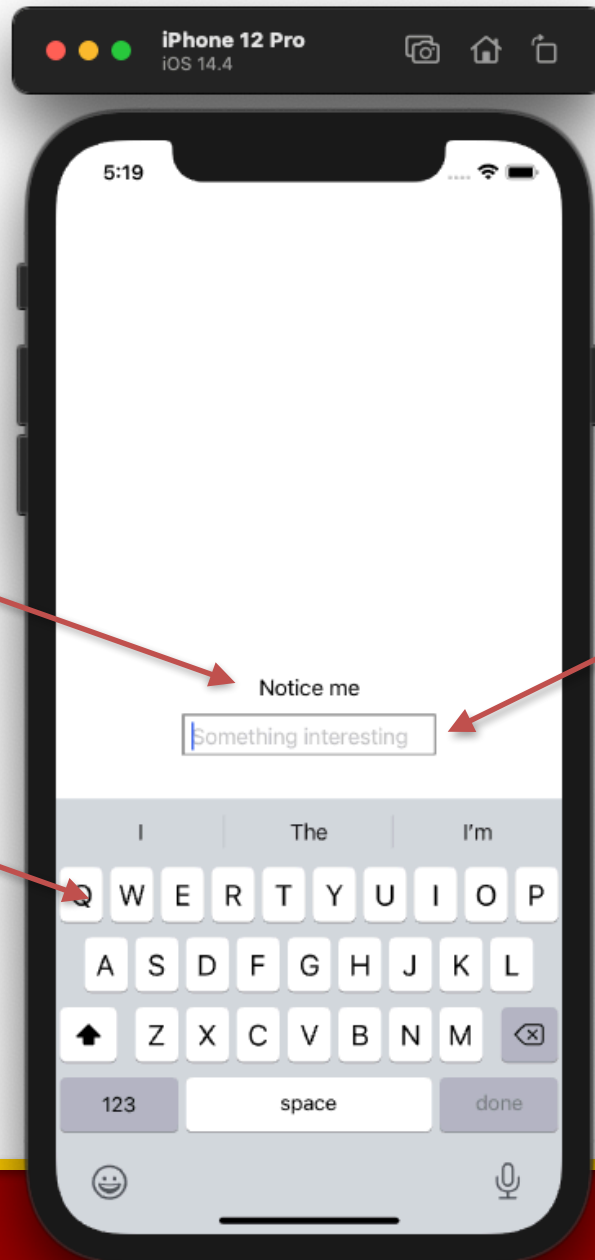




Programmatic UI Recap

ITP344X: Advanced iOS App Development W4

Jack Xu, M.S. (not quite)
boyuanxu@usc.edu



UILabel

UITextField & Delegate

IQKeyboardManager



Outline

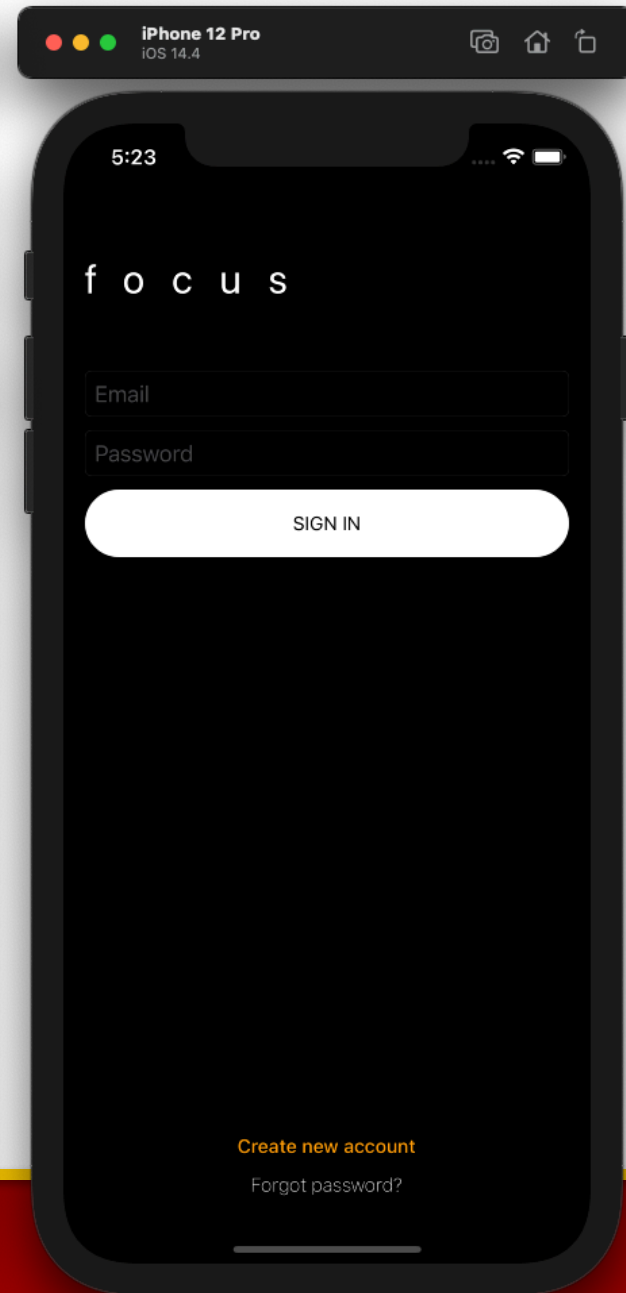
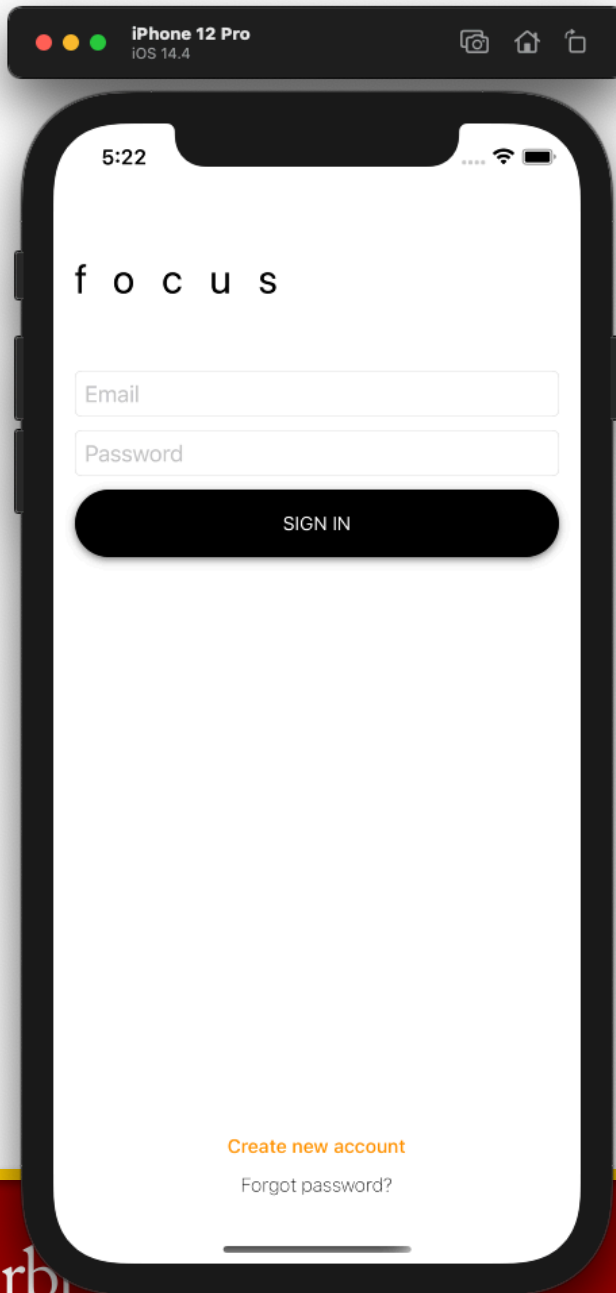
- Programmatic UI (cont.) ✓
- SnapKit ✓
- Building a Sign-In Page
- Firebase

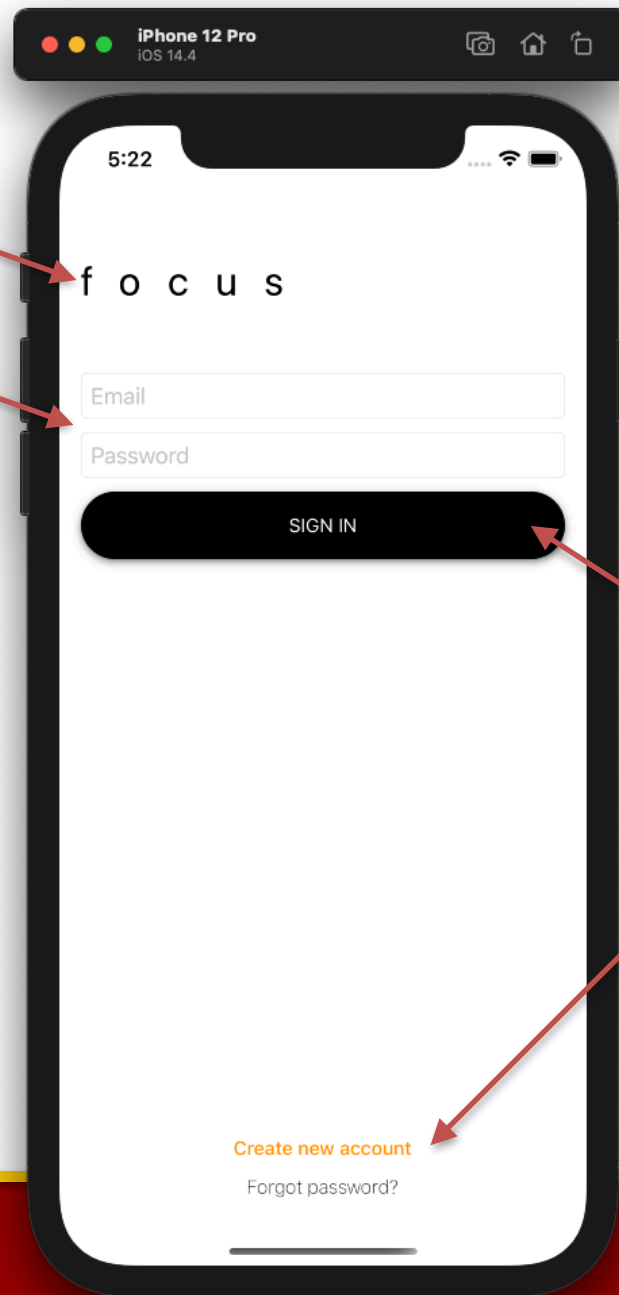


Programmatic Sign-In Page

ITP344X: Advanced iOS App Development W4

Jack Xu, M.S. (not quite)
boyuanxu@usc.edu





UILabel + NSAttributedString

UITextField

UIButton + NSAttributedString



Sign-In Page Outline

- UIView
- UIScrollView
- UIImageView
- UILabel
- UITextField
- UIButton: CALayer & addTarget



**Give a man a fish, and you feed him for a day.
Teach a man to fish, and you feed him for a lifetime.**



UIStackView

- Axis: Which direction?
- Alignment for the cross-direction?
- Subview distribution?
- Spacing between subviews?



Review: Compression & Hugging

- CompressionResistancePriority
 - The higher the value, the more it resists being compressed
- HuggingPriority
 - The higher the value, the more it resists being expanded

Intrinsic Content Size



So far, all of the examples have used constraints to define both the view's position and its size. However, some views have a natural size given their current content. This is referred to as their *intrinsic content size*. For example, a button's intrinsic content size is the size of its title plus a small margin.

Not all views have an intrinsic content size. For views that do, the intrinsic content size can define the view's height, its width, or both. Some examples are listed in Table 3-1.

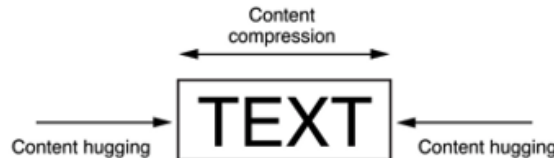
Table 3-1 Intrinsic content size for common controls

View	Intrinsic content size
UIView and NSView	No intrinsic content size.
Sliders	Defines only the width (iOS). Defines the width, the height, or both—depending on the slider's type (OS X).
Labels, buttons, switches, and text fields	Defines both the height and the width.
Text views and image views	Intrinsic content size can vary.

The intrinsic content size is based on the view's current content. A label or button's intrinsic content size is based on the amount of text shown and the font used. For other views, the intrinsic content size is even more complex. For example, an empty image view does not have an intrinsic content size. As soon as you add an image, though, its intrinsic content size is set to the image's size.



A text view's intrinsic content size varies depending on the content, on whether or not it has scrolling enabled, and on the other constraints applied to the view. For example, with scrolling enabled, the view does not have an intrinsic content size. With scrolling disabled, by default the view's intrinsic content size is calculated based on the size of the text without any line wrapping. For example, if there are no returns in the text, it calculates the height and width needed to layout the content as a single line of text. If you add constraints to specify the view's width, the intrinsic content size defines the height required to display the text given its width.

Auto Layout represents a view's intrinsic content size using a pair of constraints for each dimension. The content hugging pulls the view inward so that it fits snugly around the content. The compression resistance pushes the view outward so that it does not clip the content.





NSAttributedString

- String: What do you want to display?
- Attributes: How do you want to display?
 - [NSAttributedString.Key: Any]
 - [ setting name :  setting value]
 - Name: NSAttributedString.Key.font
 - Value: UIFont.systemFont(ofSize: 30)
- Some UI elements have .attributedText property



UITextField

- Some useful attributes:
 - .borderStyle
 - .placeholder
 - .autocorrectionType
 - .keyboardType
 - .isSecureEntry



UIButton - CALayer

- How to beautify the button?
 - Colors are for normies but we are chads
 - We need some real sh*t
- CALayer is the answer!
 - Add shadow
 - Add rounded corner



UIButton - CALayer

- Access using `button.layer`
 - `layer.shadowXXX`
 - Radius, Offset, Color, Opacity, Path
 - `layer.cornerRadius`



UIButton - addTarget

- Adding an action to a button programmatically
- `func addTarget(_ target: Any?, action: Selector, for controlEvents: UIControl.Event)`
 - Target is usually self
 - Selector — calls a function but it must be `@objc`
 - controlEvents: Event that triggers the action



Outline

- Programmatic UI (cont.) ✓
- SnapKit ✓
- Building a Sign-In Page ✓
- Firebase



Firestore Integration

ITP344X: Advanced iOS App Development W4

Jack Xu, M.S. (not quite)
boyuanxu@usc.edu