

01

Aplicando restrições nos elementos

Transcrição

De volta ao nosso projeto, identificamos que nosso layout não se adequa ao rodarmos o app no simulador de dispositivos com tamanho de tela maior, como um iPad. A *Label* com o título do projeto está muito pequeno, e as *Views* estão muito distantes uma da outra. Então, começaremos a seguir um novo gabarito para implementação, com várias especificações de tamanhos e fontes que precisarão ser aplicados em nosso projeto.

A *Label* "alura viagens", no topo do layout, terá corpo de 31 pt — mas será que se alterarmos isto no *storyboard*, a modificação também funcionará em um iPhone?

A resposta é **sim**, portanto é necessário encontrar uma forma de mantermos a formatação que funciona para o iPhone e criarmos uma customização para o iPad. Assim, conseguimos manter o que já criamos sem deixarmos de atender a outros tamanhos de tela. Como faremos isso?

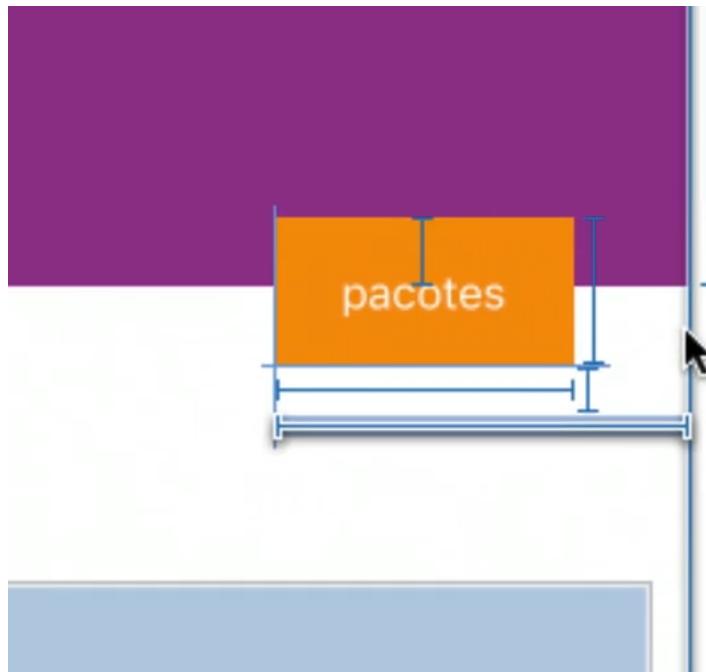
Ao selecionarmos uma *Label*, por exemplo, e mudarmos o tamanho da fonte, como neste caso, ao lado da opção "Font" há um botão com sinal de +, que serve para adicionarmos uma **customização**. Vamos clicar nele e em "Add Variation" ("Adicionar variação"), o que habilitará mais uma configuração de fonte, sendo a primeira a padrão, definida anteriormente, e esta a que acabamos de criar.

Nela, definiremos como "Style" Medium, e "Size" com 31. Feito isso, basta clicarmos em "Done". Ao encontrarmos esse símbolo de +, significa que conseguimos adicionar variações para cada elemento existente no *storyboard*. Neste sentido, o *Header* no simulador do iPad também não parece adequado e, segundo o gabarito, sua altura deve ser de 250, e não 145, como está configurada no momento.

Selecionaremos o *Header*, e se clicarmos no + ao lado de "Constant" com 145, o *popover* que se abre indica que tanto "Width" (largura) quanto "Height" (altura) estão como Regular. Tal padronização é encontrada na [documentação da Apple](#) (<https://developer.apple.com/library/archive/featuredarticles/ViewControllerPGforiPhoneOS/TheAdaptiveModel.html>), no entanto não precisamos nos preocupar em decorar essas nomenclaturas.

Clicaremos em "Add Variation" e alteraremos o valor de "wR hR" para 250. Isto aumentará a altura do *Header* quando o app estiver rodando em um iPad.

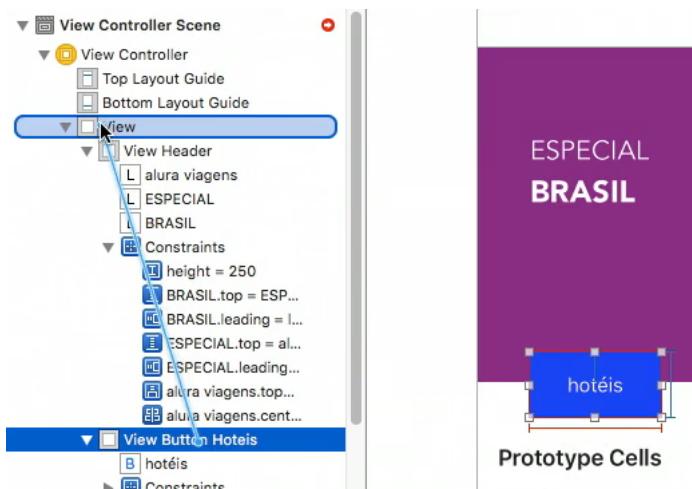
Passaremos aos ajustes das *Views*, que deverão ser posicionadas mais ao centro do layout. Entre a *Label* de "hotéis" e a extremidade esquerda, haverá uma *Constraint* de 150 mas, antes de corrigirmos isso, reparem que no *storyboard* há uma *Constraint* que vem da extremidade direita até o início da *View* "pacotes".



Isso é um pouco estranho, pois idealmente a restrição deveria ir do fim da *View* até a margem direita. Sendo assim, selecionaremos esta *Constraint* e a deletaremos, ao que o programa irá apontar um erro pela falta de restrição.

Existe uma forma ainda melhor de incluirmos estas *Constraints*. Neste caso, como queremos uma restrição que saia da lateral direita da *View* do botão em relação à tela, e não ao *Header*, no painel à esquerda selecionaremos "View Button Pacotes" mantendo a tecla "Ctrl" pressionada, e então arrastaremos o mouse à "View", hierarquicamente superior. Ao soltarmos o cursor, escolheremos "Trailing Space to Container Margin".

Em seguida, incluiremos uma variação (clicando no sinal de + e em "Add Variation"), com "wR hR" de 150 . Faremos o mesmo com a outra *View* (o botão de "hotéis"), deletando a *Constraint* que estava antes e criando outra:



Lembrando que desta vez optaremos por "Leading Space to Container Margin", que se refere à margem esquerda. Adicionaremos uma variação cujo valor será 150 também. Falta alterarmos os tamanhos das nossas *labels* de "ESPECIAL" e "BRASIL", e mudarmos a largura e a altura dos botões para 150 e 75 , respectivamente, que é o que faremos agora.

No *storyboard*, selecionaremos as *Constraints* correspondentes à altura e largura, e acrescentaremos as variações com os devidos valores. Deste modo, os botões ficarão com um tamanho bem maior, porém seus textos precisam ficar sempre centralizados na *View*. Para isso, há duas possibilidades:

- selecionar a caixa de texto, e com "Ctrl" pressionado, arrastar o mouse até a extremidade direita da View do botão, e clicar em "Center Vertically in Container". Fazer o mesmo clicando e arrastando o mouse para a base do botão, e selecionando "Center Horizontally in Container". Em seguida, é preciso atualizar a posição do botão, clicando no símbolo de triângulo amarelo de *warning* e em "Fix Misplacement";
- clicar no ícone de "Add New Alignment Constraints" e marcar os checkboxes de "Horizontally in Container" e "Vertically in Container", e apertar o botão "Add 2 Constraints". Neste caso também é necessário clicar no símbolo de triângulo amarelo.

Continuando, as *views* não estão alinhadas verticalmente com a base do *Header*, como deveriam estar. Se clicarmos na *Constraint* superior do botão "pacotes", por exemplo, teremos que "Constant" está definido com -23 , que é o quanto a restrição está acima da base, a posição final do *Header*.

Sendo a variação de 75 , idealmente deveríamos pegar a metade disso e definir como *Constraint*, para que o botão fique centralizado verticalmente em relação à base do cabeçalho. Por isso, incluiremos outra variação, e alteraremos o valor -23 para -37.5 (metade de 75). Repetiremos o procedimento no botão "hotéis".

Vamos testar a visualização do layout no iPhone 6?

As dimensões configuradas para iPhone se mantiveram, e se voltarmos à visualização do iPad, também obteremos as configurações que definimos. Ou seja, agora, sim, nosso app está responsivo. Para terminar, vamos continuar mexendo em nossas *labels*. Em "ESPECIAL" colocaremos a fonte Avenir em Book e tamanho 40 pt, e em "BRASIL", Avenir tipo Black e tamanho 40 pt.

Não podemos esquecer de acrescentar uma variação para essas configurações, caso contrário acabaremos alterando as definições para telas e dispositivos menores!

Aplicamos os designs que precisávamos sem interferirmos na responsividade do projeto, tivemos um grande progresso em customizações, e lidamos com especificidades de acordo com as telas em uso. Seguiremos fazendo outros ajustes.