

FIAP

FIAP GRADUAÇÃO

Desenvolvimento Mobile - iOS

Xcode com Objective-C

PROF. AGESANDRO SCARPIONI
AGESANDRO@FIAP.COM.BR

Diferentes iPhones com diferentes tamanhos

- Temos no mercado **quatro** tipos de telas para iPhones.
- iPhone 4s (3.5 polegadas), iPhone 5/5s (4 polegadas), iPhone 6 e 7 (4.7 polegadas) e iPhone 6 Plus e 7Plus (5.5 polegadas).
- A Apple recomenda a criação de apenas uma Storyboard para cada categoria de aparelho. Então como faremos para adaptar nosso aplicativo de uma vez só para as 4 telas?

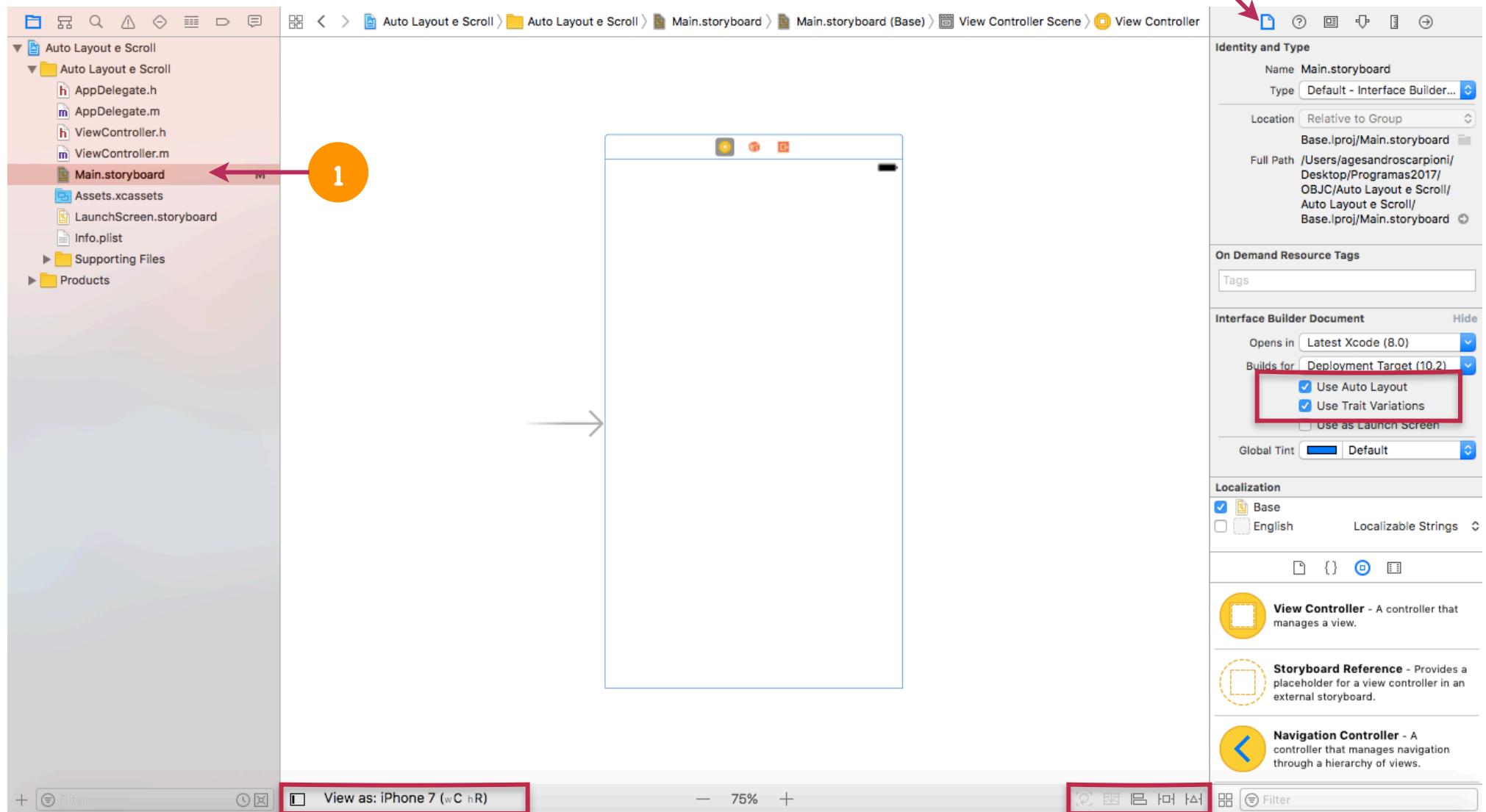
iPhones compatíveis com iOS 8 ou posterior



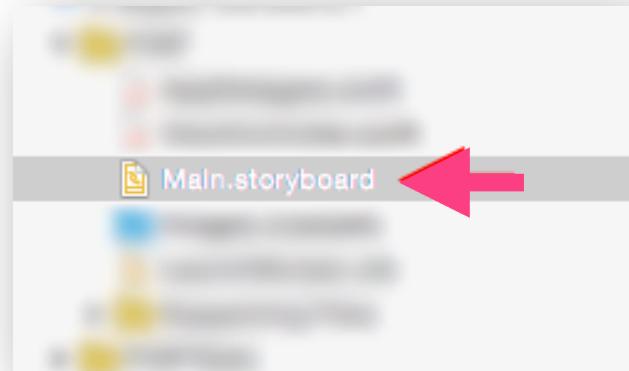
|Auto Layout

- Para economizar tempo e reduzir o tamanho do projeto, utilizaremos uma ferramenta integrada ao Xcode: **Auto Layout**.
- A partir desta ferramenta, conseguimos otimizar o conteúdo do nosso aplicativo para todas as telas de iPhones disponíveis no mercado, compatíveis com o iOS 8 ou posterior.
- O Auto Layout é nativo do Xcode, e está localizado na parte **inferior direita** do Main.storyboard.
- O Auto Layout sempre virá habilitado ao criar um novo projeto. É altamente recomendado que você não desabilite este recurso.

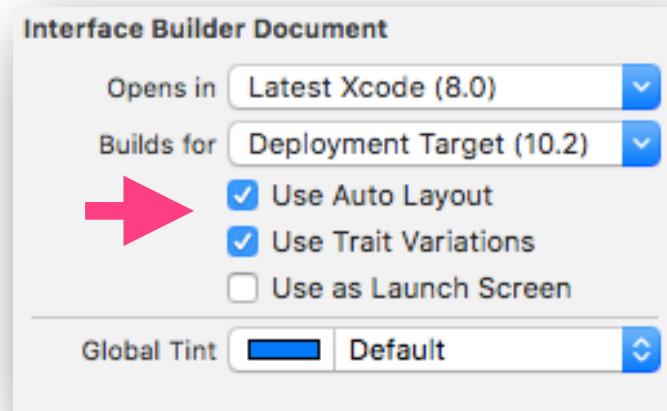
Localizando o Auto Layout no Xcode



Localizando o Auto Layout no Xcode



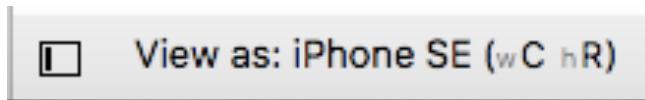
Os componentes do Auto Layout sempre serão visíveis na tela do Main.storyboard.



Para **habilitar** ou **desabilitar** o recurso, basta marcar ou desmarcar as duas opções destacadas na imagem.

Ajustando o tamanho da UIViewController

Para ajustar o tamanho da UIViewController de acordo com o tamanho do aparelho, basta clicar nesta opção.



Todas as telas disponíveis.

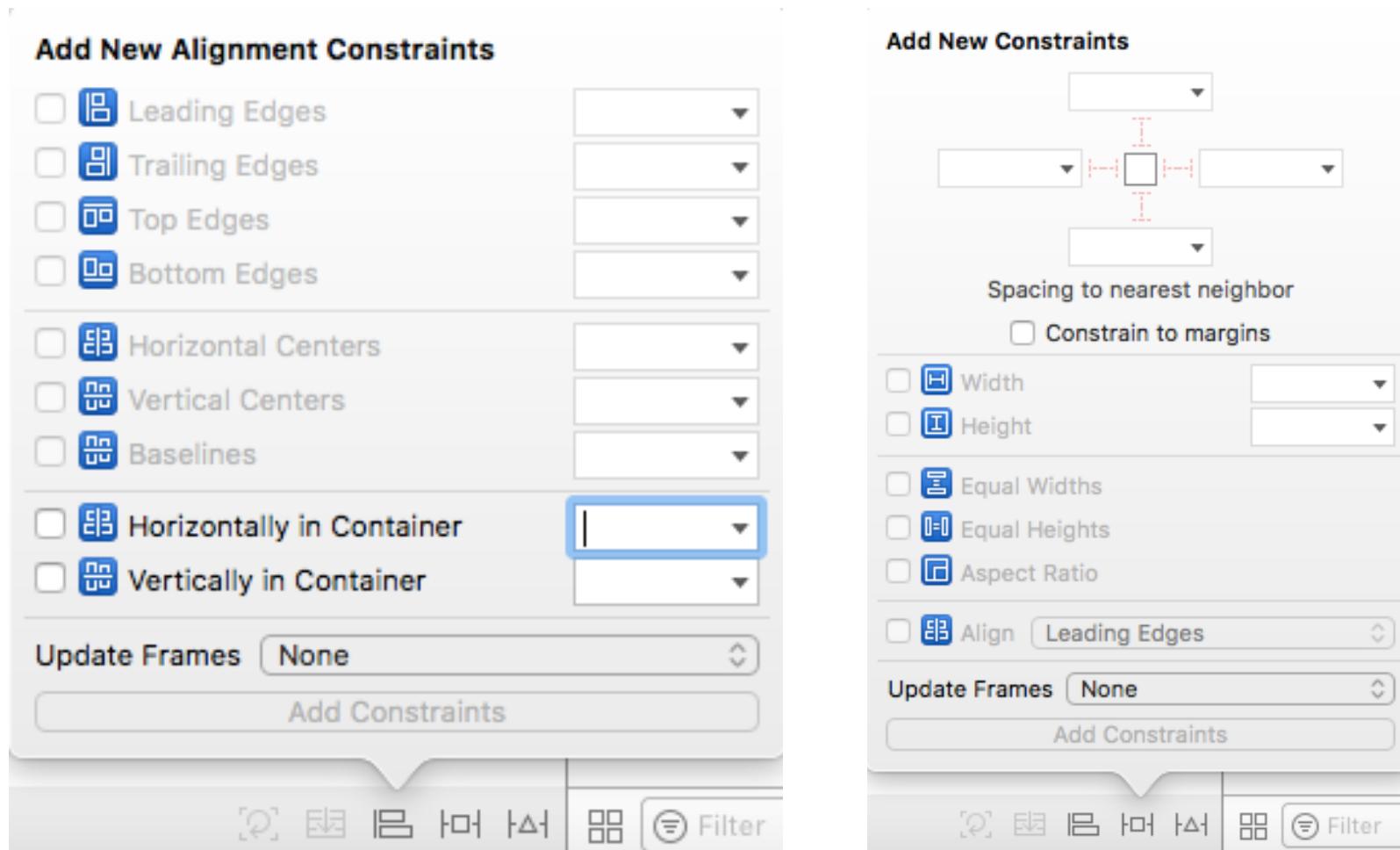
Ferramentas principais para Constraints



Para criar novas Constraints, alinha-las ou atualizar suas posições na tela, utilize destas ferramentas em destaque.

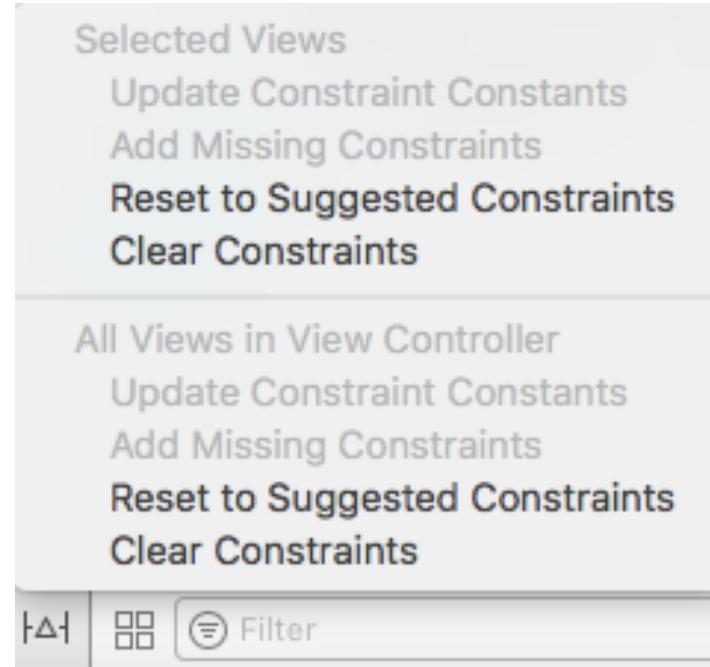
Atenção: As duas primeiras ferramentas em destaque são destinadas apenas para a criação de novas Constraints. Se for necessário alterar tamanhos, posições ou alinhamentos, você deve realizar essas possíveis alterações a partir das propriedades do objeto.

Ferramentas principais para Constraints



Ferramentas para novos alinhamentos. Ferramentas para novas Constraints.

Ferramentas principais para Constraints



Caso você tenha alterado alguma posição e não houveram mudanças visíveis, utilize este painel para atualizar os componentes presentes na tela. Este painel também é útil para adicionar de maneira automática as Constraints que estão faltando no objeto.

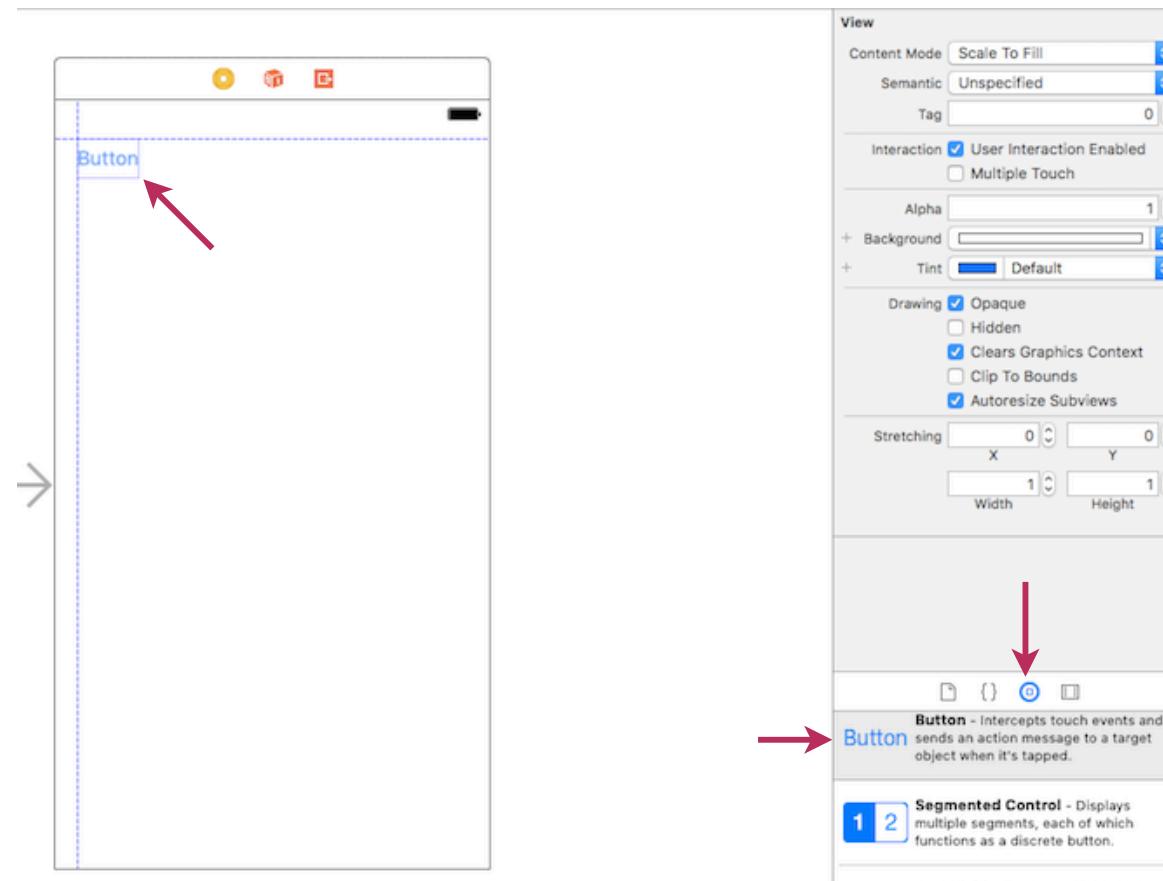


Exercício Simples 1

Criando Constraints para um UIButton.

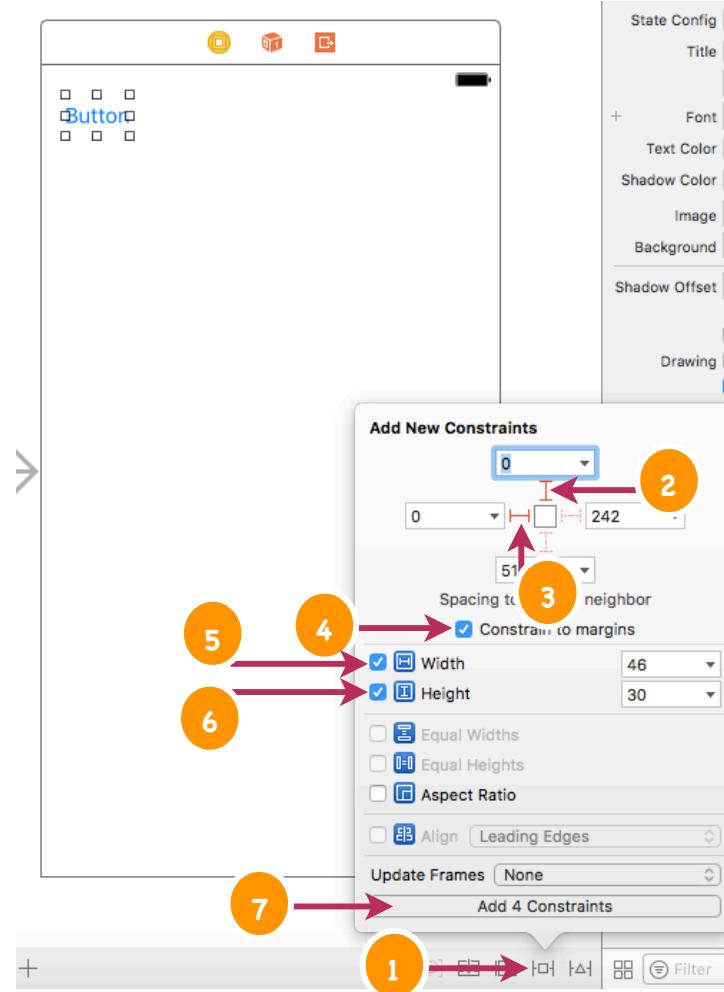
Passo 1: Adicionando um objeto a UIViewController

- Arraste um UIButton para a UIViewController. Depois, posicione o botão na região superior esquerda da tela, até que as margens de segurança indiquem a posição correta.



Passo 2: Adicionando Constraints ao objeto

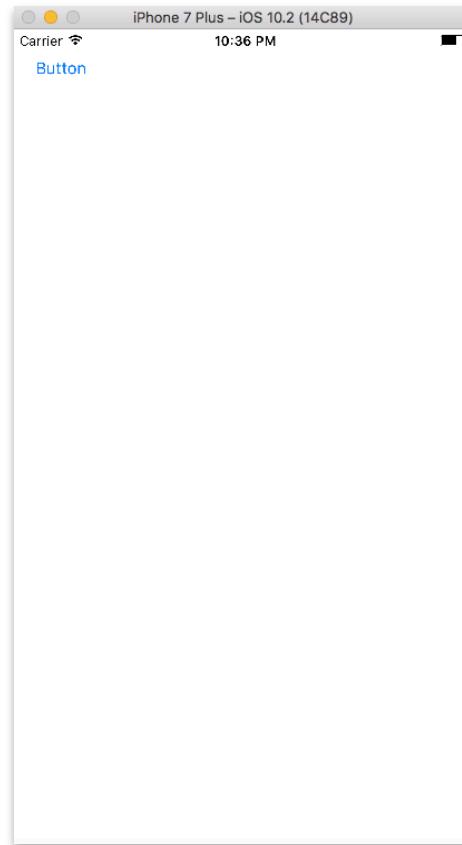
- Clique no botão inferior da tela e abra o painel para criação de **novas Constraints**, neste caso, para o UIButton. Siga a sequencia abaixo:



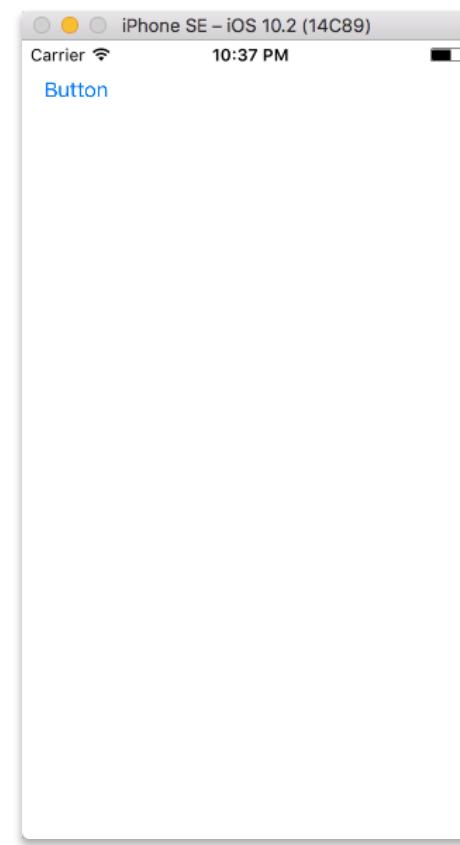
- 1: Abra o painel para **novas Constraints**.
- 2: Clique na linha pontilhada superior, para informar que o botão deve ficar colado a margem de segurança superior.
- 3: Clique na linha pontilhada esquerda, para informar que o botão deve ficar colado a margem de segurança esquerda.
- 4: Marque a opção “**Constrain to margins**” para informar ao Xcode que o objeto respeitará as margens de segurança.
- 5/6: Defina os valores para largura e altura. O Xcode recomendará os valores de acordo com o tamanho do botão.
- 7: Adicione as 4 Constraints ao objeto selecionado.

Passo 3: Conclusão

- Depois de adicionar as 4 Constraints, execute o seu aplicativo em diferentes telas no iOS Simulator para testar as posições.



iPhone 7 Plus



iPhone SE

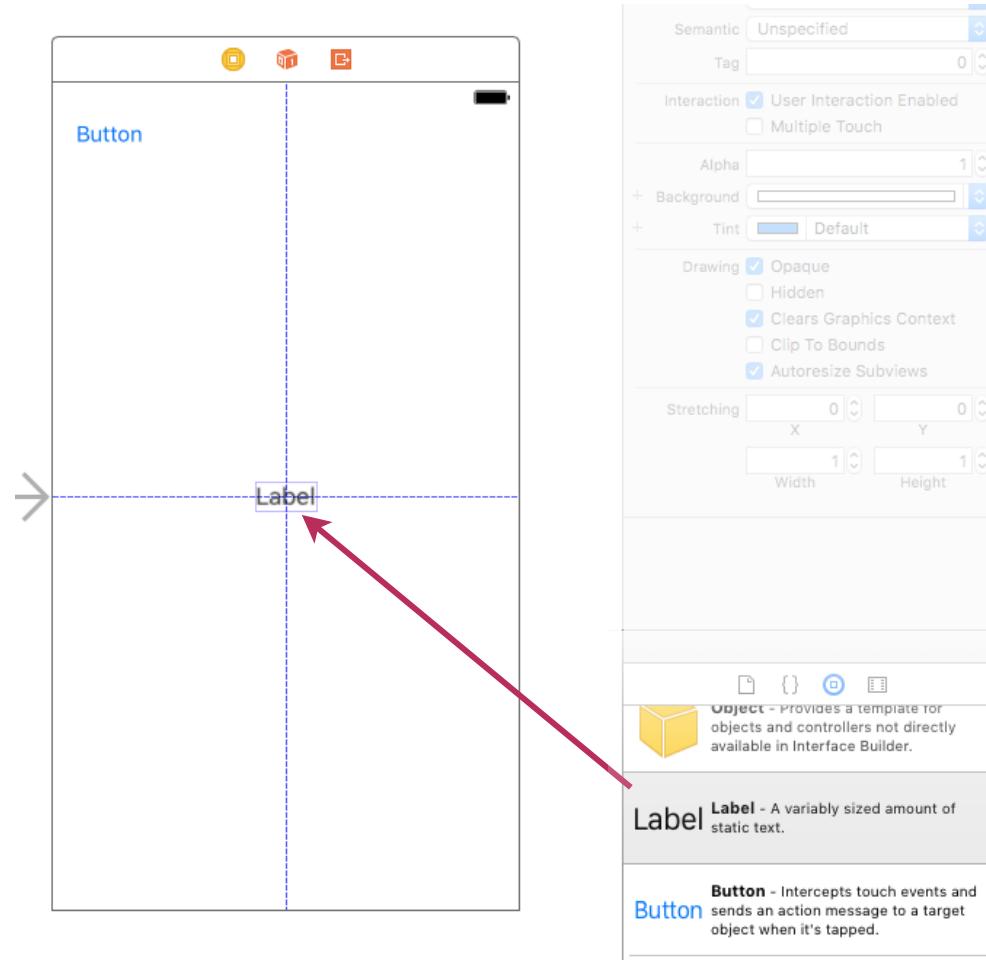


Exercício Simples 2

Alinhamentos horizontais e verticais.

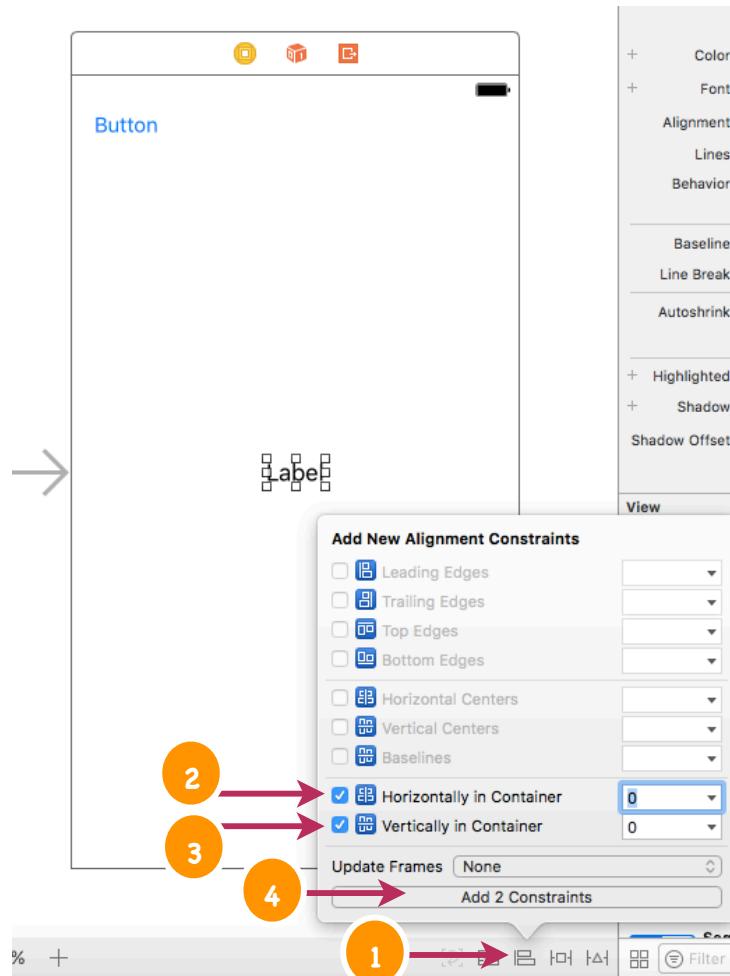
Passo 1: Adicionando uma UILabel a UIViewController

- Arraste uma UILabel para a UIViewController. Depois, posicione o botão na região central da tela, seguindo as orientações das margens de segurança.



Passo 2: Adicionando Constraints de alinhamento

- Clique no botão inferior da tela para abrir o painel de criação de Constraints para alinhamento. Siga a sequencia:



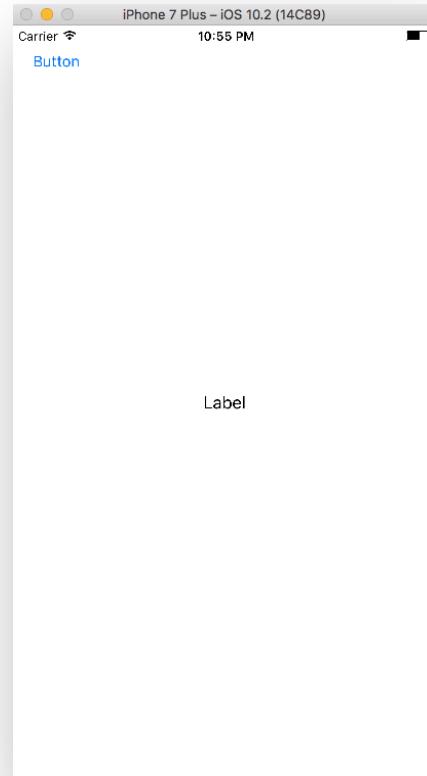
1: Abra o painel para criação de **novas** Constraints de alinhamento.

2/3: Marque as duas opções em destaque.
Deste modo, você irá centralizar a UILabel no meio da tela.

4: Adicione as 2 Constraints ao objeto selecionado.

Passo 3: Conclusão

- Depois de adicionar as 2 Constraints de alinhamento, execute o seu aplicativo em diferentes telas no iOS Simulator para testar as posições.



iPhone 7 Plus



iPhone SE

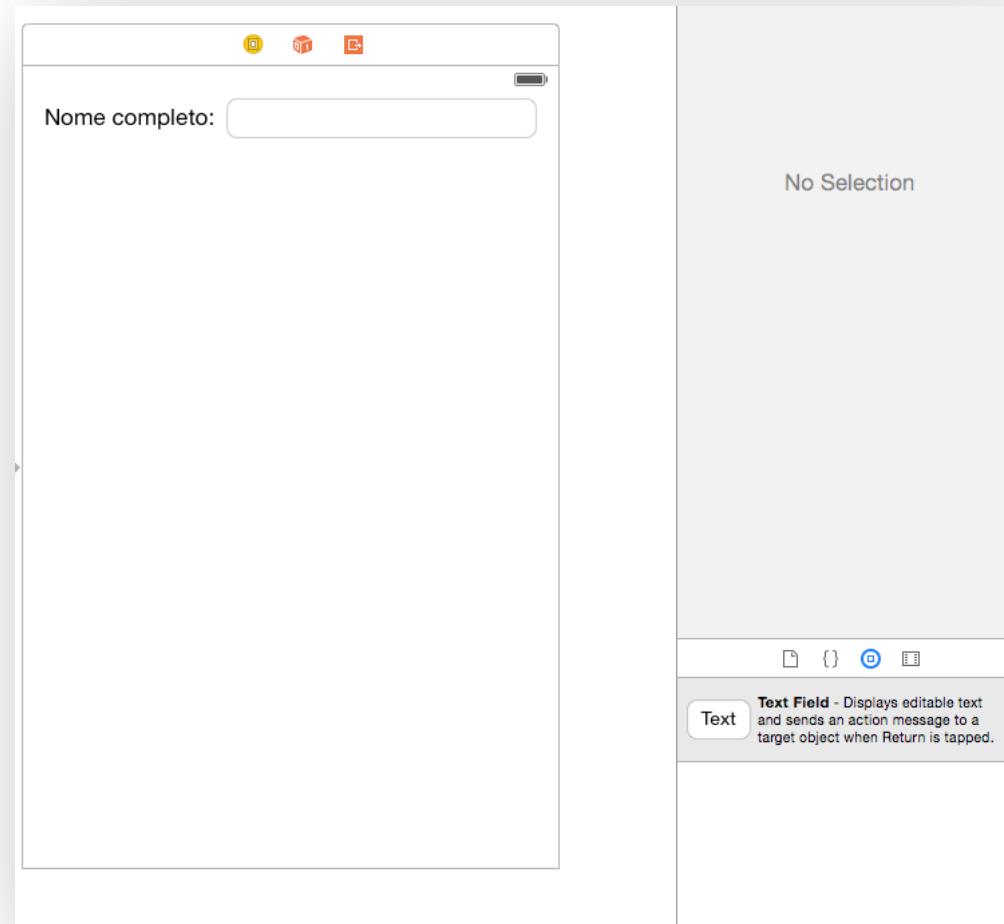


Exercício Simples 3

Alinhamentos com dois componentes.

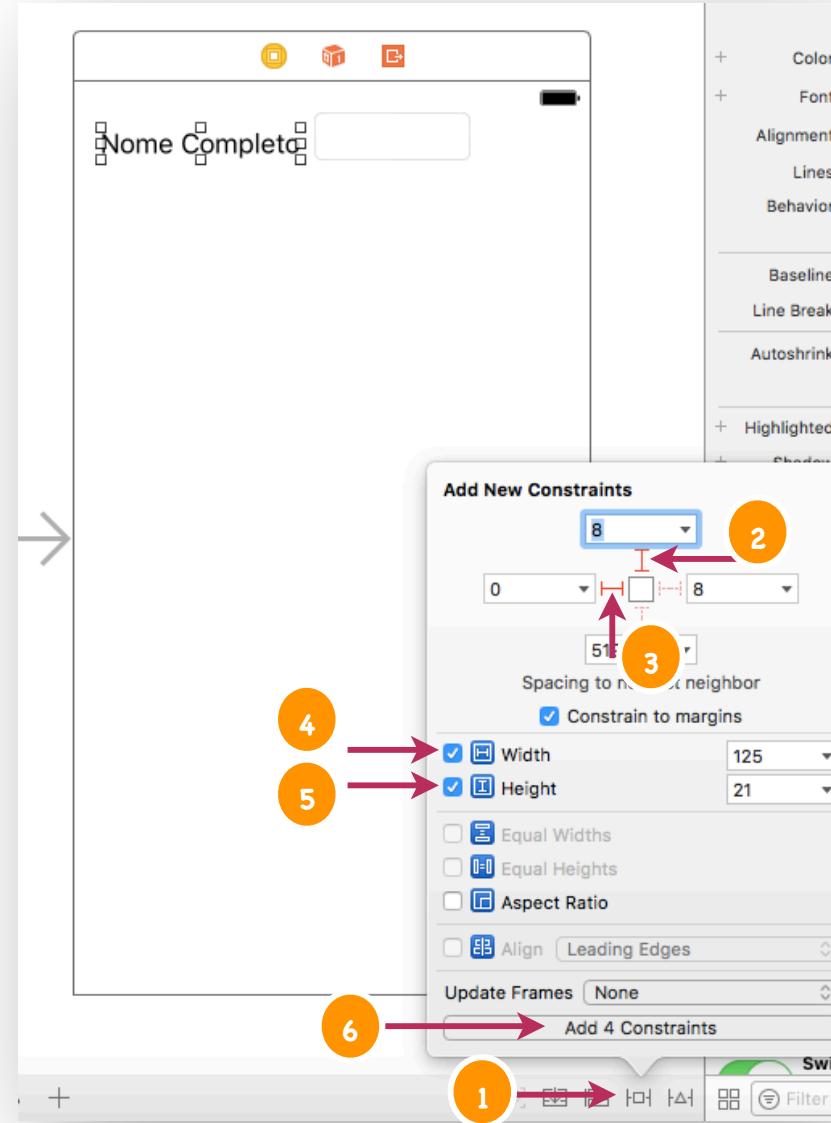
Passo 1: Adicione uma UILabel e um UITextField

- Adicione uma UILabel e troque o texto para “Nome Completo”.
Depois, adicione um UITextField, seguindo as margens de segurança.



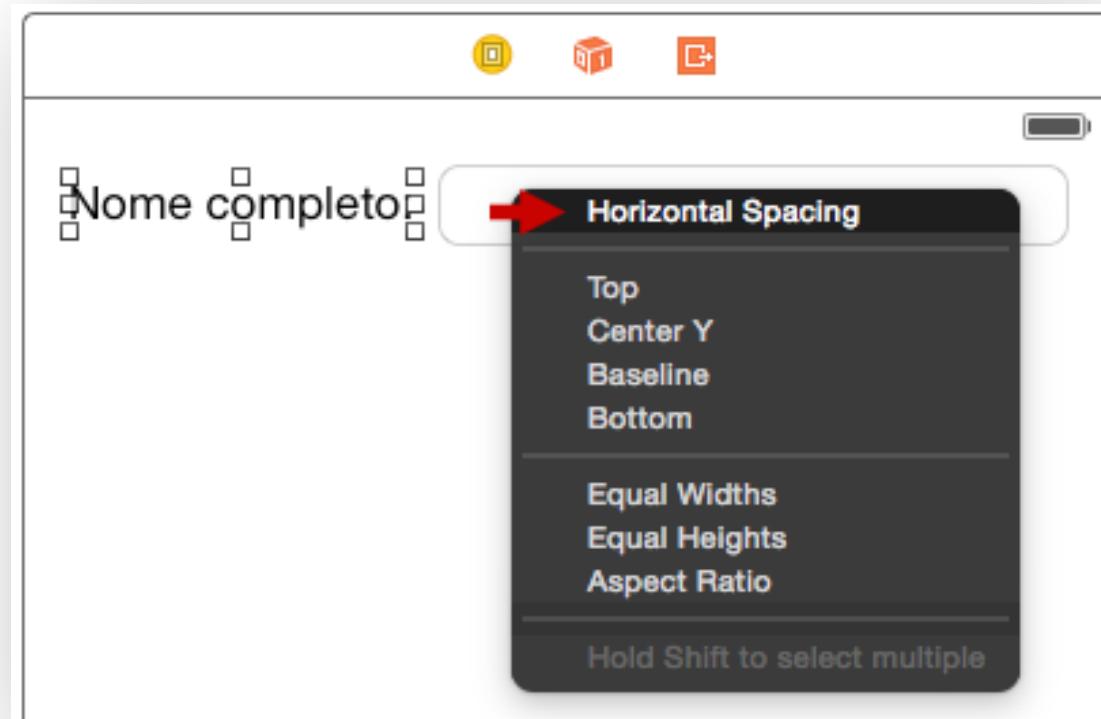
Passo 2: Constraints para UILabel

- Agora, adicione as seguintes Constraints para a UILabel inserida.



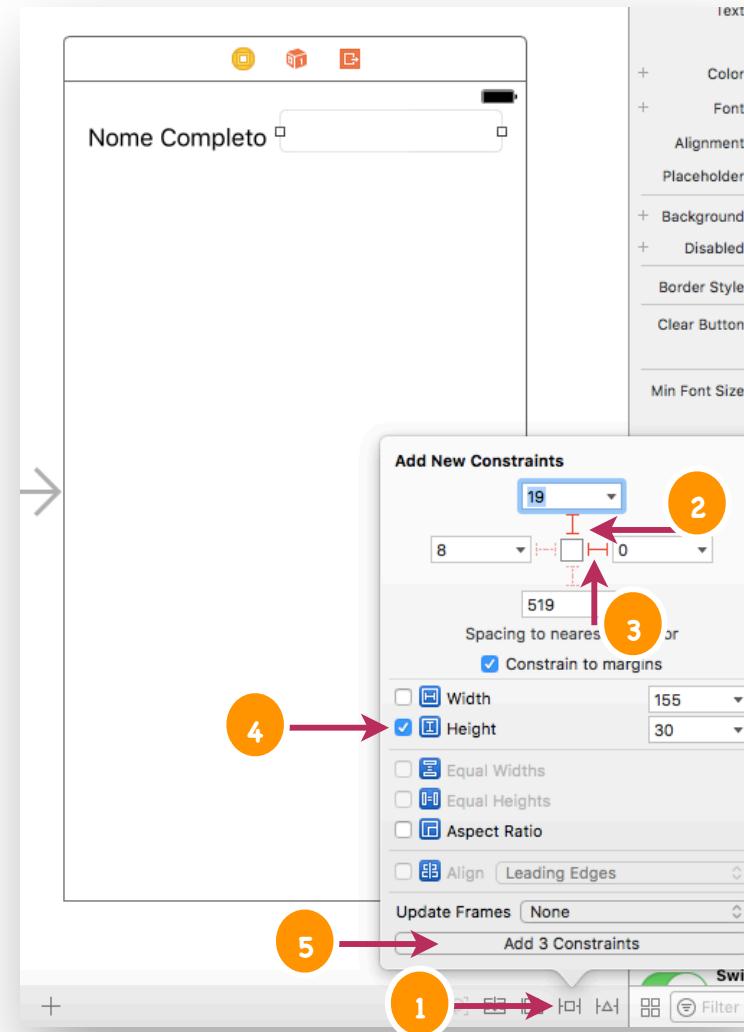
Passo 3: Horizontal Spacing

- Selecione a UILabel, segure a tecla “Control” e arraste até o UITextField. Solte a tecla sobre o UITextField e selecione a opção “Horizontal Spacing”. Ignore os possíveis *Warnings*, por enquanto.



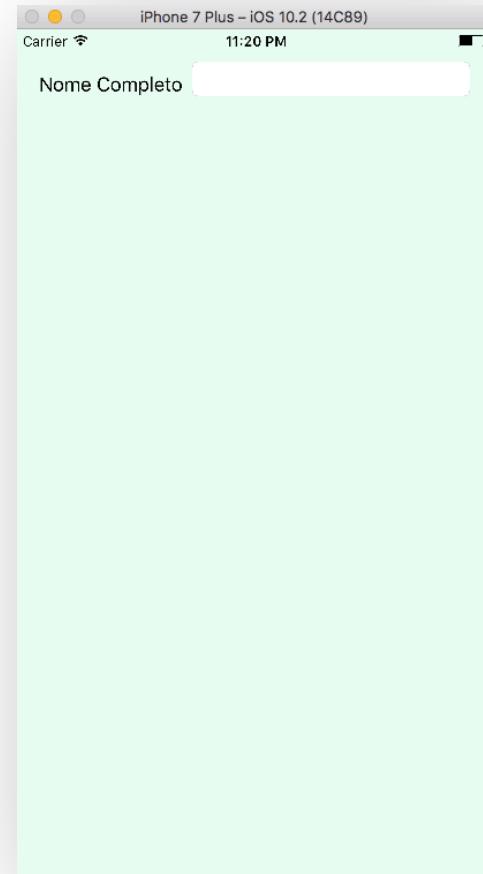
Passo 4: Constraints para UITextField

- Agora, adicione as seguintes Constraints para o UITextField inserido.

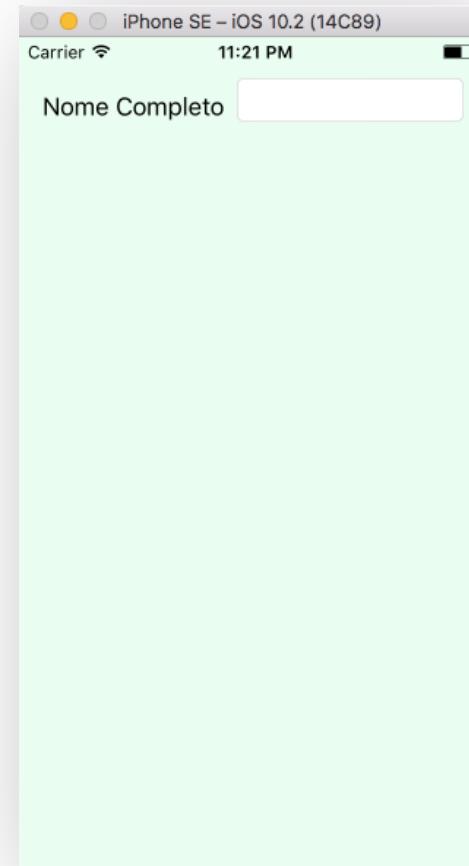


Passo 5: Conclusão

- Depois de concluir os passos anteriores, execute sua aplicação em diferentes dispositivos para visualizar o resultado.



iPhone 7 Plus



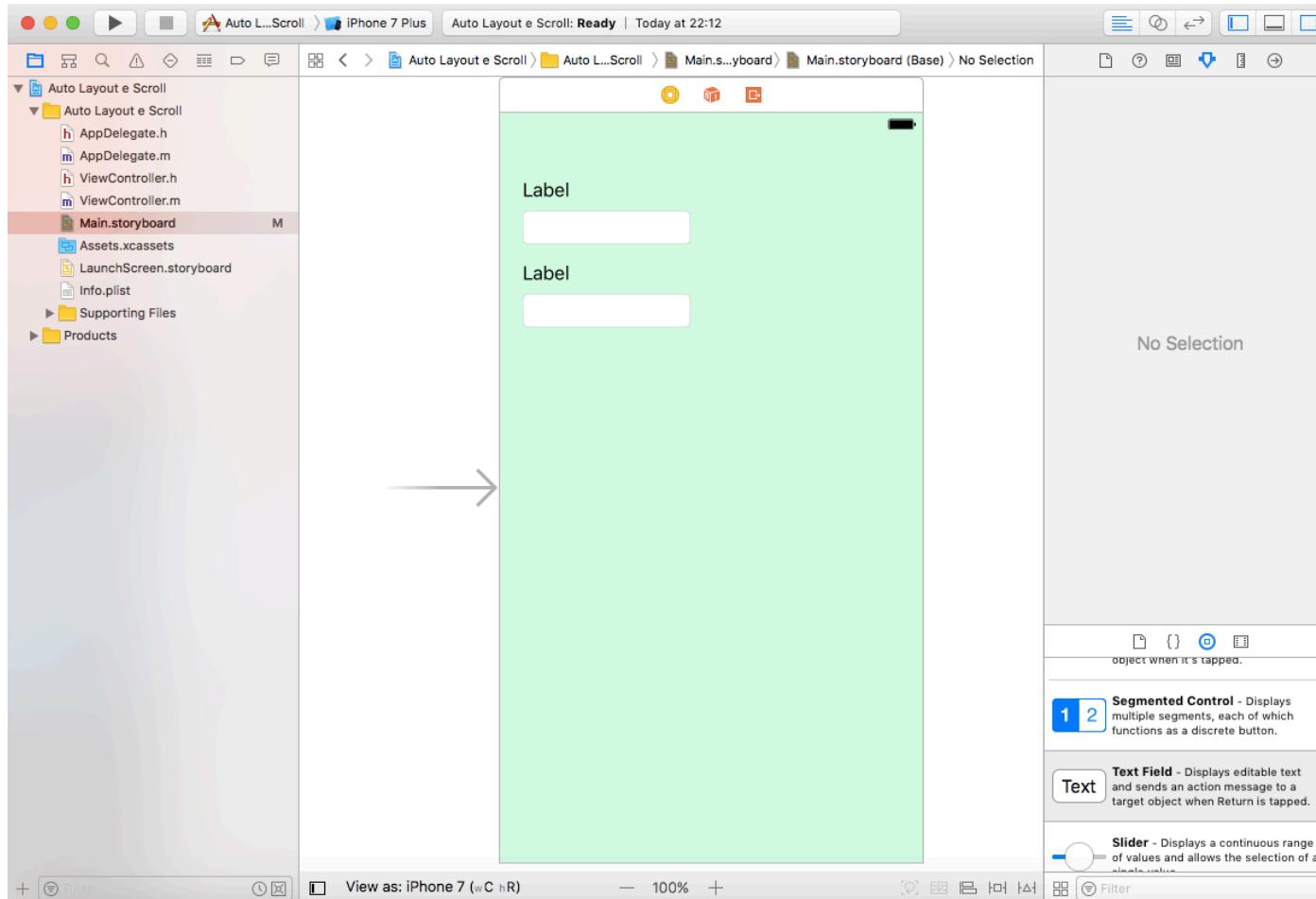
iPhone SE

Auto Layout

Criando um formulário de contato.

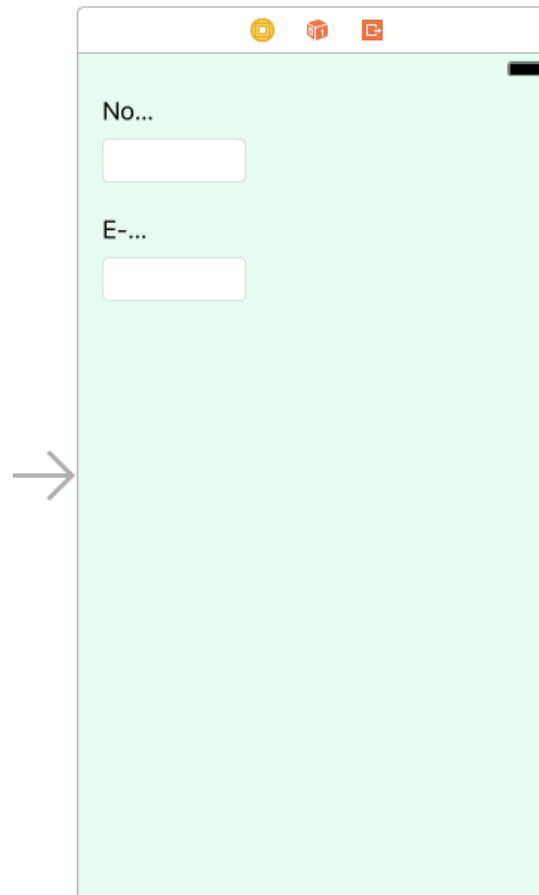
Passo 1: Adicionar elementos

- Adicione 2 UILabels e 2 UITextField respeitando as margens de segurança.



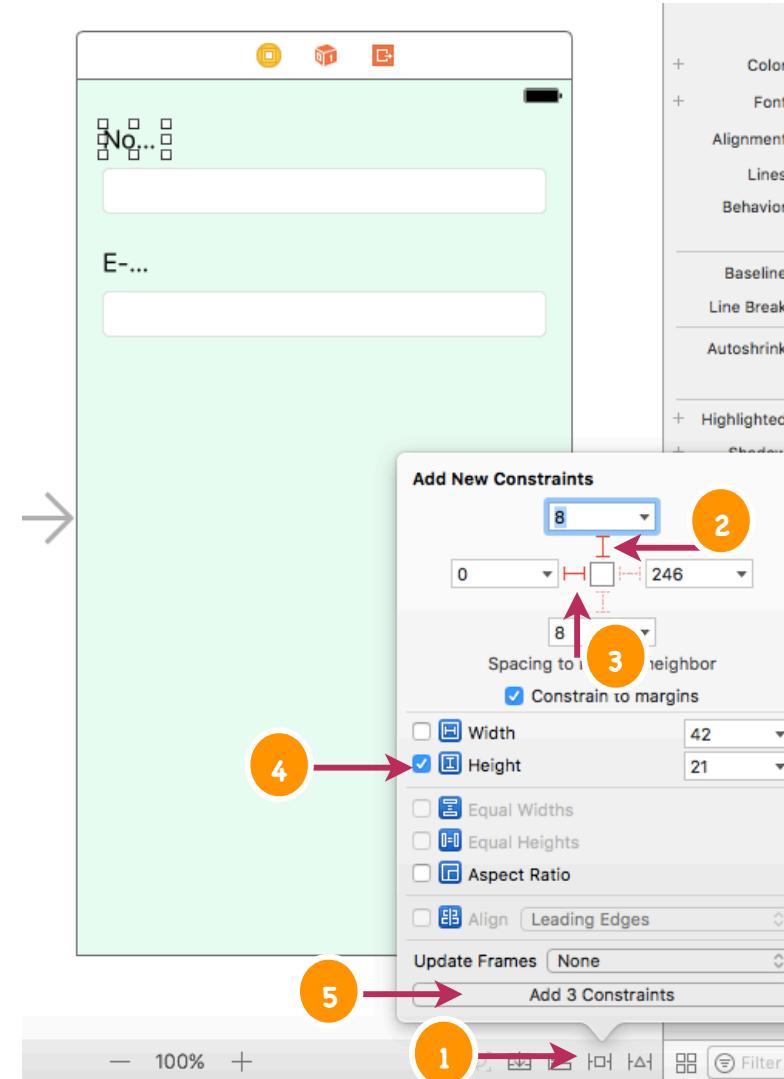
| Passo 2: Alterando o texto da UILabel

- Troque o texto da **1^a UILabel** para “Nome completo” e o texto da **2^a UILabel** para ”E-mail para contato”.



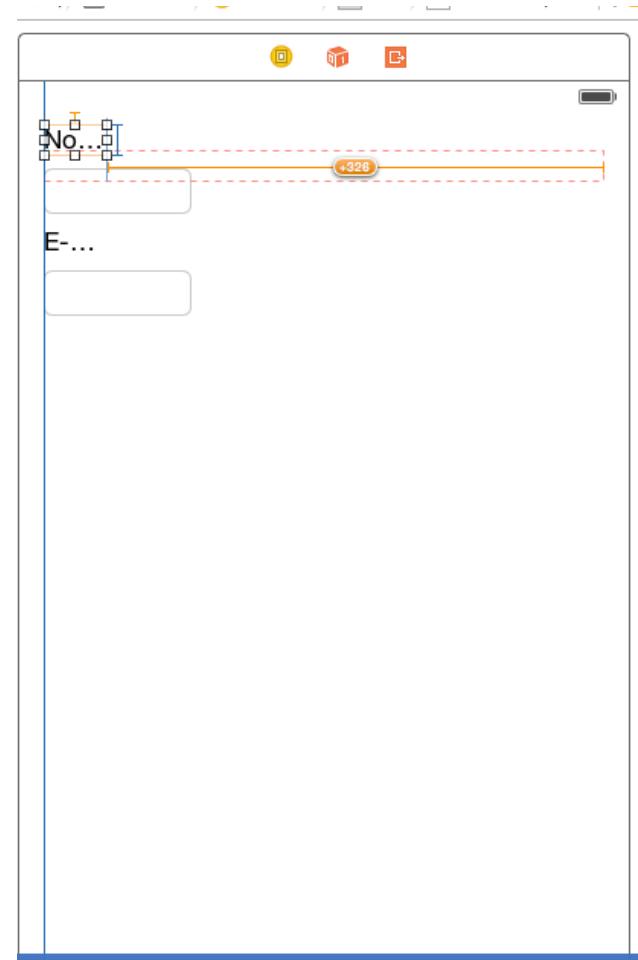
Passo 3: Constraints para 1ª UILabel

- Adicione as seguintes Constraints para a 1ª UILabel.



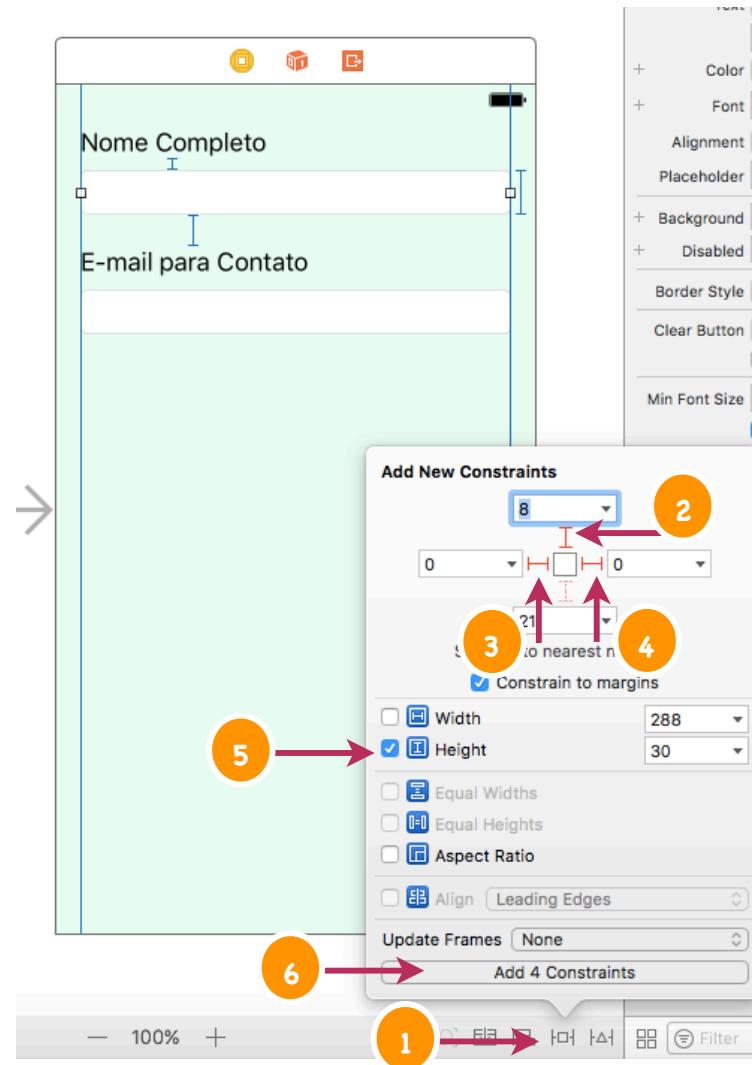
Passo 3: Warnings são comuns

- Se deparar com warnings como o que está sendo exibido abaixo é comum. Ao final do exercício, todos sumirão.



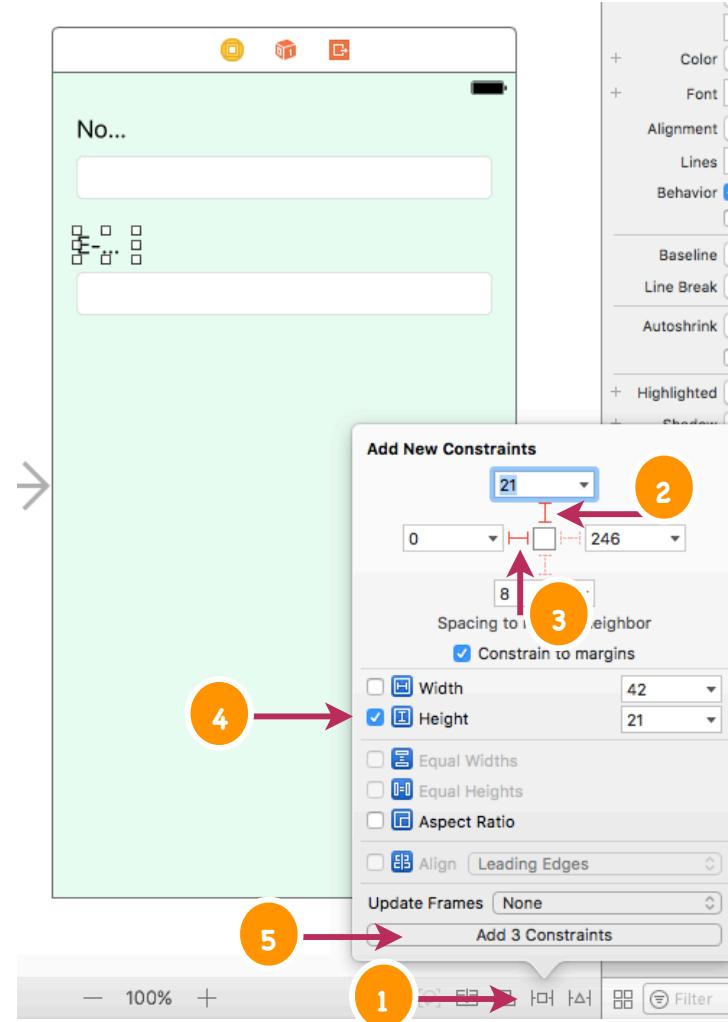
Passo 4: Constraints para o 1º UITextField

- Adicione as seguintes Constraints para o 1º UITextField.



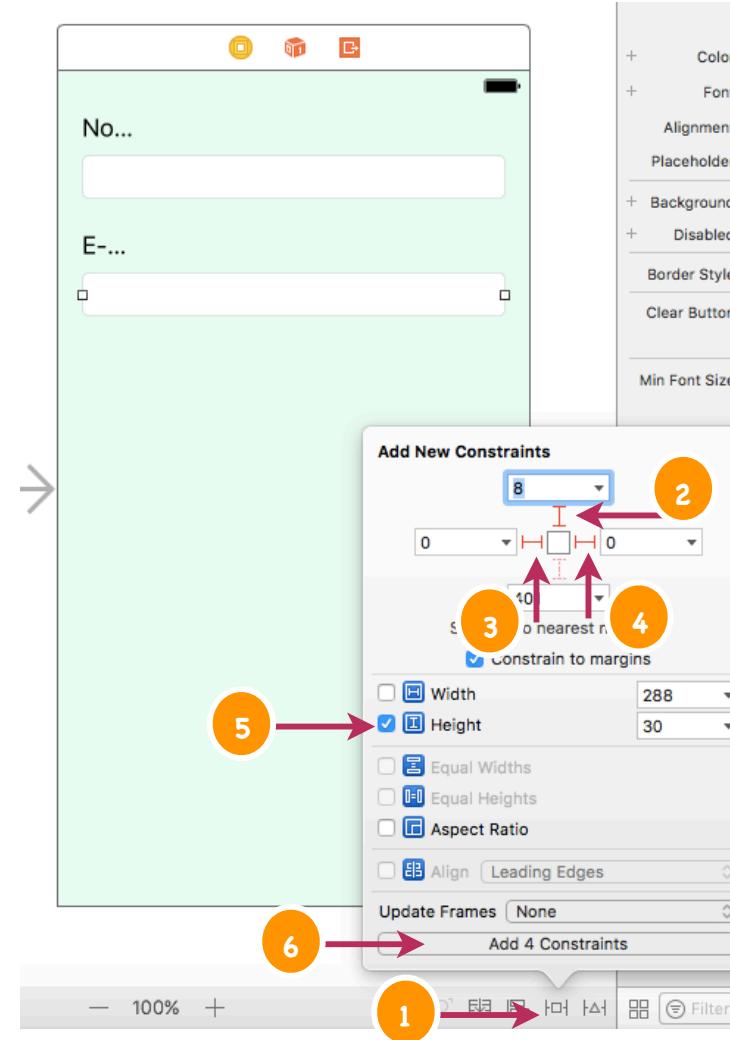
Passo 5: Constraints para 2ª UILabel

- Adicione as seguintes Constraints para a 2ª UILabel.



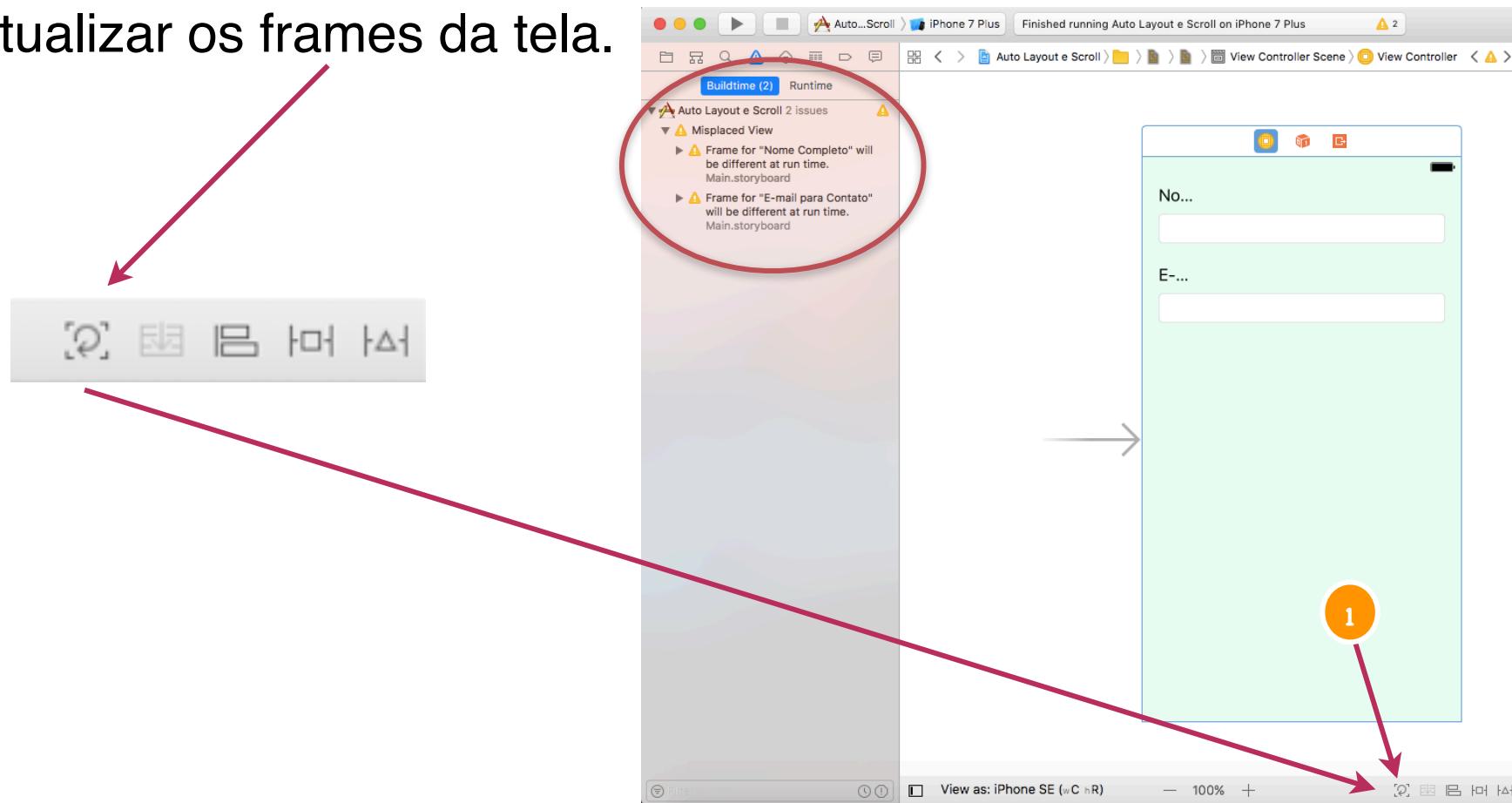
Passo 6: Constraints para o 2º UITextField

- Adicione as seguintes Constraints para o 2º UITextField.

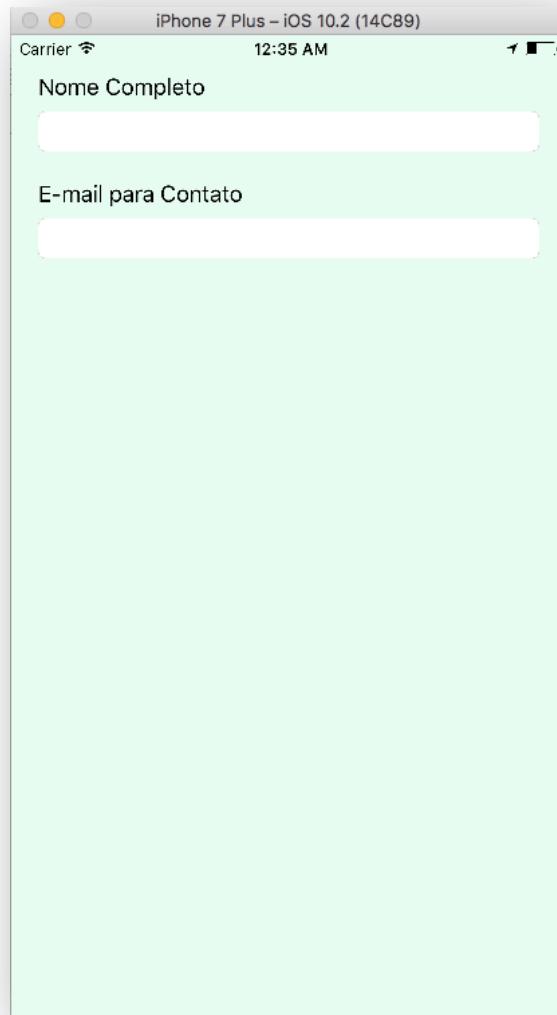


Passo 7: Removendo Warnings e atualizando a tela

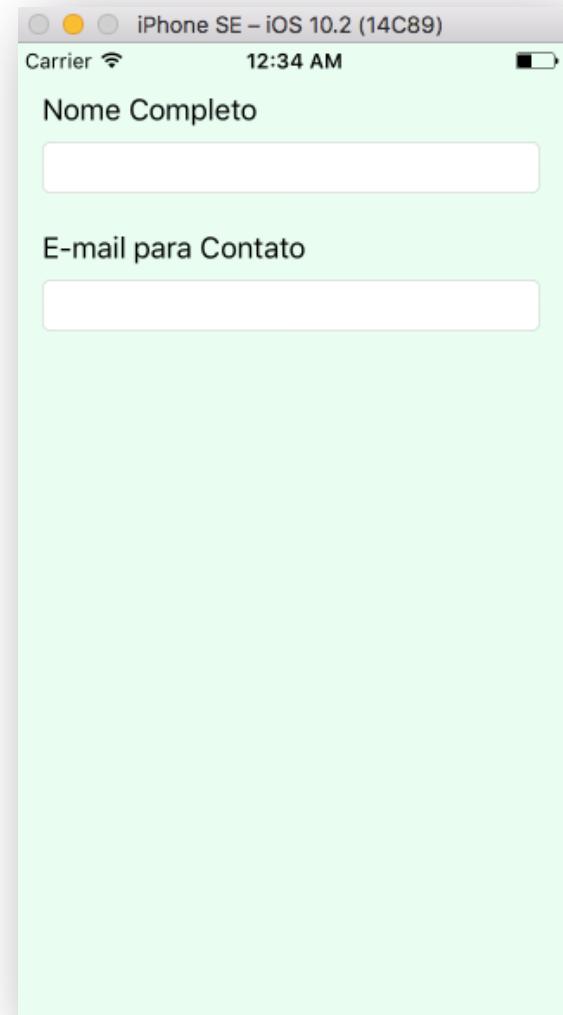
- Para visualizar o resultado das Constraints adicionadas e remover os warnings (que são avisos informando que o usuário não está visualizando a tela de maneira correta), clique na opção de atualizar os frames da tela.



Passo 8: Visualizando as alterações



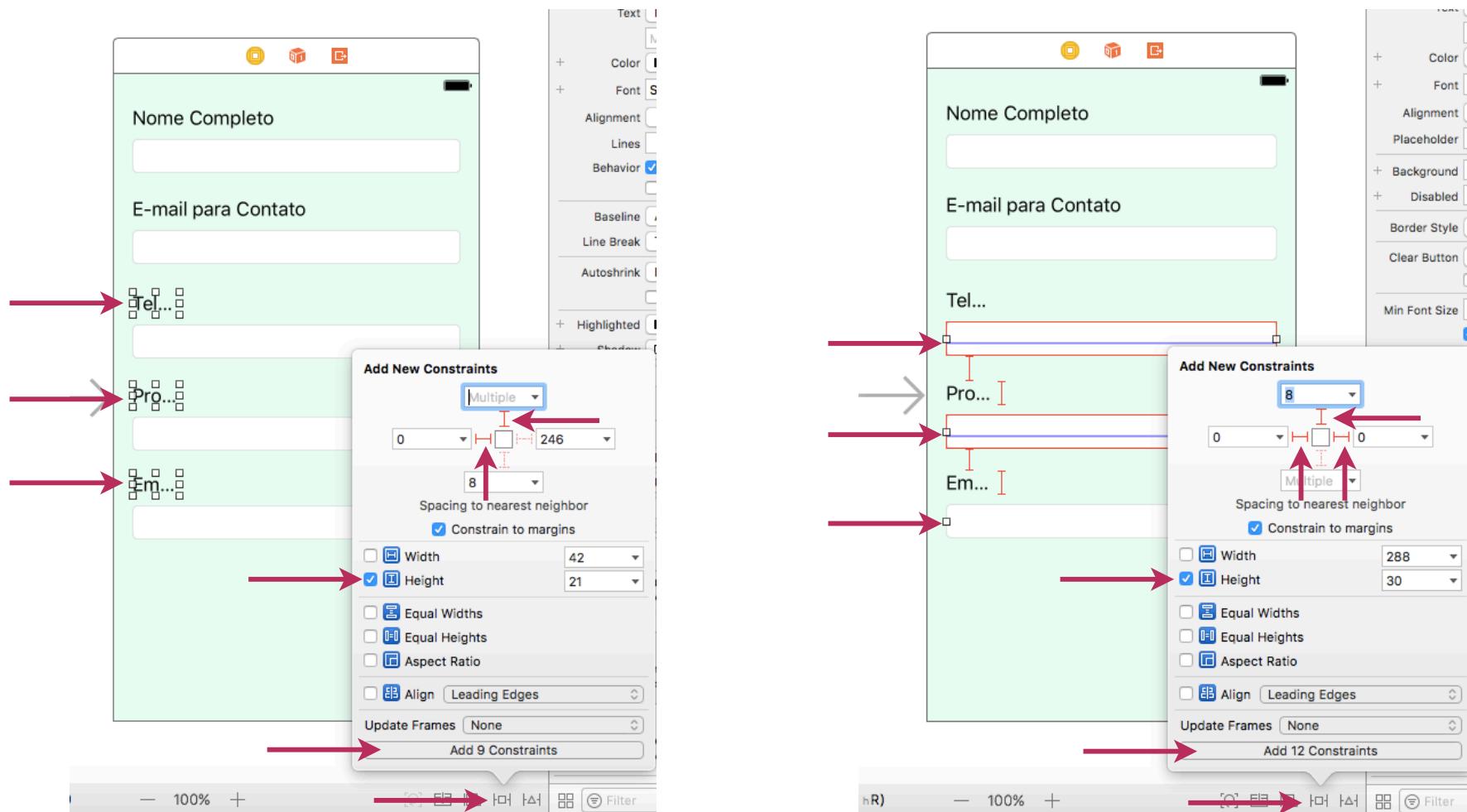
iPhone 7 Plus



iPhone SE

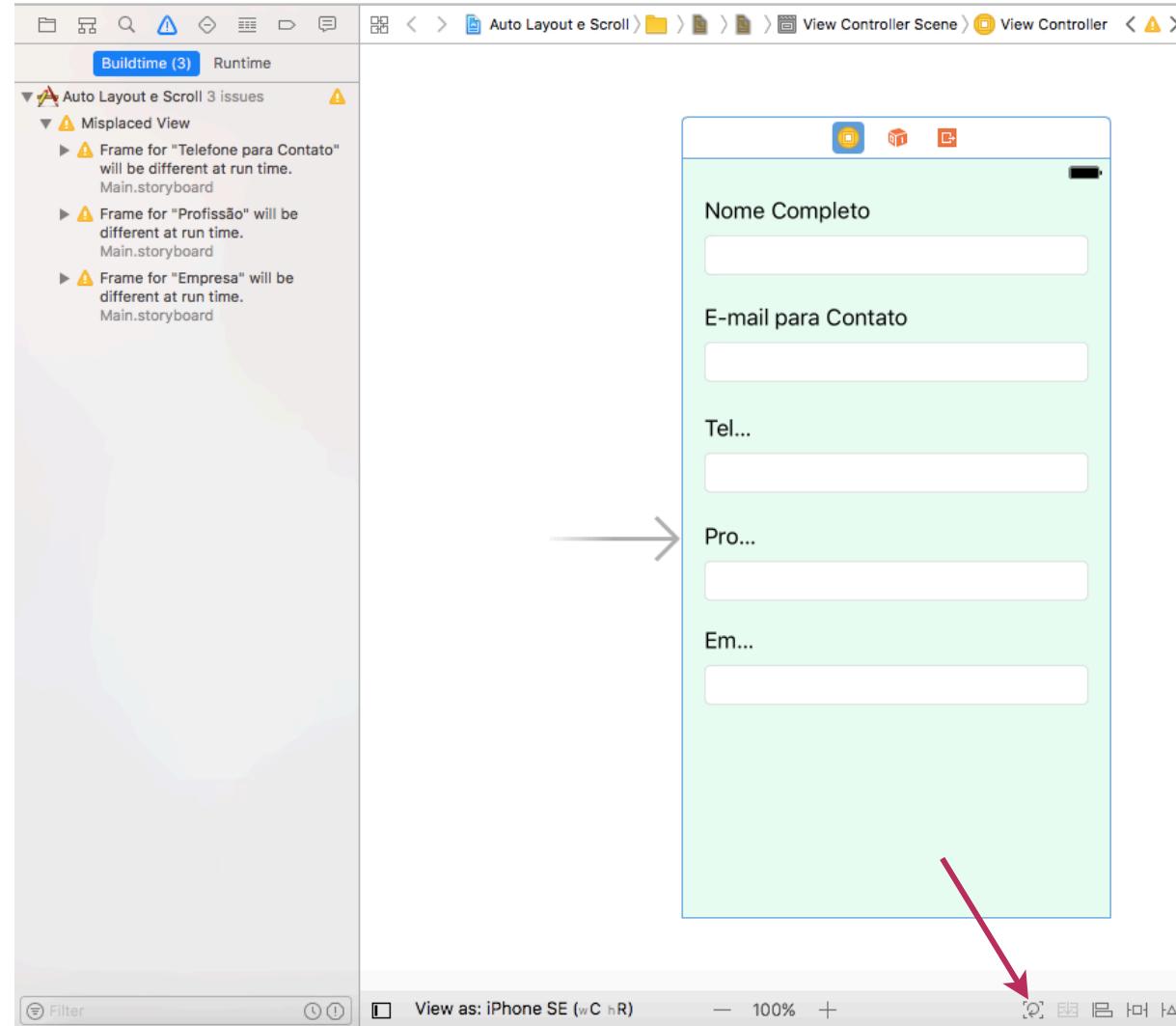
Passo 9: Adicionando mais elementos

- Para adicionar mais elementos, basta usar as mesmas Constraints utilizadas na 2^a UILabel e no 2º UITextField.

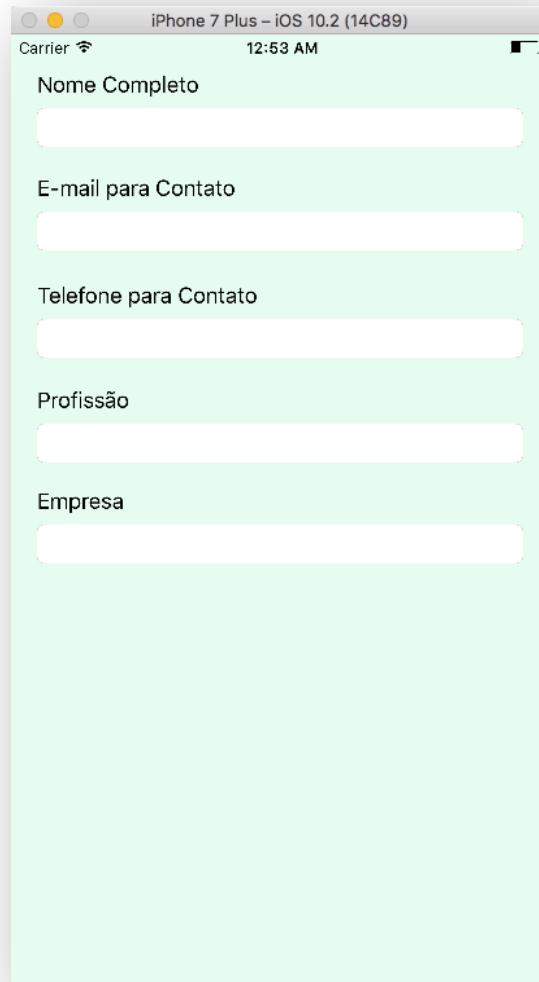


Passo 10: Removendo os Warnings

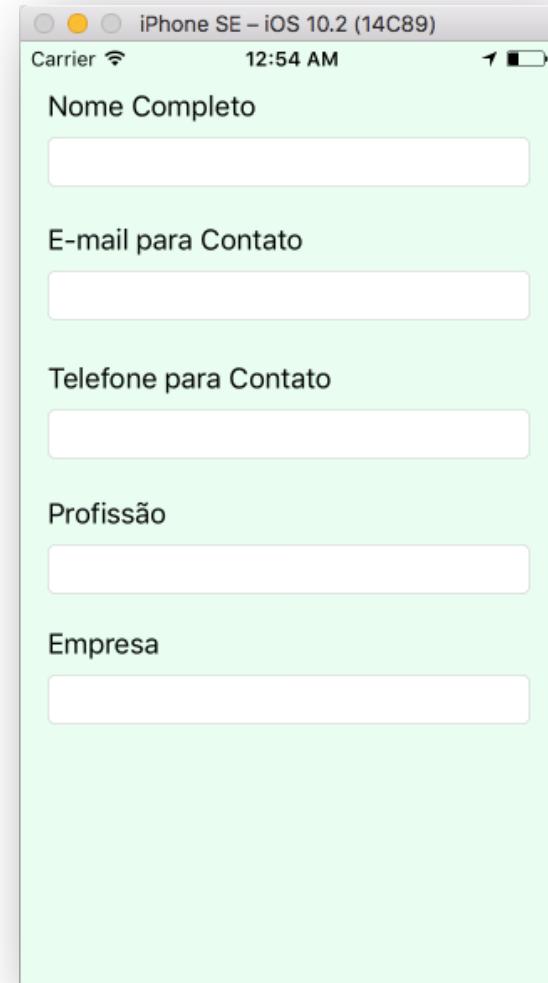
- Atualize os Frames para que saia os warnings.



Passo 11: Visualizando as alterações



iPhone 7 Plus

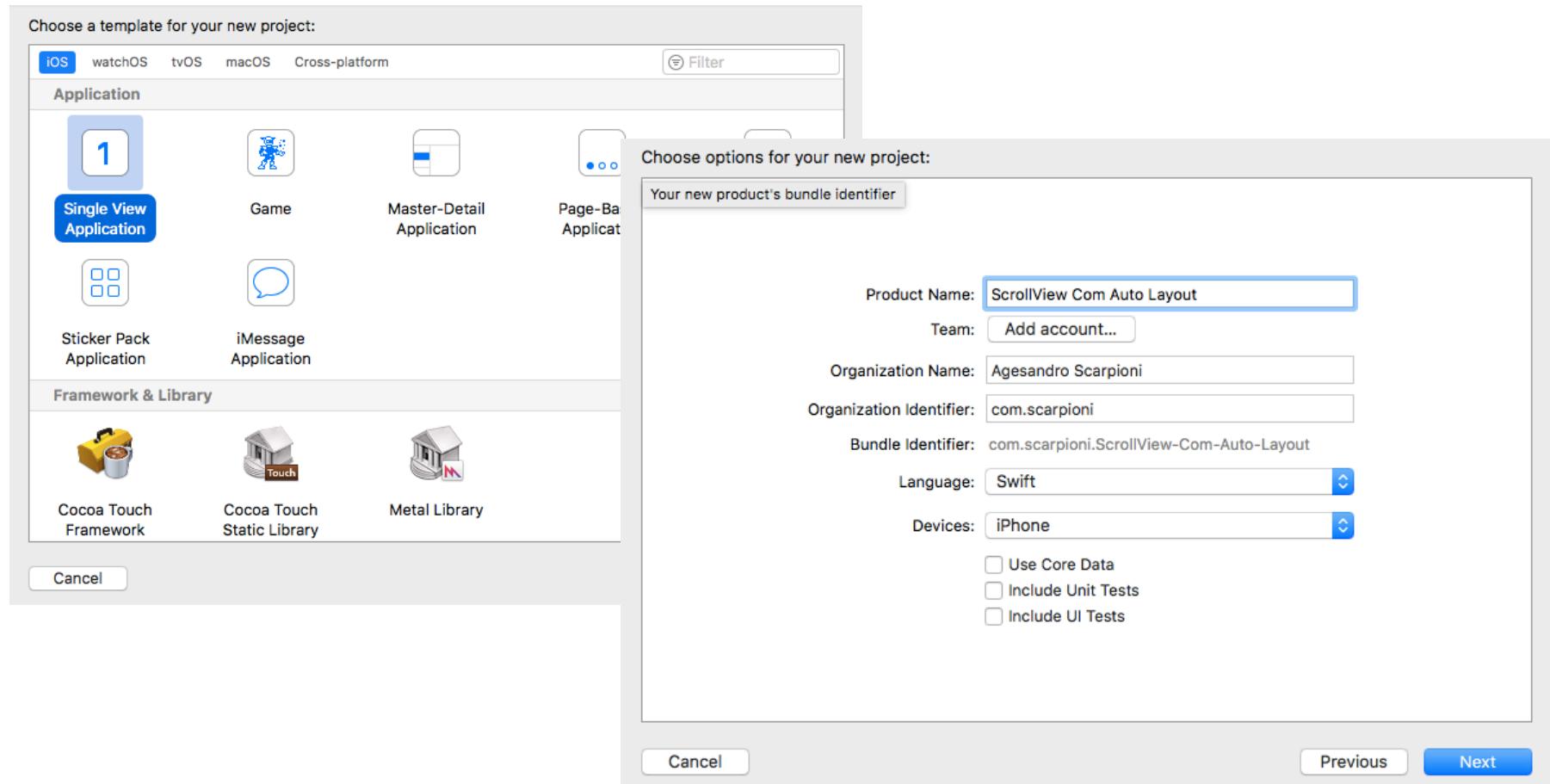


iPhone SE

Auto Layout + UIScrollView

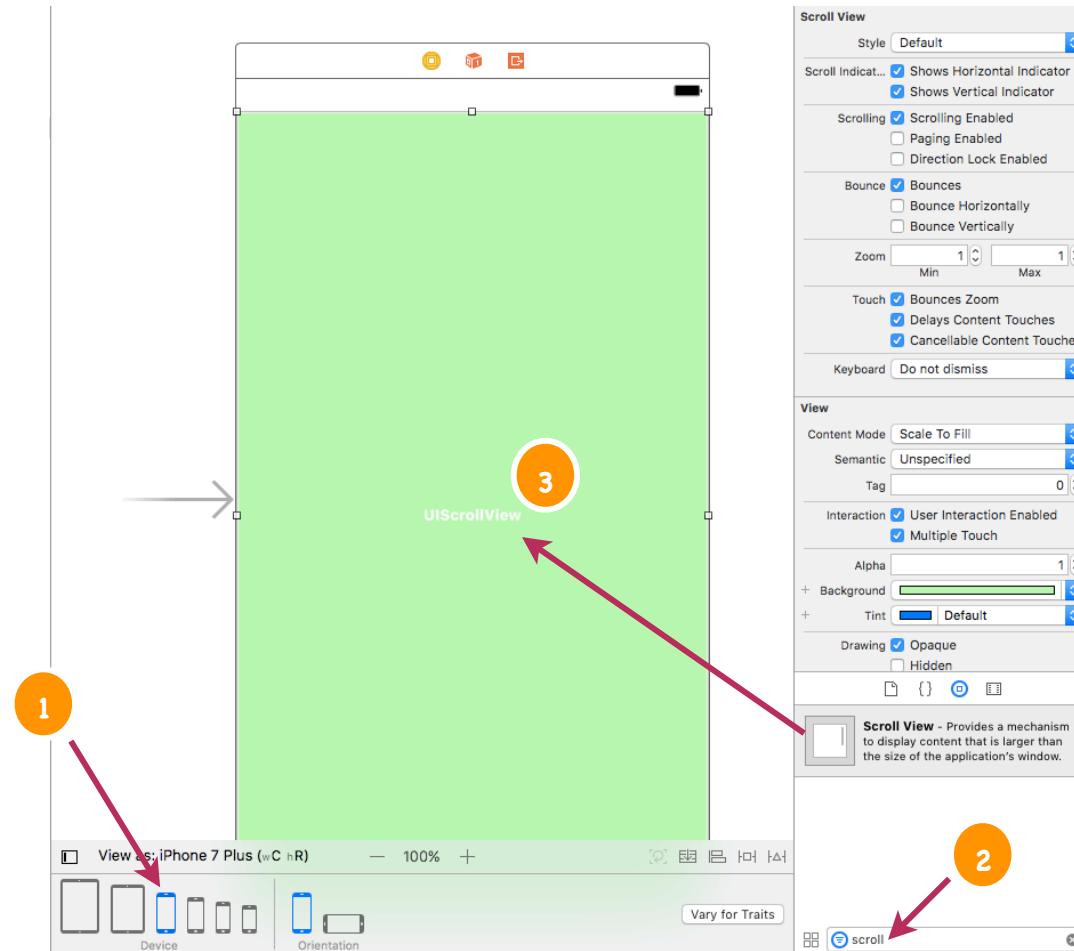
Crie um projeto Novo

- iOS - Single View Application, nomeio como ScrollView Com Auto Layout



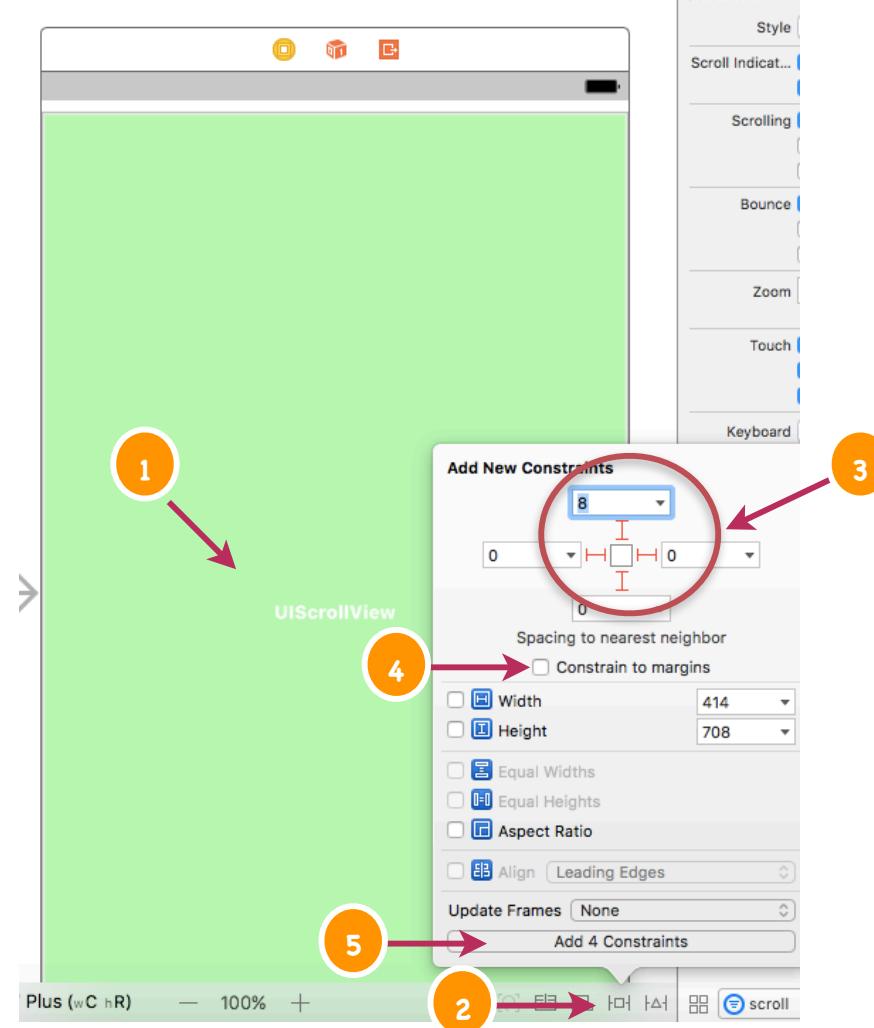
Passo 1: Alterando o tamanho da UIViewController

- Para realizar este exercício, altere o tamanho de sua UIViewController para iPhone 7 Plus e insira um Scroll View e altere o background.



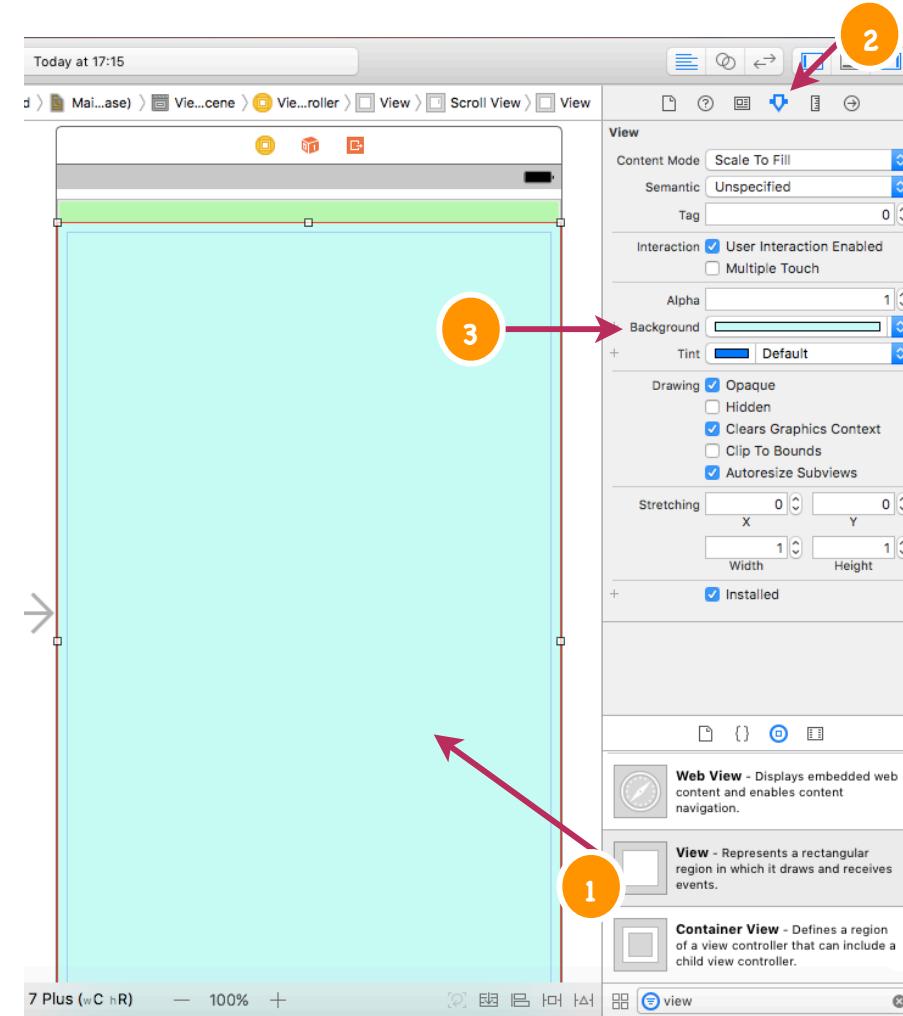
Passo 2: UIScrollView e Constraints

- Adicione as seguintes Constraints:



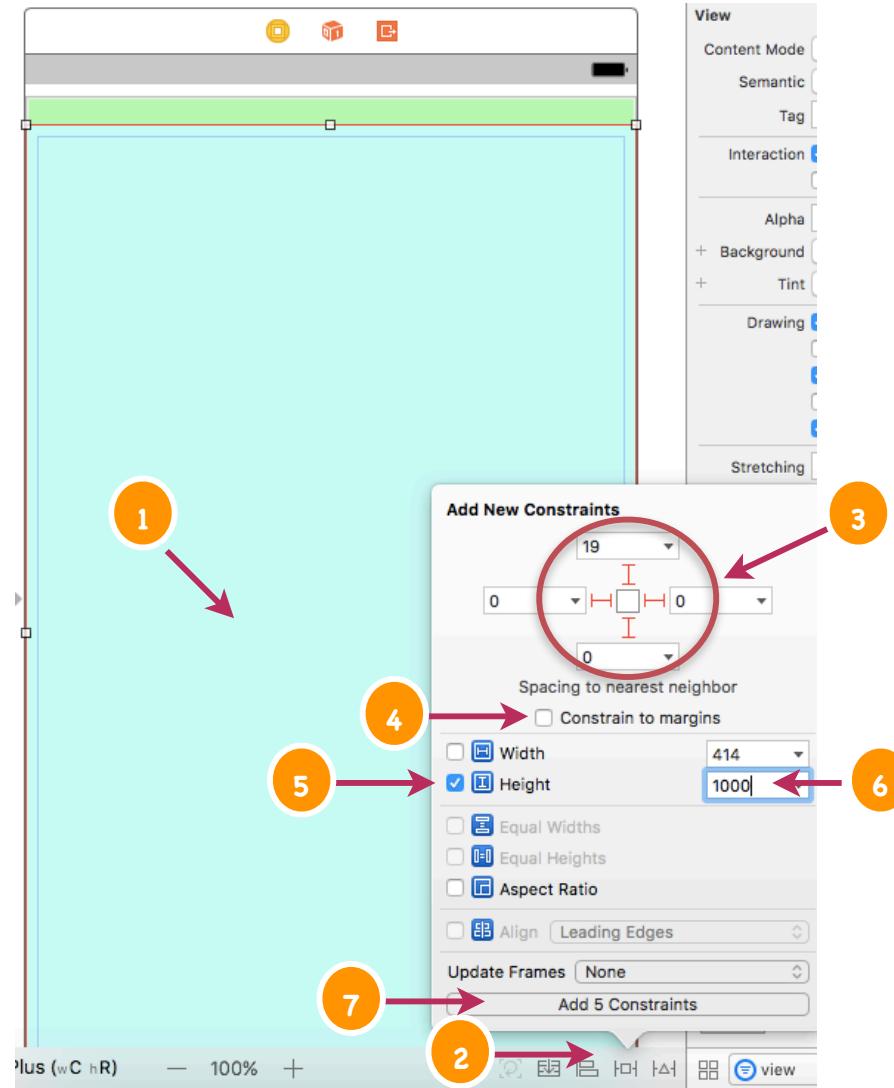
Passo 3: View e BackGround

- Insira uma View um pouco abaixo do Scroll e altere o background para outra cor, dessa forma você verá qual é o Scroll e qual é a View.



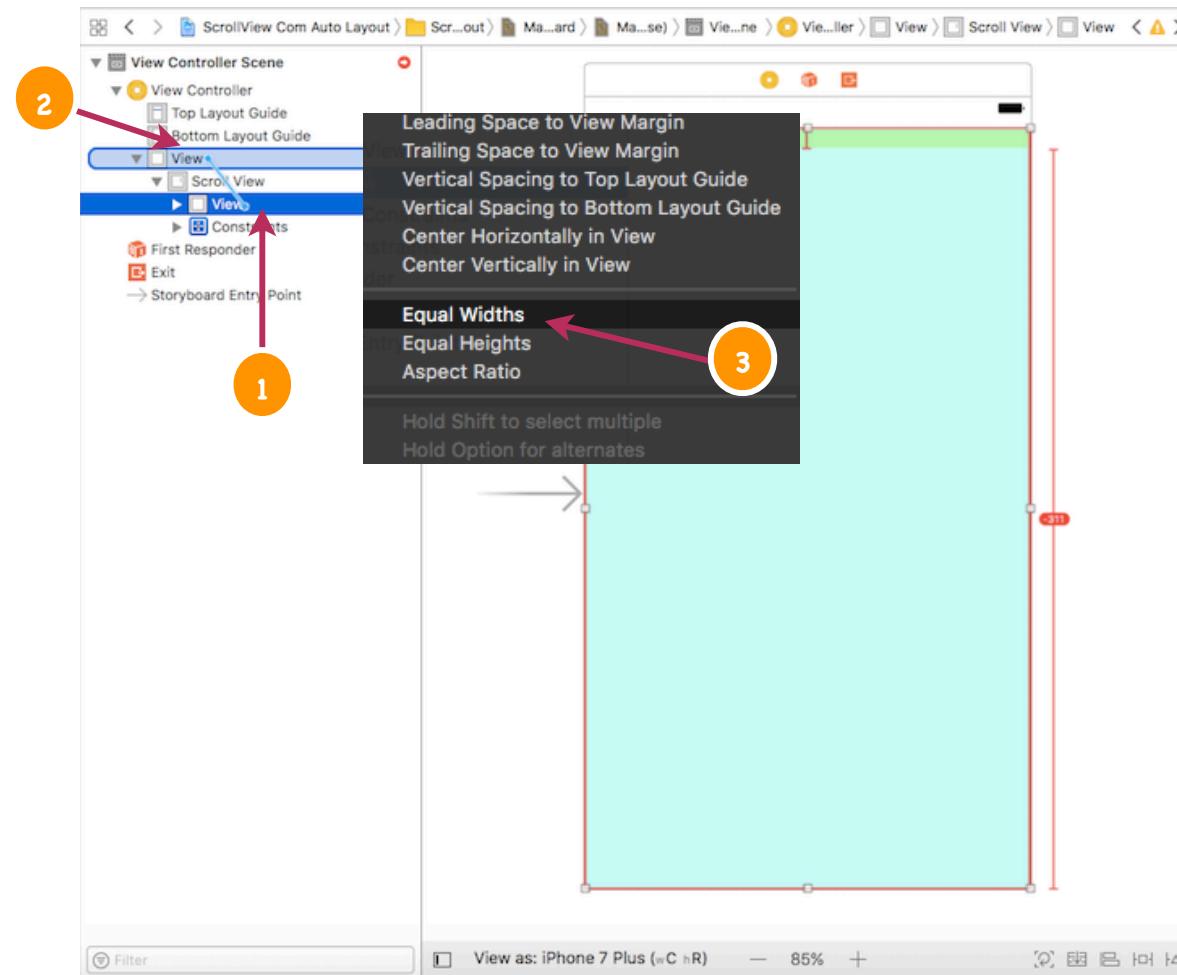
Passo 4: View e Constraints

- Adicione as seguintes Constraints:



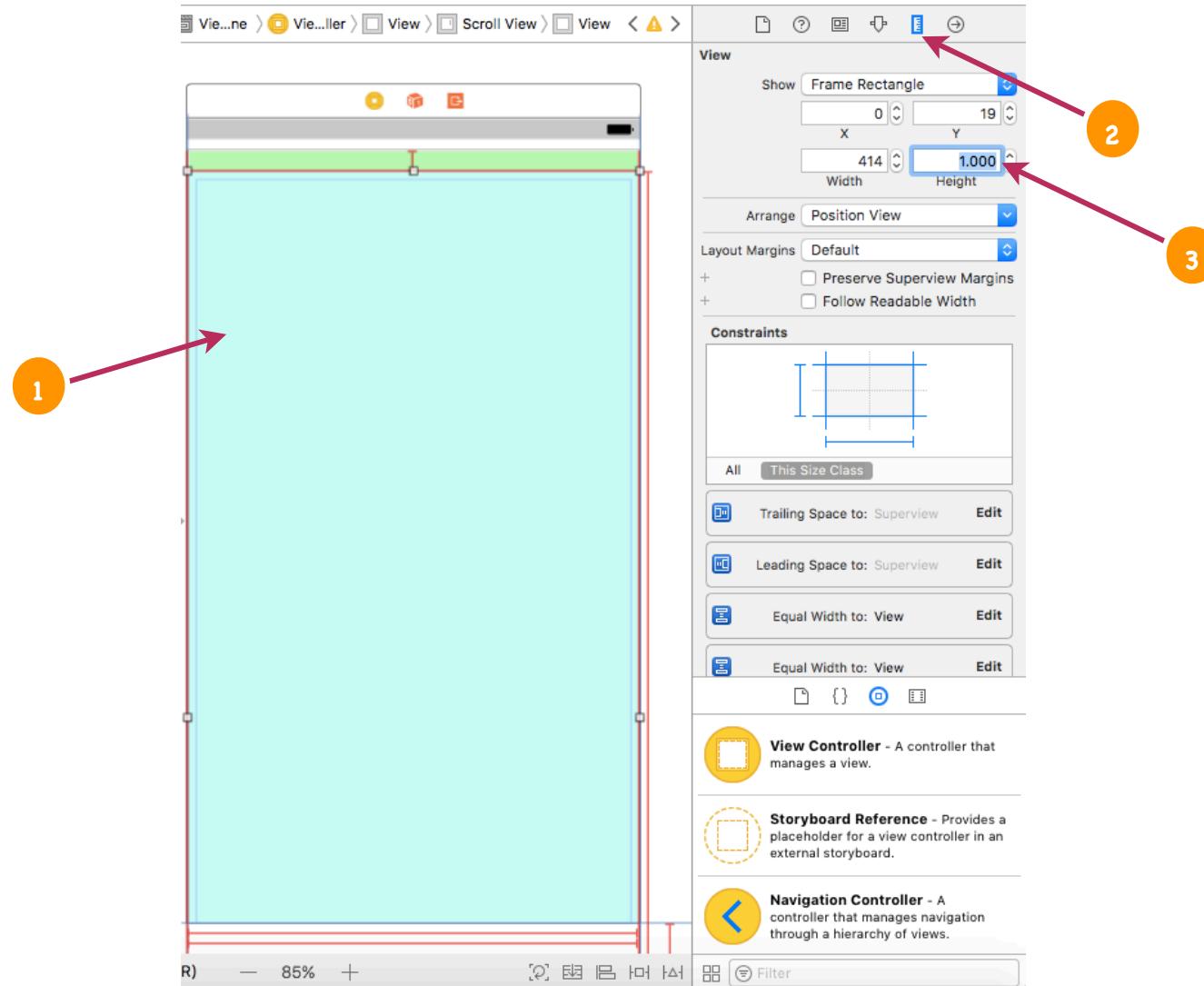
Passo 5: UITextField e Constraints

- Clique na View (1) e com o Control apertado arraste o mouse até a View principal (2), escolha Equal Widths (3).



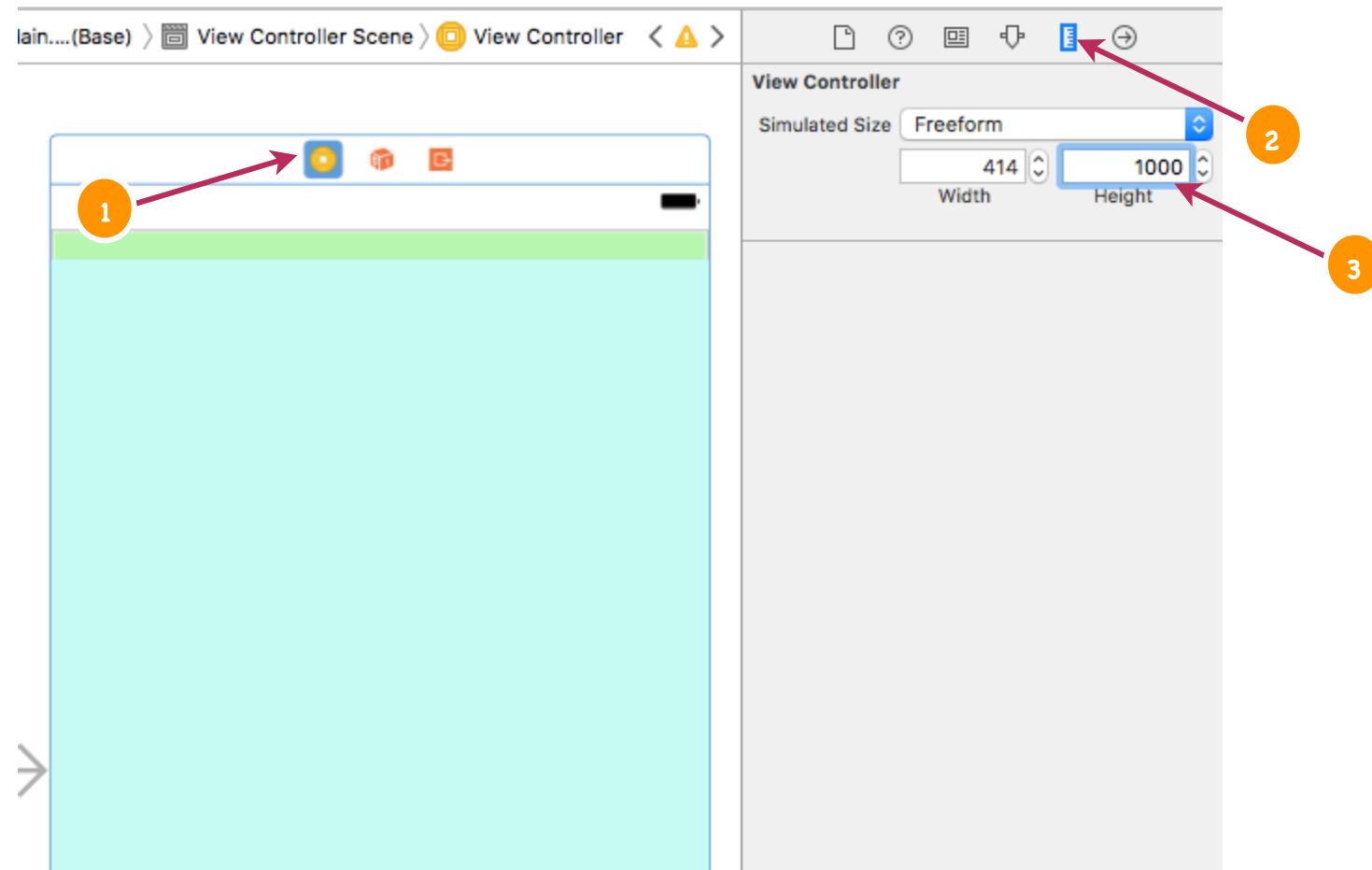
Passo 6: Size Inspector

- Com a View selecionada, clique em Size Inspector e altere o Height para 1000



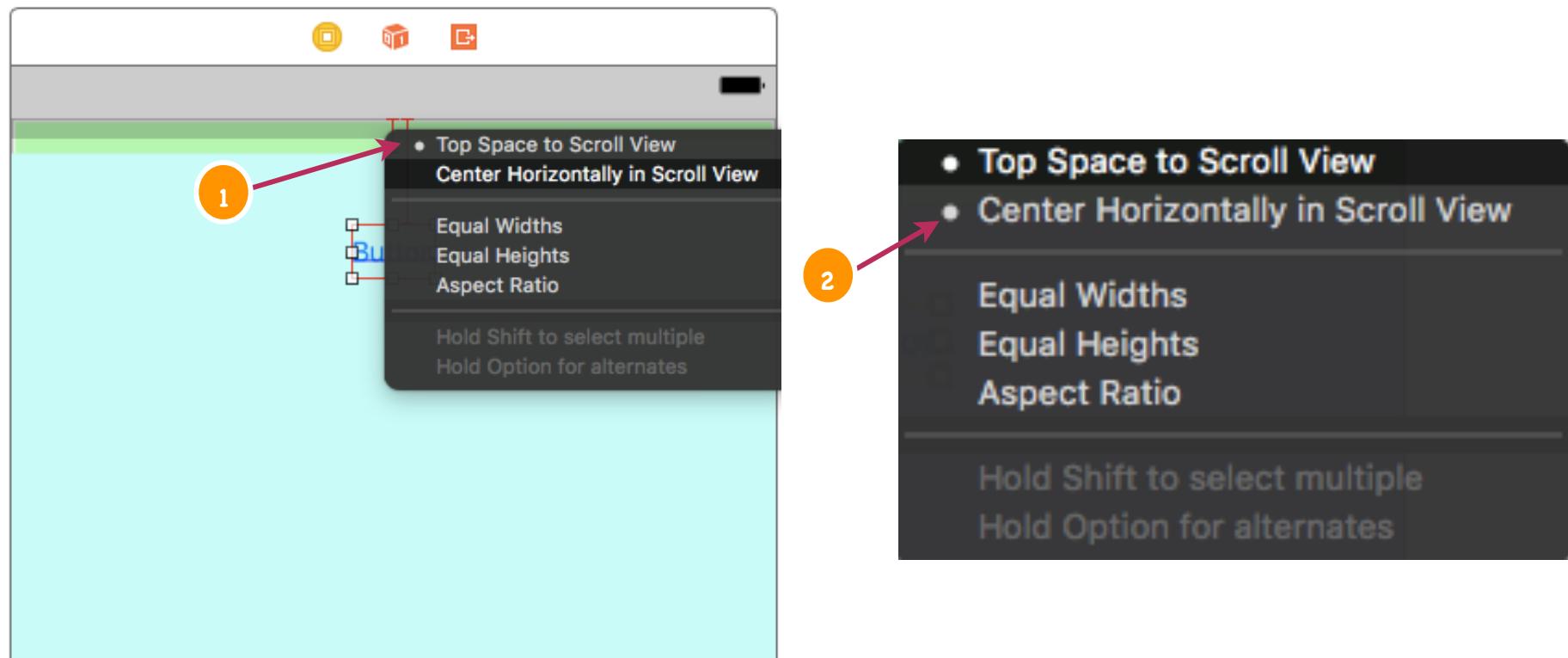
Passo 7: Altere a medida da ViewController

- Selecione a ViewController(1) e em Size Inspector (2) também altere para 1000 o Height (3).



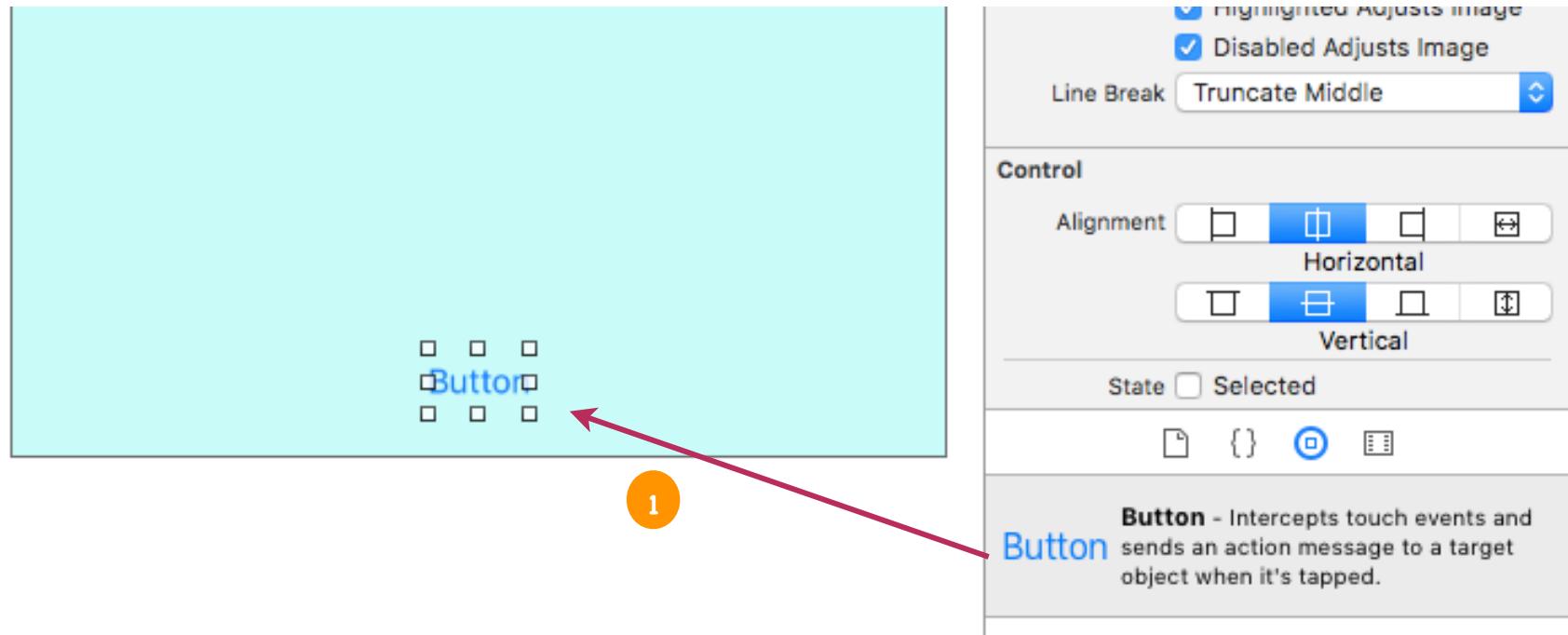
Passo 8: Insira um botão no Topo da View

- Selecione o botão e com a tecla Ctrl selecionada, arraste o mouse até a área verde que representa o ScrollView, ao soltar o botão do mouse selecione a primeira opção “Top Space to Scroll View” (1). Repita o passo para marcar a segunda opção “Center Horizontally in Scroll View” (2).



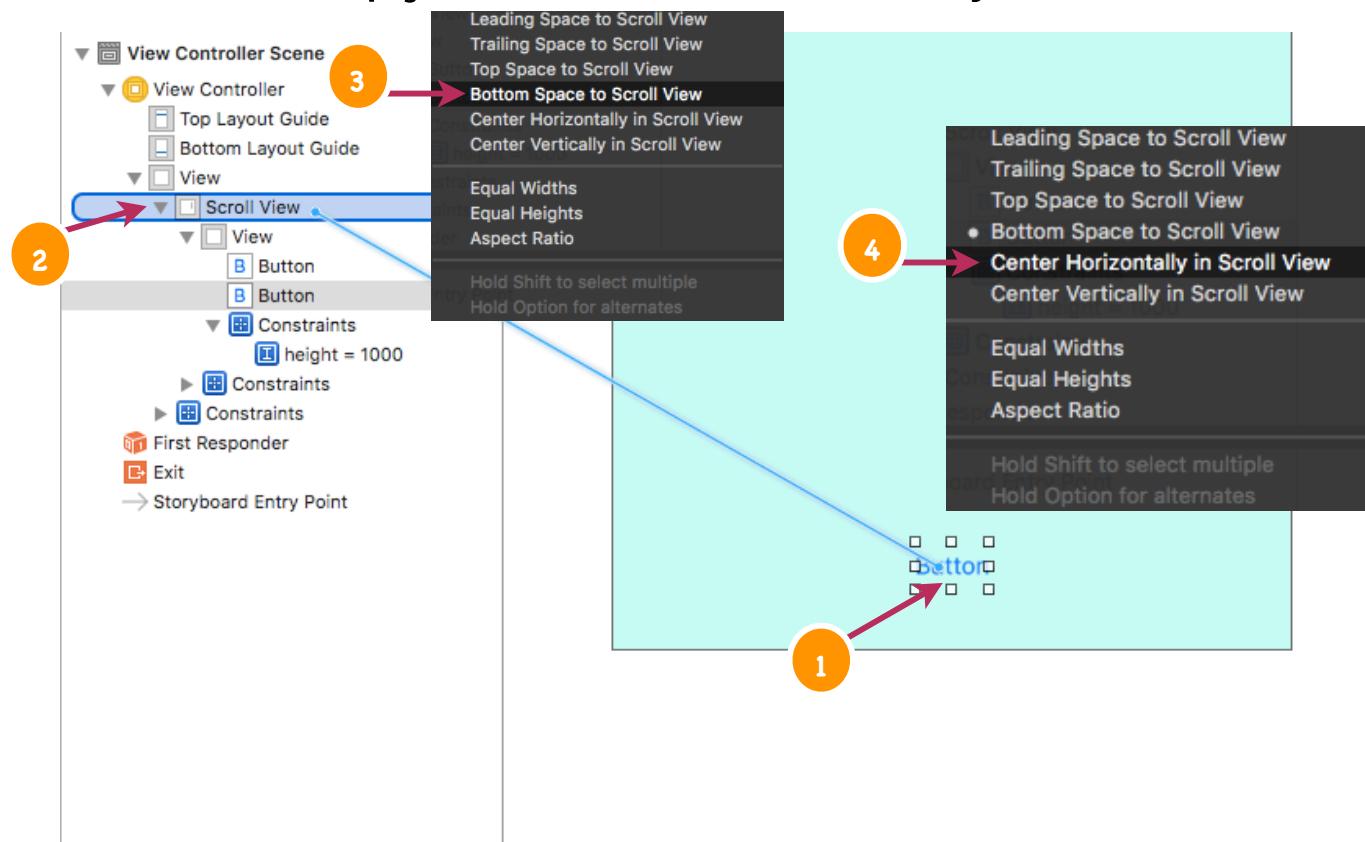
Passo 9: Insira um botão na Barra da View

- Inclua um botão na barra da View(1).



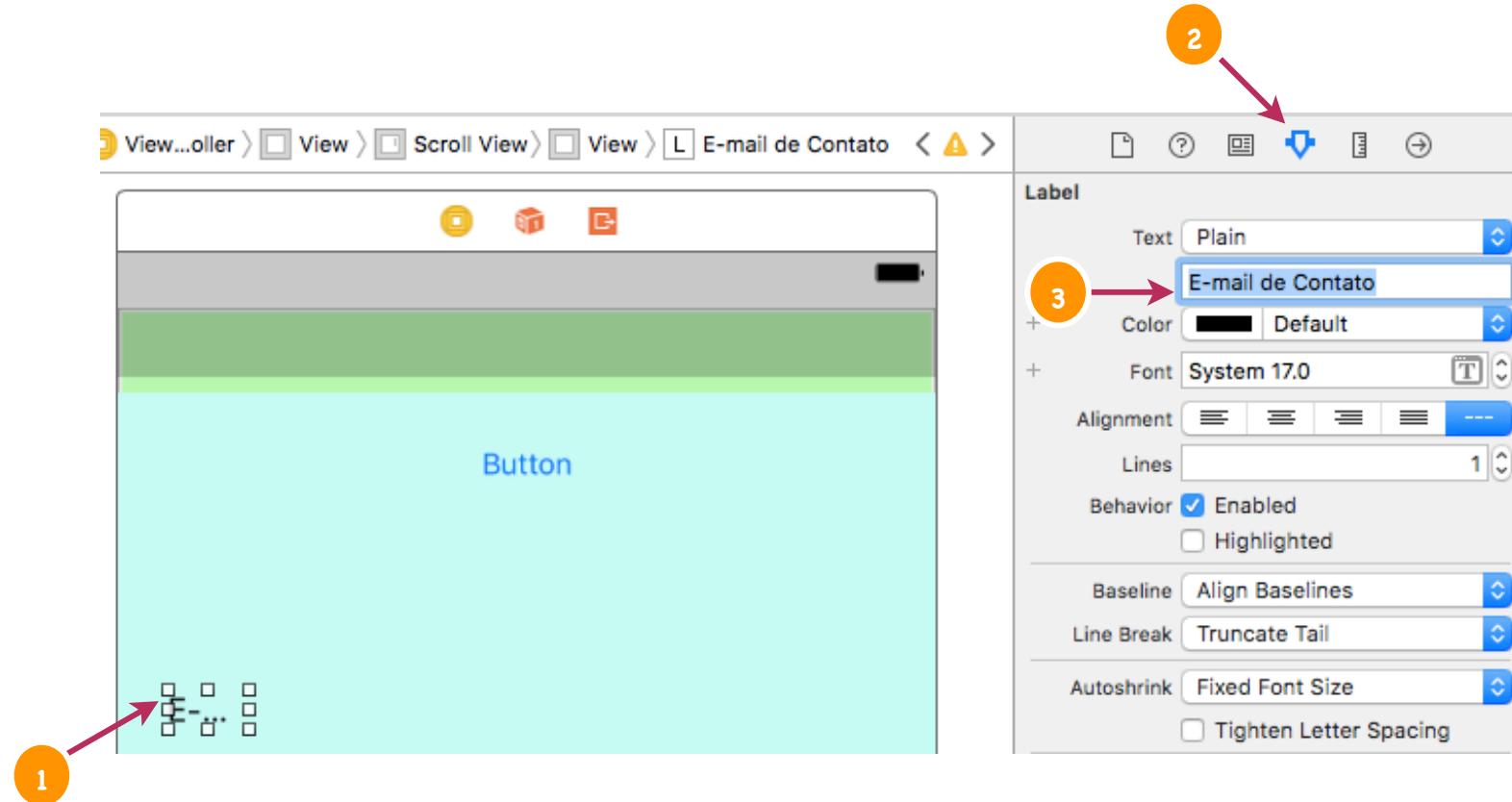
Passo 10: Insira um botão no Barra da View

- Selecione o botão e com a tecla Ctrl selecionada (1), arraste o mouse até a área indicada (2) que representa o ScrollView, ao soltar o botão do mouse selecione a opção “Bottom Space to Scroll View” (3). Repita o passo para marcar a opção “Center Horizontally in Scroll View” (4).



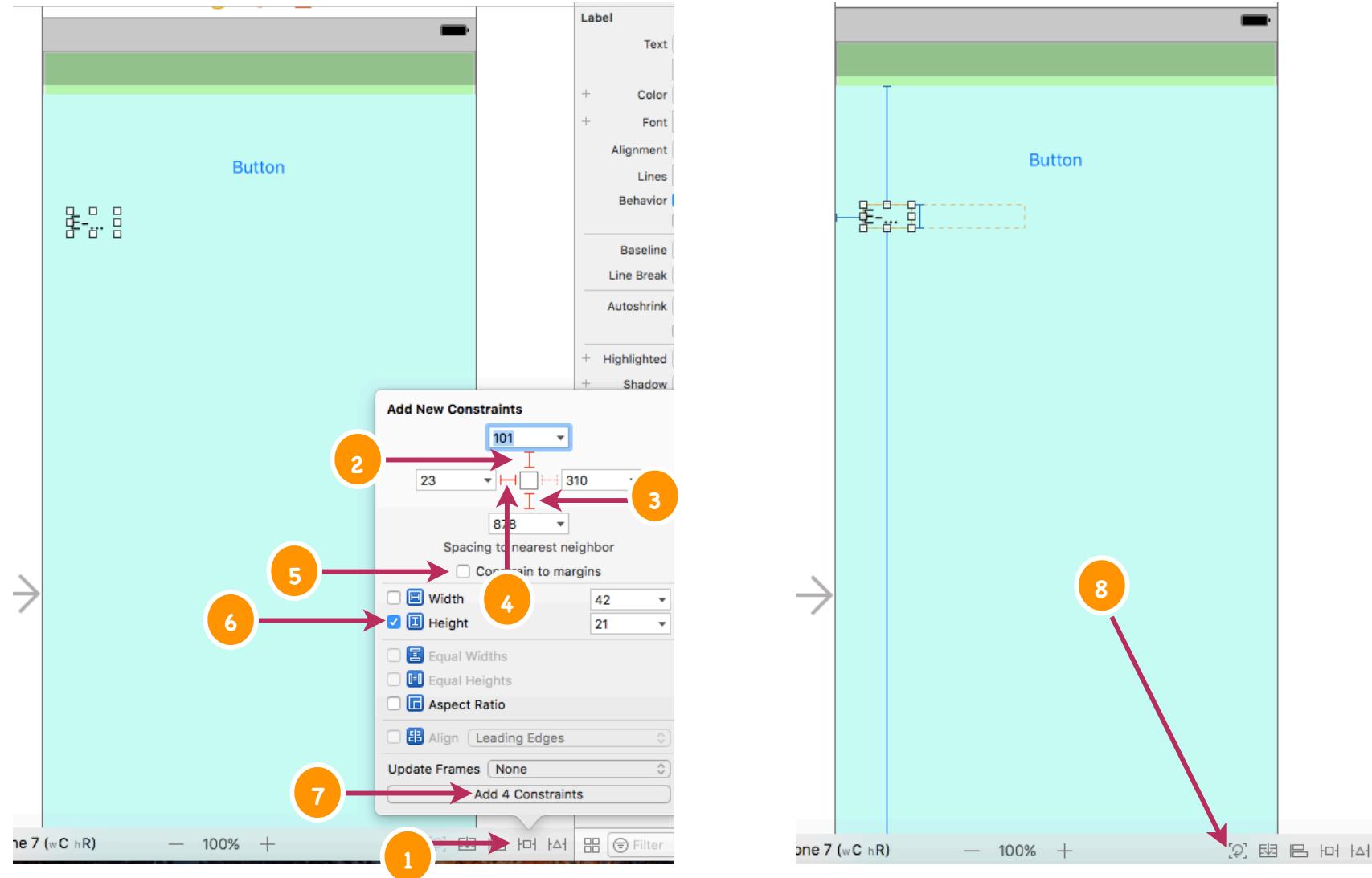
Passo 11: Insira um Label no Topo da View

- Insira um Label no topo da View e altere o atributo de texto para “E-mail de Contato”



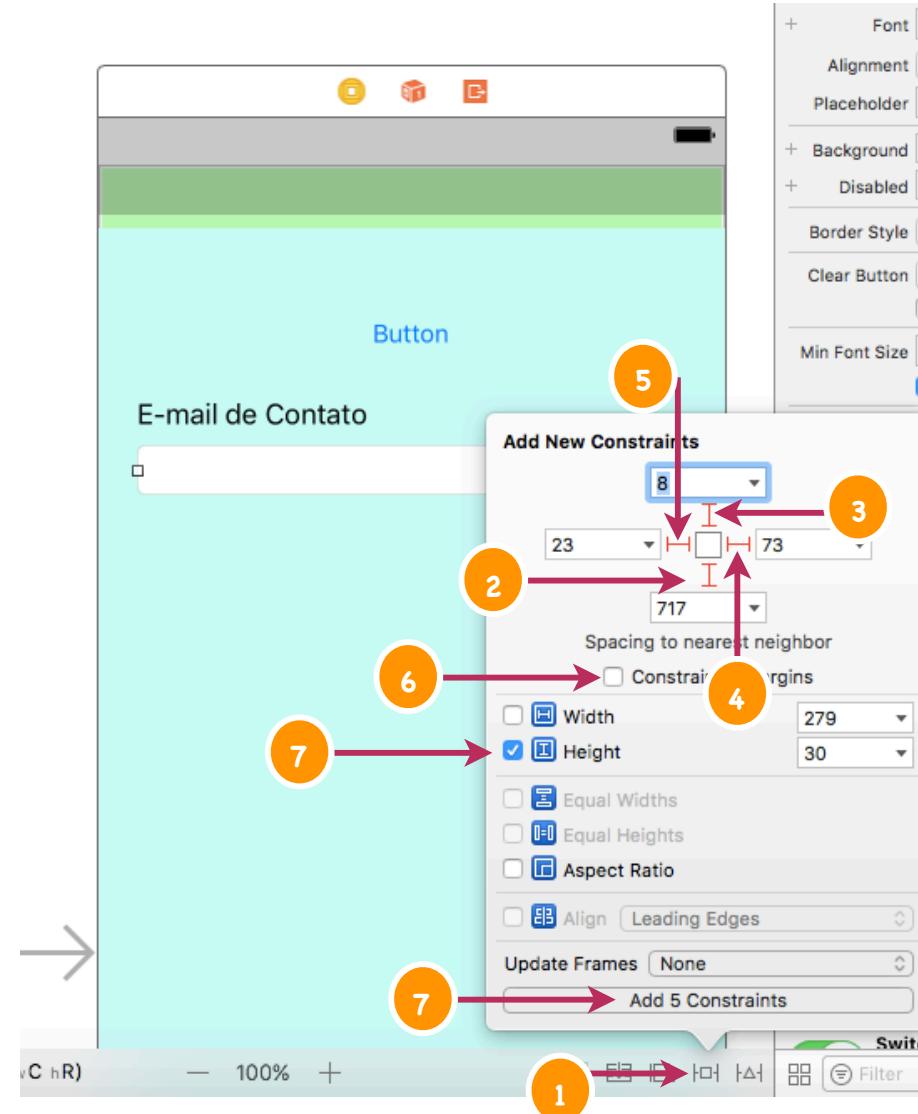
Passo 12: Label e Constraints

- Aplique as Constraints do Label e clique em atualizar.



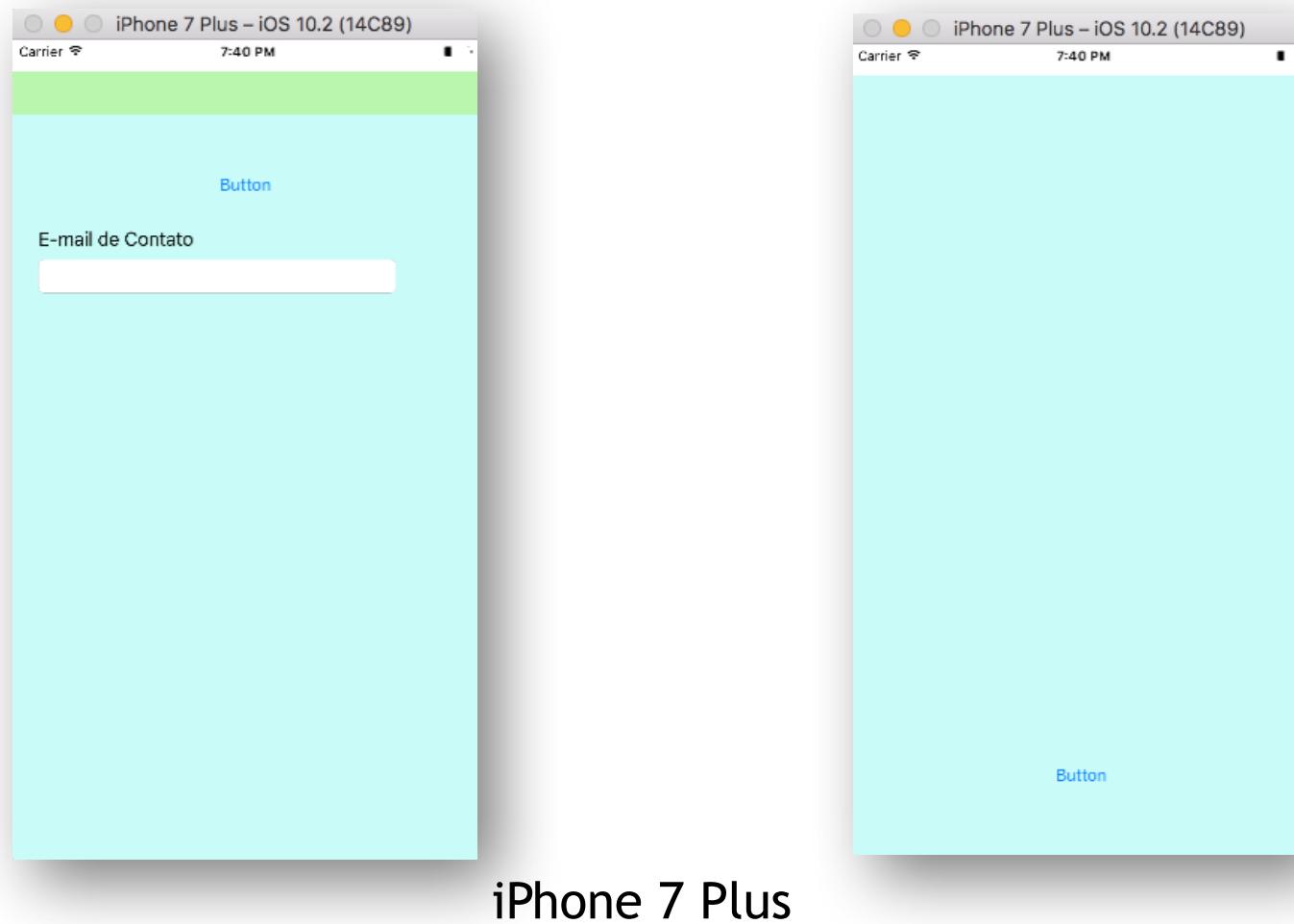
Passo 13: Text e Constraints

- Aplique as Constraints do Text Box e clique em atualizar.



Passo 9: Visualizando os resultados

- Execute seu projeto no iOS Simulator utilizando o iPhone 7 Plus e depois com o iPhone SE.



iPhone 7 Plus

Passo 9: Visualizando os resultados

- Execute seu projeto no iOS Simulator utilizando o iPhone SE.



Copyright © 2017 Prof. Agesandro Scarpioni

Todos direitos reservados. Reprodução ou divulgação total ou parcial
deste documento é expressamente proibido sem o consentimento
formal, por escrito, do Professor (autor).