Custom graphics, text input and webviews

Week 2 Day 5

LIGHTHOUSE LABS

Text Input

Text input classes

We have a few ways to get text input on iOS.

UITextField is a view that can take in a single line of text.

UITextView is a UIScrollView subclass that can take in multiple lines.

First responder

The first responder is the first link in the responder chain.

Having first responder status means that the control gets dibs on handling events

ex) text input from the keyboard or any shake events (such as “Shake to Undo”)

First responder (contd)

A control is sent the becomeFirstResponder message.

That control will keep this status until another control becomes the first responder or the current control is sent the resignFirstResponder message.

Text class delegates

Informs the delegate about activities in the text area.

By default, hitting the enter key doesn’t remove responder status, we need to do that via the delegate.

Keyboard types

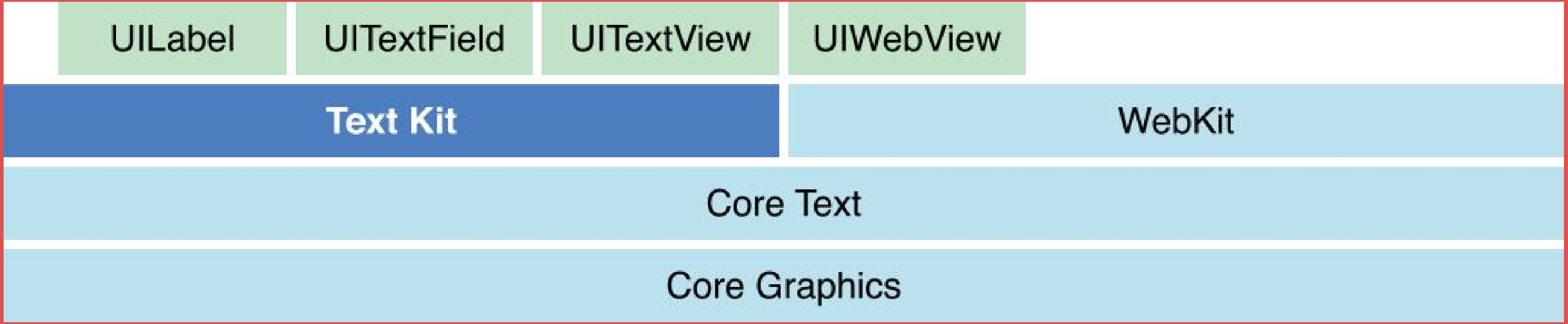
Anything that can accept text input needs to implement a protocol called UITextInputTraits

It’s used by the OS to determine what type of keyboard to display and if autocorrect is enabled.

TextKit

- TextKit is the framework that underlies text display as of iOS 7.
- Previously, fine-grained control over text layout and display was difficult using UIKit classes.
- TextKit opens up complex layouts, custom drawing effects, and many other benefits.

TextKit in context



UILabel | UITextField | UITextView | UIWebView

Text Kit | WebKit

Core Text

Core Graphics

Demo UITextView/Field

Web Views

Web views

- Web views are a very commonly used object in iOS apps.

- Use them to implement your own in-app web previews, or create web-based components inside your native app.

UIWebView, WKWebView, SFSafariViewController

- UIWebView
  - around since the first iOS SDK.
  - easy to use
  - slow javascript
- WKWebView
  - introduced in iOS 8
  - much more customizable
  - runs javascript faster.
- SFSafariViewController
  - iOS 9 +

UIWebView capabilities

- UIWebView is very basic in terms of its navigation capabilities: just loadRequest, goForward or goBack.

- It has a delegate to deal with user triggered navigation events.

Javascript in web views

- Both types of web view can be given javascript to evaluate inside of the the content they’ve loaded.
- This allows for some communication between the app and the web content.

When to use WKWebView

When you

- Need fast javascript (hybrid apps)
- Need fine-grained control of navigation
- Can’t use Safari View Controller

Safari View Controller

New in iOS 9 - A way to embed not just webkit, but the full Safari experience into your app.

Big wins: Content Blocking, Extensions (1password, etc.)

Web Views Demo

LIGHTHOUSE LABS

Drawing

Hand made locally-sourced free range pasture-to-plate pixels

Custom Graphics

To draw to the screen:

1. Subclass UIView or one of its derivatives

2. Override it’s drawing method, and add some Core Graphics drawing code.

-(void)drawRect:(CGRect)rect

Called by UIKit when a view needs (re)drawing

The rect specifies an area in the view’s bounds to redraw. If you can, try to keep the drawing just to the space shown within that passed rect.

Core Graphics

C framework with drawing functions

Drawing takes place inside of a context, a drawing area in memory that can be used to draw to the Screen, PDF, Printer.

Set Colors, Fonts, Textures, create paths, then stroke or fill

DEMO - Getting pixels on screen

- Create custom view subclass
  - Override drawRect
  - Draw something
- Create instance of custom view in storyboard
  - Create outlet if we need to pass in data

Core Graphics Paths

To draw a line or other vector shape, we need to create a path.

CGContextMoveToPoint

CGContextAddLineToPoint

CGContextStrokePath

CGContextFillPath

UIBezierPath

Objective C wrapper for Core Graphics paths

-moveToPoint:

-addLineToPoint:

-stroke

-fill

drawRect:(CGRect)rect

For drawing on screen, never call drawRect directly.

What you should call to redraw is:
-(void)setNeedsDisplay or
-(void)setNeedsDisplayInRect:(CGRect)rect

DEMO - Drawing paths
With Core Graphics and UIBezierPath

Summary

- Draw to screen with UIView’s -drawRect: and CoreGraphics or UIBezierPath
- Text input involves the responder chain, TextKit, and UITextInputTraits
- Web views are very powerful

LIGHTHOUSE LABS