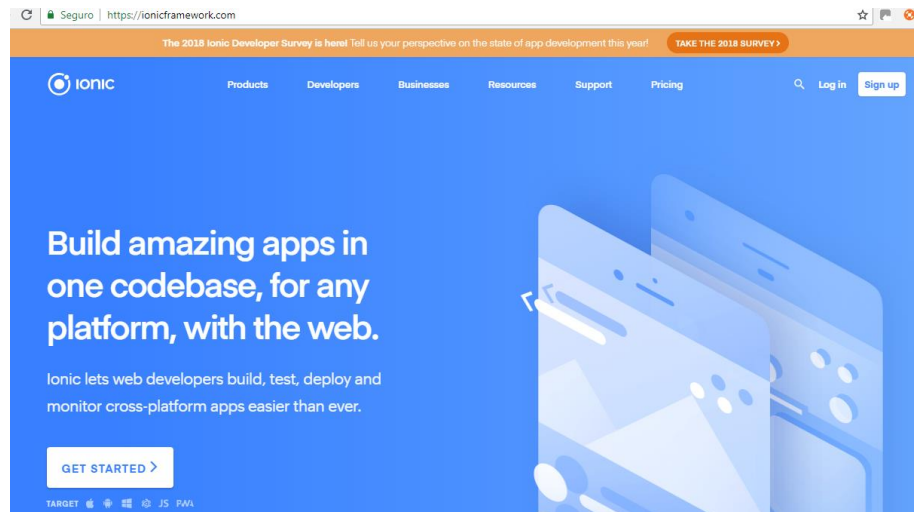


IONIC

MODIFICANDO O PROJETO

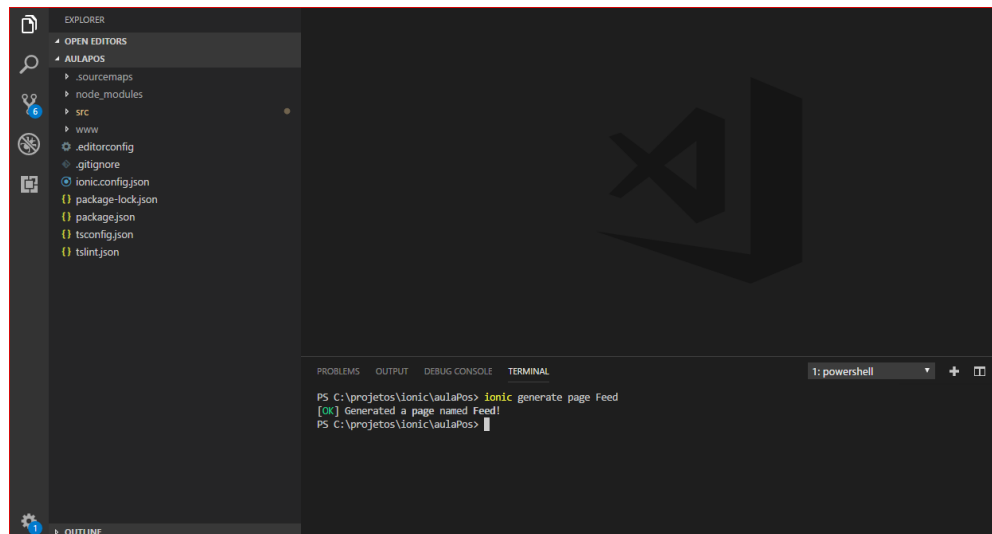


CRIANDO PÁGINAS

CODIFICANDO

PASSOS

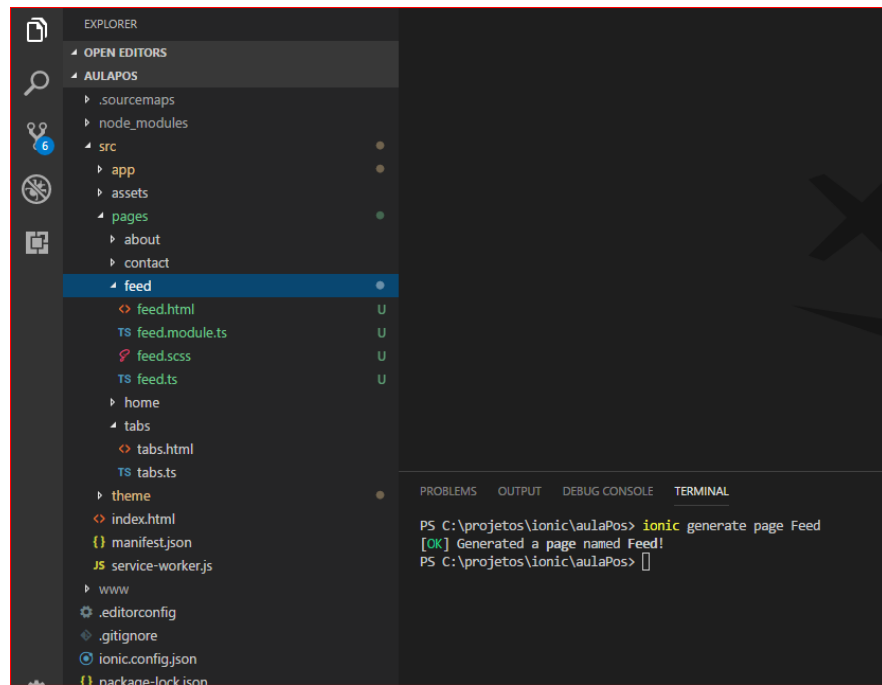
- ▶ Iremos utilizar o ionic cli para criar páginas. Para criar uma pagina de feed digite:
- ▶ *ionic generate page feed*



CODIFICANDO

PASSOS

- ▶ A estrutura básica da página Feed foi criada



CODIFICANDO

PASSOS

- ▶ Vamos utilizar alguns componentes na nossa app

The screenshot displays the Ionic 4 documentation page for the 'Cards' component. On the left is a sidebar with a search bar and a list of navigation links: HOME, INTRO, COMPONENTS, Overview, Action Sheets, Alerts, Badges, Buttons, Cards (highlighted), Checkbox, DateTime, FABs, Gestures, Grid, Icons, Inputs, Lists, Loading, and Menus. A banner at the top of the sidebar says 'Ionic 4 is coming' with a link to 'Preview the new beta docs'. The main content area is titled 'Cards' and contains two paragraphs explaining the component's purpose and use cases. Below the text is a 'Contents' list with links to 'Basic Cards', 'Card Headers', 'Card Lists', 'Card Images', 'Background Images', and 'Advanced Cards'. A 'Basic Usage' section follows, stating that cards are primarily a CSS component and providing a code snippet for a basic card structure. To the right of the documentation is a mobile app demo titled 'Basic Cards' on an iPhone screen, showing a card with the text 'This is just your basic card with some text to boot. Like it? Keep scrolling...'. At the top right of the demo are tabs for 'iOS', 'Android', and 'Windows', with 'iOS' selected. A 'Demo Source' link is located below the demo.

Cards

Cards are a great way to display important pieces of content, and are quickly emerging as a core design pattern for apps. They are a great way to contain and organize information, while also setting up predictable expectations for the user. With so much content to display at once, and often so little screen real estate, cards have fast become the design pattern of choice for many companies, including the likes of [Google](#), [Twitter](#), and [Spotify](#).

For mobile experiences, Cards make it easy to display the same information visually across many different screen sizes. They allow for more control, are flexible, and can even be animated. Cards are usually placed on top of one another, but they can also be used like a "page" and swiped between, left and right.

Contents

- [Basic Cards](#)
- [Card Headers](#)
- [Card Lists](#)
- [Card Images](#)
- [Background Images](#)
- [Advanced Cards](#)

Basic Usage

Cards are primarily a CSS component. To use a basic card, follow this structure:

```
<ion-card>

  <ion-card-header>
    Card Header
  </ion-card-header>

  <ion-card-content>
    <!-- Add card content here! -->
  </ion-card-content>
</ion-card>
```

[Demo Source](#)

Basic Cards

This is just your basic card with some text to boot. Like it? Keep scrolling...

CODIFICANDO

PASSOS

- Ir na página do ionic UI componentes e escolher o cards

Cards

Cards are a great way to display important pieces of content, and are quickly emerging as a core design pattern for apps. They are a great way to contain and organize information, while also setting up predictable expectations for the user. With so much content to display at once, and often so little screen real estate, cards have fast become the design pattern of choice for many companies, including the likes of [Google](#), [Twitter](#), and [Spotify](#).

For mobile experiences, Cards make it easy to display the same information visually across many different screen sizes. They allow for more control, are flexible, and can even be animated. Cards are usually placed on top of one another, but they can also be used like a "page" and swiped between, left and right.

Contents

- [Basic Cards](#)
- [Card Headers](#)
- [Card Lists](#)
- [Card Images](#)
- [Background Images](#)
- [Advanced Cards](#)

Basic Usage

Cards are primarily a CSS component. To use a basic card, follow this structure:

```
<ion-card>

  <ion-card-header>
    Card Header
  </ion-card-header>

  <ion-card-content>
    <!-- Add card content here! -->
  </ion-card-content>
</ion-card>
```

[Demo Source](#)

Basic Cards

This is just your basic card with some text to boot. Like it? Keep scrolling...

CODIFICANDO

PASSOS

- ▶ Copie o código do Card no estilo Social Cards e copie para a página de Feed. Em “feed/feed.html” dentro do <ion-content padding>

The screenshot displays the Ionic 4 documentation page for the 'Cards' component. On the left is a sidebar menu with a search bar and a list of components including Overview, Action Sheets, Alerts, Badges, Buttons, Cards (highlighted), Checkbox, DateTime, FABs, Gestures, Grid, Icons, Inputs, Lists, Loading, and Menus. The main content area is titled 'Cards' and contains an introduction, a list of sub-topics (Basic Cards, Card Headers, Card Lists, Card Images, Background Images, Advanced Cards), and a 'Basic Usage' section. The 'Basic Usage' section explains that cards are primarily a CSS component and provides the following HTML structure:

```
<ion-card>

  <ion-card-header>
    Card Header
  </ion-card-header>

  <ion-card-content>
    <!-- Add card content here! -->
  </ion-card-content>
</ion-card>
```

To the right of the code is a 'Demo Source' link. On the far right, a mobile device preview shows a 'Basic Cards' demo with a card containing the text: 'This is just your basic card with some text to boot. Like it? Keep scrolling...'

CODIFICANDO

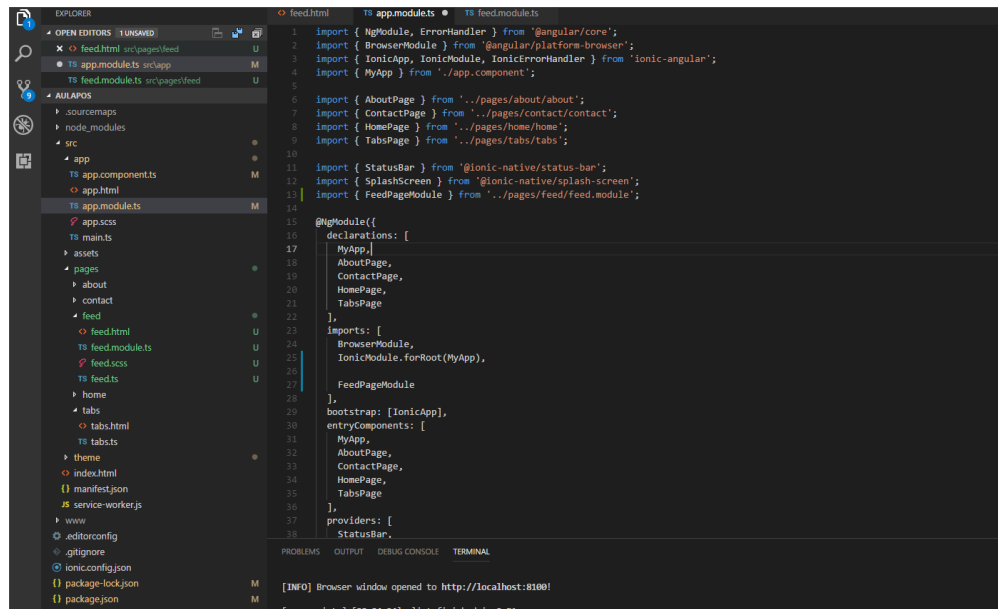
PASSOS

- ▶ Vamos colocar nossa pagina de Feed no menu junto com as outras

CODIFICANDO

PASSOS

- Ir no arquivo `app.module.ts` para importar a página de Feed no projeto.



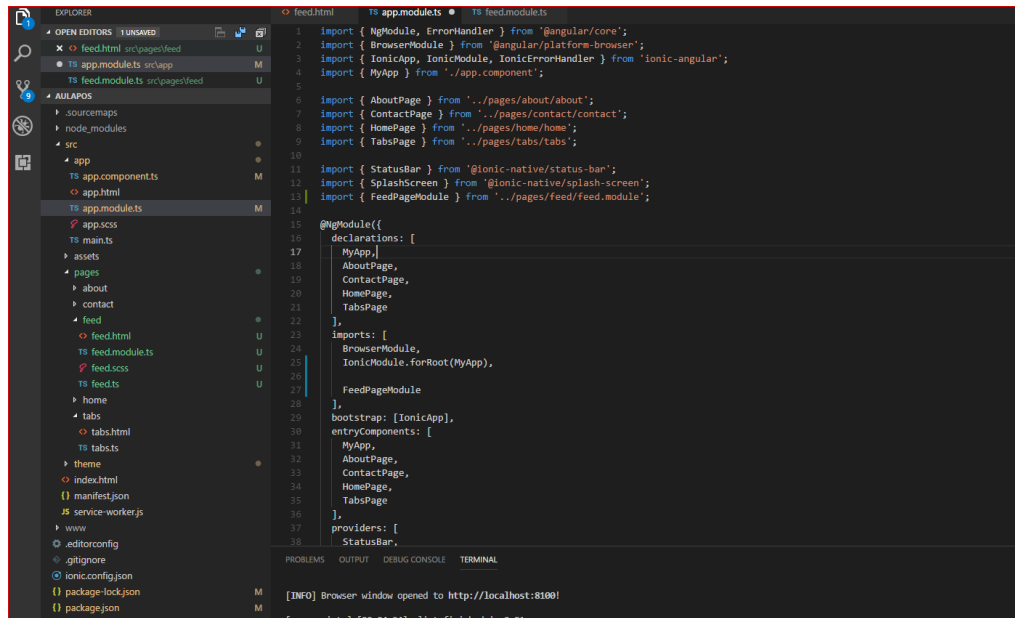
The screenshot shows an IDE with the following components:

- EXPLORER:** A file tree on the left showing the project structure. The file `app.module.ts` is selected under the `src/app` directory.
- EDITOR:** The main workspace showing the content of `app.module.ts`. The code includes imports for `NgModule`, `ErrorHandler`, `BrowserModule`, `RouterModule`, `RouterModule.forRoot`, `MyApp`, `AboutPage`, `ContactPage`, `HomePage`, `TabsPage`, `StatusBar`, `SplashScreen`, and `FeedPageModule`. The `@NgModule` decorator is also visible.
- TERMINAL:** A terminal window at the bottom showing the command `npm install` and its output, including the message `[INFO] Browser window opened to http://localhost:8100/`.

CODIFICANDO

PASSOS

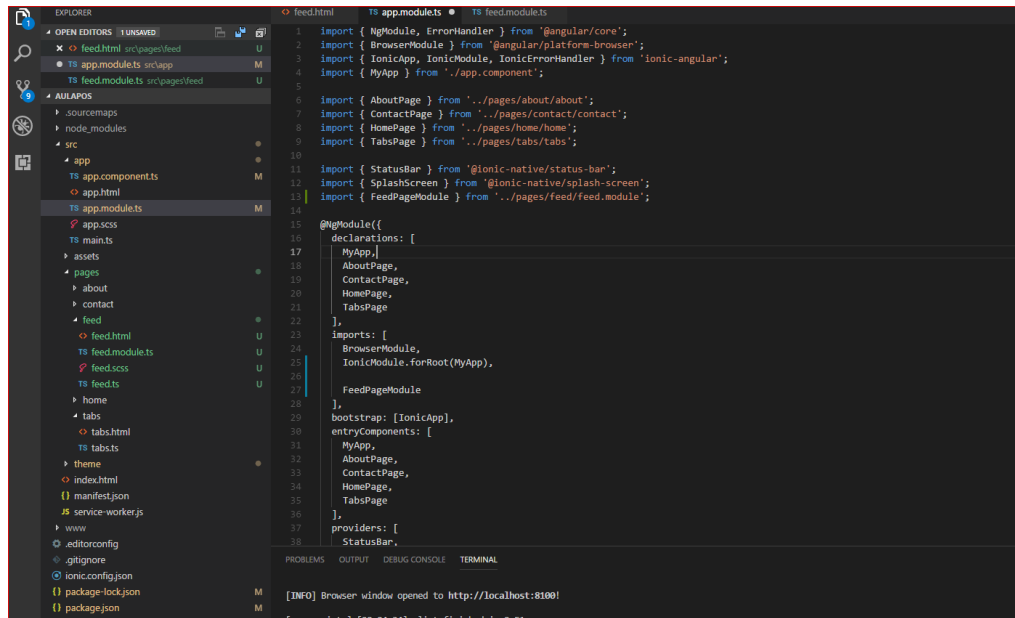
- ▶ Clicar em cima do erro que o Code gera, e importar a pagina corretamente



CODIFICANDO

PASSOS

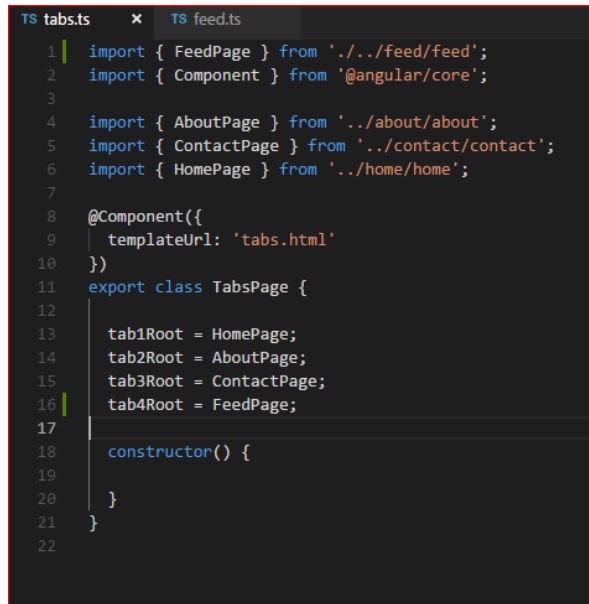
- ▶ Com isto temos o Feed Module dentro da aplicação



CODIFICANDO

PASSOS

► Para incluir o módulo nas abas, primeiramente devemos ir no arquivo de abas “tabs.ts” e importar a classe definida para a nova página “Feed” (nome está na classe em “feed.ts”



```
TS tabs.ts x TS feed.ts
1 import { FeedPage } from '../feed/feed';
2 import { Component } from '@angular/core';
3
4 import { AboutPage } from '../about/about';
5 import { ContactPage } from '../contact/contact';
6 import { HomePage } from '../home/home';
7
8 @Component({
9   templateUrl: 'tabs.html'
10 })
11 export class TabsPage {
12
13   tab1Root = HomePage;
14   tab2Root = AboutPage;
15   tab3Root = ContactPage;
16   tab4Root = FeedPage;
17
18   constructor() {
19
20   }
21 }
22
```

CODIFICANDO

PASSOS

- ▶ Inclua a página dentro do html dos tabs



```
TS tabs.ts    TS feed.ts    <> tabs.html x
1  <ion-tabs>
2    <ion-tab [root]="tab1Root" tabTitle="Home" tabIcon="home"></ion-tab>
3    <ion-tab [root]="tab2Root" tabTitle="About" tabIcon="information-circle"></ion-tab>
4    <ion-tab [root]="tab3Root" tabTitle="Contact" tabIcon="contacts"></ion-tab>
5    <ion-tab [root]="tab4Root" tabTitle="Feed" tabIcon="contacts"></ion-tab>
6  </ion-tabs>
7
```

CODIFICANDO

PASSOS

- ▶ Iniciar o projeto com ionic serve

The screenshot displays the Ionic 4 documentation page for the 'Cards' component. On the left is a sidebar with a search bar and a list of components: HOME, INTRO, COMPONENTS (Overview, Action Sheets, Alerts, Badges, Buttons, Cards, Checkbox, DateTime, FABs, Gestures, Grid, Icons, Inputs, Lists, Loading, Menus), and a 'Ionic 4 is coming' banner. The main content area is titled 'Cards' and includes an introduction, a list of sub-topics (Basic Cards, Card Headers, Card Lists, Card Images, Background Images, Advanced Cards), and a 'Basic Usage' section. The 'Basic Usage' section contains the following code snippet:

```
<ion-card>

  <ion-card-header>
    Card Header
  </ion-card-header>

  <ion-card-content>
    <!-- Add card content here! -->
  </ion-card-content>
</ion-card>
```

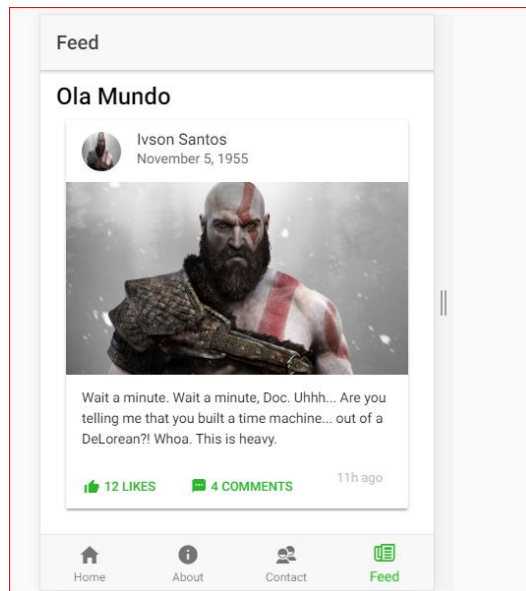
On the right side of the page, there are tabs for 'iOS', 'Android', and 'Windows'. Below these tabs is a mobile device mockup showing a 'Basic Cards' demo. The demo card contains the text: 'This is just your basic card with some text to boot. Like I? Keep scrolling...'. A 'Demo Source' link is also visible.

CRIANDO IMAGENS

CODIFICANDO

PASSOS

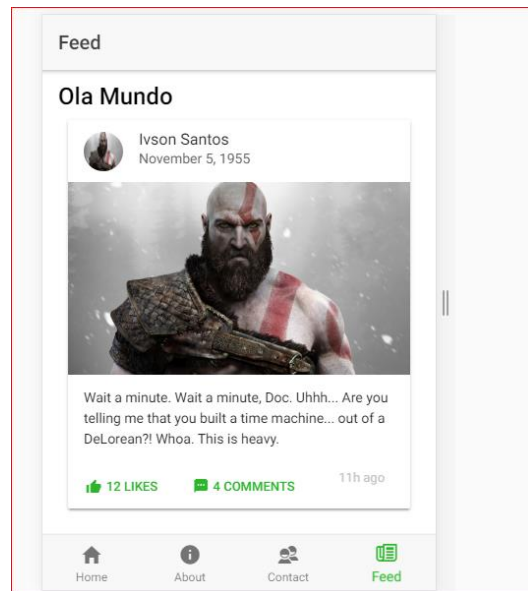
- ▶ Alterar a imagem dos itens do Feed, selecionando uma imagem desejada qualquer e colocar esta imagem na pasta “assets/imgs”



CODIFICANDO

PASSOS

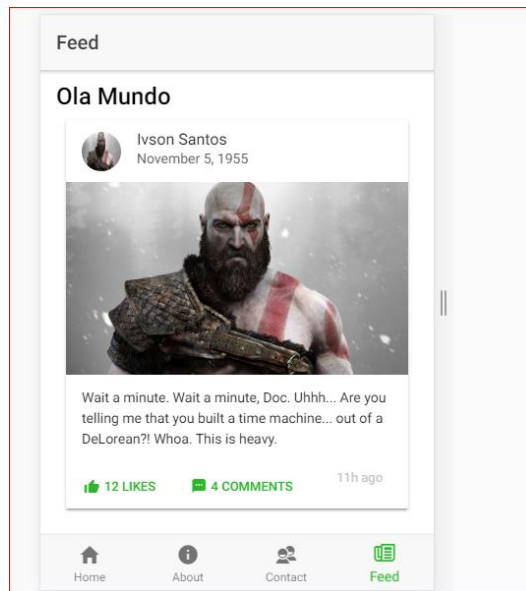
- ▶ Será criada a mesma imagem na pasta “www/assets/imgs”



CODIFICANDO

PASSOS

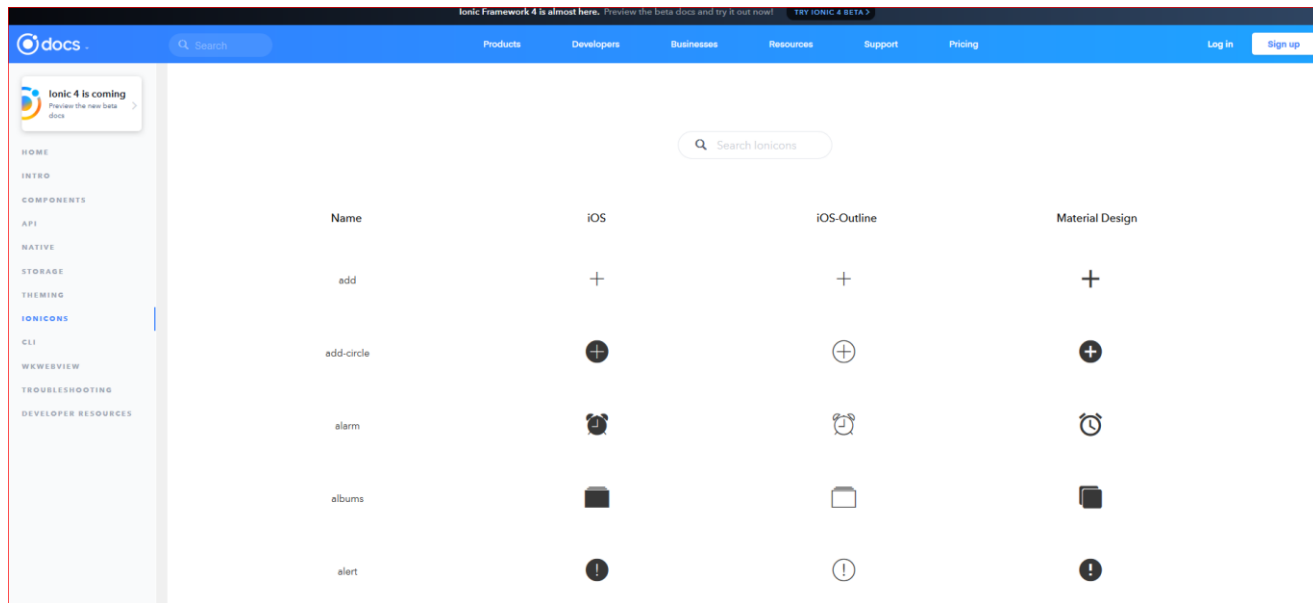
- ▶ Agora, para utilizar a imagem dentro da sua página, deve colocar o caminho “assets/imgs/<imagem>”



CODIFICANDO

PASSOS

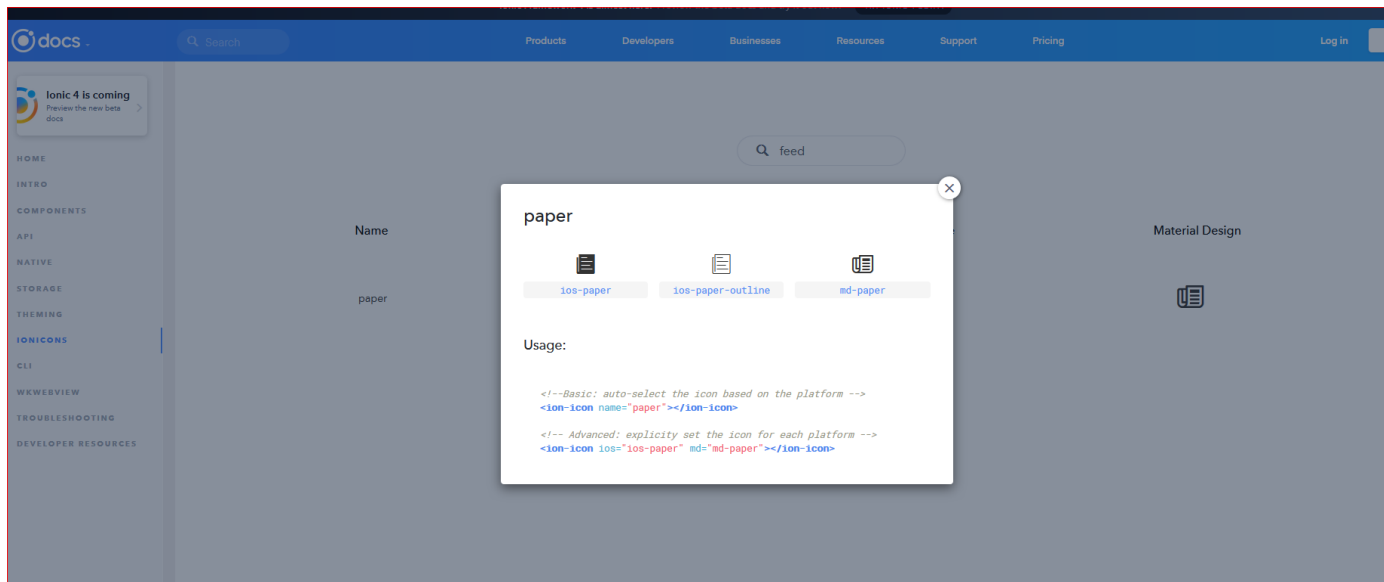
- ▶ Para alterar um ícone do ionic, selecione o ícone dentro de ionicons



CODIFICANDO

PASSOS

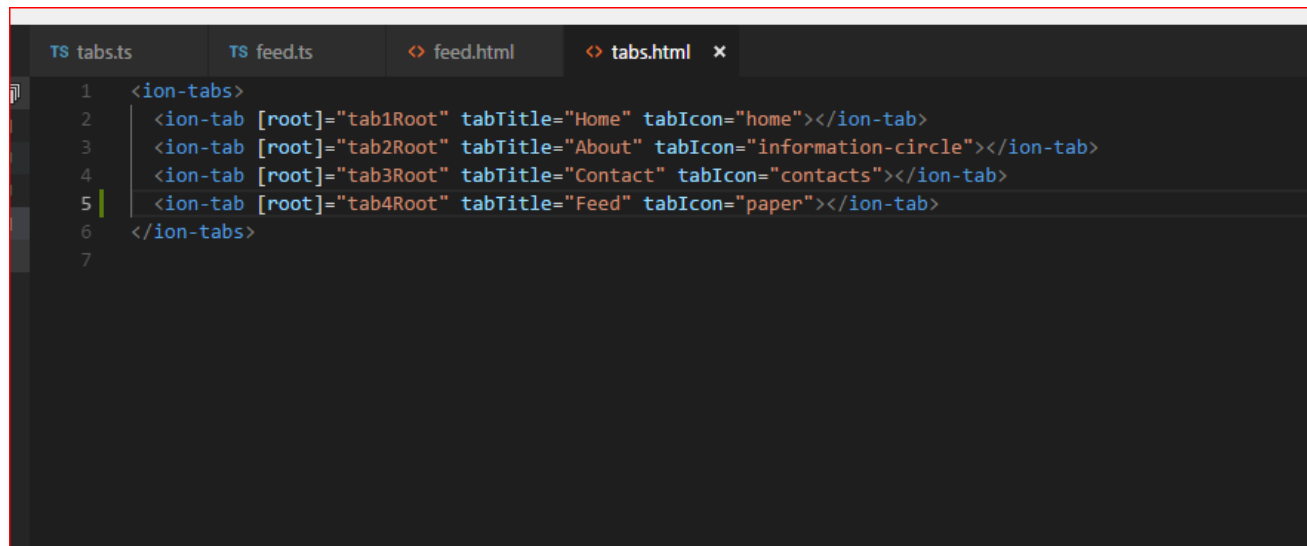
- ▶ Pesquise o ícone desejado e clique no mesmo para mostrar como se utiliza



CODIFICANDO

PASSOS

- ▶ Altere o nome do ícone para mostrar o ícone desejado



```
TS tabs.ts    TS feed.ts    <> feed.html    <> tabs.html x
1  <ion-tabs>
2    <ion-tab [root]="tab1Root" tabTitle="Home" tabIcon="home"></ion-tab>
3    <ion-tab [root]="tab2Root" tabTitle="About" tabIcon="information-circle"></ion-tab>
4    <ion-tab [root]="tab3Root" tabTitle="Contact" tabIcon="contacts"></ion-tab>
5    <ion-tab [root]="tab4Root" tabTitle="Feed" tabIcon="paper"></ion-tab>
6  </ion-tabs>
7
```

TYPESCRIPT

CODIFICANDO

PASSOS

- ▶ Crie uma variável dentro do arquivo de TypeScript

```
feed.html TS feed.ts x
1 import { Component } from '@angular/core';
2 import { IonicPage, NavController, NavParams } from 'ionic-angular';
3
4 /**
5  * Generated class for the FeedPage page.
6  *
7  * See https://ionicframework.com/docs/components/#navigation for more info on
8  * Ionic pages and navigation.
9  */
10
11 @IonicPage()
12 @Component({
13   selector: 'page-feed',
14   templateUrl: 'feed.html',
15 })
16 export class FeedPage {
17
18   public appName : String ="Ola Mundo";
19
20
21   constructor(public navCtrl: NavController, public navParams: NavParams) {}
22
23
24   ionViewDidLoad() {
25     console.log('ionViewDidLoad FeedPage');
26   }
27
28 }
29
```

CODIFICANDO

PASSOS

- ▶ Outros tipos de variáveis

```
public appName : String = "Ola Mundo";  
public numero : number;  
public qualquer : any;
```


CODIFICANDO

PASSOS

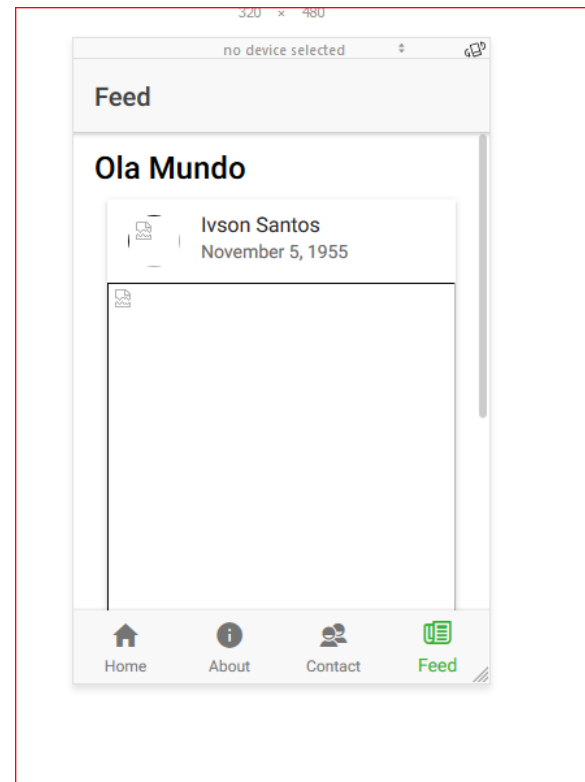
- ▶ Vincule essa variável no html para ser visualizada em tela.
- ▶ Fazer um binding por interpolação “{{ }}”

```
feed.html x TS feed.ts
4   See http://ionicframework.com/docs/components/#navigation for more info on
5   Ionic pages and navigation.
6   -->
7   <ion-header>
8
9     <ion-navbar>
10      <ion-title>Feed</ion-title>
11    </ion-navbar>
12
13  </ion-header>
14
15
16  <ion-content padding>
17
18    <h2> {{appName}} </h2>
19
20    <ion-card>
21
22      <ion-item>
23        <ion-avatar item-start>
24          
25        </ion-avatar>
26        <h2>Ivson Santos</h2>
27        <p>November 5, 1955</p>
28      </ion-item>
29
30      
31
```

CODIFICANDO

PASSOS

- ▶ Verifique as alterações no app



CODIFICANDO

PASSOS

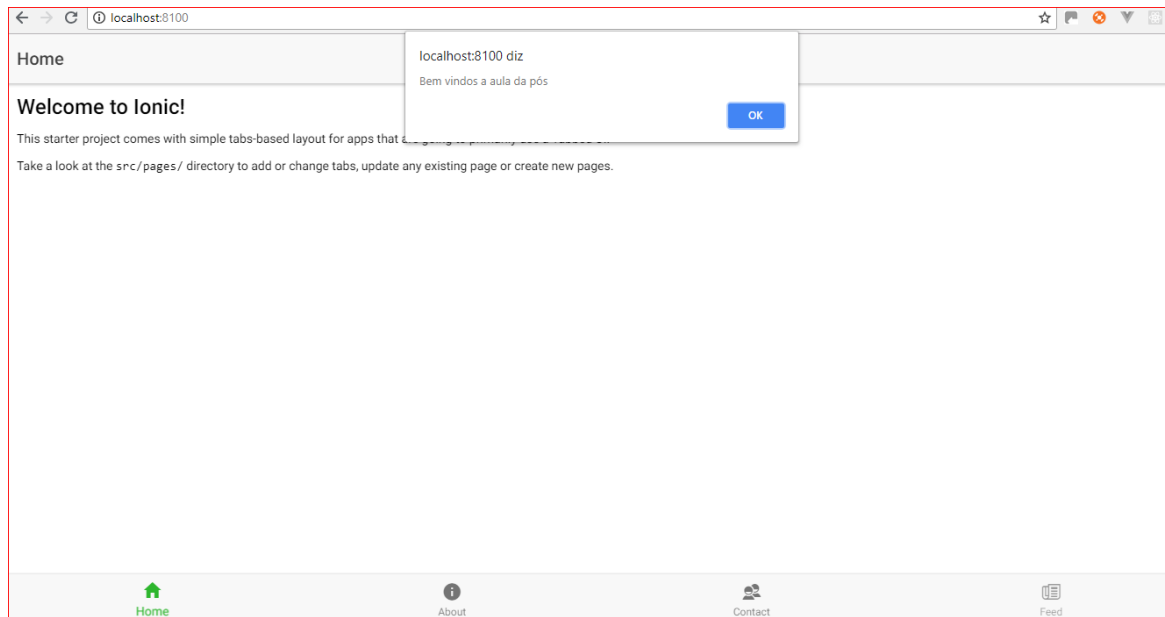
- ▶ Criar uma função para ser executada no app

```
<> feed.html x TS feed.ts x
11  @ionicPage()
12  @Component({
13    selector: 'page-feed',
14    templateUrl: 'feed.html',
15  })
16  export class FeedPage {
17
18    public appName : String ="Ola Mundo";
19    public numero : number;
20    public qualquer : any;
21
22
23    constructor(public navCtrl: NavController, public navParams: NavParams) {
24
25    }
26
27    public testaMostraEmTela():void {
28      alert("Bem vindos a aula da pós");
29    }
30
31    ionViewDidLoad() {
32      //console.log('ionViewDidLoad FeedPage');
33      this.testaMostraEmTela();
34    }
35  }
```

CODIFICANDO

PASSOS

- ▶ Verifique as alterações no app



CODIFICANDO

PASSOS

- ▶ Criar uma função com passagem de parâmetros e que escreva no console do app

```
feed.html TS feed.ts x
11 @ionicPage()
12 @Component({
13   selector: 'page-feed',
14   templateUrl: 'feed.html',
15 })
16 export class FeedPage {
17
18   public appName : String ="Ola Mundo";
19   public numero : number;
20   public qualquer : any;
21
22
23   constructor(public navCtrl: NavController, public navParams: NavParams) {
24
25   }
26
27   public testaMostraEmTela():void {
28     alert("Bem vindos a aula da pós");
29   }
30
31   ionViewDidLoad() {
32     //console.log('ionViewDidLoad FeedPage');
33     this.testaMostraEmTela();
34
35     console.log('Subtração: ' + this.testaSubtracao(10,7));
36   }
37
38   public testaSubtracao(a:number, b:number):number {
39     return a - b;
40   }
41
42 }
43
```

CSS CLASS

CODIFICANDO

PASSOS

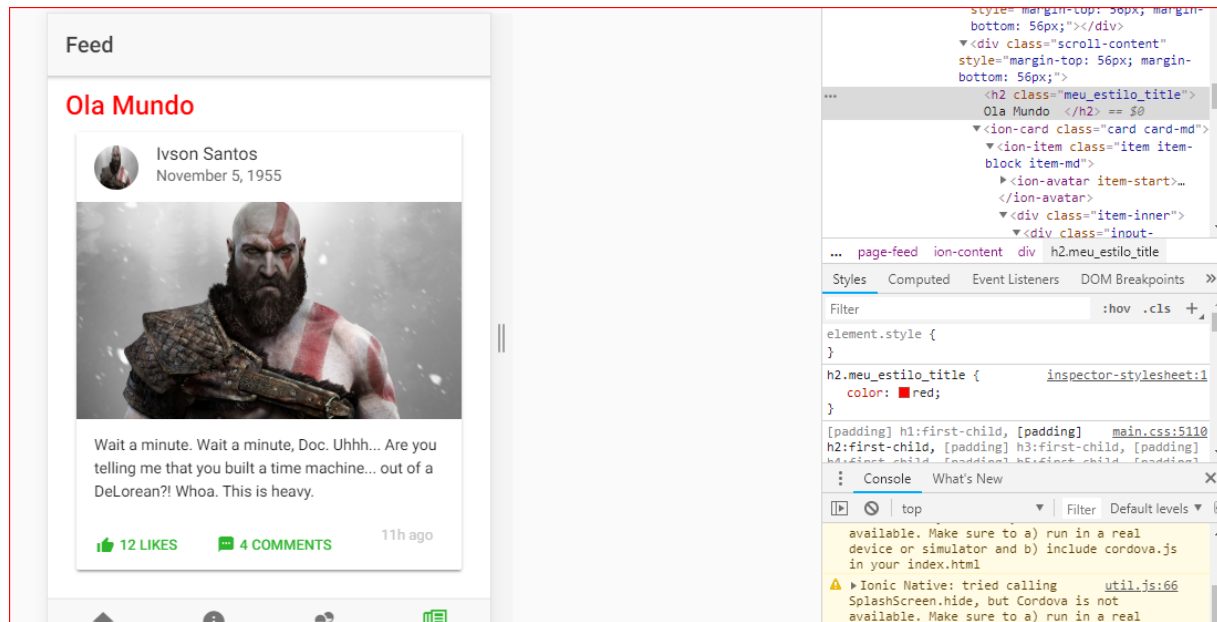
- ▶ Vá no H2 definido para a sua aplicação e crie um class para o <H2>

```
<> feed.html x TS feed.ts
4   See http://ionicframework.com/docs/components/#navigation for more info on
5   Ionic pages and navigation.
6   -->
7   <ion-header>
8
9   <ion-navbar>
10    <ion-title>Feed</ion-title>
11  </ion-navbar>
12
13 </ion-header>
14
15
16 <ion-content padding>
17
18   <h2 class="meu_estilo_title"> {{appName}} </h2>
19
20
```

CODIFICANDO

PASSOS

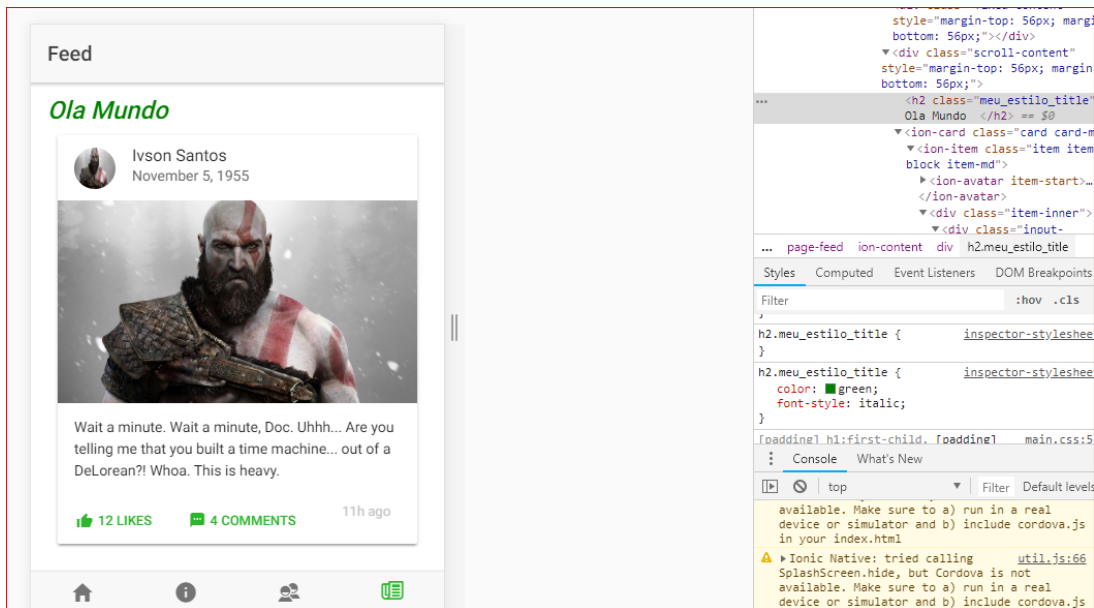
- ▶ Rode o projeto e inspecione o elemento para ver a classe criada



CODIFICANDO

PASSOS

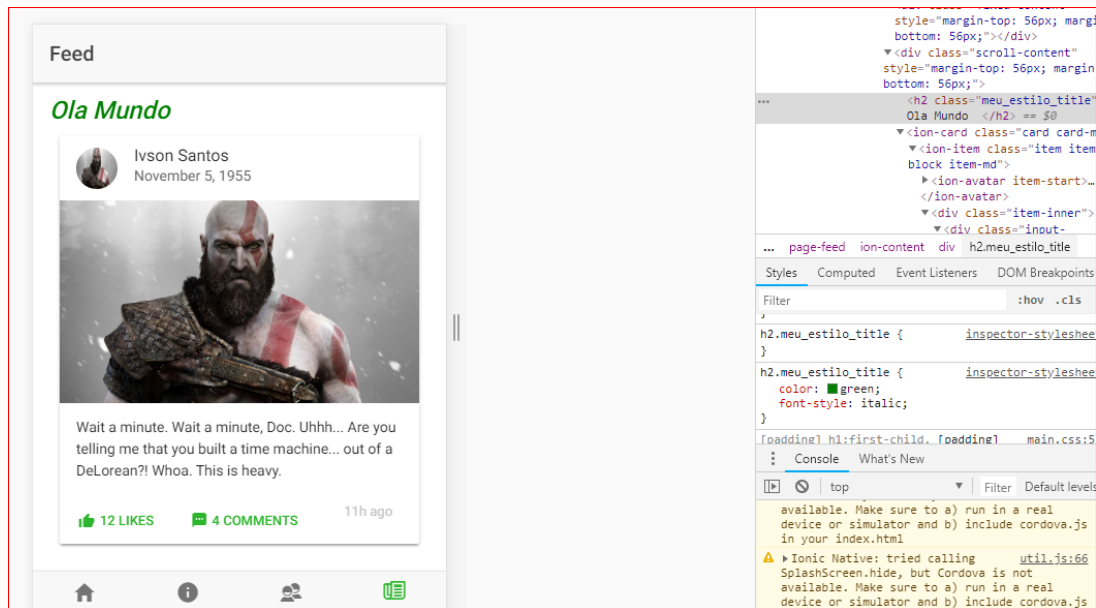
- Clique no elemento no inspector e clique na aba “Styles” e no sinal de “+” para poder editar a classe de estilo em tempo de execução.



CODIFICANDO

PASSOS

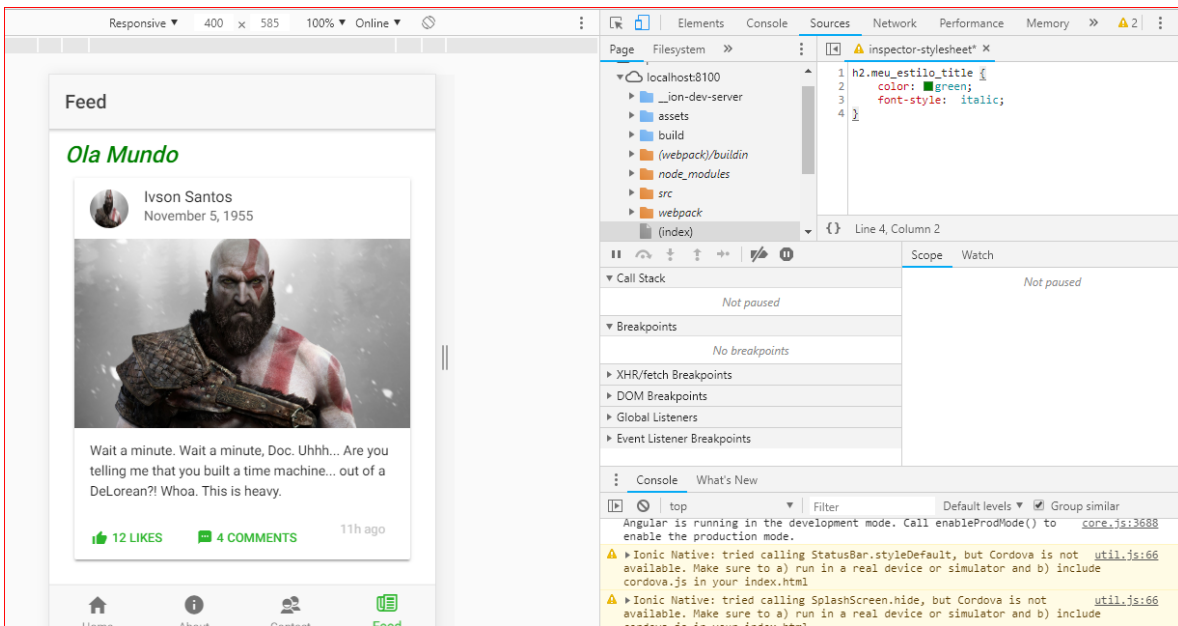
- ▶ Para copiar o estilo que estiver sendo editado, clicar no link “inspector-stylesheet”



CODIFICANDO

PASSOS

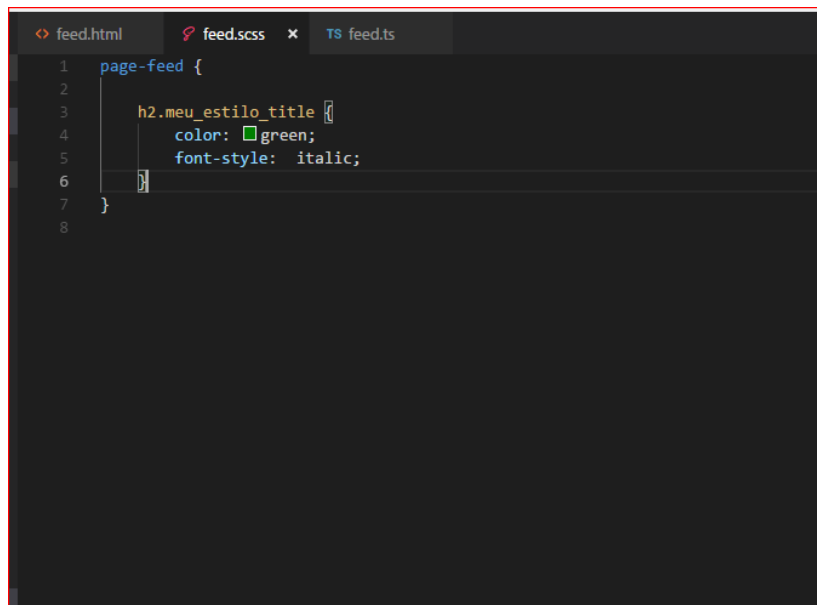
- ▶ Aqui você pode copiar o estilo que está sendo criado



CODIFICANDO

PASSOS

- ▶ Colocar o estilo dentro de feed.scss - se for utilizar apenas na página de feed, ou em app.scss – se for utilizar de forma mais global



The screenshot shows a code editor with three tabs: 'feed.html', 'feed.scss', and 'TS feed.ts'. The 'feed.scss' tab is active, displaying the following SCSS code:

```
1 page-feed {  
2  
3   h2.meu_estilo_title {  
4     color: green;  
5     font-style: italic;  
6   }  
7 }  
8
```