

Touch & multi-touch

Fabrice.Kordon@lip6.fr



The return of UIResponder

2

📱 Already known

- 🔊 Handling shake and remote control

📱 Similar principle

- 🔊 Become «first responder»
- 🔊 Support the integrated protocol

```
override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?)
```

```
override func touchesEnded(_ touches: Set<UITouch>, with event: UIEvent?)
```

```
override func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?)
```

```
override func touchesCancelled(_ touches: Set<UITouch>, with event: UIEvent?)
```

- 🔊 Remember

▶ UIApplication, UIView and UIViewController inherit from UIResponder

touches: **Set<UITouch>**



A set of UITouch



Inside a UITouch?

- tapCount (number of taps)
- view (associated view)
- location(in:) (position in the view)
- previousLocation(in:) (previous location in the view)
- type (type of touch, to be detailed later)
- force
 - ▶ 1.0 is the force of an average touch
- phase
 - ▶ began, moved, stationary, ended, cancelled



Remark, maintain touches in Objective-C?

- Better use copy rather than retain

event: **UIEvent?**

A UIEvent

- allTouches (set of touches)
- type
 - ▶ touches, motion, remoteControl, presses
- subtype
 - ▶ none, motionShake, remoteControlPlay, remoteControlPause, remoteControlStop, remoteControlPlayPause, remoteControlNextTrack, remoteControlPreviousTrack, remoteControlBeginSeekingBackward, remoteControlEndSeekingBackward, remoteControlBeginSeekingForward, remoteControlEndSeekingForward
- timestamp
- A useful method

```
func touches(for view: UIView) -> Set<UITouch>?
```

RTFM!



• and...

Constraints?

Is there a maximum number of touch



Small device

Up to five
(touchesCancelled triggered at 6th?)



Large device

15 touches observed



As a conclusion...

Key mechanism

- Brings a huge richness to you games

Rich mechanism

- Only the essential presented here

- About gestures?

- ▶ To be seen later

- About Pencil?

- ▶ To be seen later

- About Force touch?

- ▶ To be seen later



RTFM!