

Navigation Controller Controller parte 2

Utilizando StoryBoard

X-Code - ObjC

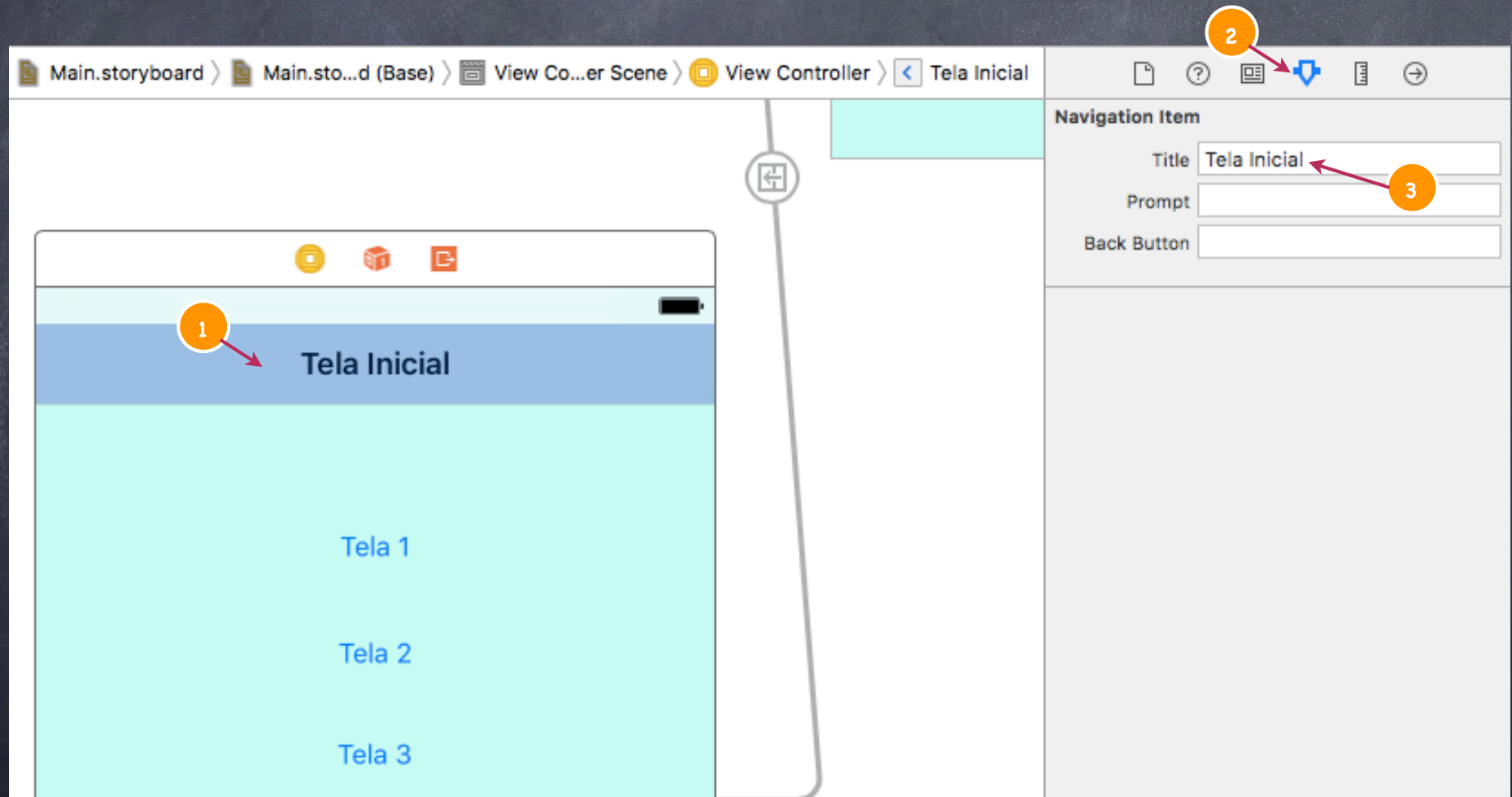
Prof. Agesandro Scarpioni

agesandro@fiap.com.br

Navigation Controller

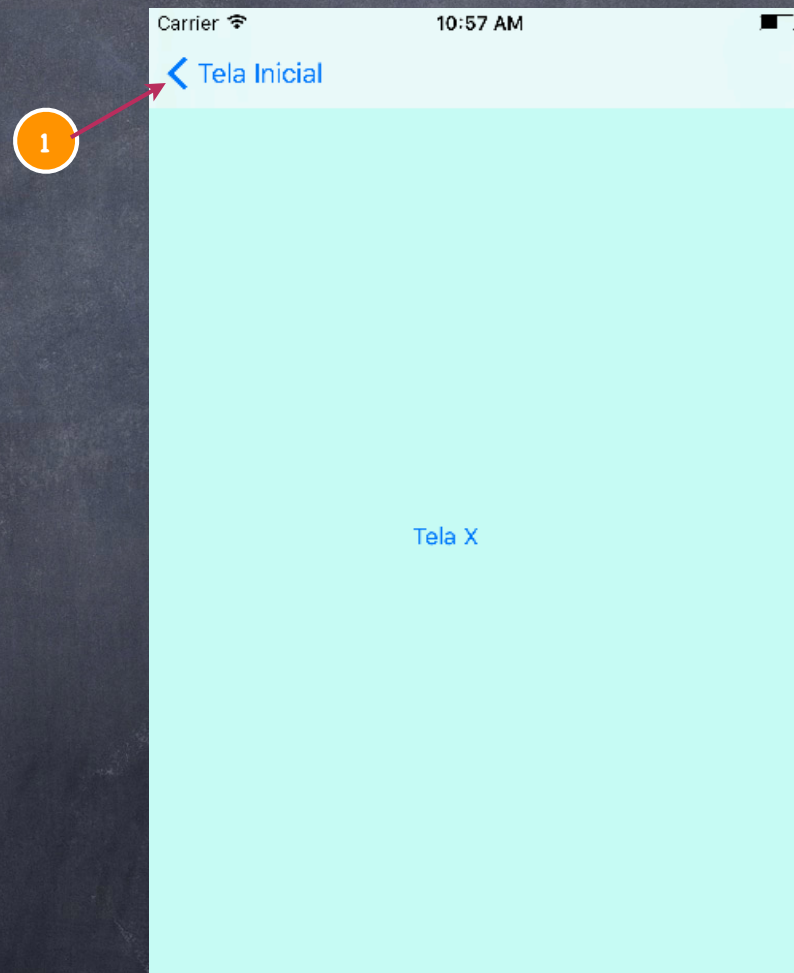
Títulos para as Scenes

- Na primeira tela coloque um título como por exemplo: "Tela Inicial", para isso clique diretamente no título (1) ou em Attributes Inspector (2) altere a propriedade Title (3).



Navigation Controller

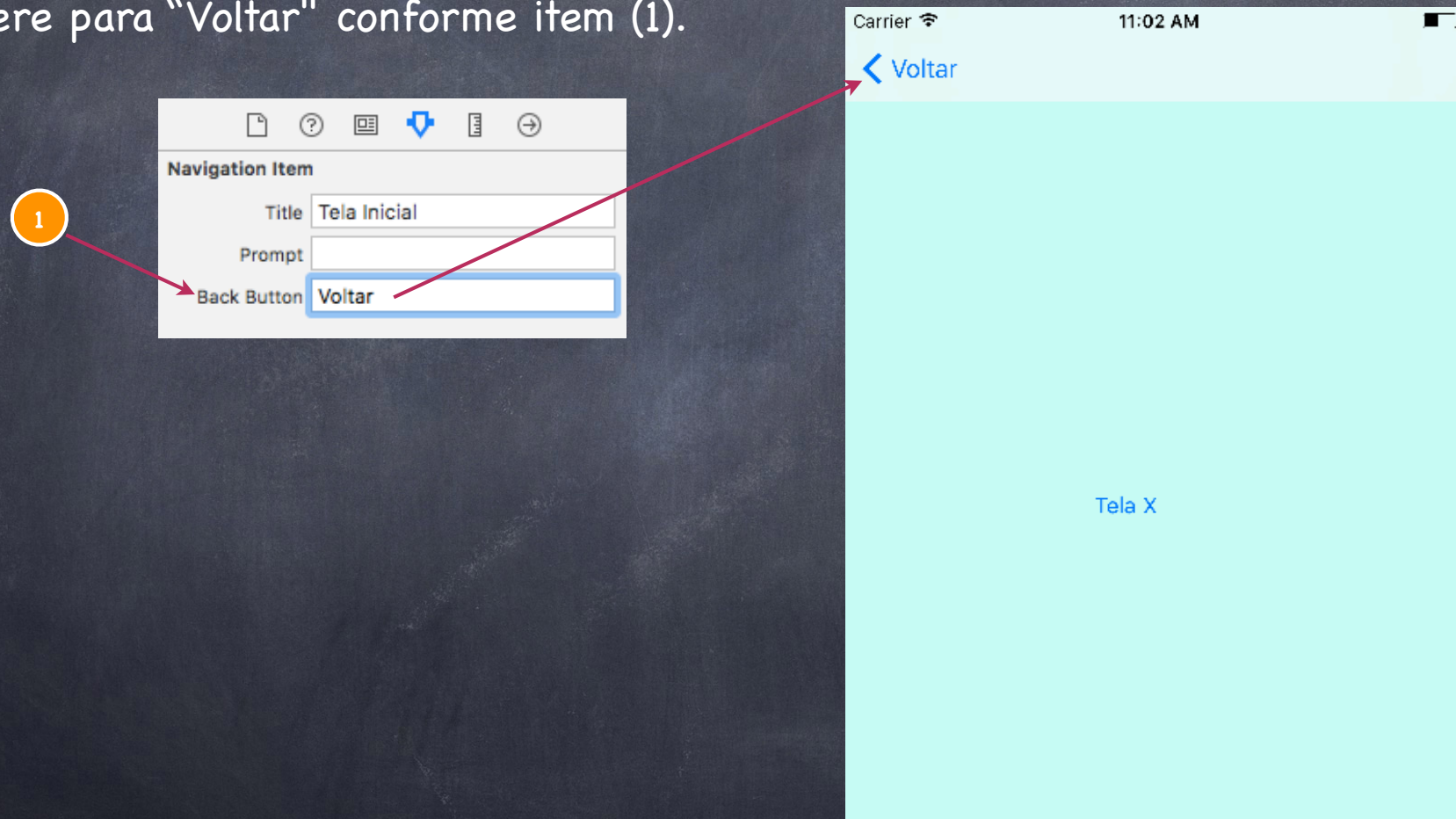
- Ao executar o app e clicar no botão "Tela 1" a Scene será exibida e o botão Back passa a ter a descrição "Tela inicial".



Navigation Controller

Botão Voltar das Scenes

- Para alterar o botão de retorno de Back que havia assumido o título da Scene principal, para a palavra "Voltar", vá na propriedade Back Button da Scene principal e altere para "Voltar" conforme item (1).



OBS: Todas as primeiras Scenes após o Root podem ter um título, basta clicar no topo da scene e preencher ou em Attributes Inspector propriedade Title, para títulos outras Scenes é preciso incluir um objeto "Navigation Item" como será mostrado no slides 6.

Navigation Controller

FIAP

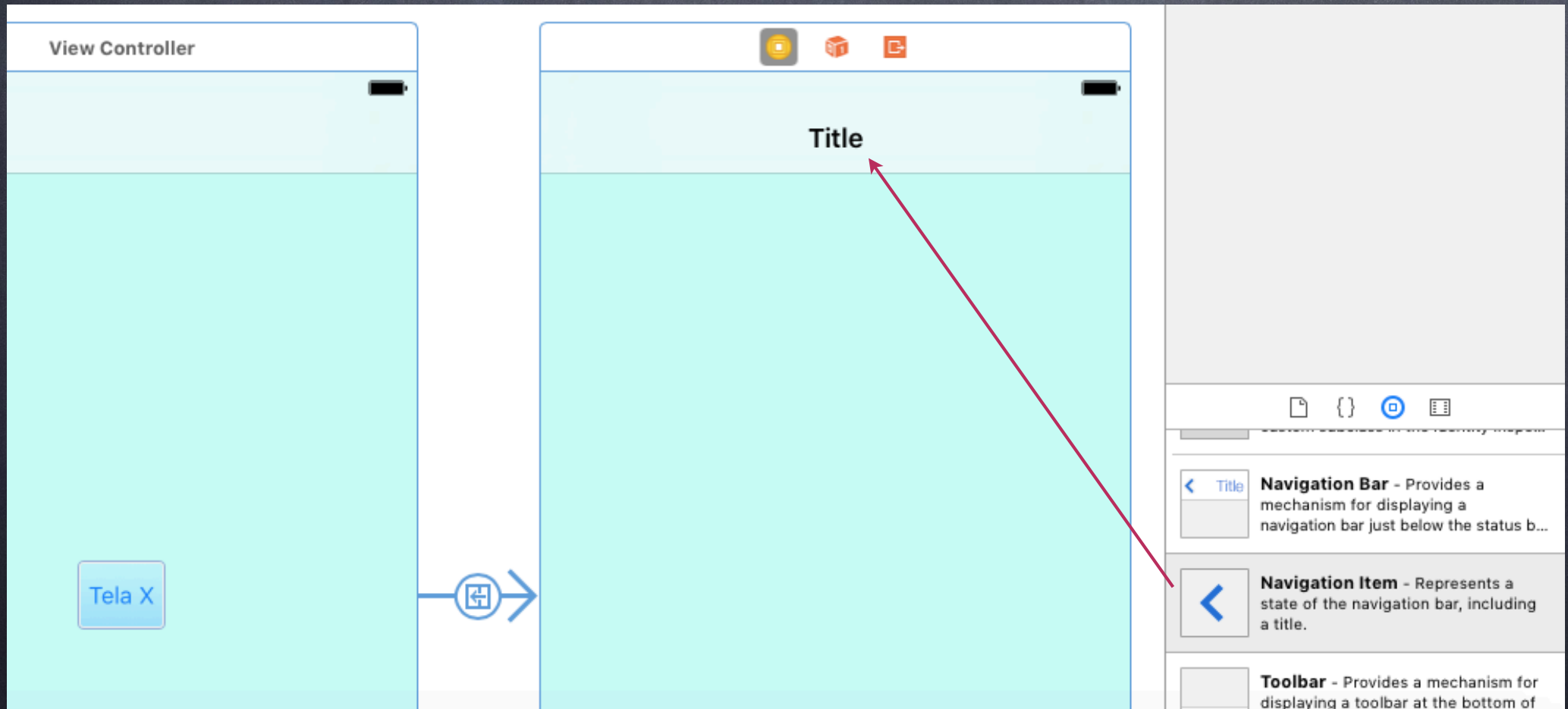
Segue via código

- Vamos navegar entre telas via código ao invés de criar nossa segue com arrastar e soltar do mouse, isto será útil quando precisarmos navegar para uma outra scene como por exemplo após um clique em uma célula de um TableView, ou um botão com ações condicionais.

Navigation Controller

Segue via código

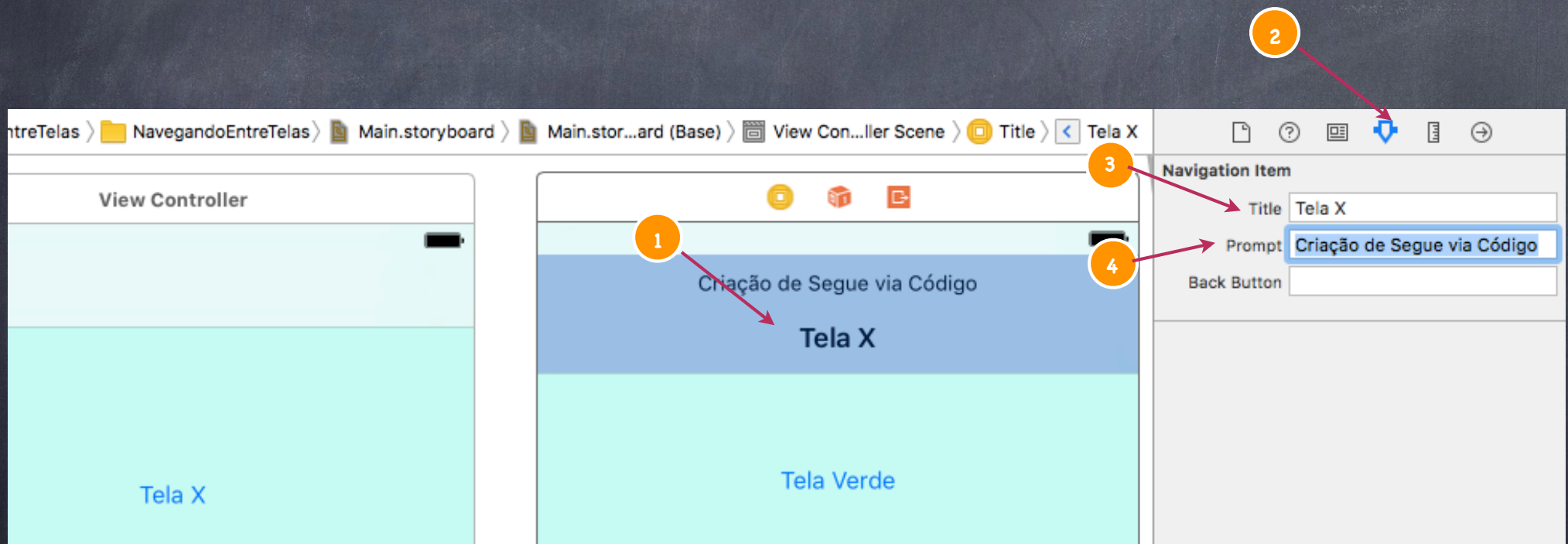
- Utilize a View Controller ao lado da tela com o botão "Tela X", e faça a ligação com Show e coloque um Navigation Item nessa tela.



Navigation Controller

Segue via código

- Na tela X com o Navigation Item selecionado (1), vá ao Attributes Inspector(2) e Altere o Title (3) para: "Tela X" e o Prompt (4) para "Criação das Segue via código".

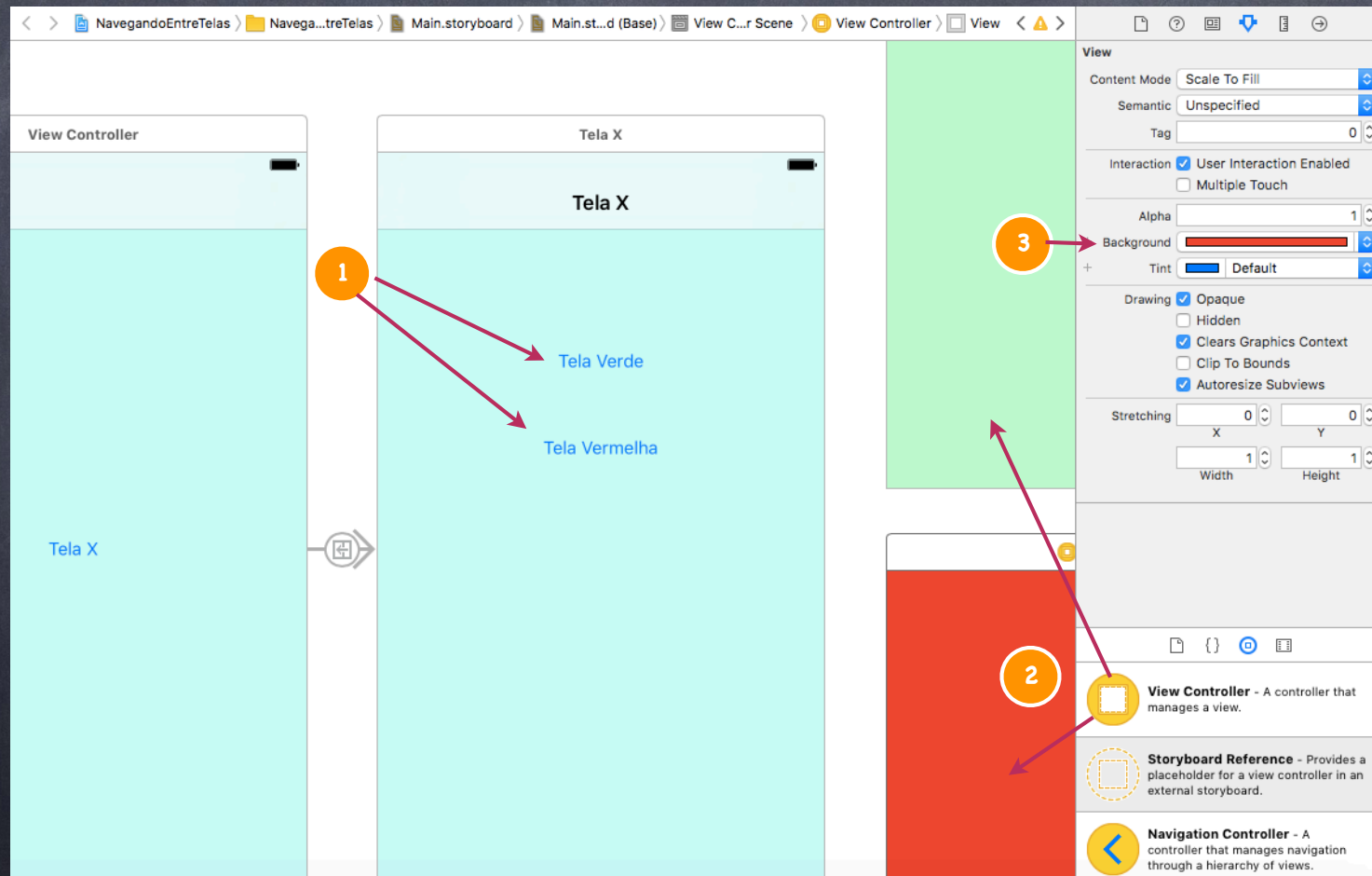


Dica: Segue é o nome da transição de uma Scene para outra.

Navigation Controller

Segue via código

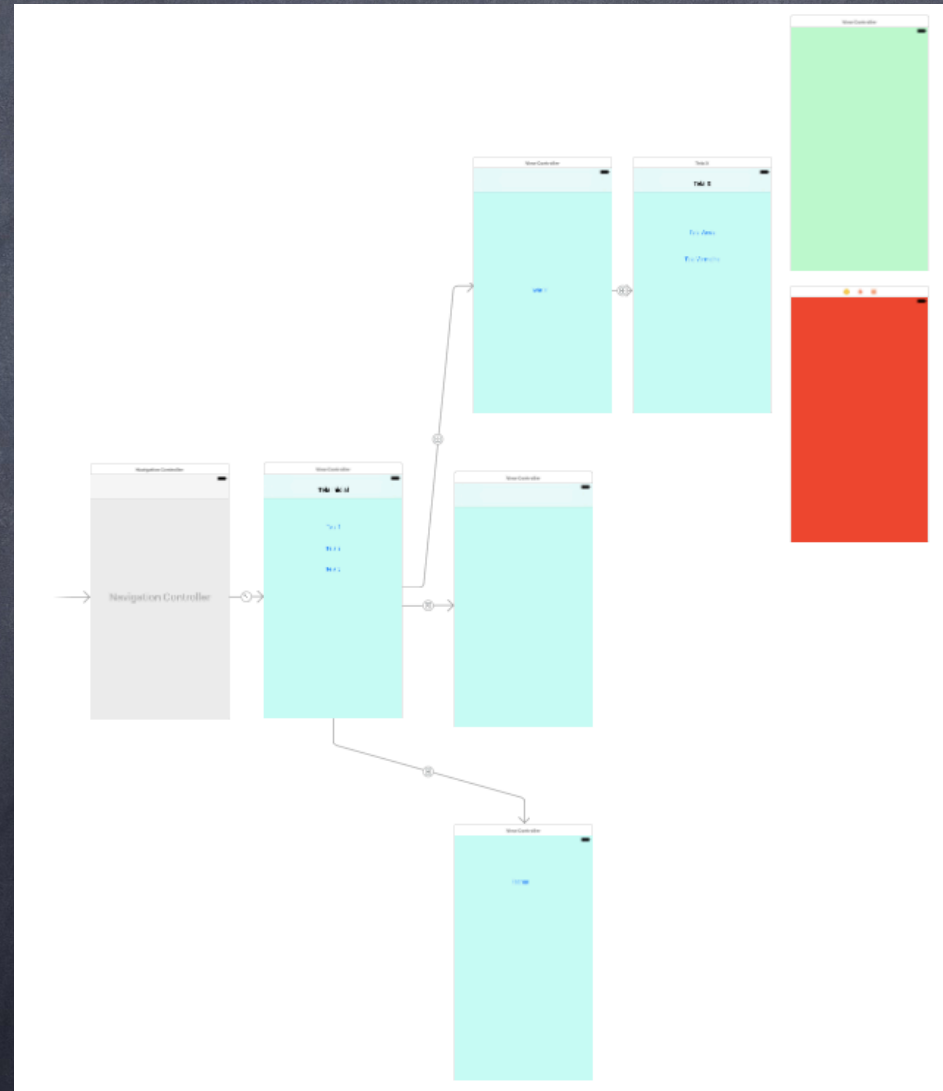
- Inclua na Tela X outros dois botões "Tela Verde" e "Tela Vermelha" (1), juntamente com outras duas ViewController's (2), altere a propriedade BackColor (3) das duas Views para diferenciar as telas, pois já existem muitas View's no storyboard.



FIAP

Segue via código

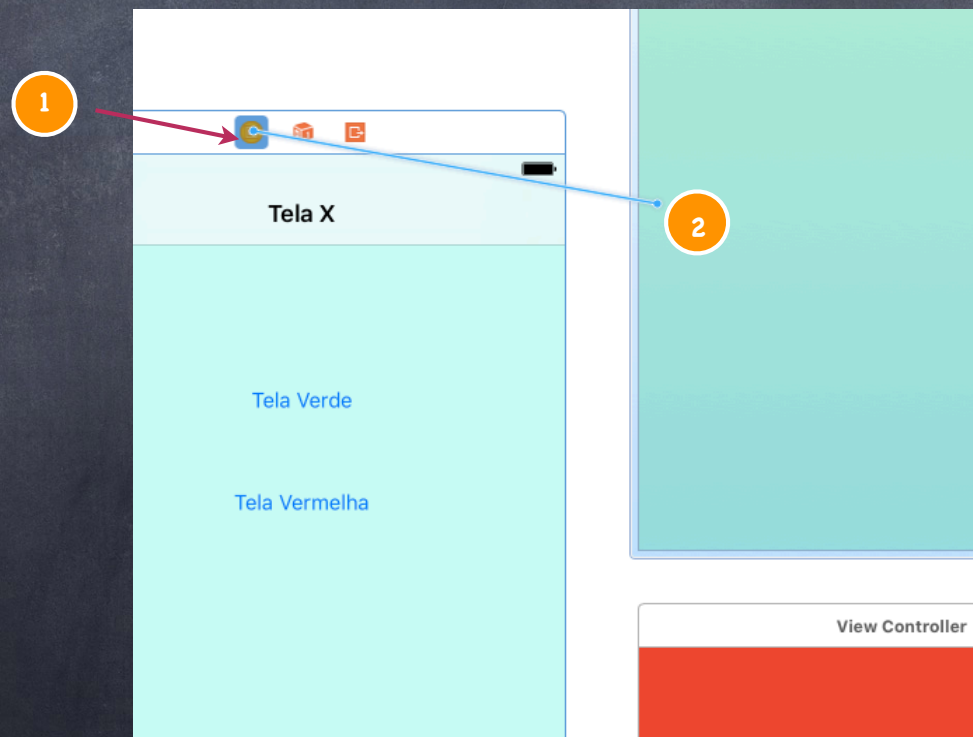
- Visão geral de todas as Scenes.



Navigation Controller

Segue via código

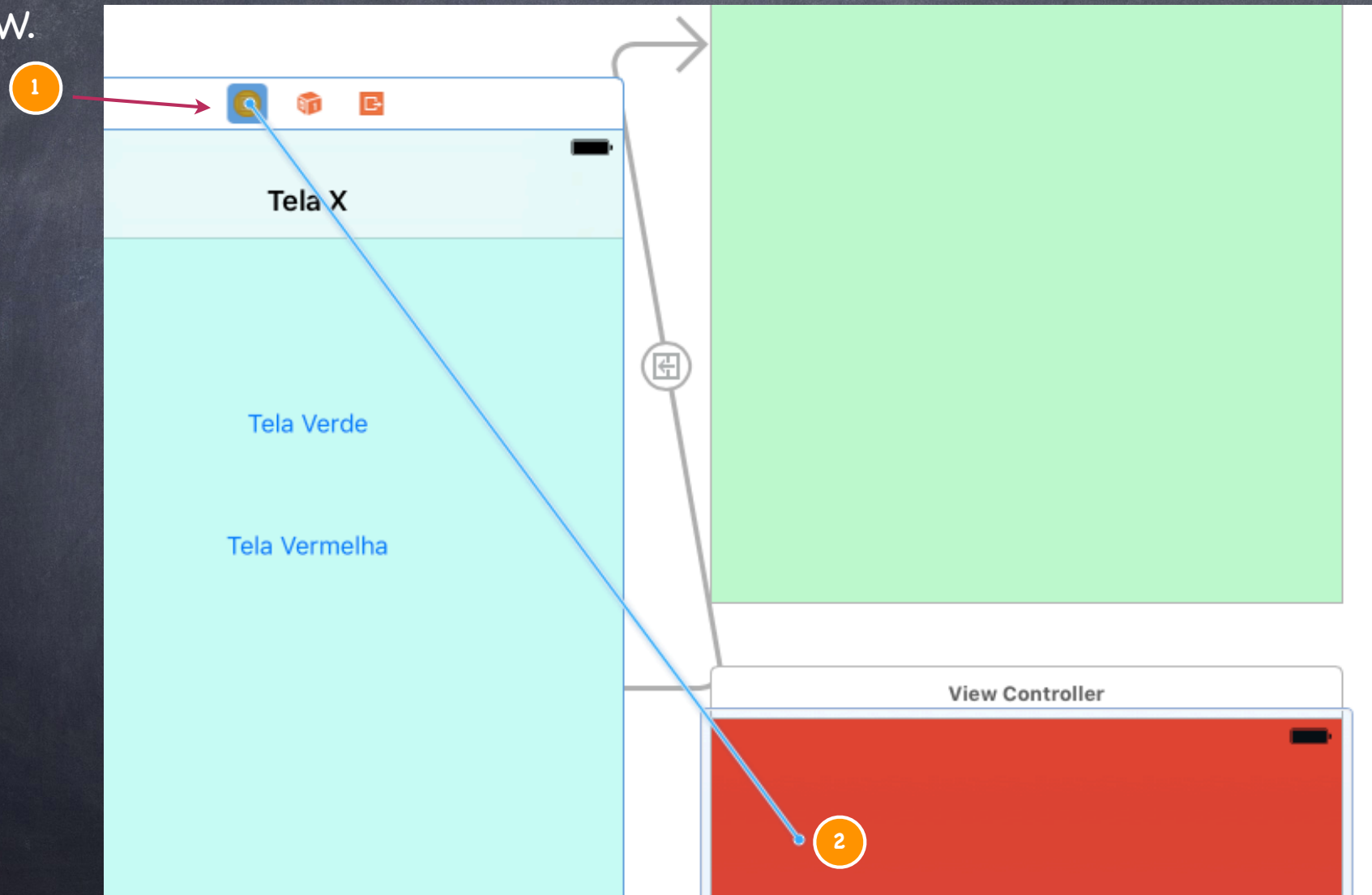
- Neste ponto duas coisas precisam ser feitas, uma é que a segue deve ser criada sem estar associada a um botão e sim de "Controller" para "Controller", e a segunda é que a Segue deve estar associada a um identificador único.
- Primeiro passo, para ligar um controlador a outro, siga os passos da imagem arrastando a linha até a tela verde com o control pressionado e escolha → SHOW.



Navigation Controller

Segue via código

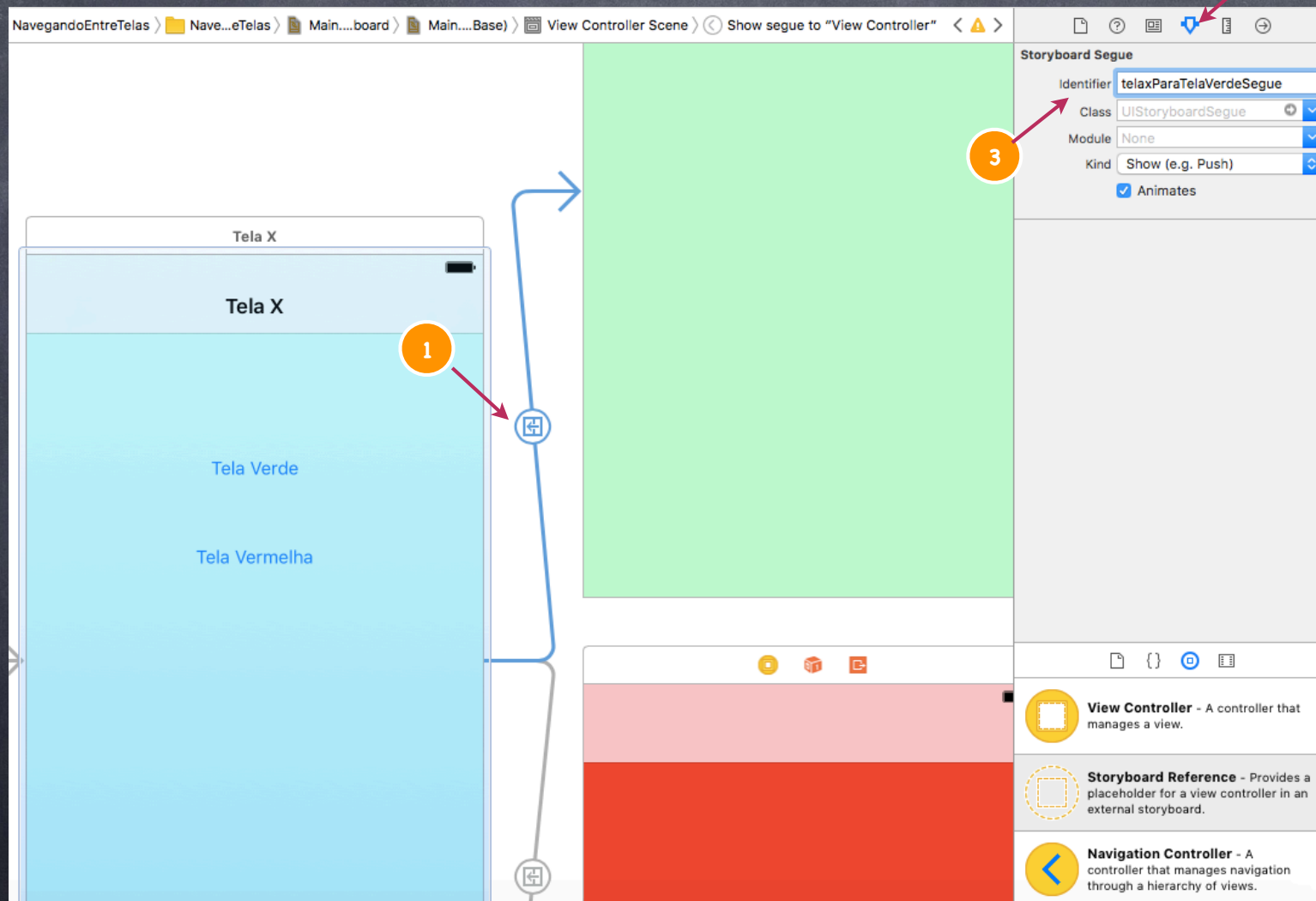
- Repita o processo ligando o "Controller" ao "Controller" da tela vermelha e escolha SHOW.



Dica: Segue é o nome da transição de uma Scene para outra.

Navigation Controller

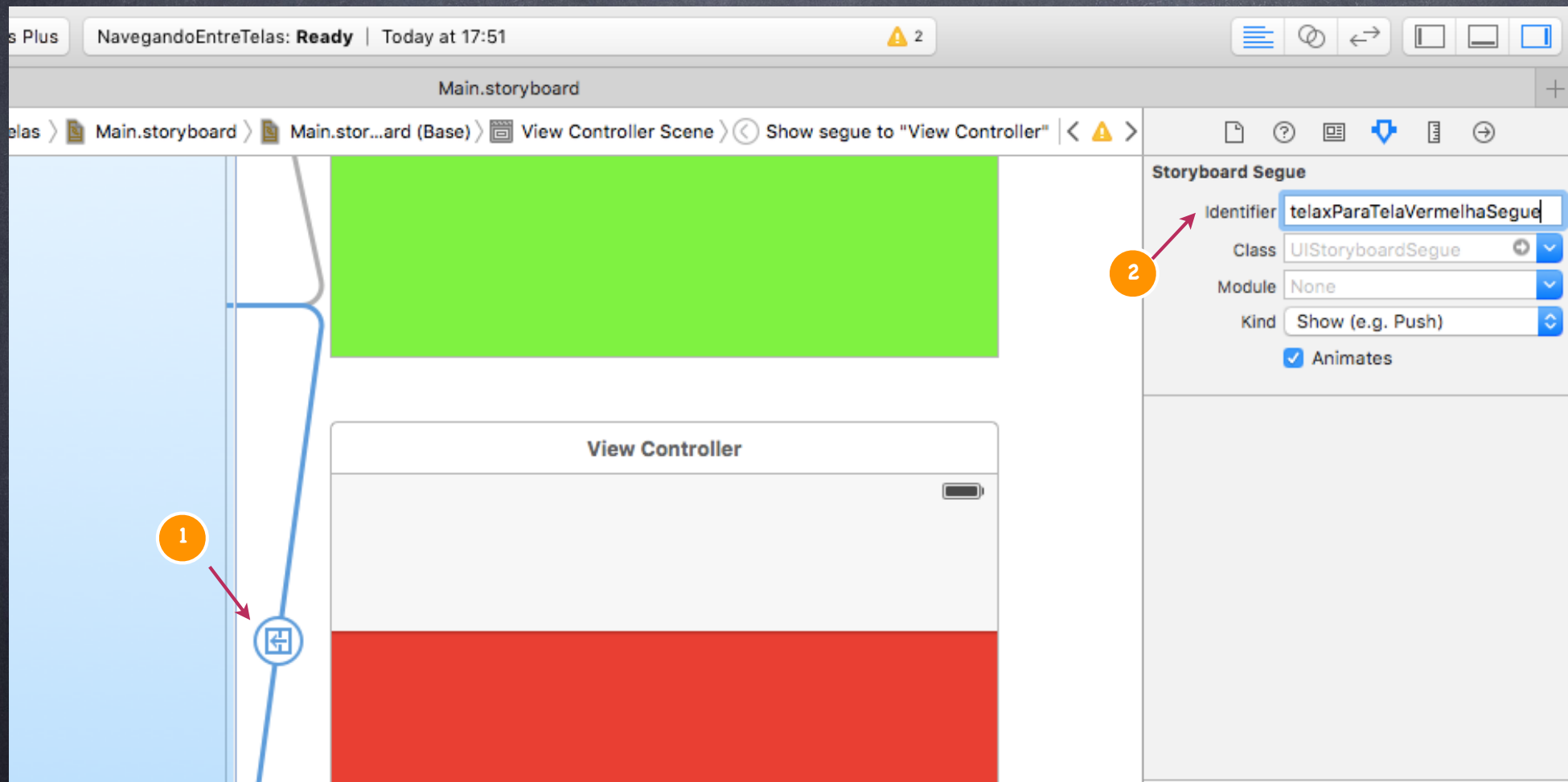
- Segundo passo, identificar a segue com nome único, para isso clique na Segue(1) entre a TelaX e a Controller verde, vá em Attributes Inspector(2), e altere o campo identifier(3) para "telaxParaTelaVerdeSegue".



Navigation Controller

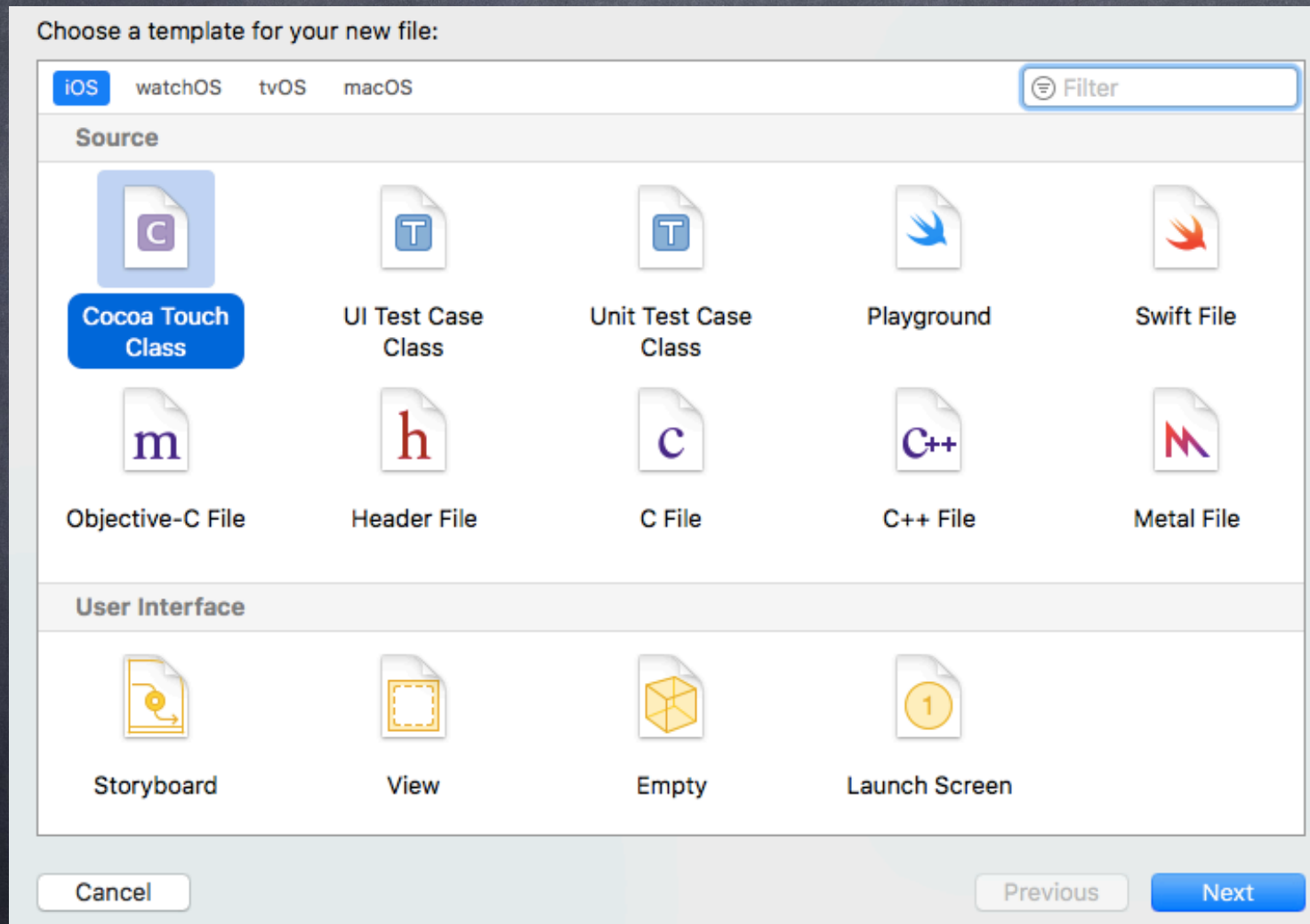
FIAP

- Repita o passo para a Scene vermelha e altere o campo identifier para "telaxParaTelaVermelhaSegue".



Segue via código

- Agora é preciso codificar, clique em File → New → File → iOS → Cocoa Touch Class → Next e vamos dar o nome da classe de TelaxViewController.



Segue via código

- TelaxViewController será o nome da Classe, subclasse de UIViewController e deixe desmarcado o check box "Also Create Xib...", já temos uma interface para esta classe.

Choose options for your new file:

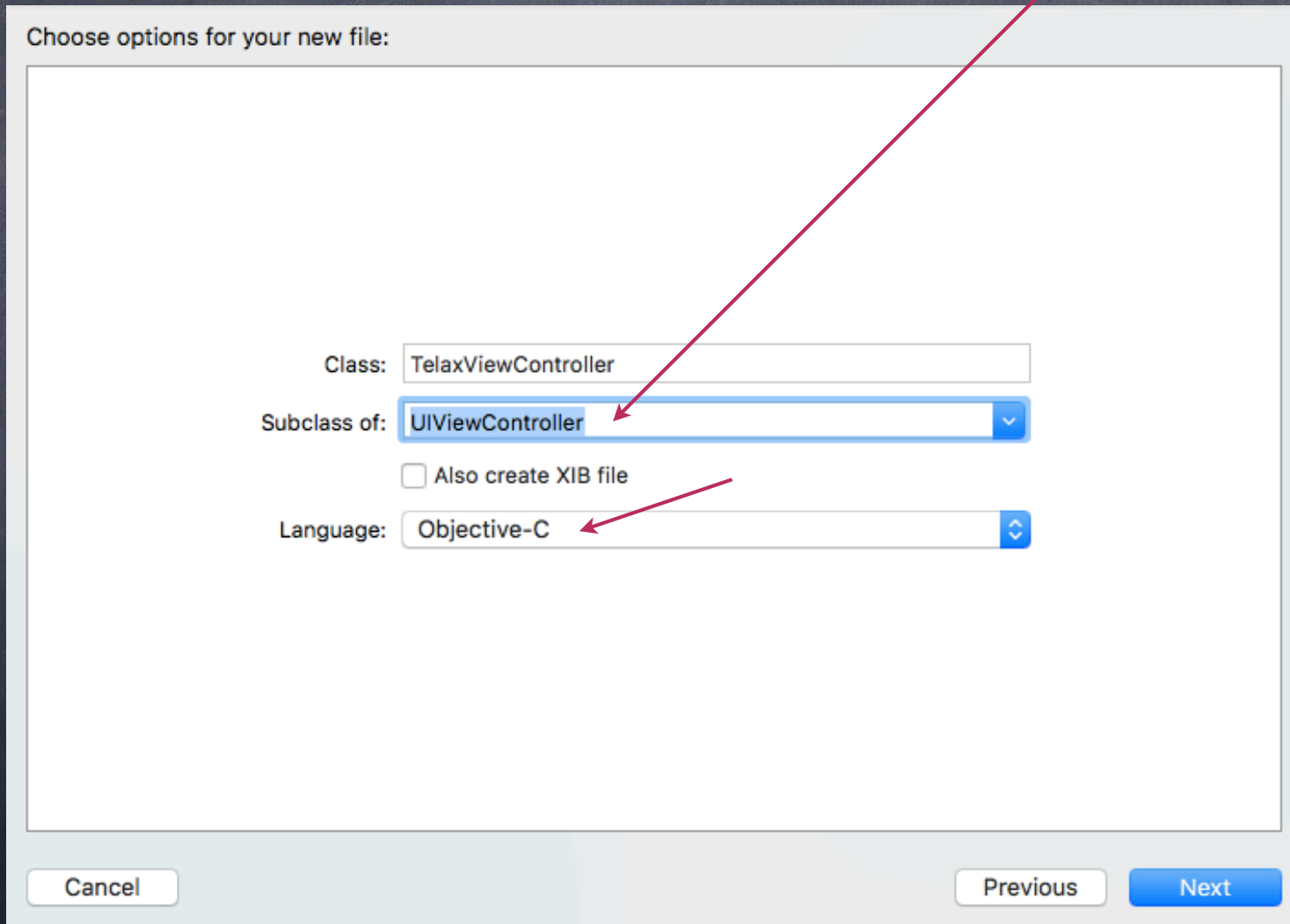
Class:

Subclass of:

☐ Also create XIB file

Language:

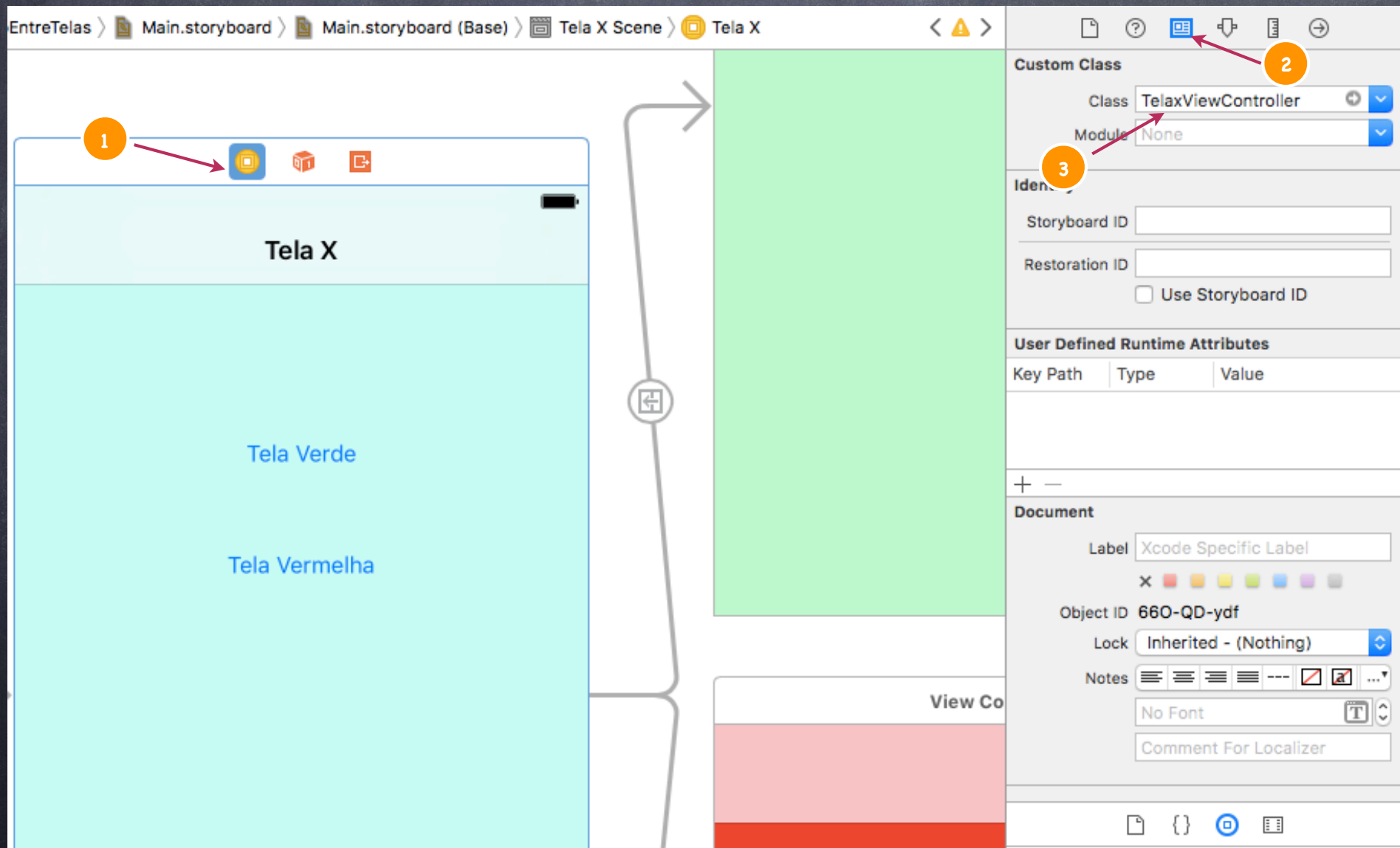
Cancel Previous Next



Segue via Código

FIAP

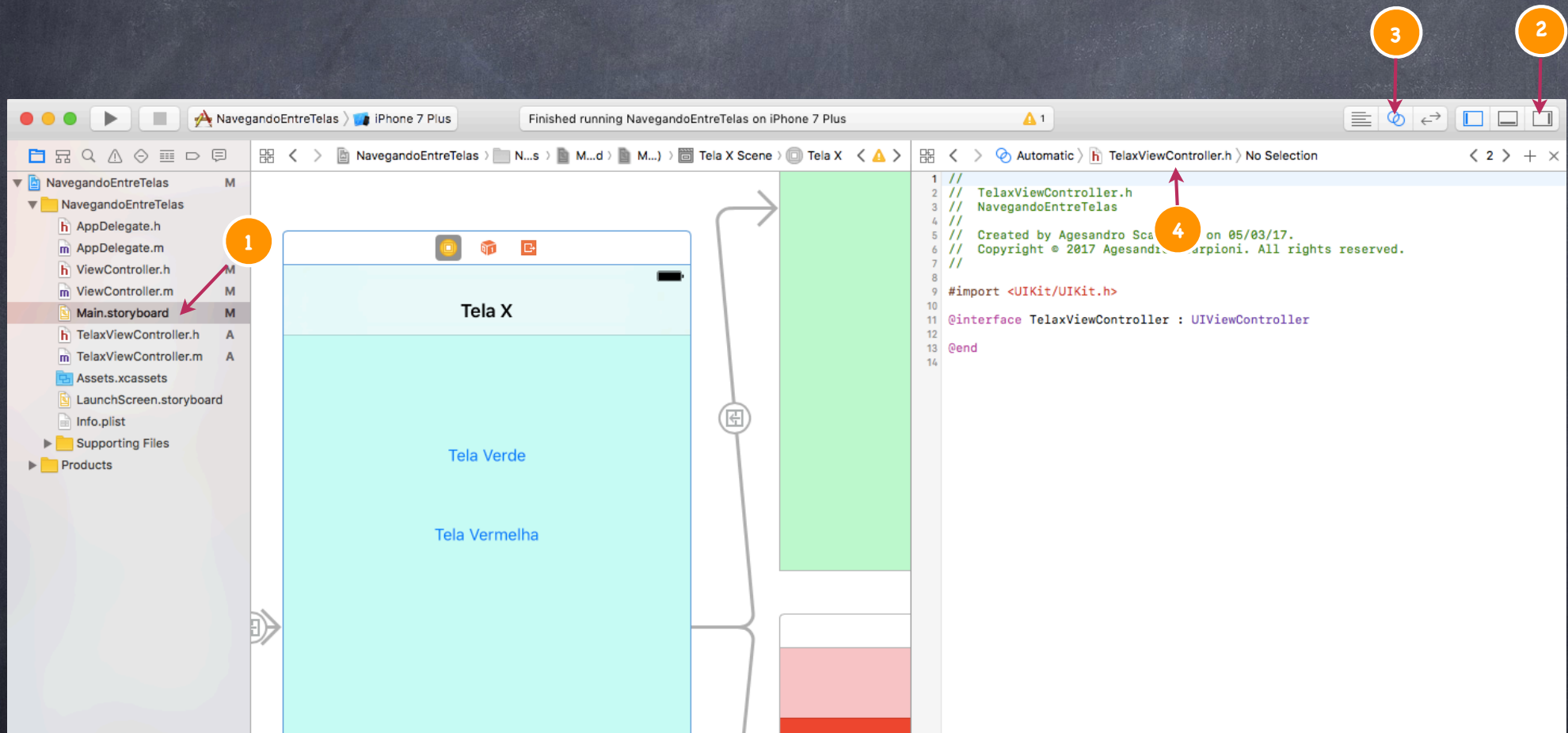
- No StoryBoard selecione o Controller da Tela X (1), abra o Identity Inspector (2) e vamos dar um nome para a classe "dona" dessa View, digite TelaxViewController.(3)



Segue via Código

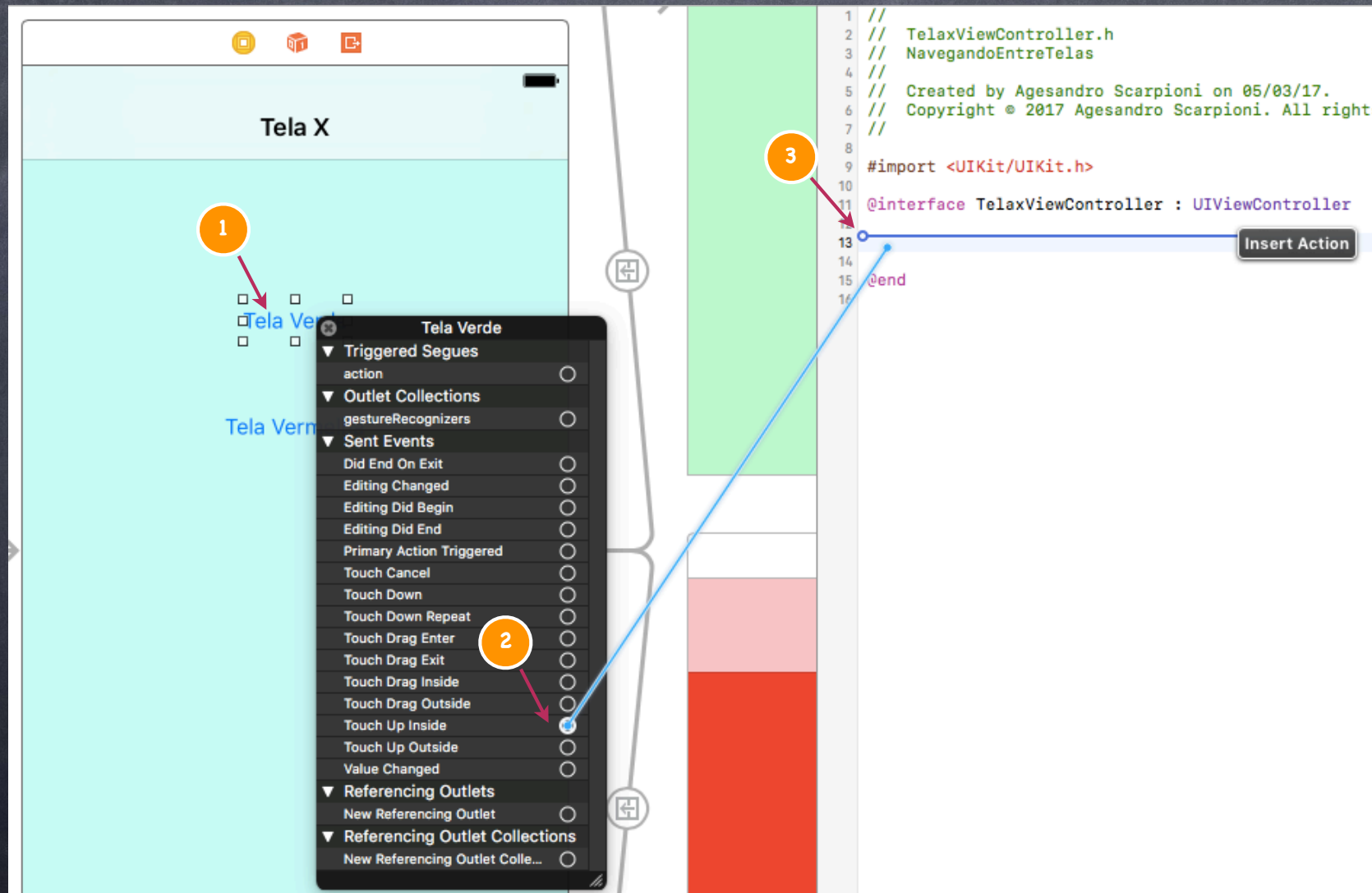
FIAP

- Prepare a tela para que seja criado o código, clique nos locais indicados para que seja aberta simultaneamente a tela de Storyboard e a classe TelaxViewController.h.



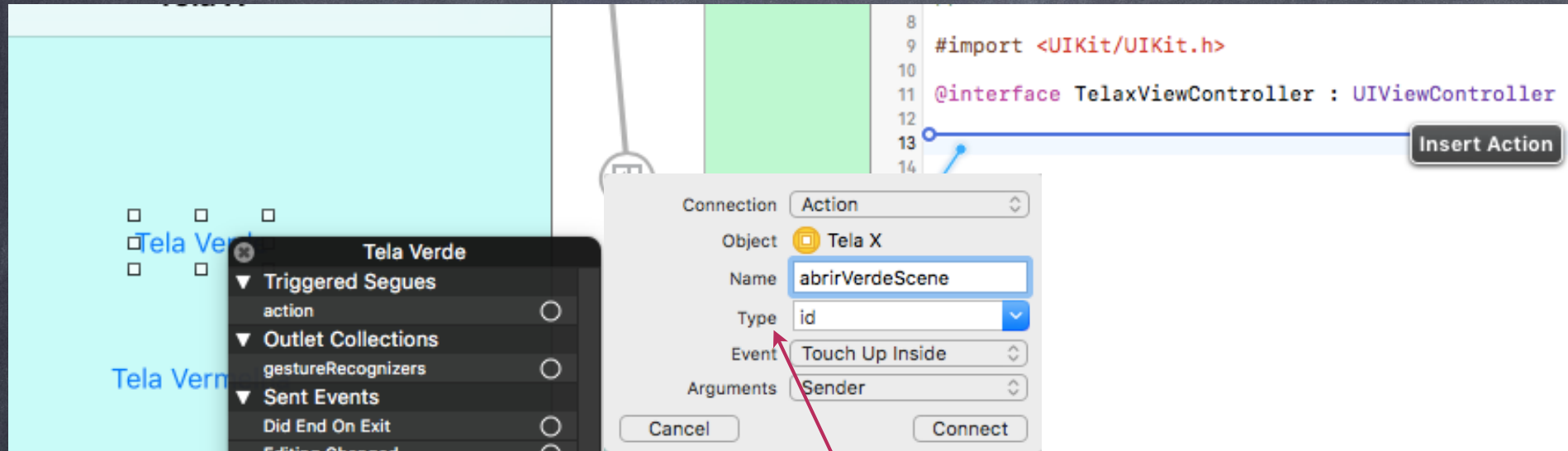
Segue via Código

- Agora que as duas telas estão abertas simultaneamente, clique com o botão direito sobre o botão Tela Verde (1), escolha o evento Touch Up Inside (2) e arraste para a tela .h no ponto indicado (3), ao aparecer o popup nomeie o Action como abrirVerdeScene.



Segue via Código

FIAP



- Repita o passo para o botão Tela Vermelha, nomeie o Action para abrirVermelhaScene.

OBS: Em versões anteriores de Xcode o Type ao invés de aparecer id irá aparecer AnyObject.

Segue via Código

FIAP

- Vamos abrir a classe TelaxViewController.m e chamar o método performSegueWithIdentifier para cada Action, esse método serve para navegar de uma Scene para outra. Após digitar estas linhas teste seu programa, Command + R.

```
1 //
2 // TelaxViewController.m
3 // NavegandoEntreTelas
4 //
5 // Created by agesandro scarpioni on 14/02/16.
6 // Copyright © 2016 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import "TelaxViewController.h"
10
11 @interface TelaxViewController ()
12
13 @end
14
15 @implementation TelaxViewController
16
17 - (void)viewDidLoad {
18     [super viewDidLoad];
19     // Do any additional setup after loading the view.
20 }
21
22 - (void)didReceiveMemoryWarning {
23     [super didReceiveMemoryWarning];
24     // Dispose of any resources that can be recreated.
25 }
26
27 /*
28 #pragma mark - Navigation
29
30 // In a storyboard-based application, you will often want to do a little preparation before navigation
31 - (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
32     // Get the new view controller using [segue destinationViewController].
33     // Pass the selected object to the new view controller.
34 }
35 */
36
37 - (IBAction)abrirVerdeScene:(id)sender {
38     [self performSegueWithIdentifier:@"telaxParaTelaVerdeSegue" sender:sender];
39 }
40
41 - (IBAction)abrirVermelhaScene:(id)sender {
42     [self performSegueWithIdentifier:@"telaxParaTelaVermelhaSegue" sender:sender];
43 }
44
45 @end
```

```
36
37 - (IBAction)abrirVerdeScene:(id)sender {
38     [self performSegueWithIdentifier:@"telaxParaTelaVerdeSegue" sender:sender];
39 }
40
41 - (IBAction)abrirVermelhaScene:(id)sender {
42     [self performSegueWithIdentifier:@"telaxParaTelaVermelhaSegue" sender:sender];
43 }
44 }
```

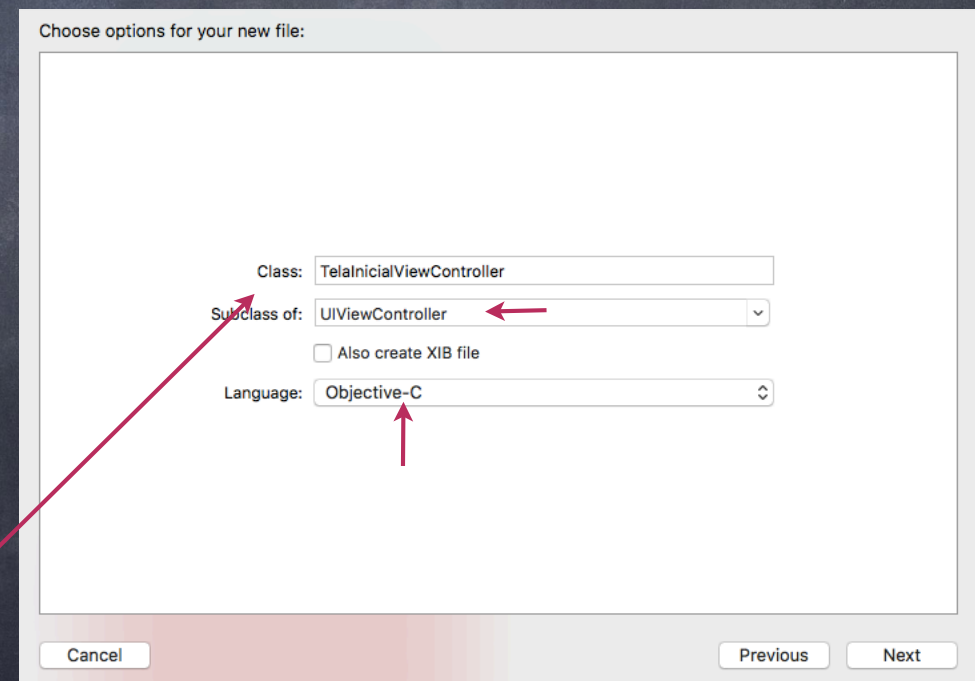
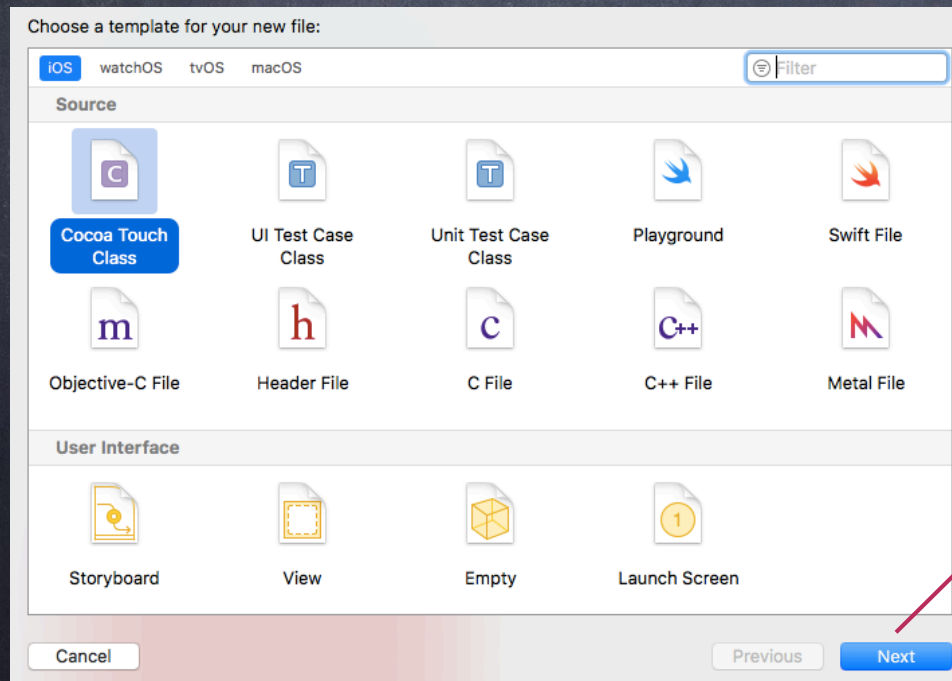

Navigation Controller

Segue via código

- Execute → Command + R e verifique que agora os botões Tela Verde e Tela Vermelha abrem telas por meio de codificação. Se algo não funcionar é bem provável que tenha errado o nome que foi dado no atributo identifier da Scene.

Navegar diretamente para tela inicial

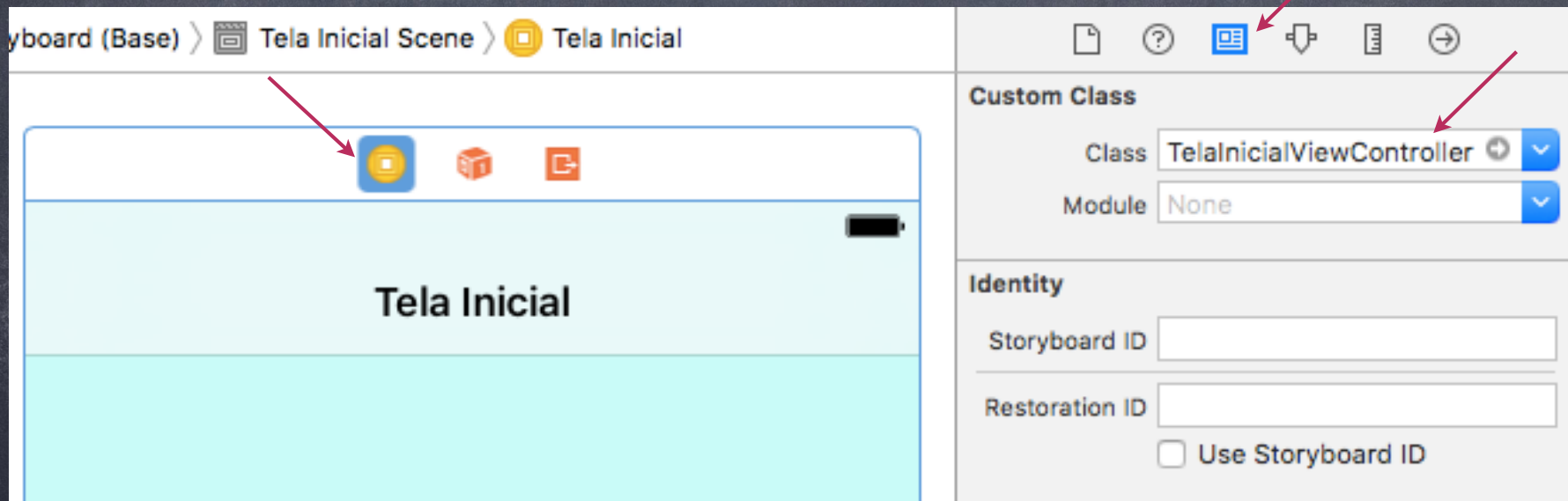
- Navegar diretamente para a Scene Inicial, na documentação o termo que faz essa "saída" é "Unwind Segue", primeiro precisamos criar um método com um nome qualquer na classe da Scene que você deseja retornar, veja o tal método no slide 24.
- Clique em File → New → File → iOS → Cocoa Touch Class → Next, daremos o nome da classe de TelaInicialViewController, esta deve ser **subclasse de UIViewController**, não marque o checkbox, não esqueça de definir a tela inicial como "dona" dessa classe, para isso veja o próximo slide.



Navigation Controller

Navegar diretamente para tela inicial

- Definindo a Scene da Tela Inicial como "dona" da classe TelaInicialViewController, digite TelaInicialViewController em Custom Class.



Navegar diretamente para tela inicial

- Crie apenas no TelaInicialViewController.m o método abaixo, não é necessário implementar. Observe que o método é um IBAction.

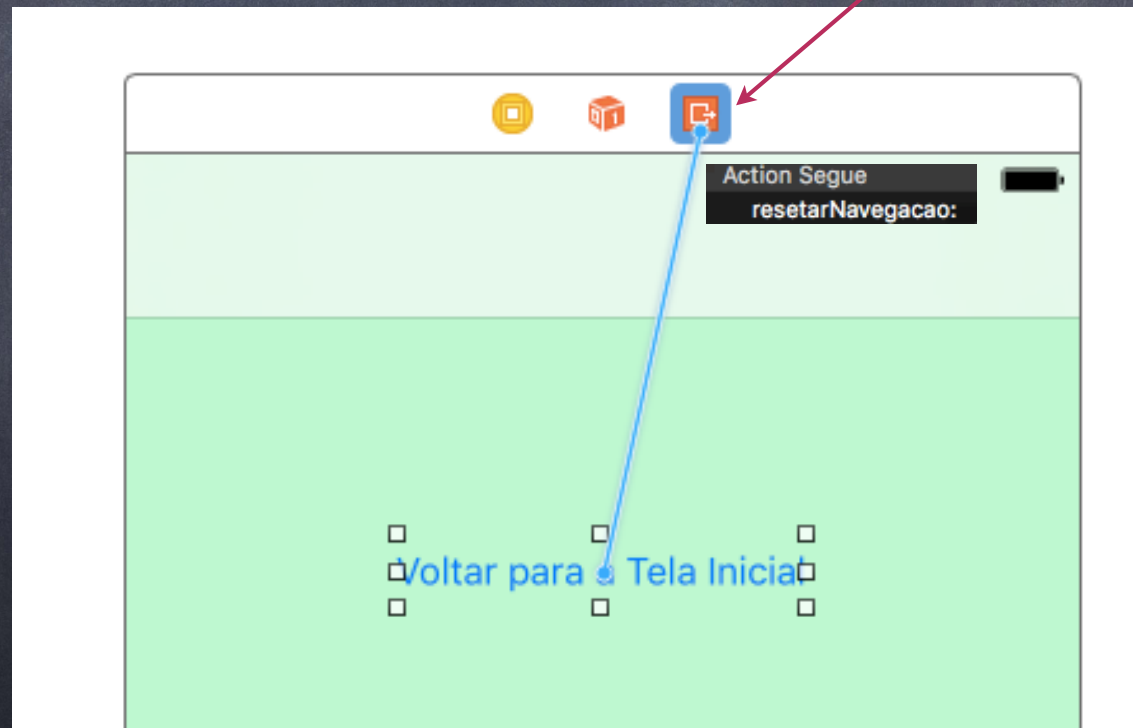
```
1 //
2 // TelaInicialViewController.m
3 // NavegandoEntreTelas
4 //
5 // Created by agesandro scarpioni on 02/03/15.
6 // Copyright (c) 2015 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import "TelaInicialViewController.h"
10
11 @interface TelaInicialViewController ()
12
13 @end
14
15 @implementation TelaInicialViewController
16
17 - (void)viewDidLoad {
18     [super viewDidLoad];
19     // Do any additional setup after loading the v
20 }
21
22 - (void)didReceiveMemoryWarning {
23     [super didReceiveMemoryWarning];
24     // Dispose of any resources that can be recrea
25 }
26
27 -(IBAction) resetarNavegacao:(UIStoryboardSegue *) segue {
28
29 }
30
31 #pragma mark - Navigation
32 /*
33 // In a storyboard-based application, you will often want to do a little preparation before navigation
34 - (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
35     // Get the new view controller using [segue destinationViewController].
36     // Pass the selected object to the new view controller.
37 }
38 */
39
40 @end
```

38 -(IBAction) resetarNavegacao:(UIStoryboardSegue *) segue {
39 // Esse método não precisa de código.
40 // Esse método recebe um argumento do tipo UIStoryboardSegue
41 // após isso devemos fazer um link entre o botão de origem
42 // e a ação Unwind (documentação), ou Exit (Xcode)
43 }

Navigation Controller

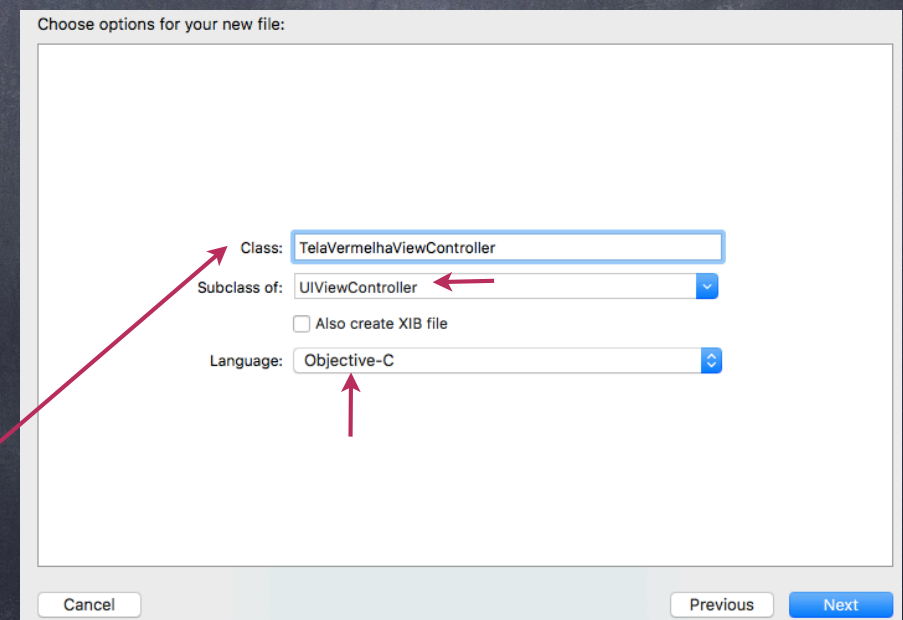
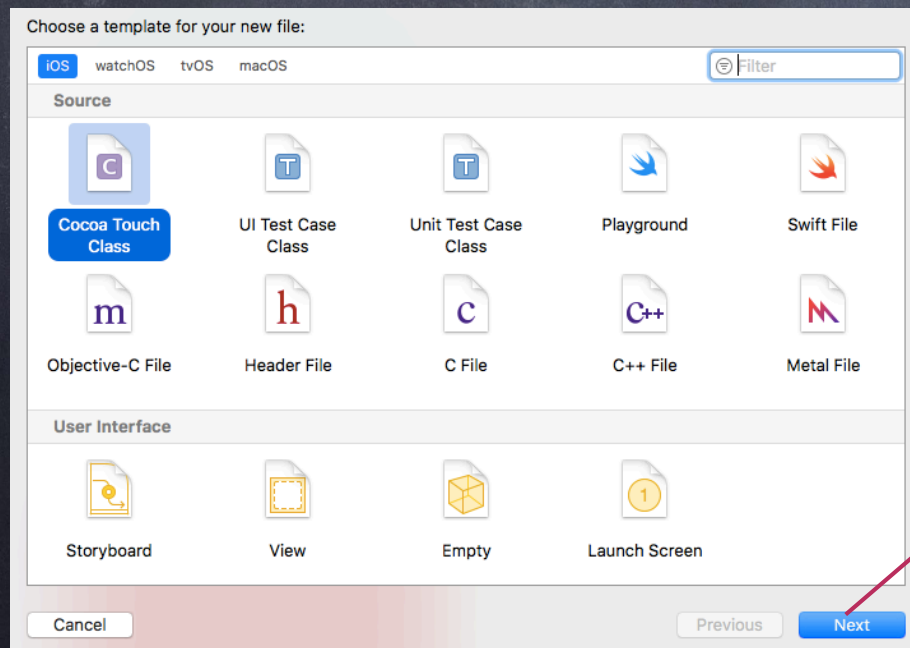
Navegar diretamente para tela inicial

- Coloque um botão na tela verde e escreva o texto conforme a imagem abaixo, em seguida selecione o botão, segure o CTRL e arraste para o botão "Exit" no Xcode (na documentação da Apple você encontra isso como "Unwind Segue").
- No popup escolha o método `resetarNavegacao`. Teste seu programa e veja que já está funcionando.



Passando dados de uma scene para outra

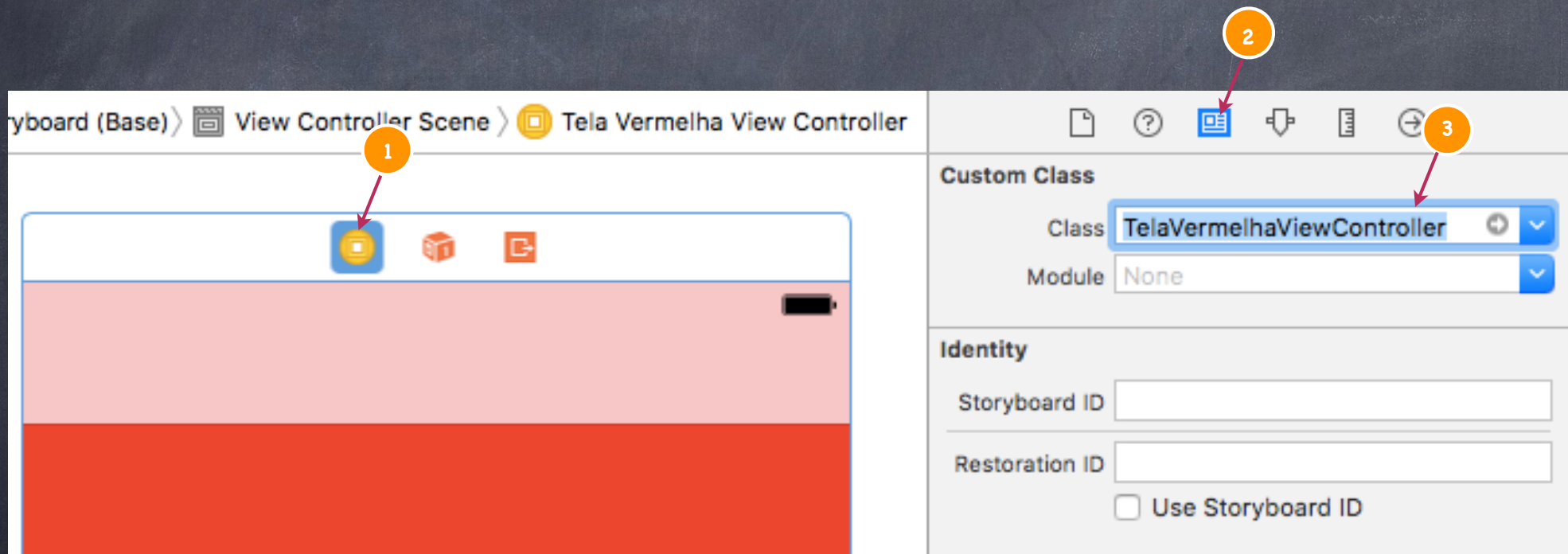
- Para isso precisamos sobrescrever um método que existe em `UIViewController` chamado "prepareForSegue", este método é executado antes da transição da próxima scene, este método passa como argumento informações sobre a scene que será executada, incluindo seu identificador e a referência para o controlador.
- Vamos criar um controller para a Tela Vermelha, clique em File → New → File → iOS → Cocoa Touch Class → Next, daremos o nome da classe de `TelaVermelhaViewController`, esta deve ser **subclasse de `UIViewController`**, não marque os checkbox, lembre-se de definir a tela Vermelha como "dona" dessa classe.



Navigation Controller

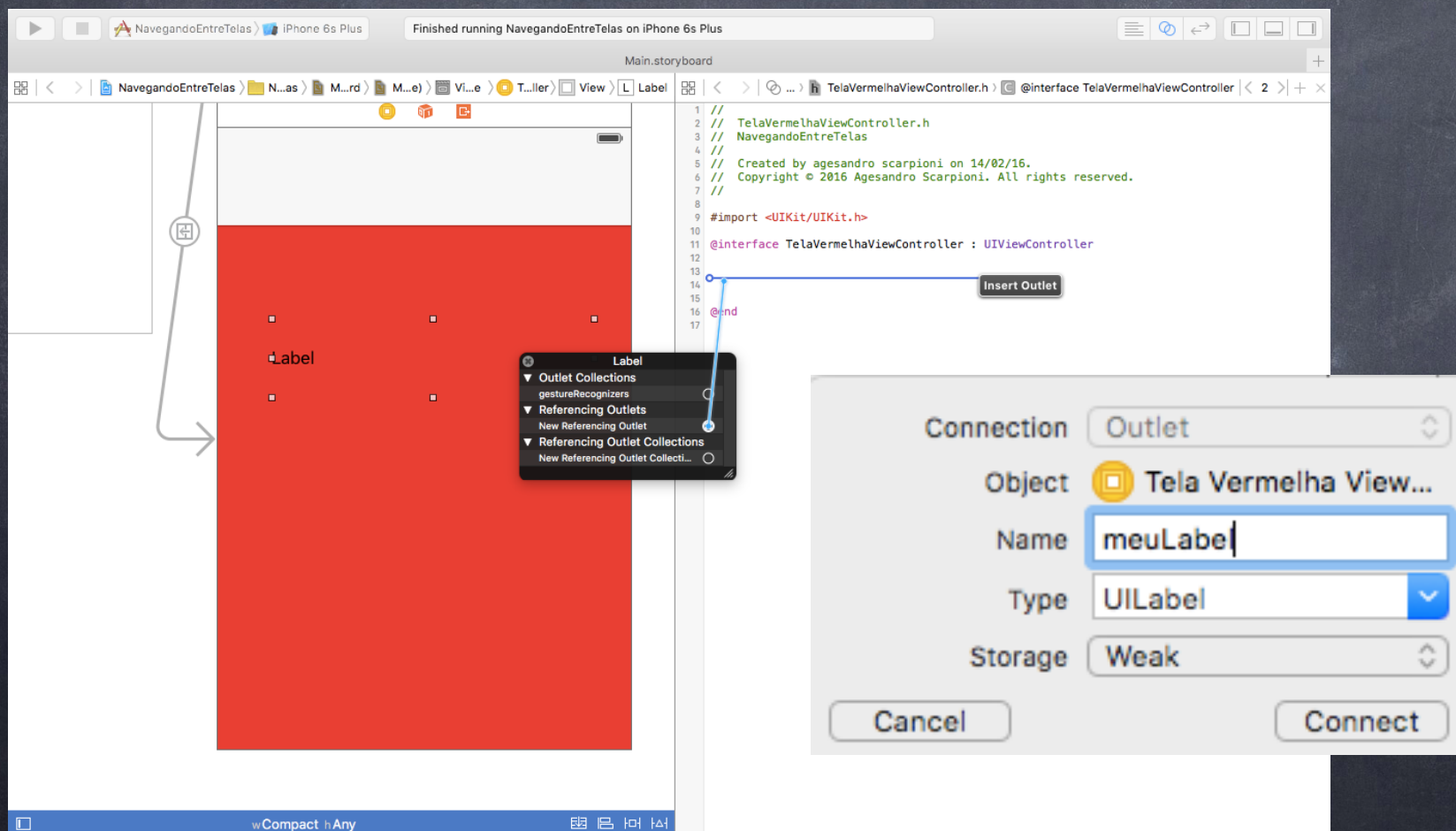
Passando dados de uma scene para outra

- Definindo a Scene da Tela Vermelha como "dona" da classe TelaVermelhaViewController, digite TelaVermelhaViewController em Custom Class.



Passando dados de uma scene para outra

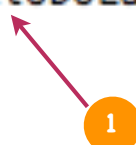
- Coloque na tela vermelha um Label e declare um Outlet desse label para exibirmos informações na tela Vermelha, botão direito sobre o Label, escolha New Referencing Outlet e arraste até a área da TelaVermelhaViewController.h, ao soltar preencha o nome do Outlet com o nome meuLabel.



Passando dados de uma scene para outra

- Declare uma propriedade do tipo NSString chamada textoDoLabel, como mostra a imagem abaixo (1):

```
1 //
2 // TelaVermelhaViewController.h
3 // NavegandoEntreTelas
4 //
5 // Created by agesandro scarpioni on 14/02/16.
6 // Copyright © 2016 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import <UIKit/UIKit.h>
10
11 @interface TelaVermelhaViewController : UIViewController
12
13
14 • @property (weak, nonatomic) IBOutlet UILabel *meuLabel;
15   @property (nonatomic, retain) NSString *textoDoLabel;
16
17
18 @end
19
```



Passando dados de uma scene para outra

- Para que a informação passada para a propriedade "textoDoLabel" apareça no Label desenhado na tela Vermelha, ainda na classe TelaVermelhaViewController.m, no método viewDidLoad digite a informação abaixo (1):

```
16
17 - (void)viewDidLoad {
18     [super viewDidLoad];
19     // Do any additional setup after loading the view.
20     _meuLabel.text = _textoDoLabel;
21     //ou
22     //self.meuLabel.text = self.textoDoLabel;
23 }
24
```

```
1 //
2 // TelaVermelhaViewController.m
3 // NavegandoEntreTelas
4 //
5 // Created by agesandro scarpioni on 14/02/16.
6 // Copyright © 2016 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import "TelaVermelhaViewController.h"
10
11 @interface TelaVermelhaViewController ()
12
13 @end
14
15 @implementation TelaVermelhaViewController
16
17 - (void)viewDidLoad {
18     [super viewDidLoad];
19     // Do any additional setup after loading the view.
20     _meuLabel.text = _textoDoLabel;
21     //ou
22     //self.meuLabel.text = self.textoDoLabel; 1
23 }
24
25 - (void)didReceiveMemoryWarning {
26     [super didReceiveMemoryWarning];
27     // Dispose of any resources that can be recreated.
28 }
29
```


Passando dados de uma scene para outra

- Vamos para o controller da tela X e importar (1) o arquivo TelaVermelhaViewController.h, depois invocar o prepareForSegue, veja o próximo slide.

```
1 //  
2 // TelaxViewController.m  
3 // NavegandoEntreTelas  
4 //  
5 // Created by agesandro scarpioni on 14/02/16.  
6 // Copyright © 2016 Agesandro Scarpioni. All rights reserved.  
7 //  
8  
9 #import "TelaxViewController.h"  
10 #import "TelaVermelhaViewController.h" ← 1  
11  
12 @interface TelaxViewController ()  
13  
14 @end  
15
```


Navigation Controller

Passando dados de uma scene para outra

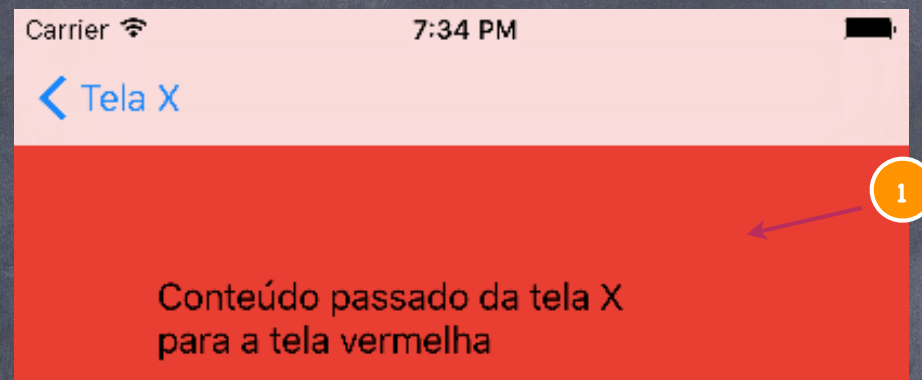
- Observe que você não precisa digitar a assinatura do método `prepareForSegue` (1), tire os comentários e monte apenas o IF.

```
1 //
2 // TelaxViewController.m
3 // NavegandoEntreTelas
4 //
5 // Created by Agesandro Scarpioni on 05/03/17.
6 // Copyright © 2017 Agesandro Scarpioni. All rights reserved.
7 //
8
9 #import "TelaxViewController.h"
10 #import "TelaVermelhaViewController.h"
11
12 @interface TelaxViewController ()
13
14 @end
15
16 @implementation TelaxViewController
17
18 - (void)viewDidLoad {
19     [super viewDidLoad];
20     // Do any additional setup after loading the view.
21 }
22
23
24 - (void)didReceiveMemoryWarning {
25     [super didReceiveMemoryWarning];
26     // Dispose of any resources that can be recreated.
27 }
28
29
30 #pragma mark - Navigation
31
32 // In a storyboard-based application, you will often want to do a little preparation before seguing
33 - (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
34     if([segue.identifier isEqualToString:@"telaxParaTelaVermelhaSegue"]){
35         TelaVermelhaViewController *t = segue.destinationViewController;
36         t.textoDoLabel = @"Conteúdo passado da tela X para a tela vermelha";
37     }
38 }
39
40 - (IBAction)abrirVerdeScene:(id)sender {
41     [self performSegueWithIdentifier:@"telaxParaTelaVerdeSegue" sender:sender];
42 }
43
44 - (IBAction)abrirVermelhaScene:(id)sender {
45     [self performSegueWithIdentifier:@"telaxParaTelaVermelhaSegue" sender:sender];
46 }
47
48
49 @end
50
```


Navigation Controller

Passando dados de uma scene para outra

- Teste seu App e verifique que os dados já são passados de uma tela para outra (1), caso seu texto seja muito grande altere a quantidade de linhas (2) que será exibida em seu label e deixe-o um pouco maior.



Prática

- Crie uma estrutura de Navigation Controller com 2 botões na tela principal, onde o botão 1 deve chamar uma tela qualquer e esta tela deve ter o texto "retroceder" para que volte para a tela com 2 botões.
- Para o segundo botão chame uma outra tela do tipo SHOW DETAIL com 1 botão "Fechar" e um botão "Sobre", o botão "Sobre" deve chamar uma tela e passar para um label a seguinte frase: "Desenvolvido por <seu nome>".