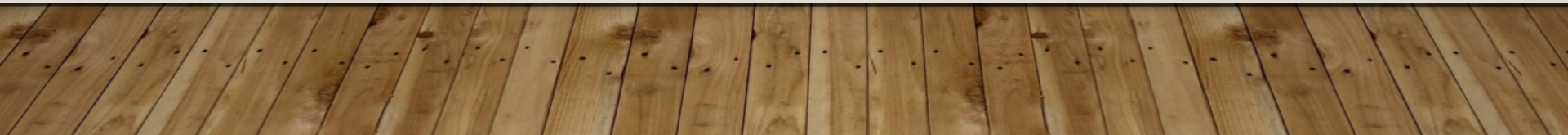


DESENVOLVIMENTO PARA IOS – APPLE 02

IOS SDK 10

Professor: Pedro Henrique
prof.pedrohenrique.iossdk@gmail.com



AGENDA

- Comentários sobre a WWDC'16;
- Animação avançada:
 - UIDynamicAnimator;
 - UIDynamicBehavior;
- Hora de Brincar!

OBJETIVOS DO DIA

- Compreender as diferenças entre a animação da aula passada e a animação da aula de hoje;
- Conhecer os meios para aplicar o `UIDynamicAnimator`.

COMENTÁRIOS SOBRE A WWDC'16



WWDC'16

- Novidades do iOS 10;
- Swift 3.0 e o futuro;
- Novidades no Objective-C;
- Novas APIs
 - Reconhecimento de voz e Siri;
 - VoIP;
 - iMessage Apps;
 - Map Extensions;
- Xcode 8

WWDC'16

- Resumo:
- <https://macmagazine.com.br/2016/06/13/wwdc-2016-apple-apresenta-as-10-novidades-mais-interessantes-do-ios-10/>
- Geral:
- <https://developer.apple.com/videos/wwdc2016/>
- Novidades do iOS 10
- <https://blogdoiphone.com/2016/06/ios-10/>

ANIMAÇÃO AVANÇADA (UIDYNAMICANIMATOR)

- Abordagem diferente em relação ao que vimos semana passada;
- Como?
 - Configura-se “atributos” relacionados à física dos objetos;
 - Em seguida, permite-se que os objetos animem (de acordo com a “física” definida) e, por fim, cheguem a um estado de equilíbrio;
 - É possível evitar o estado de equilíbrio, mas pode levar a problemas de performance;

UIDYNAMICANIMATOR

- Passo-a-passo:
 1. Criar um objeto do tipo UIDynamicAnimator;
 2. Adicionar a ele UIDynamicBehaviors (gravidade, colisões, etc);
 3. Adicionar UIDynamicItems (UIView, normalmente) ao UIDynamicBehavior;
 4. Plim! Magia acontece!

UIDYNAMICBEHAVIOR

- **UIGravityBehavior:**
 - Ângulo;
 - Magnitude ($1.0 = 1000 \text{ pt/s}^2$);
- **UICollisionBehavior**
 - Modo;
 - Boundary (“em quais lados” ocorre);
- **UIAttachmentBehavior**
 - A um item;
 - A uma âncora;
 - Ao centro;

UIDYNAMICBEHAVIOR

- **UISnapBehavior**

- Imagine quatro molas, uma em cada lado da tela. Agora você tem o controle individual sobre o “amortecimento” de cada uma; (damping)

- **UIPushBehavior**

- Modo (contínuo ou instantâneo);
- Direção;
- Magnitude (1.0 acelera uma view 100x100 em 100 pt/s²);

UIDYNAMICBEHAVIOR

- Provavelmente você irá querer fazer seu próprio **Behavior**...
- Isso se faz criando uma classe filha de **UIDynamicBehavior** e chamando o método:
 - - (**void**)addChildBehavior:(**UIDynamicBehavior ***)**behavior**;
- Dentro dos métodos sobrescritos:
 - **addItem** e **removeItem**

UIDYNAMICBEHAVIOR

- Todos os **UIDynamicBehavior** conhecem o **UIDynamicAnimator** do qual fazem parte;
- O UIDynamicBehavior tem a propriedade
 - **@property (copy) void (^action)(void);**
- A propriedade **action** é um bloco. Você pode definir para que ela faça qualquer coisa, mas faça um código eficiente, pois ele vai ser chamado **MUITAS VEZES** por segundo!

HORA DE BRINCAR!

- Quadrados que explodem!
 - Código no GitHub (vamos refazer juntos)

DESAFIO

- Fazer os quadrados aparecerem na posição onde ocorrer um evento de tap;
- Fazer os quadrados quicarem quando tocarem o “chão”, usando snap;
- Criar quadrados de cores **realmente** aleatórias.

