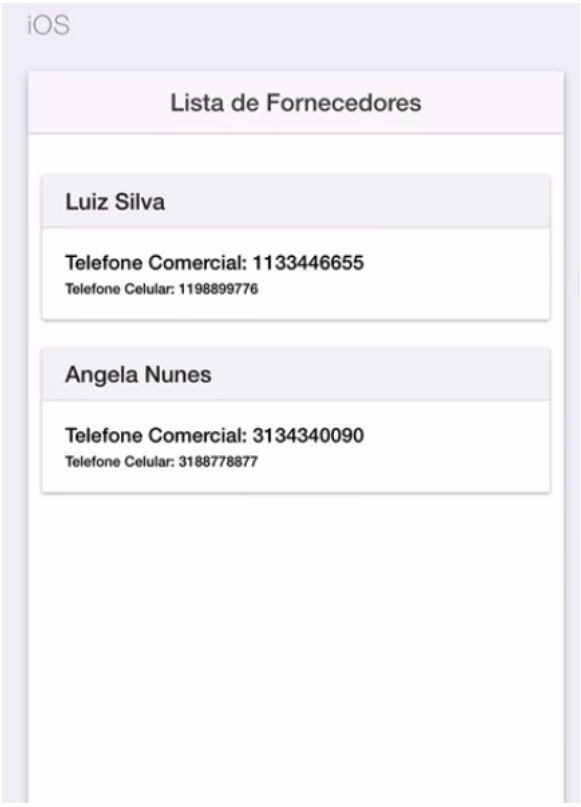
01

Desenvolvendo nossa aplicação pelo Ionic Creator

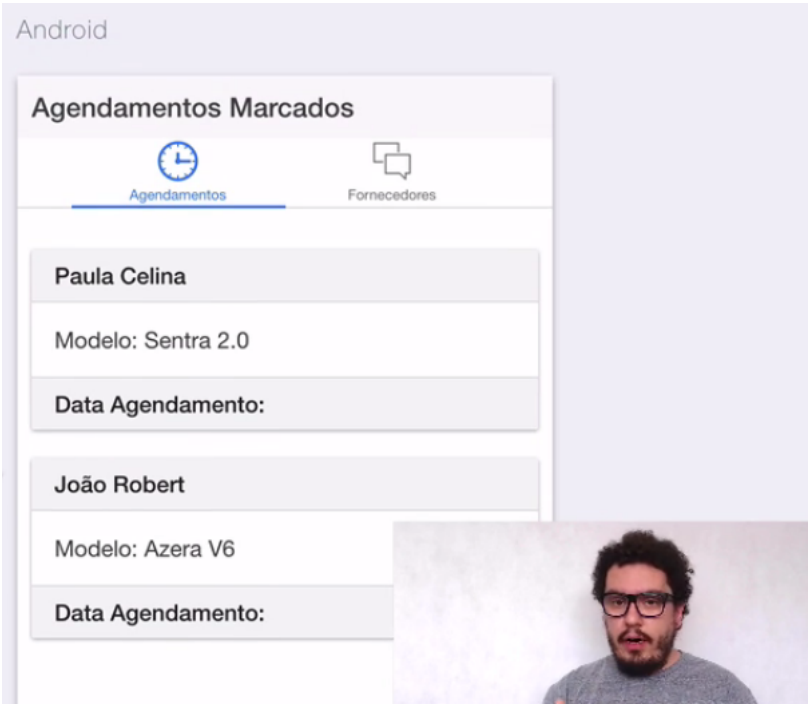
Transcrição

Nós criamos a app para o usuário. Agora, o cliente quer uma segunda aplicação, que será utilizada pelos vendedores e será mais simples. Ela terá um outro layout, usando abas.

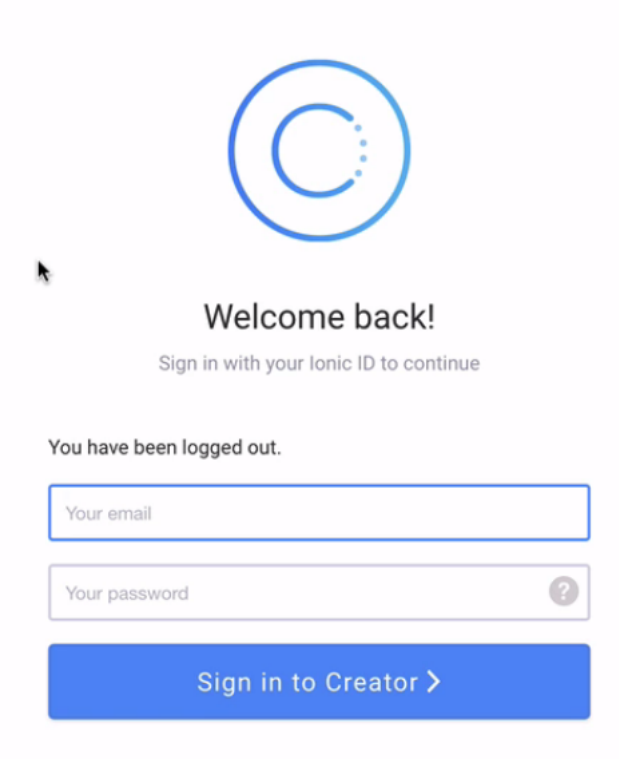
No iOS, as abas ficarão na parte inferior:



Enquanto no Android, as abas ficarão na parte de cima.

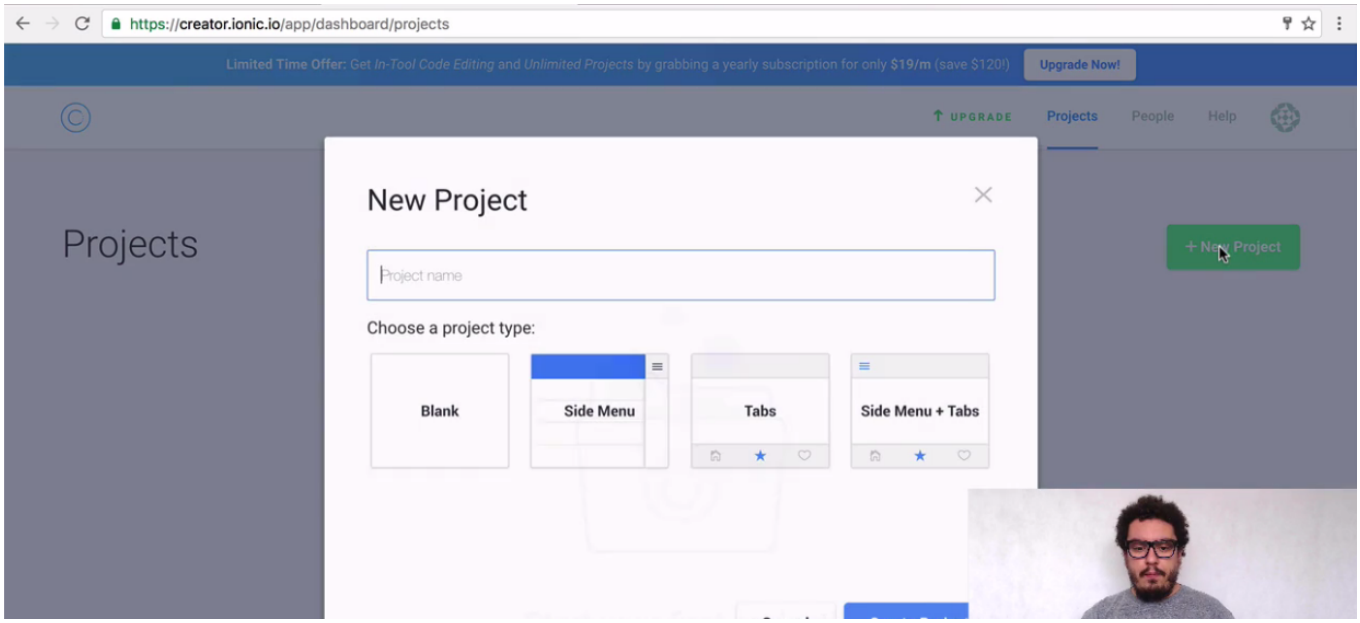


Apenas utilizando as tags do Ionic, conseguimos criar o layout para diferentes plataformas. Nós criaremos a segunda aplicação do zero, mas não pela linha de comando como fizemos anteriormente. Desta vez, usaremos o Ionic Creator (https://creator.ionic.io/app/login).

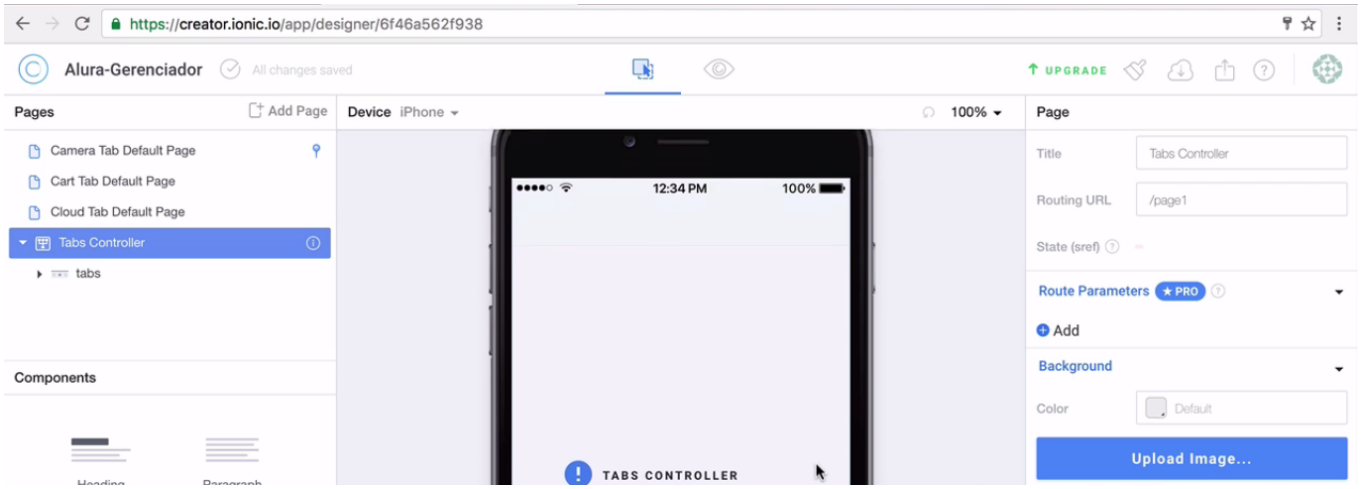


O site é dos mesmo criadores do Ionic, para você prototipar as suas aplicações. Atualmente, ele possui uma versão gratuita, em que você consegue montar um projeto - o que nos atende. Mas se você quiser criar mais projetos, você terá que pagar. Mas a maior parte das funcionalidade é gratuita.

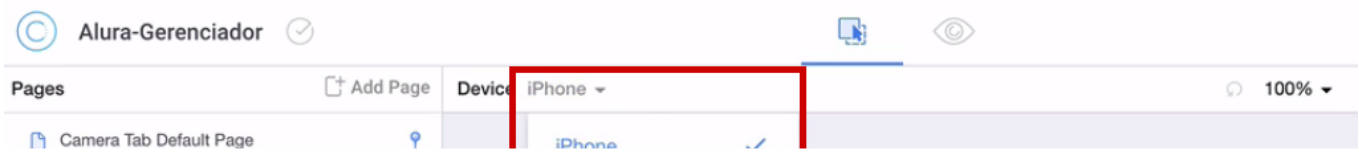
Após criarmos a nossa conta, cairemos na tela que é o dashboard principal.



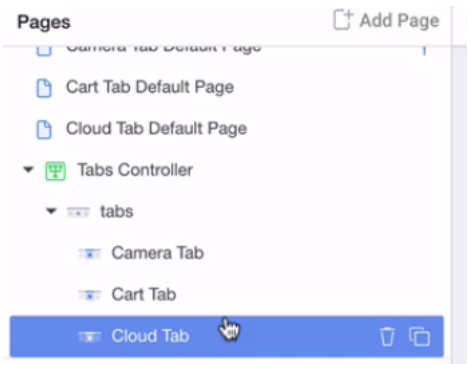
Vamos criar um novo projeto, que chamaremos de `Alura-Gerenciador`. Depois, poderemos selecionar qual template queremos usar. Nós escolheremos o "Tab", que contém abas. Clicaremos no botão `Create Project` e ele nos dará um template pronto, que poderemos customizar.



Está configurado para iOS, mas conseguimos mudar para Android.

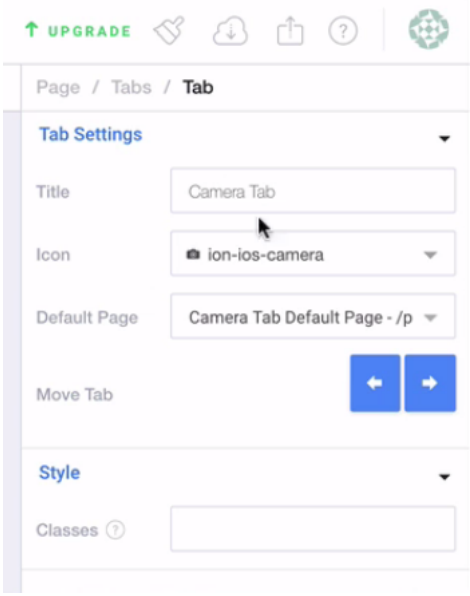


E então, as abas já surgirão no alto do aplicativo. À esquerda, temos uma parte em que podemos escolher as páginas e os componentes.



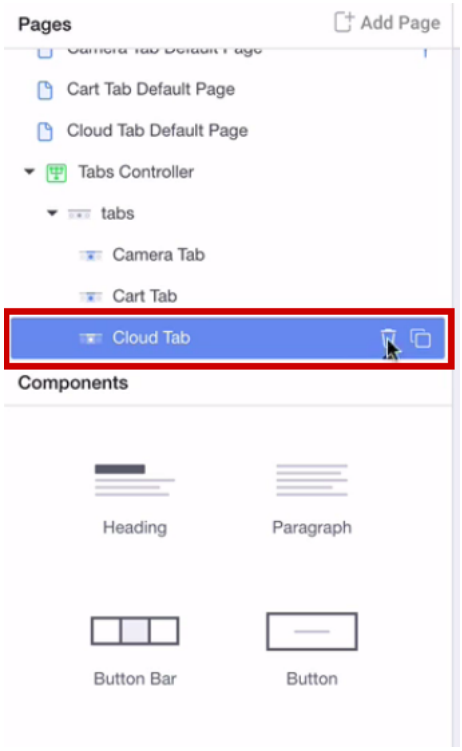
Nos componentes, basta clicar e arrastar para o aplicativo. Então, é interessante você já ter um conhecimento prévio do Ionic para utilizar o Ionic Creator. Por exemplo, no aplicativo anterior, nós construimos um card manualmente. Agora, poderemos adicionar o componente do card.

À direita, podemos configurar as abas.

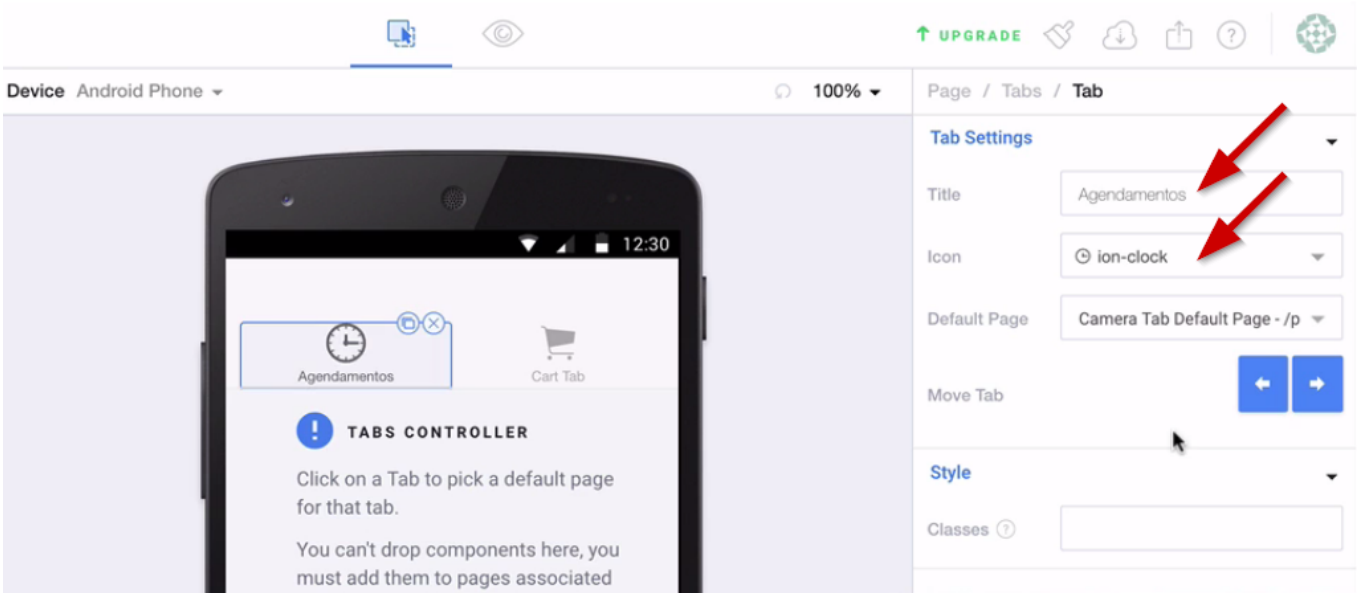


Temos a opção de selecionar os componentes da *Tab*.

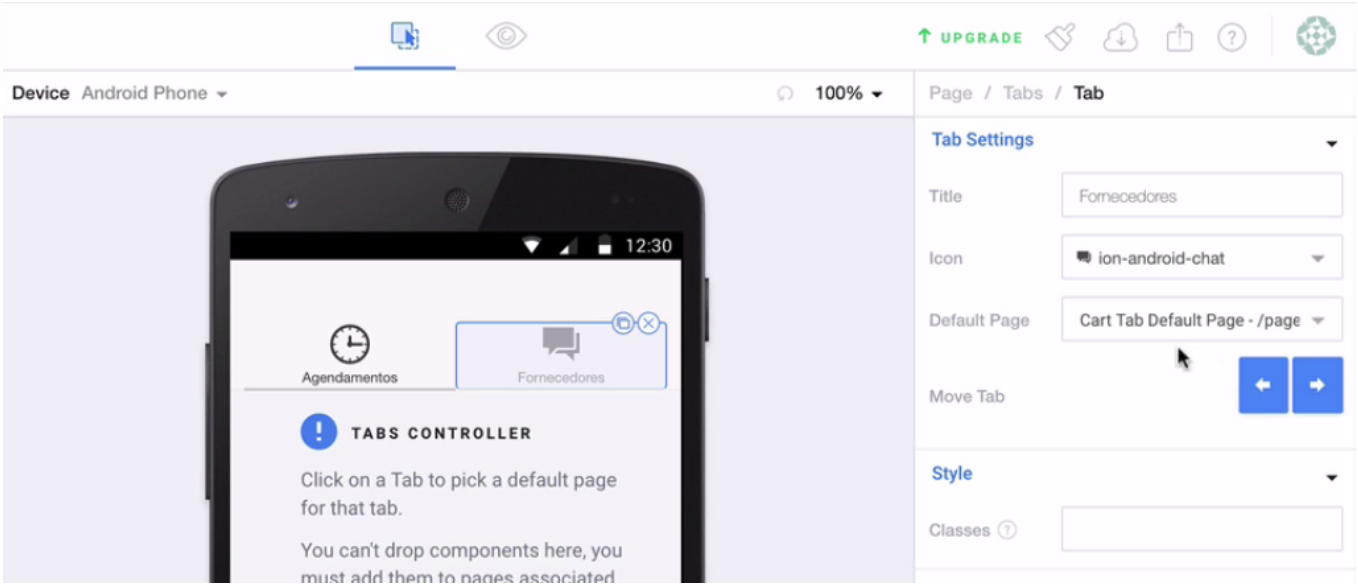
Vamos começar a customizar a app. Atualmente, temos três abas, mas queremos apenas dois. Vamos remover um na parte de páginas. Selecionaremos a aba que não queremos mais e clicaremos na lixeira.



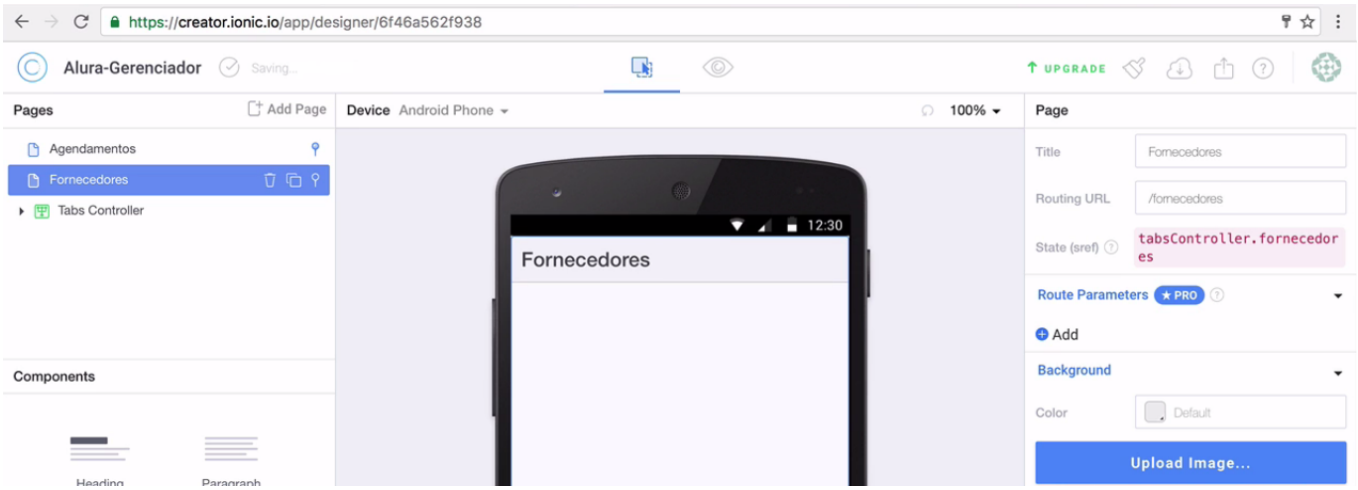
Agora teremos apenas dias abas. Na seção de abas, configuraremos que a primeira se chamará "Agendamentos" e o ícone será "Clock".



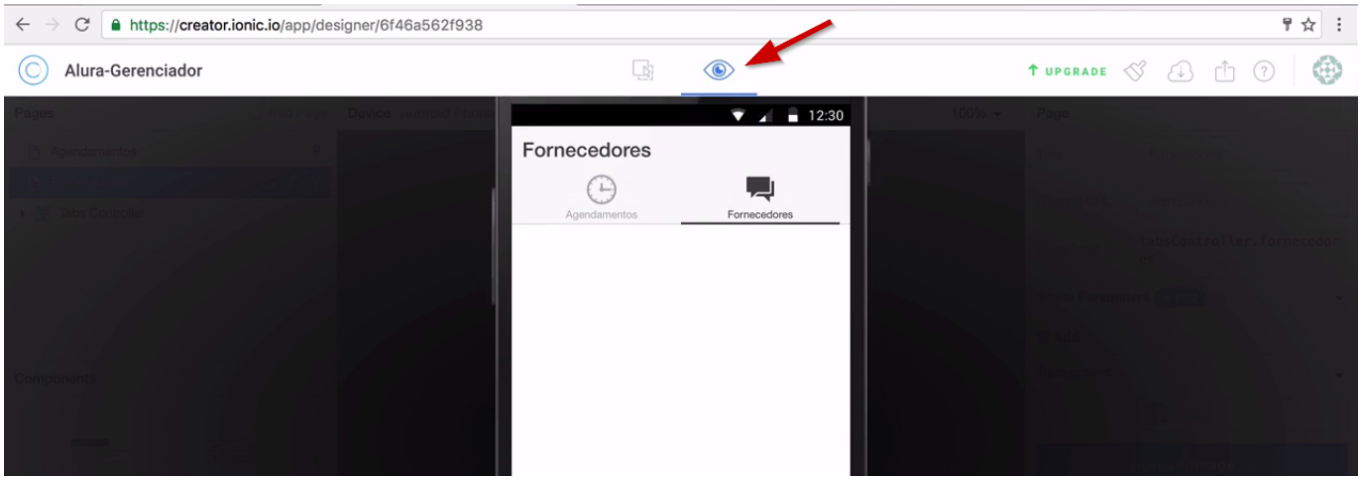
A segunda, receberá o nome de "Fornecedores" e o ícone será "ion-android-chat".



Vamos excluir uma página que era referente à Tab que já apagamos. Vamos ajustar os títulos das outras páginas para "Agendamentos" e a URL para `/agendamentos`. Renomearemos a segunda página também, no caso para "Fornecedores" e a rota para "/fornecedores".



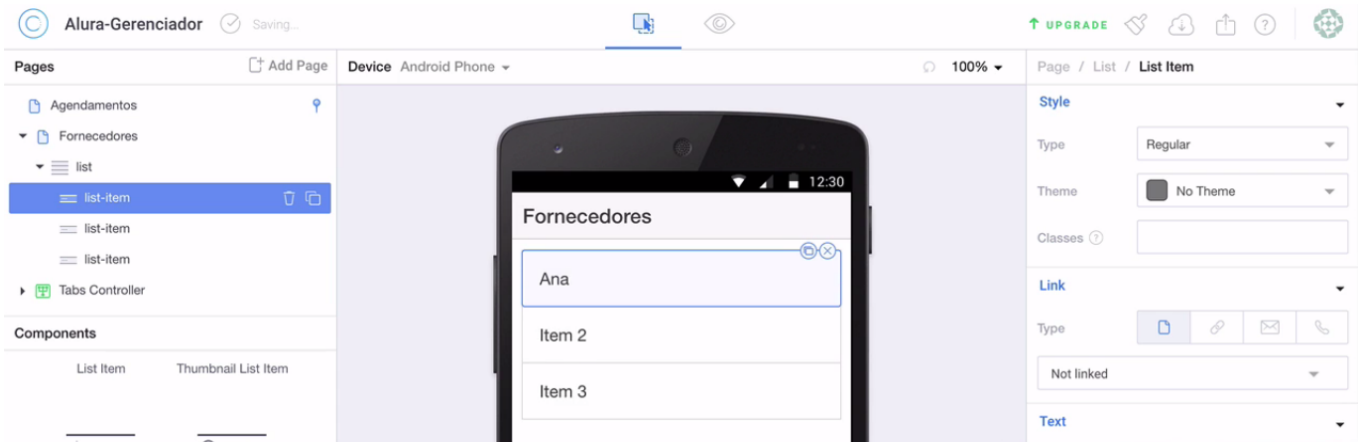
E temos a opção de ver como está ficando o nosso aplicativo.



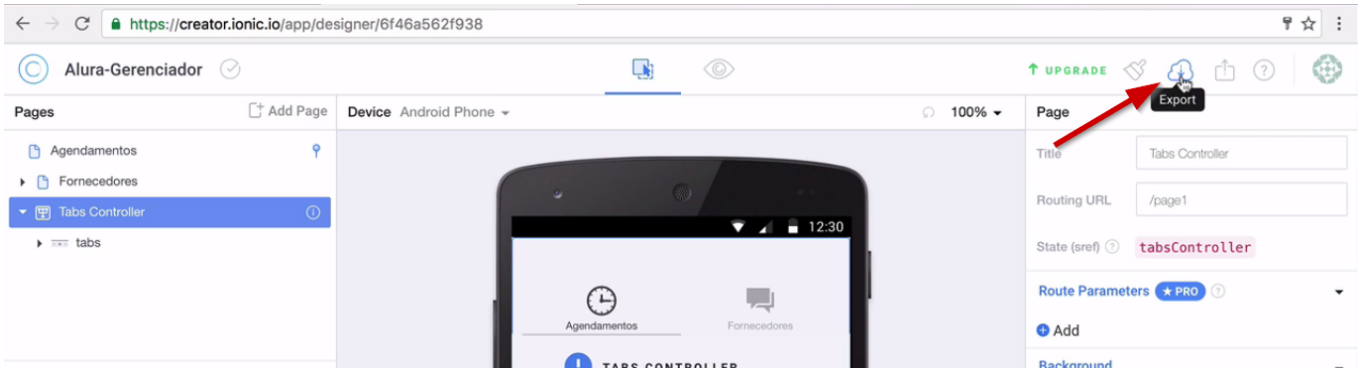
Podemos selecionar o dispositivo, por exemplo, iPhone ou um iPad.

A seguir, adicionaremos componente nestas páginas, começando pelas listas. Podemos configurar a parte de estilos. Temos ainda uma parte "Pro", com componentes que precisam ser pagos.

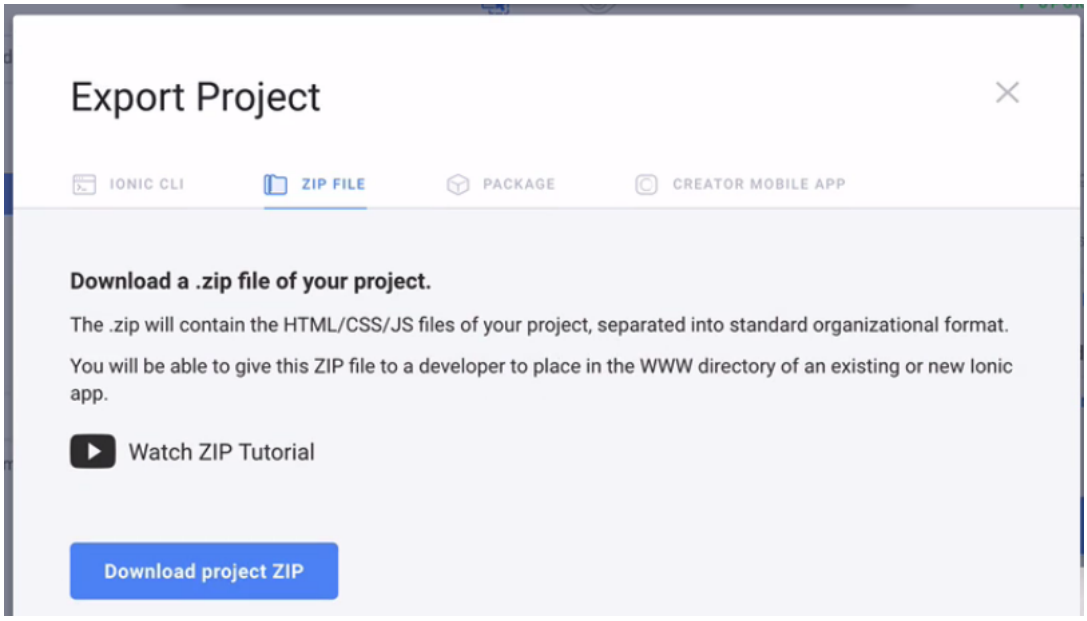
Nós temos a opção de alterar os textos dos itens. Mas nós não usaremos os textos fixos.



Já criamos o protótipo, agora, quero programar em cima dele. Para isso, iremos exportá-lo.

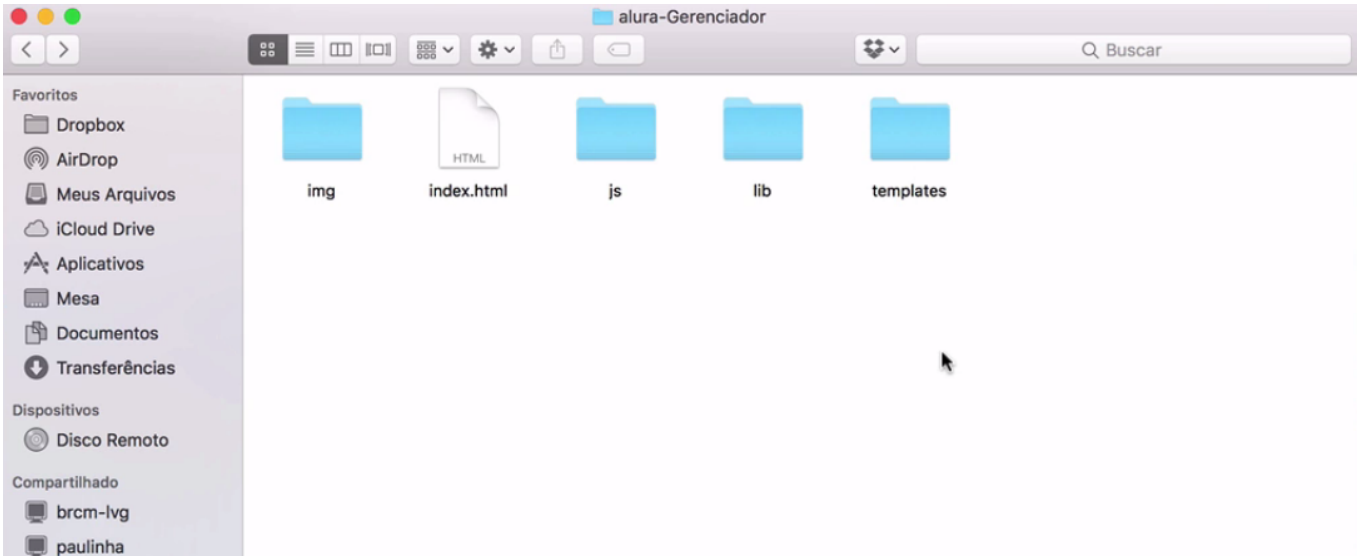


Teremos quatro opções das formas que podemos exportar o projeto. Até o momento em que gravamos a aula, o "Ionic CLI" não estava habilitado.



Mas ainda temos as opções "Zip File", "Package" e "Creator Mobile App" - ressaltando que as duas últimas são pagas. Nós usaremos a opção "Zip File".

Após fazermos o download do projeto, até termos o arquivo `alura-Gerenciador.zip` . Vamos extrair o arquivo, que não estará completo.

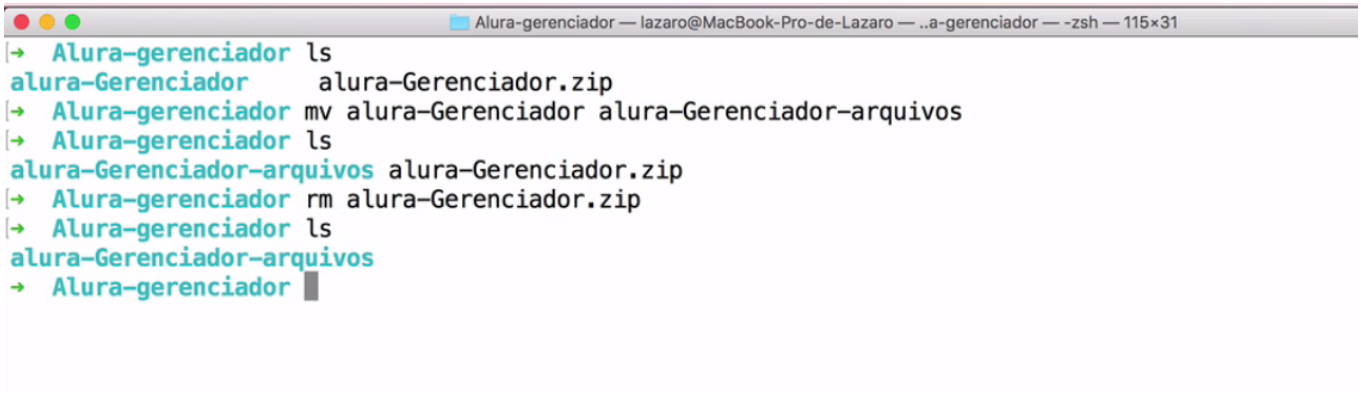


Nós vamos construir aplicação do zero, para substituir os arquivos gerados, por estes. No Terminal, vamos renomear o arquivo:

```
Alura-gerenciador mv alura-Gerenciador alura-Gerenciador-arquivos
```

Agora, podemos remover o arquivo `.zip` :

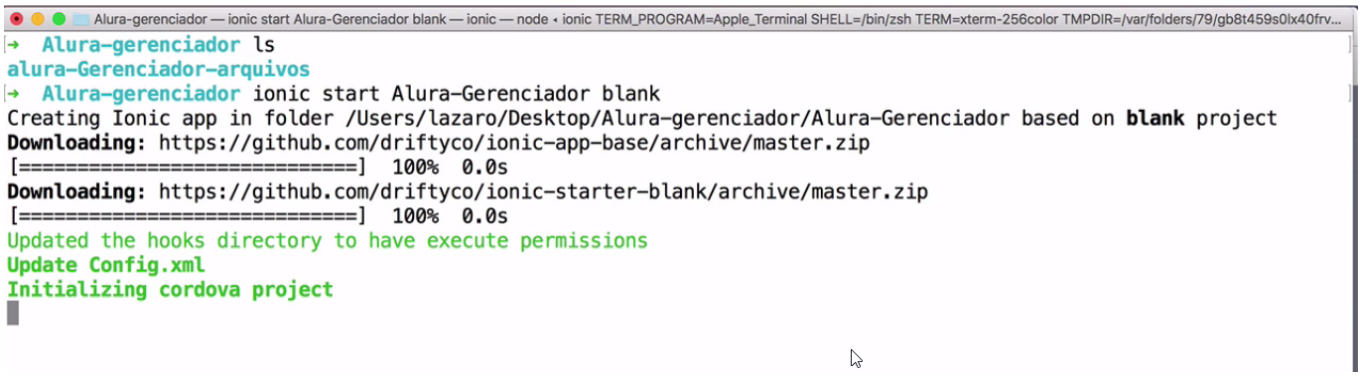
```
Alura-gerenciador rm alura-Gerenciador.zip
```



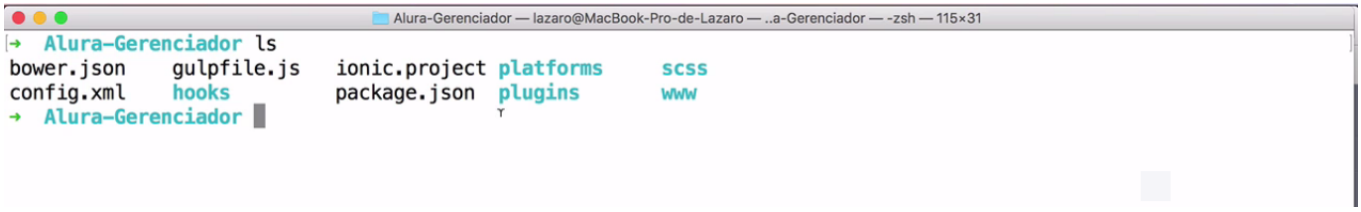
Então, o arquivo se chama `alura-Gerenciador-arquivos` e excluímos o ZIP.

Agora, criaremos uma aplicação em branco e vamos substituir os arquivos.

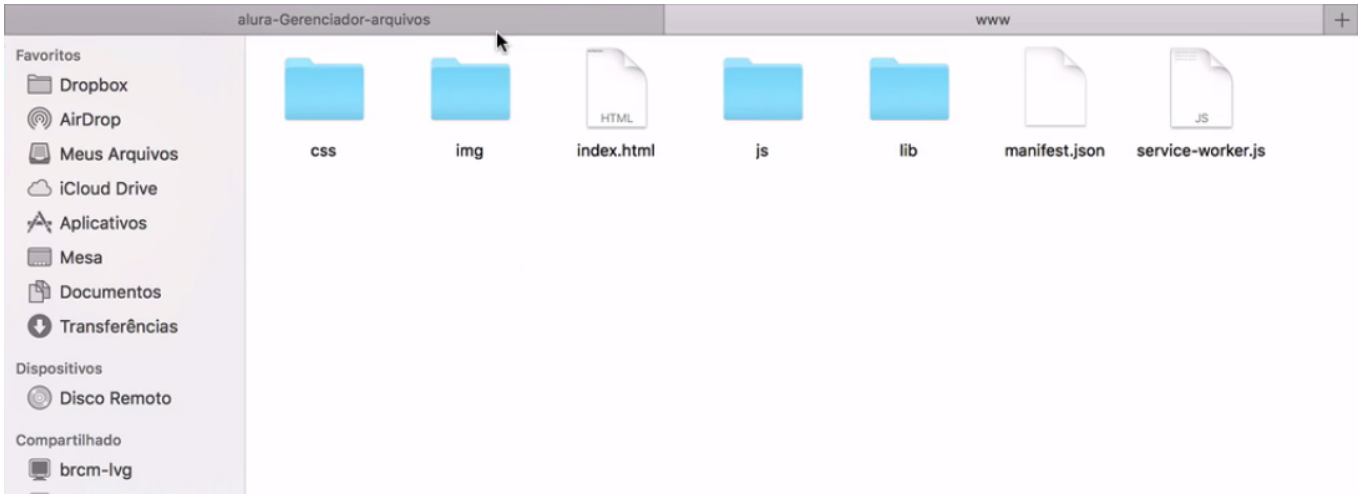
```
Alura-gerenciador ionic start Alura-Gerenciandor blank
```



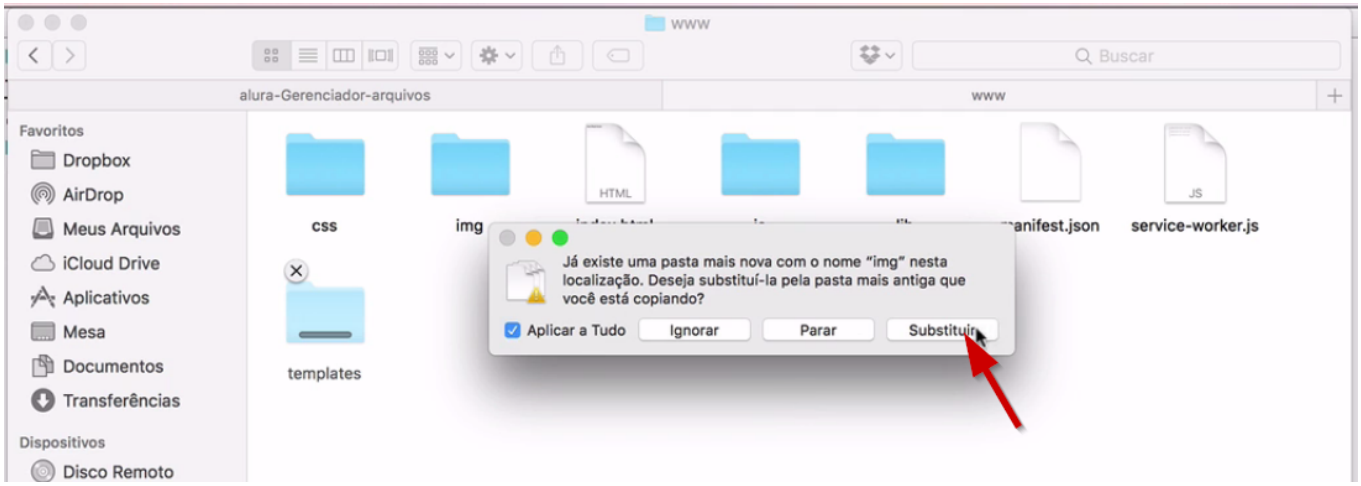
Ele irá criar a aplicação com outros arquivos.



Em seguida, vamos copiaremos os arquivos do `alura-Gerenciador-arquivos` , que foram dados pelo Ionic Creator.



E vamos colar na pasta `www` do `alura-Gerenciador` . Iremos substituir todos os arquivos:



Agora, vamos subir a aplicação no Terminal para testar:

```
ionic serve -l
```

A aplicação estará rodando.

Lembrando que não temos apenas o componente de lista, mas vários outros. Só o componente que nos permite usar o Google Maps é pago.