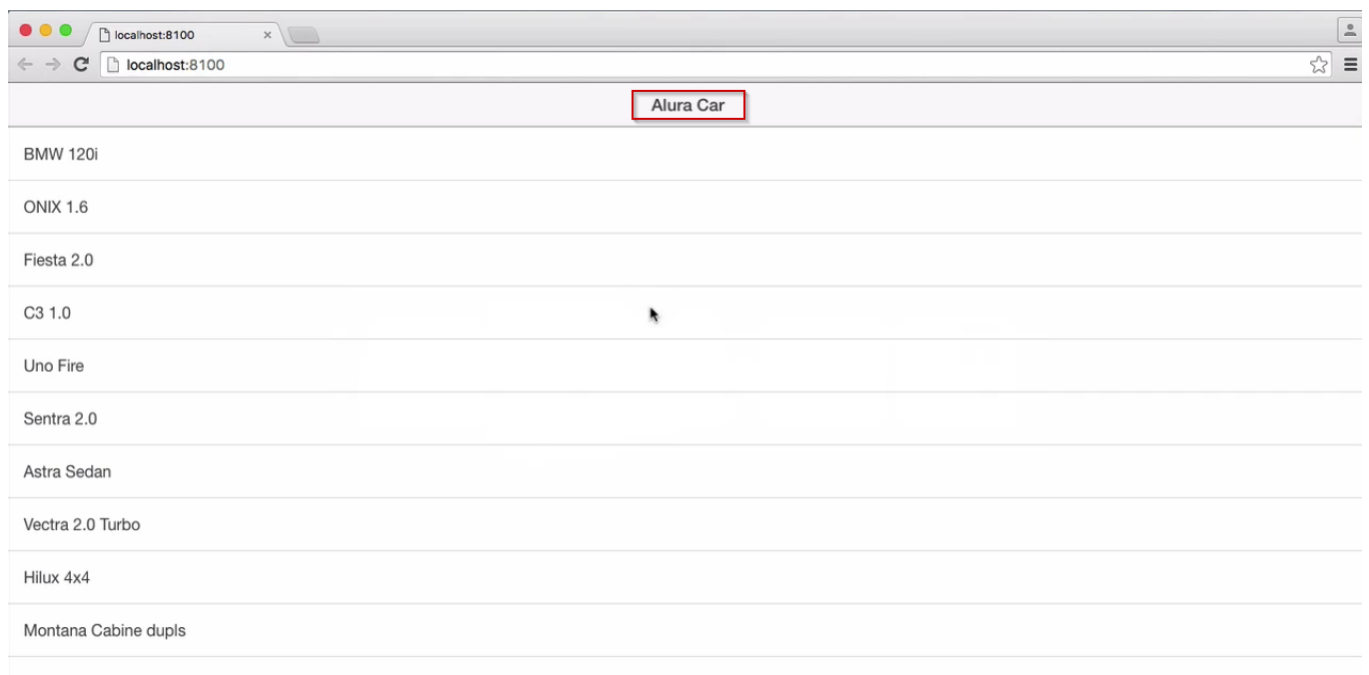


Customizando o nosso produto

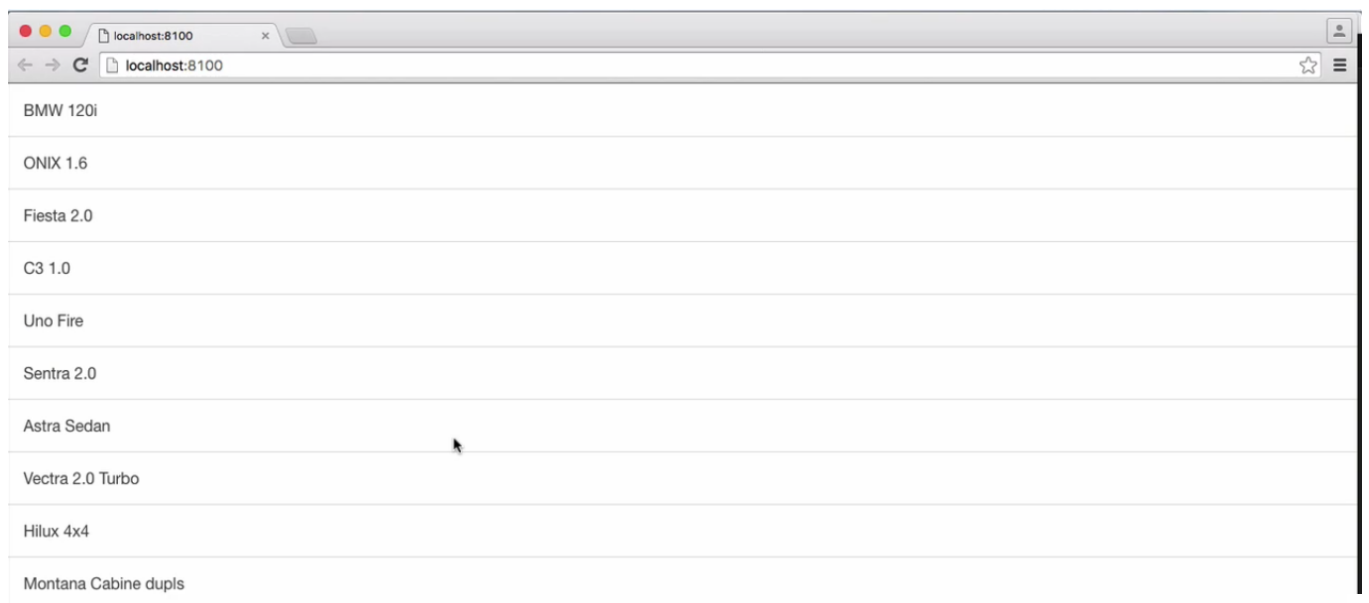
Vamos continuar com a construção da aplicação mobile. Nesta vídeo aula, queremos customizar ainda mais o aplicativo, que até o momento está assim:

Temos uma lista fornecida pela concessionário com os modelos que ela deseja vender. Mas o cabeçalho criado pelo Ionic está incomodando. O cliente quer que o nome da loja esteja presente no aplicativo. De volta ao editor de texto, faremos esta alteração no arquivo `index.html`. Dentro da tag `<ion-header-bar>`, temos a `<h1>`.

```
<ion-header-bar class="bar-stable">
  <h1 class="title">Alura Car</h1>
</ion-header-bar>
```



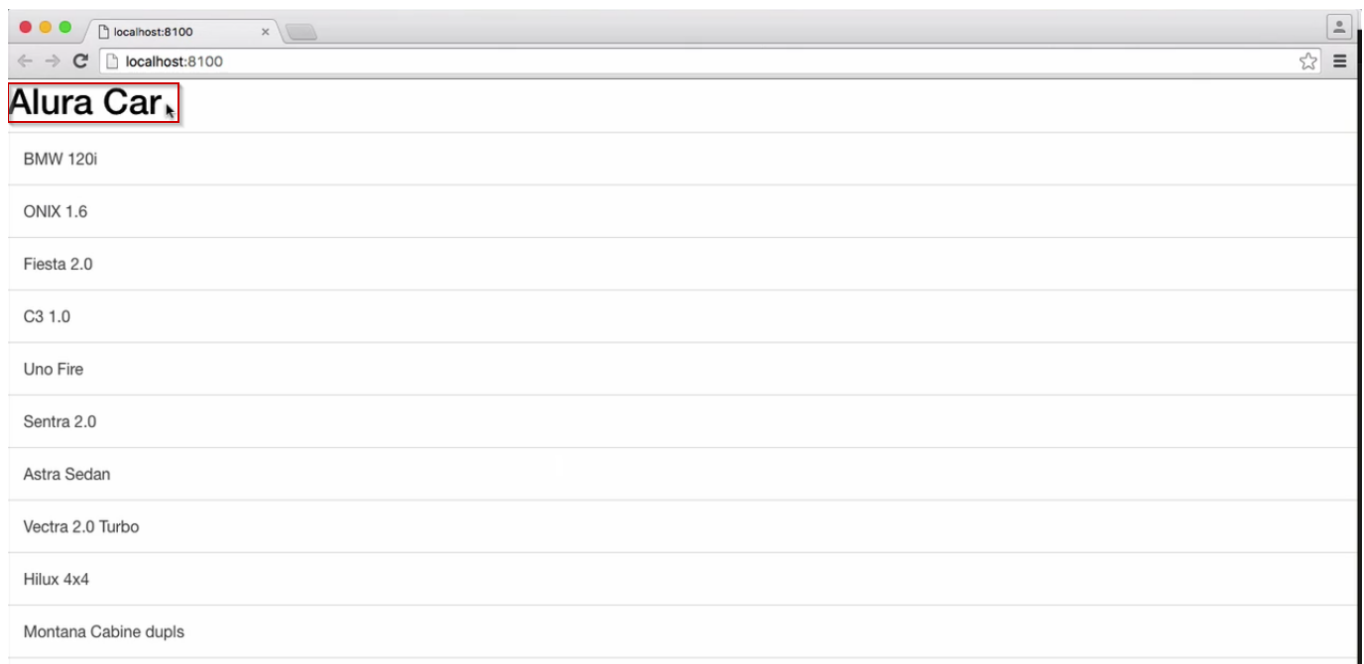
Se fizemos o teste de retirar a tag `header`, veremos que o nome da aplicação irá desaparecer.



Porque o título não estará dentro de uma tag de conteúdo. Mas se a colocarmos dentro da `<ion-content>`, o resultado será diferente.

```
<ion-content>

<h1 class="title">Alura Car</h1>
<ion-list>
  <ion-item>BMW 120i</ion-item>
  <ion-item>ONIX 1.6</ion-item>
  <ion-item>Fiesta 2.0</ion-item>
  <ion-item>C3 1.0</ion-item>
  <ion-item>Uno Fire</ion-item>
  <ion-item>Sentra 2.0</ion-item>
  <ion-item>Astra Sedan</ion-item>
  <ion-item>Vectra 2.0 Turbo</ion-item>
  <ion-item>Hilux 4x4</ion-item>
  <ion-item>Montana Cabine dupla</ion-item>
</ion-list>
</ion-content>
```



Agora, veremos o `<h1>`. Porém, perdeu as características de um `<header>` encontrado nos mobiles. O layout anterior era melhor, por isso, vamos voltar o código como estava. Nela, o nome da app tem um fundo e uma fonte mais apropriada.

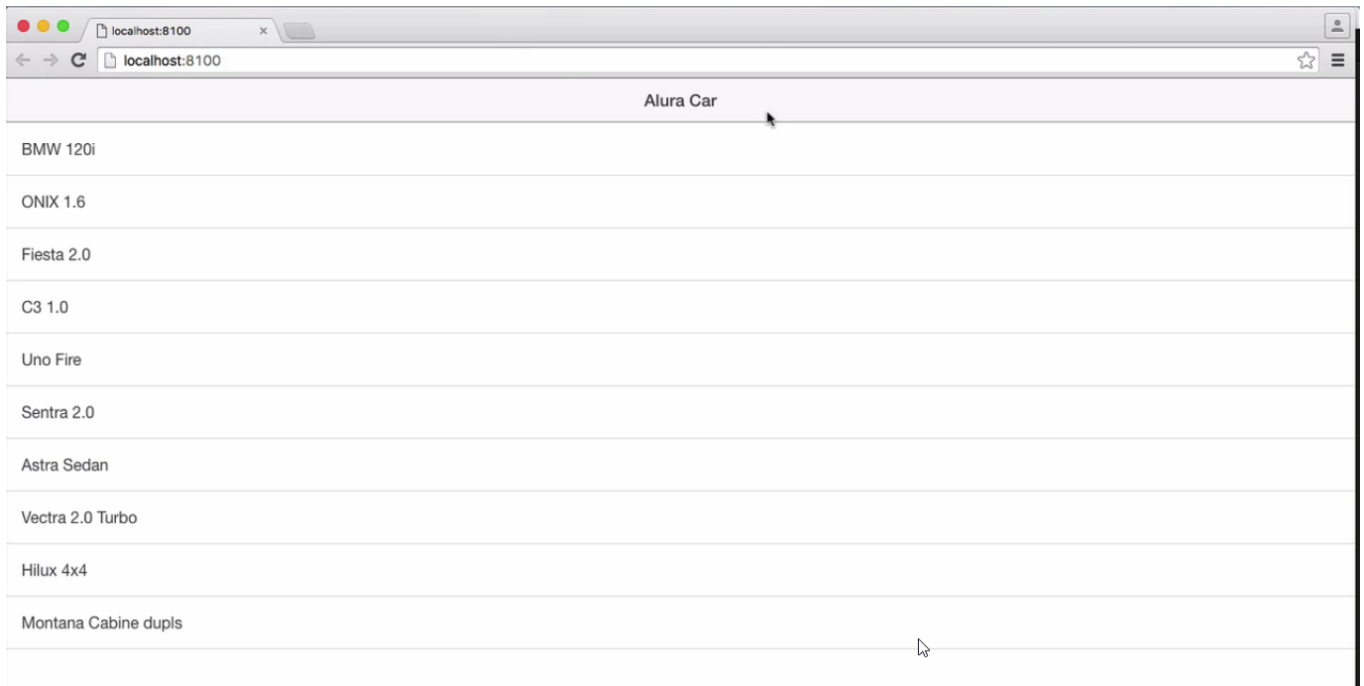
```
<ion-pane>
<ion-header-bar class="bar-stable">
  <h1 class="title">Alura Car</h1>
  <ion-list>
    <ion-item>BMW 120i</ion-item>
    <ion-item>ONIX 1.6</ion-item>
    <ion-item>Fiesta 2.0</ion-item>
    <ion-item>C3 1.0</ion-item>
    <ion-item>Uno Fire</ion-item>
    <ion-item>Sentra 2.0</ion-item>
    <ion-item>Astra Sedan</ion-item>
    <ion-item>Vectra 2.0 Turbo</ion-item>
  </ion-list>
</ion-header-bar>
</ion-pane>
```

```

<ion-item>Hilux 4x4</ion-item>
<ion-item>Montana Cabine dupla</ion-item>

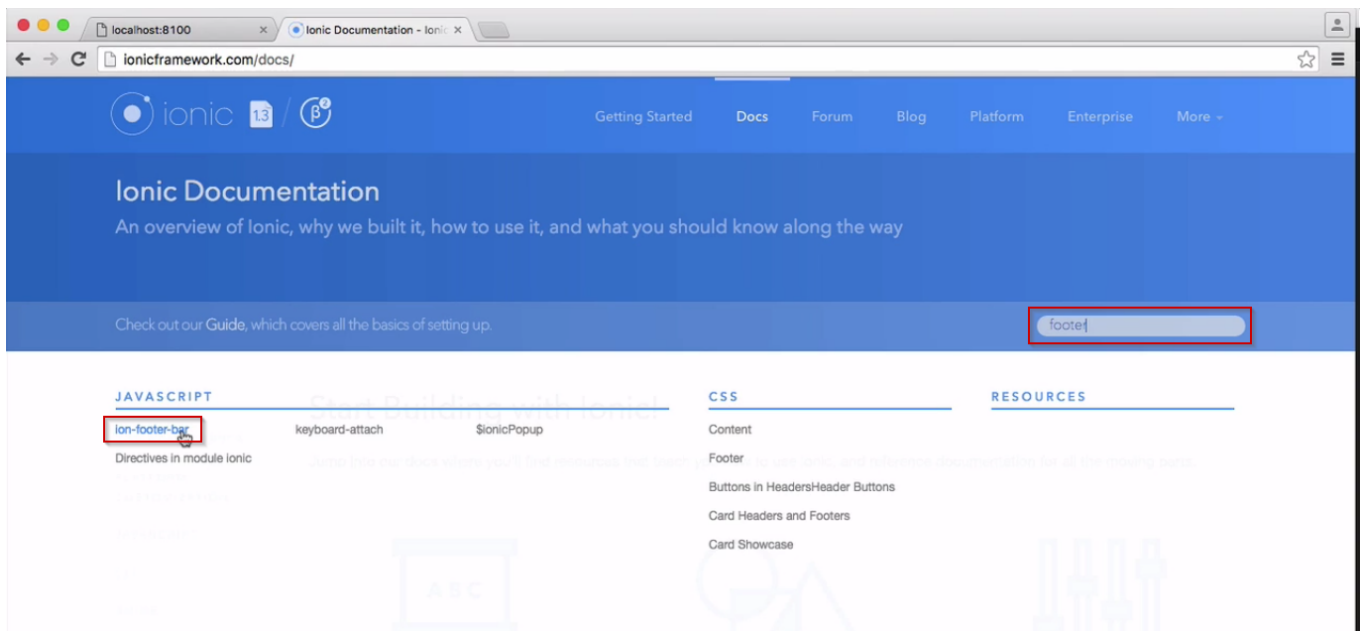
</ion-list>
</ion-content>

```



Nós precisamos também que o endereço da concessionária esteja visível em todas as páginas. Podemos fazer isto adicionando uma tag de rodapé.

Se pesquisarmos na [documentação \(http://ionicframework.com/docs/\)](http://ionicframework.com/docs/) do Ionic pelo termo "footer".



Neste link, podemos ver que ele é da árvore de JavaScript e agrupado dentro de "Headers/Footers". Também encontraremos um resumo do que é feito pela tag. Observe que o footer será adicionado abaixo do <ion-content>. Vamos adicionar no index.html as linhas correspondentes ao <ion-footer-bar>.

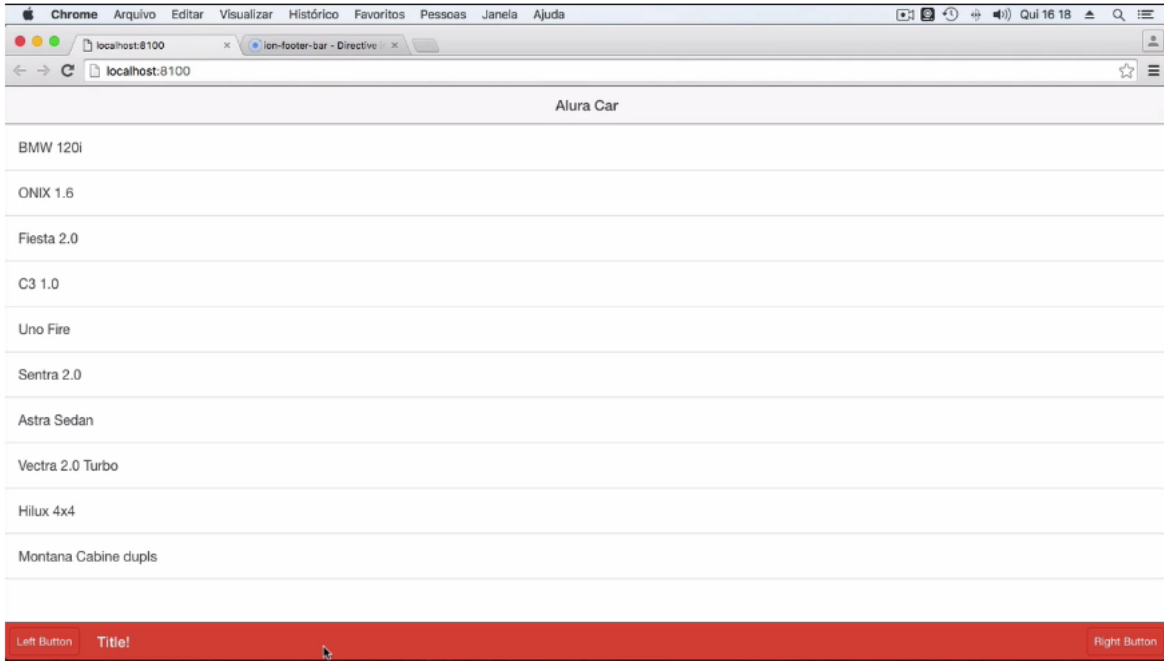
```

\...
<ion-footer-bar align-title="left" class="bar-assertive">

```

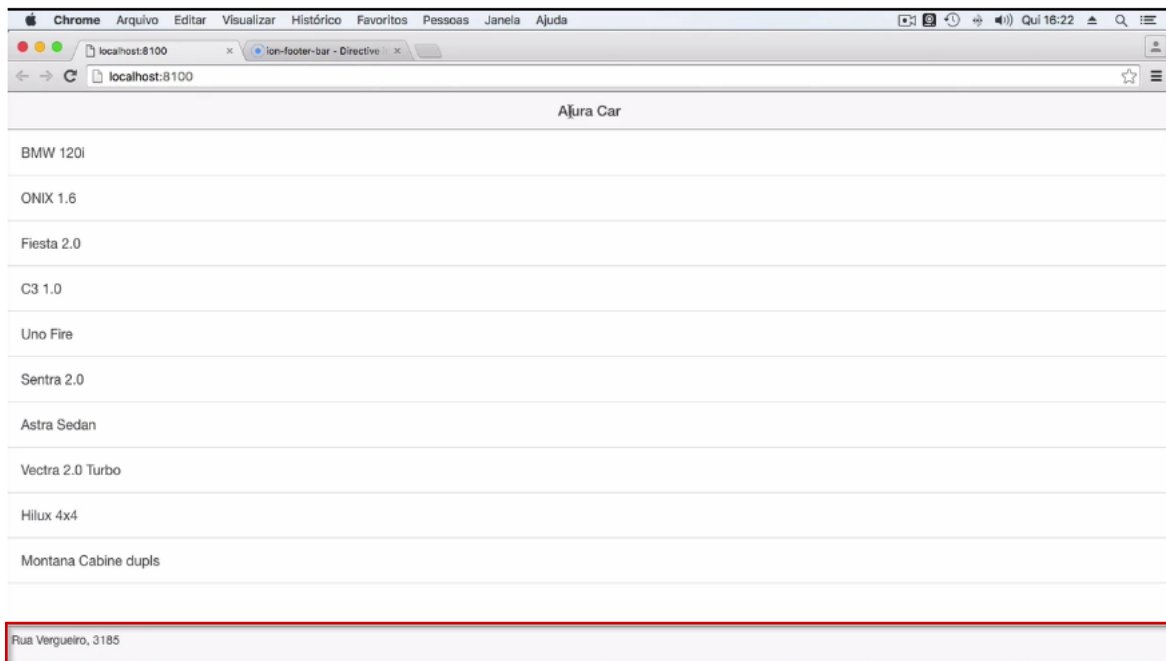
```
<div class="buttons">
  <button class="button">Left Button</button>
</div>
<h1 class="title">Title!</h1>
<div class="buttons" ng-click="doSomething()">
  <button class="button">Right Button</button>
</div>
</ion-footer-bar>
```

Veremos que a aplicação ganhará um rodapé vermelho.



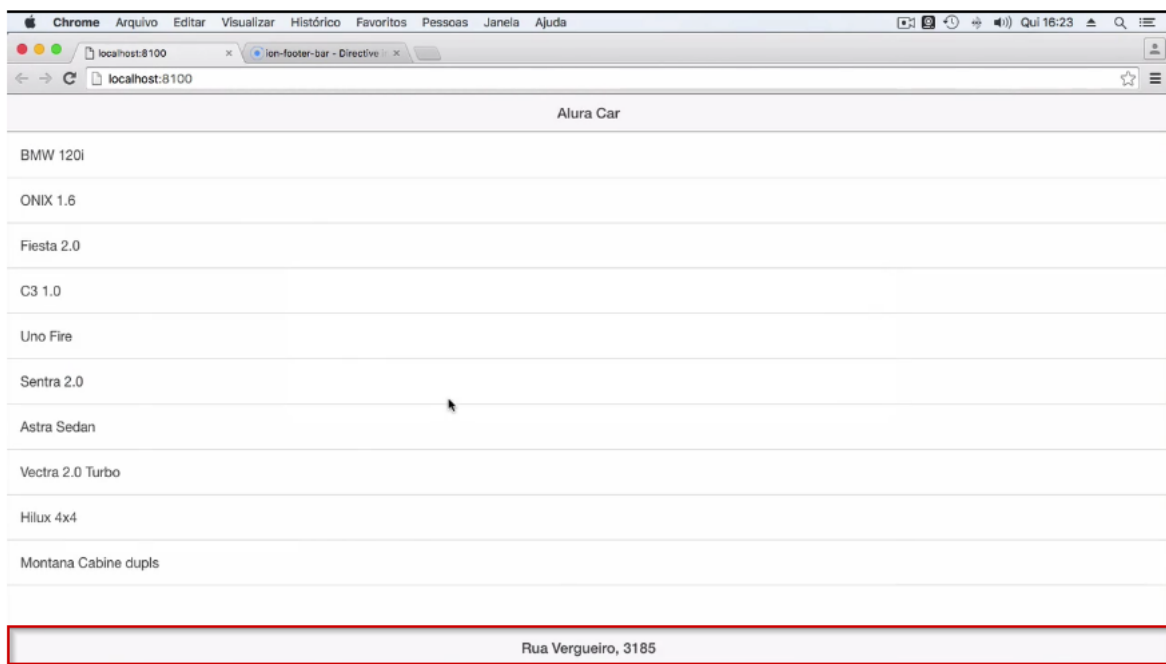
Mas ele não ficou como gostaríamos. Vamos criar com rodapé customizado. Para isto, usaremos a tag `<ion-footer-bar>` e iremos nos basear no `header` na customização do rodapé. Adicionaremos o `class="bar-stable"` e o endereço após fecharmos a tag `<ion-content>`.

```
<ion-footer-bar class="bar-stable">
  Rua Vergueiro, 3185
</ion-footer-bar>
```



Mas o rodapé ficou bem diferente do cabeçalho: a fonte, o alinhamento não estão parecidos. Por isso, vamos adicionar a tag `<h1>` e o `class="title"`.

```
<ion-footer-bar class="bar-stable">
  <h1 class="title"> Rua Vergueiro, 3185 </h1>
</ion-footer-bar>
```



Em seguida, iremos diminuir o tamanho destinado para a lista. Precisamos criar um ambiente de laboratório em que seja possível visualizar como a aplicação será exibida em qualquer dispositivo.

No Terminal, usaremos o comando `q` (que simboliza a palavra em inglês *quite*) para sairmos do servidor. Após pressionar, sairemos do servidor.

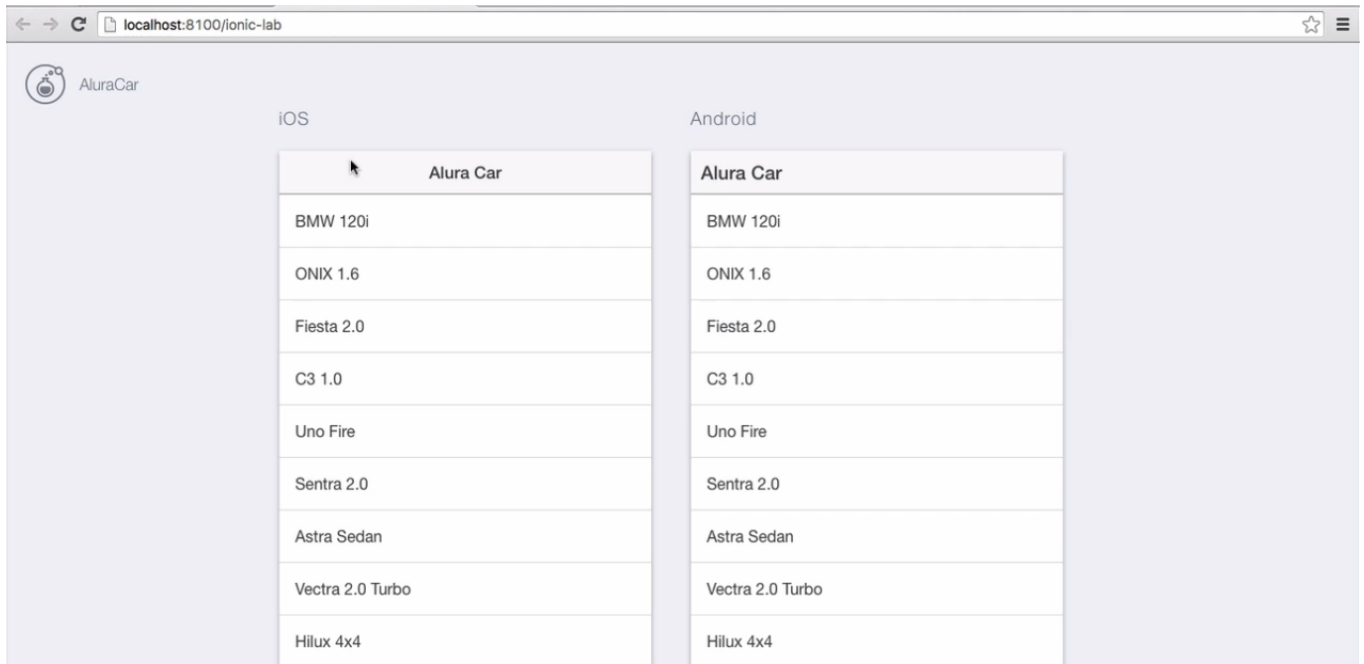
```
ionic $ q
Alura-Verde:AluraCar alura$
```

Com isso, a aplicação foi derrubada.

Vamos limpar o Terminal usando o comando `clear`. Agora, queremos que o servidor rode em modo de laboratório. Para isto, usaremos o o parâmetro `--lab`.

```
Alura-Verde: AluraCar alura$ ionic serve --lab
```

Veja o resultado que teremos:



Podemos visualizar simultaneamente como ficará tanto no iOS como no Android. Observe que existe diferença no alinhamento do nome da app de um sistema operacional para outro. Este é um ganho oferecido pelo Ionic: saber como ficará a sua aplicação após o *build*.

Nesta aula, conhecemos algumas diretivas novas referentes ao cabeçalho e rodapé. Vimos como consultar a documentação e customizar o uso de uma *tag*.