1

Anyone can Animate

(Even if they can't draw)

Andy Dent's

strongly-held opinions

weakly-held pencils

Animation as Life?

(but not as we know it)

See lots of links on my github site (also be on the last slide)

github.com/AndyDentFree/DDD19

- Val Head's book *Interface Animation*, articles and videos
- Disney's book The Illusion of Life
- As video and gifs 8 of which are on the following pages
- Really good article illustrating them with longer clips from Disney and Pixar movies, and explanations

Principle 1 - Squash and Stretch

Principle 2 - Anticipation
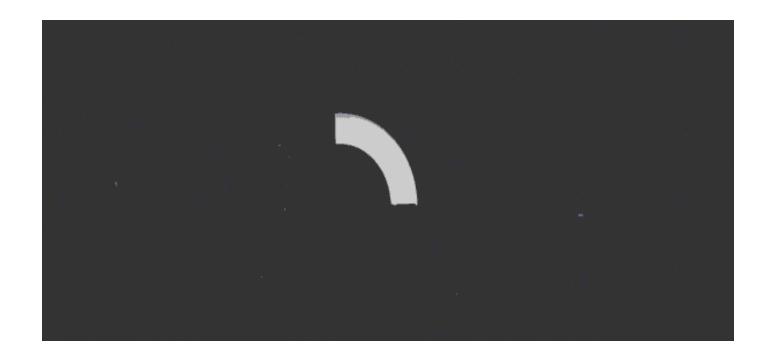
Principle 5 - Follow Through & Overlapping
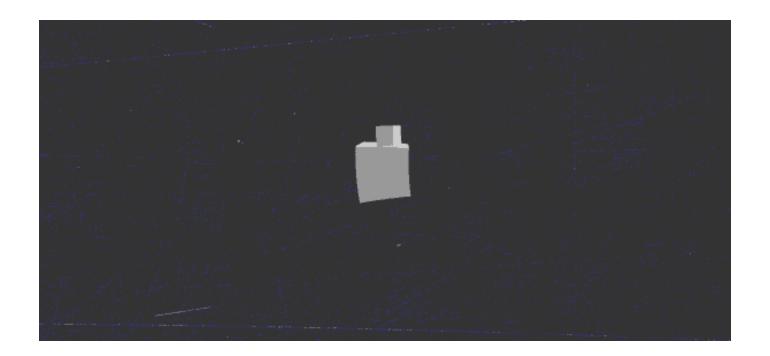
Principle 6 - Slow In and Slow Out
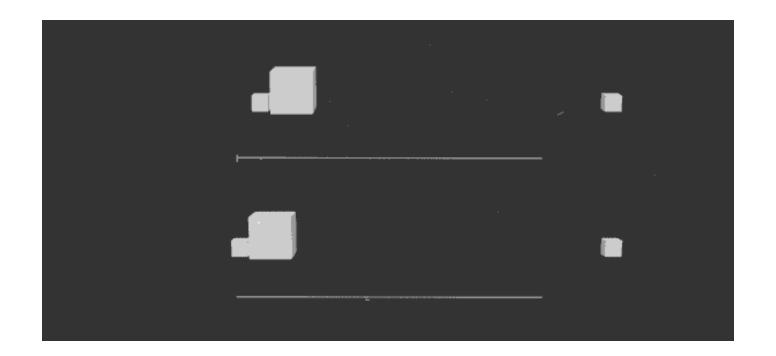
Principle 7 - Arcs
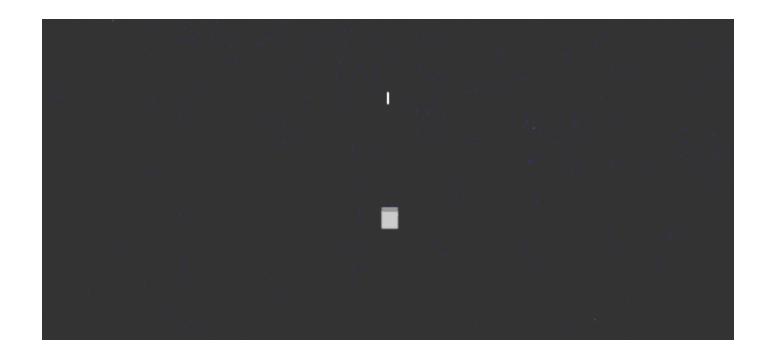
Principle 8 - Secondary Action

Principle 9- Timing

Principle 10 - Exaggeration

Why Animate? - Designers

Distract

Amuse

Inform

Why Animate? - Developer's View

D istract

I nform

E ntertain

Animation as a new Developer Hell

Animation's aren't the only thing that go in circles.

The designer-developer loop finds new things to design that have to be manually translated to code.

and then the loop closes

Coming to save us (last time)

New Breed of Tools

Start with Sketch or other vector prototype

Add animation

Generate working code

Mostly Mobile Tools

Supernova Studio

Haiku

Flow

Web world

SVG-based

GSAP Greensock JS-driven

other?

Simple Animation Example

Supernova Editor

Refresh Project Localization iOS Android React Flutter − Q + 100% Screen Navigation Simulator Publish Export CodeX Settings Left Right

playFullsize

PlayFullsizeViewController.swift ⌄ Swift 4.2 ⌄ ☑ Sho...

Animate on Tap on toolbarRemnant +
■ play-editing-toolbar +
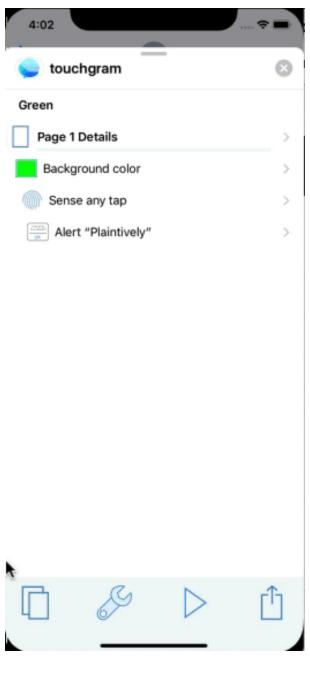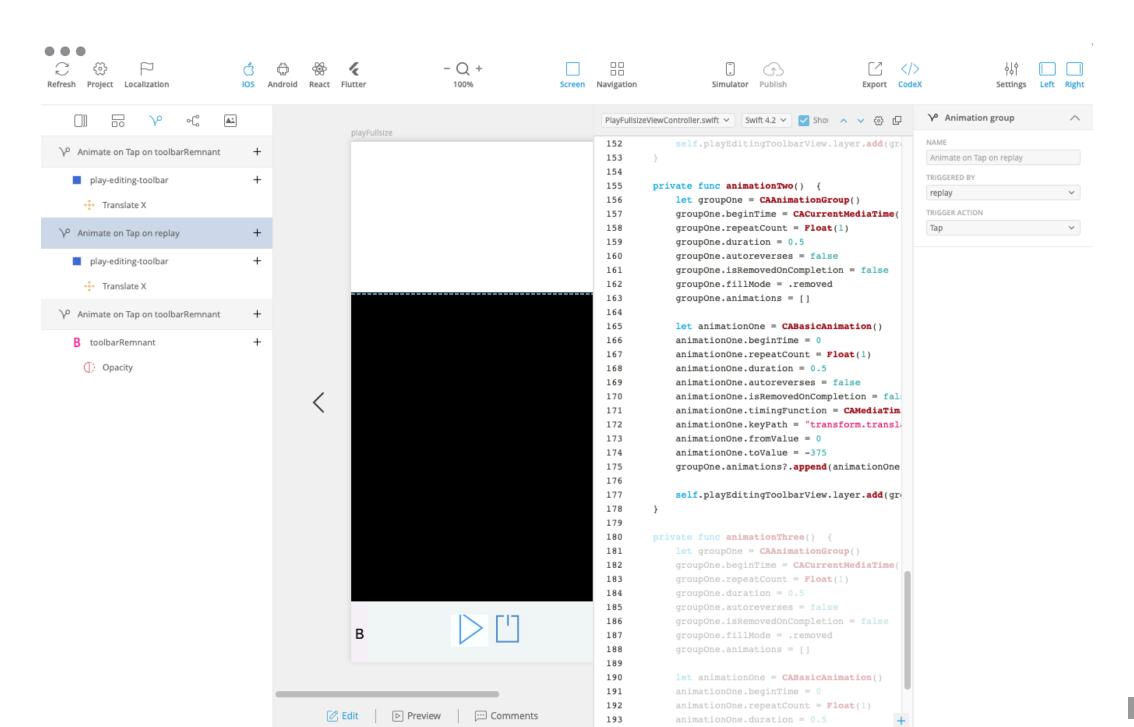Translate X

Animate on Tap on replay +
■ play-editing-toolbar +
Translate X

Animate on Tap on toolbarRemnant +
B toolbarRemnant +
Opacity

Animation group ∧

NAME
Animate on Tap on replay

TRIGGERED BY
replay ⌄

TRIGGER ACTION
Tap ⌄

```swift
152        self.playEditingToolbarView.layer.add(gr
153    }
154
155    private func animationTwo()  {
156        let groupOne = CAAnimationGroup()
157        groupOne.beginTime = CACurrentMediaTime(
158        groupOne.repeatCount = Float(1)
159        groupOne.duration = 0.5
160        groupOne.autoreverses = false
161        groupOne.isRemovedOnCompletion = false
162        groupOne.fillMode = .removed
163        groupOne.animations = []
164
165        let animationOne = CABasicAnimation()
166        animationOne.beginTime = 0
167        animationOne.repeatCount = Float(1)
168        animationOne.duration = 0.5
169        animationOne.autoreverses = false
170        animationOne.isRemovedOnCompletion = fal
171        animationOne.timingFunction = CAMediaTim.
172        animationOne.keyPath = "transform.transl.
173        animationOne.fromValue = 0
174        animationOne.toValue = -375
175        groupOne.animations?.append(animationOne
176
177        self.playEditingToolbarView.layer.add(gr
178    }
179
180    private func animationThree()  {
181        let groupOne = CAAnimationGroup()
182        groupOne.beginTime = CACurrentMediaTime(
183        groupOne.repeatCount = Float(1)
184        groupOne.duration = 0.5
185        groupOne.autoreverses = false
186        groupOne.isRemovedOnCompletion = false
187        groupOne.fillMode = .removed
188        groupOne.animations = []
189
190        let animationOne = CABasicAnimation()
191        animationOne.beginTime = 0
192        animationOne.repeatCount = Float(1)
193        animationOne.duration = 0.5
```

B

Edit Preview Comments

Further Details

github.com/AndyDentFree/DDD19

Contains this presentation and links to all tools mentioned, full samples and media to build anything shown in the movies.

Twitter: @andydentperth