Settings

Mobile Application Development in iOS
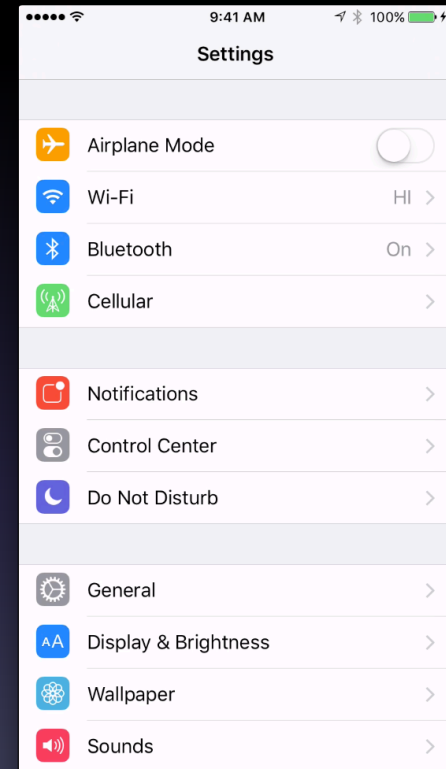
School of EECS

Washington State University

Instructor: Larry Holder

Outline

- **In-app settings on device**
  - UserDefaults

- **Global settings on device**
  - Settings Bundle

- **Settings across devices**
  - iCloud key-value store

UserDefaults

- Persistent key-value store for app
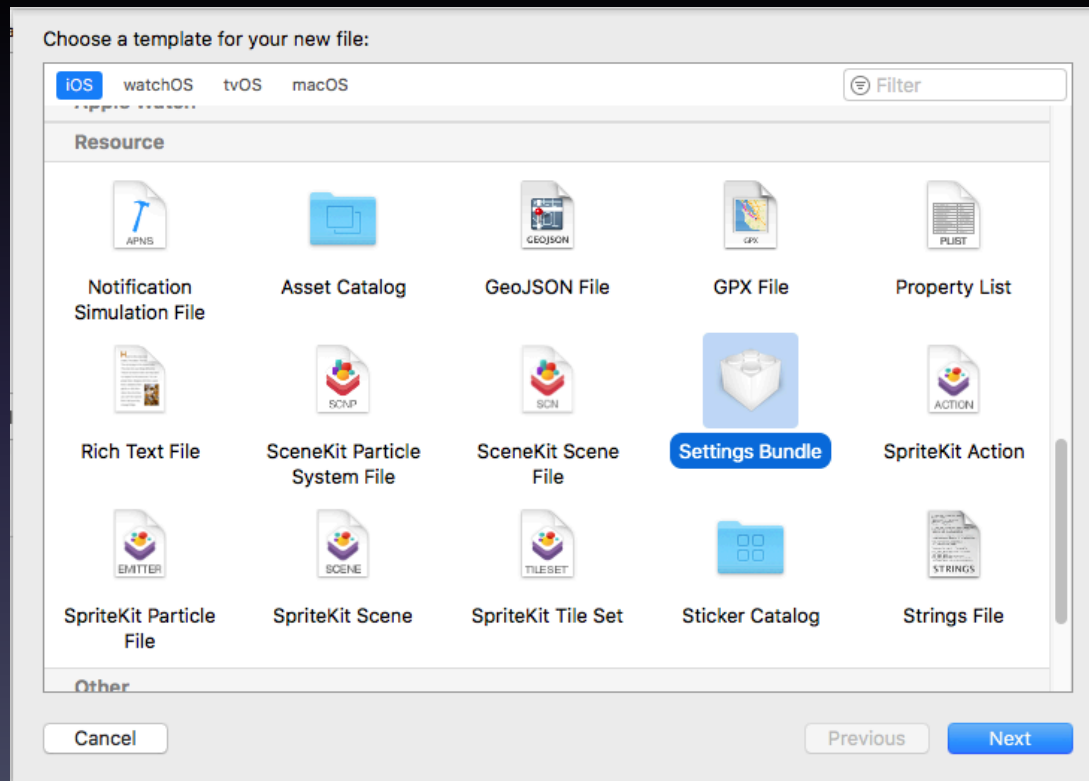
- Accessing user defaults

```
if (UserDefaults.standard.object(forKey: "volume") != nil) {
  self.volume = UserDefaults.standard.integer(forKey: "volume")
} else {
  self.volume = 10
}
```

- Setting user defaults

```
let volume = 15
UserDefaults.standard.set(volume, forKey: "volume")
```
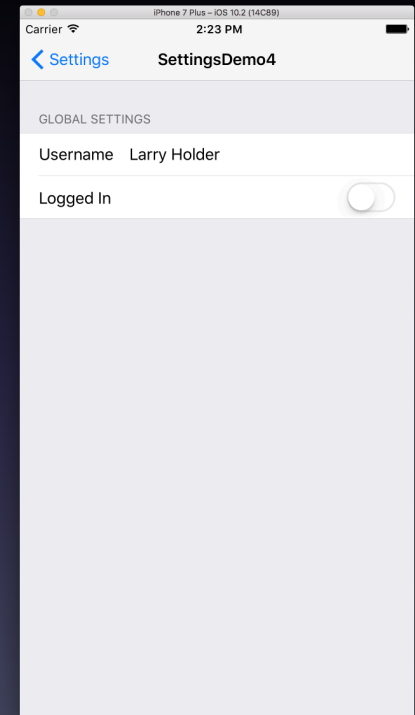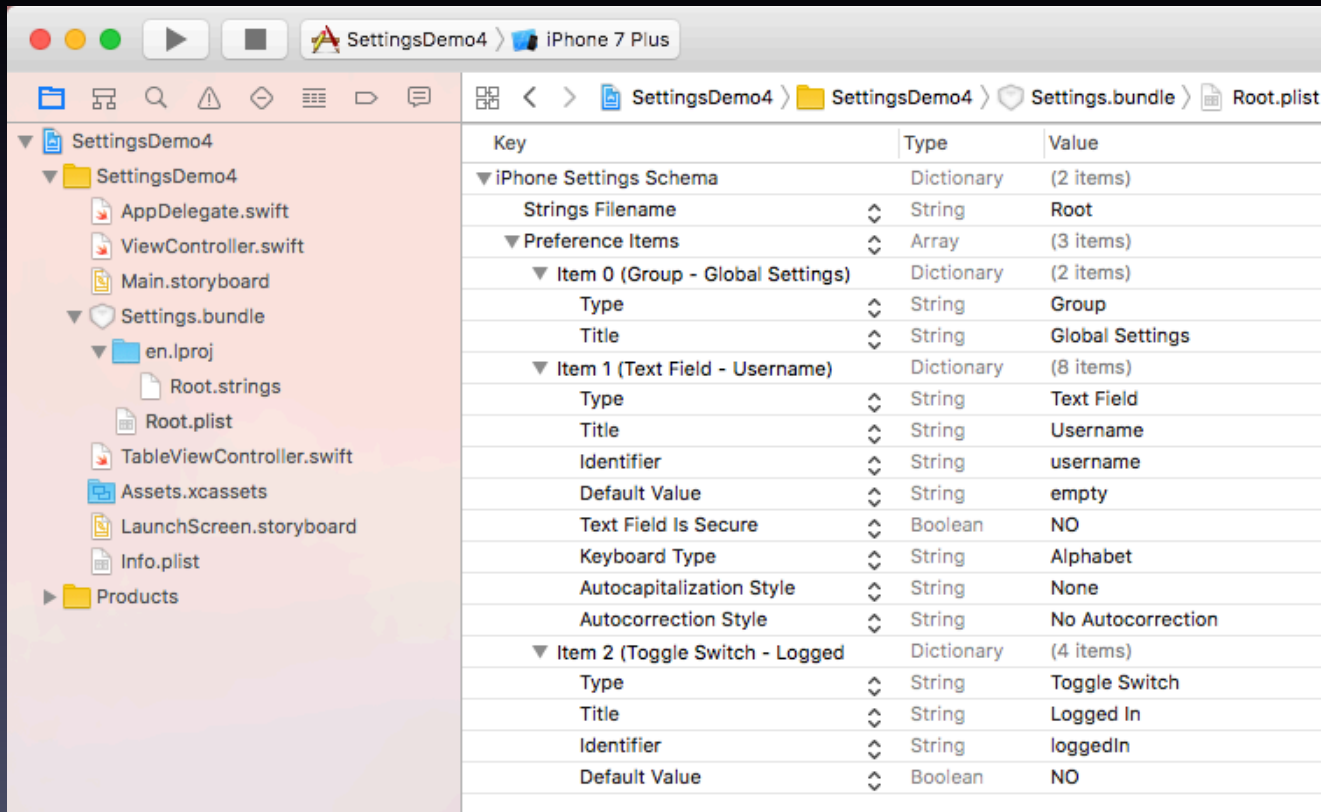
Settings Bundle

- Add Settings Bundle to app

Settings Bundle

- Modify root.plist to create settings

Settings Bundle

- Set in device's Global Settings

- Access in UserDefaults as before

```
if (UserDefaults.standard.object(forKey: "username") != nil) {
  self.username = UserDefaults.standard.string(forKey: "username")
} else {
  self.username = ""
}
```

- Notify app when changed (next time…)

iCloud Key-Value Store

- Settings across apps & devices

- Add iCloud capabilities to app

- Need an App ID with iCloud entitlements

iCloud Key-Value Store (1)

- Create new project

- Give it a unique name, not previously used to generate App ID

- In Capabilities, turn on iCloud
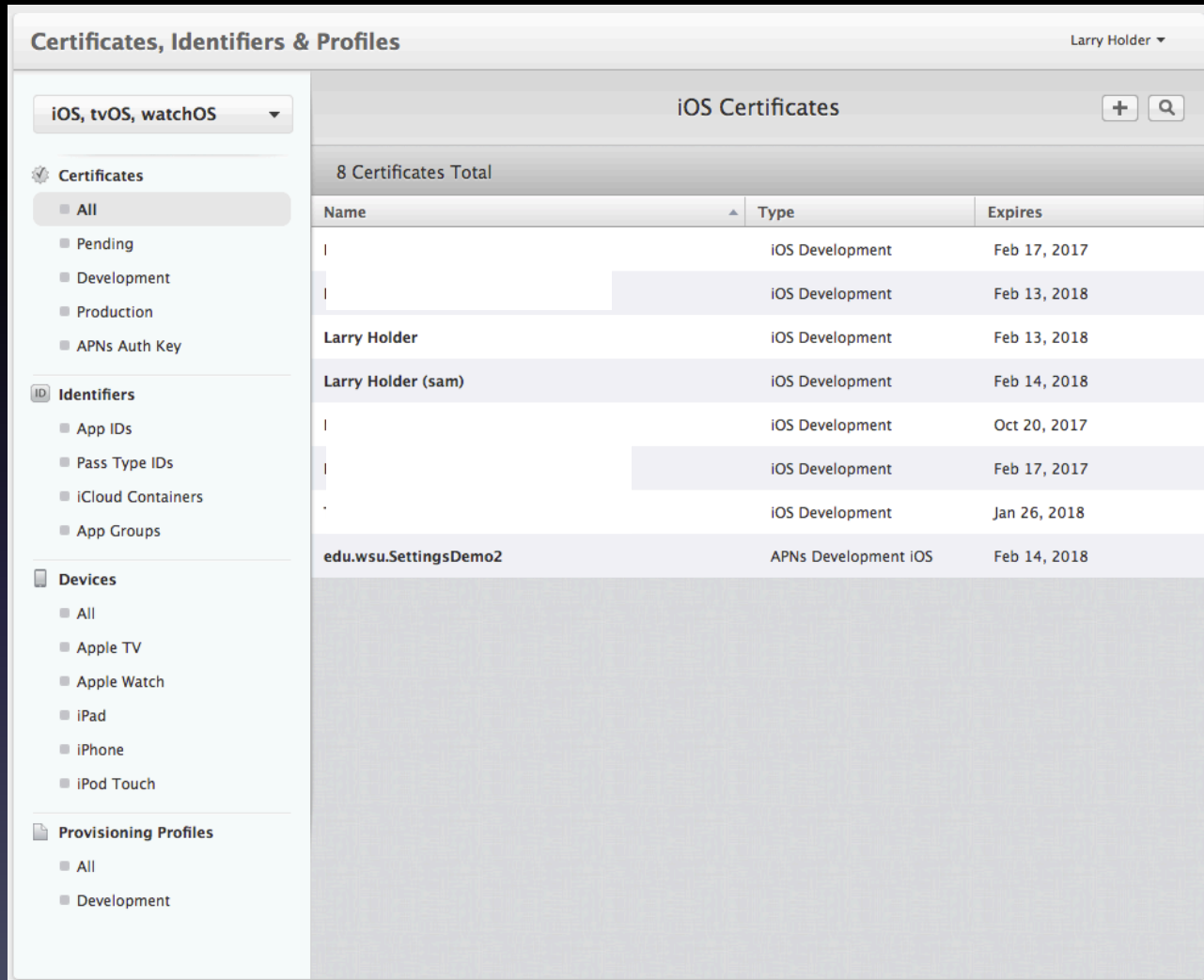 - Key-value storage already checked
  - Also check CloudKit

iCloud Key-Value Store (1)
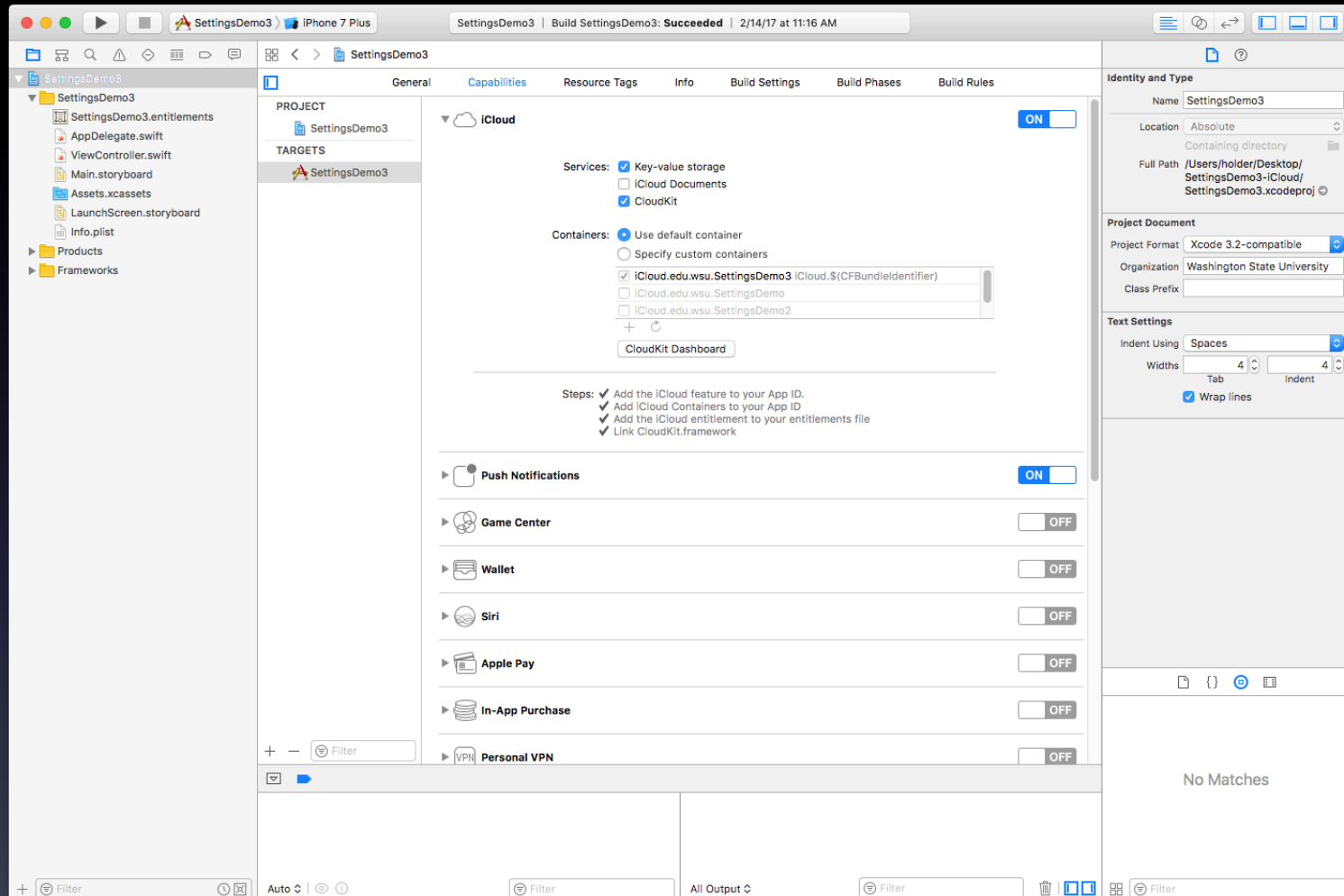
iCloud Key-Value Store (2)

- Login to developer account

- Will need at least one real device

- Make sure device is registered on developer account

- Under Certificates, make sure you have one for your development environment
  - Should be auto-generated

- Create iCloud Container and register
  - Must have same name as app

iCloud Key-Value Store (2a)

iCloud Key-Value Store (2b)

iCloud Key-Value Store (3)

- Create explicit App ID for this app

  – Wildcard App ID won't work for iCloud use

  – Check iCloud App Services (need CloudKit)

  – Select and assign your iCloud container

  – Register

iCloud Key-Value Store (3a)

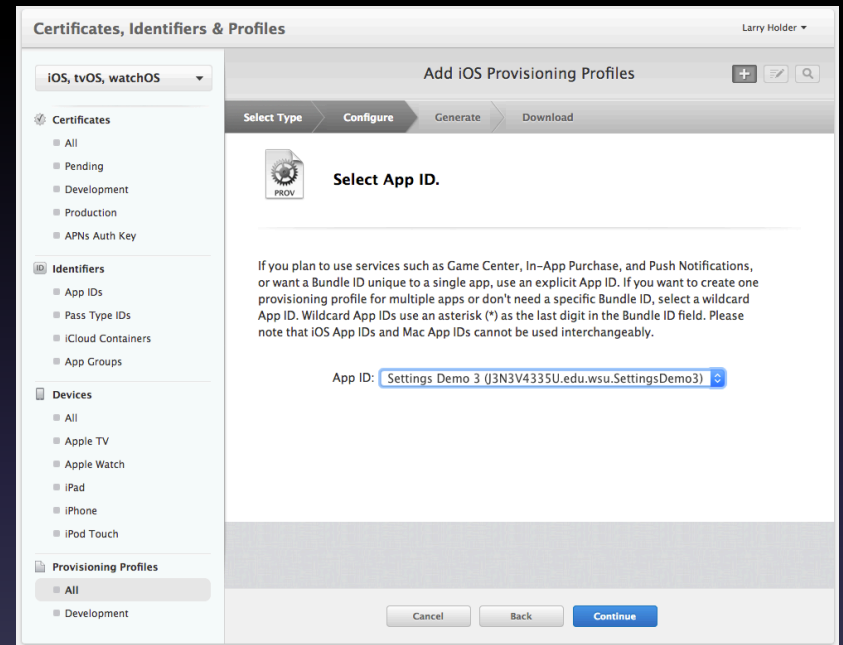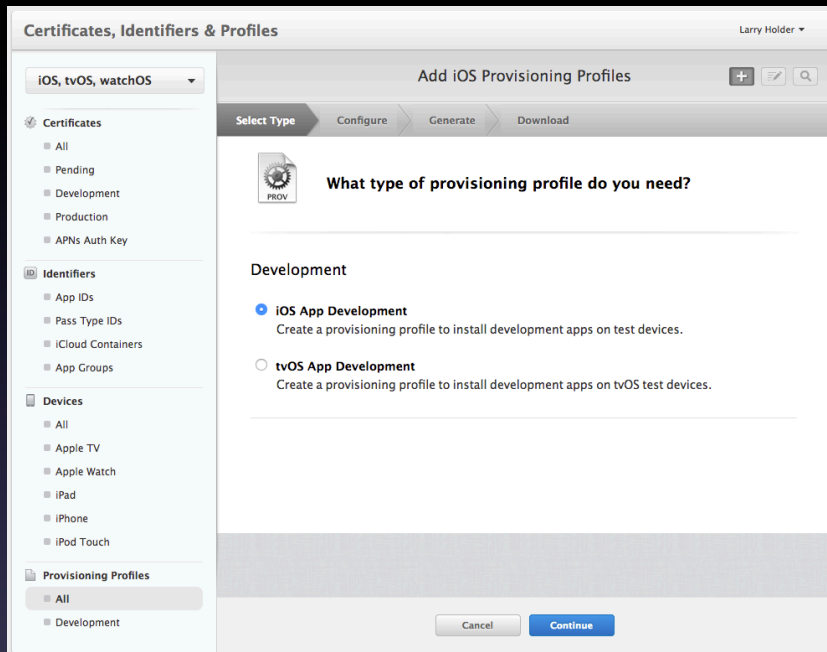# iCloud Key-Value Store (3b)
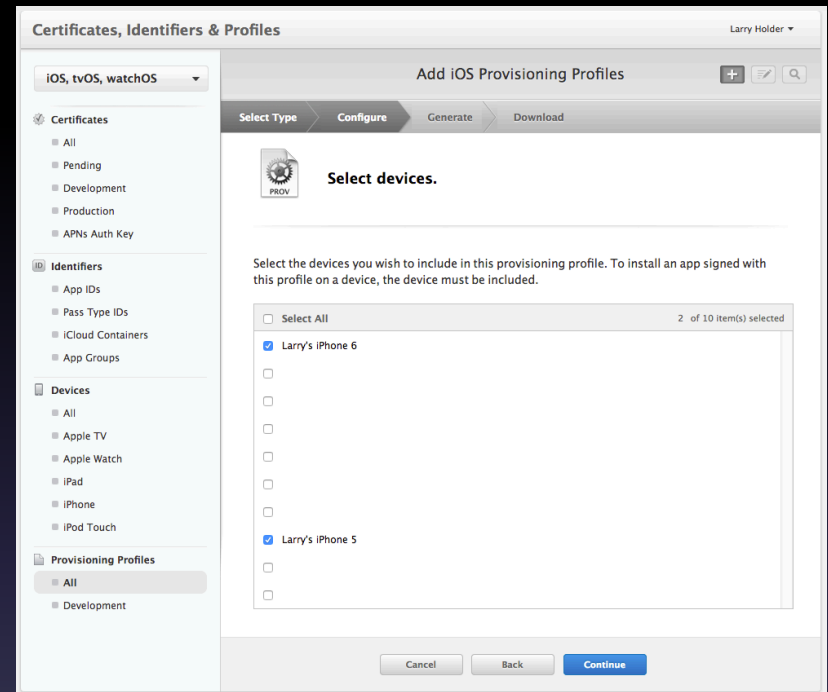
iCloud Key-Value Store (4)

- Create Provisioning Profile
  - iOS App Development
  - Select newly-created App ID
  - Select your development certificate(s)
  - Select your devices
  - Give profile a name
  - Download, double-click to install in Xcode
- Uncheck automatically manage signing
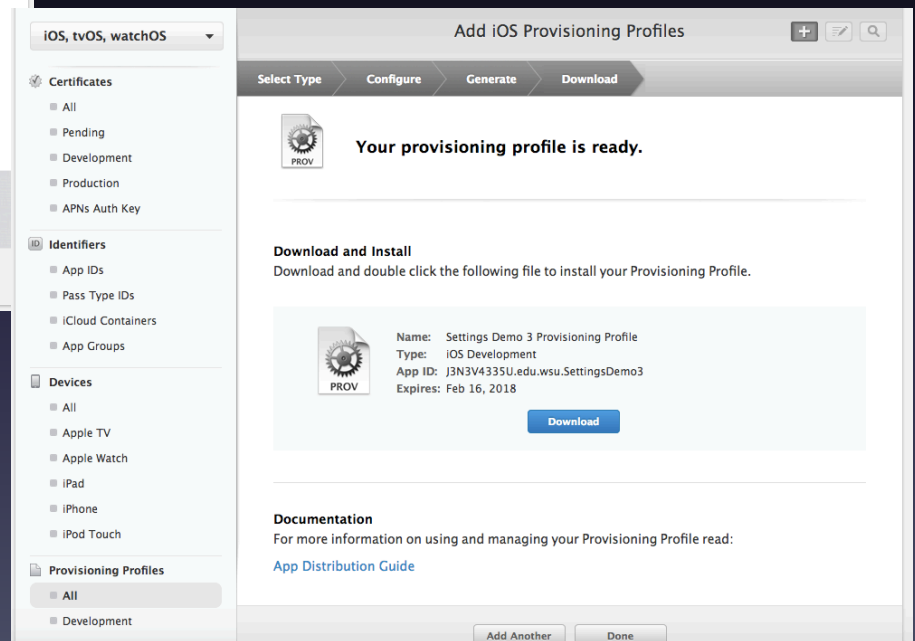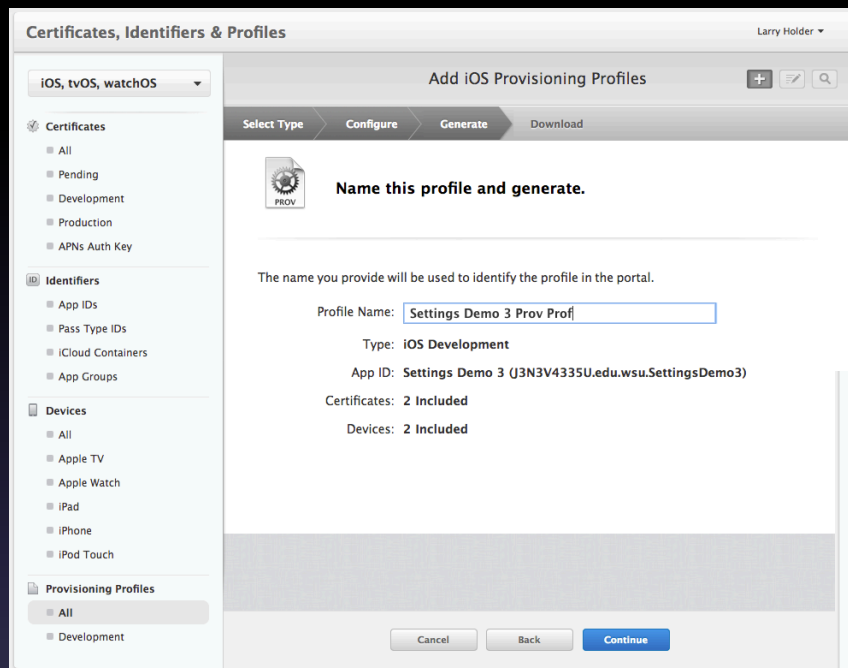- Select new provisioning profile
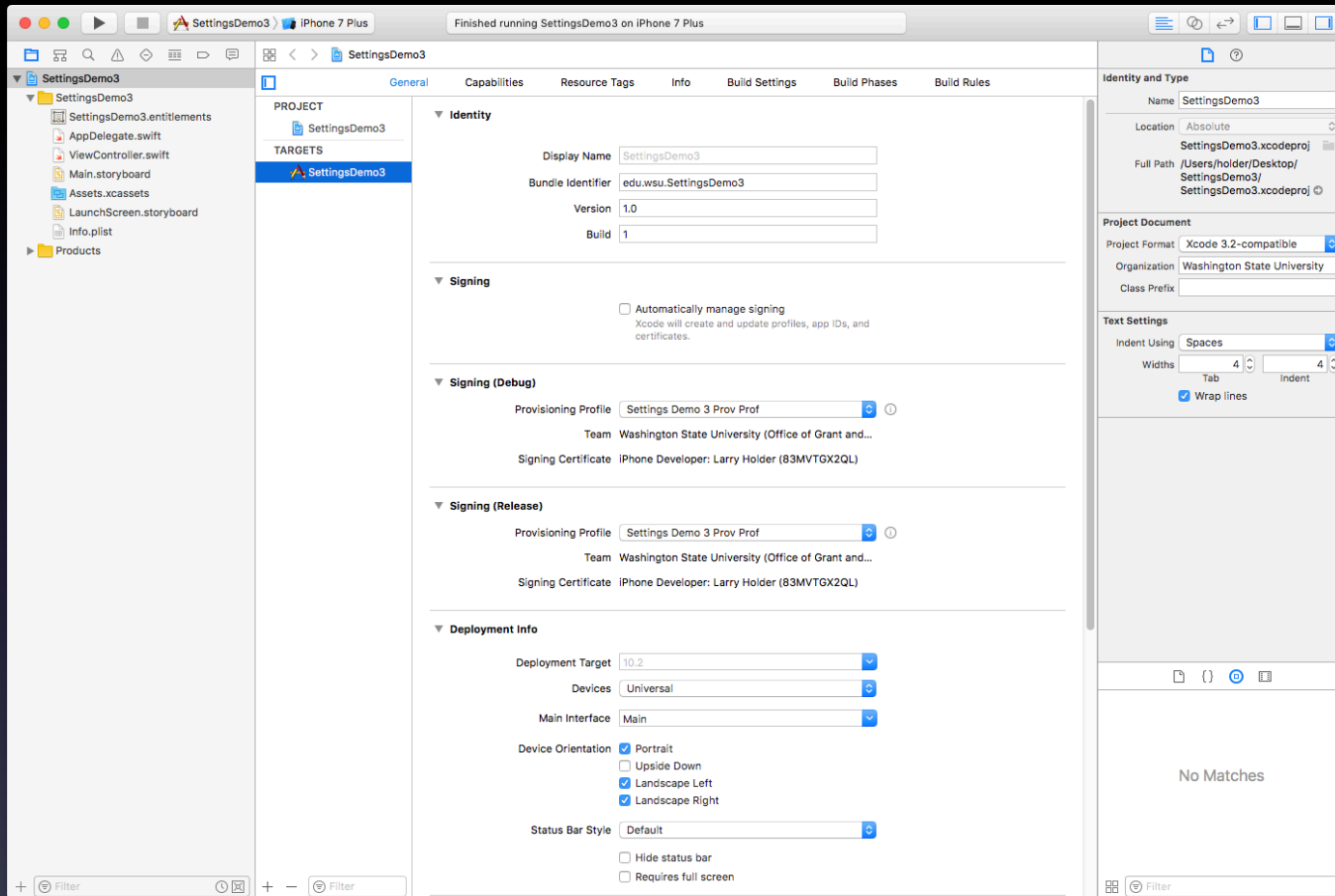
iCloud Key-Value Store (4a)

iCloud Key-Value Store (4b)
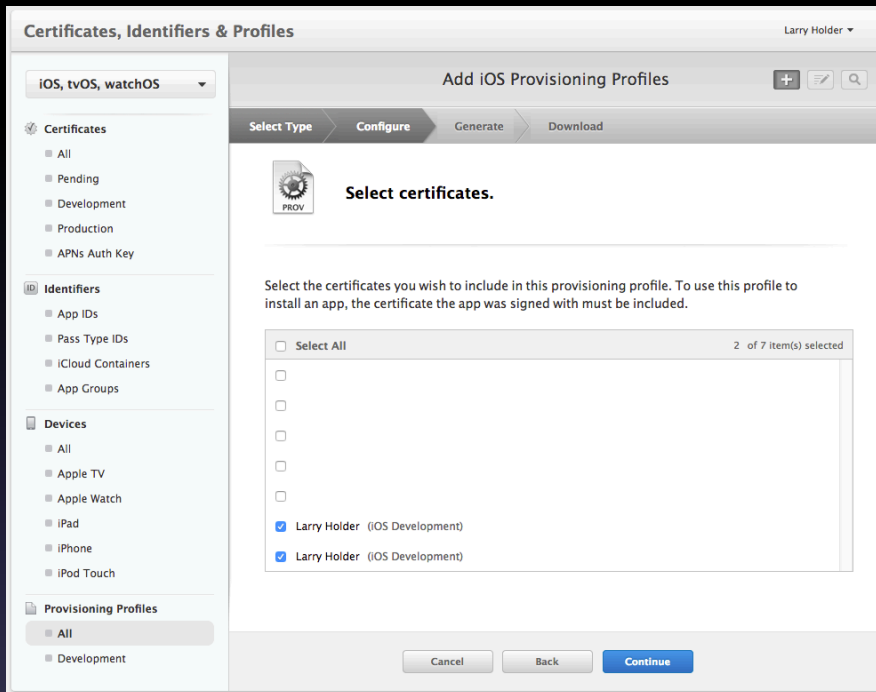
iCloud Key-Value Store (4c)

iCloud Key-Value Store (4d)

iCloud Key-Value Store (5)

- Run app on simulator and device

  – Make sure both signed in to same iCloud account

- Changes via simulator propagate to device

  – But not vice versa (not sure why?)

Code

```
// Initialize key-value store
var keyStore = NSUbiquitousKeyValueStore()

// viewDidLoad
keyStore.synchronize()

// Set a key to a value
keyStore.set("value", forKey: "keyName")
keyStore.synchronize() // optional if you can wait

// Read value of key (mind your latency)
if let storedString = keyStore.string(forKey: "keyName") {
  print("Cloud Value = \(storedString)")
} else {
  print("Cloud Value = empty")
}
```

Resources

- Preferences and Settings

  - developer.apple.com/library/content/documentation/Cocoa/Conceptual/UserDefaults