



ADVANCED iOS APPLICATION DEVELOPMENT

MPCS 51032 • SPRING 2020 • SESSION 4

EMBEDDED FRAMEWORKS

EMBEDDED FRAMEWORKS

- Frameworks
 - Code
 - Resources (images, sounds)
 - You can share them between projects
 - Working with pre/post build scripts, you can automate including them in various projects
- You might want to package your code the same way

EMBEDDED FRAMEWORKS

- Pre iOS8
 - Include static libraries, but the process was not straightforward
 - Frameworks were a matter of semantics
- iOS8 added ability to create true custom frameworks
 - Common code used between projects

EMBEDDED FRAMEWORKS

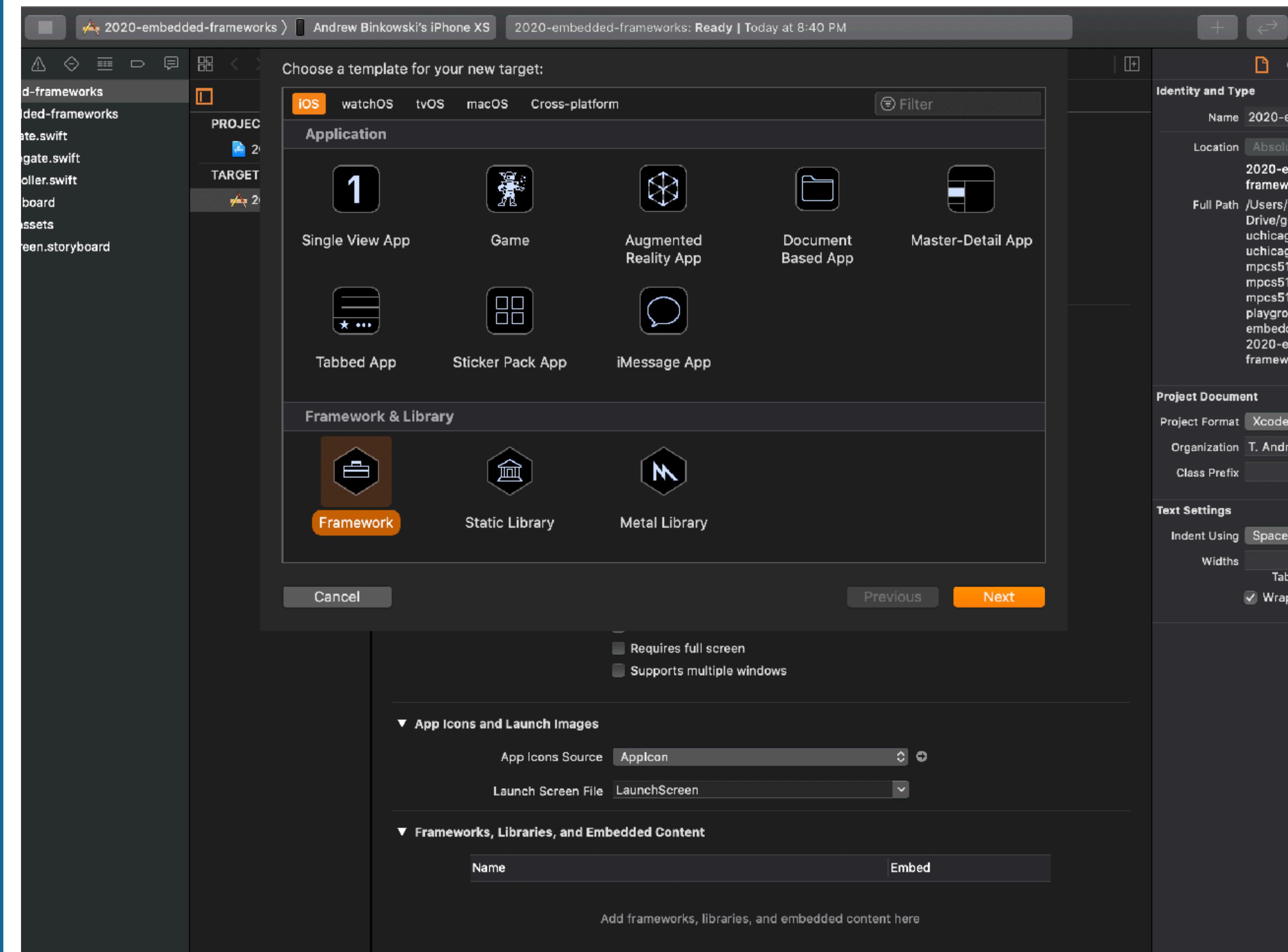
SUBTITLE

- Required if you want to “share” code within your own application
 - Extensions
 - Watch app



EMBEDDED FRAMEWORKS

- Create a new framework
 - Editor > Add Target



EMBEDDED FRAMEWORKS

Choose options for your new target:

Product Name: DataKit

Team: University of Chicago (Department of Co... ▾)

Organization Name: T. Andrew Binkowski

Organization Identifier: mobi.uchicago

Bundle Identifier: mobi.uchicago.DataKit

Language: Swift ▾

Include Unit Tests

Project: 2020-embedded-frameworks ▾

Embed in Application: 2020-embedded-frameworks ▾

Cancel Previous Finish

Identity and Type

Name 2020-embedded-frameworks

Location Absolute

2020-embedded-frameworks.xcodeproj

Full Path /Users/tabinkowski/GDrive/g-Teaching/uchicago.mobi/uchicago.mobi-courses/mpcs51032/mpcs51032-2020-spring/playground/session-4/embedded-frameworks/2020-embedded-frameworks.xcodeproj

Project Document

Project Format Xcode 9.3-compatible

Organization T. Andrew Binkowski

Class Prefix

Text Settings

Indent Using Spaces

Widths Tab 2

Wrap lines

EMBEDDED FRAMEWORKS

The screenshot shows the Xcode interface with the project "2017-Session3-EmbeddedFrameworks" selected. The "General" tab is active in the settings editor. In the left sidebar, under "TARGETS", there is a new target named "GreetingKit". A yellow callout bubble points to this target with the text "New target in file Navigator". The "Identity" section shows the display name "GreetingKit", bundle identifier "mobi.uchicago.GreetingKit", version "1.0", and build setting "\$(CURRENT_PROJECT_VERSION)". The "Signing" section has "Automatically manage signing" checked. The "Deployment Info" section shows a deployment target of "10.3" and devices set to "Universal". The right side of the screen displays the "Identity and Type" and "Project Document" panes.

2017-Session3-EmbeddedFrameworks > iPhone 7 Plus 2017-Session3-EmbeddedFrameworks: Ready | Today at 1:34 PM

2017-Session3-EmbeddedFrameworks

General Resource Tags Info Build Settings Build Phases Build Rules

PROJECT

2017-Session3-Em... Targets

Display Name GreetingKit

Bundle Identifier mobi.uchicago.GreetingKit

Version 1.0

Build \$(CURRENT_PROJECT_VERSION)

SIGNING

Automatically manage signing

Xcode will create and update profiles, app IDs, and certificates.

Team None

Provisioning Profile None Required

Signing Certificate Don't Code Sign

New target in file Navigator

Deployment Info

Deployment Target 10.3

Devices Universal

Identity and Type

Name 2017-Session3-EmbeddedFrameworks

Location Absolute

2017-Session3-EmbeddedFrameworks.xcodeproj

Full Path /Users/tabinkowski/Documents/Development/GitHub/uchicago-mobi/mcps51032-2017-spring-playground/2017-Session3-EmbeddedFrameworks/2017-Session3-EmbeddedFrameworks.xcodeproj

Project Document

Project Format Xcode 3.2-compatible

Organization T. Andrew Binkowski

Class Prefix

Text Settings

Indent Using Spaces

Widths 2 Tab 2 Indent

Wrap lines

App Extensions Allow app extension API only

EMBEDDED FRAMEWORKS

Linked to main project

+

-

Filter

▼ App Icons and Launch Images

App Icons Source AppIcon

Launch Images Source Use Asset Catalog...

Launch Screen File LaunchScreen

▼ Embedded Binaries

 DataKit.framework ...in build/Debug-iphoneos

+

-

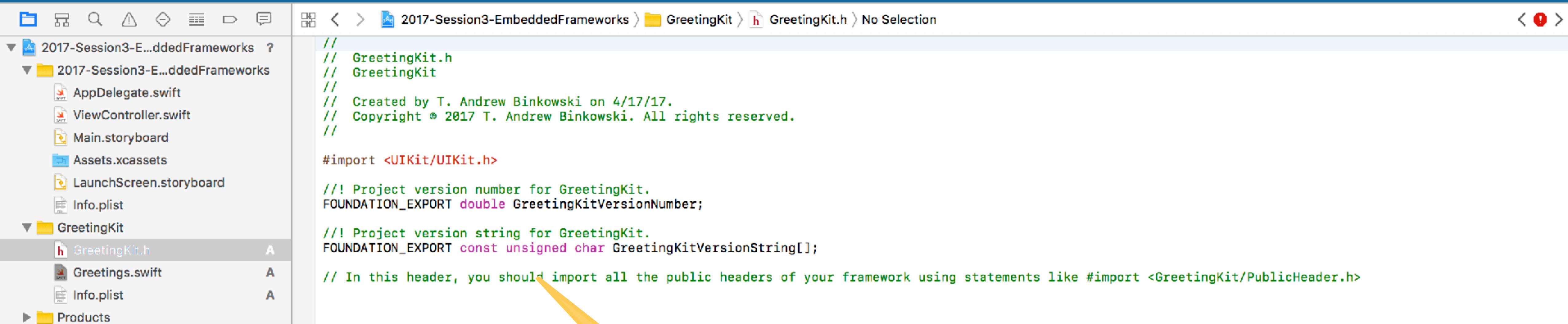
▼ Linked Frameworks and Libraries

Name	Status
 CloudKit.framework	Required ▲
 DataKit.framework	Required ▼

+

-

EMBEDDED FRAMEWORKS



The screenshot shows the Xcode interface with a project named "2017-Session3-EmbeddedFrameworks". The "GreetingKit" folder contains the "GreetingKit.h" file, which is currently selected. The code in the editor is as follows:

```
// GreetingKit.h
// GreetingKit
//
// Created by T. Andrew Binkowski on 4/17/17.
// Copyright © 2017 T. Andrew Binkowski. All rights reserved.
//

#import <UIKit/UIKit.h>

// Project version number for GreetingKit.
FOUNDATION_EXPORT double GreetingKitVersionNumber;

// Project version string for GreetingKit.
FOUNDATION_EXPORT const unsigned char GreetingKitVersionString[];

// In this header, you should import all the public headers of your framework using statements like #import <Framework/PublicHeader.h>
```

If you are using objective C or if you have any objective C files in your framework, then you will include all public headers of your framework here.

If you don't have objective C code in our framework, leave this file as is.

EMBEDDED FRAMEWORKS

- Not all Cocoa Touch APIs are available for use in extensions
- For instance extensions cannot do the following:
 - Access the camera or microphone on an iOS device
 - Receive data using AirDrop (It can however send data using AirDrop)
 - Perform long-running background tasks
 - Use any API marked in header files with the NS_EXTENSION_UNAVAILABLE macro, or similar unavailability macro, or any API in an unavailable framework for example EventKit and HealthKit are unavailable to app extensions.
 - Access a `sharedApplication` object, and so cannot use any of the methods on that object



There are different applications

EMBEDDED FRAMEWORKS

Team **None** 

Provisioning Profile **None Required**

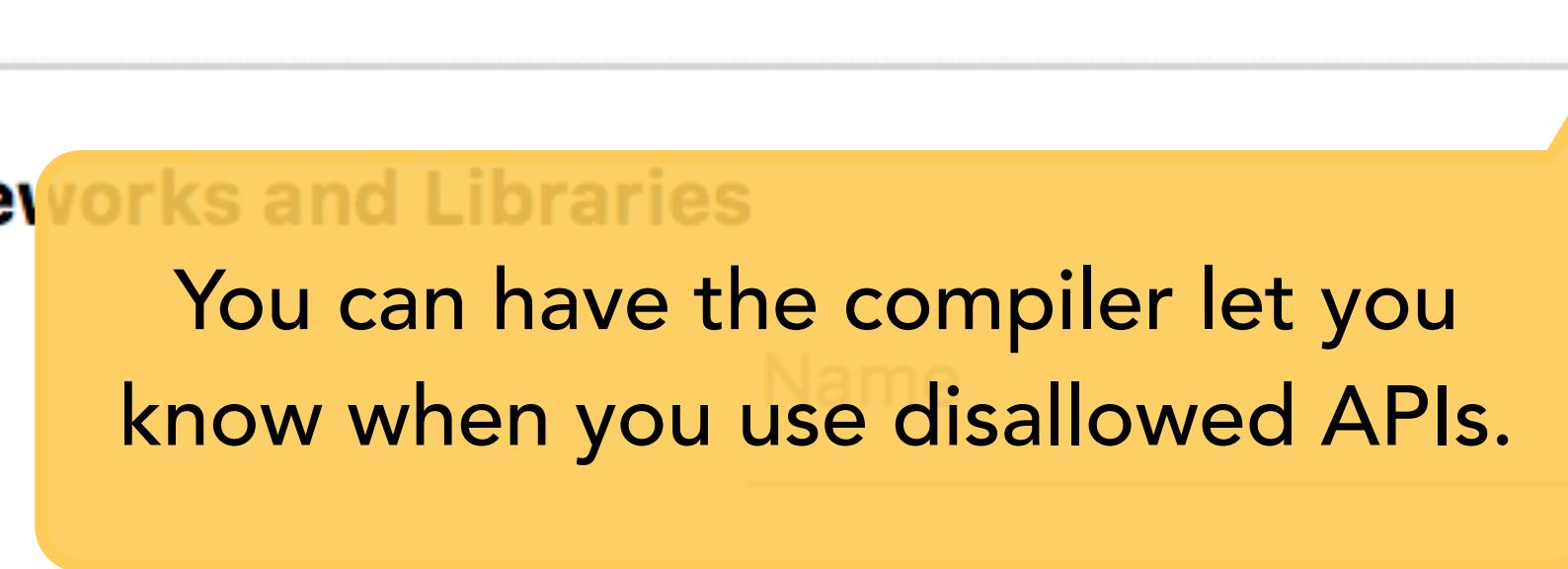
Signing Certificate **Don't Code Sign**

▼ Deployment Info

Deployment Target **10.3** 

Devices **Universal** 

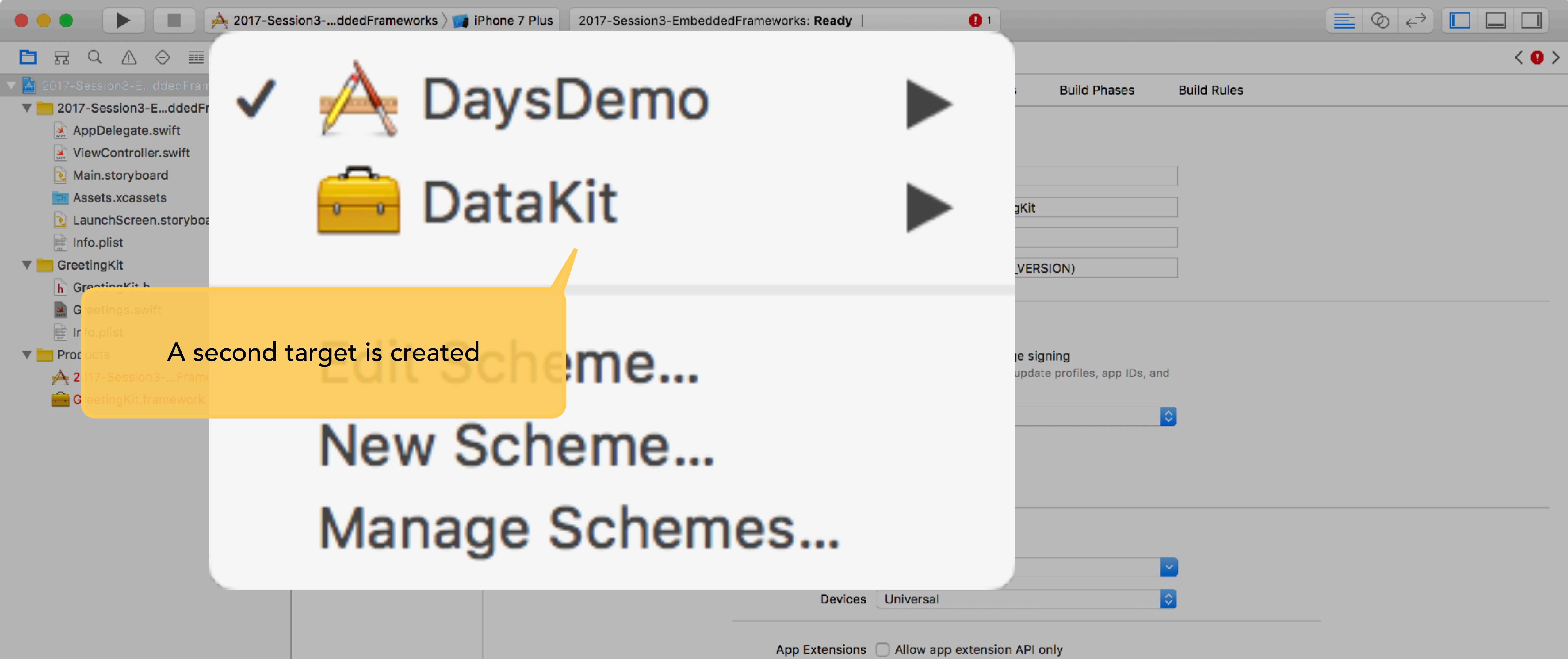
App Extensions Allow app extension API only

▼ Linked Frameworks and Libraries 

You can have the compiler let you know when you use disallowed APIs.

Name Status

EMBEDDED FRAMEWORKS



EMBEDDED FRAMEWORKS



The screenshot shows the Xcode interface with the project "2017-Session3-EmbeddedFrameworks" selected. The "GreetingKit" folder is open, and "Greetings.swift" is the active file. The code in "Greetings.swift" is as follows:

```
//  
//  Greetings.swift  
//  2017-Session3-EmbeddedFrameworks  
//  
//  Created by T. Andrew Binkowski on 4/17/17.  
//  Copyright © 2017 T. Andrew Binkowski. All rights reserved.  
  
import Foundation  
  
public class Greetings {  
  
    public init() {  
        // This initializer intentionally left empty  
    }  
  
    public func happy() {  
        print("😊")  
    }  
  
    public func crazy() {  
        print("😅")  
    }  
}
```

A yellow callout bubble with the text "Pay attention to access control" points to the "public" keyword in the "happy()" and "crazy()" functions.

EMBEDDED FRAMEWORKS

ACCESS CONTROL

- If your class is marked **private**
 - All variables, inits, and functions will default to private
- If your class is marked **internal (which is default), public, or open**
 - All variables, inits, and functions will default to internal
- A subclass's superclass must be at least as accessible
- Classes and class members declared public in Objective-C are imported into Swift as open
 - No such distinction in Objective-C

```
import Foundation

public class Greetings {

    public init() {
        // This initializer intentionally left empty
    }

    public func happy() {
        print("😊")
    }

    public func crazy() {
        print("😅")
    }
}
```

The default init is picking up the default internal because the last thing Swift wants to do is expose your init as public API unless it is explicitly instructed to do so.

EMBEDDED FRAMEWORKS

ACCESS CONTROL

- Open vs. Public

is a new access level introduced in Swift 3. It is more public than `public` but less public than `open`. It is used to implement protocols and make them [overridable](#).

available with the Swift 3 snapshot from August 7, 2016, and with Xcode 8 beta 6.

ort:

An `open` class is *accessible* and *subclassable* outside of the defining module. A member is *accessible* and *overridable* outside of the defining module.

A `public` class is *accessible* but *not subclassable* outside of the defining module. A class member is *accessible* but *not overridable* outside of the defining module.

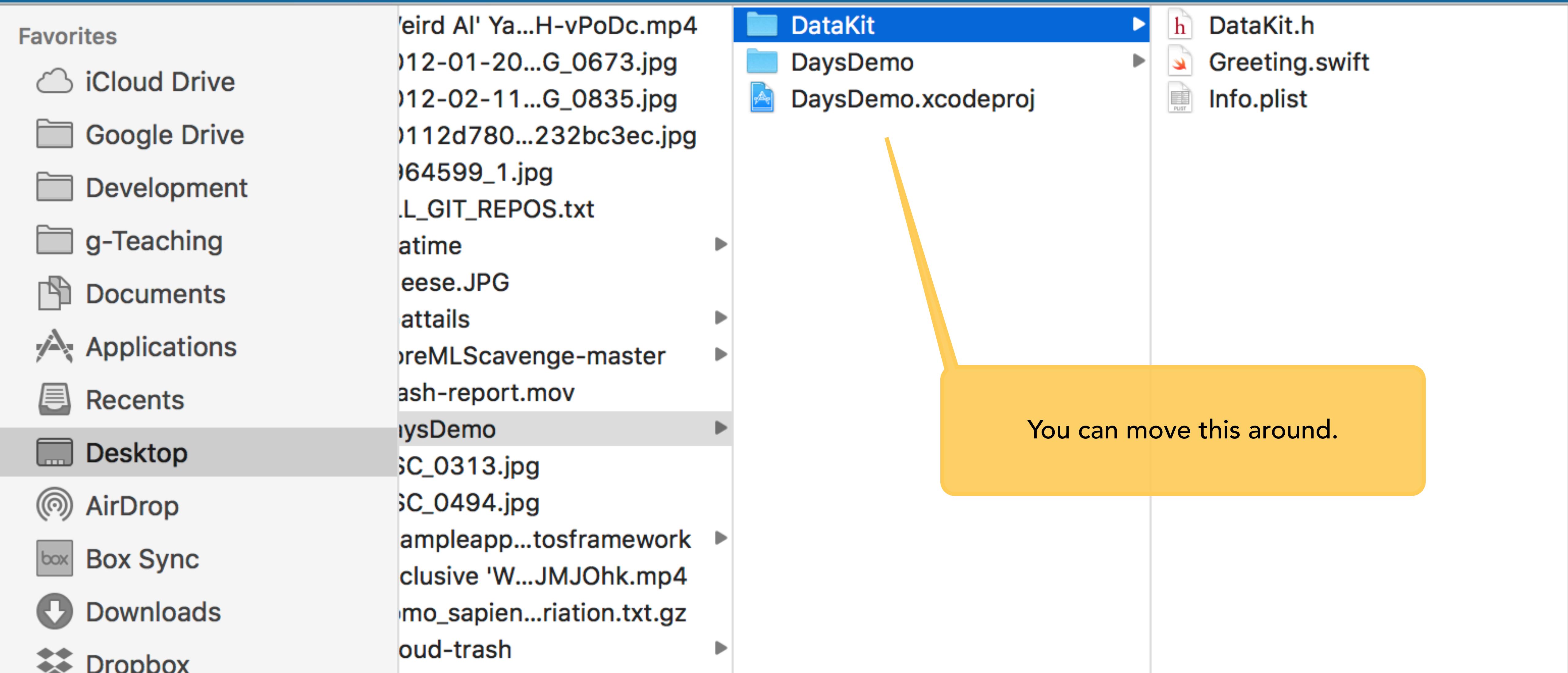
`open` is what `public` used to be in previous Swift releases and the access of `public` was restricted. Or, as Chris Lattner puts it in [SE-0177: Allow distinguishing between public and open](#):

“`open`” is now simply “more public than `public`”, providing a very simple and clean model.

For example, `open var hashCode` is a property which is accessible and can be overridden by subclasses.

Open is more public than
public

EMBEDDED FRAMEWORKS



EMBEDDED FRAMEWORKS

- Playground
 - Embedded Framework



2017-Session3-EmbeddedFrameworks > iPhone X Running 2018-Session4-EmbeddedFrameworks on iPhone X 4

2018-Session4-EmbeddedFrameworks > GreetingKit > Greetings.swift > No Selection

```
1 //  
2 //  Greetings.swift  
3 //  2017-Session3-EmbeddedFrameworks  
4 //  
5 //  Created by T. Andrew Binkowski on 4/17/17.  
6 //  Copyright © 2017 T. Andrew Binkowski. All rights reserved.  
7 //  
8  
9 import Foundation  
10  
11 public class Greetings {  
12  
13     public init() {  
14         // This initializer intentionally left empty  
15     }  
16  
17     public func happy() {  
18         print("😊")  
19     }  
20  
21     public func crazy() {  
22         print("😎")  
23     }  
24  
25 }  
26
```

2018-Session4-EmbeddedFrameworks

Applications/Xcode.app/Contents/Develo



ADVANCED iOS APPLICATION DEVELOPMENT

MPCS 51032 • SPRING 2020 • SESSION 4