

Navigation Controller Controller

Utilizando StoryBoard

X-Code – ObjC e Swift
Prof. Agesandro Scarpioni

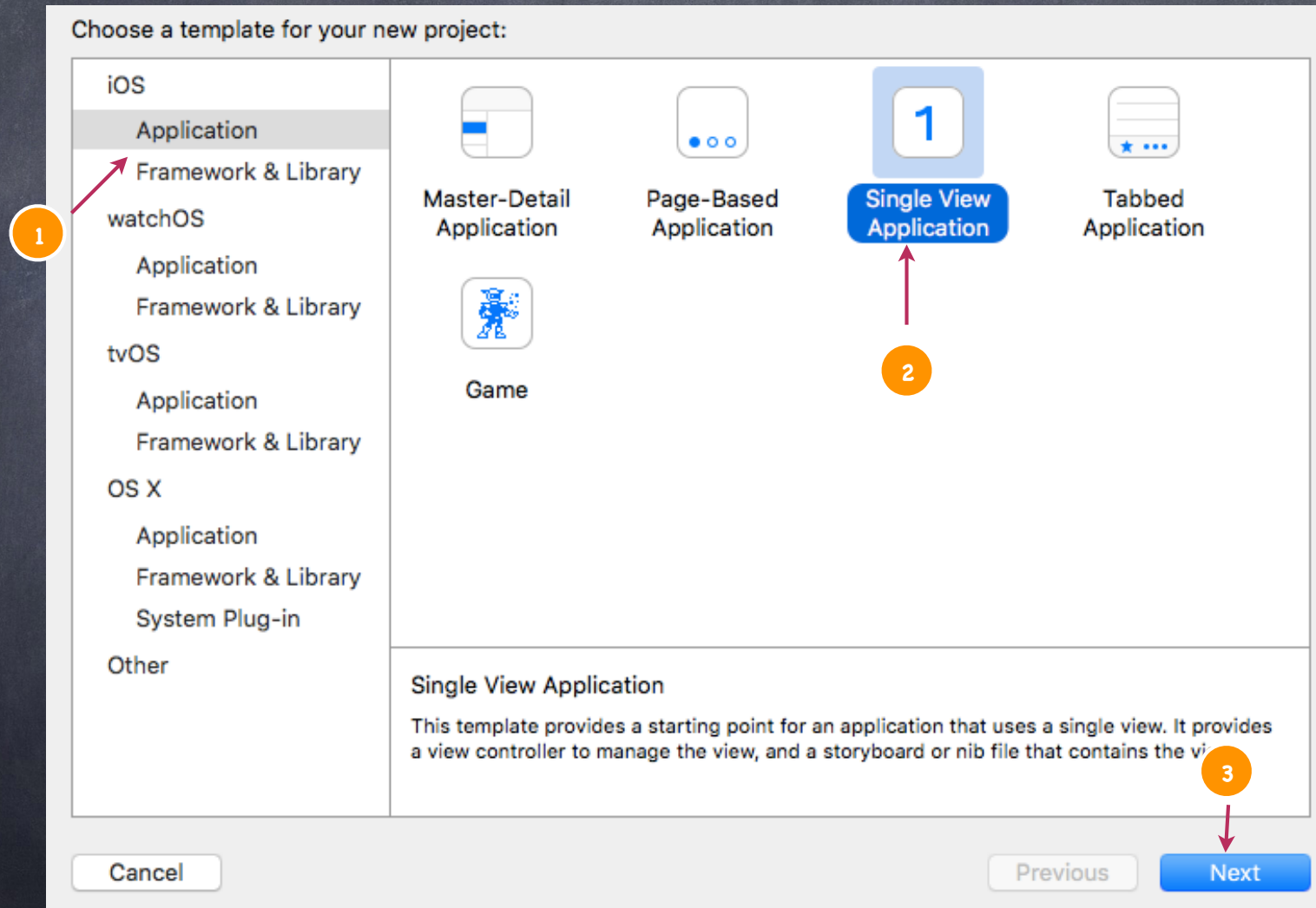
Navigation Controller

com uso de StoryBoard

- Vamos ver como é fácil navegar entre telas utilizando o storyboard, esta funcionalidade foi implementada a partir do IOS 5 com Xcode4.
- Não vamos abordar como faríamos a navegação entre telas da forma mais trabalhosa com uso do arquivo xib.
- Existem vantagens e desvantagens em utilizar o XIB ou Storyboard. Por exemplo: XIB é mais leve o Storyboard é mais prático, por outro lado o XIB é bom para o desenvolvimento de projetos com vários programadores, pois cada um pode se dedicar a uma parte do projeto.

Navigation Controller

- Vamos criar um projeto novo do tipo iOS application (1) - Single View Application (2) clique em Next(3).



Navigation Controller ObjC

- Nomeie o projeto como "NavegandoEntreTelas"(1), escolha o device iPhone(2).

Choose options for your new project:

Product Name:

Organization Name:

Organization Identifier:

Bundle Identifier:

Language:

Devices:

☐ Use Core Data

☐ Include Unit Tests

☐ Include UI Tests

Navigation Controller Swift

- Nomeie o projeto como "NavegandoEntreTelas Swift"(1), escolha o device iPhone(2).

Choose options for your new project:

Product Name:

Organization Name:

Organization Identifier:

Bundle Identifier:

Language:

Devices:

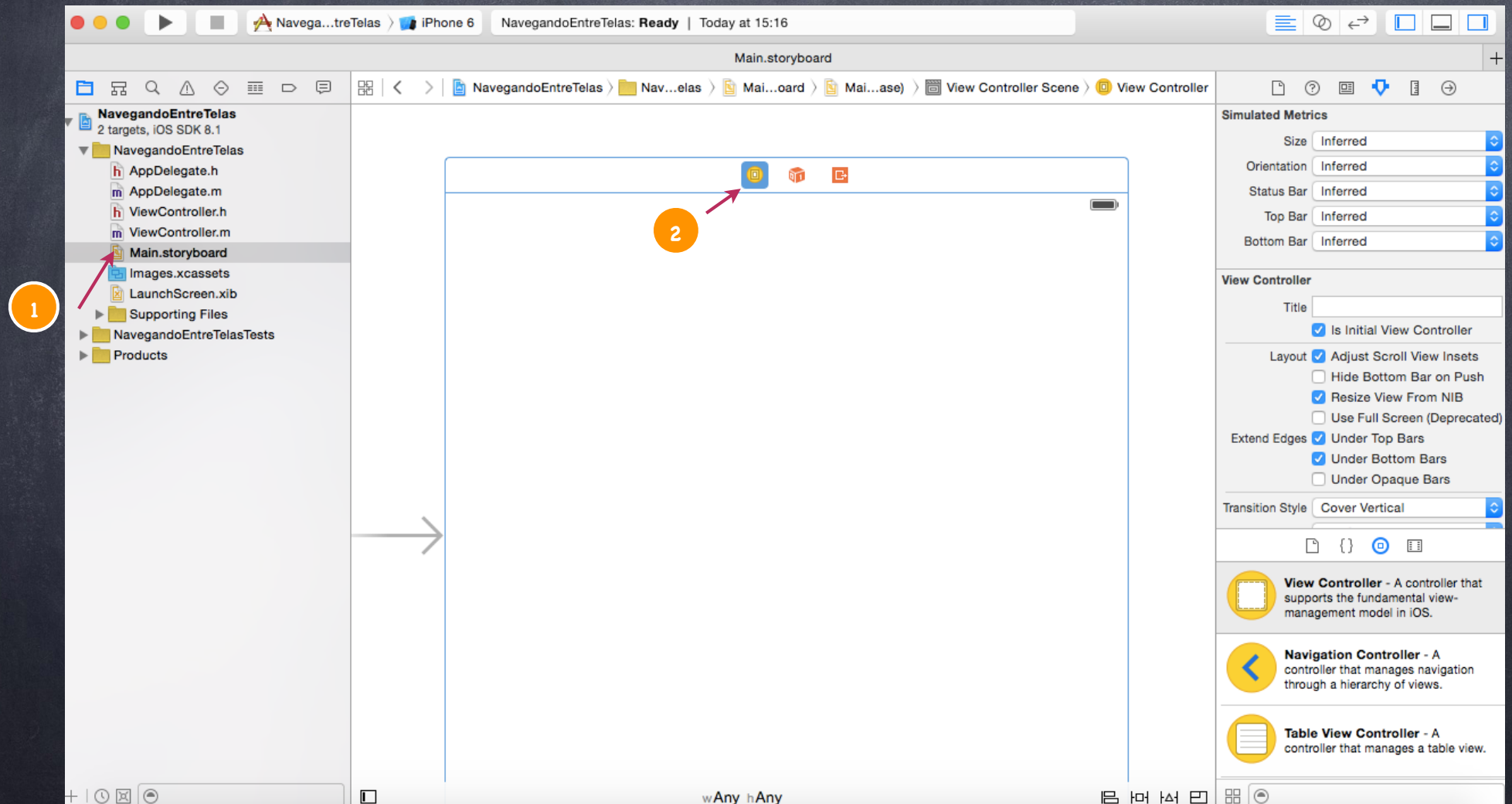
☐ Use Core Data

☐ Include Unit Tests

☐ Include UI Tests

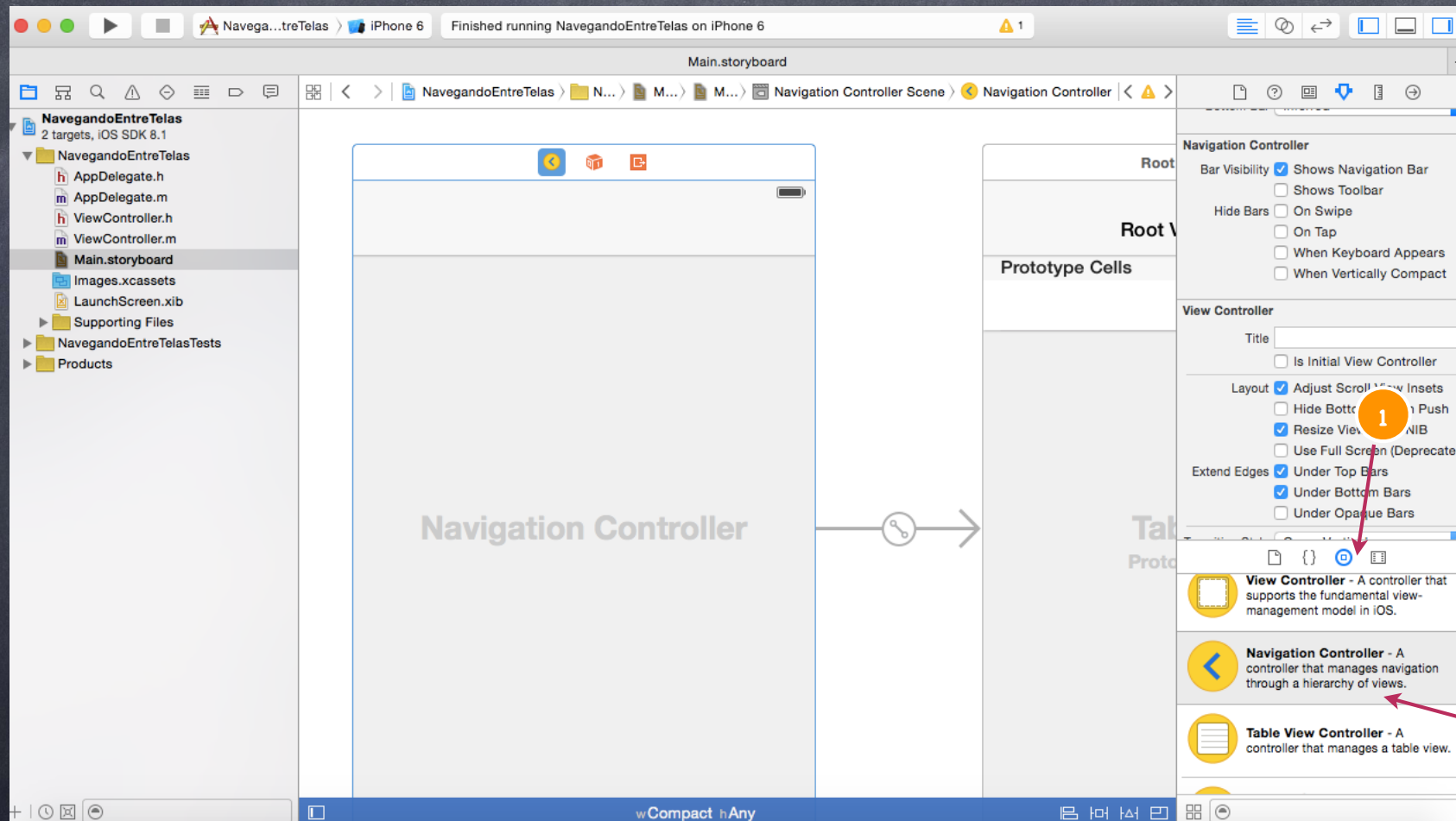
Navigation Controller

- Vá em StoryBoard(1) selecione a viewController(2) e a apague, após apagar esta view mesmo que ao incluirmos outras telas precisaremos ligar a View a um Controller, faremos isso no slide 23.



Navigation Controller

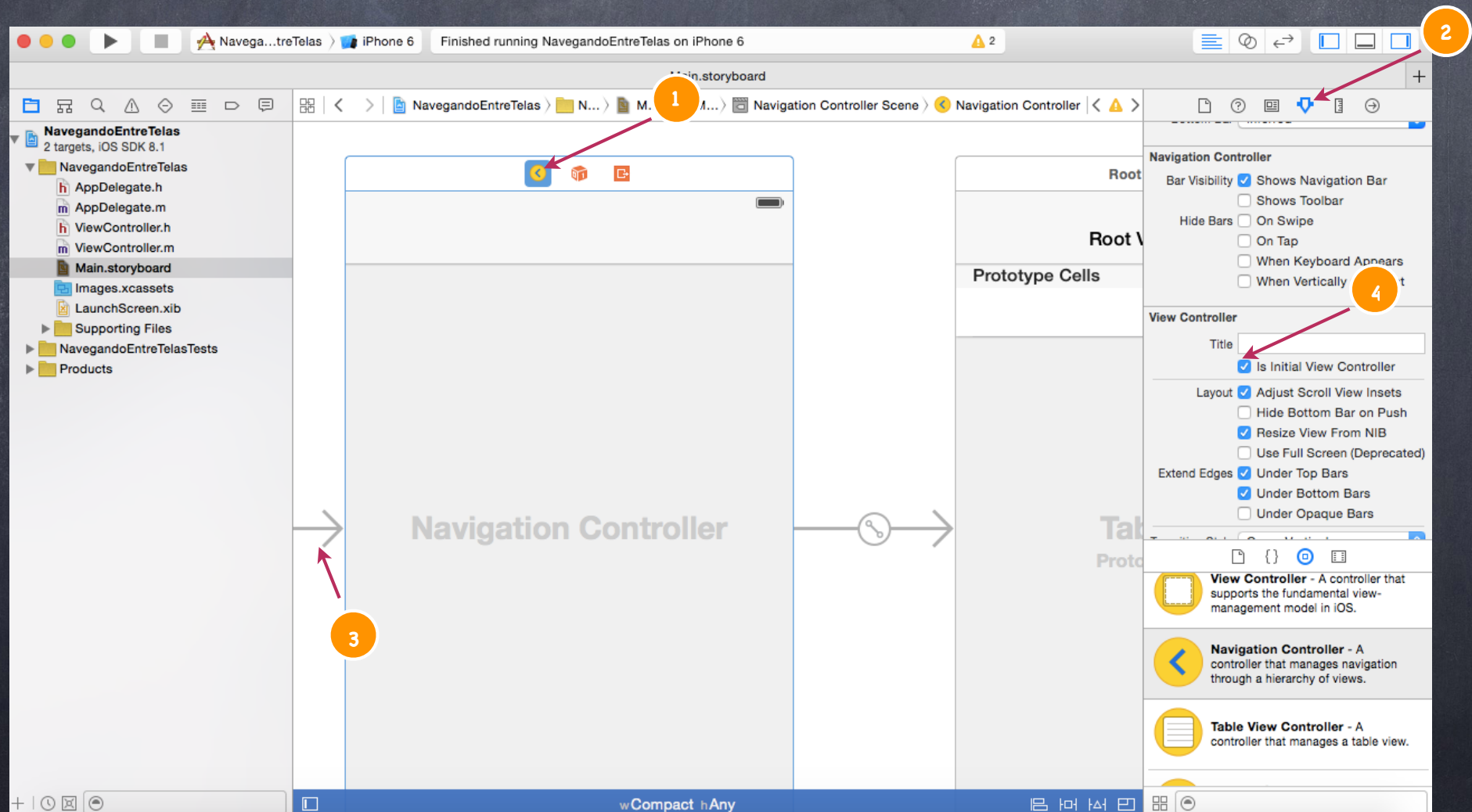
- Em Object Library (1), coloque um Navigator Controller(2) no local de onde foi retirado o ViewController.



Dica: Uma outra opção é ao invés de apagar a View e colocar o Navigator Controller é incluir antes da View o Navigator pelo menu → Editor → Embed In - Navigation Controller, até porque nos próximos passos iremos recolocar a View em nosso StoryBoard.

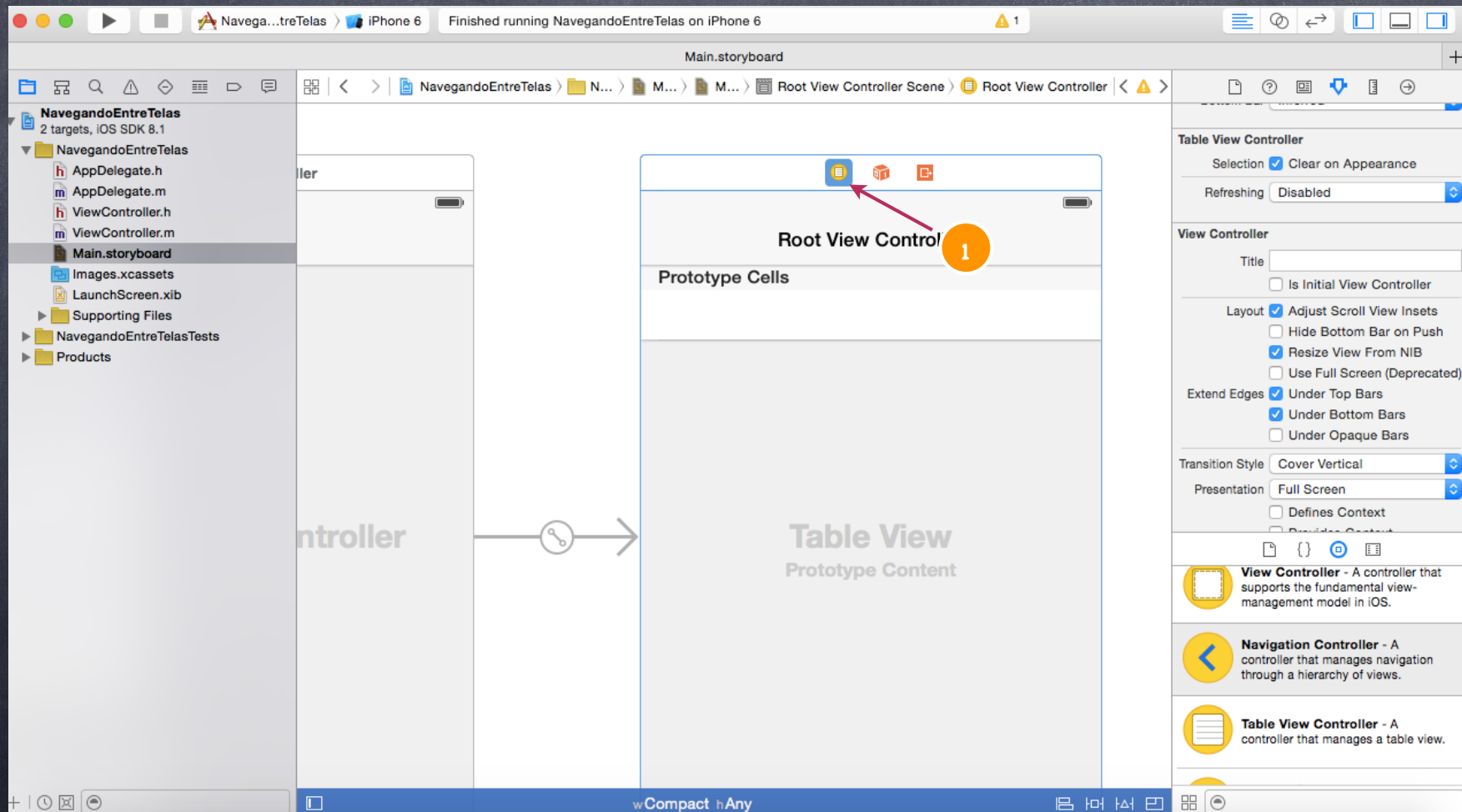
Navigation Controller

- Ao executar Command + R não ira funcionar pois não sabemos quem será a View Inicial, para informar a view inicial clique nos itens sinalizados (1,2,3,4), e ira aparecer uma seta indicando a View Inicial(3), após esse passo ao executarmos o programa uma tela irá aparecer.



Navigation Controller

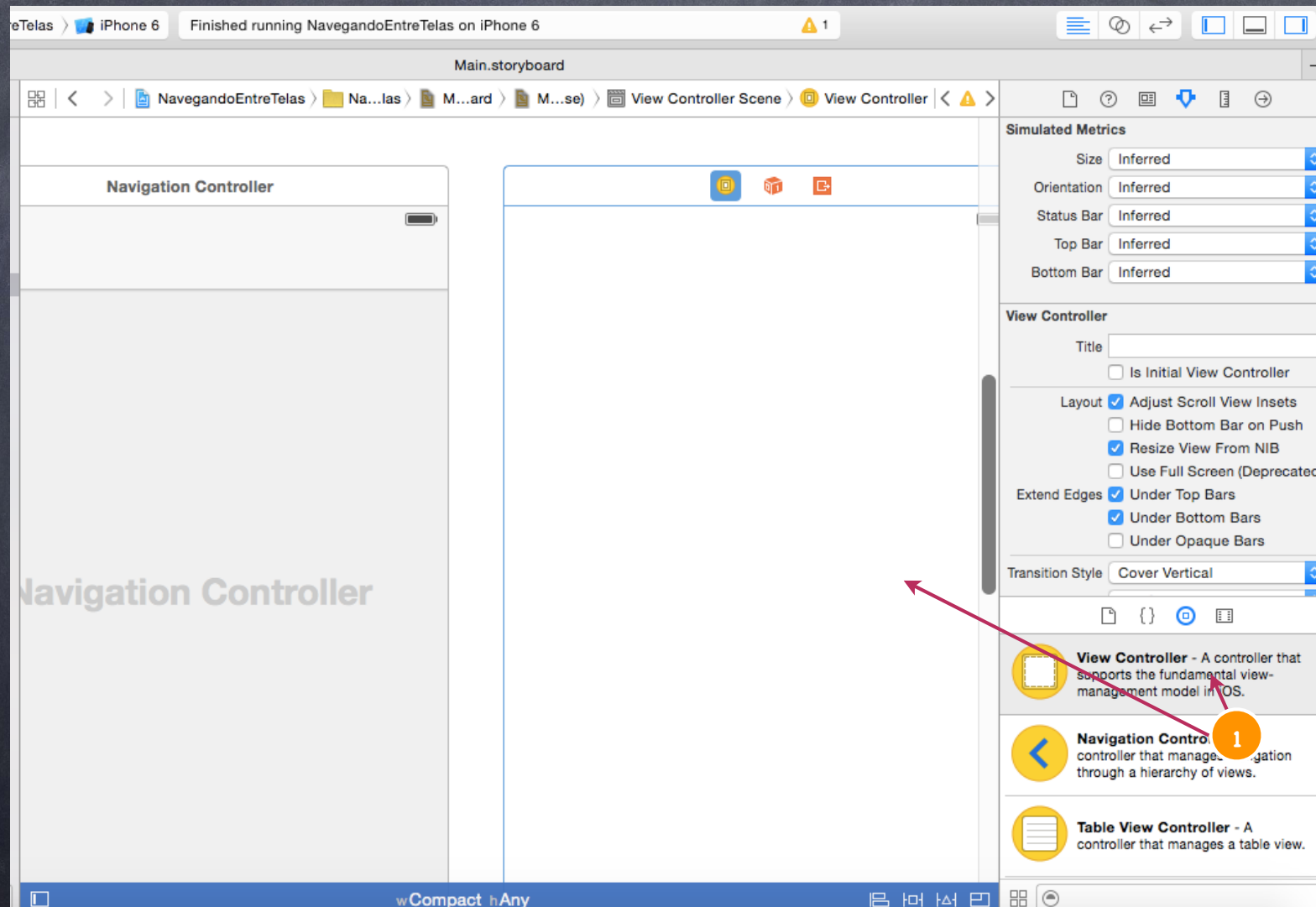
- Selecione a segunda tela(1) e a apague, não vamos utilizar um Table View, teremos posteriormente uma aula só para TableView.



OBS: Não é possível navegar entre Scenes (Views) sem um Navigation Controller.

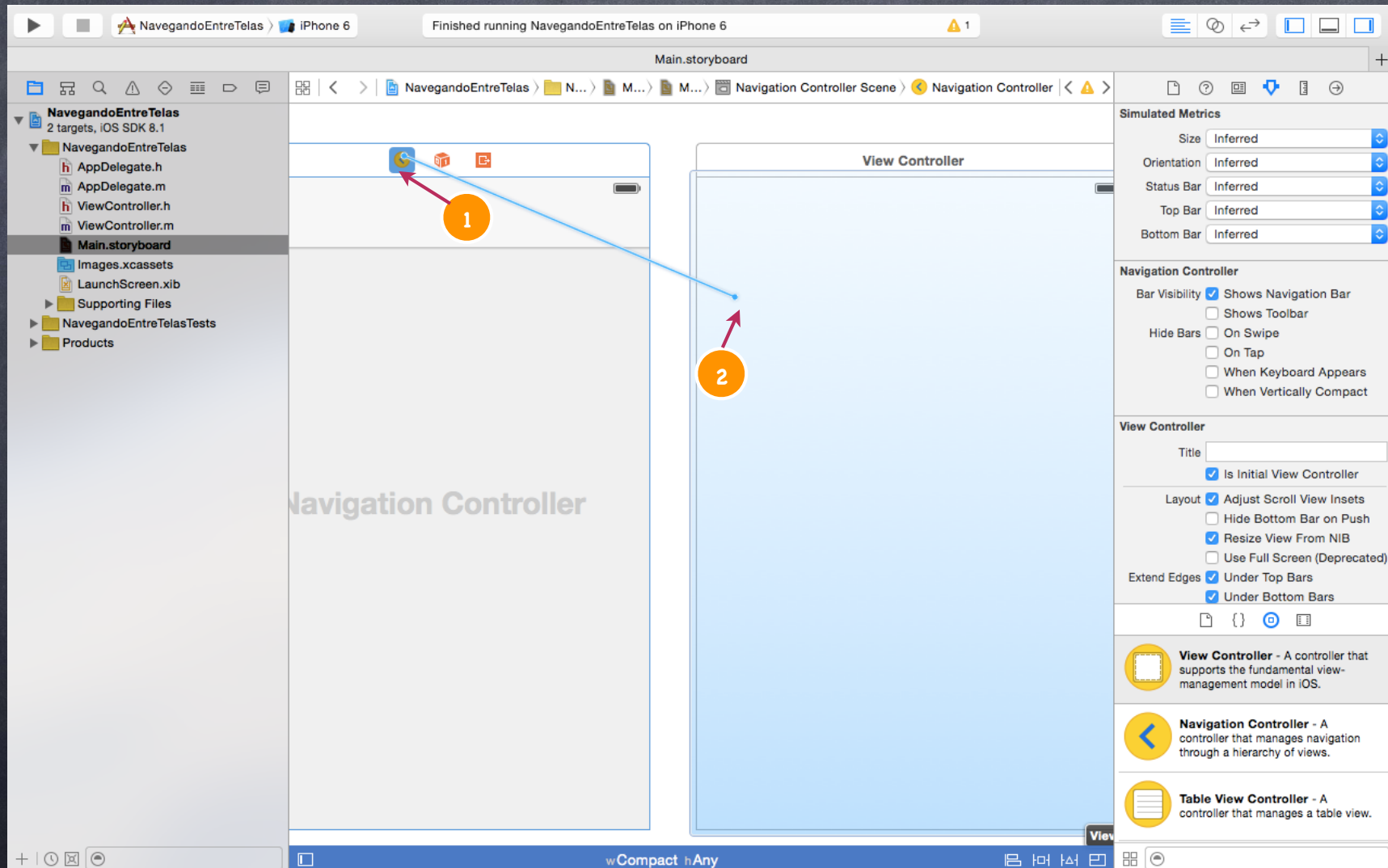
Navigation Controller

- Inclua no local da segunda tela uma tela do tipo View Controller(1).



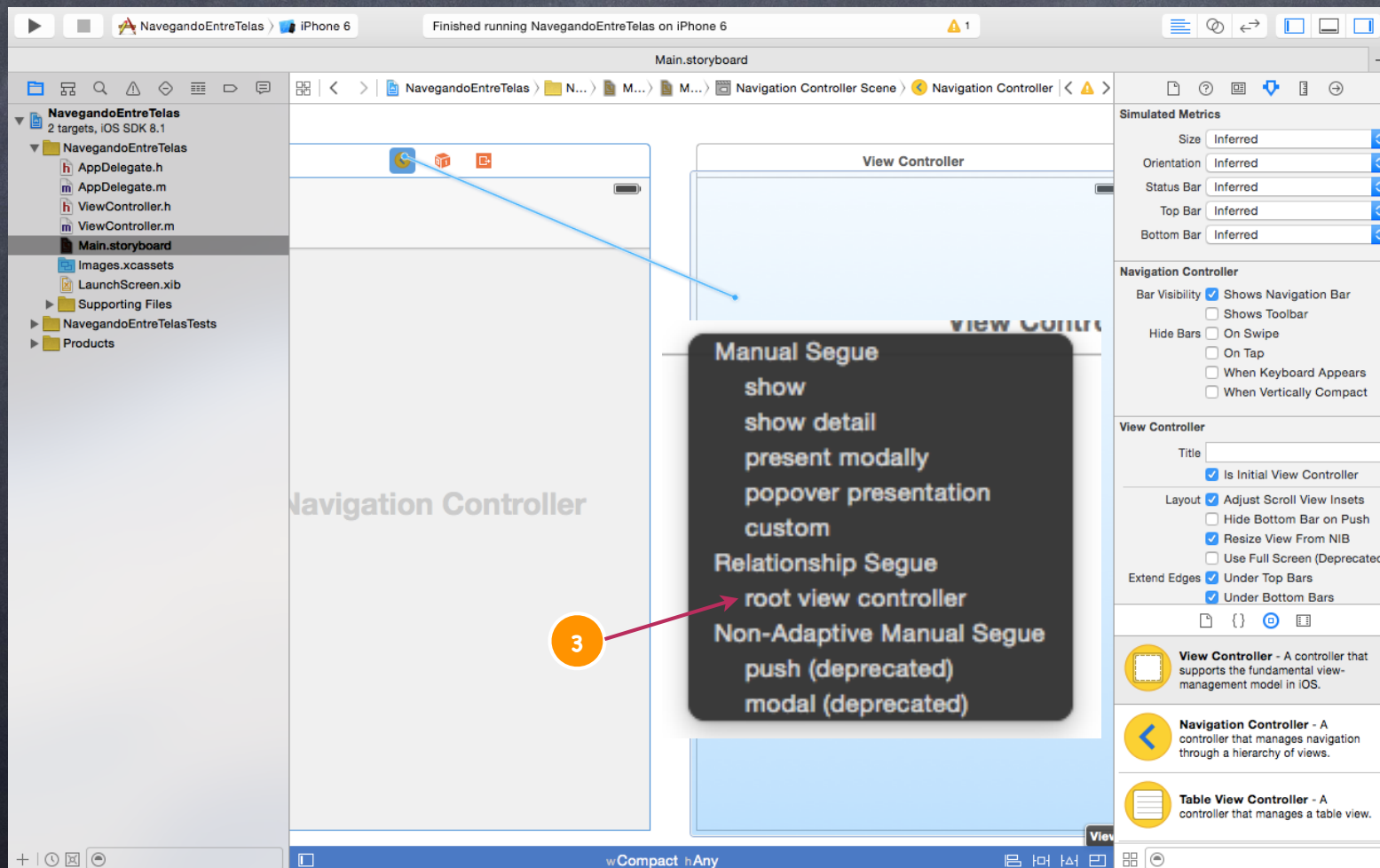
Navigation Controller

- Vamos refazer a conexão do Navigation com o segunda tela, segurando o control e clicando com o botão direito sobre o Navigation(1) e arrastando até o View Controller(2).



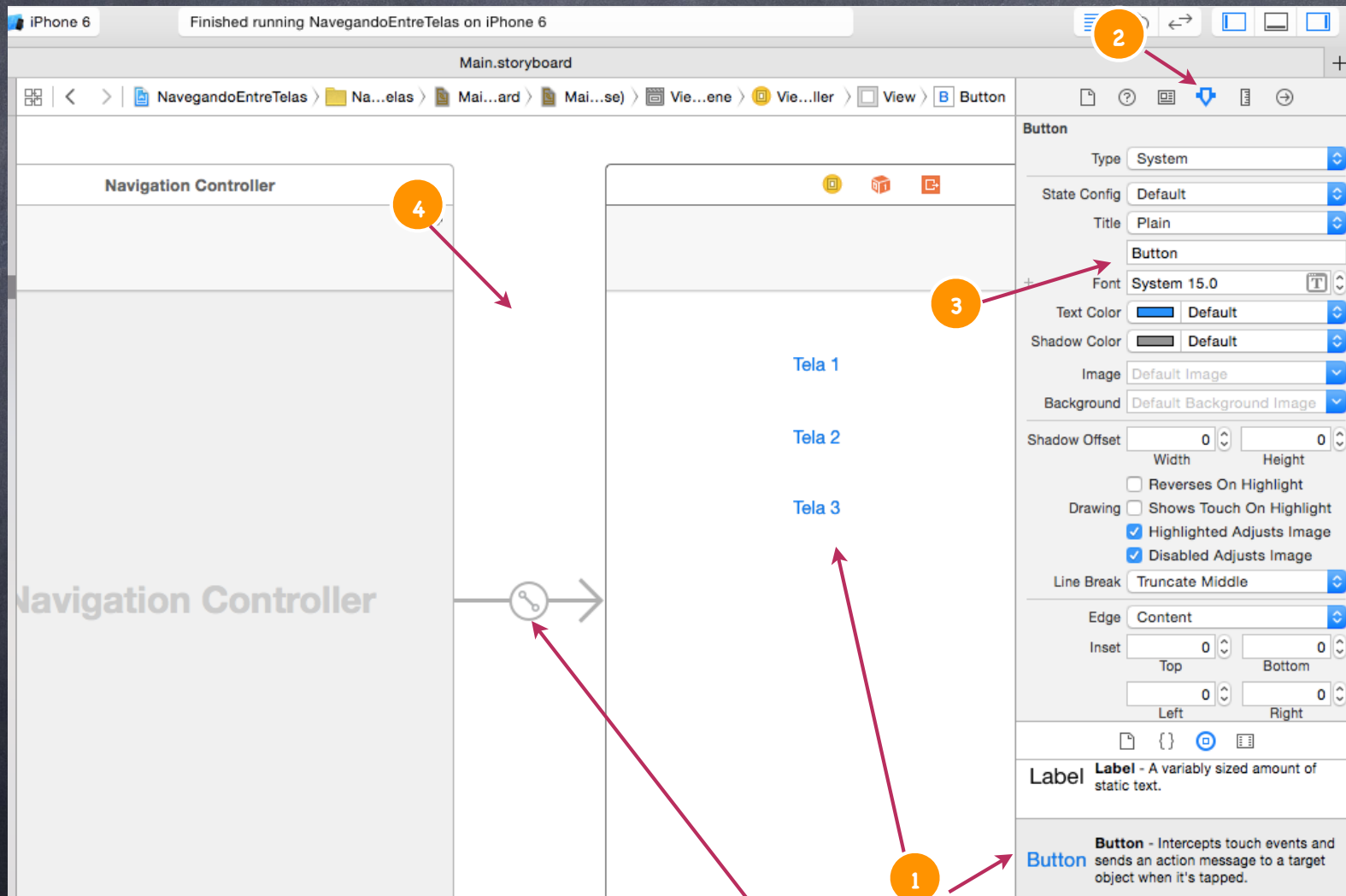
Navigation Controller

- Ao clicar com o botão direito sobre o Navigation e arrastar o mouse até o View Controller, um popup irá aparecer, escolha a última opção "root view controller" (3).



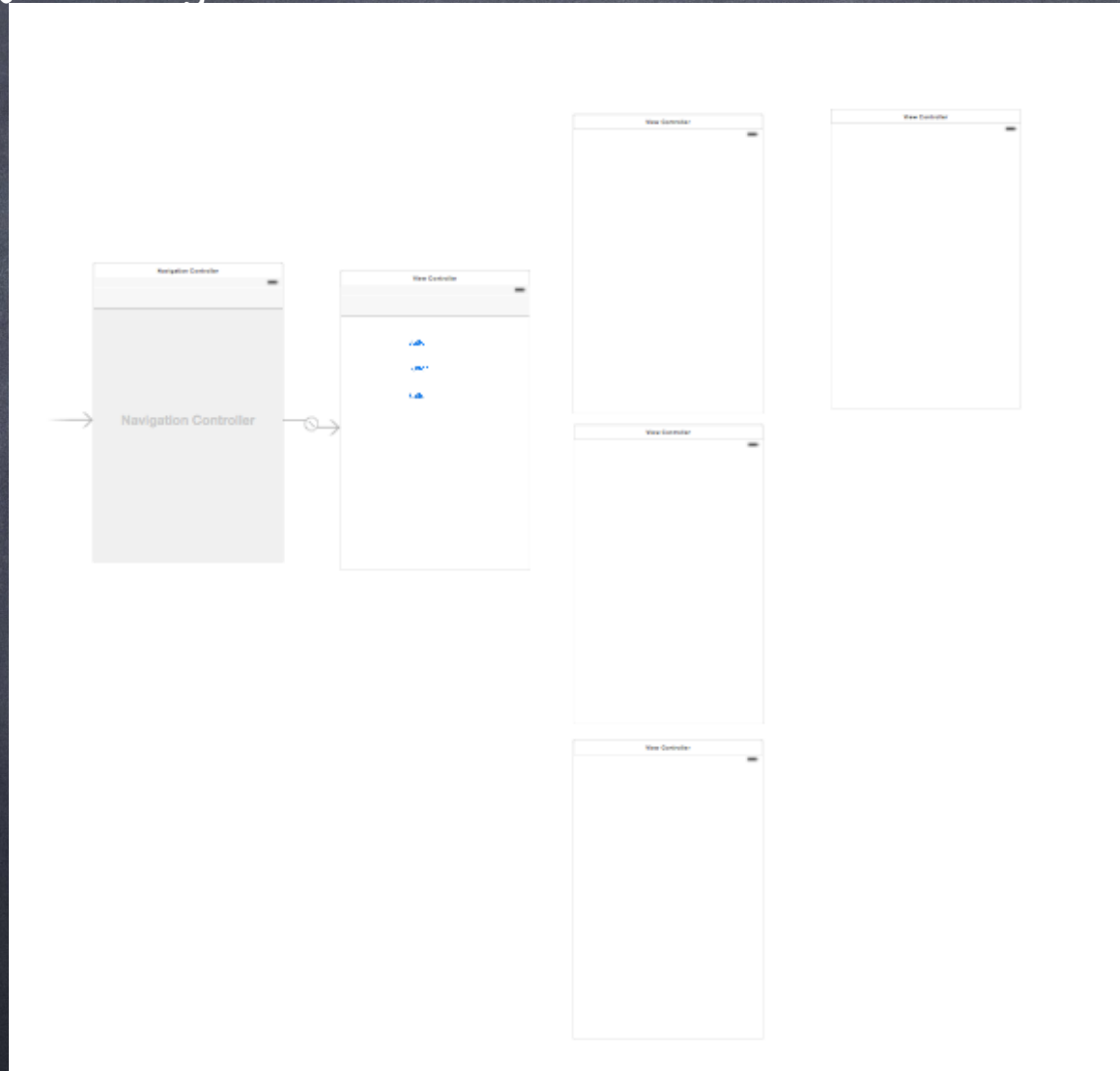
Navigation Controller

- Inclua 3 botões do tipo Button (1), no attributes inspector (2) troque os textos dos botões (3) como mostra na figura, dê um doble click na área branca (4) para aumentar ou diminuir o zoom.



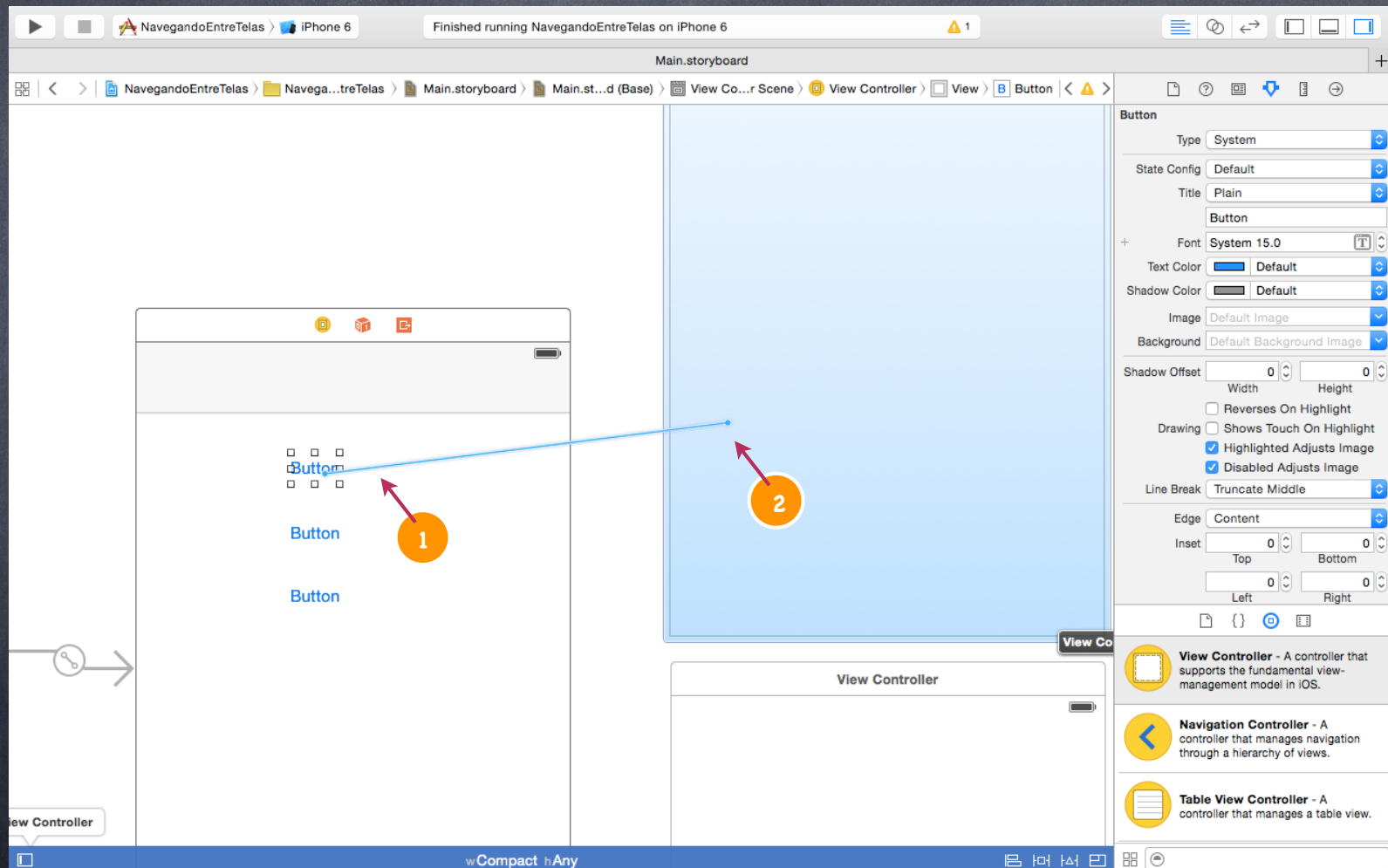
Navigation Controller

- Adicione outras telas do tipo ViewController para que possamos criar uma estrutura de navegação entre elas. Para aumentar ou diminuir o Zoom clique 2x na área do Storyboard. Veja a imagem abaixo:



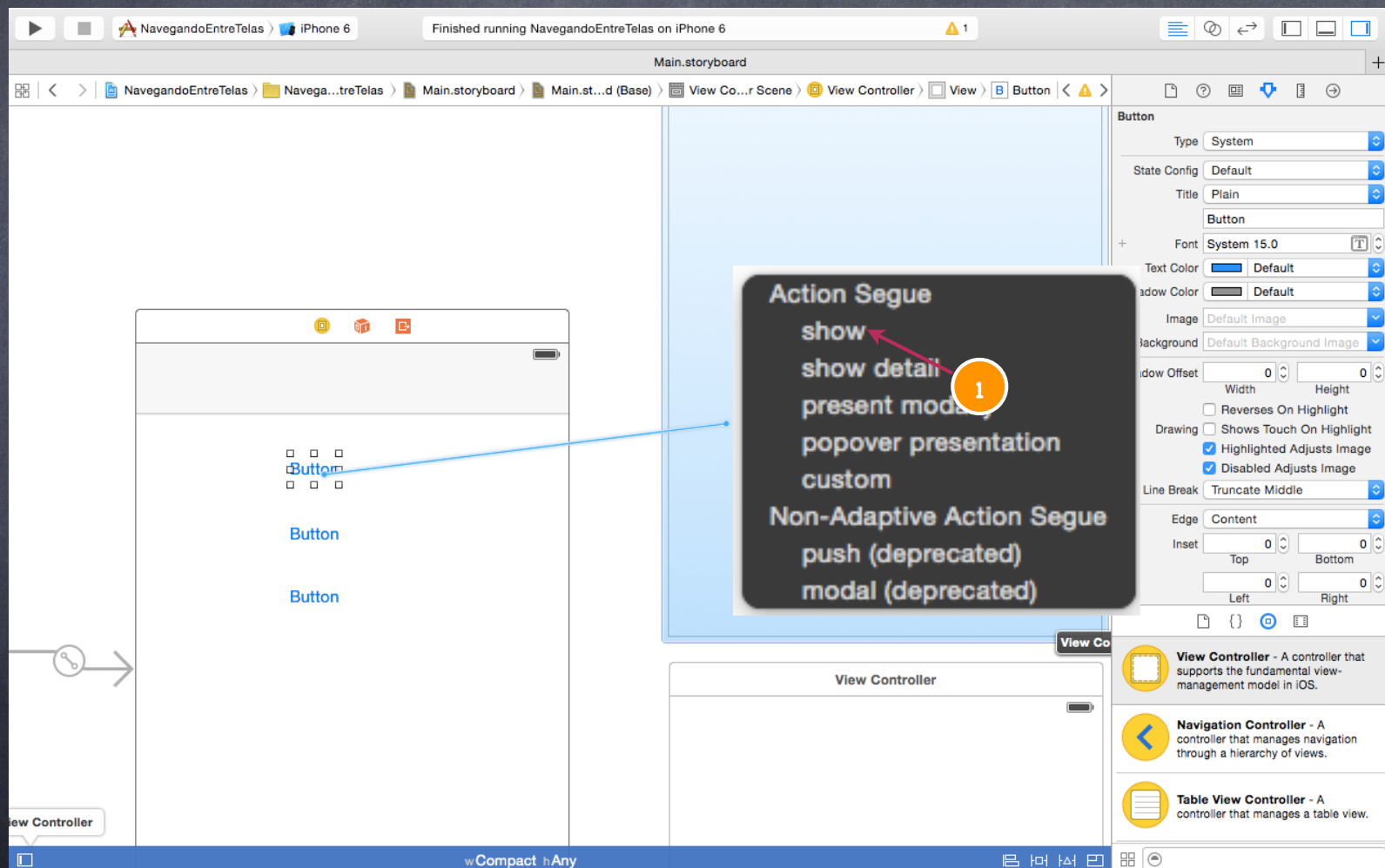
Navigator Controller

- Para ligar um botão a uma tela, clique com o botão direito sobre o objeto(1) e com o control apertado arraste o mouse até a área da ViewController (2) que você deseja interligar.



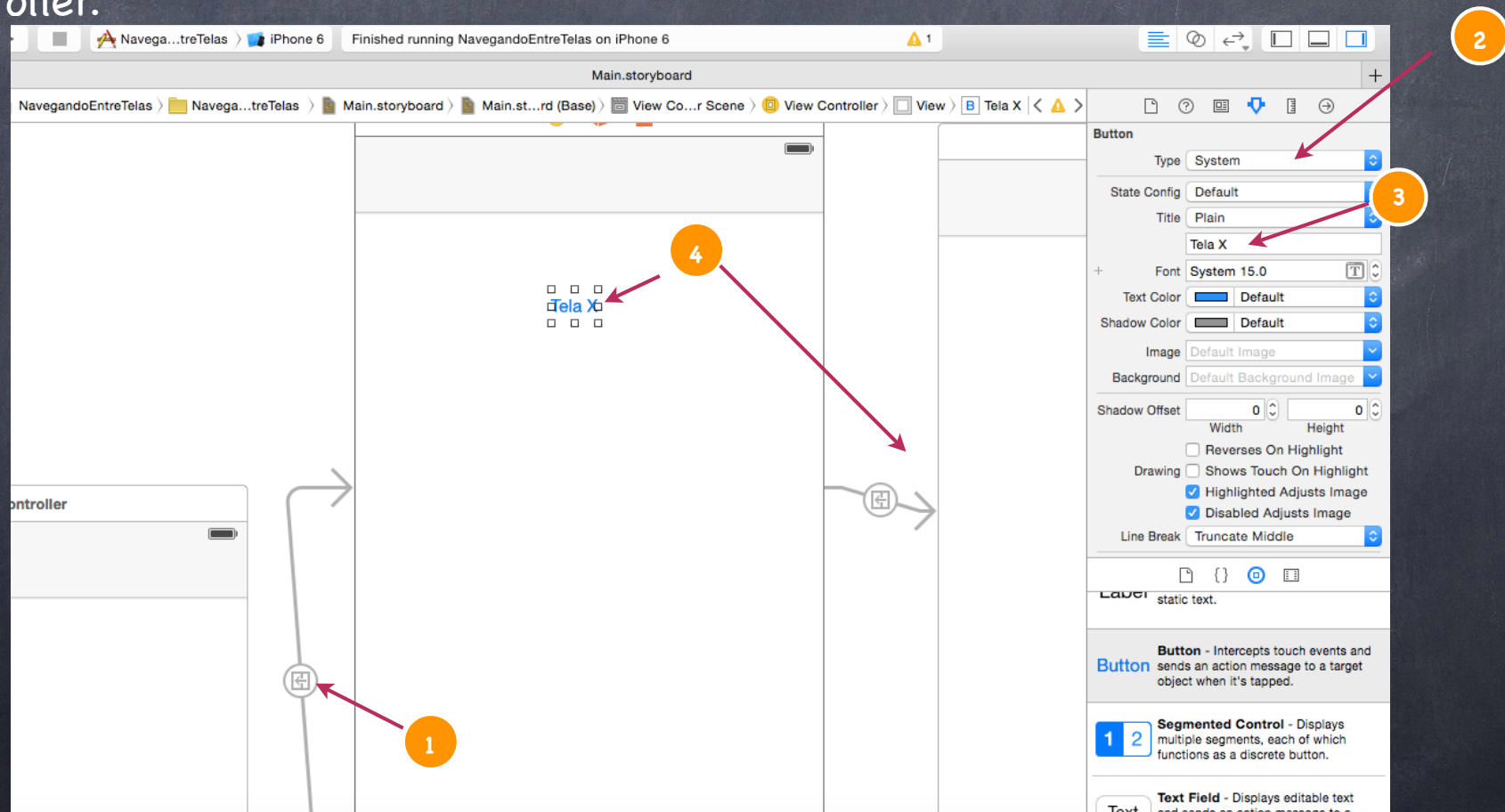
Navigation Controller

- Ao soltar o botão do mouse irá aparecer um popup e escolha Show(1).



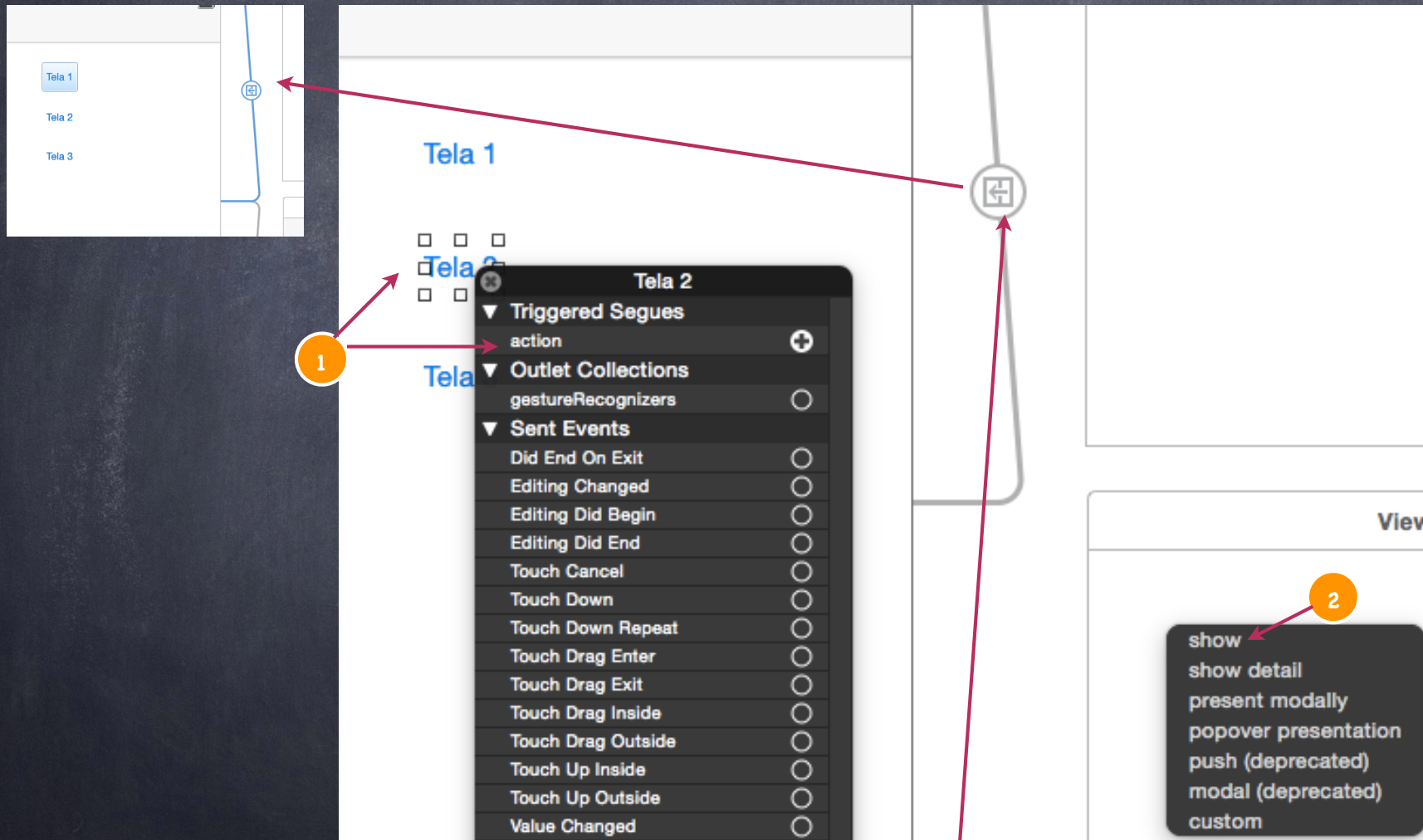
Navigation Controller

- Repare que de forma automática apareceu uma marcação (1) indicando o fluxo da tela, coloque um botão novo na próxima tela que acabamos de "ligar", troque o nome (3) para Tela X, faça isso pelo attributes inspector (2). Faça uma nova ligação do tipo show desse botão para a próxima tela (4). Lembre-se que para fazer a ligação, basta clicar com o botão direito sobre o Tela X e com o control pressionado arrastar até a próxima ViewController.



Navigation Controller

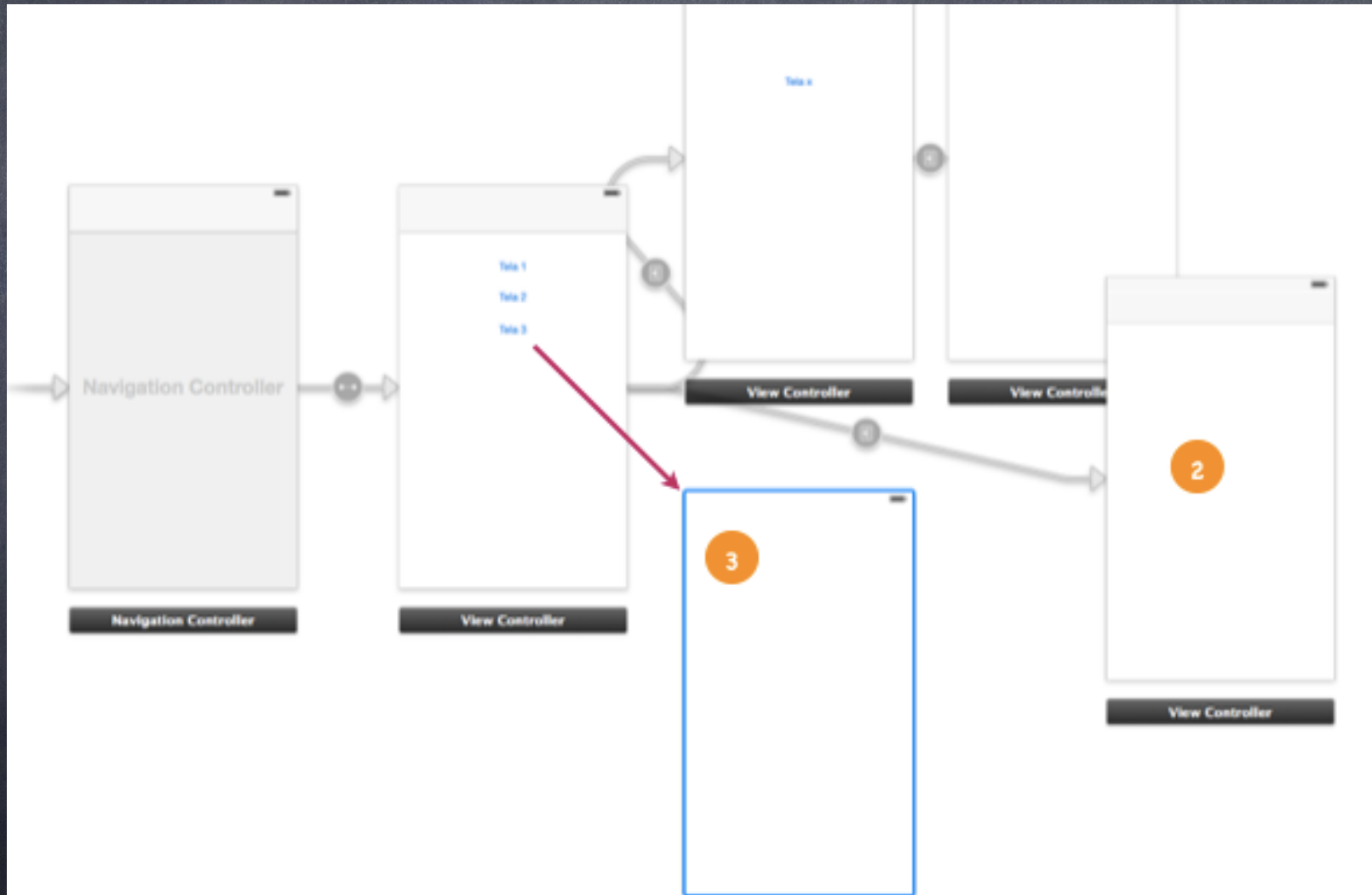
- Repita o passo para o botão "Tela 2" (1) clique em action e arraste até a outra view fazendo uma ligação do tipo show (2).



Dica: Ao clicar no nó que liga as Scenes ou Views é exibido em destaque o botão que chama a Scene em questão.

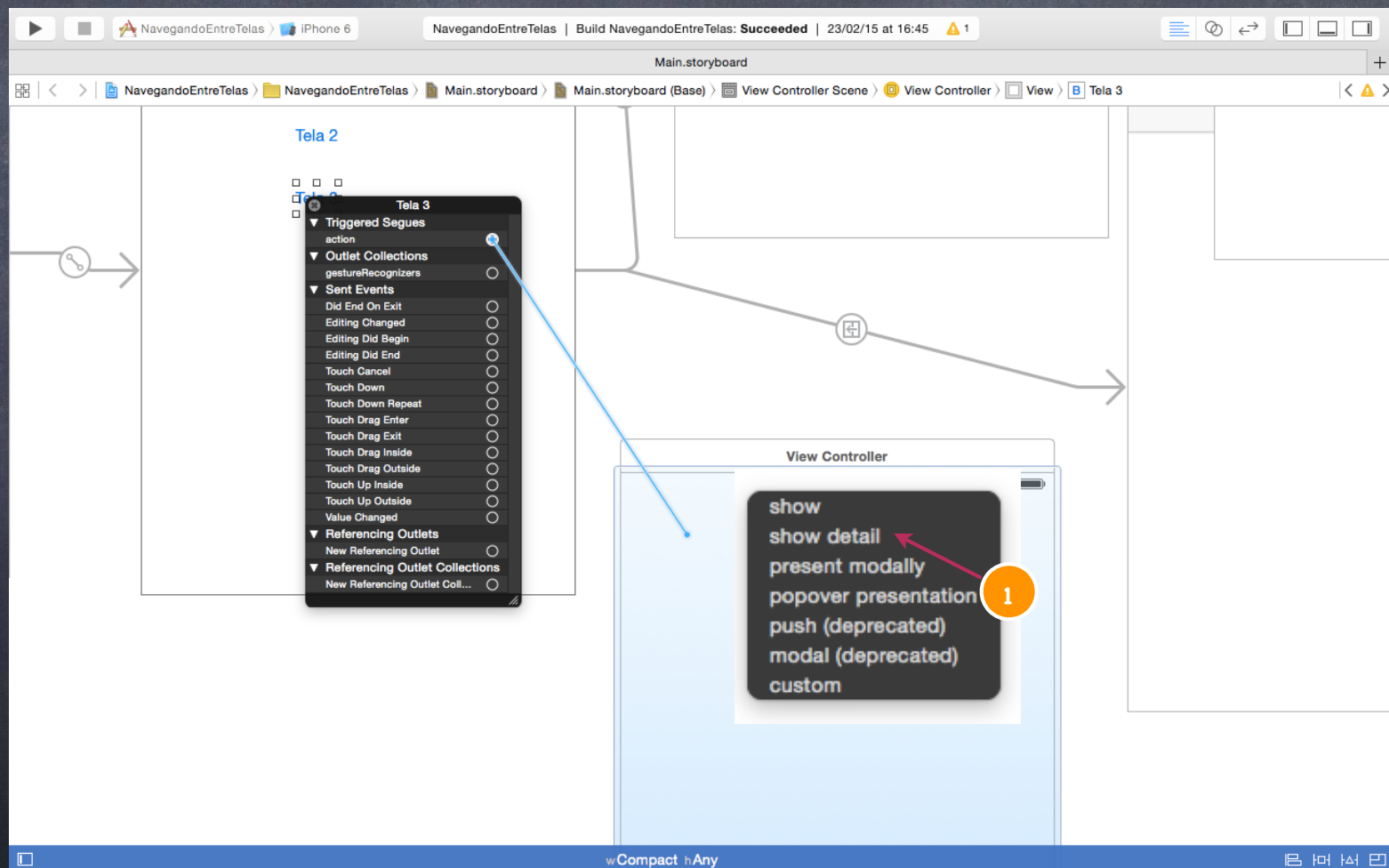
Navigation Controller

- Diminua o Zoom e reorganize (2, 3) suas telas para que possamos fazer a ligação do botão “Tela 3” com a última ViewController (3).



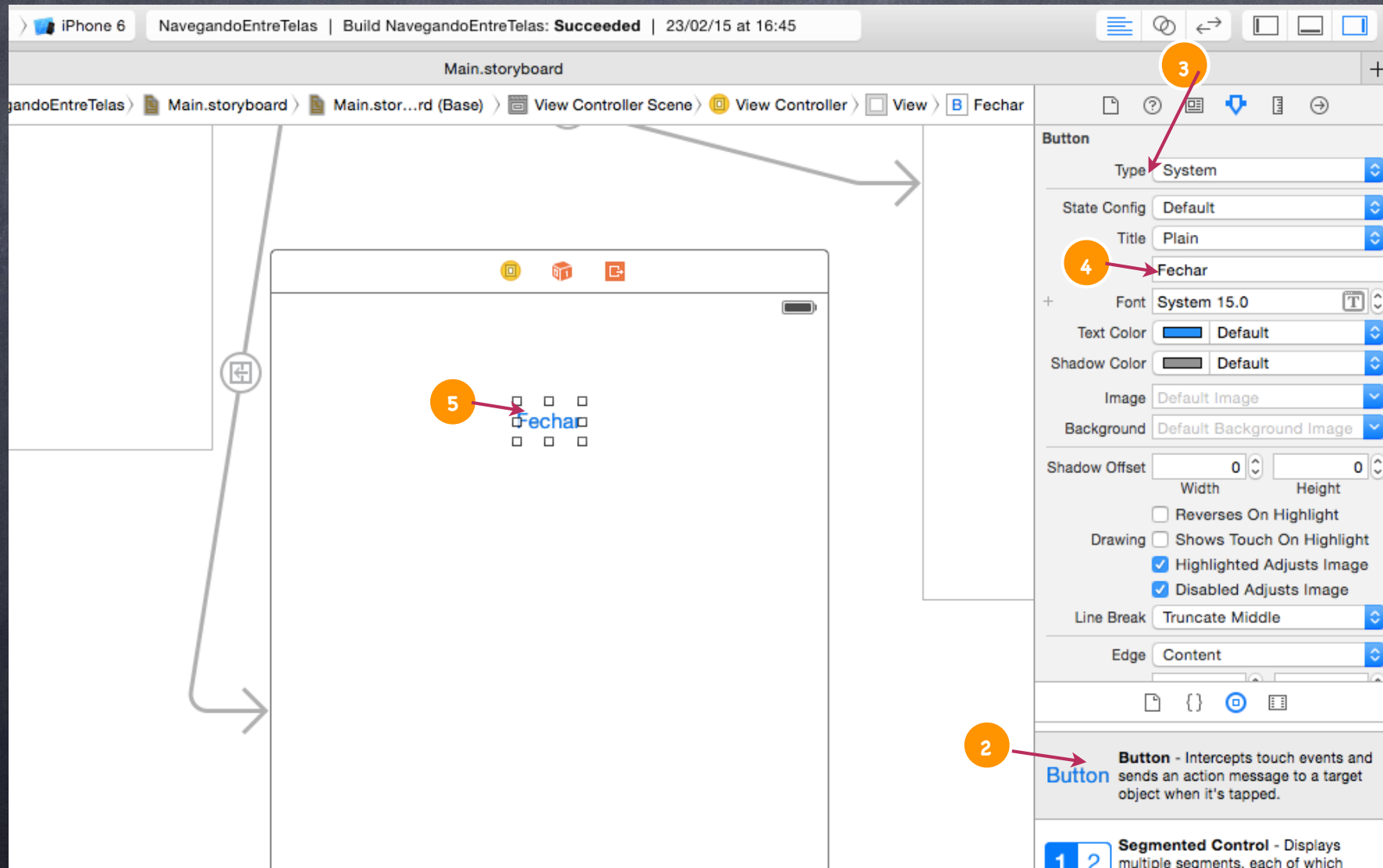
Navigation Controller

- Repita o processo clicando com o botão direito sobre o botão Tela 3 e arraste até a quarta e última View Controller, porem, não escolha Show, escolha **Show detail** ou seja o antigo **modal**.



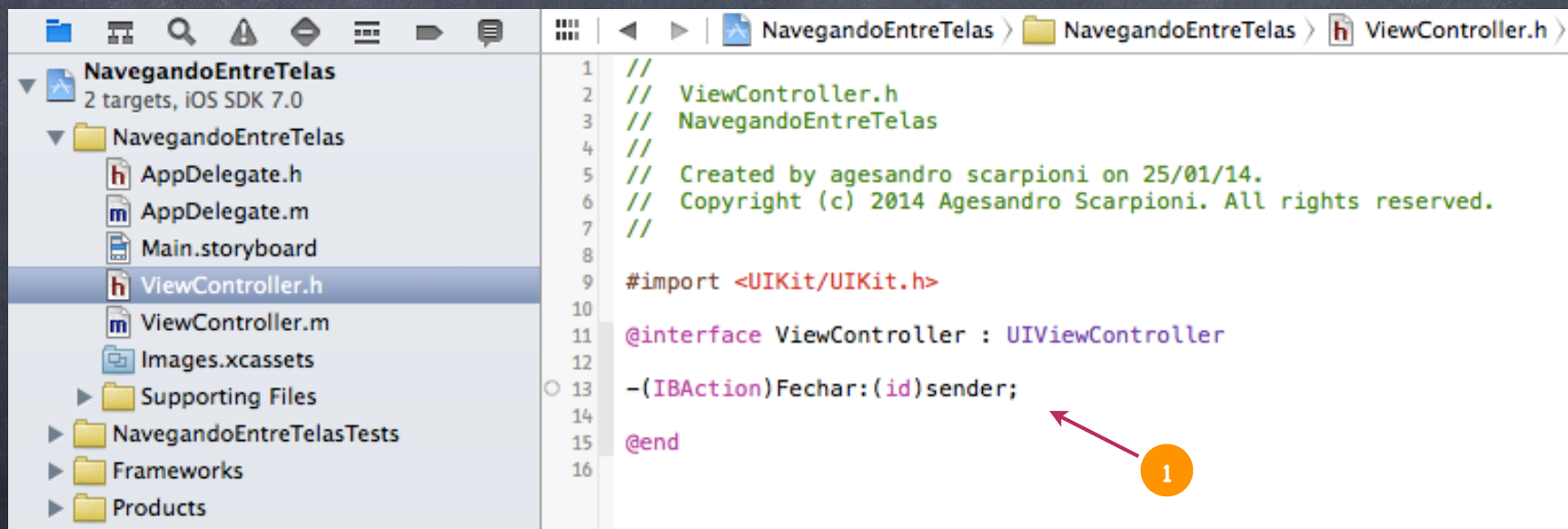
Navigation Controller

- Coloque um Button (2), altere no Attributes Inspector (3) o texto para Fechar (4), ou digite diretamente no botão (5).



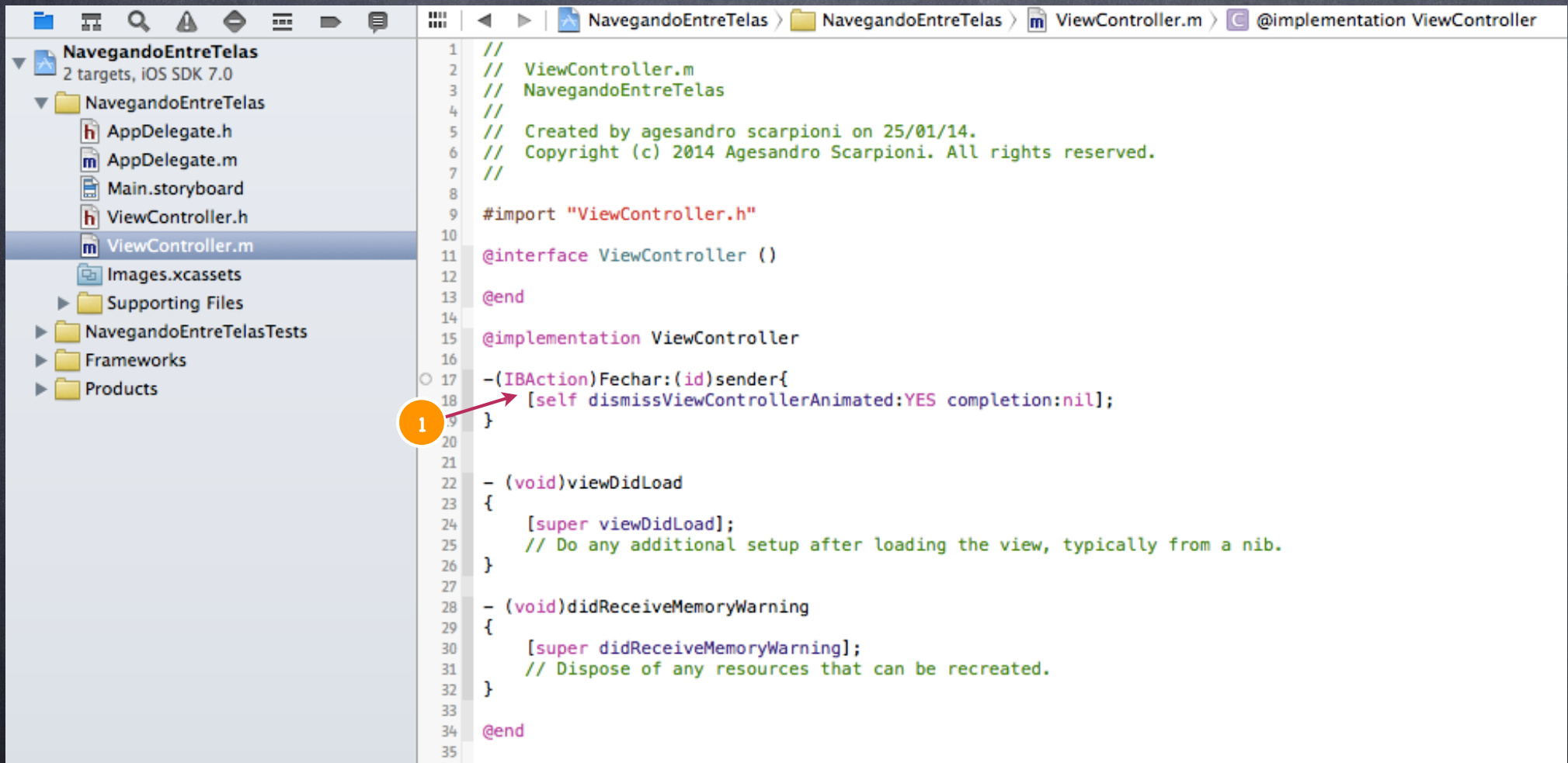
Navigation Controller

- Na classe viewController.h vamos declarar uma linha para que possamos fazer a implementação na classe viewController.m, essa implementação é necessária para que a tela modal seja fechada, se não quisermos digitar, é possível fazer de forma automática clicando com o botão direito sobre o fechar, arrastar até a área de programação, porém, para isso as duas telas (desenho e .h) devem ser exibidas simultaneamente.



Fechar em Obj-C

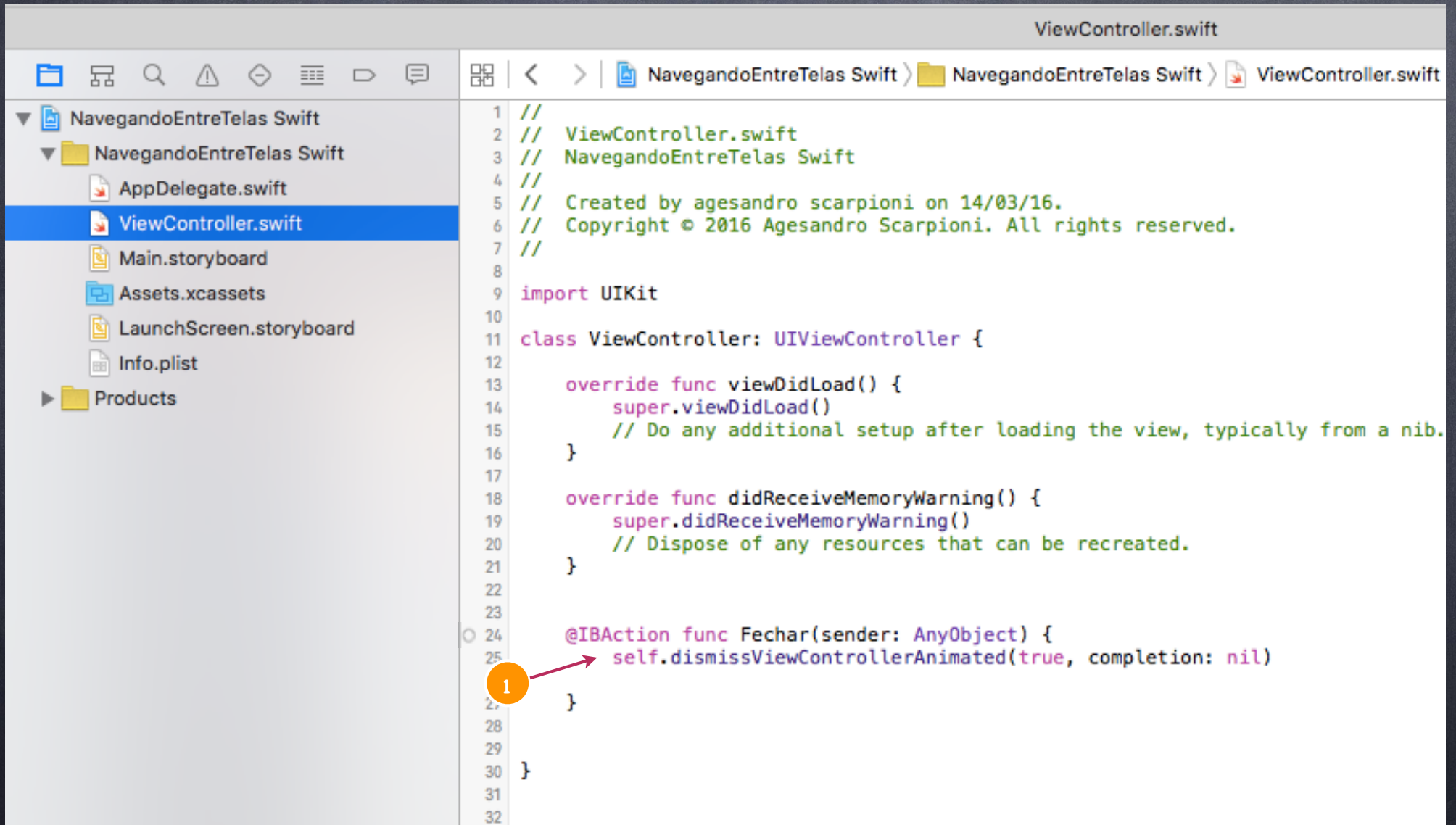
- Na classe viewController.m vamos implementar o método que fecha uma tela modal , dismissViewControlerAnimated (1).



```
1 //
2 // ViewController.m
3 // NavegandoEntreTelas
4 //
5 // Created by agesandro scarpioni on 25/01/14.
6 // Copyright (c) 2014 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import "ViewController.h"
10
11 @interface ViewController ()
12
13 @end
14
15 @implementation ViewController
16
17 -(IBAction)Fechar:(id)sender{
18     [self dismissViewControllerAnimated:YES completion:nil];
19 }
20
21
22 -(void)viewDidLoad
23 {
24     [super viewDidLoad];
25     // Do any additional setup after loading the view, typically from a nib.
26 }
27
28 -(void)didReceiveMemoryWarning
29 {
30     [super didReceiveMemoryWarning];
31     // Dispose of any resources that can be recreated.
32 }
33
34 @end
35
```

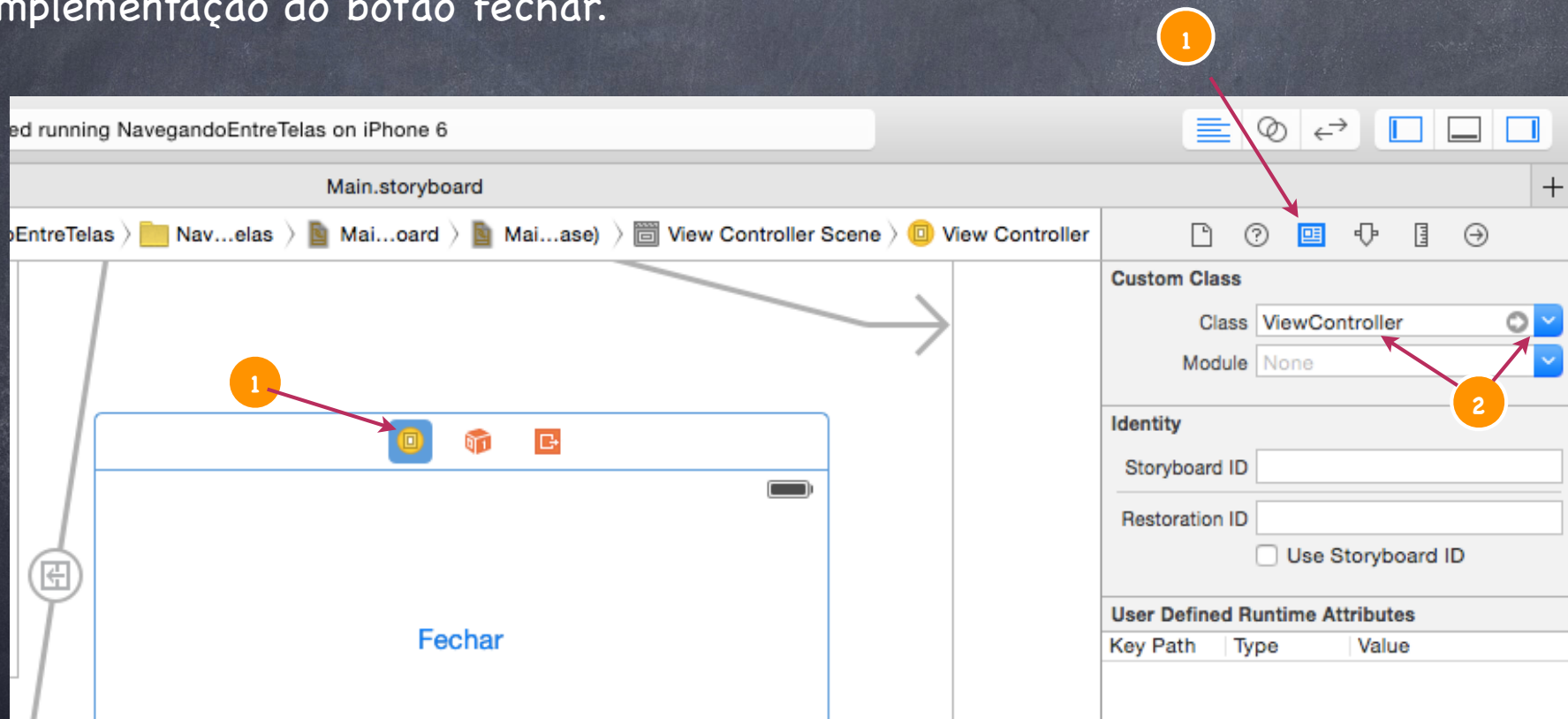

Fechar em Swift

- Na classe viewController.swift vamos implementar o método que fecha uma tela modal , dismissViewControllerAnimated (1).



Navigation Controller

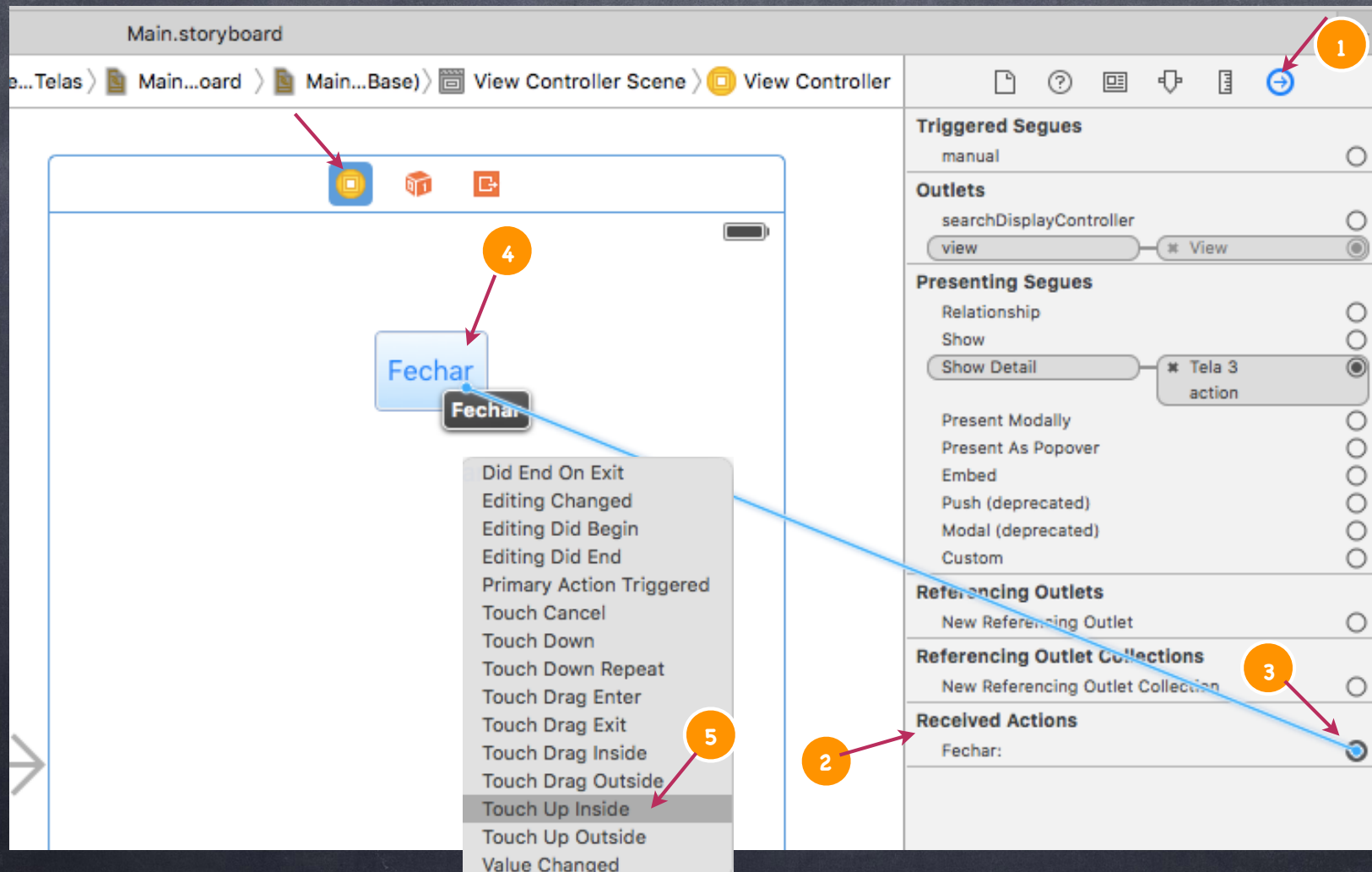
- Vamos alterar a classe “dona” da view no Identity Inspector(1) de UIViewController para ViewController(2), pois aproveitamos a classe ViewController que já havia sido criada com o projeto e nela fizemos a implementação do botão fechar.



Navigation Controller

FIAP

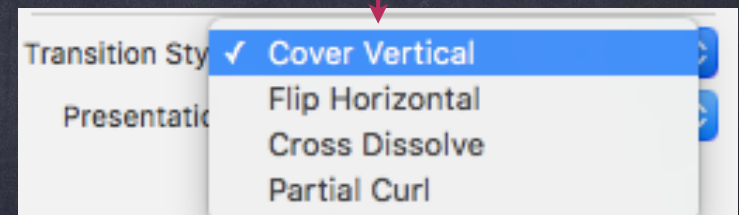
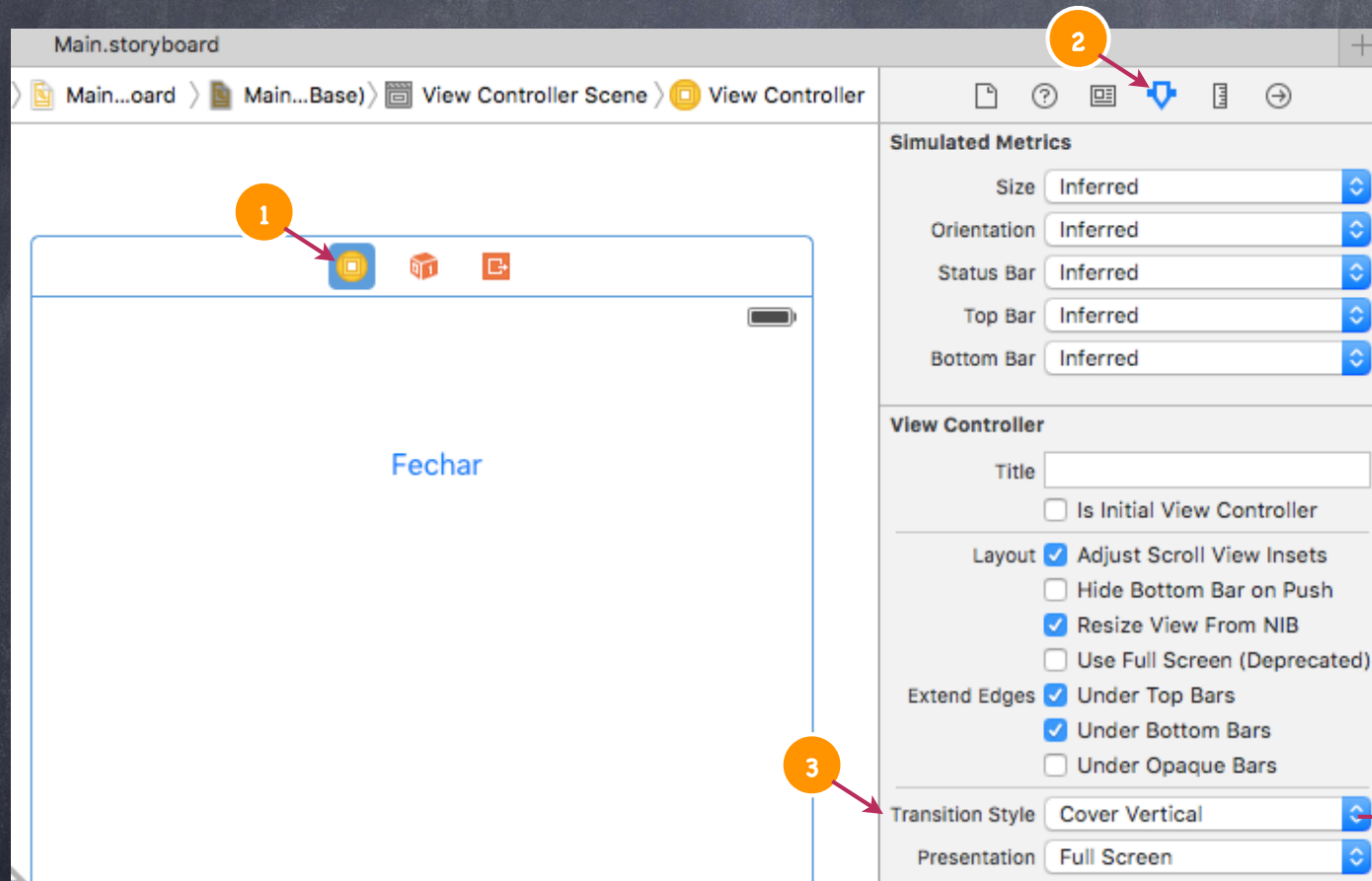
- Veja que no Connections Inspector (1), depois de trocarmos o nome da classe para ViewController, aparece o item Received Actions (2), clique na bolinha que fica depois da palavra fechar (3) e arraste até o botão (4) selecione Touch Up Inside(5).



Atenção: O método "Fechar" só apareceu porque ligamos a classe viewController como dona da View onde está o botão "Fechar".

Navigation Controller

- É possível trocar a forma de transição entre View's (1,2,3)



Navigation Controller

FIA/P

- Esse exemplo é uma transição do tipo Partial Curl



Navigation Controller

- Execute sua aplicação (command + R) e observe a diferença de navegação entre as telas show e show detail.
- Repare que os botões de Voltar nas Scenes que segue com show estão como Back, podemos alterar isto no próximo conjunto de slides.
- Ao colocarmos um título no primeiro ViewController (Scene) o botão Back passa a ter a mesma descrição da tela principal.