

CSC 471 / 371
Mobile Application
Development for iOS



Prof. Xiaoping Jia
School of Computing, CDM
DePaul University
xjia@cdm.depaul.edu
[@DePaulSWEEng](https://twitter.com/DePaulSWEEng)

Build Your
First iOS App



Outline

- Xcode IDE basics
- Creating a new project
- An anatomy of an iOS project
- Widgets and attributes
- App icons and images
- Running apps in iOS simulators



DEPAUL UNIVERSITY 3

Xcode IDE and iOS SDK



- We will be using
 - SDK for iOS 9
 - Xcode 7.2
- Download and install the latest SDK & IDE from the App Store
- PLEASE, PLEASE
 - Once you get everything working
 - **DO NOT upgrade or change versions**



DEPAUL UNIVERSITY 4

What's New in Xcode 7



- Support iOS 9 and Watch OS 2
- Improvements
 - Swift programming language 2.1
 - Playground
 - Adaptive layout
- Stack view
- Storyboard references
- UI testing

DEPAUL UNIVERSITY 5

What's Unique About Mobile Apps

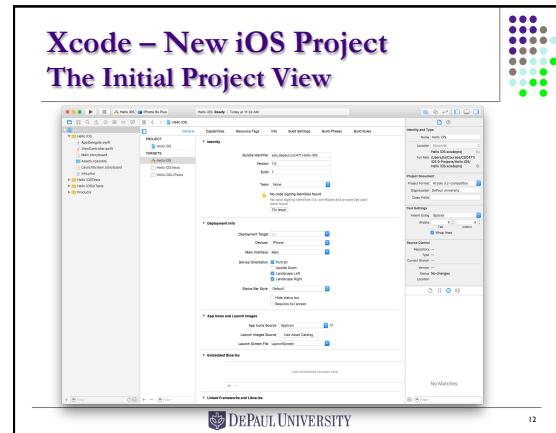
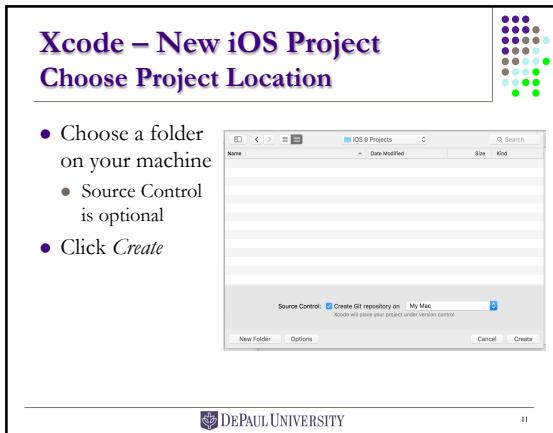
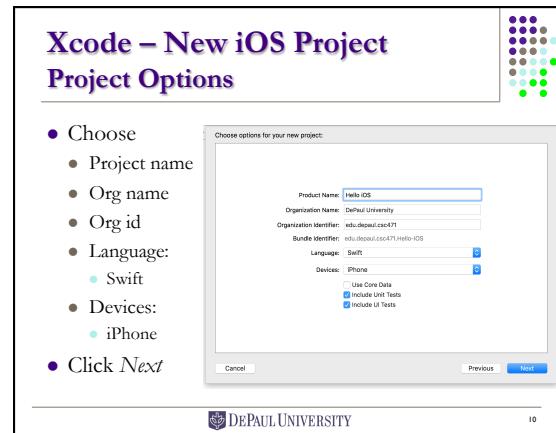
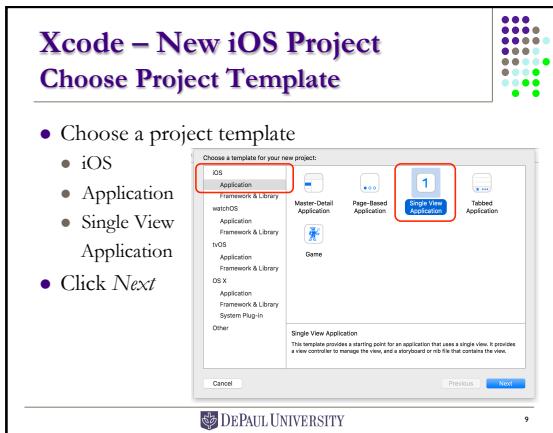
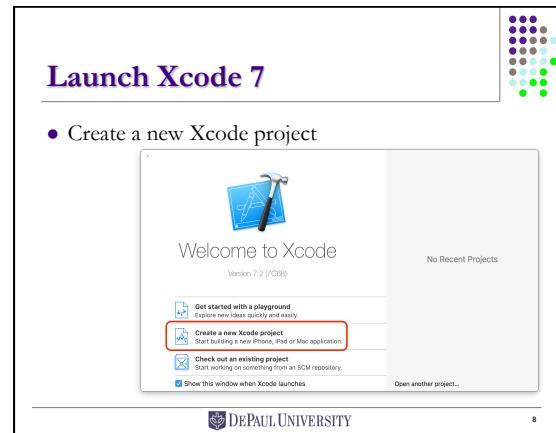
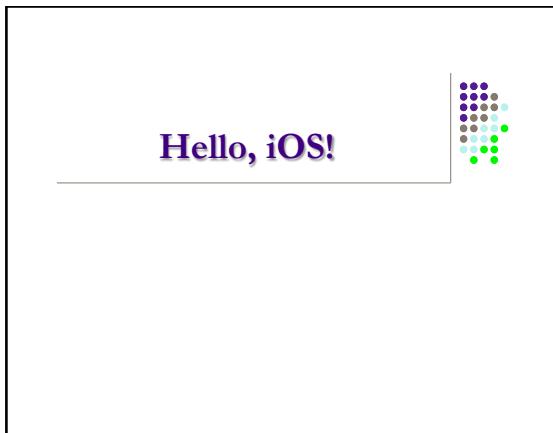


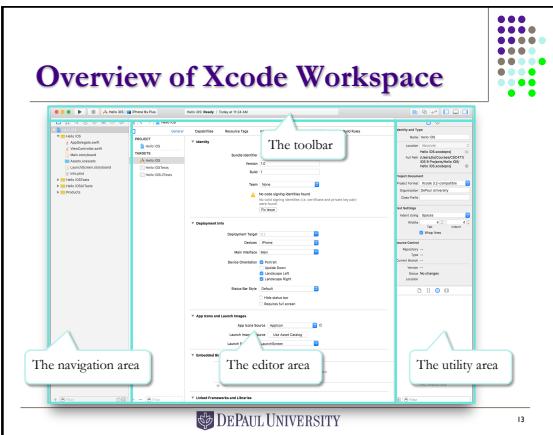
- One active application at a time
- One window on screen
- Quick response time
- Limited screen size
- Limited system resources
 - Memory, battery power
- No garbage collection
 - You have more responsibilities

Until iOS 9
Multi-tasking and split screen in iOS 9

Only due to larger screens.
Most restrictions still apply.

DEPAUL UNIVERSITY 6



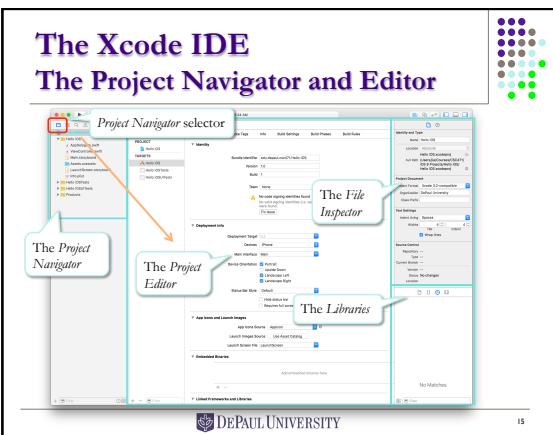


Editors in Xcode

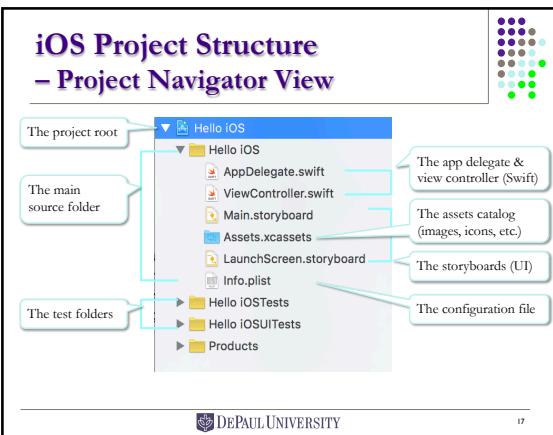
- Project editor
 - Edit project-level properties and configurations
 - build options, target architectures, and app entitlements, etc.
- Interface Builder
 - Graphically create and edit user interface files.
 - storyboards
- Source editor
 - Edit text files
 - Swift source code, etc.

DEPAUL UNIVERSITY

14



An Anatomy of an iOS Project

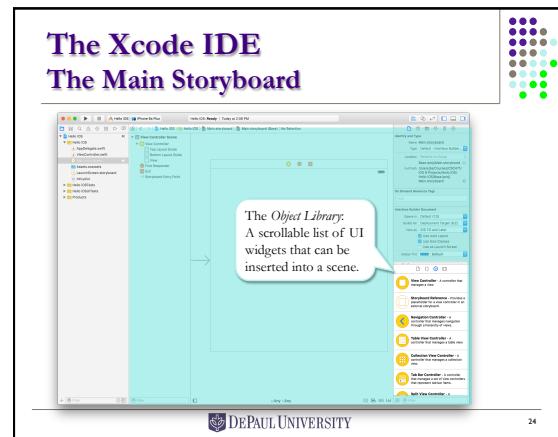
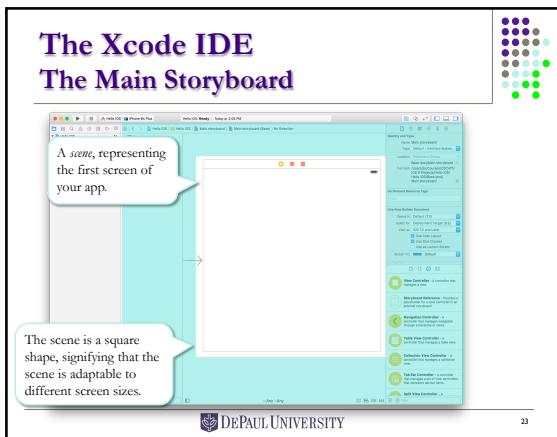
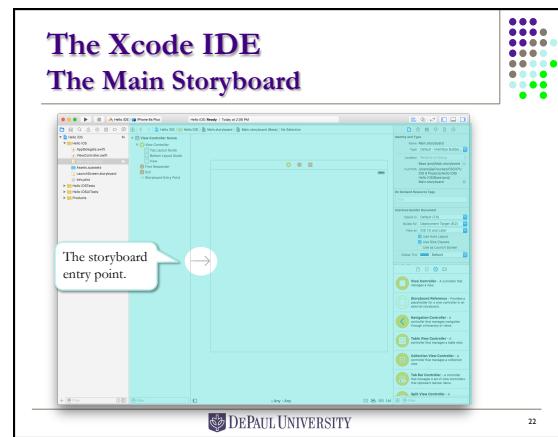
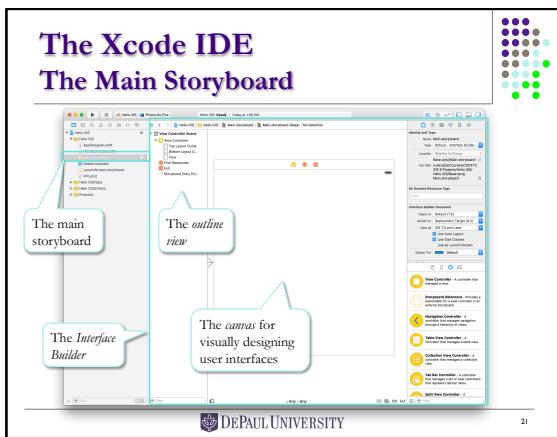
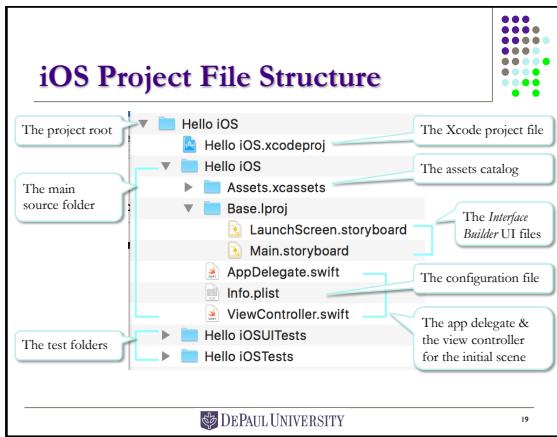


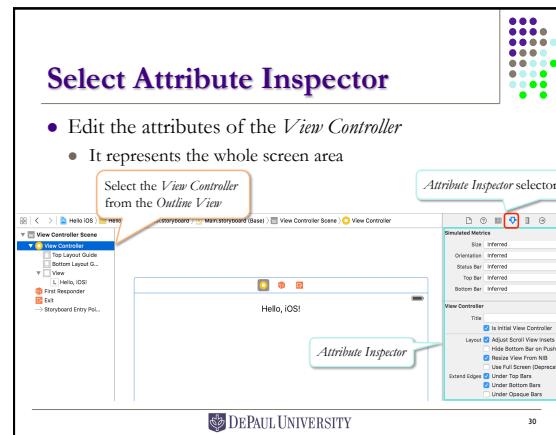
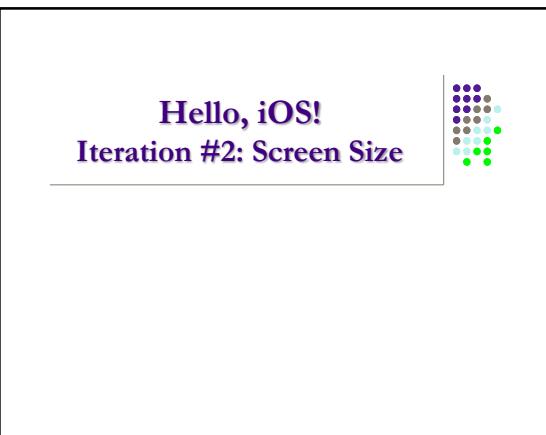
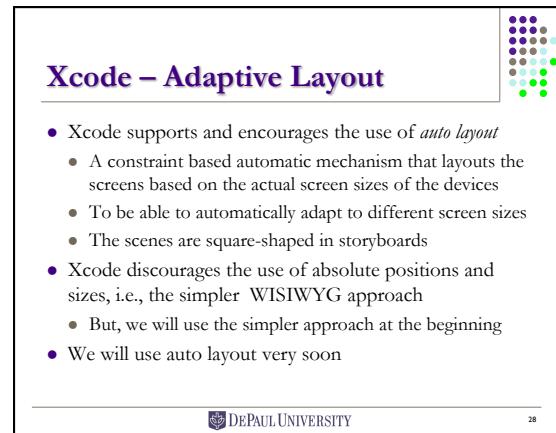
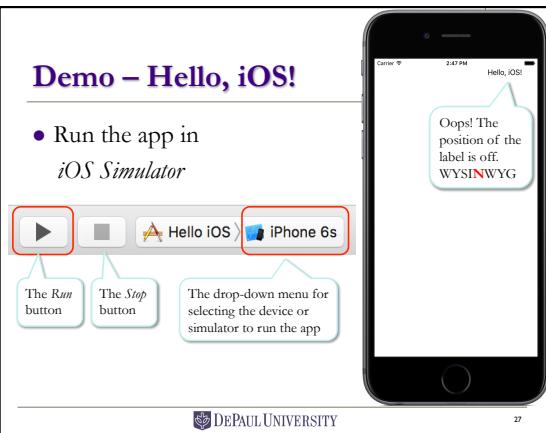
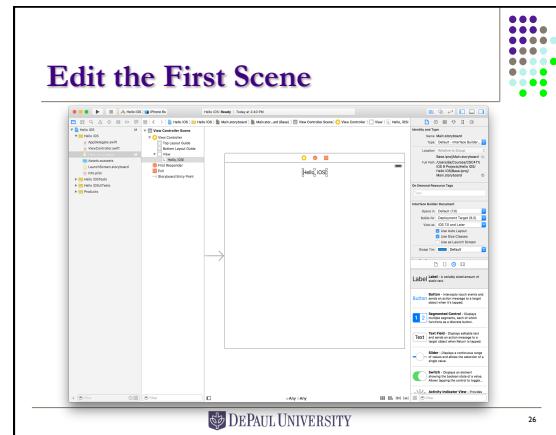
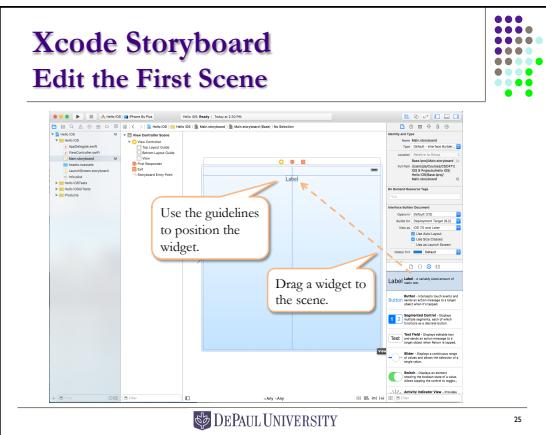
File Types in an Xcode Project

- *.storyboard
 - UI views and widgets
 - Multiple views
 - Application-wide screen flows
- *.xcassets
 - Asset Catalogs
 - Resources
 - images, icons, and data etc.
- *.plist
 - Property list files
 - Configurations
 - Application data
- *.swift
 - Swift source files
 - App delegate
 - View controllers

DEPAUL UNIVERSITY

18





Set Screen Size

- In the *Simulated Metrics* section of the *Attribute Inspector*
 - Change the *Size* attribute to *iPhone 4.7-inch*
- You are now designing the layout for a specific screen size only

DEPAUL UNIVERSITY

Set Screen Size – Rectangular-Shaped Scene

The scene now is a rectangular shape, representing the dimension of the screen of an iPhone

DEPAUL UNIVERSITY

Set Screen Size – Reposition the Label and Run

DEPAUL UNIVERSITY

Hello, iOS! Iteration #3: Attributes

Attribute Inspector

- Edit the attributes of the selected widget

Widgets can also be selected from the *Outline View*

Selected the widget

Attribute Inspector

DEPAUL UNIVERSITY

Edit Widget Attributes

- Change the font style & size and the color of the label
 - Changes in attributes may affect the layout

Attributes Color & Font changed

The label needs resizing

DEPAUL UNIVERSITY

Edit Widget Attributes

- Adjust layout
- Run

DEPAUL UNIVERSITY 37

Hello, iOS! Iteration #4: App Icons

DEPAUL UNIVERSITY

App Icons

- Every app has an *app icon*
- The app icon appears in several contexts and in different sizes, in *points* (pt)
 - The *Home* screen, 60x60
 - Results of *Search*, 40x40
 - The *Settings* app, 29x29
- *Point*, a logical unit that maintains a constant physical dimension on devices with different screen densities.

DEPAUL UNIVERSITY 39

App Icons in Different Contexts

The Home screen 60x60pt
The Search results 40x40pt
The Settings 29x29pt

DEPAUL UNIVERSITY 40

iPhone Screen Resolutions

	iPhone 3G iPhone 3GS	iPhone 4 iPhone4S	iPhone 5 iPhone 5S	iPhone 6 iPhone 6S	iPhone 6+ iPhone 6S+
Size (diagonal)	3.5"	3.5"	4"	4.7"	5.5"
Display type	Classic	Retina	Retina	Retina (HD)	Retina HD
Pixels (px)	320x480	640x960	640x1136	750x1334	1080x1920
Density (ppi)	163	326	326	326	401
Points (pt)	320x480	320x480	320x568	375x667	414x736
Scale factor	1x	2x	2x	2x	3x

- An icon or an image should be the same size *in points* across all devices
 - Different sizes *in pixels*
- For best results, provide icons/images in different pixel sizes

DEPAUL UNIVERSITY 41

Managing Images with Xcode Asset Catalogs

- Xcode uses *asset catalogs* to simplify the management of icons and images
- Each asset catalog consists of
 - a *name*, used to refer to the asset in your app
 - a *set of images* of different sizes (in pixel)
- *At runtime*, iOS will load the image from the set that is most appropriate for the current scale factor
 - An image will be scaled if necessary
 - Runtime cost and reduced quality

DEPAUL UNIVERSITY 42

Add App Icons

- In Project Navigator,
 - Select `images.xcassets`
 - Select `AppIcon`

The image wells.
You can drag and drop images into these wells.

The image set view

DEPAUL UNIVERSITY 43

Add App Icons

- You should provide images of sizes:
 - Home screen (60pt):
 - 120x120 (2x), 180x180 (3x)
 - Search (40pt):
 - 80x80 (2x), 120x120 (3x)
 - Settings (29pt):
 - 58x58 (2x), 87x87 (3x)
- Use PNG format –
 - Portable Network Graphics format
- Also support vector PDF format

DEPAUL UNIVERSITY 44

Add App Icons

- Drag and drop image files into the image wells

DEPAUL UNIVERSITY 45

App Icons of Hello iOS

DEPAUL UNIVERSITY 46

Demo – Hello, iOS!

- Missed something?
 - Watch the lecture over again
- You need to feel comfortable and confident in dealing with Xcode
 - Experiment with various options

DEPAUL UNIVERSITY 47

Assignments

DEPAUL UNIVERSITY

Join the iOS Developer Program

- Enrolled students will be invited to developer program
- Login to the iOS Developer Program
- Download and install Xcode 7

DEPAUL UNIVERSITY

49



Assignment Submission

- Submit a single zip file for each project
 - Zip the contents of the project folder
- Include your source code files
 - *.swift, *.plist, *.storyboard etc.
 - Image files
 - Project files
- Before you submit
 - Build the code, make sure everything compiles and works

DEPAUL UNIVERSITY

51

Programming Assignment 1

- Go though the process and create your own *Hello World* app.
 - Do it.
 - Don't worry, if you stumble. Most of you will at first.
- Once you get it working
 - Experiment with some variations and try it again!
 - Explore Xcode features
 - This needs to be second nature

DEPAUL UNIVERSITY

50



Next ...

- Learn Swift programming language
- Dive into iOS SDK

❖ Xcode, iOS, WatchOS are trademarks of Apple Inc.

DEPAUL UNIVERSITY

52

