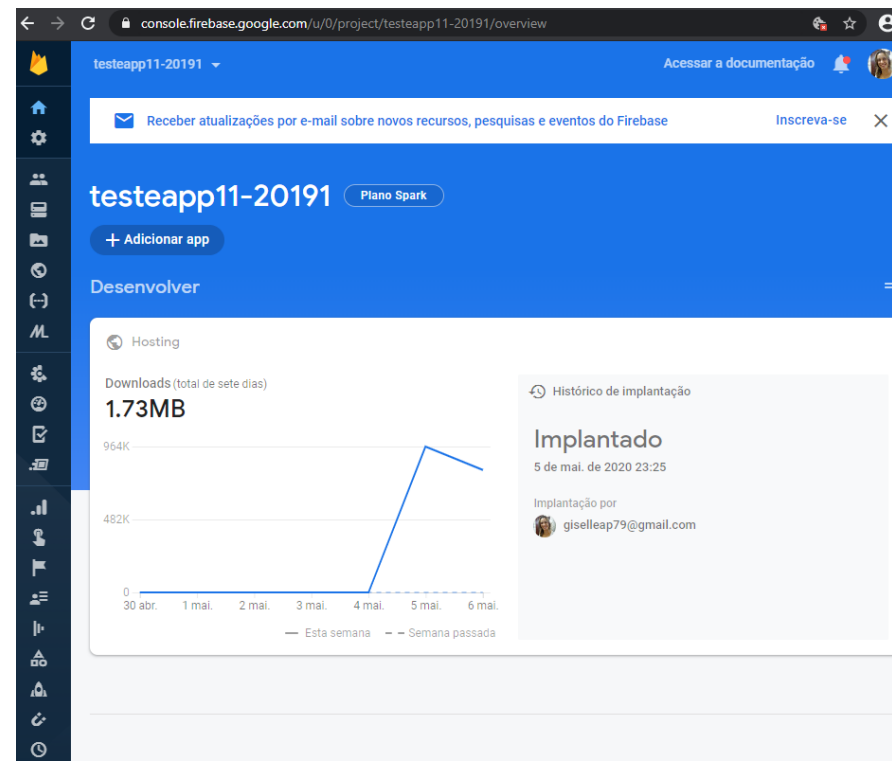
Twitter

Nessas áreas direcionadas ao Desenvolvedores, buscamos os códigos de acesso para integração dos nossos apps com a plataforma.

Autenticação Firebase

A plataforma Firebase possui um [plugin](#) completo [para autenticação](#) de usuários que tira praticamente todo o trabalho pesado do seu aplicativo. Acesse o [console do Firebase](#), abra seu projeto e clique no botão “+ Adicionar aplicativo”:

E selecione então a opção “Web”:




× Adicionar o Firebase ao seu app da Web

1 Registrar app

Apelido do app ?

TesteApp11

☐ Também configure o **Firebase Hosting** para este app. [Saiba mais](#) 

A configuração do Hosting também pode ser feita depois. Comece a usar a qualquer momento sem pagar nada.

Registrar app

2 Adicionar SDK do Firebase

Acessar a do

× Adicionar o Firebase ao seu app da Web

- ✓ Registrar app
- 2 Adicionar SDK do Firebase

Copie e cole esses scripts na parte inferior da tag <body>, mas antes de usar qualquer serviço do Firebase:

```
<!-- The core Firebase JS SDK is always required and must be listed first -->
<script src="https://www.gstatic.com/firebasejs/7.14.3/firebase-app.js"></scrip

<!-- TODO: Add SDKs for Firebase products that you want to use
https://firebase.google.com/docs/web/setup#available-libraries -->

<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "AIza...",
    authDomain: "testeapp11-20191.firebaseio.com",
    databaseURL: "https://testeapp11-20191.firebaseio.com",
    projectId: "testeapp11-20191",
    storageBucket: "testeapp11-20191.appspot.com",
    messagingSenderId: "...518",
    appId: "1:7...917"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

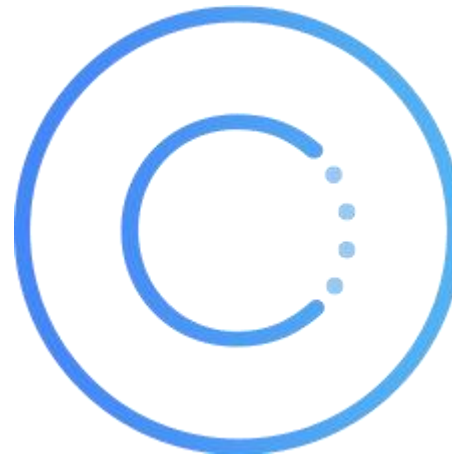
Saiba mais sobre o Firebase para Web: [Primeiros passos](#), [Referência da API Web SDK](#), [Amostras](#)

Continuar no console

Criando o formulário de contato para o projeto de hoje.

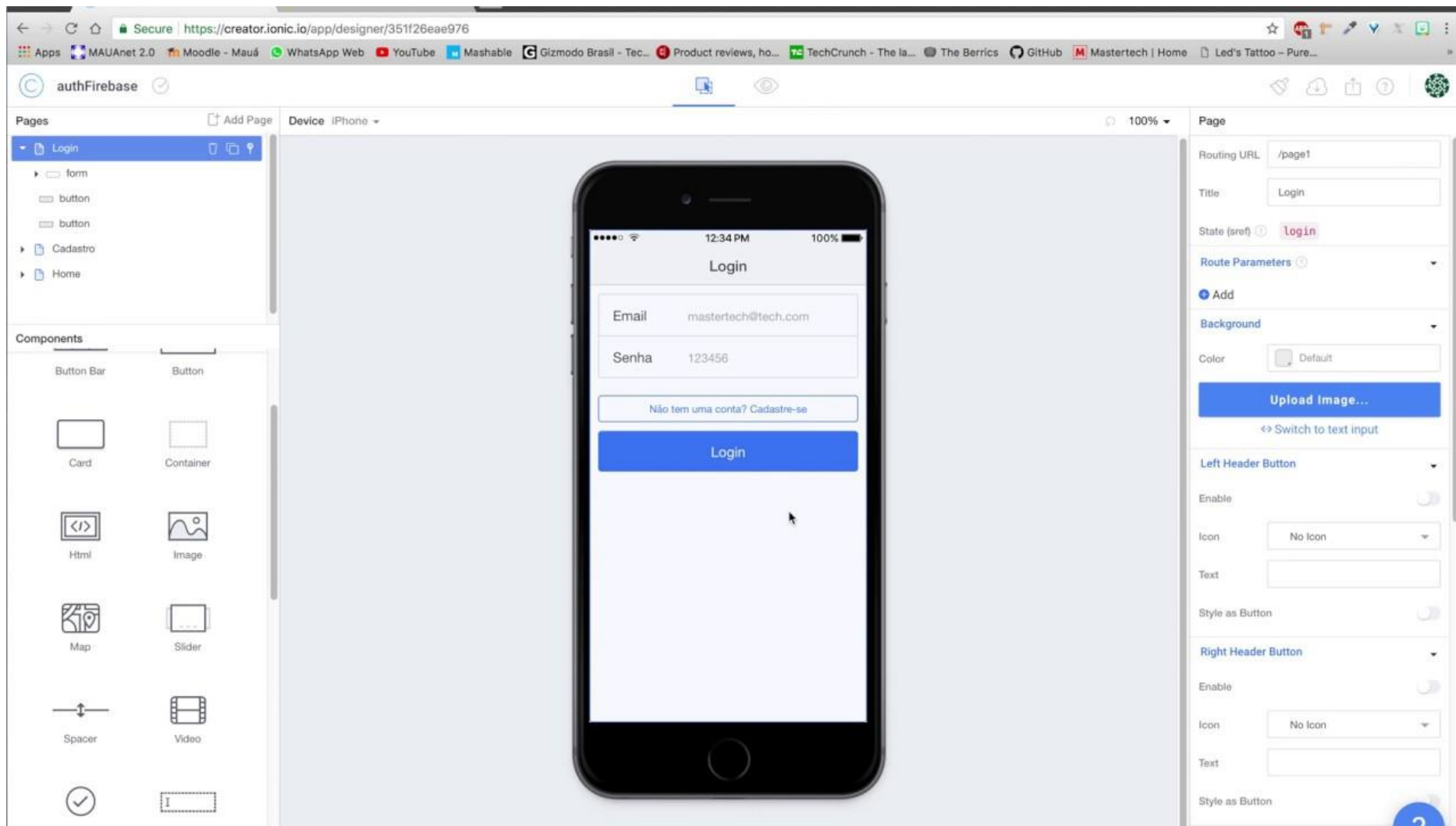
Ionic Creator

<https://creator.ionic.io>



Técnico de Informática

Desenvolvimento de App Mobile



Técnico de Informática

Desenvolvimento de App Mobile



project

creator [Guias](#) [Vídeos tutoriais](#) [Apoio, suporte](#) [Conecte-se](#)

[Guias](#) > [ZIP Exportar um Projeto Ionic](#)

GUIAS DE RECURSOS

- Navegação e links especiais
- Temas e SCSS
- Edição de código personalizado >
- Exportando seu aplicativo >
 - ZIP Exportar um Projeto Ionic**
- Grupos e colaboração

COMPONENTES

- Páginas e cabeçalhos >
- Titulos
- Parágrafos
- Botões e barras de botões
- Imagens
- Vídeos
- HTML
- Cartas
- Recipientes
- Slider
- Formulários >
- Listas >
- Google Maps

Distribuído por [readme](#)

ZIP Exportar um Projeto Ionic

[SUGERIR EDIÇÕES](#)

Cada projeto de aplicativo no Creator funciona como um aplicativo Ionic normal nos bastidores. Exportar o código bruto para o seu projeto é fácil e você pode trabalhar nele localmente com a CLI.

Importante
Exportar seu projeto como um ZIP funciona apenas de uma maneira. Não há como importar um projeto de volta para o Creator.

Observação: você precisará do Ionic CLI local instalado para poder trabalhar em seu projeto fora do Creator.

1. Exporte seu arquivo ZIP

Clique no ícone de exportação no canto superior direito do Criador para abrir o pop-up modal Exportar.

	Page / List / List Item
Link	
Type	

Exportar para o PC para gerar outras plataformas

→ Somente a versão paga do IonicCreator gera as plataformas pela interface gráfica e os arquivos de instalação.

2b. (Ionic v3.x) Configurar um projeto Ionic local

⚠ Apenas se aplica a projetos Ionic v3.x Creator

Você só pode exportar para um projeto Ionic v3.x se escolher Ionic v3.x ao criar o projeto. Não há como exportar um projeto do Ionic 1 para o Ionic 3.

Esta etapa requer que você tenha a versão mais recente do [Ionic CLI local instalada](#).

Para iniciar um novo projeto Ionic, digite o comando:

```
ionic start myApp blank --type=ionic-angular
```

Agora vá para o diretório recém-criado e você verá o diretório chamado dentro dele. `myApp src`

Copie e cole o conteúdo de sua exportação zip no diretório. Você desejará substituir o diretório, diretório e `src app pages index.html`

3. [Opcional] Instale os plug-ins Cordova (apenas v1)

Se você estiver usando plug-ins Cordova em seu aplicativo, eles precisarão ser instalados em seu projeto local. Para cada plugin Cordova que você estava usando, digite o comando:

```
ionic plugin add [cordova-plugin-name]
```

4. Visualize seu aplicativo

Agora você exportou com sucesso seu aplicativo e configurou-o como um projeto Ionic local. Para visualizar seu aplicativo, digite o comando:

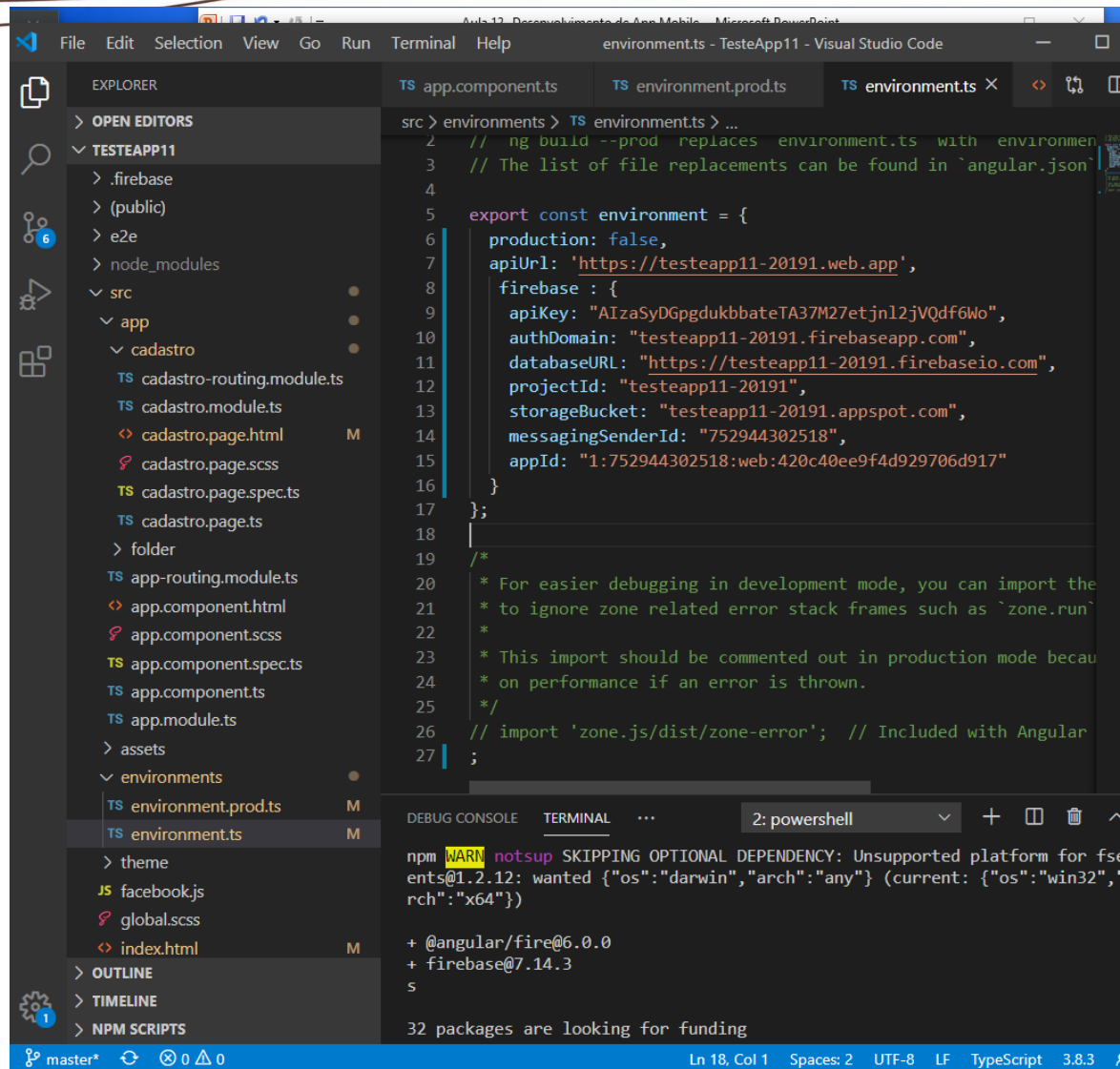
Etapas para concluir a exportação.

Conectando seu App ao Firebase pelo Ionic

<https://github.com/angular/angularfire/blob/91ec37eed9c33ed5bd084ebb016a4bb70916bf3f/docs/ionic/v3.md>

<https://github.com/angular/angularfire/>

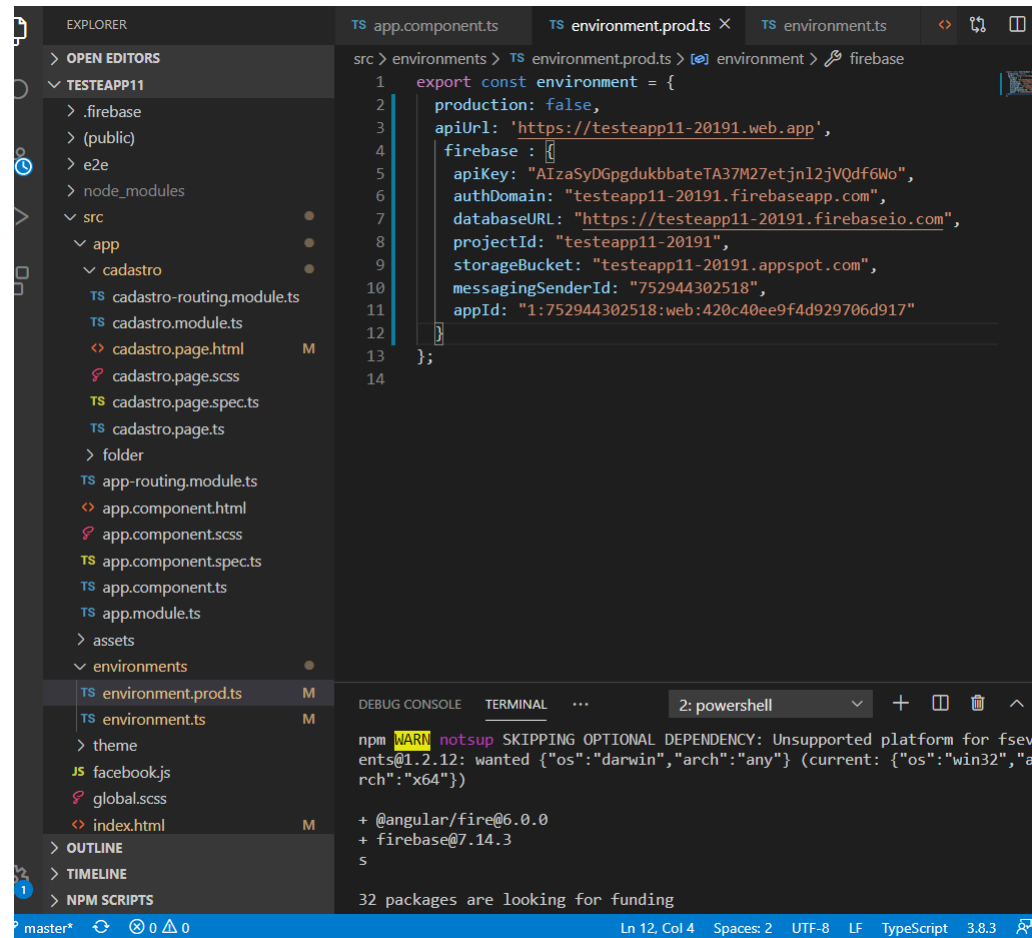
Lembra do arquivo `“/environments/environment.ts”` que guarda como variáveis dos nossos diferentes ambientes de código? Então, vamos colar esses valores de configuração do Firebase:



```
src > environments > TS environments.ts > ...
2 // ng build --prod replaces environment.ts with environment.prod.ts
3 // The list of file replacements can be found in `angular.json`
4
5 export const environment = {
6   production: false,
7   apiUrl: 'https://testeapp11-20191.web.app',
8   firebase: {
9     apiKey: "AIzaSyDGpgdukbbateTA37M2etjnl2jVQdf6Wo",
10    authDomain: "testeapp11-20191.firebaseio.com",
11    databaseURL: "https://testeapp11-20191.firebaseio.com",
12    projectId: "testeapp11-20191",
13    storageBucket: "testeapp11-20191.appspot.com",
14    messagingSenderId: "752944302518",
15    appId: "1:752944302518:web:420c40ee9f4d929706d917"
16  }
17 };
18
19 /*
20  * For easier debugging in development mode, you can import the
21  * to ignore zone related error stack frames such as `zone.run`
22  *
23  * This import should be commented out in production mode because
24  * on performance if an error is thrown.
25  */
26 // import 'zone.js/dist/zone-error'; // Included with Angular CLI
27 ;
```

```
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fs-ents@1.2.12: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
+ @angular/fire@6.0.0
+ firebase@7.14.3
s
32 packages are looking for funding
```

E não se esqueça de fazer a mesma coisa com o arquivo de ambiente de produção “*environments/environment.prod.ts*” :



The screenshot shows the Visual Studio Code interface. On the left, the Explorer pane displays the file structure of a project, with the `environment.prod.ts` file selected under the `environments` folder. The main editor area shows the content of `environment.prod.ts`, which defines the production environment configuration for a Firebase application. The code includes fields for `production`, `apiKey`, `authDomain`, `databaseURL`, `projectId`, `storageBucket`, `messagingSenderId`, and `appId`. The bottom panel shows the Terminal with the output of the `npm install` command, indicating that the dependencies are being installed successfully.

```
src > environments > TS environment.prod.ts > environment > firebase
1  export const environment = {
2    production: false,
3    apiUrl: 'https://testeapp11-20191.web.app',
4    firebase: {
5      apiKey: "AIzaSyDGpgdukbateTA37M27etjn12jVQdf6Wo",
6      authDomain: "testeapp11-20191.firebaseio.com",
7      databaseURL: "https://testeapp11-20191.firebaseio.com",
8      projectId: "testeapp11-20191",
9      storageBucket: "testeapp11-20191.appspot.com",
10     messagingSenderId: "752944302518",
11     appId: "1:752944302518:web:420c40ee9f4d929706d917"
12   }
13 };
14
```

DEBUG CONSOLE TERMINAL ... 2: powershell

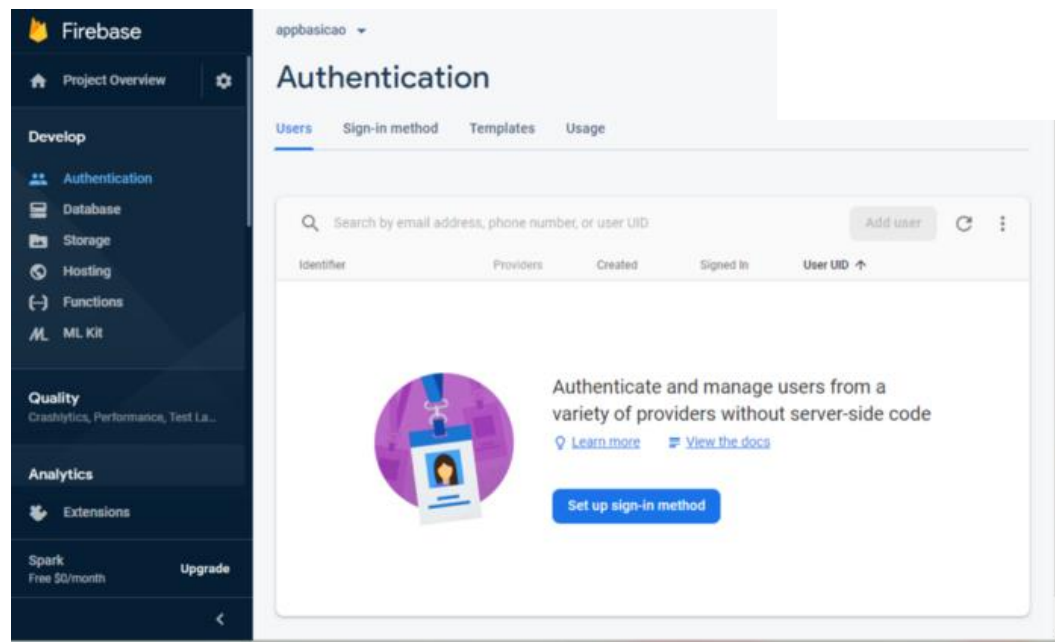
```
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.12: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
+ @angular/fire@6.0.0
+ firebase@7.14.3
32 packages are looking for funding
```

Métodos de autenticação

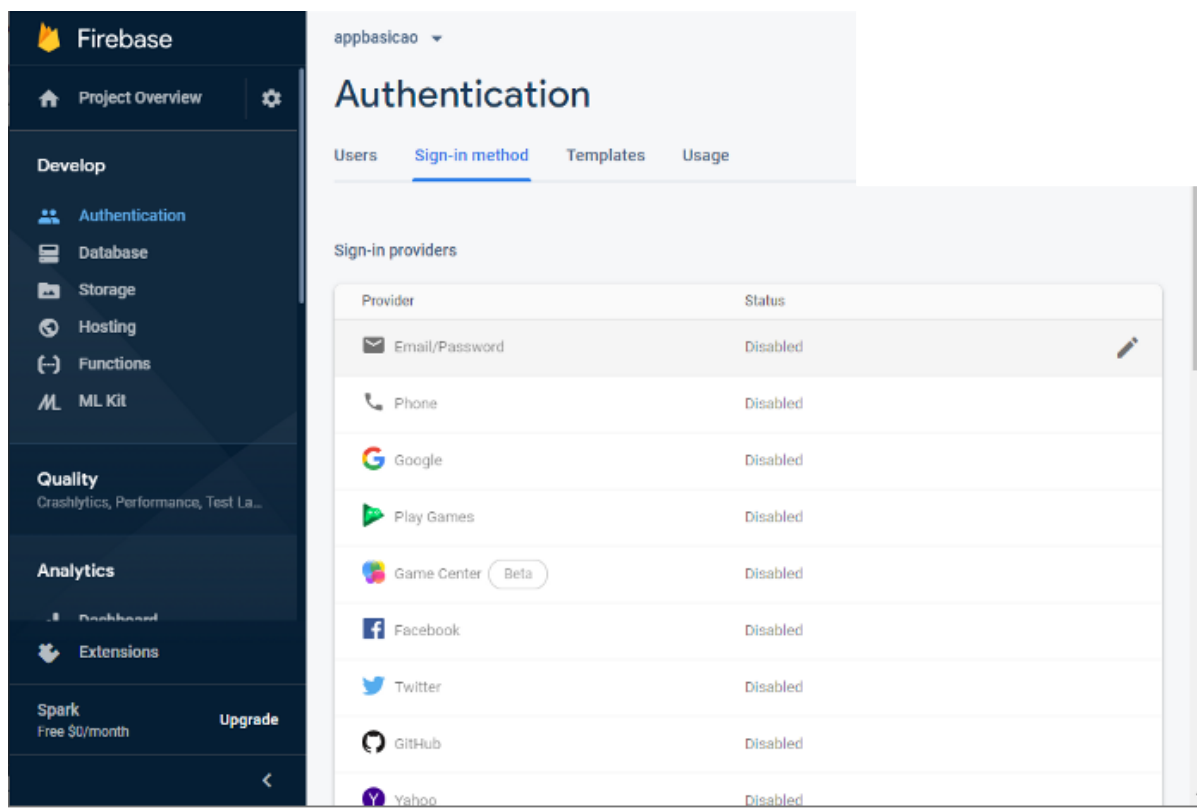
O Firebase possui vários métodos de autenticação para você escolher e colocar no seu aplicativo.

Neste tutorial, vamos usar uma opção de email e senha.

Clique no item do menu “*Autenticação*”, no console do Firebase de seu projeto:



Na página de autenticação, selecione o “*Método de login*” e habilite a opção “*Email / Senha*” :




The screenshot shows the Firebase Authentication console for a project named 'appbasicao'. The 'Sign-in method' tab is selected, displaying a list of providers and their status. The 'Email/Password' provider is highlighted, and its status is 'Disabled'.

Provider	Status
Email/Password	Disabled
Phone	Disabled
Google	Disabled
Play Games	Disabled
Game Center (Beta)	Disabled
Facebook	Disabled
Twitter	Disabled
GitHub	Disabled
Yahoo	Disabled


Técnico de Informática

Desenvolvimento de App Mobile



 Email/Password

☒ Enable

Allow users to sign up using their email address and password. Our SDKs also provide email address verification, password recovery, and email address change primitives. [Learn more](#) 

Email link (passwordless sign-in) ☐ Enable

Cancel **Save**

Firestore no Ionic

Para utilizar o Firestore no Ionic, instale o módulo necessário com o comando npm.

Digite o comando abaixo na sua janela Shell:

npm install @angular/fire firebase --save

Inicialização do Firestore

Depois de instalado ou o módulo do Firestore no Ionic, é necessário inicializá-lo dentro do arquivo “*app.module.ts*” .

Importe o módulo “*@ angular / fire / auth*” que você acabou de instalar e depois inicialize o módulo com o comando abaixo:

firebase.initializeApp (environment.firestore);

Desenvolvimento de App Mobile



```

47 this.selectedIndex = this.currentPage.findIndex(page => page.id === id);
DEBUG CONSOLE  TERMINAL  ...  2: powershell  +  [ ]  [X]  ^  X
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.12 (node_modules\karma\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.12: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.3 (node_modules\s\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.3: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.12 (node_modules\@angular\compiler-cli\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.12: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
+ @angular/fire@6.0.0
+ firebase@7.14.3
added 58 packages from 60 contributors and audited 20386 packages in 178.52s
32 packages are looking for funding
  run `npm fund` for details
found 3 low severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details
PS C:\Users\UENF\TesteApp11>

```

E então, fazemos o login no Firebase. Digite o comando abaixo:

firebase login

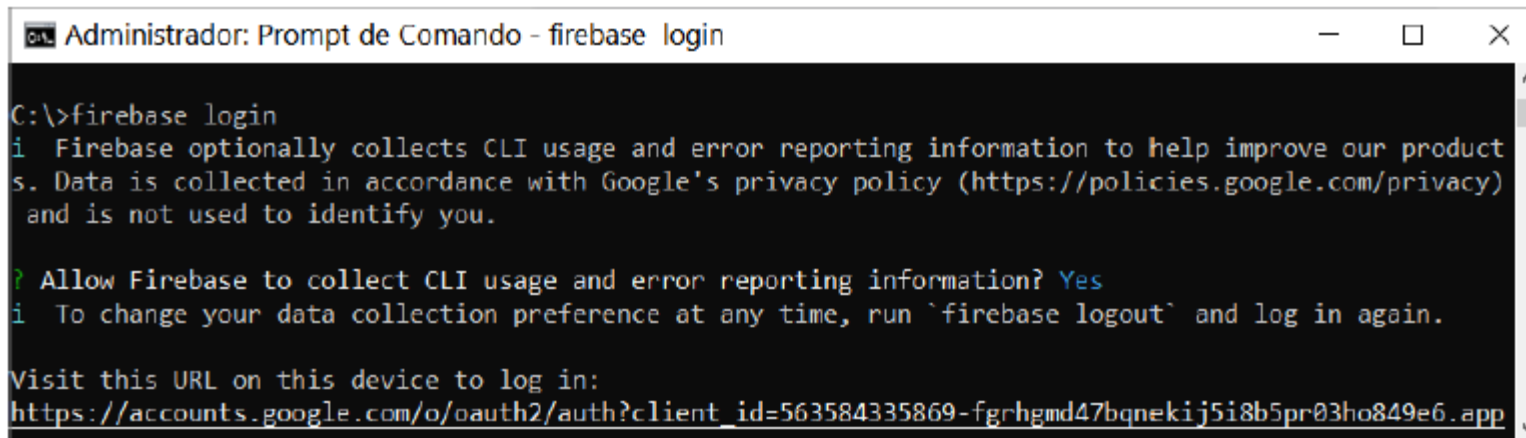
Faça a autenticação requisitada e aguarde a mensagem de sucesso.

Depois iniciemos o Firebase com o comando:

firebase init

Pressione Y, para confirmar que quer proceder com a inicialização e depois selecione a opção “Hosting”, usando a setinha para baixo e depois apertando a tecla de espaço, para hospedar seu app na plataforma Firebase.

Na próxima opção selecione para criar um novo projeto no Firebase:



```
Administrador: Prompt de Comando - firebase login

C:\>firebase login
i  Firebase optionally collects CLI usage and error reporting information to help improve our products. 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 usage and error reporting information? Yes
i  To change your data collection preference at any time, run `firebase logout` and log in again.

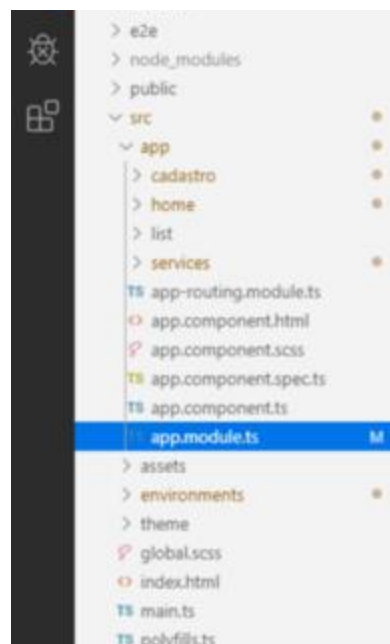
Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client_id=563584335869-fgrhgmd47bqnekij5i8b5pr03ho849e6.app
```

```

> node-pre-gyp install --fallback-to-build --library=static_library
node-pre-gyp WARN Using request for node-pre-gyp https download
[grpc] Success:
2-x64-unknown\grpc_node.node" is installed via remote

> core-js@3.3.6 postinstall
core-js
> node postinstall || echo "ignore"

```



Terminal Help app.module.ts - Untitled (Workspace) - Visual Studio Code

TS app.module.ts X

```

> src > app > TS app.module.ts > AppModule
7 import { StatusBar } from '@ionic-native/status-bar/ngx';
8
9 import { AppComponent } from './app.component';
10 import { AppRoutingModuleModule } from './app-routing.module';
11 import { HttpClientModule } from '@angular/common/http';
12
13 import { environment } from 'src/environments/environment';
14 import { AuthenticationService } from './services/authentication/authentication.service';
15 import { AngularFireAuthModule } from '@angular/fire/auth';
16
17 import * as firebase from 'firebase';
18 firebase.initializeApp(environment.firebase);
19
20
21 @NgModule({
22   declarations: [AppComponent],
23   entryComponents: [],
24   imports: [
25     BrowserModule,
26     IonicModule.forRoot(),
27     AppRoutingModuleModule,
28     HttpClientModule,
29     AngularFireAuthModule
30   ],
31   providers: [
32     StatusBar,
33     SplashScreen,
34     AuthenticationService,
35     { provide: RouteReuseStrategy, useClass: IonicRouteStrategy }
36   ]

```

```
C:\appbasicao>firebase init

#####  #####  #####  #####  #####  #####  #####
##      ##  ##      ##  ##      ##      ##  ##      ##
#####  ##  #####  #####  #####  #####  #####  #####
##      ##  ##      ##  ##      ##      ##  ##      ##
##      #####  ##  #####  #####  ##      ##  #####

You're about to initialize a Firebase project in this directory:

  C:\appbasicao

Before we get started, keep in mind:

  * You are currently outside your home directory

? Are you ready to proceed? Yes
? Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices.
  ( ) Database: Deploy Firebase Realtime Database Rules
  ( ) Firestore: Deploy rules and create indexes for Firestore
  ( ) Functions: Configure and deploy Cloud Functions
> (*) Hosting: Configure and deploy Firebase Hosting sites
  ( ) Storage: Deploy Cloud Storage security rules
```

Before we get started, keep in mind:

- * You are currently outside your home directory

? Are you ready to proceed? **Yes**

? Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices. **Hosting: Configure and deploy Firebase Hosting sites**

=== Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

? Please select an option:

Use an existing project

> **Create a new project**

Add Firebase to an existing Google Cloud Platform project

Don't set up a default project

Before we get started, keep in mind:

- * You are currently outside your home directory

? Are you ready to proceed? **Yes**

? Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices. **Hosting: Configure and deploy Firebase Hosting sites**

=== Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

? Please select an option: **Create a new project**

i If you want to create a project in a Google Cloud organization or folder, please use "firebase projects:create" instead, and return to this command when you've created the project.

? Please specify a unique project id (warning: cannot be modified afterward) [6-30 characters]:

() **testeapp11**

CA: Administrador: Prompt de Comando - firebase init

project.

? Please specify a unique project id (warning: cannot be modified afterward) [6-30 characters]:

testeapp11

? What would you like to call your project? (defaults to your project ID)

✓ Creating Google Cloud Platform project

✓ Adding Firebase resources to Google Cloud Platform project

=== Your Firebase project is ready! ===

Project information:

- Project ID: **testeapp11**

- Project Name: **testeapp11**

Firebase console is available at

[https://console.firebase.google.com/project/ testeapp11](https://console.firebase.google.com/project/testeapp11)

i Using project **testeapp11** (**testeapp11**)

=== Hosting Setup

```
Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with firebase deploy. If you have a build process for your assets, use your build's output directory.
```

```
? What do you want to use as your public directory? (public)
```



Quando requisitar o diretório público para o seu projeto, digite `www` e pressione "Enter". Usaremos esse diretório que foi gerado no build do Ionic.

Prompt de Comando - firebase init

=== Hosting Setup

```
Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with firebase deploy. If you have a build process for your assets, use your build's output directory.
```

```
? What do you want to use as your public directory? www
```

```
? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) _
```

E depois digite Y (Sim) para o app de página única:

Prompt de Comando - firebase init

=== Hosting Setup

```
Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with firebase deploy. If you have a build process for your assets, use your build's output directory.
```

```
? What do you want to use as your public directory? www
```

```
? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) y
```


Técnico de Informática

Desenvolvimento de App Mobile



E então **N** (Não) para subescrever o arquivo `index.html`, que foi gerado no build dentro da pasta `www`:

```
Prompt de Comando

? What do you want to use as your public directory? www
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? File www/index.html already exists. Overwrite? No
i Skipping write of www/index.html
i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...
+ Firebase initialization complete!
```

Deploy

Agora vamos fazer o deploy do seu app no projeto criado na plataforma Firebase. Digite a linha de comando abaixo na raíz de seu projeto:

firebase deploy

```
Prompt de Comando

C:\>firebase deploy

--- Deploying

i deploying hosting
i hosting|beginning deploy...
i hosting|found 904 files in www
+ hosting|file upload complete
i hosting|finalizing version...
+ hosting|version finalized
i hosting|releasing new version...
+ hosting|release complete
+ Deploy complete!
```

Serviço de autenticação

Para o código que efetua como chamadas para o plugin de autenticação do Firebase, vamos criar um serviço específico para isso.

Digite o comando abaixo na janela Shell:

ionic generate

```
? What would you like to generate?  
page  
component  
> service  
module  
class  
directive  
guard  
(Move up and down to reveal more choices)
```

E então selecione a opção *"service"* .
Digite então o nome do serviço *"/ services / authentication / authentication"* e aperte *"Enter"* :

```
? What would you like to generate? service
? Name/path of service: /services/authentication/authentication
> ng.cmd generate service /services/authentication/authentication
CREATE src/app/services/authentication/authentication.service.spec.ts (373 bytes)
CREATE src/app/services/authentication/authentication.service.ts (143 bytes)
[OK] Generated service!
```

Os arquivos de serviço serão criados automaticamente na pasta *"services"* .
Adicione ao arquivo *"authentication.service.ts"* ou o código necessário para chamadas de autenticação do Firebase.

Técnico de Informática

Desenvolvimento de App Mobile

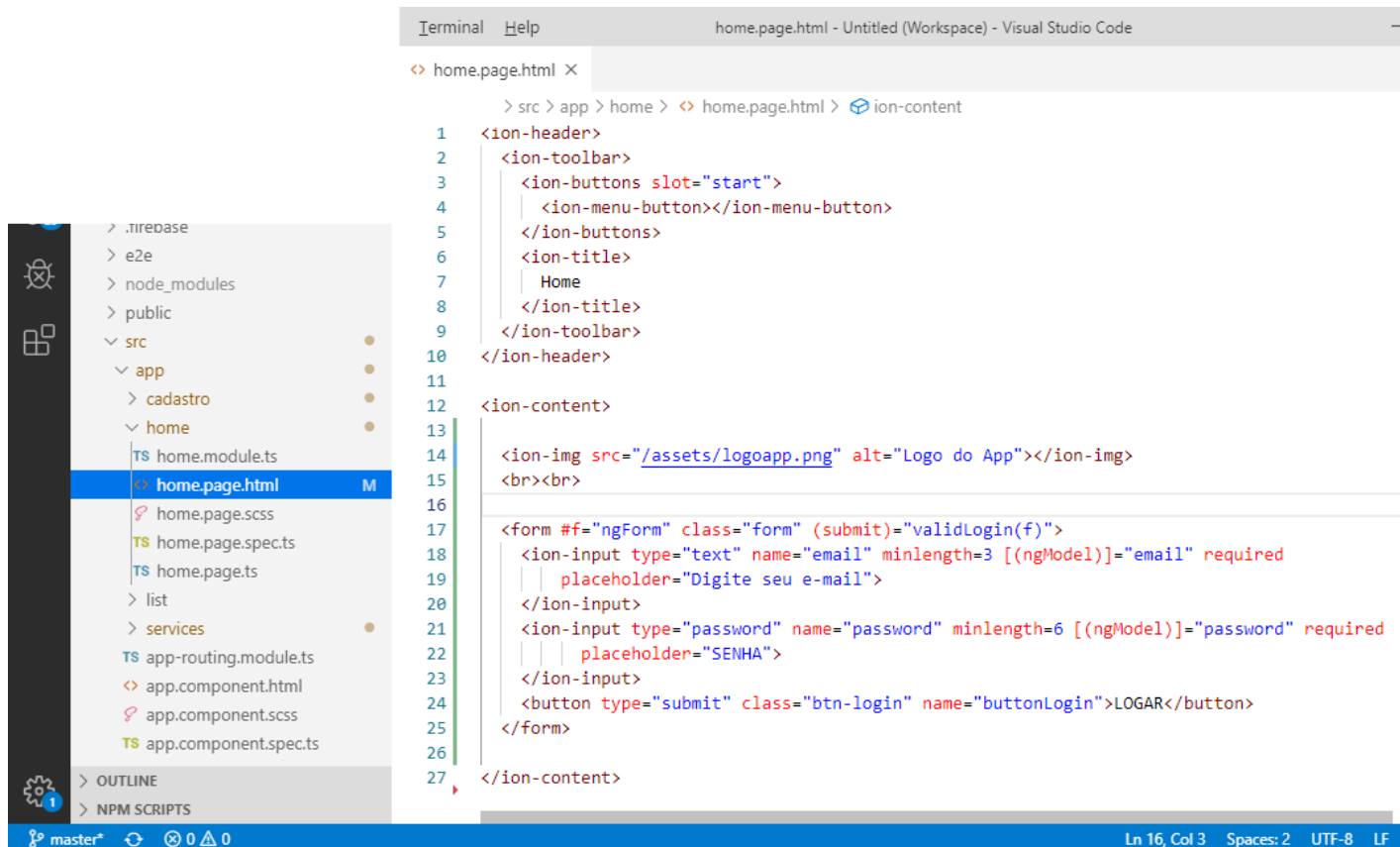


The screenshot displays the Visual Studio Code interface with a TypeScript file named `authentication.service.ts` open. The file is part of a project structure located at `src/app/services/authentication`. The code defines an `AuthenticationService` class that is injectable and provides a root provider. It implements two methods: `registerUser` and `loginUser`, both using the Firebase SDK for authentication. The `registerUser` method creates a new user with email and password, while the `loginUser` method signs in a user and returns a boolean indicating success. The interface also shows a file explorer on the left with the project structure and a terminal at the bottom.

```
1 import { Injectable } from '@angular/core';
2 import * as firebase from 'firebase/app';
3
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class AuthenticationService {
8
9   constructor() { }
10
11   registerUser(email, pwd) {
12     return new Promise<any>((resolve, reject) => {
13       firebase.auth().createUserWithEmailAndPassword(email, pwd)
14         .then((res) => {
15           console.log(res);
16           resolve();
17         }).catch((error) => {
18           console.log(error);
19           reject();
20         });
21     });
22   }
23
24   loginUser(email, pwd) {
25     return new Promise<any>((resolve, reject) => {
26       firebase.auth().signInWithEmailAndPassword(email, pwd)
27         .then((res) => {
28           console.log(res);
29           if (res.user) {
30             res.user.getIdToken().then((token) => {
31               console.log(token);
32               return true;
33             });
34           }
35           resolve();
36         });
37     });
38   }
39 }
```

Formulário de autenticação

Vamos adicionar os campos de email e a senha na página “*home.page.html*” do nosso aplicativo para podermos utilizar a autenticação do Firebase:



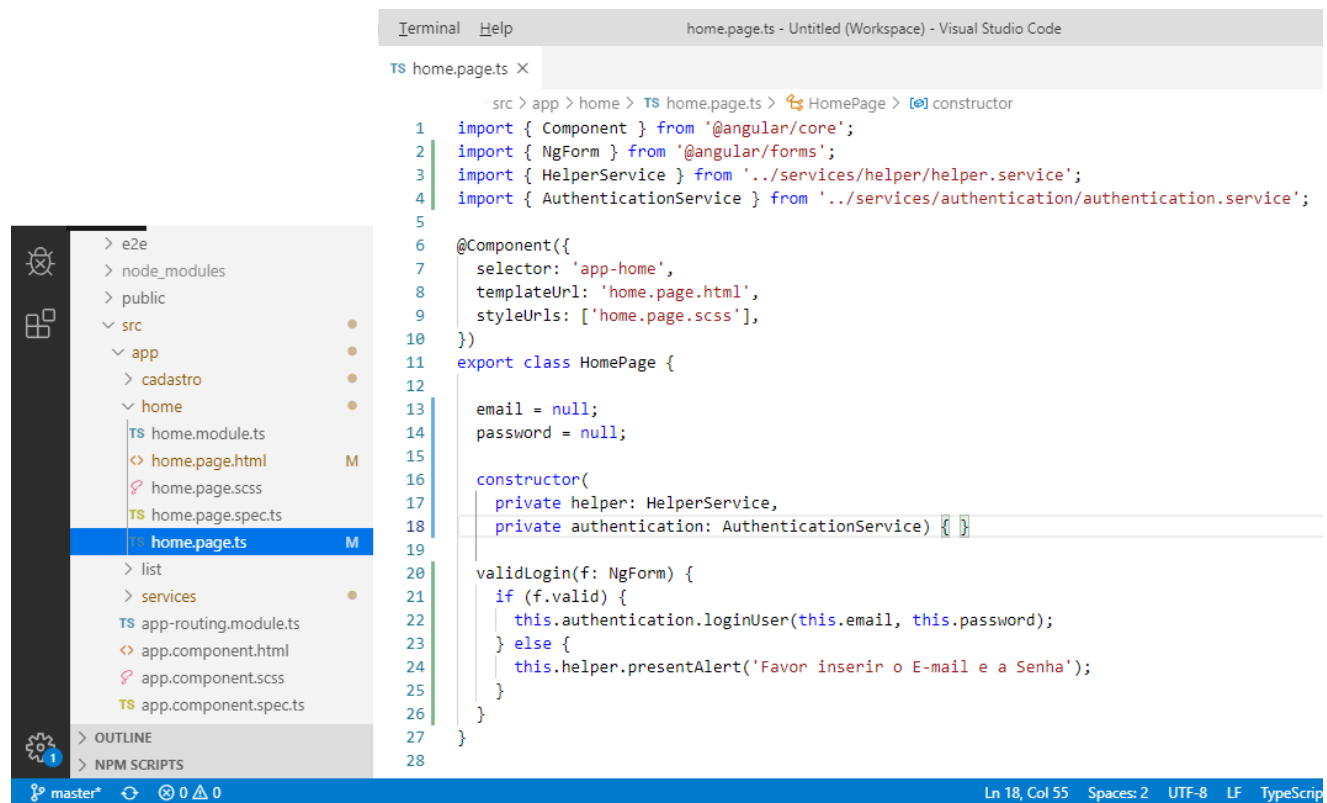
```
Terminal  Help  home.page.html - Untitled (Workspace) - Visual Studio Code

<> home.page.html X

> src > app > home > <> home.page.html > ion-content

1 <ion-header>
2 <ion-toolbar>
3   <ion-buttons slot="start">
4     <ion-menu-button></ion-menu-button>
5   </ion-buttons>
6   <ion-title>
7     Home
8   </ion-title>
9 </ion-toolbar>
10 </ion-header>
11
12 <ion-content>
13
14 <ion-img src="/assets/logoapp.png" alt="Logo do App"></ion-img>
15 <br><br>
16
17 <form #f="ngForm" class="form" (submit)="validLogin(f)">
18   <ion-input type="text" name="email" minlength=3 [(ngModel)]= "email" required
19     | placeholder="Digite seu e-mail">
20   </ion-input>
21   <ion-input type="password" name="password" minlength=6 [(ngModel)]= "password" required
22     | placeholder="SENHA">
23   </ion-input>
24   <button type="submit" class="btn-login" name="buttonLogin">LOGAR</button>
25 </form>
26
27 </ion-content>
```

E então no arquivo “*home.page.ts*” vamos criar o método que chama a autenticação do Firebase após o clique no botão “*Logar*” :



```
Terminal  Help  home.page.ts - Untitled (Workspace) - Visual Studio Code

TS home.page.ts x

src > app > home > TS home.page.ts > HomePage > constructor

1  import { Component } from '@angular/core';
2  import { NgForm } from '@angular/forms';
3  import { HelperService } from '../services/helper/helper.service';
4  import { AuthenticationService } from '../services/authentication/authentication.service';
5
6  @Component({
7    selector: 'app-home',
8    templateUrl: 'home.page.html',
9    styleUrls: ['home.page.scss'],
10 })
11 export class HomePage {
12
13   email = null;
14   password = null;
15
16   constructor(
17     private helper: HelperService,
18     private authentication: AuthenticationService) {}
19
20   validLogin(f: NgForm) {
21     if (f.valid) {
22       this.authentication.loginUser(this.email, this.password);
23     } else {
24       this.helper.presentAlert('Favor inserir o E-mail e a Senha');
25     }
26   }
27 }
28
```

Helper Service

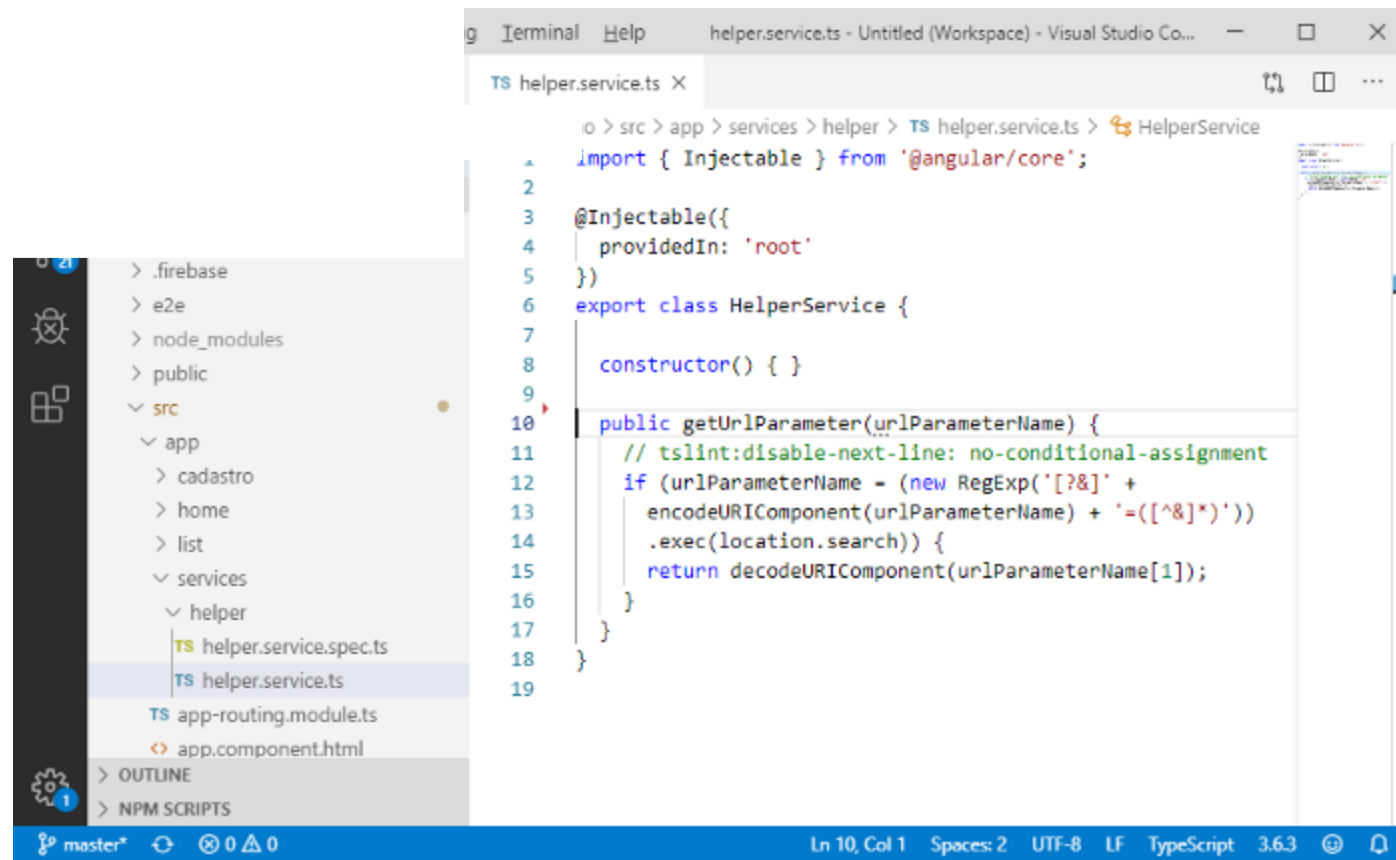
Da mesma forma que o *global.scss* pode ser acessado por todas as páginas, tem alguns métodos em TypeScript que gostaríamos de poder acessar de várias páginas diferentes, tipo métodos utilitários.

Para isso, podemos criar um **serviço**, chamá-lo de “*Helper*” e colocar os métodos que quisermos lá. Digite na janela shell:

ionic g service “services/helper”

```
er (app@0.0.0) ionic g service services/helper/helper
> ng.cmd generate service services/helper/helper
CREATE src/app/services/helper/helper.service.spec.ts (333 bytes)
CREATE src/app/services/helper/helper.service.ts (135 bytes)
[OK] Generated service!
```

Após criado o nosso serviço *Helper*, vamos adicionar nele uma função que recebe os parâmetros passados pela URL do app. Vamos fazer uma função genérica que poderá ser acessada em qualquer página. É só colocá-la dentro do arquivo ***“services/helper/helper.service.ts”***:



The screenshot shows the Visual Studio Code interface. On the left, the file explorer displays the project structure with the following folders and files:

- .firebase
- e2e
- node_modules
- public
- src
 - app
 - cadastro
 - home
 - list
 - services
 - helper
 - helper.service.spec.ts
 - helper.service.ts
 - app-routing.module.ts
 - app.component.html- OUTLINE
- NPM SCRIPTS

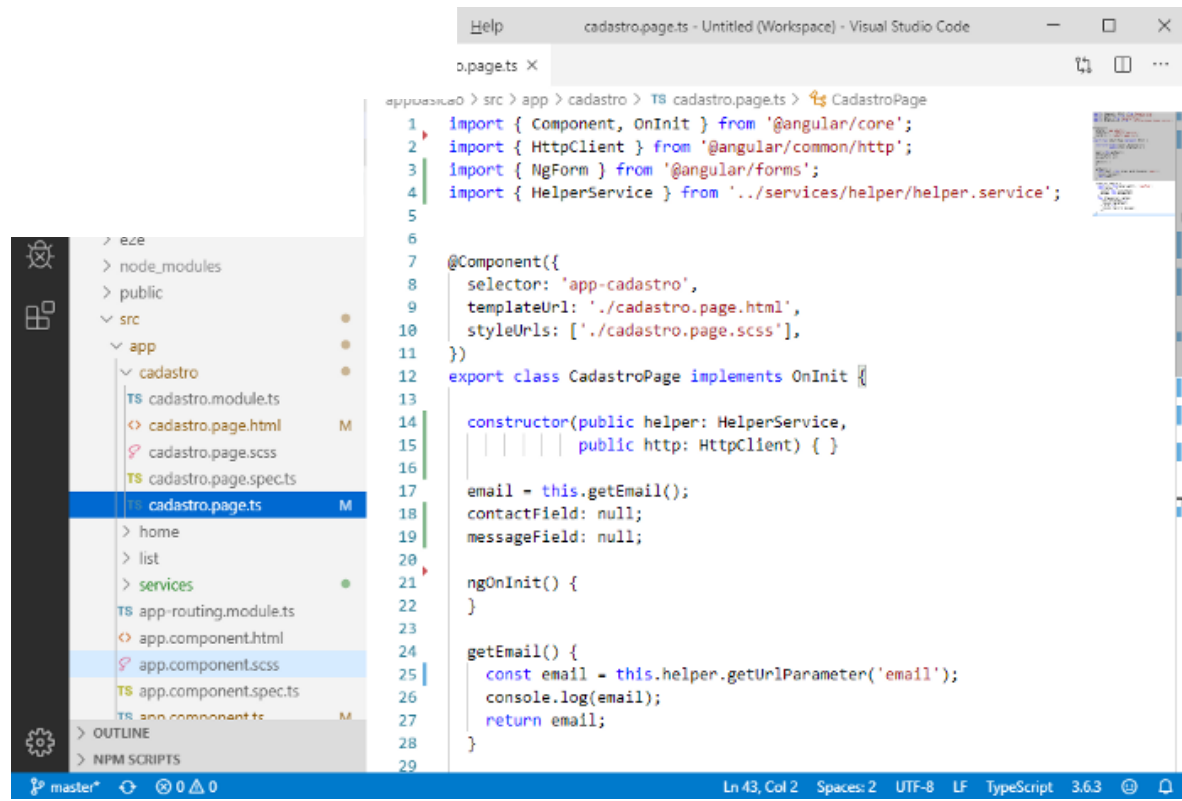
The main editor window shows the `helper.service.ts` file with the following TypeScript code:

```
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class HelperService {
7
8    constructor() { }
9
10   public getUrlParameter(urlParameterName) {
11     // tslint:disable-next-line: no-conditional-assignment
12     if (urlParameterName = (new RegExp('[?&]' +
13       encodeURIComponent(urlParameterName) + '=([^\&]*)'))
14       .exec(location.search)) {
15       return decodeURIComponent(urlParameterName[1]);
16     }
17   }
18 }
19
```

The status bar at the bottom indicates the current position is Line 10, Column 1, with 2 spaces, UTF-8 encoding, LF line endings, and TypeScript 3.6.3.

Você poderá usar essa função lá nos arquivos `.ts` de suas páginas, como por exemplo no arquivo `"cadastro.page.ts"`.

No arquivo `.ts` de sua página, importe a classe **HelperService** e passe-a como parâmetro no construtor para inicializá-la. A função genérica estará então disponível para ser acessada:



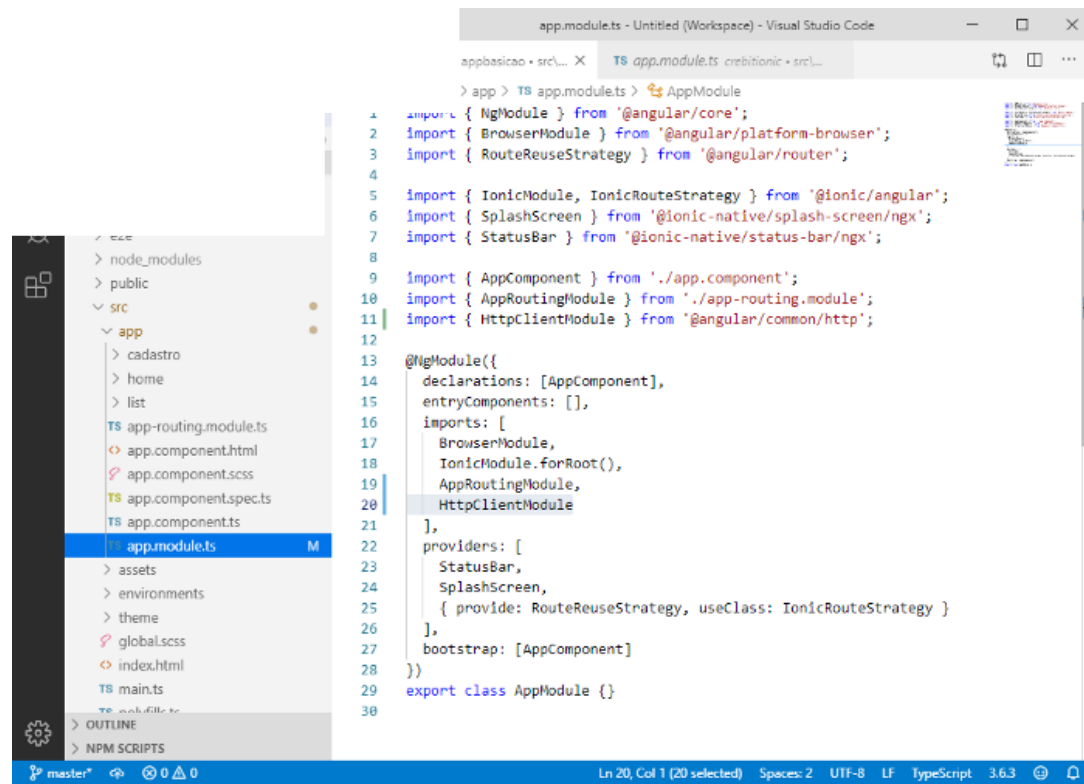
```
Help cadastro.page.ts - Untitled (Workspace) - Visual Studio Code
p.page.ts x
app\src\src > src > app > cadastro > TS cadastro.page.ts > CadastroPage
1 import { Component, OnInit } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3 import { NgForm } from '@angular/forms';
4 import { HelperService } from '../services/helper/helper.service';
5
6
7 @Component({
8   selector: 'app-cadastro',
9   templateUrl: './cadastro.page.html',
10  styleUrls: ['./cadastro.page.scss'],
11 })
12 export class CadastroPage implements OnInit {
13
14   constructor(public helper: HelperService,
15              public http: HttpClient) { }
16
17   email = this.getEmail();
18   contactField: null;
19   messageField: null;
20
21   ngOnInit() {
22   }
23
24   getEmail() {
25     const email = this.helper.getUriParameter('email');
26     console.log(email);
27     return email;
28   }
29 }
```

Observe na execução do servidor do Ionic como fica. Passe um valor no parâmetro *email* na Url do aplicativo, e depois veja na ferramenta de inspeção do seu navegador, esse mesmo valor sendo exibido na janela do console quando você abre a página de Cadastro:

HttpClient

E agora?! Estão prontos para salvarmos os campos de nosso cadastro num banco de dados?

Comece adicionando o módulo *HttpClient* no aplicativo através do arquivo ***“app.module.ts”***. Observe as linhas 11 e 20 abaixo para a importação do módulo:



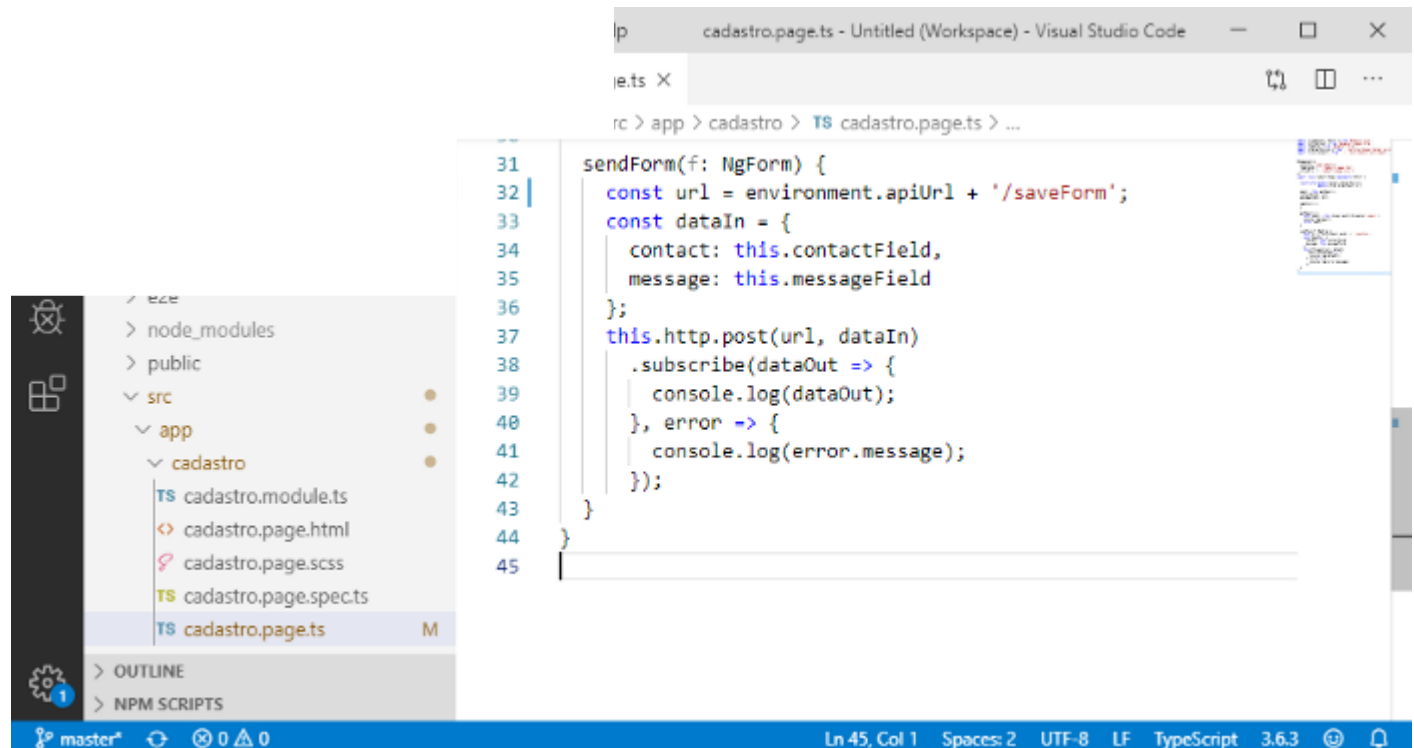
```
app.module.ts - Untitled (Workspace) - Visual Studio Code

appbasico + src\... X TS app.module.ts crebitionic + src\...

> app > TS app.module.ts > AppModule

1 import { NgModule } from '@angular/core';
2 import { BrowserModule } from '@angular/platform-browser';
3 import { RouteReuseStrategy } from '@angular/router';
4
5 import { IonicModule, IonicRouteStrategy } from '@ionic/angular';
6 import { SplashScreen } from '@ionic-native/splash-screen/ngx';
7 import { StatusBar } from '@ionic-native/status-bar/ngx';
8
9 import { AppComponent } from './app.component';
10 import { AppRoutingModule } from './app-routing.module';
11 import { HttpClientModule } from '@angular/common/http';
12
13 @NgModule({
14   declarations: [AppComponent],
15   entryComponents: [],
16   imports: [
17     BrowserModule,
18     IonicModule.forRoot(),
19     AppRoutingModule,
20     HttpClientModule
21   ],
22   providers: [
23     StatusBar,
24     SplashScreen,
25     { provide: RouteReuseStrategy, useClass: IonicRouteStrategy }
26   ],
27   bootstrap: [AppComponent]
28 })
29 export class AppModule {}
30
```

No arquivo `.ts` da sua página, você vai adicionar então a chamada HTTP para a API de armazenamento dos dados do nosso Cadastro:



```
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45
```

```
sendForm(f: NgForm) {  
  const url = environment.apiUrl + '/saveForm';  
  const dataIn = {  
    contact: this.contactField,  
    message: this.messageField  
  };  
  this.http.post(url, dataIn)  
    .subscribe(dataOut => {  
      console.log(dataOut);  
    }, error => {  
      console.log(error.message);  
    });  
}
```

Explorer sidebar:

- node_modules
- public
- src
 - app
 - cadastro
 - cadastro.module.ts
 - cadastro.page.html
 - cadastro.page.scss
 - cadastro.page.spec.ts
 - cadastro.page.ts

STATUS: master 0 0 0

Ln 45, Col 1 Spaces: 2 UTF-8 LF TypeScript 3.6.3

Environments

Com Ionic você pode criar automaticamente [diferentes ambientes](#) para rodar o seu código. Basta setar as variáveis de configuração desses ambientes separadamente dentro dos arquivos da pasta “*environments*”.

Abra o arquivo “*/environments/environment.ts*” e adicione como uma variável a URL da API de backend que iremos utilizar para armazenar os dados de nosso cadastro:

Para gerar as plataformas Android e IOS

***cordova ou cdv

- ionic integrations disable capacitor
- ionic cdv platform add ios
- ionic cdv platform add android
- ionic cdv build android
- Ionic cdv build ios



Fique Sabendo!

No **IOS**, os arquivos de instalação tem extensão **IPA**, e no **Android** os arquivos de instalação tem extensão **APK**

<https://ionicframework.com/docs/v3/intro/deploying/>

Técnico de Informática

Desenvolvimento de App Mobile



Dispositivos iOS

Ao contrário do Android, os desenvolvedores de iOS precisam gerar um perfil de provisionamento para codificar seus aplicativos para teste. A boa notícia é que, a partir do iOS9, você pode desenvolver e testar seus aplicativos em seu dispositivo iOS sem uma conta paga de desenvolvedor da Apple. Isso é particularmente excelente para desenvolvedores que desejam experimentar o desenvolvimento móvel com o Ionic, pois economiza custos, mas ainda oferece muitos recursos de ter uma conta completa de desenvolvedor da Apple. Para uma análise completa dos recursos incluídos, verifique [os documentos da Apple](#).

Requisitos

- Xcode 7 ou superior
- iOS 9
- Um [ID Apple](#) gratuito ou conta paga de desenvolvedor Apple



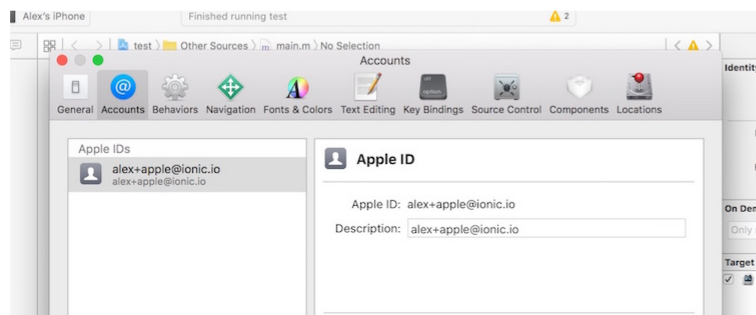
Criação de um perfil de provisionamento

Para começar, você precisará configurar um perfil de provisionamento para codificar seus aplicativos.

Usando um ID Apple

1. Abra as preferências do Xcode (Xcode > Preferências ...)
2. Clique na guia 'Contas'
3. Faça login com seu ID Apple (+> Adicionar ID Apple ...)

Depois de fazer o login com sucesso, uma nova 'Equipe Pessoal' com a função 'Grátis' aparecerá abaixo do seu ID Apple.



É necessário ter um ID da Apple ou uma conta paga de desenvolvedor.

Quando se adquire um produto da Apple, o ID é gerado no ato da compra, podendo ser usado para desenvolver aplicativos também!

Use os
outros
templates

```
22 | * This import should be commented out in production mode because it will...
```

DEBUG CONSOLE TERMINAL ... 2: node + [] [] ^ x

Please enter the full name of your app. You can change this at any time. To bypass this prompt next time, supply **name**, the first argument to **ionic start**.

? **Project name:** TesteApp12

Let's pick the perfect starter template!

Starter templates are ready-to-go Ionic apps that come packed with everything you need to build your app. To bypass this prompt next time, supply **template**, the second argument to **ionic start**.

? **Starter template:** (Use arrow keys)

> tabs	A starting project with a simple tabbed interface
sidemenu	A starting project with a side menu with navigation in the content area
blank	A blank starter project
list	A starting project with a list
my-first-app	An example application that builds a camera with gallery
conference	A kitchen-sink application that shows off all Ionic has to offer

Simulado Prático - 1

- Disponível na plataforma.
- Vale 100 pontos.
- Todo conteúdo visto até semana passada.
- 10 questões de múltipla escolha.

➤ Quarta-feira será habilitado o Simulado Prático -2.



Aula 10 - Atividade ▼

Arquivos anexados:  [UC Desenvolvimento de App Mobile - Atividade 10.docx](#) ▼ (15,72 KB)



Aula 11 - Atividade ▼

Arquivos anexados:  [UC Desenvolvimento de App Mobile - Atividade 11.docx](#) ▼ (15,782 KB)



Simulado Prático - 1 ▼

Simulado Prático 1

Desenvolvimento de App Mobile - Bonsucesso



Desenvolvimento de App Mobile – Aula 12

Atividade 12

Individual ou dupla

Coloque os nomes

**(Ambos da dupla devem enviar
para ficar registrado na plataforma)**


1. Atualize sua aplicação usando os recursos de banco de dados (Firestore) ofertados pelo Firebase, preferencialmente para criar um formulário de cadastro de voluntários simples (no caso de ação / tema social).
Atualize também o relatório simplificado explicando os recursos adicionados na aplicação, usando "screen shots" para ilustrar.
Atualize a aplicação no seu "GitHub".

Resposta :

As respostas na forma de arquivo texto (TXT, DOC, PDF, HTML, link do Google Docs...) para o instrutor, através da plataforma.



Aula 11 - Atividade ▼

Arquivos anexados:  [UC Desenvolvimento de App Mobile - Atividade 11.docx](#) ▼ (15,782 KB)



Simulado Prático - 1 ▼

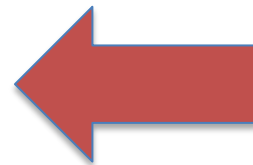
Simulado Prático 1

Desenvolvimento de App Mobile - Bonsucesso



Aula 12 - Atividade ▼

Arquivos anexados:  [UC Desenvolvimento de App Mobile - Atividade 12.docx](#) ▼ (15,814 KB)



Próxima Aula

Acessando dispositivos do celular como Câmera por exemplo.

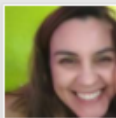
LINHA DE DISCUSSÃO: AULA 12- DÚVIDAS GERAIS

Selecionar: [Todos](#) [Nenhum](#)

Ações de mensagem ▾

Expandir tudo

Fechar tudo



GISELLE AZEVEDO PINTO ★

Aula 12- Dúvidas Gerais

Coloque aqui suas dúvidas sobre a Aula 12.

Executou os procedimentos?

Apareceu alguma mensagem de erro ? Qual foi? Conseguiu resolver? Se não conseguiu anota aqui para resolvermos "juntos"!

Consegue adaptar a aplicação para um outro cenário?

Coloque suas anotações e comentários aqui. Esse espaço é seu!

Responder

Citar

Editar

Excluir

Selecionar: [Todos](#) [Nenhum](#)

Ações de mensagem ▾

Expandir tudo

Fechar tudo



Dúvidas?

O que achou?

Todos os arquivos disponíveis na plataforma!



Não deixe de praticar.

Gere um novo projeto para praticar!

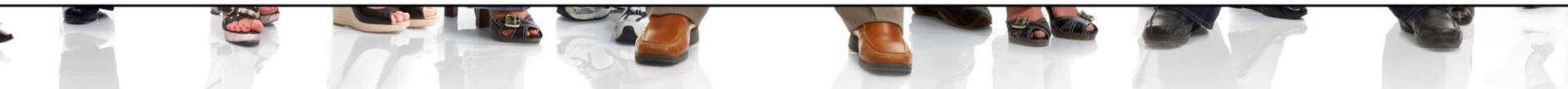
Concluindo...

- Adapte o projeto FormContato para você usar.
- Não deixe de praticar!





Até a próxima!



Anexo

TS environment.ts

TS app.module.ts

TS contato-form.ts x



```
11 selector: 'contato-form',
12 templateUrl: 'contato-form.html'
13 })
14 export class ContatoFormComponent {
15
16     contatoForm: FormGroup;
17
18     constructor(public formbuilder: FormBuilder) {
19         this.contatoForm = this.formbuilder.group({
20             nome: [null, [Validators.required, Validators.minLength(5)]],
21             empresa: [null, [Validators.minLength(3)]],
22             telefone: [null, [Validators.required, Validators.minLength(10), Validators.maxLength(15)]],
23             cep: [null, [Validators.required, Validators.minLength(8), Validators.maxLength(9)]],
24             rua: [null, [Validators.required]],
25             bairro: [null, [Validators.required]],
26             numero: [null],
27             cidade: [null, [Validators.required]],
28             uf: [null, [Validators.required]]
29         })
30     }
31 }
```

EXEMPLO
DE COD

EXEMPLO DE COD

```
TS environment.ts  TS app.module.ts  TS contato-form.ts ●
1  import { Component } from '@angular/core';
2  import { FormGroup, FormBuilder, Validators } from '@angular/forms';
3  import { AngularFireDatabase } from '@angular/fire/database'
4
5  /**
6   * Generated class for the ContatoFormComponent component.
7   *
8   * See https://angular.io/api/core/Component for more info on Angular
9   * Components.
10  */
11  @Component({
12    selector: 'contato-form',
13    templateUrl: 'contato-form.html'
```


TS environment.ts

TS app.module.ts

TS contato-form.ts ●

```
11  @Component({
12      selector: 'contato-form',
13      templateUrl: 'contato-form.html'
14  })
15  export class ContatoFormComponent {
16
17      contatoForm: FormGroup;
18
19      constructor(
20          public FormBuilder: FormBuilder,
21          public db: AngularFireDatabase
22      ) {
23          this.contatoForm = this.formBuilder.group({
```

EXEMPLO
DE COD

```
TS environment.ts  TS app.module.ts  TS contato-form.ts  [Q]
21 public db: AngularFireDatabase
22 ) {
23   this.contatoForm = this.formbuilder.group({
24     nome: [null, [Validators.required, Validators.minLength(5)]],
25     empresa: [null, [Validators.minLength(3)]],
26     telefone: [null, [Validators.required, Validators.minLength(10), Validators.ma
27     cep: [null, [Validators.required, Validators.minLength(8), Validators.maxLengt
28     rua: [null, [Validators.required]],
29     bairro: [null, [Validators.required]],
30     numero: [null],
31     cidade: [null, [Validators.required]],
32     uf: [null, [Validators.required]]
33   })
34 }
35
36 cadastraContato() {
37   this.db.database.ref('/contatos').push([this.contatoForm.value])
38   .then(() => {
39     console.log('salvou')
40   })
41 }
```

EXEMPLO
DE COD

}

EXPLORER

OPEN EDITORS 1 UNSAVED

- TS contato-form.ts src/co... U
- TS app.module.ts src/app M
- TS environment.ts src/env... U

FIREBASE-CRUD

- app
- TS app.component.ts M
- <> app.html
- TS app.module.ts M
- app.scss
- TS main.ts
- assets
- components
 - contato-form
 - <> contato-form.html U

TS contato-form.ts

```
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45
```

TS app.module.ts

```
AngularFireModule.initializeApp  
ComponentsModule,  
HttpModule  
],  
bootstrap: [IonicApp],  
entryComponents: [  
  MyApp,  
  HomePage,  
  CadastroPage  
],  
providers: [  
  StatusBar,  
  SplashScreen,  
  AngularFireDatabase,  
  {provide: ErrorHandler, useClass  
]  
})  
export class AppModule {}
```

EXEMPLO
DE COD