

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 11
 - Xcode 9 and Swift 4
 - **Mandatory for all programming assignments and the final project**
- 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 9




- Support iOS 11 and Watch OS 4
- Improvements
 - Swift programming language, version 4
 - Playground
 - Adaptive layout
- Stack view
- Storyboard references
- UI testing

DEPAUL UNIVERSITY 5

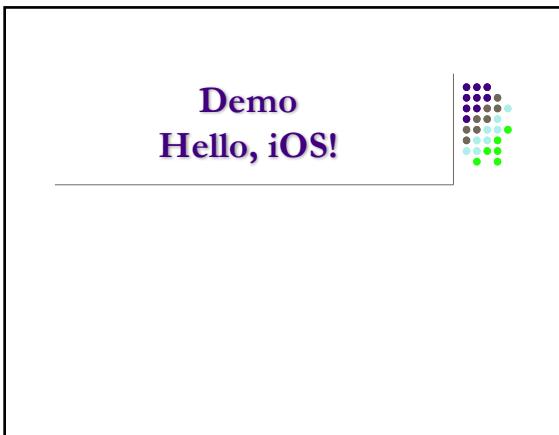
What's Unique About Mobile Apps

- 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



Launch Xcode 9

- Create a new Xcode project

Xcode – New iOS Project Choose Project Template

- Choose a project template
 - iOS
 - Application
 - Single View Application
- Click *Next*

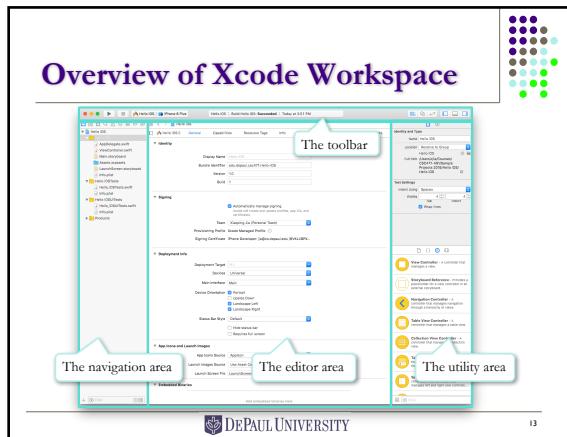
Xcode – New iOS Project Project Options

- Choose
 - Project name
 - Team
 - Org name
 - Org id
 - Language: Swift
 - Devices: iPhone
- Click *Next*

Xcode – New iOS Project Choose Project Location

- Choose a folder on your machine
 - Source Control is optional
- Click *Create*

Xcode – New iOS Project The Initial Project View

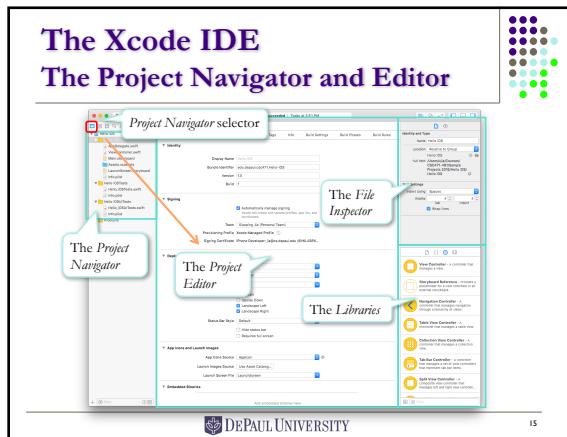


Editors in Xcode

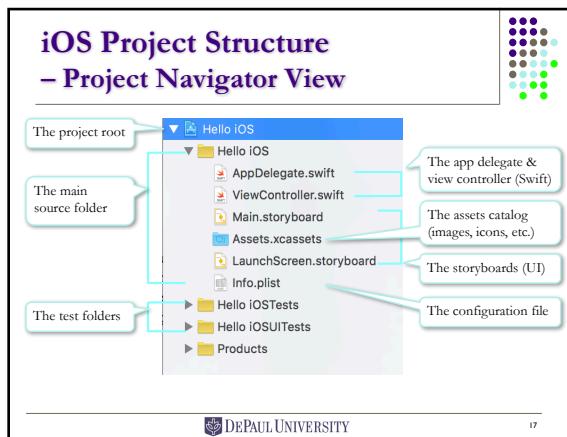
- Project editor
 - Edit project-level properties and configurations
 - build options, target architectures, and app entitlements, etc.
- Interface Builder
 - Graphically create and edit user interface files.
 - storyboards
- Source editor
 - Edit text files
 - Swift source code, etc.

DEPAUL UNIVERSITY

14



An Anatomy of an iOS Project

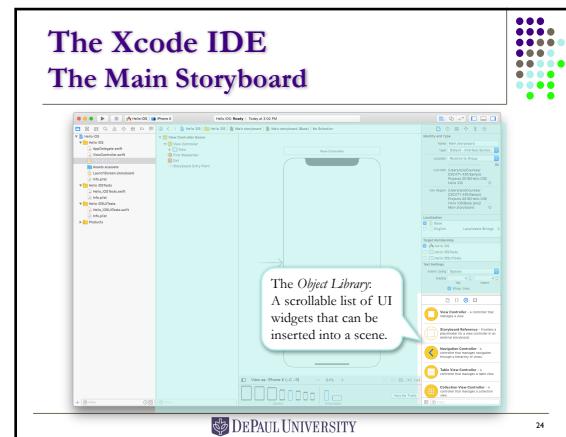
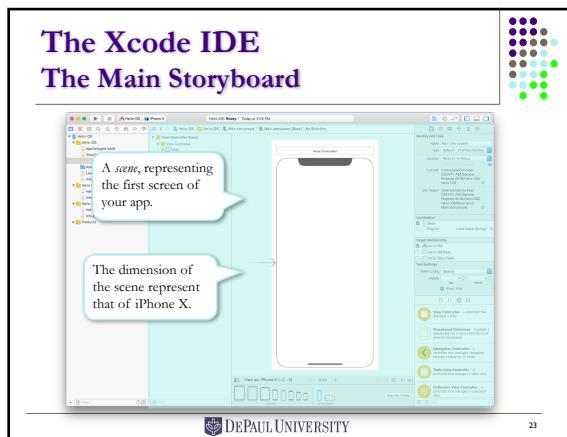
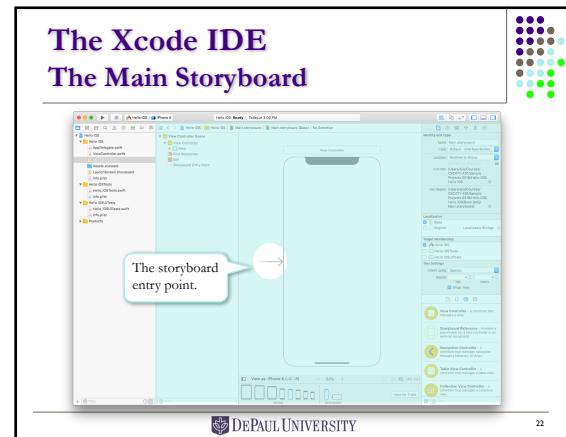
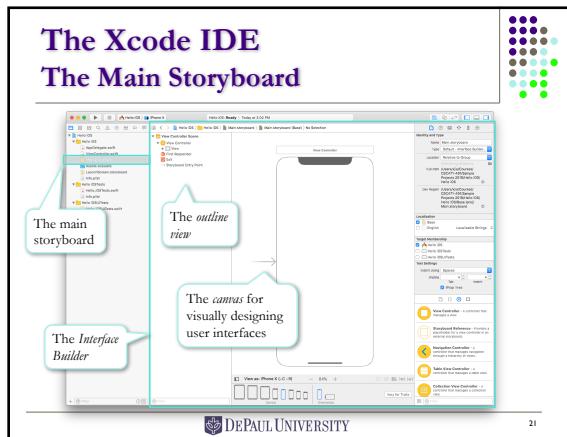
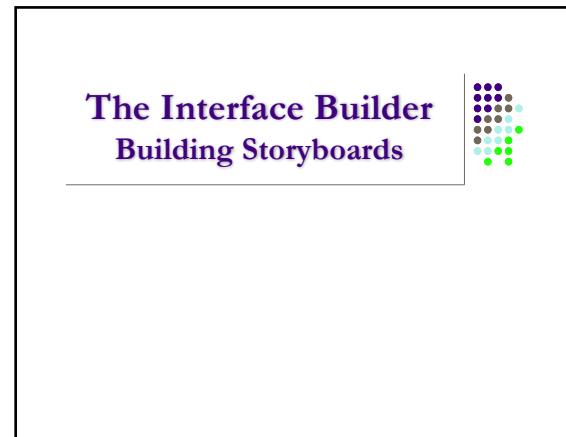
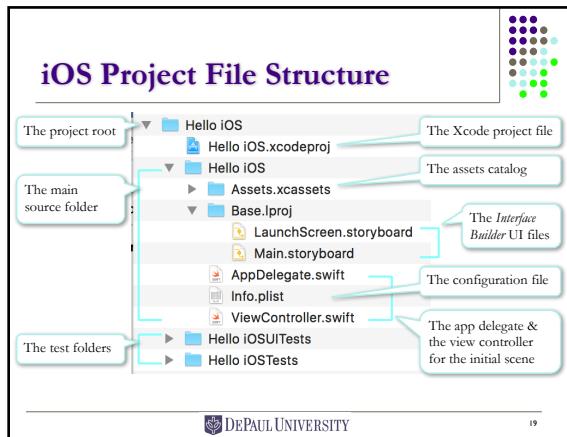


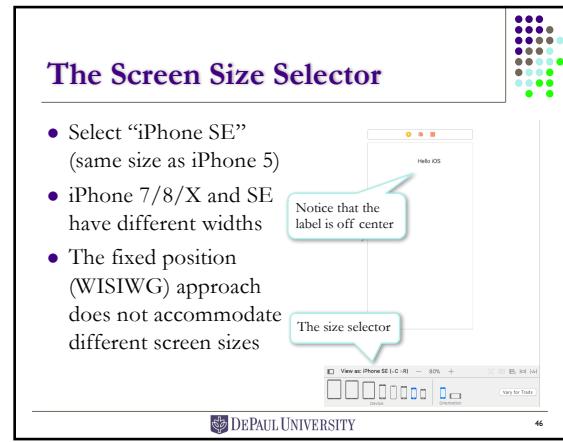
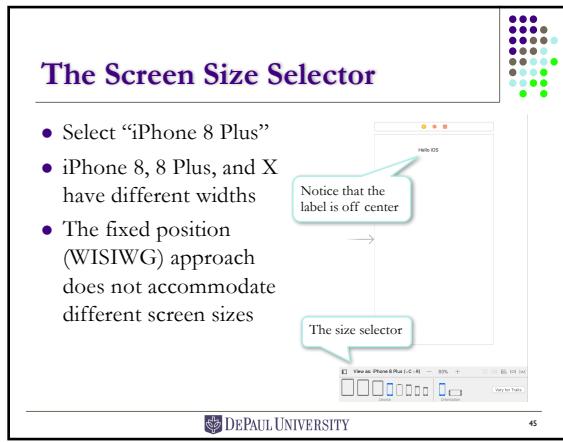
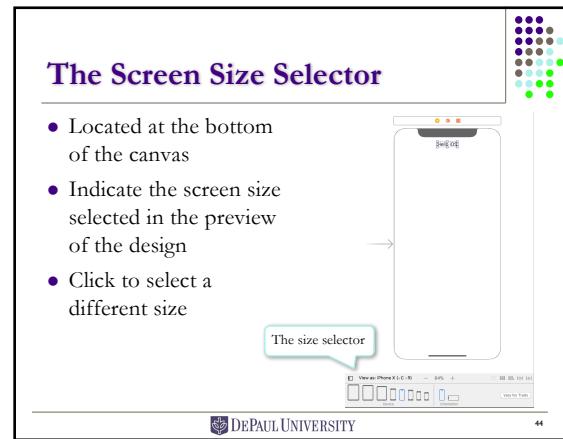
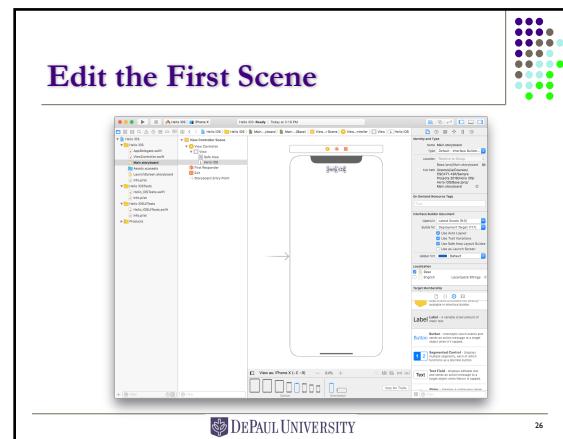
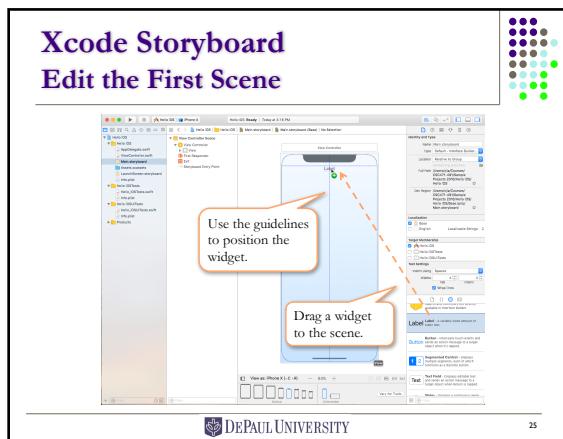
File Types in an Xcode Project

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

DEPAUL UNIVERSITY

18





The Screen Size Selector

- Select “iPhone 8” landscape orientation
- The fixed position (WISIWG) approach does not accommodate orientation changes

DEPAUL UNIVERSITY 47

Xcode – Adaptive Layout

- Xcode supports and encourages the use of *auto layout*
 - A constraint based automatic mechanism that layouts the screens based on the actual screen sizes of the devices
 - To be able to automatically adapt to different screen sizes and both orientations
- Xcode discourages the use of absolute positions and sizes, i.e., the simpler WISIWYG approach
 - But, we will start with the simpler approach at the beginning
- We will be using auto layout very soon

DEPAUL UNIVERSITY 28

Hello, iOS! Iteration #2: Attributes

DEPAUL UNIVERSITY

The Attribute Inspector

- Edit the attributes of the selected widget

Attribute Inspector selector
DEPAUL UNIVERSITY 35

Edit Widget Attributes

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

DEPAUL UNIVERSITY 36

Edit Widget Attributes

- Adjust layout
- Run

DEPAUL UNIVERSITY 37

Hello, iOS! Iteration #3: App Icons



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 *Spotlight 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

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

iPhone Screen Resolutions

	iPhone 3G iPhone 3GS	iPhone 4 iPhone 4S	iPhone 5 iPhone 5S iPhone SE	iPhone 6 iPhone 6S	iPhone 6+ iPhone 6S+ iPhone 7+/8+	iPhone X
Size (diagonal)	3.5"	3.5"	4"	4.7"	5.5"	5.8"
Display type	Classic	Retina	Retina	Retina (HD)	Retina HD True Tone	AMOLED HDR10
Pixels (px)	320×480	640×960	640×1136	750×1334	1080×1920	1125×2436
Density (ppi)	163	326	326	326	401	458
Points (pt)	320×480	320×480	320×568	375×667	414×736	375×812
Aspect ratio	3:2	3:2	~16:9	16:9	16:9	~19.5:9
Scale factor	1x	2x	2x	2x	3x	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

Add App Icons

- In the *Project Navigator*
 - Select *images.xcassets*
 - Select *AppIcon*



DEPAUL UNIVERSITY

43

Add App Icons

- You should provide images of sizes:
 - Home screen (60pt):
 - 120x120 (2x), 180x180 (3x)
 - Spotlight Search (40pt):
 - 80x80 (2x), 120x120 (3x)
 - Settings (29pt):
 - 58x58 (2x), 87x87 (3x)
 - Notification (20pt):
 - 40x40 (2x), 60x60 (3x)
- Use PNG format –
 - Portable Network Graphics format
- Also support vector PDF format

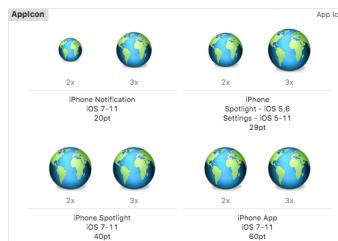


DEPAUL UNIVERSITY

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

Join the iOS Developer Program

- Enrolled students will be invited to the developer program
- Login to the iOS Developer Program
- Download and install Xcode 9

DEPAUL UNIVERSITY

49

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

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

Next ...

- Learn Swift programming language
- Dive into iOS SDK

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

DEPAUL UNIVERSITY

52