

*cit* 

# Hilton Pintor

Desenvolvedor (iOS/tvOS/watchOS)



[hiltonpintor@gmail.com](mailto:hiltonpintor@gmail.com)

// Ementa

## **Swift:**

- Por que desenvolver em swift?
- Variáveis, constantes e operadores
- Tipos, optionals
- Coleções (arrays, dicionários)
- Fluxo de controle
- Condicionais e loops
- Funções e closures
- Enums
- Classes e structs

## **Storyboard:**

- Model
- View
- Controller
  
- Delegação
- UIKit
  - Labels
  - Botões
  - Textfields
  - Tableview
- Storyboard
- Navegação
- Persistência local

## Swift:

- ~~— Por que desenvolver em swift?~~
- ~~— Variáveis, constantes e operadores~~
- ~~— Tipos, optionals~~
- ~~— Coleções (arrays, dicionários)~~
- ~~— Fluxo de controle~~
- ~~— Condicionais e loops~~
- ~~— Funções e closures~~
- ~~— Enums~~
- ~~— Classes e structs~~

## Storyboard:

- ~~— Model~~
- ~~— View~~
- ~~— Controller~~
- ~~— Delegação~~
- ~~— UIKit~~
  - ~~— Labels~~
  - ~~— Botões~~
  - ~~— Textfields~~
  - ~~— Tableview~~
- ~~— Storyboard~~
- ~~— Navegação~~
- ~~— Persistência local~~

## // Proposta semana 02

Segunda

TableView

Terça

Persistência

Quarta

Projeto

Quinta

Projeto

Sexta

Projeto

Extra

# // Aula 09

// Projeto



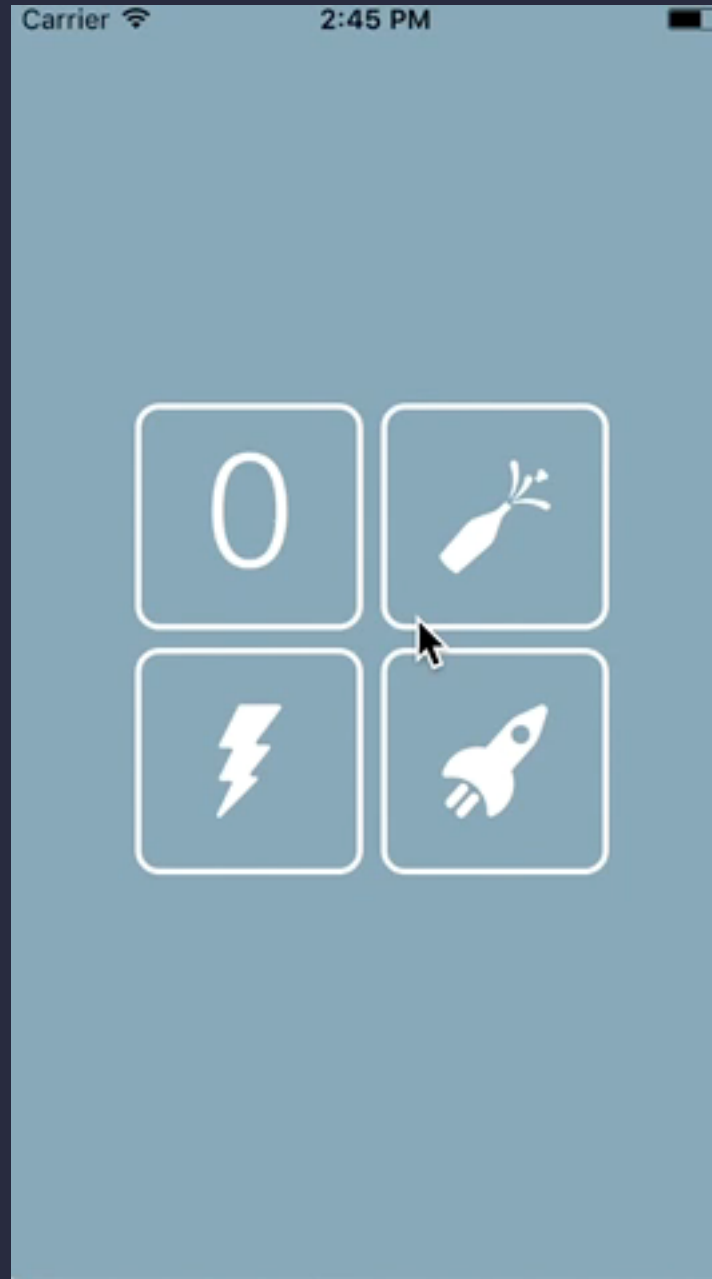
## // Fizz Buzz - jogo/teste

Escreva um programa que imprima números de **1 a 100**:

- Mas para múltiplos de **3**, imprima "**Fizz**" ao invés do número.
- Para múltiplos de **5**, imprima "**Buzz**" ao invés do número.
- E para números que são múltiplos de **3 e 5**, imprima "**FizzBuzz**"

1, 2, Fizz, 4, Buzz, Fizz, 7, 8, Fizz, Buzz, 11, Fizz, 13, 14, Fizz Buzz, 16, 17, Fizz, 19, Buzz, Fizz, 22, 23, Fizz, Buzz, 26, Fizz, 28, 29, Fizz Buzz, 31, 32, Fizz, 34, Buzz, Fizz, ...

## // Fizz Buzz - App



## // Tela de Jogo

Faça um app que permita que o jogador jogue uma partida de **fizz buzz** através do uso de **botões** da interface de seu app.

Quando o usuário **errar** o valor da vez, ou o jogo **acabar**:

- o usuário deve ser direcionado para uma **tela de cadastro**

## // Tela de Cadastro

Nessa tela o usuário pode:

- inserir seu **nome** (num **text field**)
- submeter sua **foto** (que deve ser mostrada em uma **Image View**)
- ver o seu **score** *número de acertos* (numa **label**)
- **salvar** os valores cadastrados (direcionando para **tela de scores**)
- **cancelar** (voltando para **tela de jogo** para **jogar novamente**)

## // Tela de Scores

Nessa tela o usuário pode:

- Ver uma **Table View** onde cada célula tem:
  - **Nome** do jogador
  - **Foto** do jogador
  - **Score** obtido
- **Deletar** células da **Table View**
- Observar que o conteúdo da **Table View** tem **persistência**
- **Voltar** para tela de jogo sem passar pela de cadastro (**unwind segue**)

## // Extra

- Ver os scores ordenados do maior pro menor

// Avaliação

## // Domínio dos Conteúdos

- programação em Swift
- MVC
- StoryBoard
- Botões
  - IBActions
- Labels
- Text Fields
- Image View
- Navigation Controller
- Segues (show)
- Segues (unwind)
- Pedir permissão de privacidade
- Table View
  - Células customizadas
  - Data Source
  - Interação com Células
    - seleção
    - inserção
    - remoção
- Persistência (Core Data)

## // Domínio dos Conteúdos

- **Quarta, Quinta e Sexta** serão utilizados para a avaliação
- É permitido:
  - Consultas online
  - Consultas offline
  - Tirar dúvidas com instrutor
  - Tirar dúvidas com colegas
  - Fazer projeto em casa
- É proibido:
  - entregar projeto feito por terceiros
  - programar pelo coleguinha



// Submissão

ASSUNTO: [JORNADA2018] Avaliação

CORPO:

- Nome Completo
- zip/rar do projeto

[hiltonpintor@gmail.com](mailto:hiltonpintor@gmail.com)

# DÚVIDAS

