SCIENCES SORBONNE UNIVERSITÉ

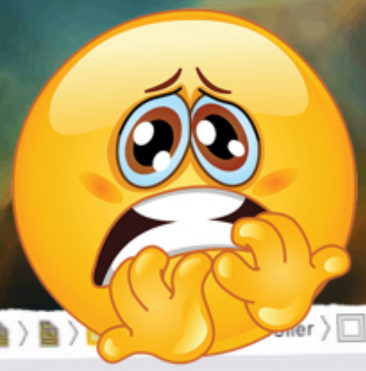
# Views, handling actions

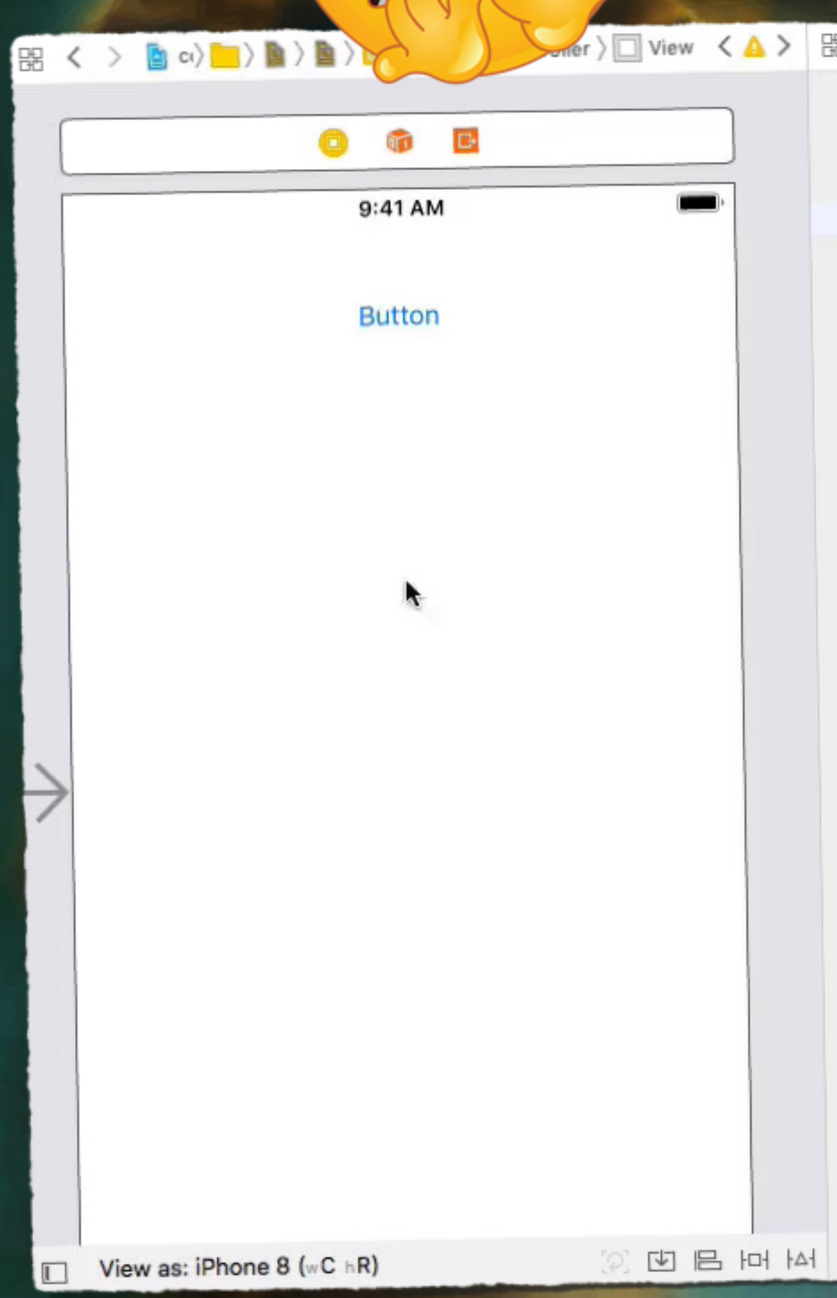
Fabrice.Kordon@lip6.fr

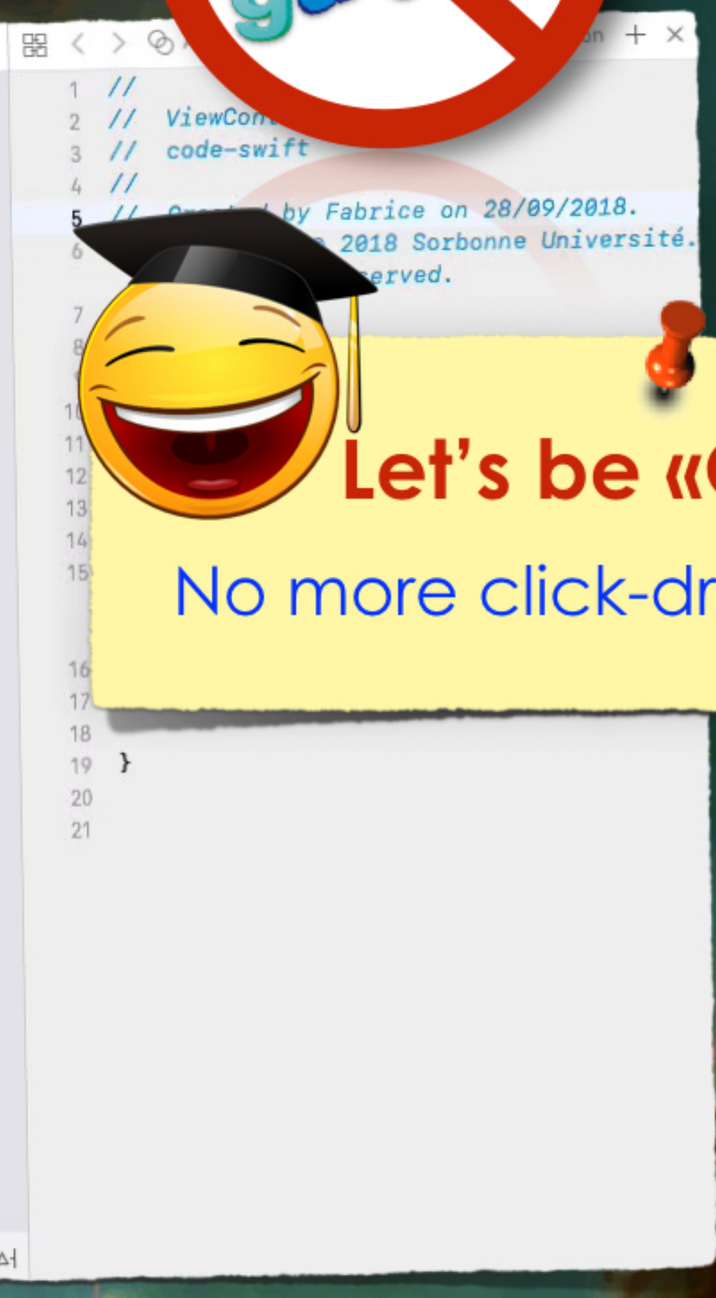
LIP6

# As an introduction…








## Let's be «Geeks»

No more click-drag-end-drop!

# How to catch actions?



## I — create (programmatically) the «widget»

- It is a view (like all widgets)… with more features
- UIButton properties
  - **tintColor, backgroundColor,titleLabel, etc.**
- Method to setup the title
  - **setTitle:forState / setTitle**


## II — associate a target

- addTarget:action:forState / addTarget
  - **The object to be invoked**
  - **The method to be invoked**
  - **The event to be caught**

# How to catch actions?



## I — crec rogrammatically) the «widget»

- It is a view (
- UIButton prop
  - **tintColor, bac**
- Method to se
  - **setTitle:forState / setTitle**

**This stands for any «widget»**

Be careful about caught events

## II — a

- addTar
  - **The ob**
  - **The me**
  - **The ev**

**The event-list**

touchDown, touchDownRepeat, touchDragInside, touchDragOutside, touchDragEnter, touchDragExit, touchUpInside, touchUpOutside, touchCancel, valueChanged, editingDidBegin, editingChanged, editingDidEnd, editingDidEndOnExit, allTouchEvents, allEditingEvents, applicationReserved, systemReserved, allEvents

# More on event catching



## Disassociate a target?

- Technique 1
  - **Make a new association**
- Technique 2
  - **The removeTarget method**

## Remark about addTarget & removeTarget

- They are defined at a «higher level»
- Also stand in UIView
  - **UIButton (and so) inherit from UIView**

# As a conclusion…



- Once again, is'n't it easy?


- You can «read your mistakes»
- Remind the videos about event-based programming
  - This is event-based programming