Gerontechnology II

Collecting Smart Phone Sensor Data for Gerontechnology

Using iOS

Introduction to iOS

- iOS devices and sensors
- Xcode
- Swift
- Getting started with Sensor App

iOS Devices

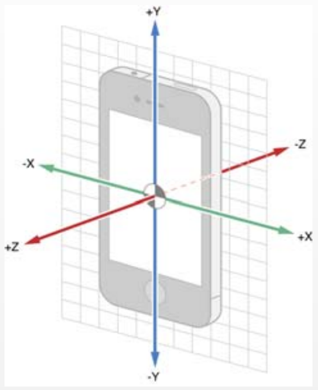
iPad

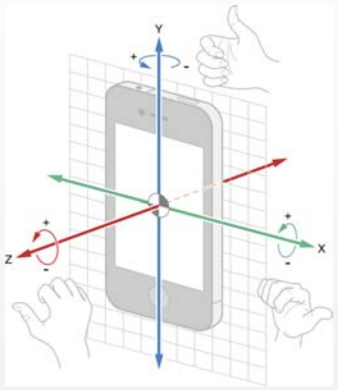
iPhone

Apple Watch

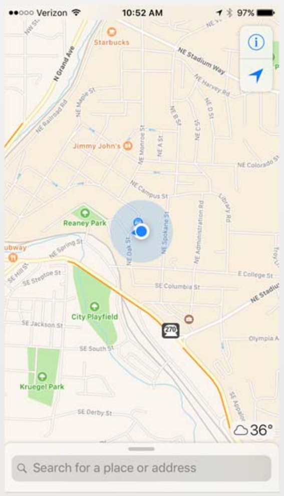
iOS Sensors

- Motion
  - Accelerometer
  - Gyroscope





- Location and course
  - GPS

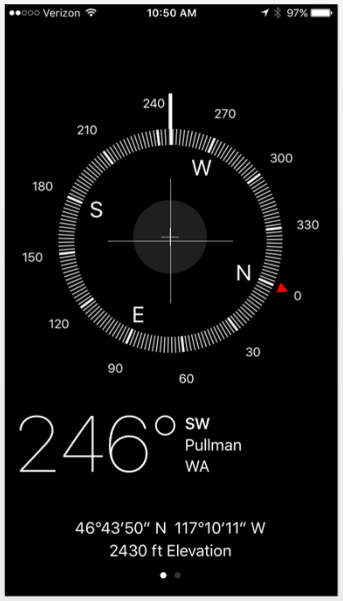


Available on iPad, iPhone and Apple Watch.

iOS Sensors

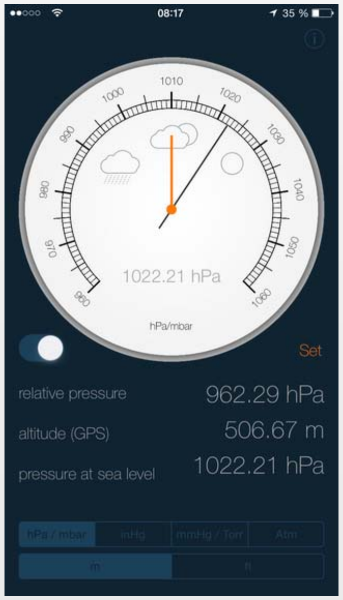
iPhone/iPad only
- Magnetometer
- Barometer

Watch only
- Heart rate
  - Green and infrared LEDs
  - Photodiodes



246° SW
Pullman
WA

46°43′50″ N 117°10′11″ W
2430 ft Elevation



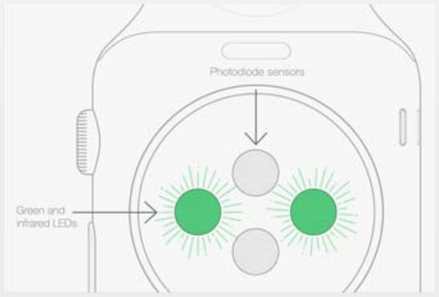
1022.21 hPa

relative pressure 962.29 hPa

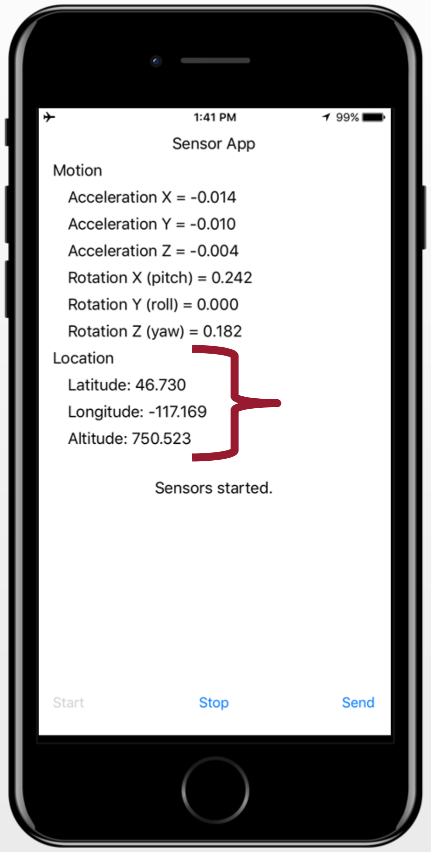
altitude (GPS) 506.67 m

pressure at sea level 1022.21 hPa

WATCH · SERIES 2 · 42MM CERAMIC CASE · SAPPHIRE CRYSTAL · CERAMIC BACK · GPS · WR-50M



Sensor App



Sensor App

Motion

Acceleration X = -0.014

Acceleration Y = -0.010

Acceleration Z = -0.004

Rotation X (pitch) = 0.242

Rotation Y (roll) = 0.000

Rotation Z (yaw) = 0.182

Location

Latitude: 46.730

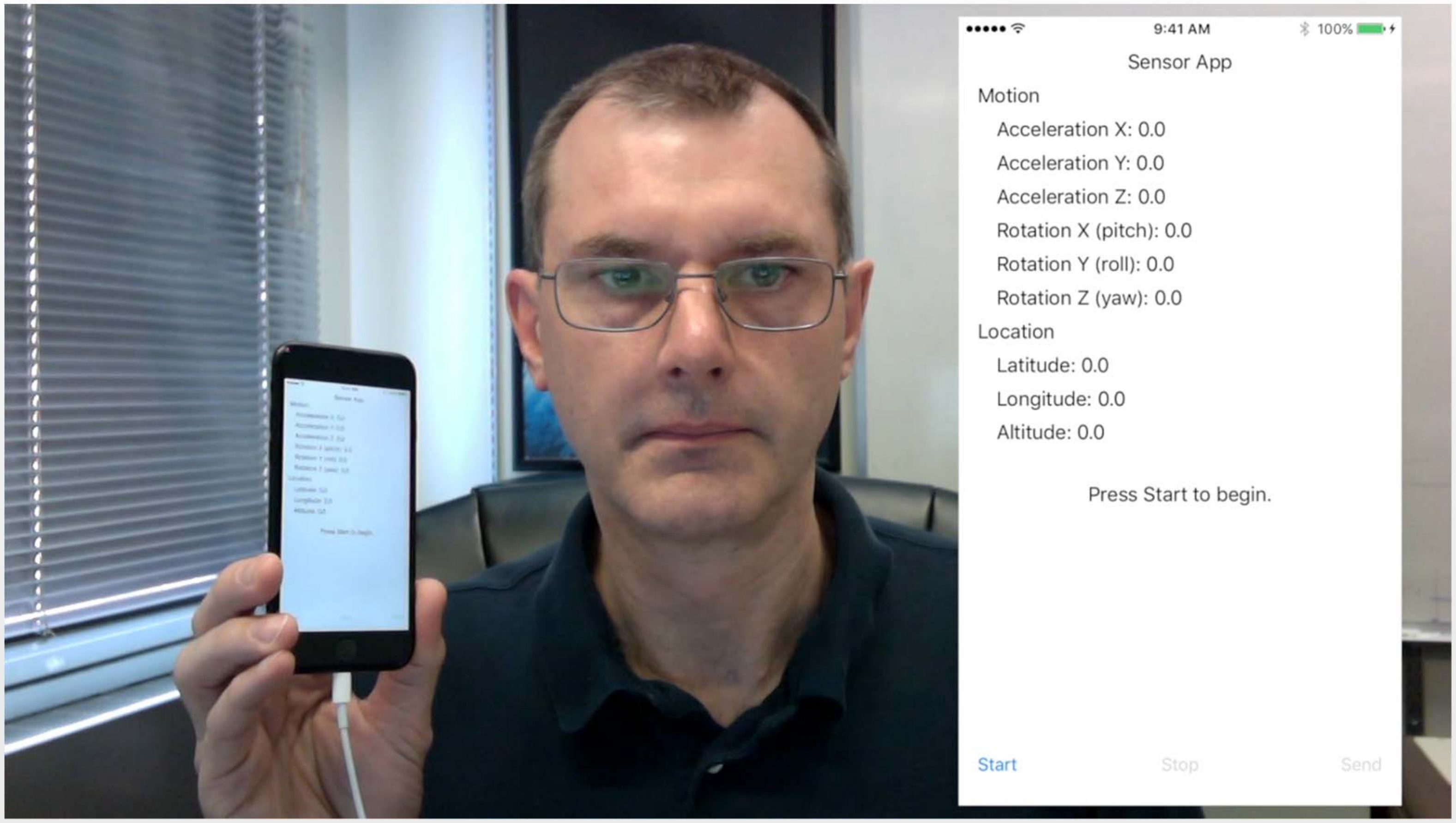
Longitude: -117.169

Altitude: 750.523

Sensors started.

Include if time

Sensor App



Sensor App

Motion

Acceleration X: 0.0

Acceleration Y: 0.0

Acceleration Z: 0.0

Rotation X (pitch): 0.0

Rotation Y (roll): 0.0

Rotation Z (yaw): 0.0

Location

Latitude: 0.0

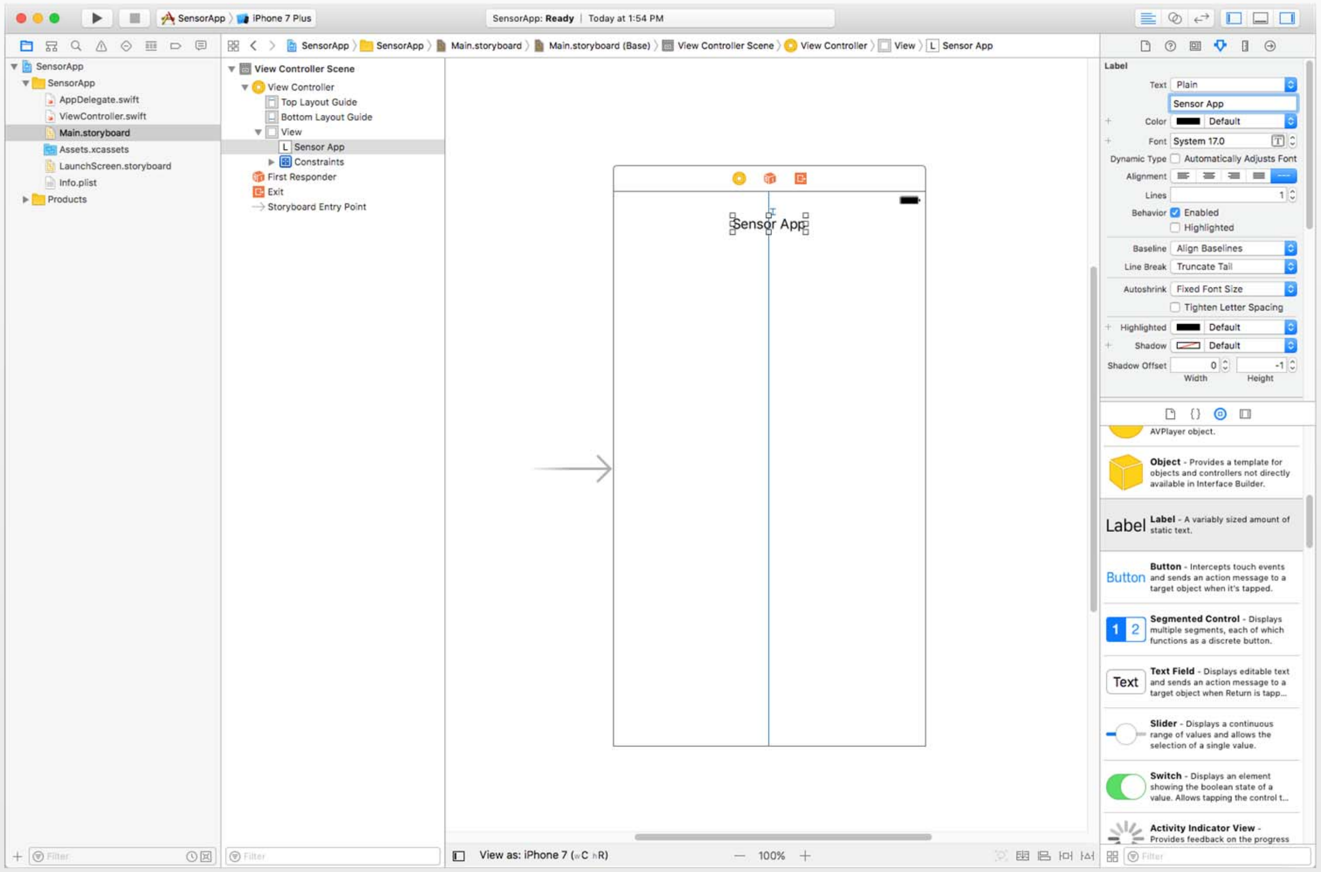
Longitude: 0.0

Altitude: 0.0

Press Start to begin.

Xcode

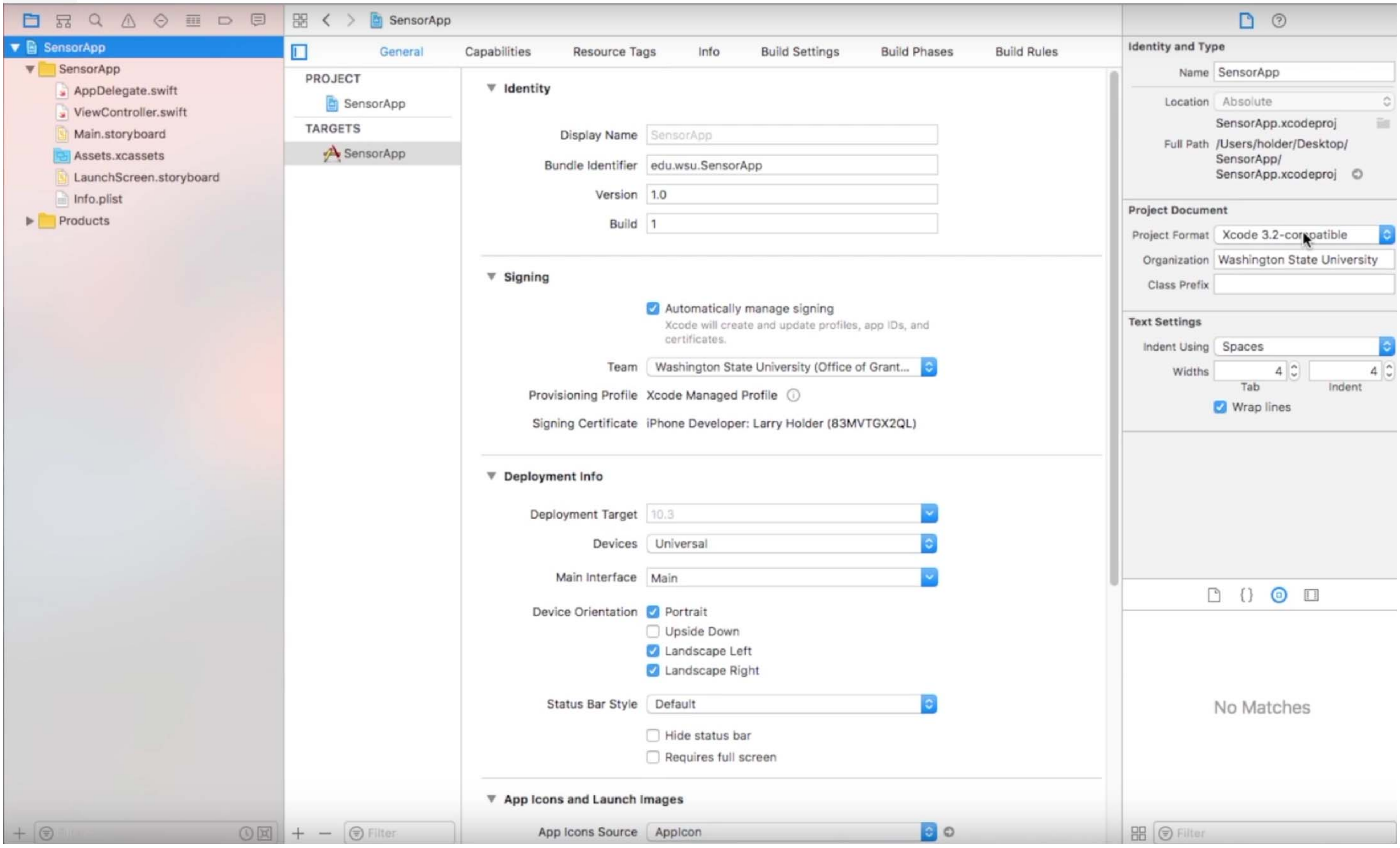
- Main iOS app development environment
- Storyboard: Visual editor for app interface
- iOS simulator


SensorApp
iPhone 7 Plus
SensorApp: Ready | Today at 1:54 PM
SensorApp > SensorApp > Main.storyboard > Main.storyboard (Base) > View Controller Scene > View Controller > View > Sensor App
SensorApp
SensorApp
AppDelegate.swift
ViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
View Controller Scene
View Controller
Top Layout Guide
Bottom Layout Guide
View
Sensor App
Constraints
First Responder
Exit
Storyboard Entry Point
Sensor App
View as: iPhone 7 (wC hR)
100%
Filter
Label
AVPlayer object.
Object - Provides a template for objects and controllers not directly available in Interface Builder.
Label - A variably sized amount of static text.
Button - Intercepts touch events and sends an action message to a target object when it's tapped.
Segmented Control - Displays multiple segments, each of which functions as a discrete button.
Text Field - Displays editable text and sends an action message to a target object when Return is tapp...
Slider - Displays a continuous range of values and allows the selection of a single value.
Switch - Displays an element showing the boolean state of a value. Allows tapping the control t...
Activity Indicator View - Provides feedback on the progress
Button
Text
1 2

Label

Text	Plain
	Sensor App
Color	Default
Font	System 17.0
Dynamic Type	Automatically Adjusts Font
Alignment	
Lines	1
Behavior	✓ Enabled
	Highlighted
Baseline	Align Baselines
Line Break	Truncate Tail
Autoshrink	Fixed Font Size
	Tighten Letter Spacing
Highlighted	Default
Shadow	Default
Shadow Offset	0 (Width), -1 (Height)

Xcode



Identity

Display Name: SensorApp
Bundle Identifier: edu.wsu.SensorApp
Version: 1.0
Build: 1

Signing

✓ Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.

Team: Washington State University (Office of Grant...
Provisioning Profile: Xcode Managed Profile
Signing Certificate: iPhone Developer: Larry Holder (83MVTGX2QL)

Deployment Info

Deployment Target: 10.3
Devices: Universal
Main Interface: Main
Device Orientation: ✓ Portrait; ☐ Upside Down; ✓ Landscape Left; ✓ Landscape Right
Status Bar Style: Default
☐ Hide status bar
☐ Requires full screen

App Icons and Launch Images

App Icons Source: AppIcon

Identity and Type

Name: SensorApp
Location: Absolute
SensorApp.xcodeproj
Full Path: /Users/holder/Desktop/SensorApp/SensorApp.xcodeproj

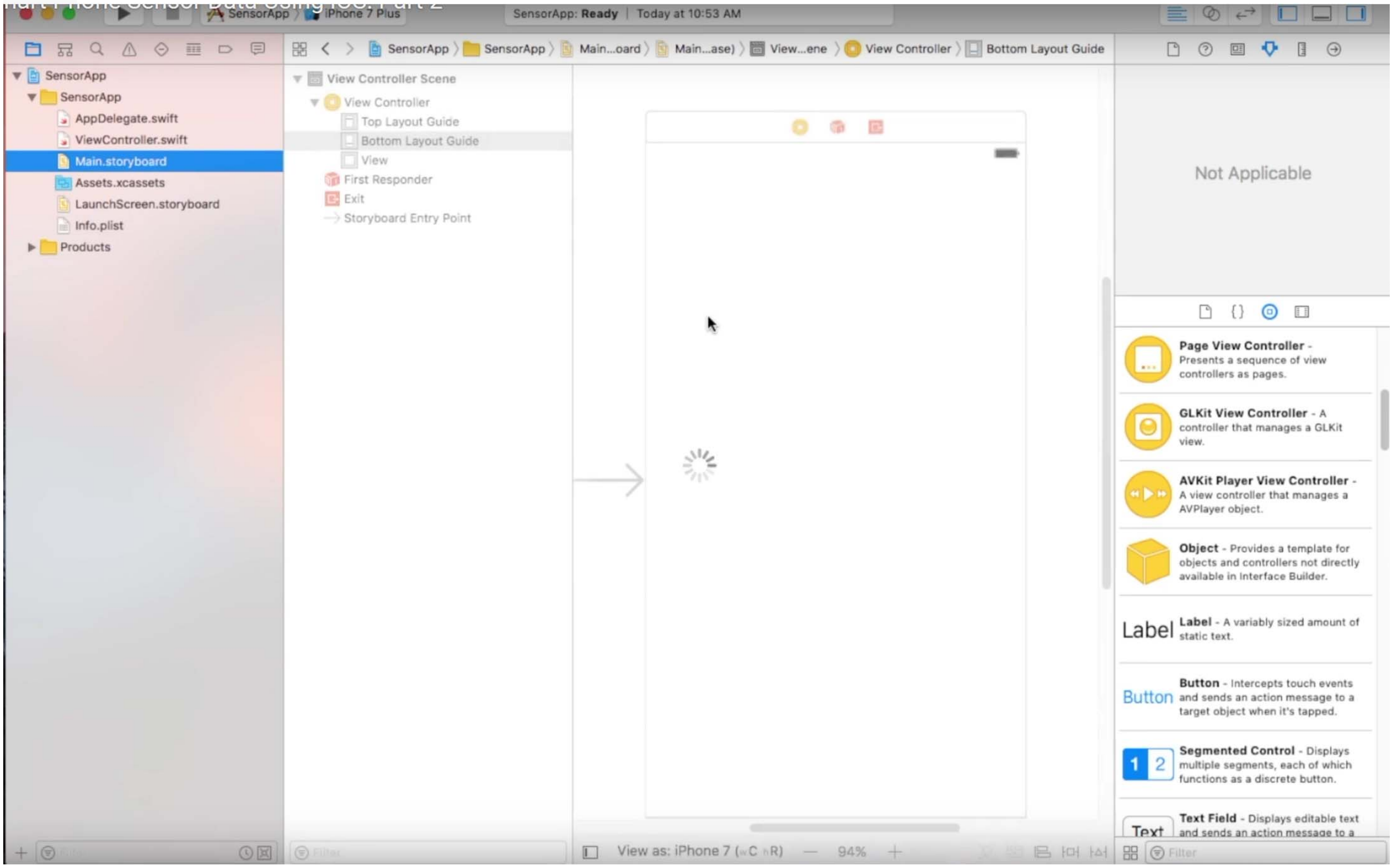
Project Document

Project Format: Xcode 3.2-compatible
Organization: Washington State University
Class Prefix:

Text Settings

Indent Using: Spaces
Widths: Tab 4, Indent 4
✓ Wrap lines

Storyboard (Main.storyboard)



Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.

AVKit Player View Controller - A view controller that manages a AVPlayer object.

Object - Provides a template for objects and controllers not directly available in Interface Builder.

Label - A variably sized amount of static text.

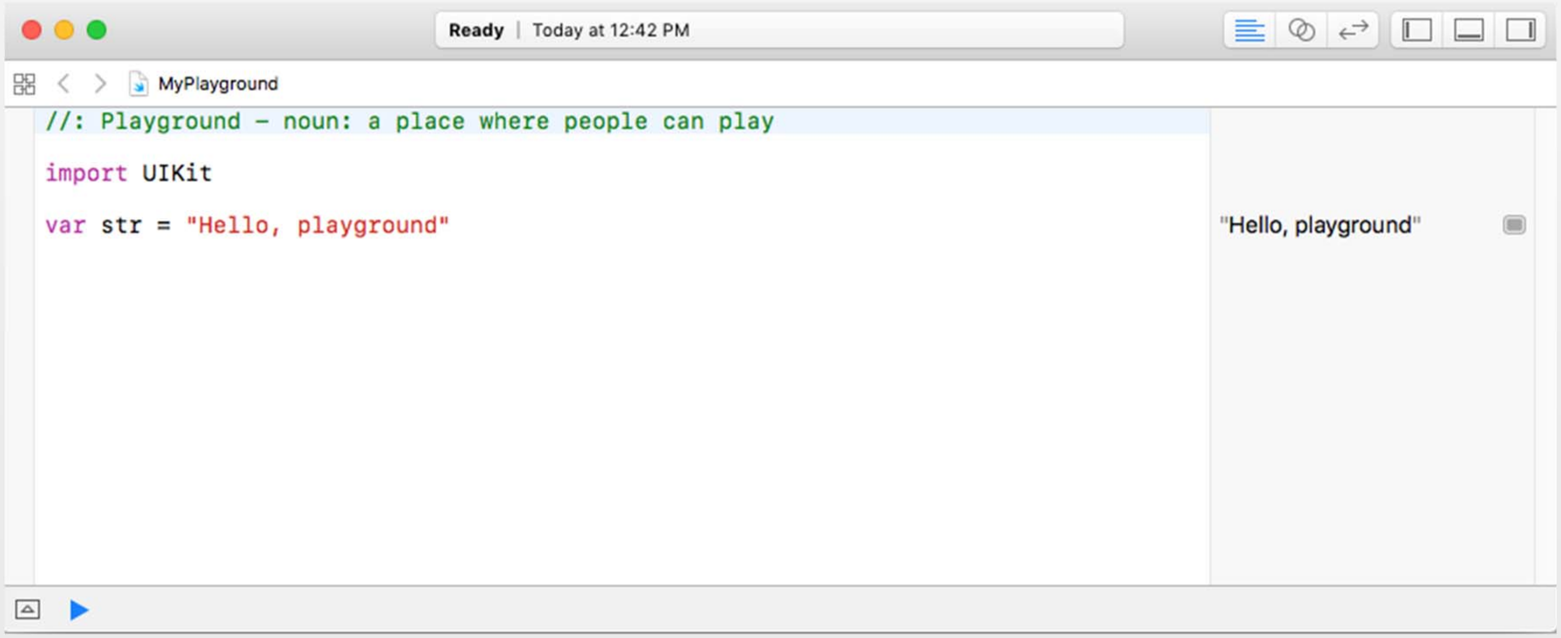
Button - Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a

Swift

- Programming language for iOS
- Swift Tour at http://swift.org/getting-started
- Swift Playground



```
//: Playground - noun: a place where people can play

import UIKit

var str = "Hello, playground"
```

```
"Hello, playground"
```

Constants, Variables and Types

- Constants (let)
- Variables (var)
- Basic types: Bool, Int, Float, Double, String
- Collection types: Array, Set, Dictionary

```
let numCandy = 4  // let numCandy:Int = 4
var shoppingList = ["coffee": 3, "candy": numCandy]
for (item, amount) in shoppingList {
        print("\(item): \(amount)")
}
```

Optional Type

- Optional variable (?) can be empty or a value
- Access value by unwrapping (!)

```
let possibleStr: String? = "Hello" // optional type
print(possibleStr)

let unwrappedStr: String = possibleStr! // unwrapping
print(unwrappedStr)
```

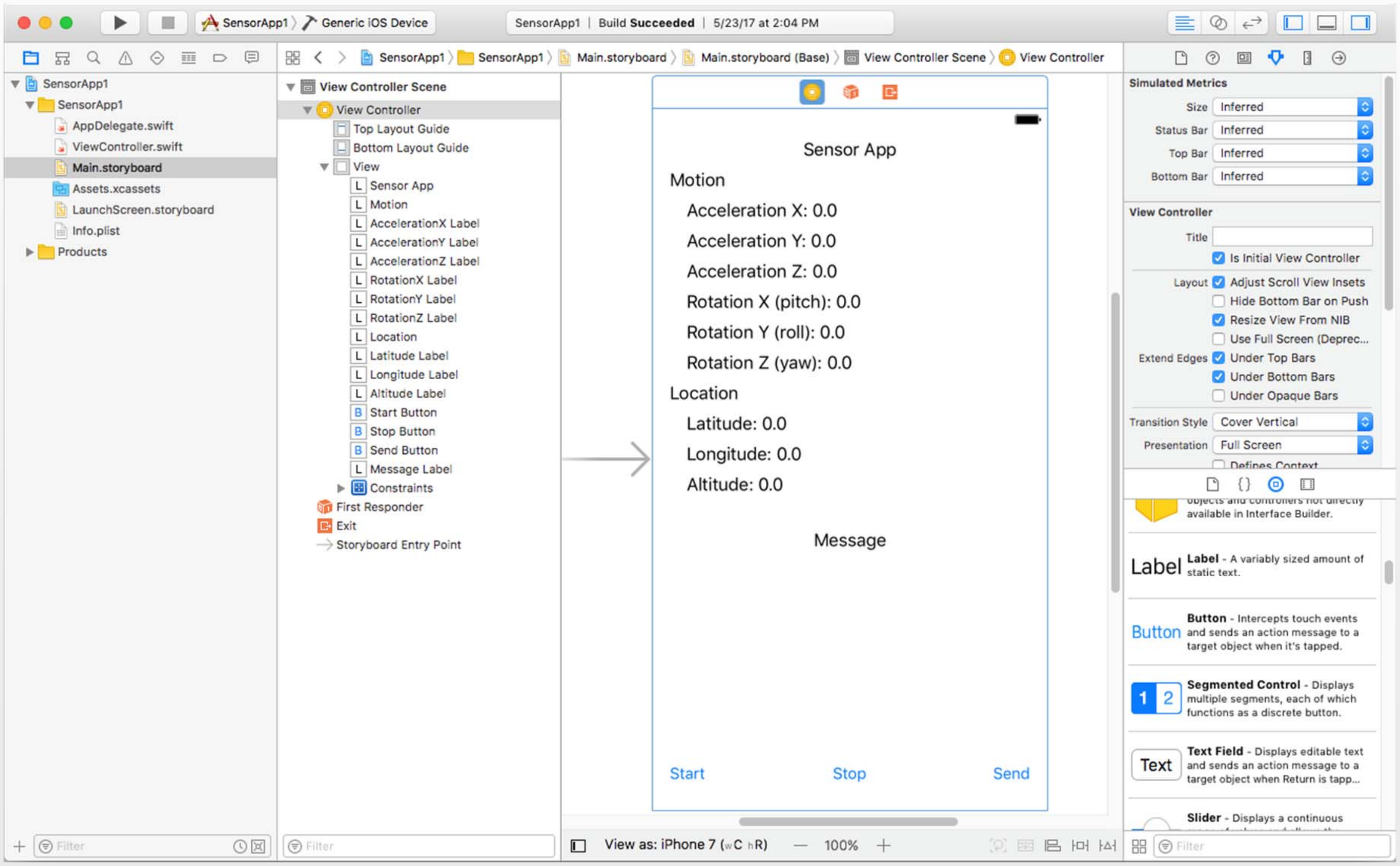
Functions

```swift
func addTwoInts (first: Int, second: Int) -> Int {
  let sum = first + second
  return sum
}
addTwoInts(first: 3, second: 4)

func addTwoInts2 (_ first: Int, _ second: Int) -> Int {
  let sum = first + second
  return sum
}
addTwoInts2(3,4)
```

Classes

```
class Location {
  var latitude: Float
  var longitude: Float
  var altitude: Float = 0.0
  var heading: Float?

  init (latitude lat: Float, longitude lon: Float) {
    self.latitude = lat
    self.longitude = lon
  }

  func jump () {
    altitude += 2.0
  }
}
var location1 = Location(latitude: 46.73, longitude: -117.17)
location1.jump()
```

Sensor App Interface



SensorApp1 〉 Generic iOS Device

SensorApp1 | Build **Succeeded** | 5/23/17 at 2:04 PM

SensorApp1 〉 SensorApp1 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 View Controller Scene 〉 View Controller

- SensorApp1
  - SensorApp1
    - AppDelegate.swift
    - ViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - Products

- View Controller Scene
  - View Controller
    - Top Layout Guide
    - Bottom Layout Guide
    - View
      - Sensor App
      - Motion
      - AccelerationX Label
      - AccelerationY Label
      - AccelerationZ Label
      - RotationX Label
      - RotationY Label
      - RotationZ Label
      - Location
      - Latitude Label
      - Longitude Label
      - Altitude Label
      - Start Button
      - Stop Button
      - Send Button
      - Message Label
      - Constraints
  - First Responder
  - Exit
  - Storyboard Entry Point

Sensor App

Motion

Acceleration X: 0.0

Acceleration Y: 0.0

Acceleration Z: 0.0

Rotation X (pitch): 0.0

Rotation Y (roll): 0.0

Rotation Z (yaw): 0.0

Location

Latitude: 0.0

Longitude: 0.0

Altitude: 0.0

Message

Start Stop Send

View as: iPhone 7 (wC hR) — 100% +

Simulated Metrics

Size	Inferred
Status Bar	Inferred
Top Bar	Inferred
Bottom Bar	Inferred

View Controller

Title	
	☑ Is Initial View Controller
Layout	☑ Adjust Scroll View Insets
	☐ Hide Bottom Bar on Push
	☑ Resize View From NIB
	☐ Use Full Screen (Deprec...
Extend Edges	☑ Under Top Bars
	☑ Under Bottom Bars
	☐ Under Opaque Bars
Transition Style	Cover Vertical
Presentation	Full Screen
	☐ Defines Context

objects and controllers not directly available in Interface Builder.

Label - A variably sized amount of static text.

Button - Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a target object when Return is tapp...

Slider - Displays a continuous

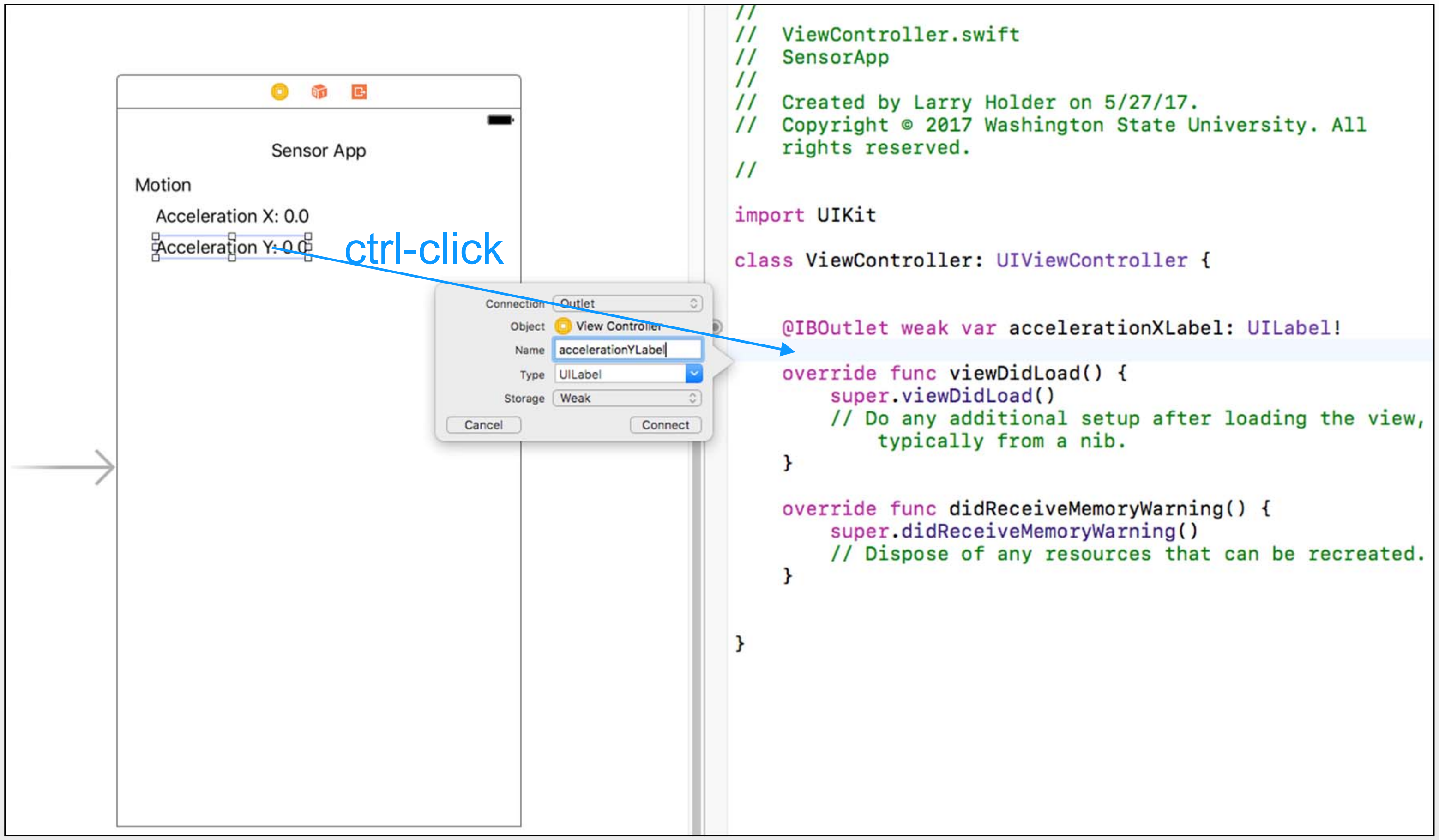
Labels and Buttons

- UILabel
  - var messageLabel: UILabel!
  - messageLabel.text = “Press Start to begin.”
- UIButton
  - var startButton: UIButton!
  - startButton.isEnabled = true

Outlets and Actions

- Outlet
  - Provides access to view elements
  - @IBOutlet weak var messageLabel: UILabel!
- Action
  - Reacts to user interaction with view elements
  - @IBAction func startTapped (_ sender: UIButton) {…}
- Connection created by Ctrl-Click from element to ViewController class

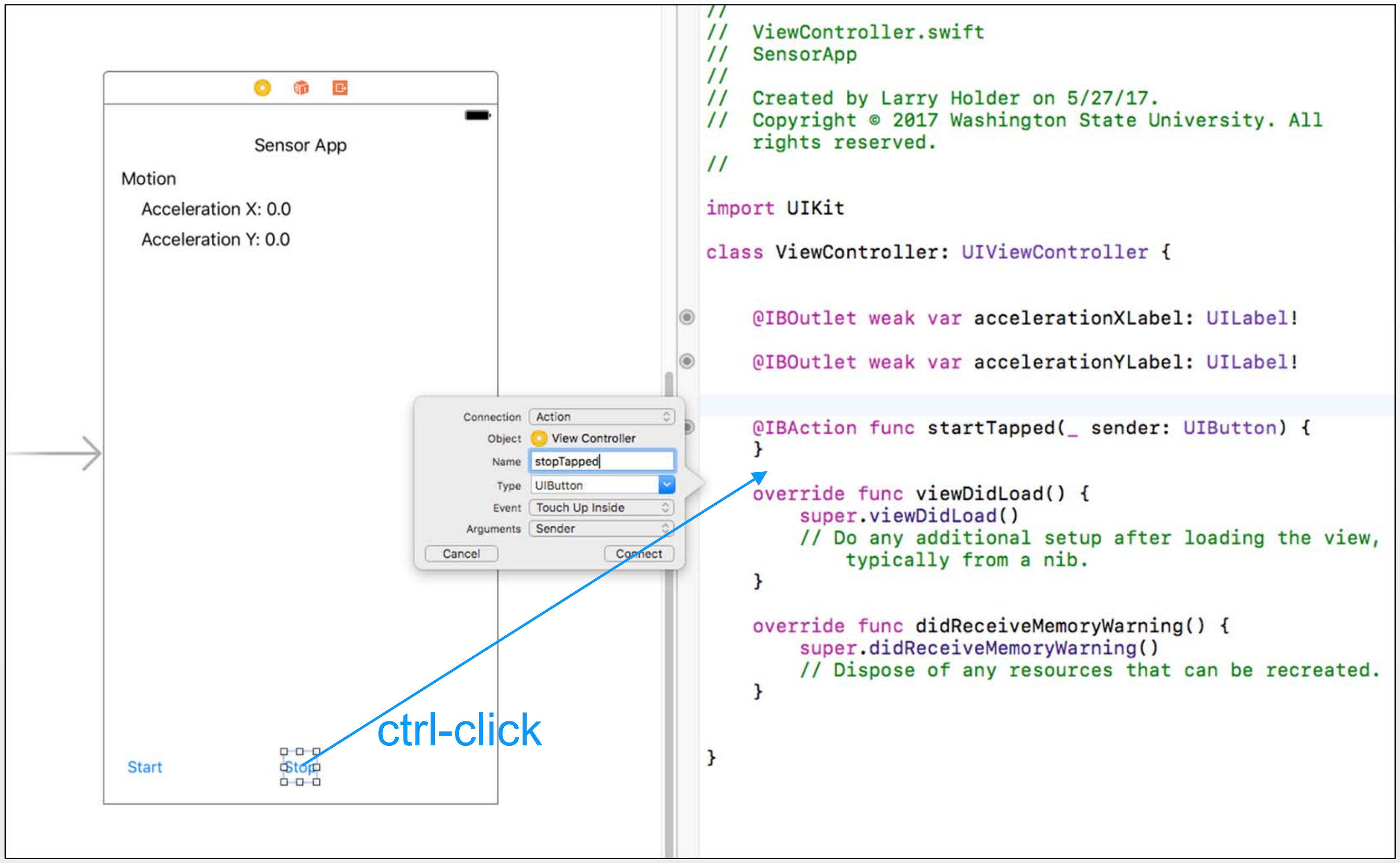
Connecting Label Outlets



Connection: Outlet
Object: View Controller
Name: accelerationYLabel
Type: UILabel
Storage: Weak

```
//
//  ViewController.swift
//  SensorApp
//
//  Created by Larry Holder on 5/27/17.
//  Copyright © 2017 Washington State University. All
    rights reserved.
//

import UIKit

class ViewController: UIViewController {


    @IBOutlet weak var accelerationXLabel: UILabel!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view,
            typically from a nib.
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }


}
```

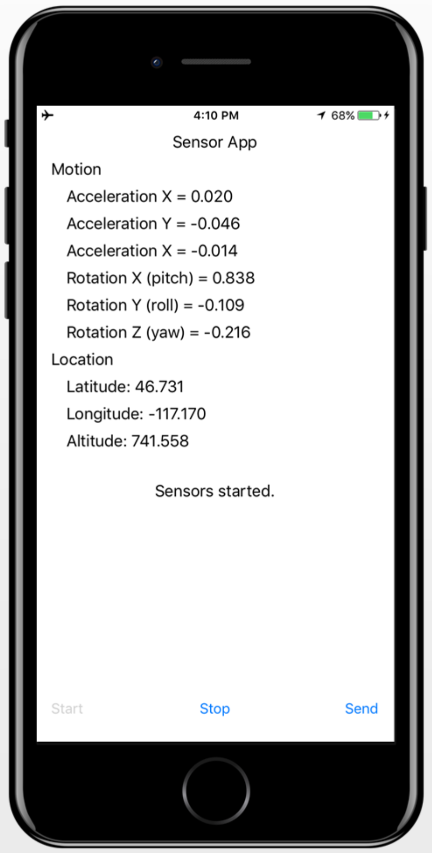
Connecting Button Actions



Connection	Action
Object	View Controller
Name	stopTapped
Type	UIButton
Event	Touch Up Inside
Arguments	Sender

```
//
//  ViewController.swift
//  SensorApp
//
//  Created by Larry Holder on 5/27/17.
//  Copyright © 2017 Washington State University. All
    rights reserved.
//

import UIKit

class ViewController: UIViewController {

    @IBOutlet weak var accelerationXLabel: UILabel!

    @IBOutlet weak var accelerationYLabel: UILabel!

    @IBAction func startTapped(_ sender: UIButton) {
    }

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view,
            typically from a nib.
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }


}
```

Sensor App



4:10 PM

Sensor App

Motion

Acceleration X = 0.020

Acceleration Y = -0.046

Acceleration X = -0.014

Rotation X (pitch) = 0.838

Rotation Y (roll) = -0.109

Rotation Z (yaw) = -0.216

Location

Latitude: 46.731

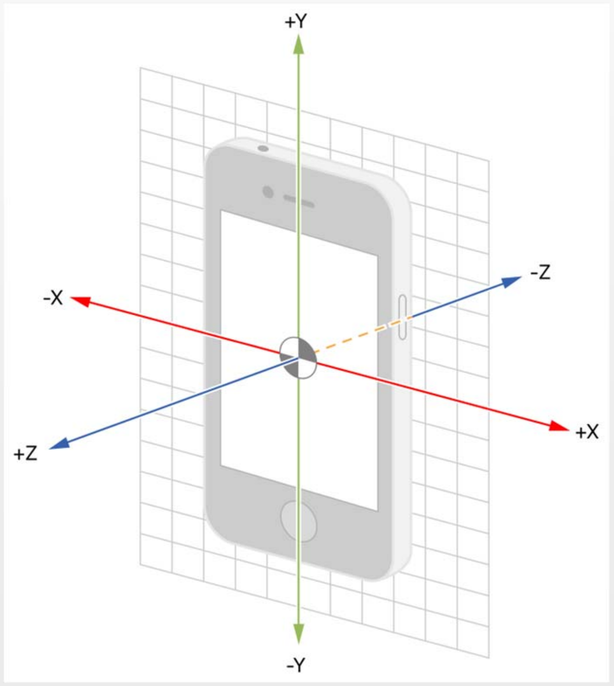
Longitude: -117.170

Altitude: 741.558

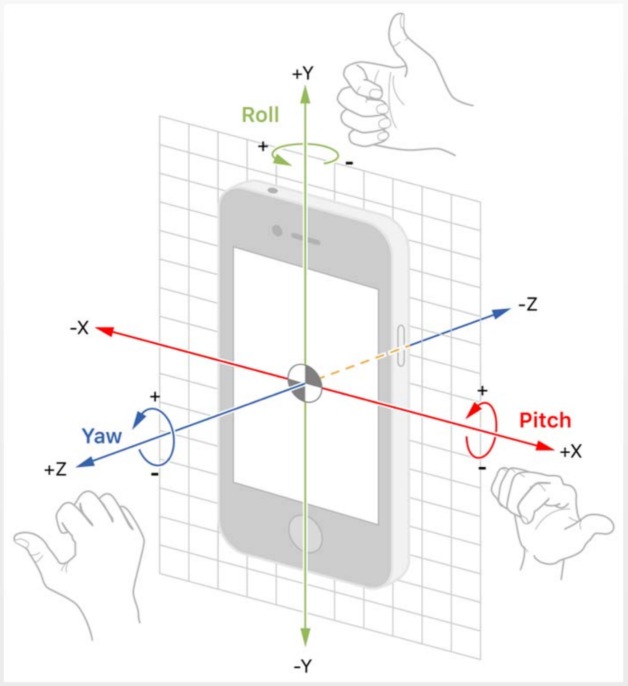
Sensors started.

Motion Sensors

- Accelerometer

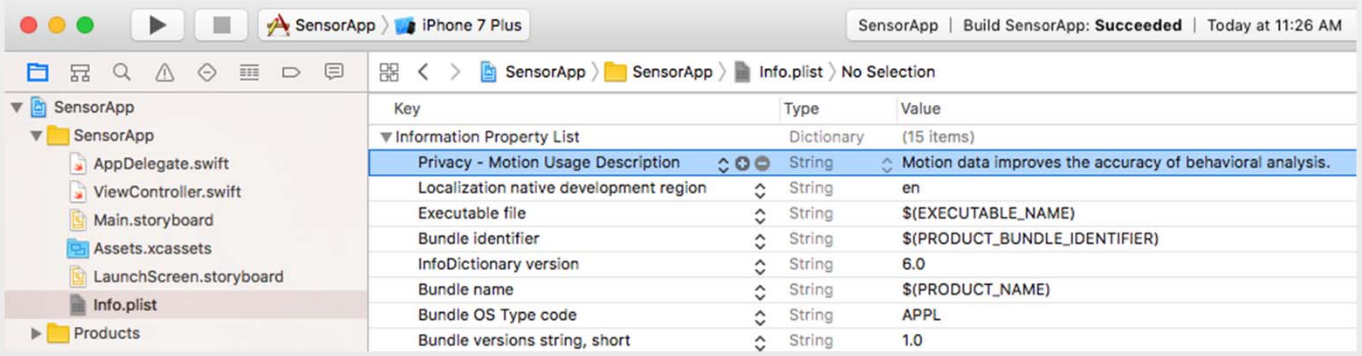


- Gyroscope



Motion Sensor Authorization

- App must provide description for why motion data is necessary
  - To protect user privacy
  - App terminates if not provided
- Info.plist
  - "Privacy – Motion Usage Description"

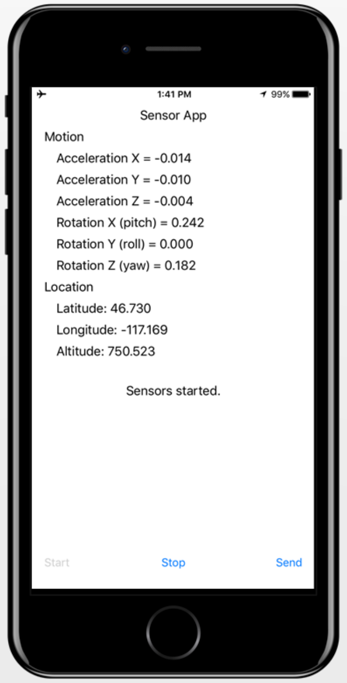

SensorApp 〉 iPhone 7 Plus
SensorApp | Build SensorApp: **Succeeded** | Today at 11:26 AM
SensorApp 〉 SensorApp 〉 Info.plist 〉 No Selection

- SensorApp
  - SensorApp
    - AppDelegate.swift
    - ViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - Products

Key	Type	Value
▼ Information Property List	Dictionary	(15 items)
Privacy - Motion Usage Description	String	Motion data improves the accuracy of behavioral analysis.
Localization native development region	String	en
Executable file	String	$(EXECUTABLE_NAME)
Bundle identifier	String	$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0

Motion Sensors

- Accelerometer and gyroscope
- Sensor availability
- Sensor authorization
- Updates to Sensor App

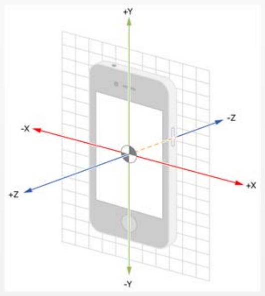


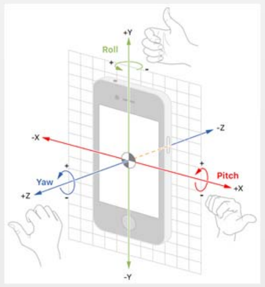
Motion
  Acceleration X = -0.014
  Acceleration Y = -0.010
  Acceleration Z = -0.004
  Rotation X (pitch) = 0.242
  Rotation Y (roll) = 0.000
  Rotation Z (yaw) = 0.182
Location
  Latitude: 46.730
  Longitude: -117.169
  Altitude: 750.523

Sensors started.

Core Motion Framework

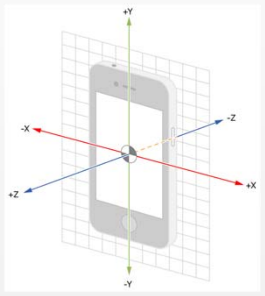
- Import CoreMotion
- Create instance of CMMotionManager
- Set update interval
- Start updates: Calls handler
  - Handler gets CMDeviceMotion object
    - userAcceleration.x/y/z (minus gravity)
    - attitude.yaw/pitch/roll
- Stop updates
- developer.apple.com/documentation/coremotion

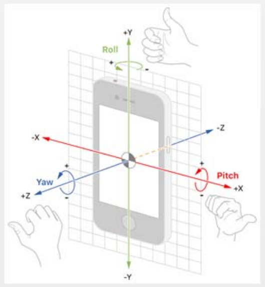




Core Motion Framework

- Import CoreMotion
- Create instance of CMMotionManager
- Set update interval
- Start updates: Calls handler
  - Handler gets CMDeviceMotion object
    - userAcceleration.x/y/z (minus gravity)
    - attitude.yaw/pitch/roll
- Stop updates
- developer.apple.com/documentation/coremotion





Core Motion Initialization

```
import CoreMotion

class ViewController: UIViewController {

  var motionManager: CMMotionManager!

  func initializeMotion() { // call from viewDidLoad
    motionManager = CMMotionManager()
    motionManager.deviceMotionUpdateInterval = 0.1 // secs
  }

  func startMotionUpdates () { // call from startTapped
    motionManager.startDeviceMotionUpdates(
      to: OperationQueue.main, withHandler: motionHandler)
  }

  func stopMotionUpdates () { // call from stopTapped
    motionManager.stopDeviceMotionUpdates()
  }
```

Motion Handler

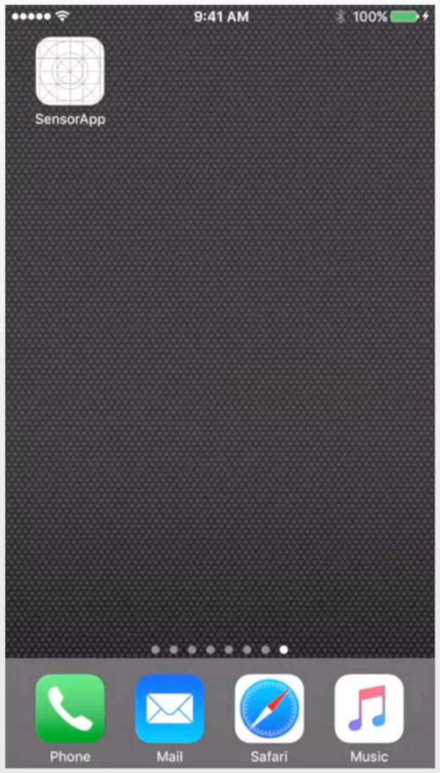
```
func motionHandler (deviceMotion: CMDeviceMotion?, error: Error?)
{
  if let err = error {
    print("motionHandler error: \(err.localizedDescription)")
  } else {
    if let dm = deviceMotion {
      self.processMotionData(dm)
    } else {
      print("motionHandler: deviceMotion = nil")
    }
  }
}
```

Process Device Motion Data

```
func processMotionData (_ dm: CMDeviceMotion) {
  let accX = String(format: "%.3f", dm.userAcceleration.x)
  let accY = String(format: "%.3f", dm.userAcceleration.y)
  let accZ = String(format: "%.3f", dm.userAcceleration.z)
  let rotX = String(format: "%.3f", dm.attitude.pitch)
  let rotY = String(format: "%.3f", dm.attitude.roll)
  let rotZ = String(format: "%.3f", dm.attitude.yaw)
  accelerationXLabel.text = "Acceleration X = \(accX)"
  accelerationYLabel.text = "Acceleration Y = \(accY)"
  accelerationZLabel.text = "Acceleration Z = \(accZ)"
  rotationXLabel.text = "Rotation X (pitch) = \(rotX)"
  rotationYLabel.text = "Rotation Y (roll) = \(rotY)"
  rotationZLabel.text = "Rotation Z (yaw) = \(rotZ)"
}
```

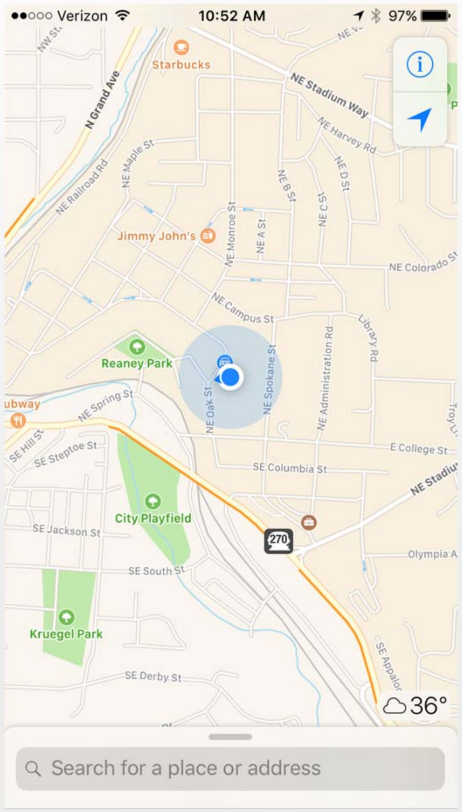
Testing Core Motion

- Not included in iOS simulator
  - Handler never called
- Must use real device

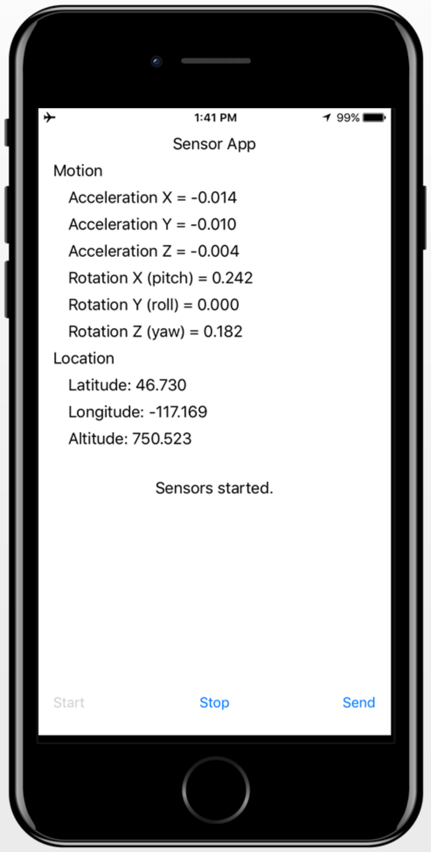


Location Sensors

- GPS
- And if available…
  - Wifi
  - Bluetooth
  - Magnetometer
  - Barometer
  - Cellular hardware



Sensor App



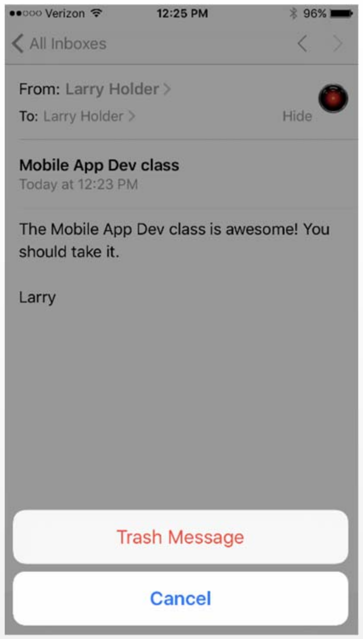
```
Motion
   Acceleration X = -0.014
   Acceleration Y = -0.010
   Acceleration Z = -0.004
   Rotation X (pitch) = 0.242
   Rotation Y (roll) = 0.000
   Rotation Z (yaw) = 0.182
Location
   Latitude: 46.730
   Longitude: -117.169
   Altitude: 750.523

Sensors started.
```

Send: Communication Over the Web

- HTTP vs. HTTPS
- GET vs. POST requests
- JSON
- Web server

Alerts

- Alerts
- Local notifications
- Remote (push) notifications


●●○○○ Verizon 12:25 PM 96%

All Inboxes

From: Larry Holder

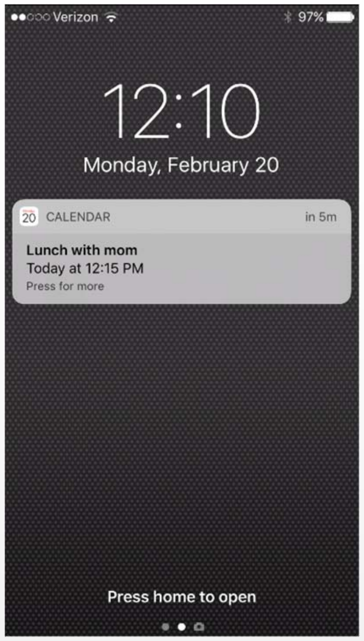
To: Larry Holder Hide

Mobile App Dev class

Today at 12:23 PM

The Mobile App Dev class is awesome! You should take it.

Larry


●●○○○ Verizon 97%

12:10

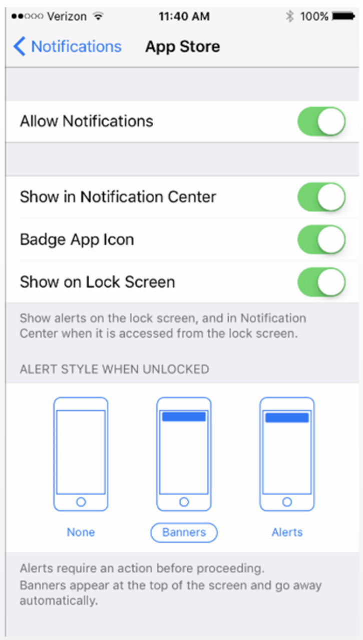
Monday, February 20

20 CALENDAR in 5m

Lunch with mom

Today at 12:15 PM

Press for more


●●○○○ Verizon 11:40 AM 100%

Notifications App Store

Allow Notifications: on

Show in Notification Center: on

Badge App Icon: on

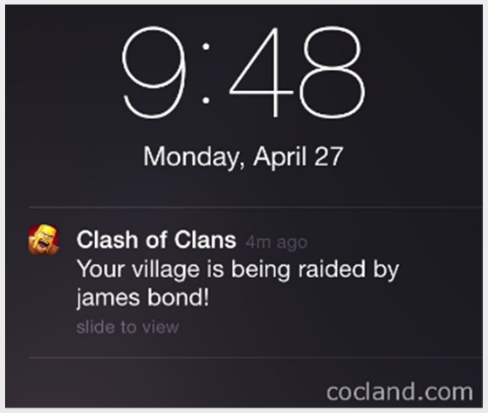
Show on Lock Screen: on

Show alerts on the lock screen, and in Notification Center when it is accessed from the lock screen.

ALERT STYLE WHEN UNLOCKED

None [✓ Banners] Alerts

Alerts require an action before proceeding.
Banners appear at the top of the screen and go away automatically.


9:48

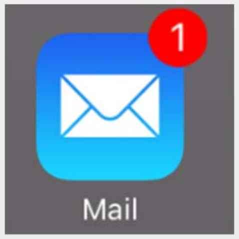
Monday, April 27

Clash of Clans 4m ago

Your village is being raided by james bond!

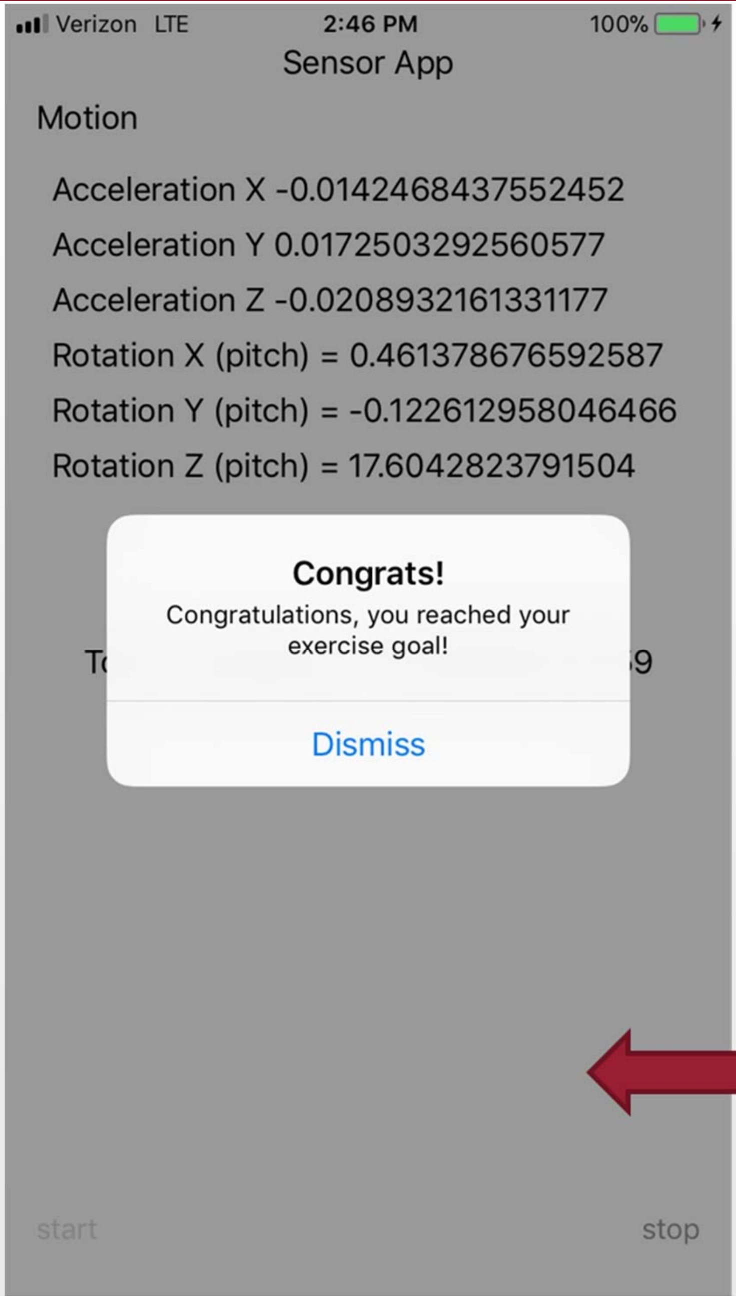
slide to view

cocland.com



```
func showAlert(titleText: String, messageText: String) {
        let alert = UIAlertController(title: titleText, message:
messageText, preferredStyle: .alert)
        let action = UIAlertAction(title: "Dismiss", style:
.default, handler: { (action) in alert.dismiss(animated: true,
completion: nil)
        })

        alert.addAction(action)

        self.present(alert, animated: true, completion: nil)
    }
```

```
var totalMotion = 0.0

func processMotionData(_ dm: CMDeviceMotion) {
        let accX = dm.userAcceleration.x
        // omitting previous lines here
        totalMotion += abs(accX) + abs(accY) + abs(accZ)
        if totalMotion > 100.0 {
             DispatchQueue.main.async {
                  self.showAlert(titleText: "Congrats!",
messageText: "Congratulations, you reached your exercise goal!")
             }
             totalMotion = 0.0
         }
        print("totalMotion", totalMotion)
}
```



Verizon LTE 2:46 PM 100%

Sensor App

Motion

Acceleration X -0.0142468437552452

Acceleration Y 0.0172503292560577

Acceleration Z -0.0208932161331177

Rotation X (pitch) = 0.461378676592587

Rotation Y (pitch) = -0.122612958046466

Rotation Z (pitch) = 17.6042823791504

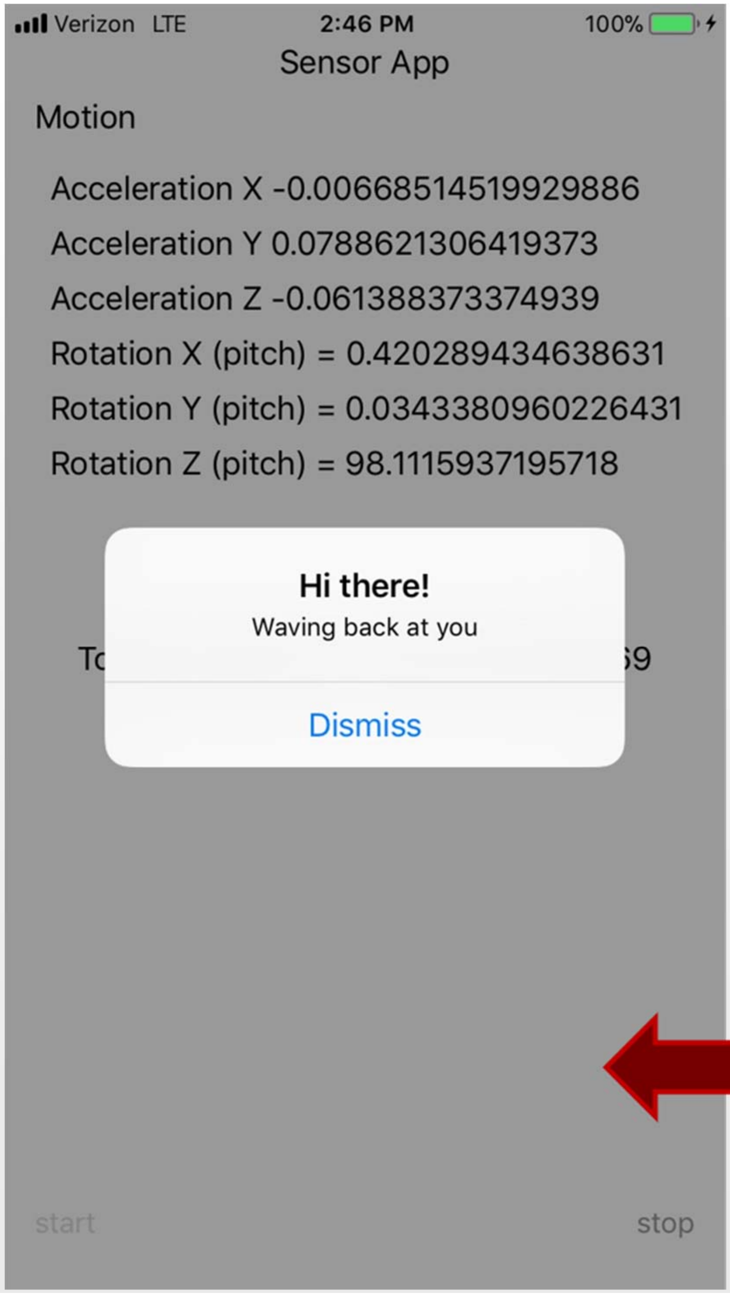
Congrats!

Congratulations, you reached your exercise goal!

Example

Try running the app

```
var totalRotation = 0.0

func processMotionData(_ dm: CMDeviceMotion) {
        let accX = dm.userAcceleration.x
        // omitting previous lines here
        if totalRotation > 1000.0 {
             DispatchQueue.main.async {
                  self.showAlert(titleText: "Hi there!",
messageText: "Waving back at you")
             }
             totalRotation = 0.0
        }
}
```



Verizon LTE 2:46 PM 100%

Sensor App

Motion

Acceleration X -0.00668514519929886

Acceleration Y 0.0788621306419373

Acceleration Z -0.061388373374939

Rotation X (pitch) = 0.420289434638631

Rotation Y (pitch) = 0.0343380960226431

Rotation Z (pitch) = 98.1115937195718

Hi there!

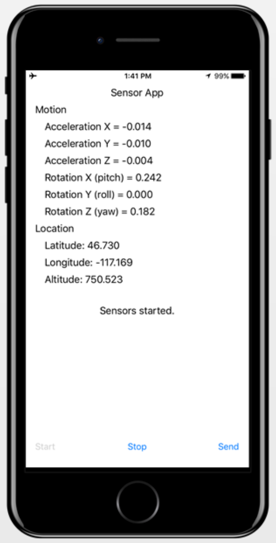
Waving back at you

Example

Try running the app

Review

- iOS, Xcode, Swift
- Sensors
- CoreMotion and CoreLocation frameworks
- Alerts
- iOS Communication over the Web
- Sensor App



Sensor App

Motion

Acceleration X = -0.014

Acceleration Y = -0.010

Acceleration Z = -0.004

Rotation X (pitch) = 0.242

Rotation Y (roll) = 0.000

Rotation Z (yaw) = 0.182

Location

Latitude: 46.730

Longitude: -117.169

Altitude: 750.523

Sensors started.

Activity Recognition

2015-09-09	21:57:08	Yaw	-1.7854
2015-09-09	21:57:08	Pitch	0.1206
2015-09-09	21:57:08	Roll	0.8295
2015-09-09	21:57:08	AccelerationX	0.0380
2015-09-09	21:57:08	AccelerationY	-0.0987
2015-09-09	21:57:08	AccelerationZ	0.0860
2015-09-09	21:57:08	Latitude	46.7388
2015-09-09	21:57:08	Longitude	-117.1720
2015-09-09	21:57:08	Altitude	729.6390
2015-09-09	21:57:08	Course	286.5230
2015-09-09	21:57:08	Speed	0.0000
2015-09-09	21:57:09	Yaw	-1.6375
2015-09-09	21:57:09	Pitch	0.2450
2015-09-09	21:57:09	Roll	0.6756
2015-09-09	21:57:09	AccelerationX	-0.0029
2015-09-09	21:57:09	AccelerationY	0.0067
2015-09-09	21:57:09	AccelerationZ	-0.0120
2015-09-09	21:57:09	Latitude	46.7388
2015-09-09	21:57:09	Longitude	-117.1720
2015-09-09	21:57:09	Altitude	729.6810
2015-09-09	21:57:09	Course	241.1720
2015-09-09	21:57:09	Speed	0.0000
2015-09-09	21:57:10	Yaw	-1.2686
2015-09-09	21:57:10	Pitch	0.1570
2015-09-09	21:57:10	Roll	-0.3932
2015-09-09	21:57:10	AccelerationX	0.1057
2015-09-09	21:57:10	AccelerationY	-0.1179
2015-09-09	21:57:10	AccelerationZ	-0.0262
2015-09-09	21:57:10	Latitude	46.7388
2015-09-09	21:57:10	Longitude	-117.1720
2015-09-09	21:57:10	Altitude	729.6500
2015-09-09	21:57:10	Course	273.8670
2015-09-09	21:57:10	Speed	0.0000

v = <*251, 3, 21, 1260, 75612, -1.0539, -1.7854, -6.8586, -1.3717, -1.2686, 0.3238, 0.2718, -1.2686, 1, 1, -0.5343, -0.3292, -2.1163, 9.8273, 1.2776, 1.9655, -0.7784, -0.2150, …*>

$AR: v \rightarrow Eat$