

PROF. MARCOS CALDAS
DESENVOLVIMENTO
DE SISTEMAS

PROF. MARCOS CALDAS
DESENVOLVIMENTO
DE SISTEMAS

IONIC

PROF. MARCOS CALDAS
DESENVOLVIMENTO
DE SISTEMAS

PROF. MARCOS CALDAS
DESENVOLVIMENTO
DE SISTEMAS

Iniciando o projeto

ionic start myApp tabs --capacitor

Framework Angular

<https://capacitorjs.com/docs/apis/camera>

Instalar os pacotes conforme documentação:

<https://capacitorjs.com/docs/web/pwa-elements>

- npm install @ionic/pwa-elements
- npm install @capacitor/camera

capacitorjs.com/docs/web/pwa-elements

capacitor DOCS

- ▼ Getting Started
 - Introduction
 - Environment Setup
 - Installing Capacitor
 - Building Your UI
 - Using with Ionic Framework
- VS Code Extension
- Capacitor Templates
- FAQs
- ▼ Basics
 - Development Workflow
 - Using Plugins
 - Configuring Your App
 - JavaScript API
- ▼ Upgrade Guides
 - Upgrading to 5.0

// Call the element loader after the app has been rendered the first time
`defineCustomElements(window);`

Angular

main.ts:

```
import { enableProdMode } from '@angular/core';
import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';

import { AppModule } from './app/app.module';
import { environment } from './environments/environment';

import { defineCustomElements } from '@ionic/pwa-elements/loader';

if (environment.production) {
  enableProdMode();
}

platformBrowserDynamic()
  .bootstrapModule(AppModule)
  .catch(err => console.log(err));

// Call the element loader after the platform has been bootstrapped
defineCustomElements(window);
```

Copy

<https://capacitorjs.com/docs/web/pwa-elements>

main.ts

```
TS main.ts M X
src > TS main.ts > ...
1  import { enableProdMode } from '@angular/core';
2  import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';
3
4  import { AppModule } from './app/app.module';
5  import { environment } from './environments/environment';
6
7  import { defineCustomElements } from '@ionic/pwa-elements/loader';
8
9  if (environment.production) {
10    enableProdMode();
11  }
12
13  platformBrowserDynamic().bootstrapModule(AppModule)
14    .catch(err => console.log(err));
15
16  defineCustomElements(window);
```

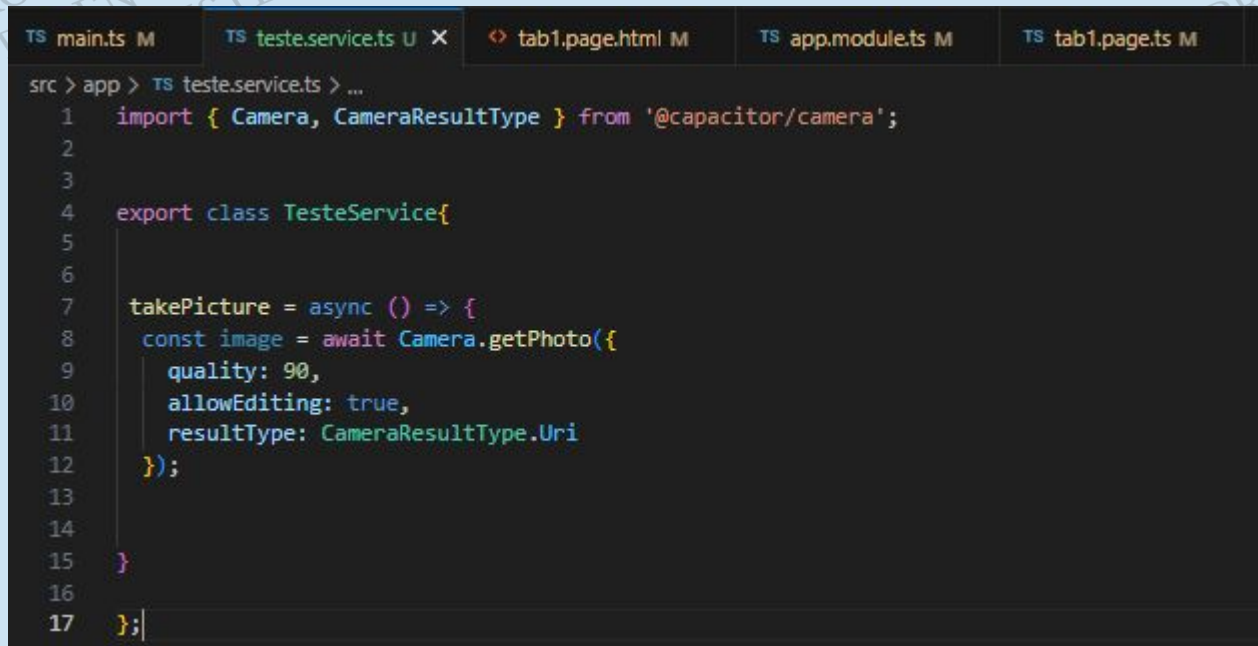
app.module.ts

TS main.ts M TS teste.service.ts U tab1.page.html M TS app.module.ts M X TS tab1.page.ts M

src > app > TS app.module.ts > ...

```
1 | import { TesteService } from './teste.service';
2 | import { NgModule } from '@angular/core';
3 | import { BrowserModule } from '@angular/platform-browser';
4 | import { RouteReuseStrategy } from '@angular/router';
5 |
6 | import { IonicModule, IonicRouteStrategy } from '@ionic/angular';
7 |
8 | import { AppRoutingModule } from './app-routing.module';
9 | import { AppComponent } from './app.component';
10 |
11 | @NgModule({
12 |   declarations: [AppComponent],
13 |   imports: [BrowserModule, IonicModule.forRoot(), AppRoutingModule],
14 |   providers: [{ provide: RouteReuseStrategy, useClass: IonicRouteStrategy },
15 |     TesteService
16 |   ],
17 |   bootstrap: [AppComponent],
18 | })
19 | export class AppModule {}
20 |
```

Criar um serviço **ionic generate service teste** e inserir o código abaixo:



The screenshot shows an IDE with several tabs open: 'main.ts M', 'teste.service.ts U X', 'tab1.page.html M', 'app.module.ts M', and 'tab1.page.ts M'. The active tab is 'teste.service.ts U X'. The code in the editor is as follows:

```
src > app > TS teste.service.ts > ...
1  import { Camera, CameraResultType } from '@capacitor/camera';
2
3
4  export class TesteService{
5
6
7    takePicture = async () => {
8      const image = await Camera.getPhoto({
9        quality: 90,
10       allowEditing: true,
11       resultType: CameraResultType.Uri
12     });
13
14
15   }
16
17   };
```

<https://capacitorjs.com/docs/apis/camera>

tab1.pag.ts

```
TS main.ts M    TS teste.service.ts U    <> tab1.page.html M    TS app.module.ts M    TS tab1.page.ts M X
src > app > tab1 > TS tab1.page.ts > ...
1  import { Component } from '@angular/core';
2  import { TesteService } from '../teste.service';
3
4  @Component({
5    selector: 'app-tab1',
6    templateUrl: 'tab1.page.html',
7    styleUrls: ['tab1.page.scss']
8  })
9  export class Tab1Page {
10
11    constructor(public testeService: TesteService) {}
12
13  }
```


tab1.page.html

```
TS main.ts M    TS teste.service.ts U    <> tab1.page.html M X    TS app.module.ts M    TS tab1.page.ts M

src > app > tab1 > <> tab1.page.html > ...
Go to component
1  <ion-header [translucent]="true">
2    <ion-toolbar>
3      <ion-title>
4        Tab 1
5      </ion-title>
6    </ion-toolbar>
7  </ion-header>
8
9  <ion-content [fullscreen]="true">
10    <ion-header collapse="condense">
11      <ion-toolbar>
12        <ion-title size="large">Tab 1</ion-title>
13      </ion-toolbar>
14    </ion-header>
15
16    <ion-fab-button (click)="testeService.takePicture()">
17      <ion-icon name="camera">
18
19      </ion-icon>
20    </ion-fab-button>
21    <app-explore-container name="Tab 1 page"></app-explore-container>
22  </ion-content>
```

Rodar a aplicação na pasta do projeto **ionic serve**.

A câmera só vai abrir se o computador tiver uma câmera.

