

Featured

#WWDC15

What's New in Xcode

Session 104

Ken Orr Developer Tools







Swift 2



Error handling

Availability

Testability

Protocol Extensions

Swift Migrator

```
override public func readFromData(data: NSData,
ofType typeName: String) throws {
    unarchivedList = NSKeyedUnarchiver.
        unarchiveObjectWithData(data) as? List

    if let unarchivedList = unarchivedList {
        listPresenter?.setList(unarchivedList)
    }

    return
}
```

```
throw NSError(domain: NSCocoaErrorDomain,
code: NSFileReadCorruptFileError,
userInfo: [
    NSLocalizedDescriptionKey:
        NSLocalizedString("Could not read
file.", comment: "Read error
description"),
    NSLocalizedFailureReasonErrorKey:
        NSLocalizedString("File was in an
invalid format.", comment: "Read
failure reason")
])
}
```

```
override public func dataOfType(typeName: String)
throws -> NSData {
    let outError: NSError! = NSError(domain:
        "Migrator", code: 0, userInfo: nil)
    if let archiveableList = listPresenter?.
        archiveableList {
        return NSKeyedArchiver.
            archivedDataWithRootObject
            (archiveableList)
    }
}
```

```
override public func readFromData(data: NSData,
ofType typeName: String, error outError:
NSErrorPointer) -> Bool {
    unarchivedList = NSKeyedUnarchiver.
        unarchiveObjectWithData(data) as? List

    if let unarchivedList = unarchivedList {
        listPresenter?.setList(unarchivedList)
    }

    return true
}
```

```
if outError != nil {
    outError.memory = NSError(domain:
        NSCocoaErrorDomain, code:
        NSFileReadCorruptFileError, userInfo:
        [
            NSLocalizedDescriptionKey:
                NSLocalizedString("Could not read
file.", comment: "Read error
description"),
            NSLocalizedFailureReasonErrorKey:
                NSLocalizedString("File was in an
invalid format.", comment: "Read
failure reason")
        ])
}

return false
}
```

```
override public func dataOfType(typeName: String,
error outError: NSErrorPointer) -> NSData? {
    if let archiveableList = listPresenter?.
        archiveableList {
        return NSKeyedArchiver.

```






Playgrounds

Ready | Today at 4:12 PM

Mandelbrot

Mandelbrot Set Rendering

This playground renders the Mandelbrot set by running code embedded within the playground’s Sources folder. You can explore the Sources folder by opening the Project Navigator (CMD+1) and clicking the triangle next to the playground file.

```
// Create a Mandelbrot set
let m = MandelbrotView(frame: rect)

// Create “Seahorse Valley”
m.mandelbrotRect = newMandelbrotRect(points[“b”]!)

// Create Mandelbrot set within the Mandelbrot set
m.mandelbrotRect = newMandelbrotRect(points[“e”]!)
```

Mandelbrot_Source...

- 10 sec +

Ready | Today at 4:12 PM

Mandelbrot

Mandelbrot Set Rendering

This playground renders the Mandelbrot set by running code embedded within the playground’s Sources folder. You can explore the Sources folder by opening the Project Navigator (CMD+1) and clicking the triangle next to the playground file.

```
// Create a Mandelbrot set
let m = MandelbrotView(frame: rect)

// Create “Seahorse Valley”
m.mandelbrotRect = newMandelbrotRect(points[“b”]!)

// Create Mandelbrot set within the Mandelbrot set
m.mandelbrotRect = newMandelbrotRect(points[“e”]!)
```

Mandelbrot_Source...

- 10 sec +

Ready | Today at 4:12 PM

Mandelbrot

Mandelbrot Set Rendering

This playground renders the Mandelbrot set by running code embedded within the playground’s Sources folder. You can explore the Sources folder by opening the Project Navigator (CMD+1) and clicking the triangle next to the playground file.

```
// Create a Mandelbrot set
let m = MandelbrotView(frame: rect)

// Create “Seahorse Valley”
m.mandelbrotRect = newMandelbrotRect(points[“b”]!)

// Create Mandelbrot set within the Mandelbrot set
m.mandelbrotRect = newMandelbrotRect(points[“e”]!)
```

Mandelbrot_Source...

- 10 sec +

Ready | Today at 4:12 PM

Mandelbrot

Mandelbrot Set Rendering

This playground renders the Mandelbrot set by running code embedded within the playground’s Sources folder. You can explore the Sources folder by opening the Project Navigator (CMD+1) and clicking the triangle next to the playground file.

```
// Create a Mandelbrot set
let m = MandelbrotView(frame: rect)

// Create “Seahorse Valley”
m.mandelbrotRect = newMandelbrotRect(points[“b”]!)

// Create Mandelbrot set within the Mandelbrot set
m.mandelbrotRect = newMandelbrotRect(points[“e”]!)
```

Mandelbrot_Source...

- 10 sec +

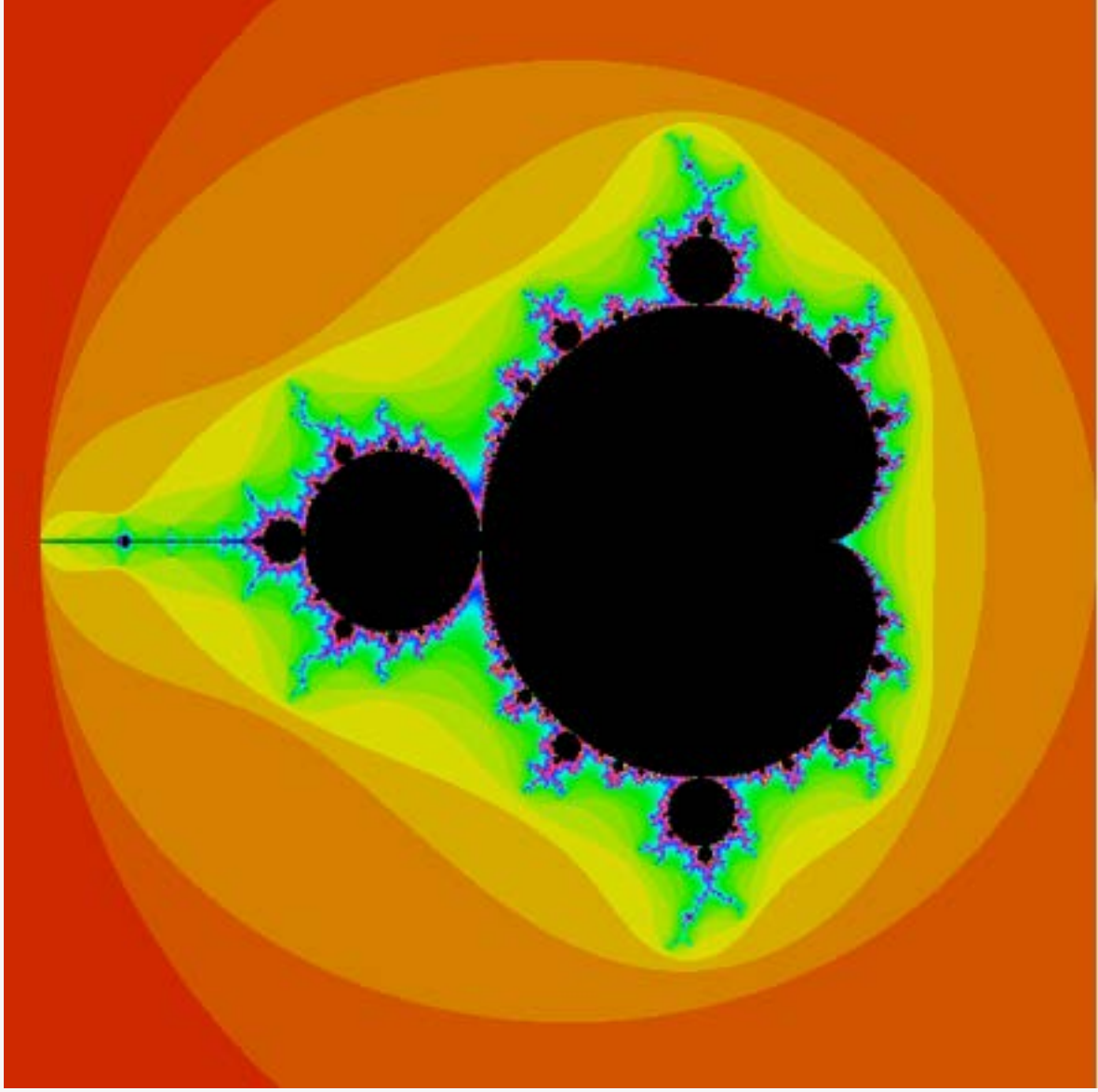
Ready | Today at 4:12 PM

Mandelbrot

Mandelbrot Set Rendering

This playground renders the Mandelbrot set by running code embedded within the playground's Sources folder. You can explore the Sources folder by opening the Project Navigator (CMD+1) and clicking the triangle next to the playground file.

```
// Create a Mandelbrot set
let m = MandelbrotView(frame: rect)
```



```
// Create "Seahorse Valley"
m.mandelbrotRect = newMandelbrotRect(points["b"]!)

// Create Mandelbrot set within the Mandelbrot set
m.mandelbrotRect = newMandelbrotRect(points["e"]!)
```

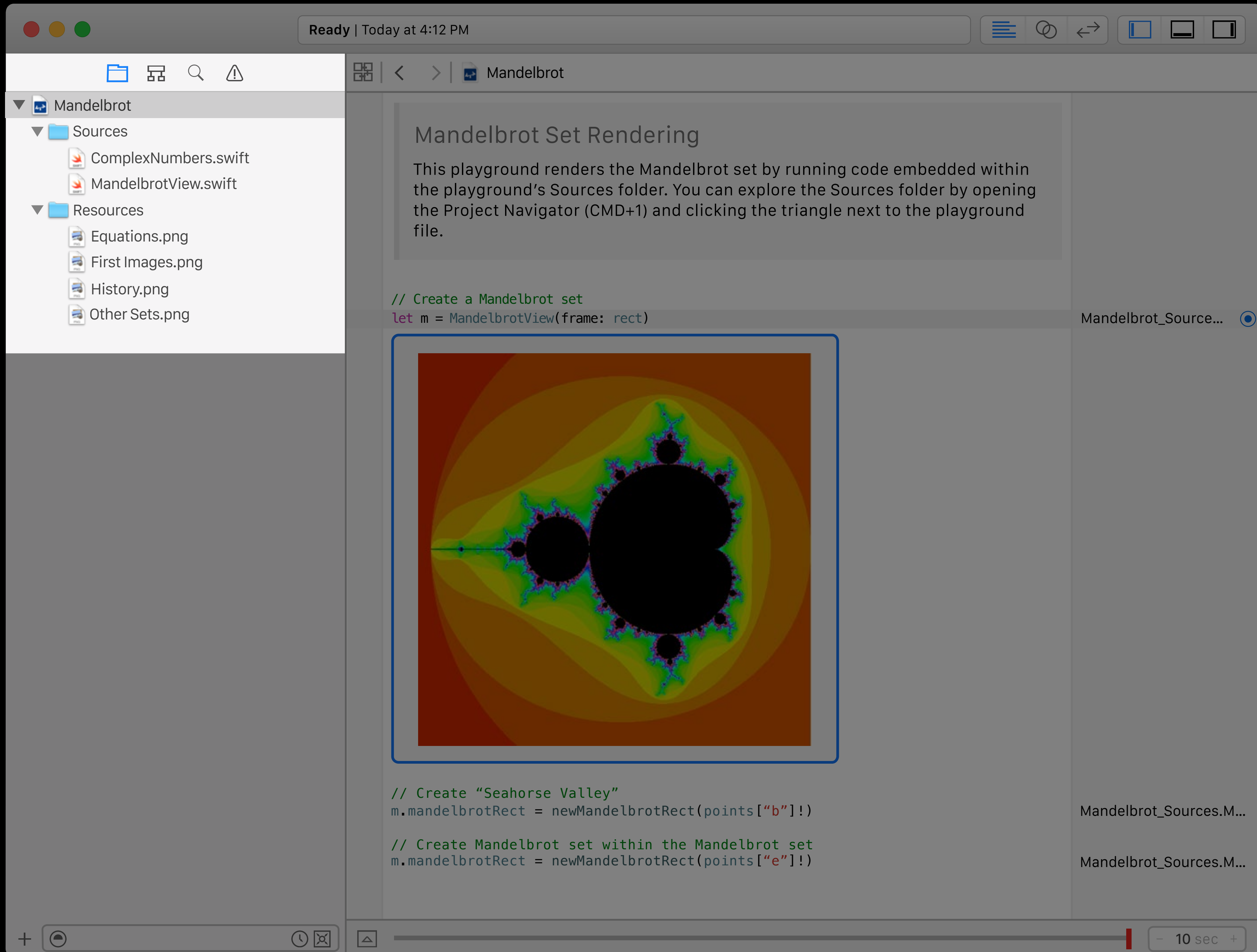
Mandelbrot_Source...

Mandelbrot_Sources.M...

Mandelbrot_Sources.M...

- 10 sec +

The image is a screenshot of an Xcode playground window titled "Mandelbrot". The top status bar shows "Ready | Today at 4:12 PM" and standard window controls. The left sidebar contains a project navigator with a folder named "Mandelbrot". Inside this folder, there are two subfolders: "Sources" and "Resources". The "Sources" folder contains two Swift files: "ComplexNumbers.swift" and "MandelbrotView.swift". The "Resources" folder contains four image files: "Equations.png", "First Images.png", "History.png", and "Other Sets.png". The main editor area displays Swift code. At the top, there is a text box with the title "Mandelbrot Set Rendering" and a paragraph explaining that the playground renders the Mandelbrot set by running code in the Sources folder. Below this, the code starts with a comment "// Create a Mandelbrot set" followed by the line "let m = MandelbrotView(frame: rect)". This line is highlighted with a blue selection bar. Below the code, there is a large rectangular image of a Mandelbrot set, which is a complex fractal with a central black region and colorful, intricate patterns. Below the image, there is more code. It starts with a comment "// Create 'Seahorse Valley'", followed by "m.mandelbrotRect = newMandelbrotRect(points["b"]!)", and then another comment "// Create Mandelbrot set within the Mandelbrot set" followed by "m.mandelbrotRect = newMandelbrotRect(points["e"]!)". To the right of the code editor, there is a sidebar with two entries: "Mandelbrot_Source..." and "Mandelbrot_Sources.M...", each with a blue circular icon. The bottom of the window features a toolbar with various icons, including a magnifying glass, a play button, and a timeline slider set to "10 sec".



Ready | Today at 4:12 PM

GuidedTour

Control Flow

Simple Values

Control Flow

Functions and Closures

Objects and Classes

Enumerations and Structures

Protocols and Extensions

Generics

Sources

Resources

```
case let x where x.hasSuffix("pepper"):
    let vegetableComment = "Is it a spicy \ \(x)?"
default:
    let vegetableComment = "Everything tastes good in soup."
}
```

Experiment: Try removing the default case. What error do you get?

Notice how `let` can be used in a pattern to assign the value that matched that part of a pattern to a constant.

After executing the code inside the switch case that matched, the program exits from the switch statement. Execution doesn't continue to the next case, so there is no need to explicitly break out the switch at the end of each case's code.

You use `for-in` to iterate over item in a dictionary by providing a pair of names to use for each key-value pair. Dictionaries are an unordered collection, so their keys and values are iterated over in an arbitrary order.

```
let interestingNumbers = [
    "Prime": [2, 3, 5, 7, 11, 13],
    "Fibonacci": [1, 1, 2, 3, 5, 8],
    "Square": [1, 4, 9, 16, 25],
]
var largest = 0
for (kind, numbers) in interestingNumbers {
    for number in numbers {
        if number > largest {
            largest = number
        }
    }
}
```

Is it a spicy red pepper?

[Prime: [2, 3, 5, 7, 11, 13]

0

(8 times)

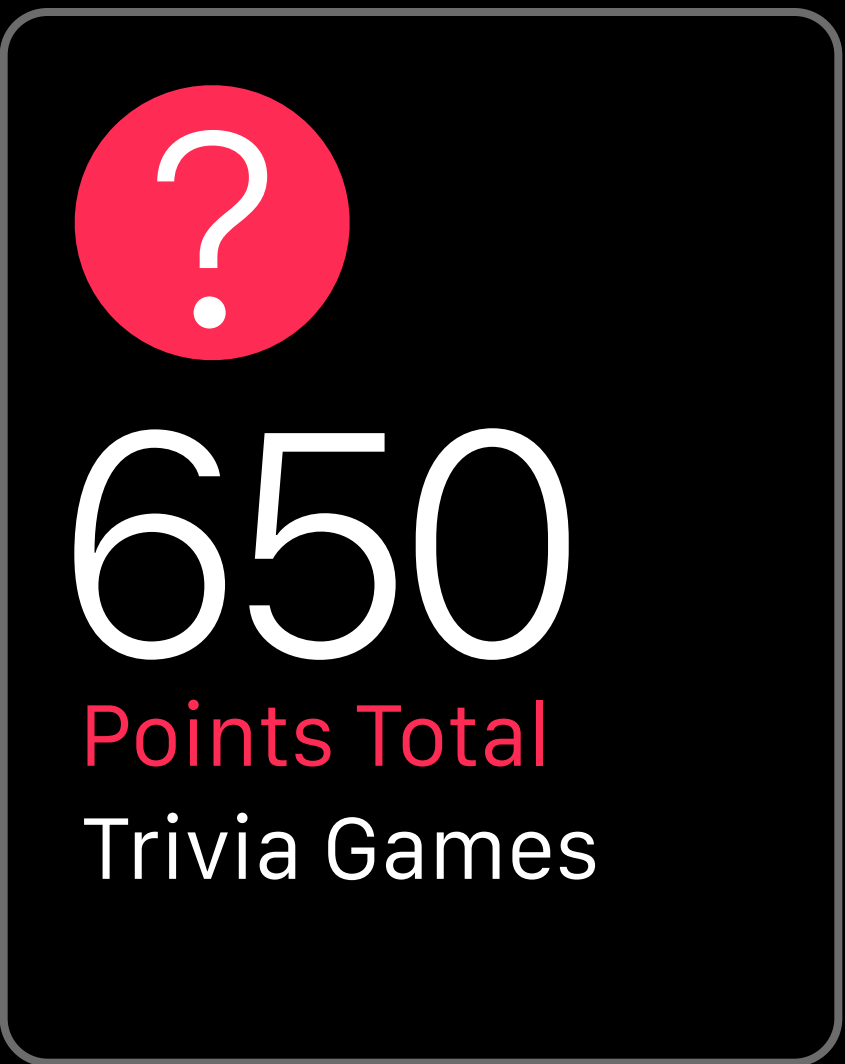
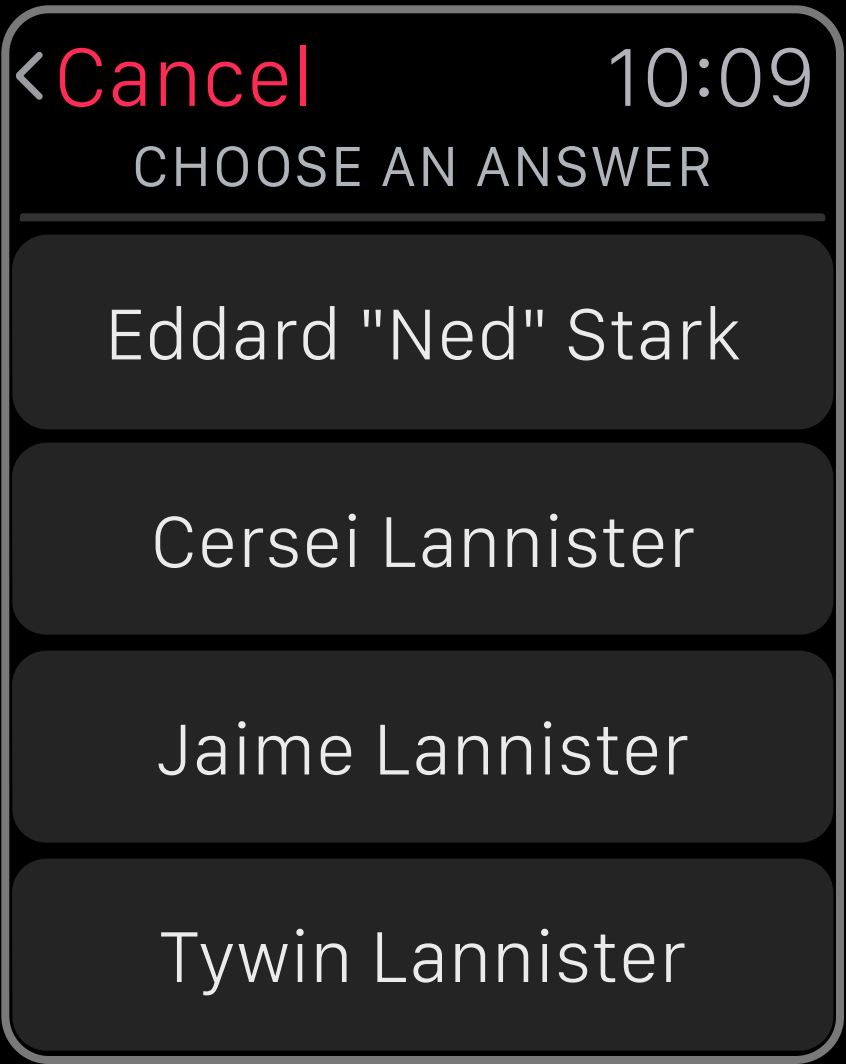
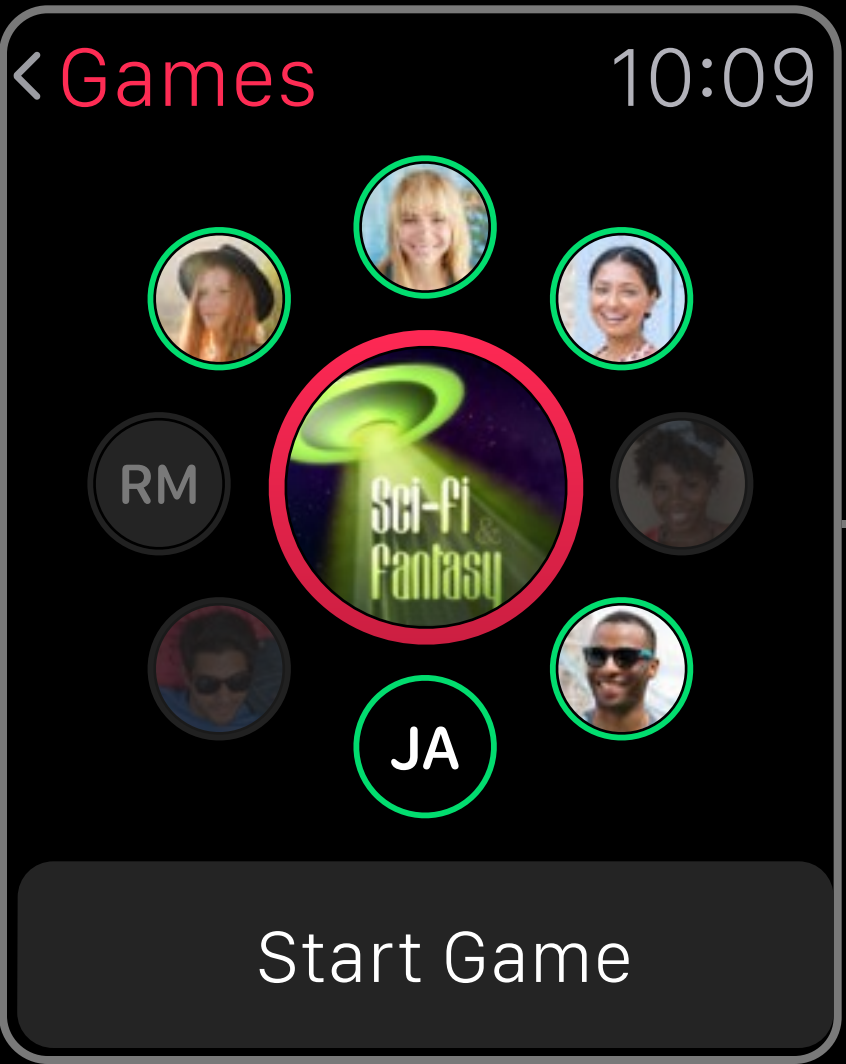
30 sec







watchOS 2



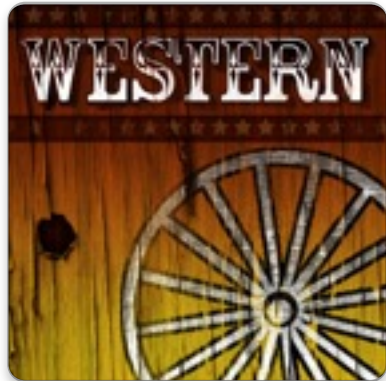




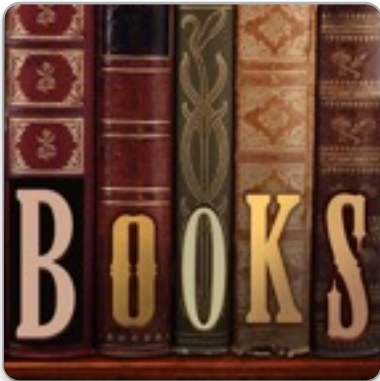
TUE 9 10:09

Trivia
Sci-fi & Fantasy
In Progress...

26:50 AAPL 131.39



Western Movies



Classic Books



Drama Movies



Thriller Movies



Engineering



Classical Music



Business Concepts



Science



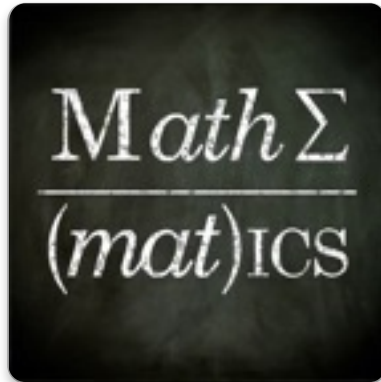
Sci-Fi & Fantasy



Sports History



World History



Mathematics



Horror Movies



Ancient History



Reality TV



Health & Medicine



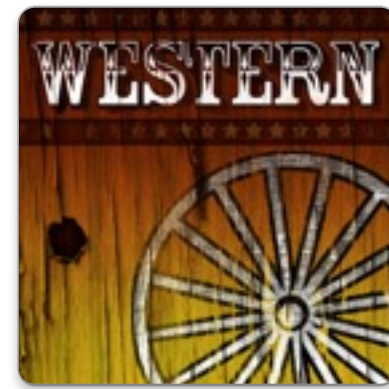
9:41 AM

100%

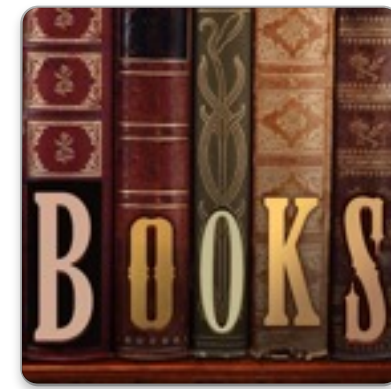
< Games

Choose Genre

Next



Western Movies



Classic Books



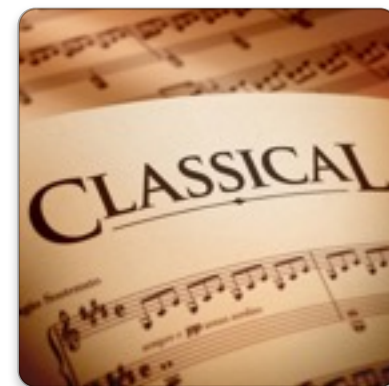
Drama Movies



Thriller Movies



Engineering



Classical Music



Business Concepts



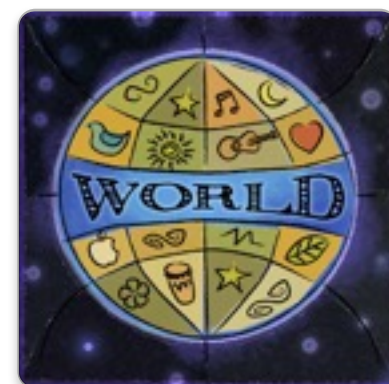
Science



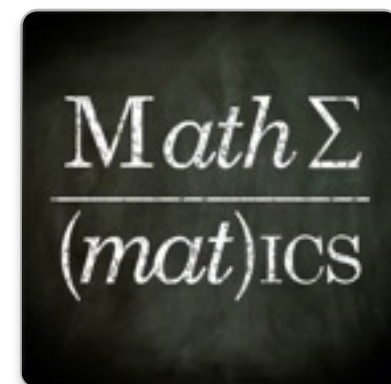
Sci-Fi & Fantasy



Sports History



World History



Mathematics



Horror Movies



Ancient History




Reality TV

9:41 AM100%

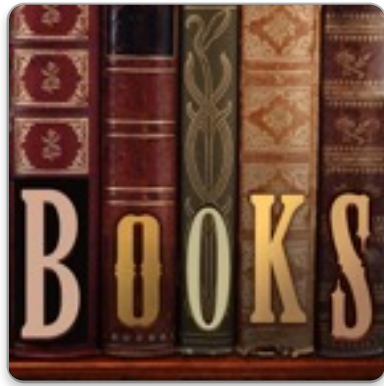
< Games

Choose Genre


Next




Western Movies




Classic Books



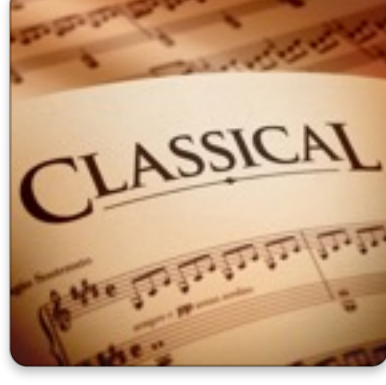
Drama Movies




Thriller Movies




Engineering




Classical Music



Business Concepts



Science



Sci-Fi & Fantasy

< Messages

Friends

Details

Amy Smith

Anyone up for a game of Trivia this morning?

That sounds great Amy. I can kick-off the game. Which genre should I choose?

Johnny Appleseed

I am at WWDC watching the "What's New in Xcode" session.

Angelia O'Neil

Yes!

Sci-fi & Fantasy sound good?

Ryan Davidson

I am in! Sci-fi & Fantasy is my favorite trivia genre. Let's play.

Debbie Brown

Invite me.

I've got the game running and I am going to invite all of you to play.

 iMessage 



App Thinning

Bitcode



Bitcode



Bitcode

Slicing



Bitcode



Slicing



Bitcode



Slicing



On Demand
Resources

Demo

Jon Hess Developer Tools

Demo

Implementing UI Designs in Interface Builder

Pacific Heights

Wednesday 1:30PM

App Thinning in Xcode

Presidio

Wednesday 9:00AM

Introducing On Demand Resources

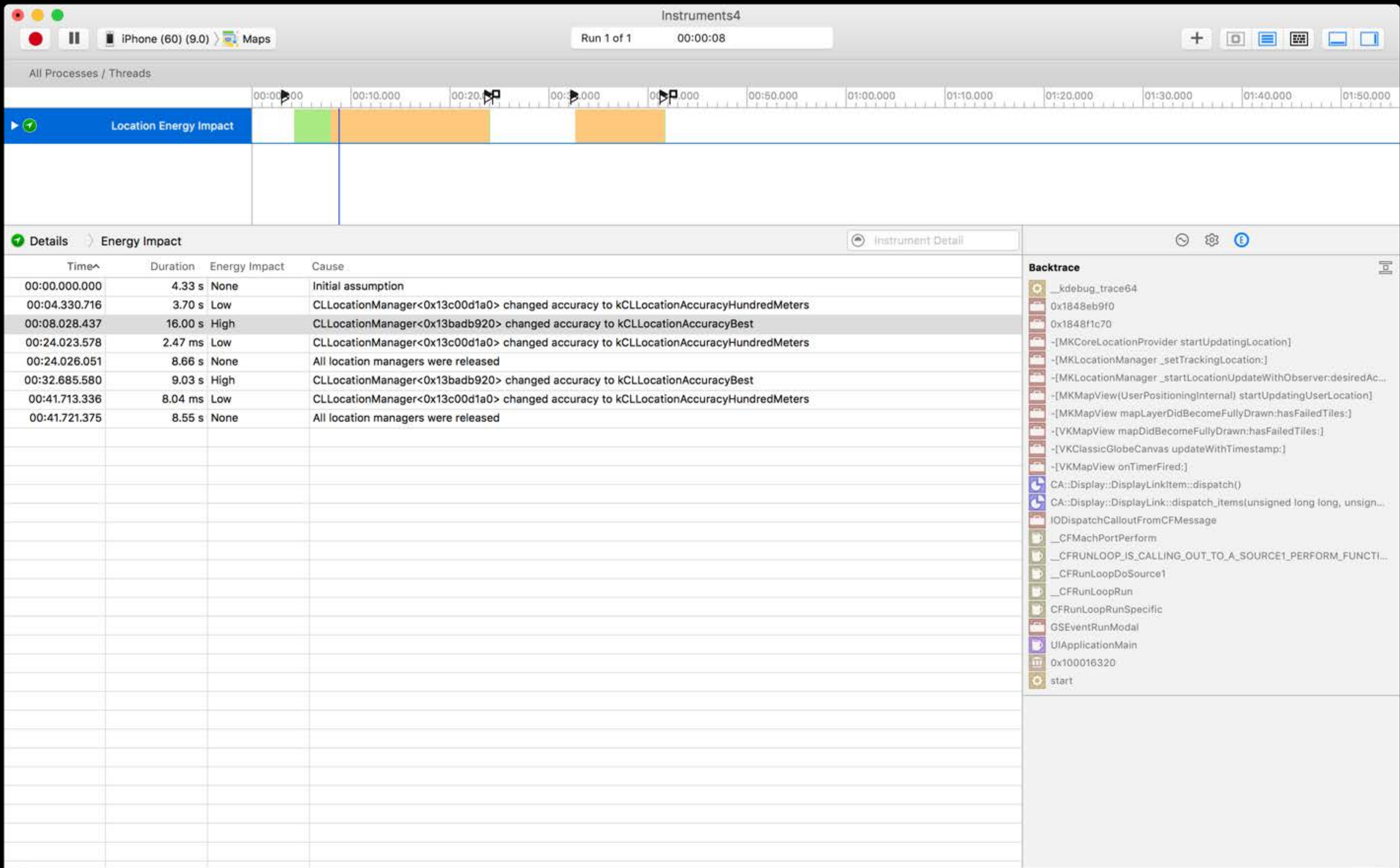
Pacific Heights

Wednesday 4:30PM

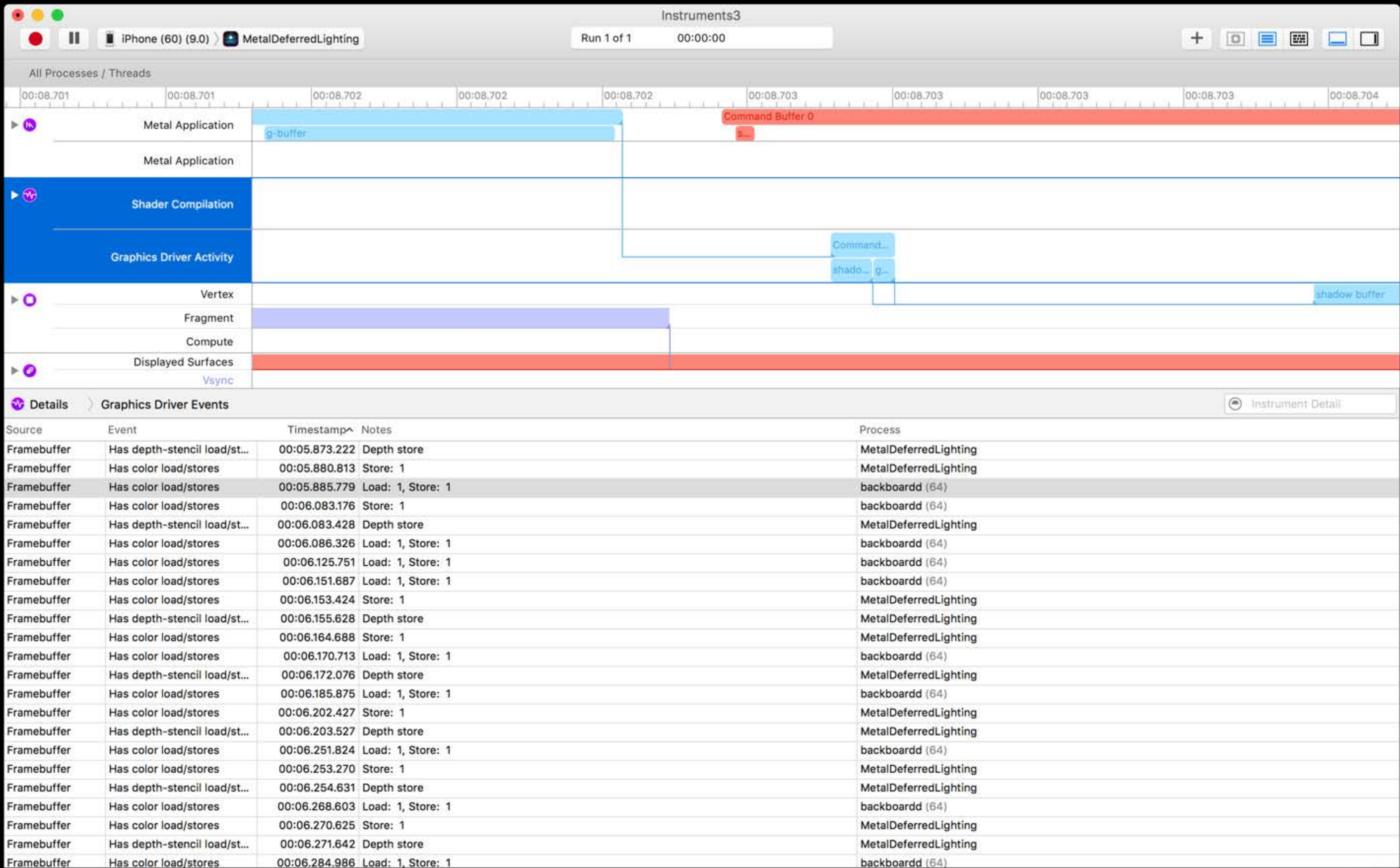


Debugging and Profiling Tools

Location Instrument



Metal System Trace





Address Sanitizer

TestASan PID 81755

Thread 1 Queue: com.ap...ain-thread (serial)

0 __asan::AsanDie()

5 __asan_report_store8

6 -[AppDelegate applicationDidFinishLa...

7 __CFNOTIFICATIONCENTER_IS_CAL...

23 NSApplicationMain

24 main

25 start

26 start

Thread 2 Queue: com.ap...h-manager (serial)

Thread 3

Thread 4

Thread 5

Thread 6

Memory

1 byte inside a 8-byte heap region at 0x...

Memory allocated at (1)

0 wrap_malloc

1 -[DrawManager init]

2 -[AppDelegate applicationDidFin...

3 __CFNOTIFICATIONCENTER_IS...

19 NSApplicationMain

20 main

21 tlv_get_addr

22 0x00000002

Memory deallocated at (1)

0 wrap_free

1 -[DrawManager calculateAxes]

2 -[AppDelegate applicationDidFin...

3 __CFNOTIFICATIONCENTER_IS...

19 NSApplicationMain

20 main

21 tlv_get_addr

22 0x00000002

TestASan > TestASan > AppDelegate.m > No Selection

1 //

2 // AppDelegate.m

3 // TestASan

4 //

5 // Copyright © 2015 Apple. All rights reserved.

6 //

7

8 #import "AppDelegate.h"

9 #import "DrawManager.h"

10

11 @interface AppDelegate ()

12

13 @property (weak) IBOutlet UIWindow *window;

14 @end

15

16 @implementation AppDelegate

17

18 - (void)applicationDidFinishLaunching:(NSNotification *)aNotification

19 {

20 DrawManager *manager = [[DrawManager alloc] init];

21 [manager calculateAxes];

22

23 NSInteger *x = manager.xAxis;

24 *x = 7;

25

26 [manager invalidateLayout];

27 }

28

29 - (void)applicationWillTerminate:(NSNotification *)aNotification {

30 // Insert code here to tear down your application

31 }

32

33 @end

34

Thread 1: Use of deallocated memory detected

TestASan > Thread 1 > 6 -[AppDelegate applicationDidFinishLaunching:]

self = (AppDelegate *) 0x6020000007f0

_cmd = (SEL) "applicationDidFinishLaunching:"

aNotification = (NSConcreteNotification *) @"NSApplicationDidFinish...

manager = (DrawManager *) 0x6020000101d0

x = (NSInteger *) 0x7fff5fbfe450

Global init order: f6

Poisoned by user: f7

Container overflow: fc

Array cookie: ac

Intra object redzone: bb

ASan internal: fe

Left alloca redzone: ca

Right alloca redzone: cb

==81755==ABORTING

AddressSanitizer report breakpoint hit. Use 'thread info -s' to get extended information about the report.

(lldb)

Local

All Output

Demo

Kate Stone Developer Tools

Demo

Advanced Debugging and the Address Sanitizer

Mission

Friday 9:00AM

What's New in LLDB

Nob Hill

Tuesday 2:30PM



Crash Logs



Fully Symbolicated
App Store and Test Flight
Full Xcode Integration

ArchivesCrashes

App Store1.0 Release

Memory issue when logging

2301 devices

iOS Apps

Adventure

PhotoAlbum

Recipes

Trivia

OS X Apps

Adventure

Recipes

✓

Splash Screen array out of b...

2379 devices

✓

Filtering leads to out of boun...

2227 devices

✓

Trivia: Trivia_GamePlayerPick...

2003 devices

✓

Crashed while filtering playe...

1883 devices

✓

Hard to reproduce crash whe...

1820 devices

Assertion in image loading

1522 devices

Crash when mashing answer...

1226 devices

Assertion when multiple play...

1122 devices

✓

Blew the stack when loading...

1087 devices

Trivia: Trivia_GamePlayerPick...

815 devices

Crash when inviting too man...

735 devices

○

Memory issue when logging

▼ Thread 0

0

libsystem_kernel...

__pthread_kill

1

libsystem_pthrea...

pthread_kill

2

libsystem_c.dylib

abort

3

libsystem_malloc...

free

4

UsageStatistics

-[UsageLog endLogging]

5

Trivia

Trivia.GameNavigationControllerDelegate.navigationController (navigationCont...

6

Trivia

@objc @objc Trivia.GameNavigationControllerDelegate.navigationController (T...

7

UIKit

-[UINavigationController navigationTransitionView:didEndTransition:fromView:t...

8

UIKit

-[UINavigationController _startCustomTransition:]_block_invoke

9

UIKit

-[__UIViewControllerTransitionContext completeTransition:]

10

UIKit

-[__UINavigationControllerParallaxTransition animateTransition:]_block_invoke93

11

UIKit

-[UIViewAnimationBlockDelegate _didEndBlockAnimation:finished:context:]

12

UIKit

-[UIViewAnimationState sendDelegateAnimationDidStop:finished:]

13

UIKit

-[UILayoutContainerView layoutSubviews]

14

UIKit

-[UIView(CALayerDelegate) layoutSublayersOfLayer:]

15

QuartzCore

-[CALayer layoutSublayers]

16

QuartzCore

CA::Layer::layout_if_needed(CA::Transaction*)

17

QuartzCore

CA::Layer::layout_and_display_if_needed(CA::Transaction*)

18

QuartzCore

CA::Context::commit_transaction(CA::Transaction*)

19

QuartzCore

CA::Transaction::commit()

20

QuartzCore

CA::Transaction::observer_callback(__CFRunLoopObserver*, unsigned long, voi...

21

CoreFoundation

__CFRUNLOOP_IS_CALLING_OUT_TO_AN_OBSERVER_CALLBACK_FUNCTION__

22

CoreFoundation

__CFRunLoopDoObservers

23

CoreFoundation

CFRunLoopRunSpecific

24

UIKit

-[UIApplication _run]

25

UIKit

UIApplicationMain

26

Trivia

main

Crash Information

Memory issue when logging

2301 devices

Open in Project...

Crash Log Details

Binary

Trivia 2.0 (55)

Thread

0

iOS Version

9.0 (13A4254n)

Device

iPhone 6 Plus

Notes

We've been unable to reproduce this crash, but several users have reported it.

Last 2 Weeks

Operating System

2301

TOTAL

IOS 8.3

42%

IOS 8.4

36%

IOS 9.0

23%

05/26 05/28 05/30 06/01 06/03 06/05 06/07

Mark As Resolved

< 1 of 7 >

Demo

Itai Rom Developer Tools



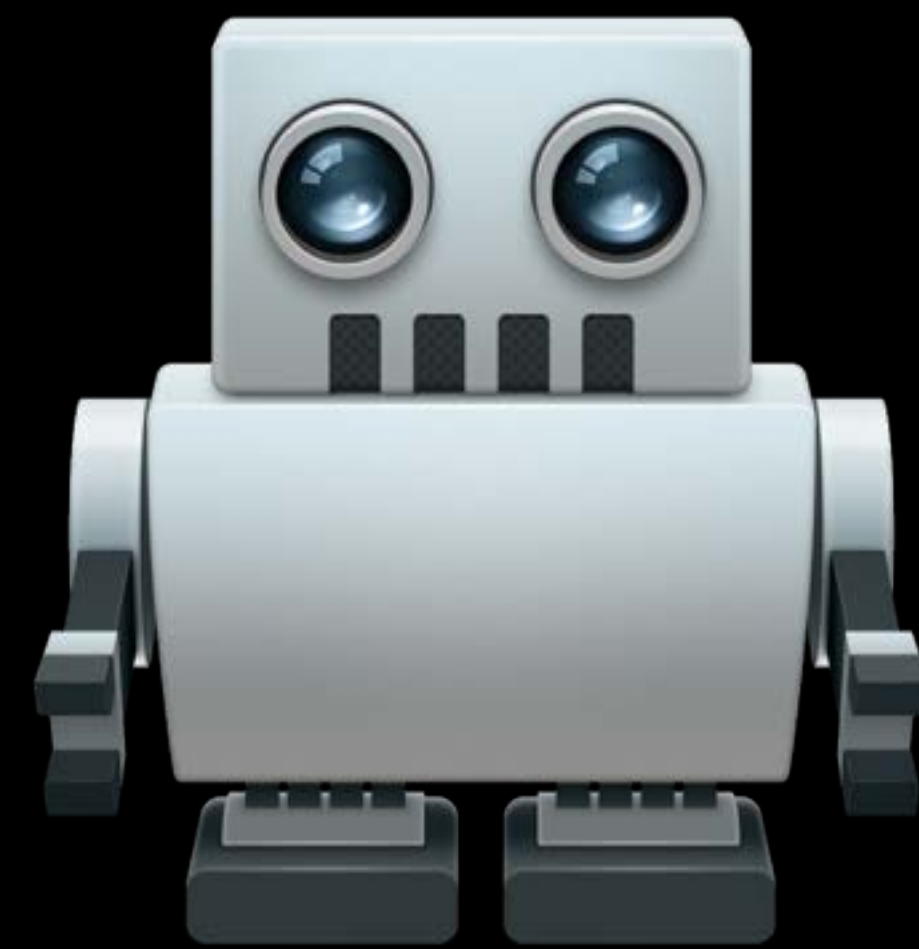
Testing



Test Navigator



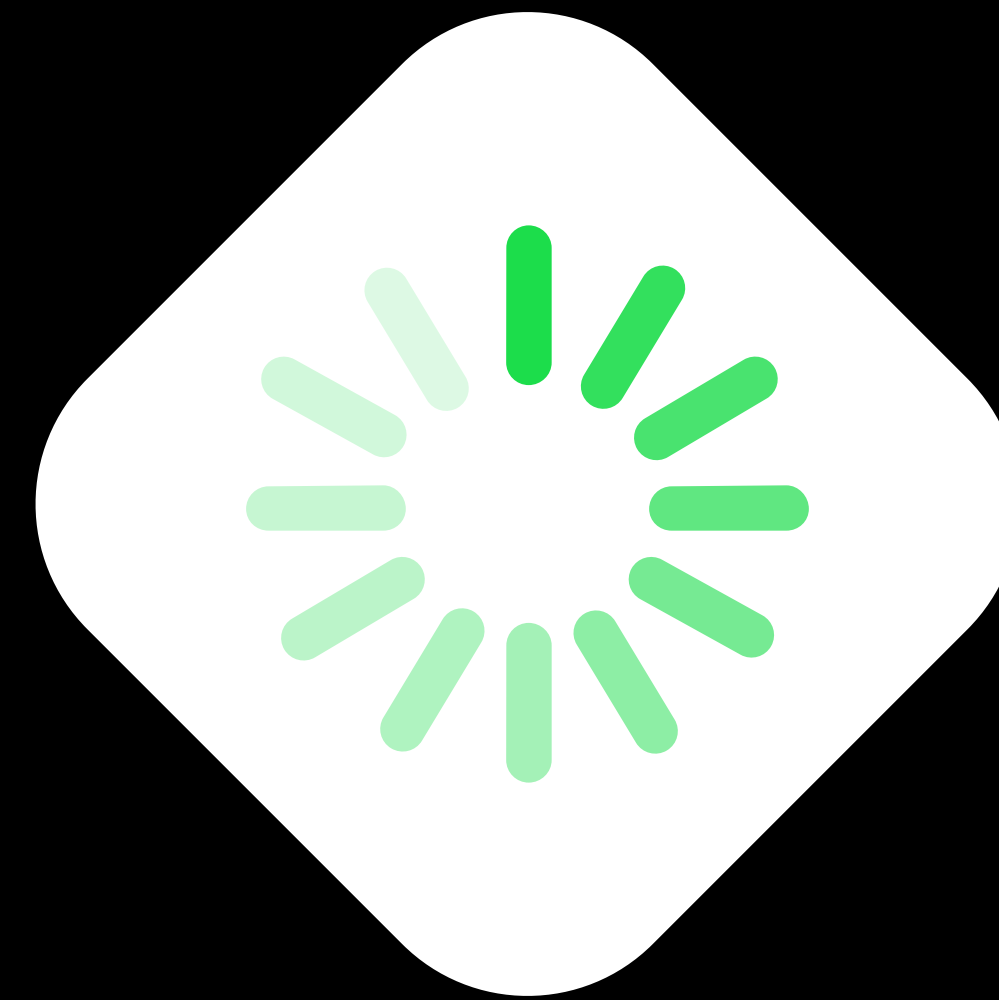
XCTest



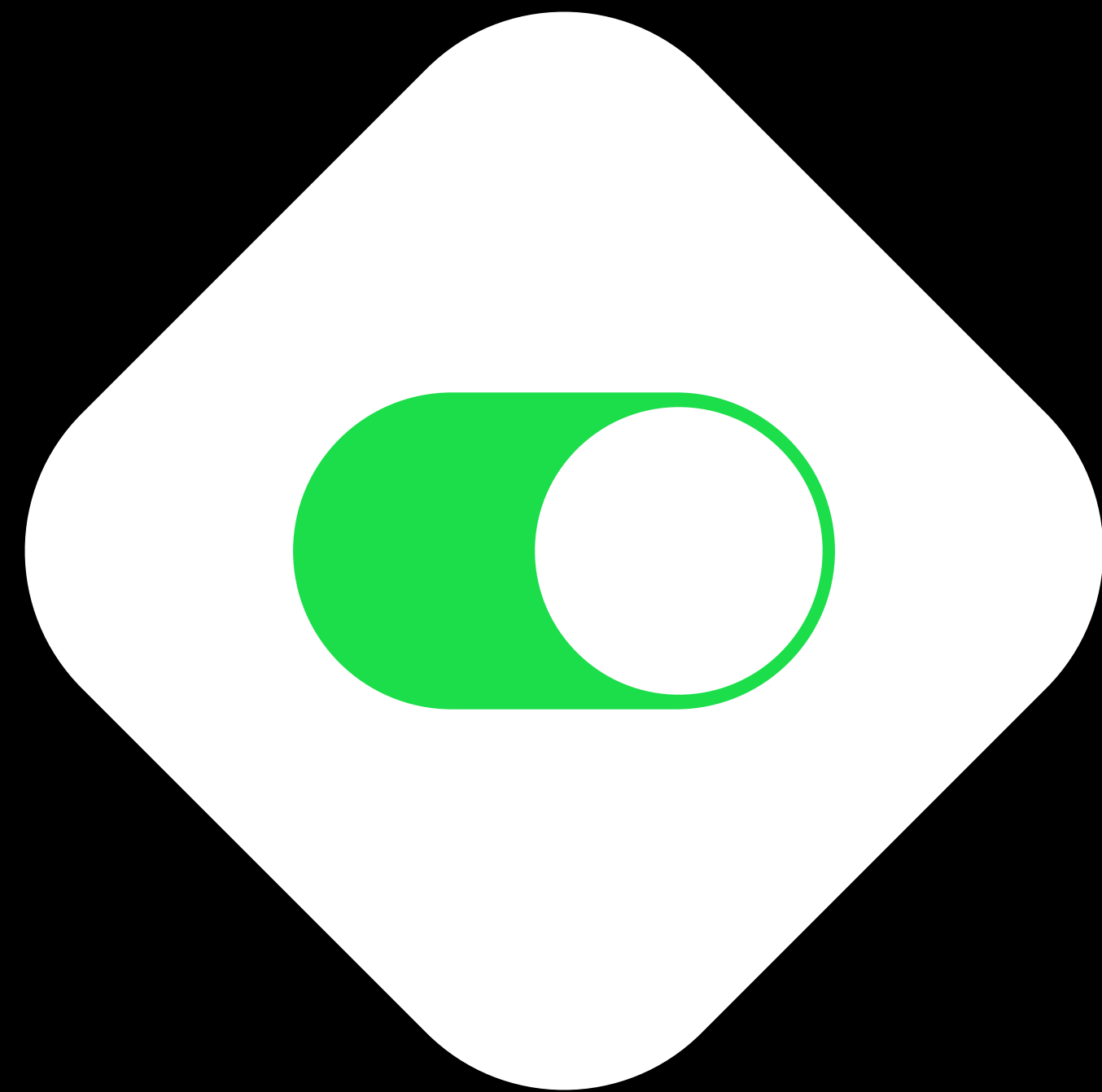
Xcode Server



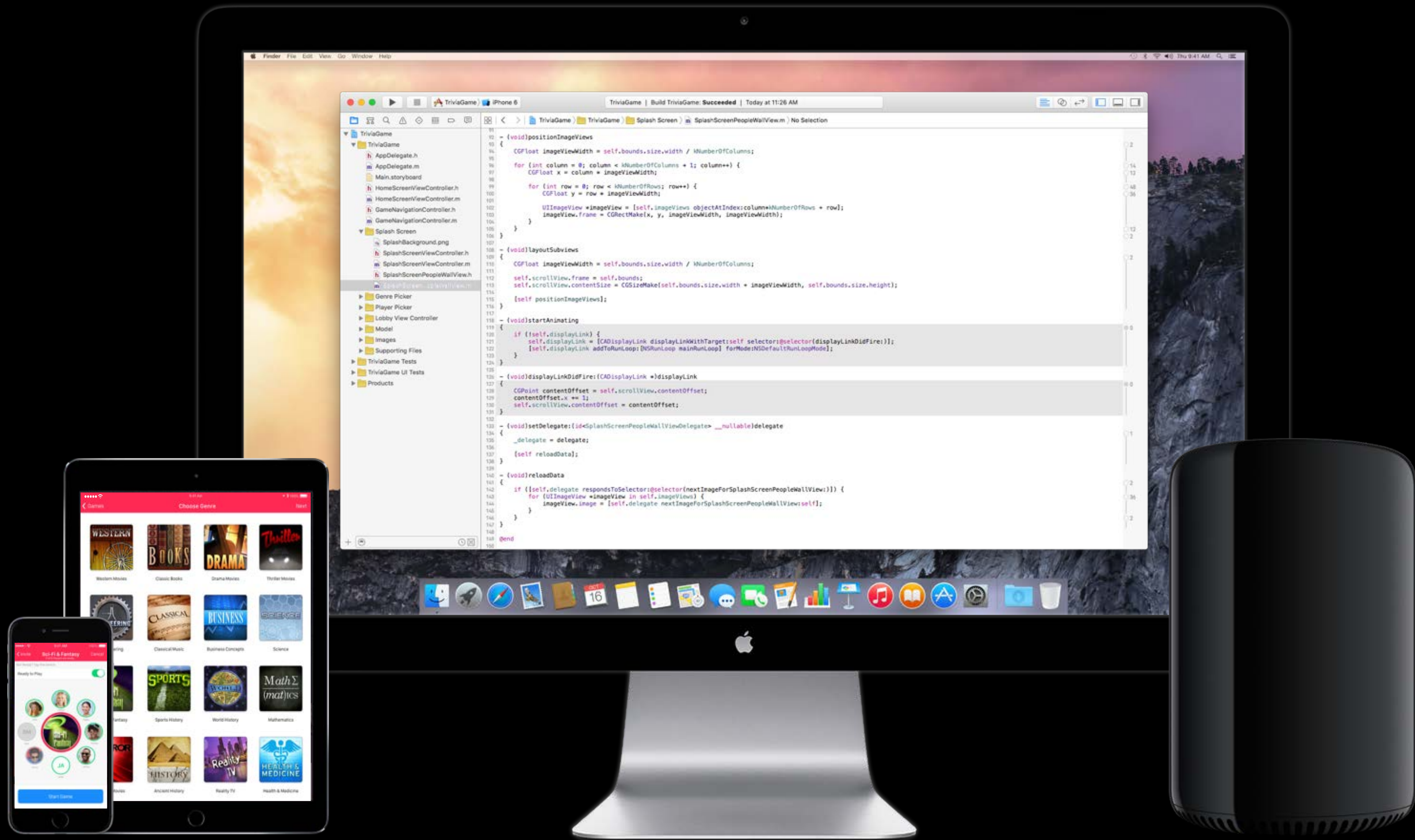
Performance



Asynchronous









User Interface Testing





Code Coverage

- ▶  **Build**
3 targets
- ▶ ▶ **Run**
Debug
- ▶  **Test**
Debug
- ▶  **Profile**
Release
- ▶  **Analyze**
Debug
- ▶  **Archive**
Release
- ▶  **Install**
Debug

InfoArgumentsDiagnostics

Build Configuration

Debug

Code Coverage



☒ Gather coverage data

Debugger

☒ Debug executable








Debug Process As

☒ Me
☐ root

Tests	Test Application Data	Test Location	Test
▶  Trivia Tests	None	◊ None	◊ <input checked="" type="checkbox"/>
▶  Trivia Tests UI Tests	None	◊ None	◊ <input checked="" type="checkbox"/>

+

—

- ▶  **Build**
3 targets
- ▶  **Run**
Debug
- ▶  **Test**
Debug
- ▶  **Profile**
Release
- ▶  **Analyze**
Debug
- ▶  **Archive**
Release
- ▶  **Install**
Debug

InfoArgumentsDiagnostics

Build Configuration

Debug

Code Coverage



☒ Gather coverage data

Debugger

☒ Debug executable


Debug Process As


☒ Me
☐ root

Tests	Test Application Data	Test Location	Test
▶  Trivia Tests	None	◊ None	◊ <input checked="" type="checkbox"/>
▶  Trivia Tests UI Tests	None	◊ None	◊ <input checked="" type="checkbox"/>

+

-


▶  **Build**
3 targets

▶  **Run**
Debug

▶  **Test**
Debug

▶  **Profile**
Release

▶  **Analyze**
Debug

▶  **Archive**
Release

▶  **Install**
Debug

Info

Arguments

Diagnostics

Build Configuration

Debug

Code Coverage



Gather coverage data

Debugger



Debug executable

Debug Process As



Me



root

Tests

Test Application Data

Test Location

Test



Trivia Tests

None



None



Trivia Tests UI Tests

None



None



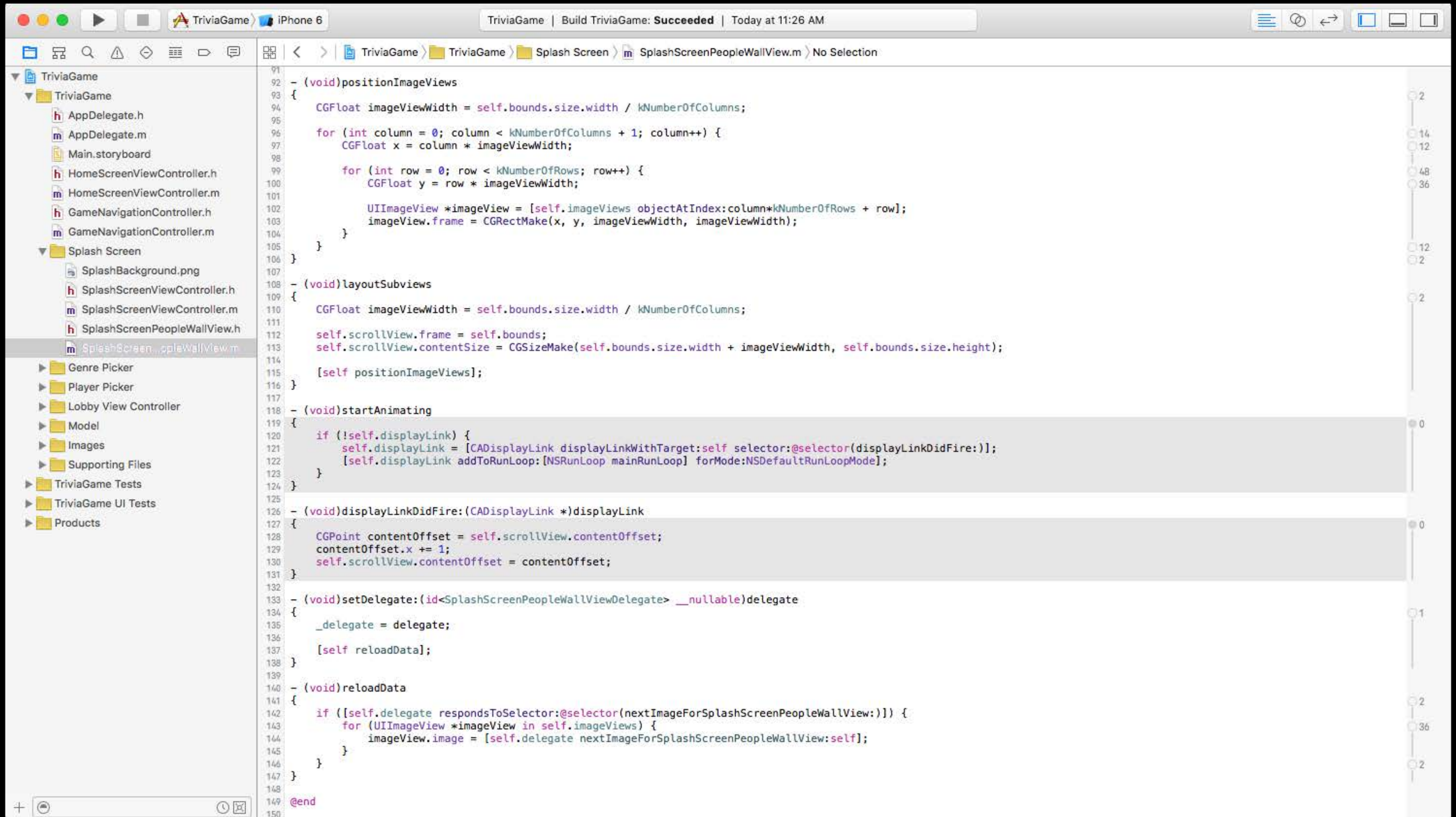
Duplicate Scheme

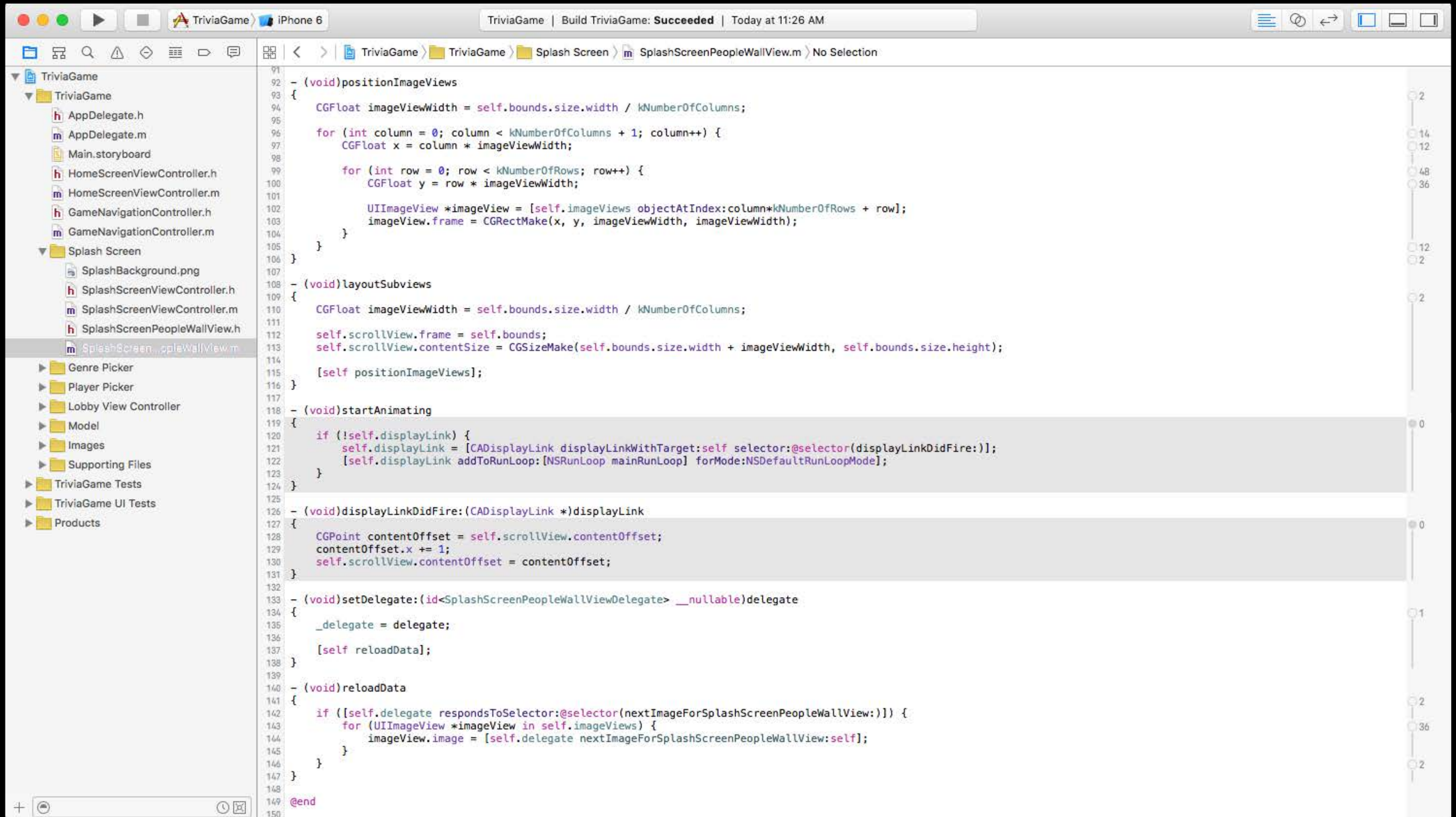
Manage Schemes...

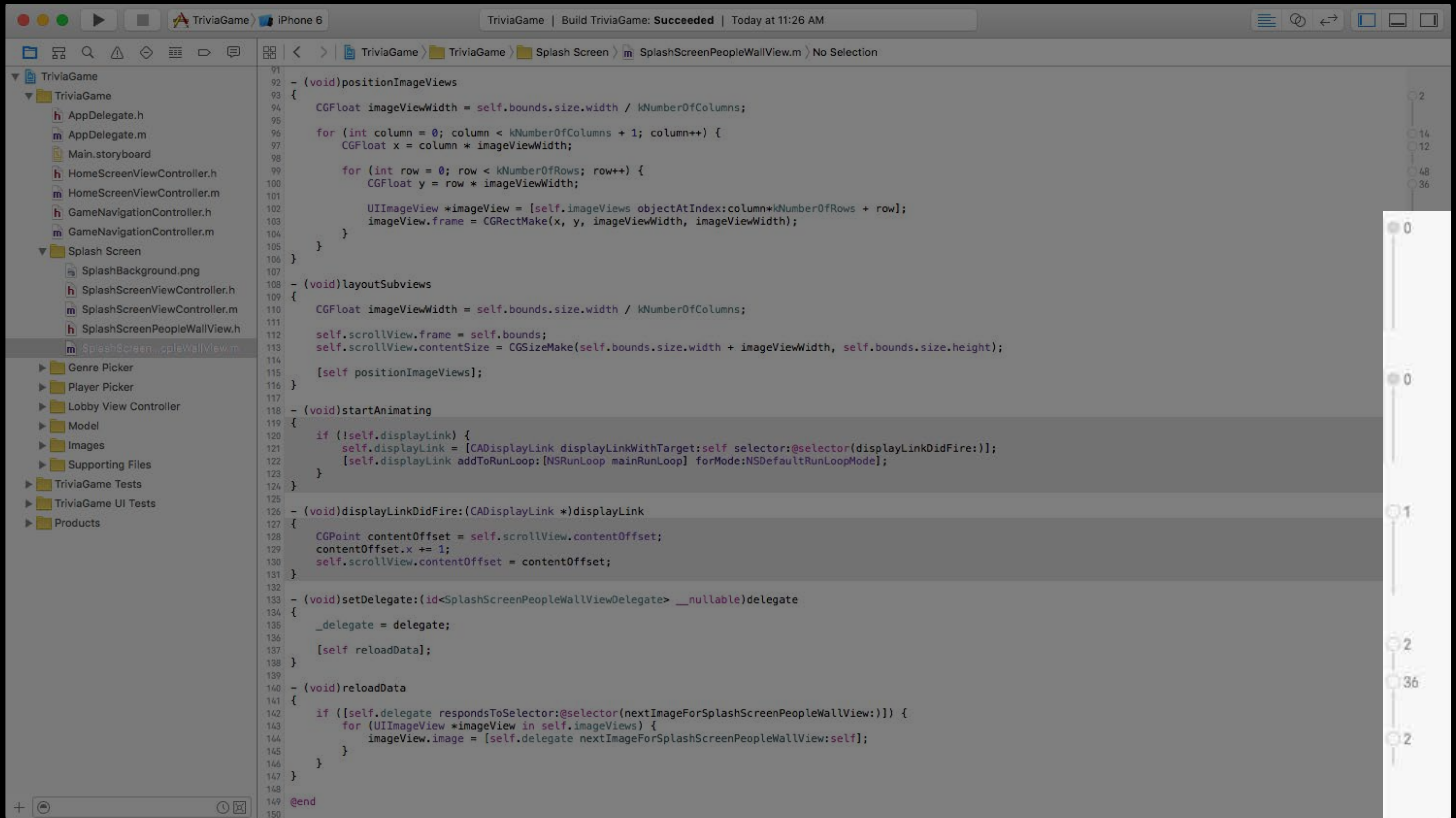


Shared

Close







Demo

Joar Wingfors Developer Tools

Demo

UI Testing in Xcode

Nob Hill

Wednesday 11:00AM

Continuous Integration and Code Coverage in Xcode

Presidio

Thursday 10:00AM

Action Editor

Accurate Rendering

iOS Energy Gauge

User Interface Recording

Texture Atlas Support

Incremental SCM Logs

GPU System Trace

Bitcode

User Interface Testing

Local ODR hosting

Instruments Track View

Data Sets

Code Coverage

SCM with ssh

Level Editor

App Extension Crash Logs

App Slicing

Action Editor

Stack View

Playground Pages

Address Sanitizer

ODR Resource Tagging

Global Illumination

ODR Debug Gauge

Storyboard References

OS X Crash Logs

View Debugger Focus

SCM On-Ramp





