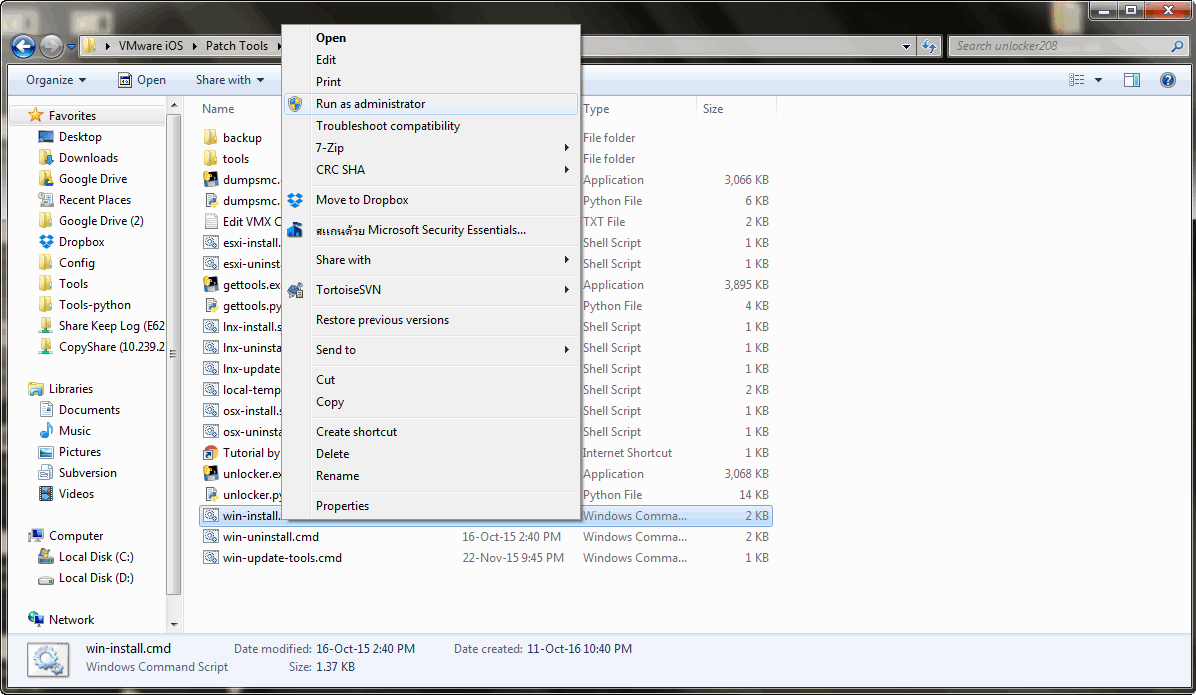
**How to install VMware on window and patch iOS for VMware**

**Step1: Install VMware on window**

* Install VMware-workstation 12
* Add Serial Key on VMware-workstation 12

**Step2: Patch the VMware**

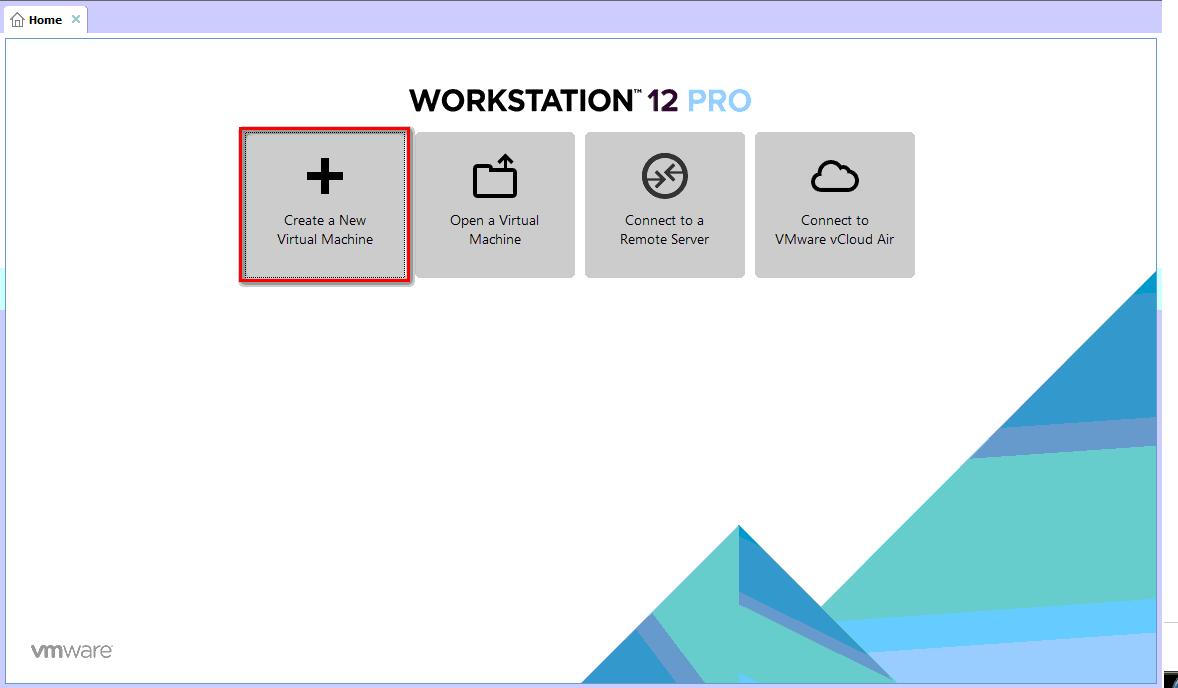
* Install patch iOS for VMware (Patch Tools > unlocker208 > win-install.cmd)



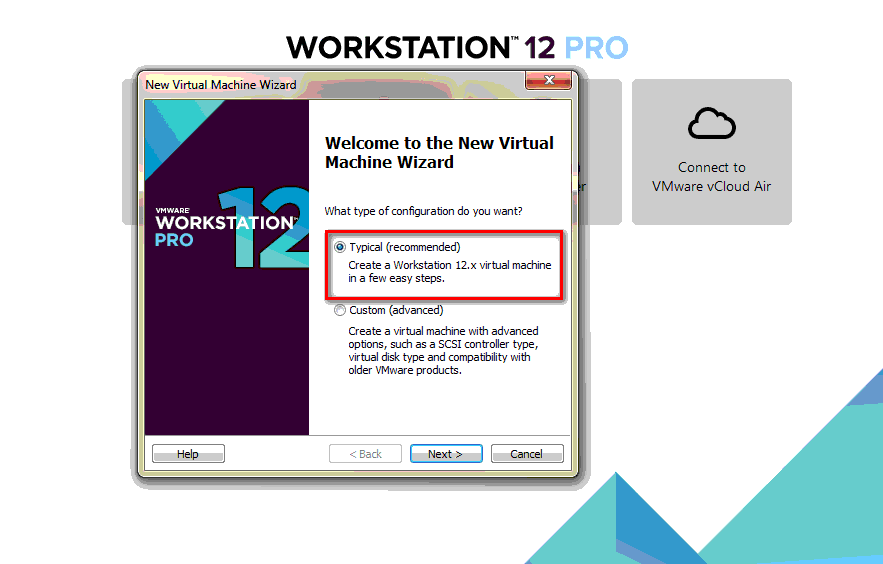
**How to install MacOS Sierra on VMware Window (Patch iOS)**

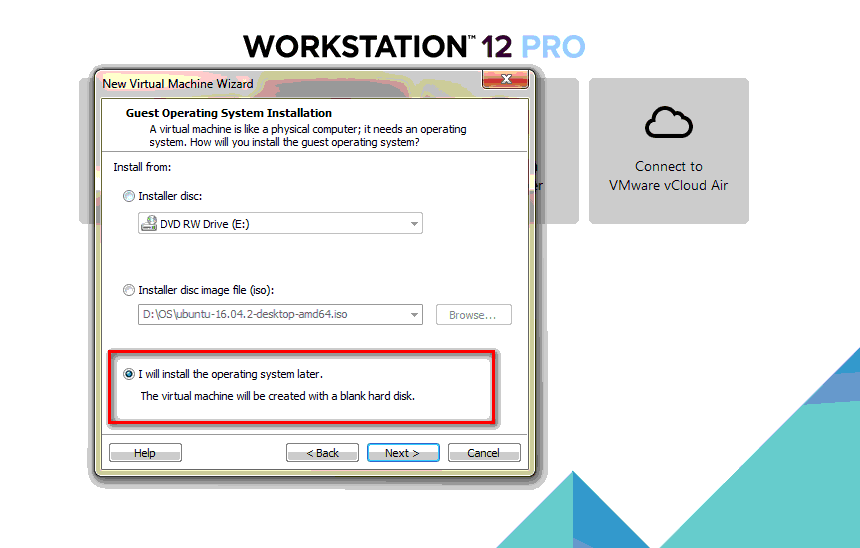
**Step1: Create New Virtual Machine**

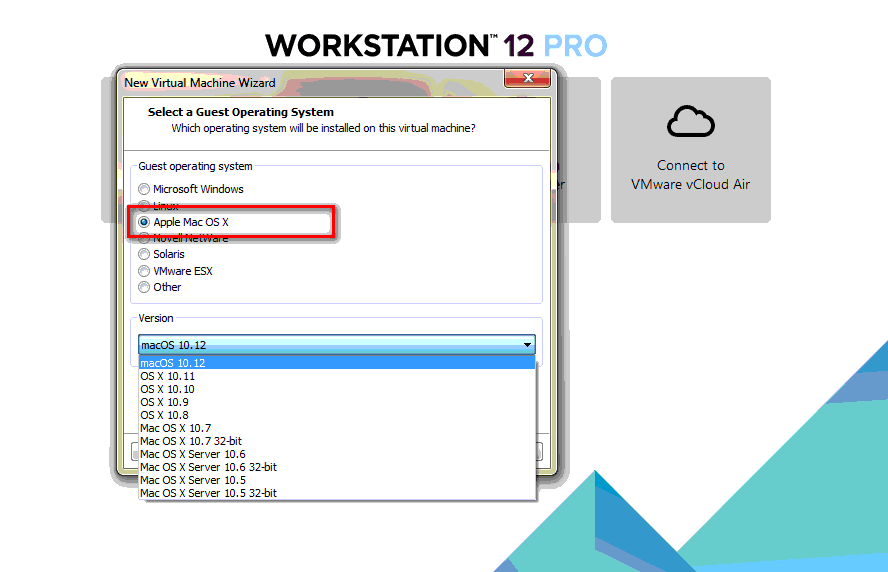
* Select Create a New Virtual Machine

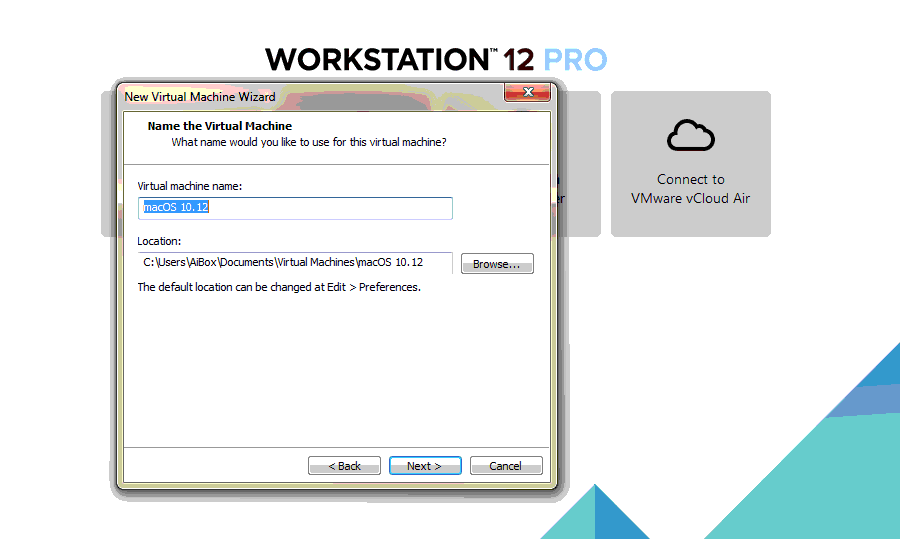


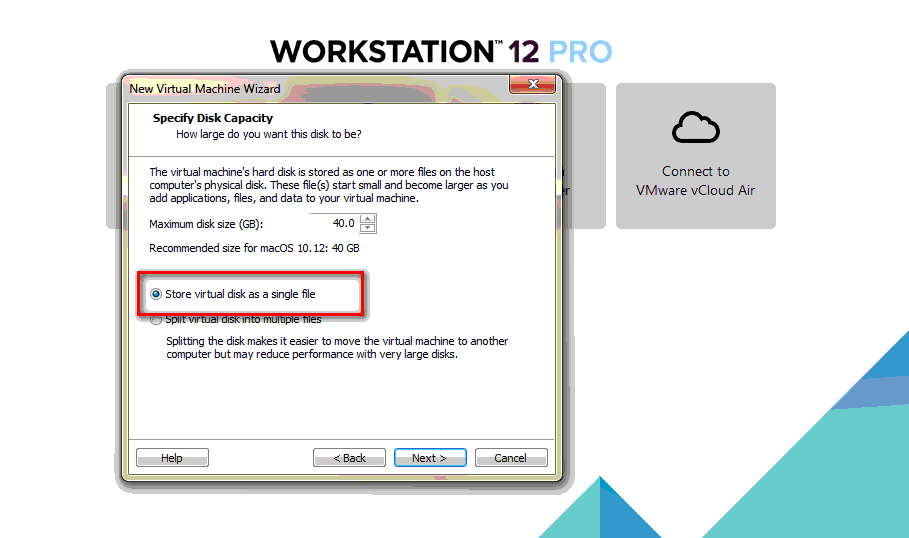
* Select configuration -> Typical (recommended) option -> Next





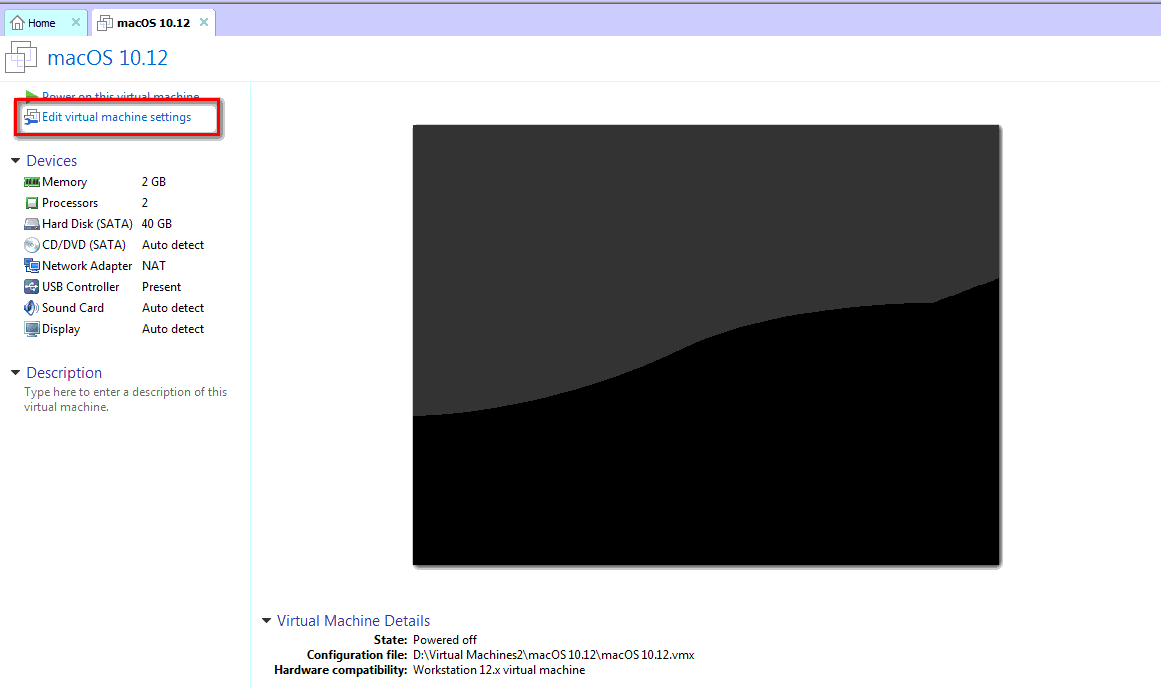




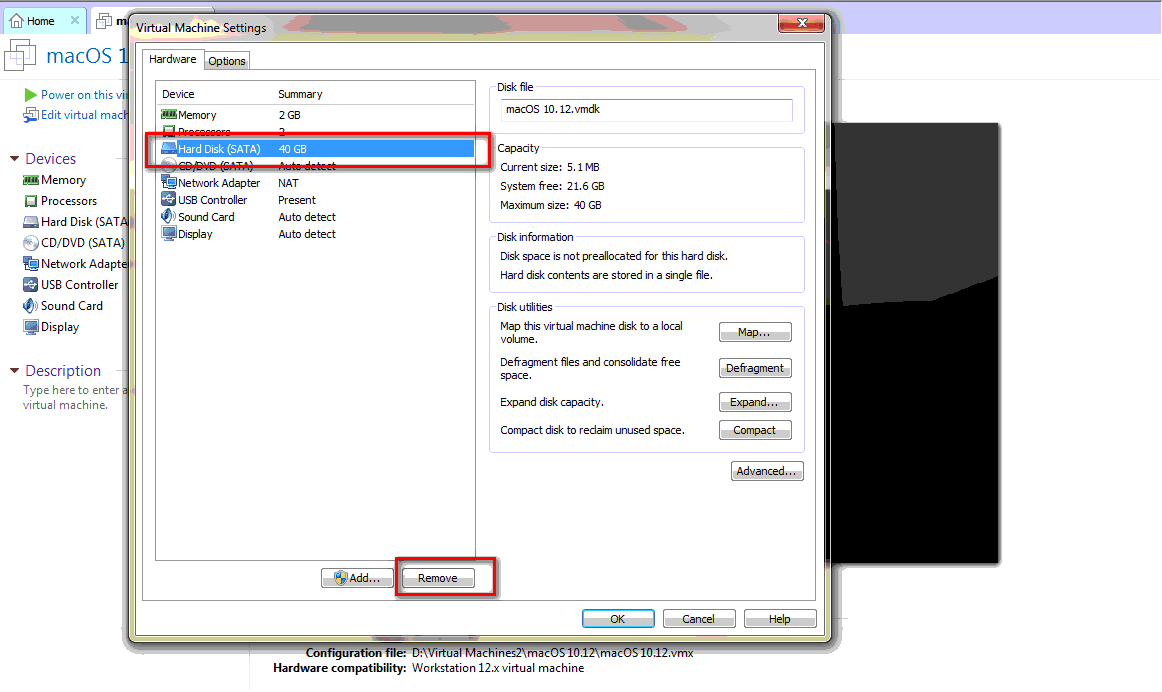


**Step2: Edit Virtual Machine**

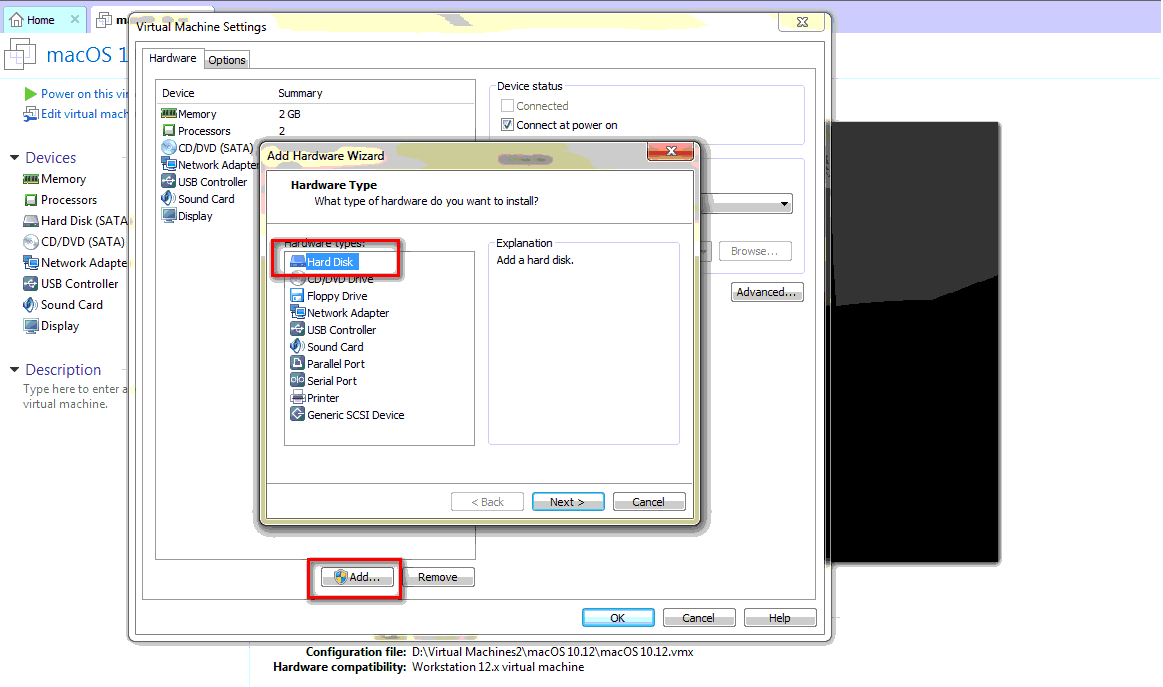
* Select Edit Virtual Machine Settings

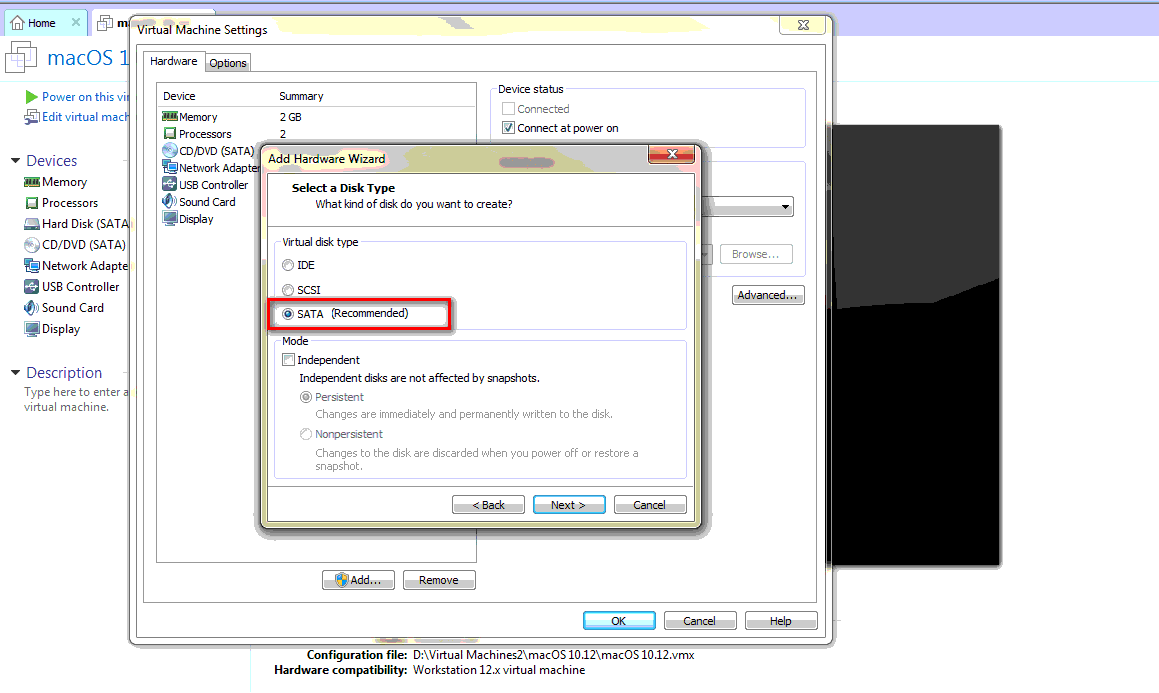


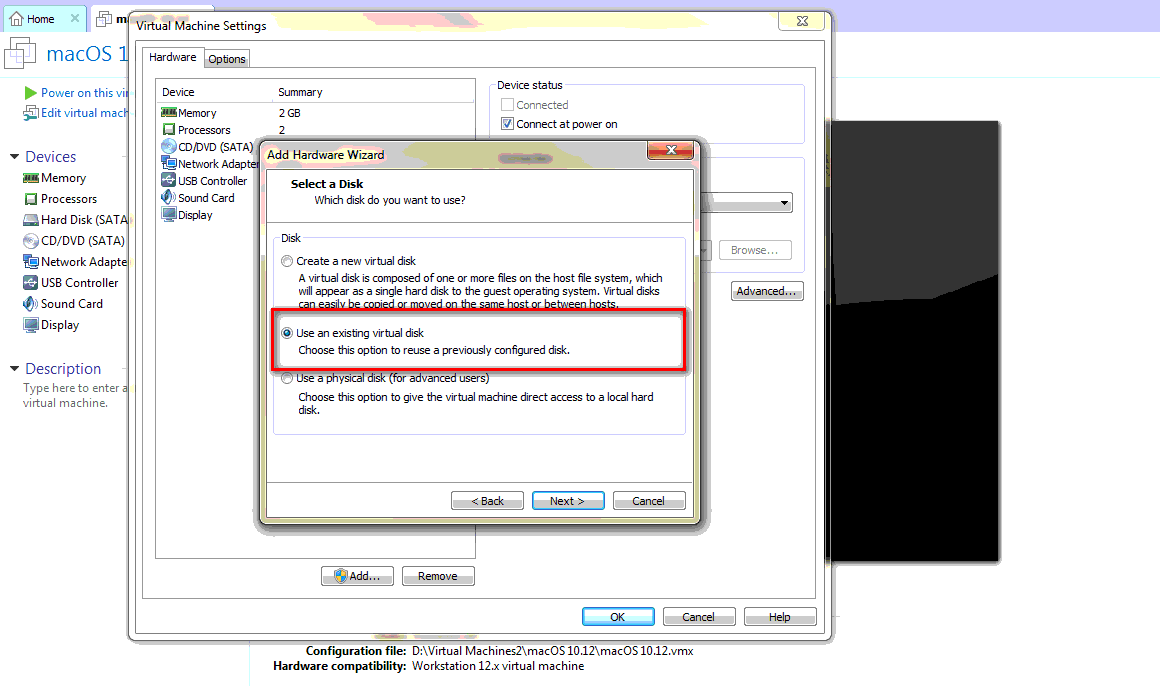
* Remove Hard Disk



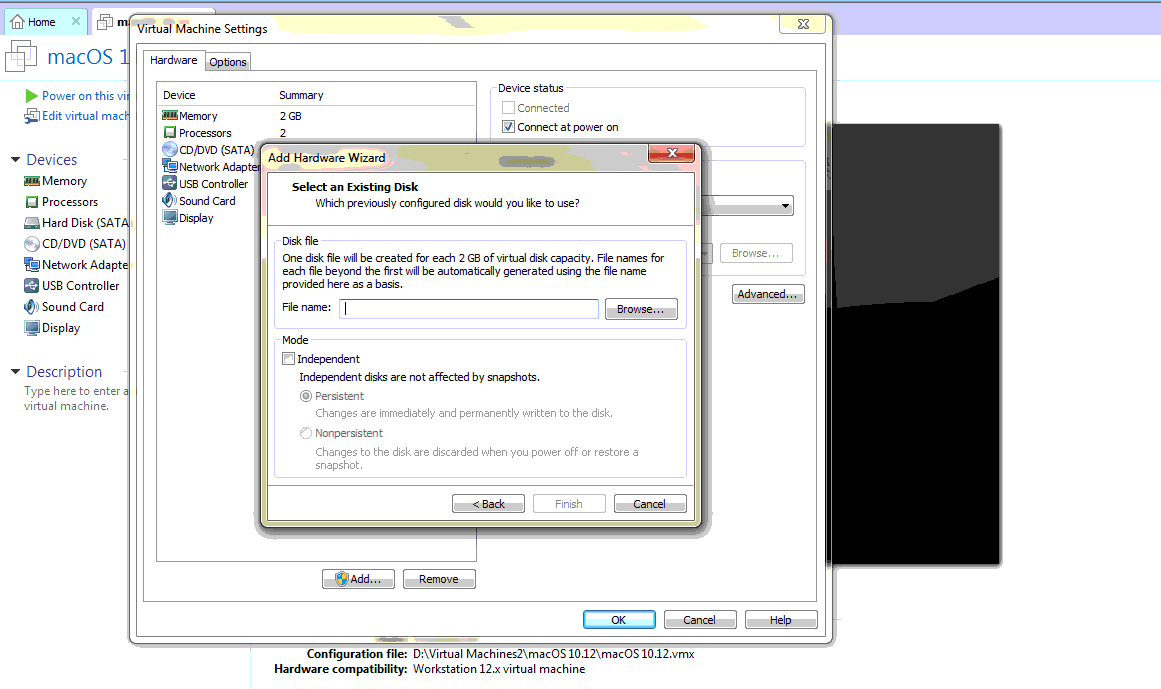
* Add new Hard Disk (macOS 10.12 Sierra)



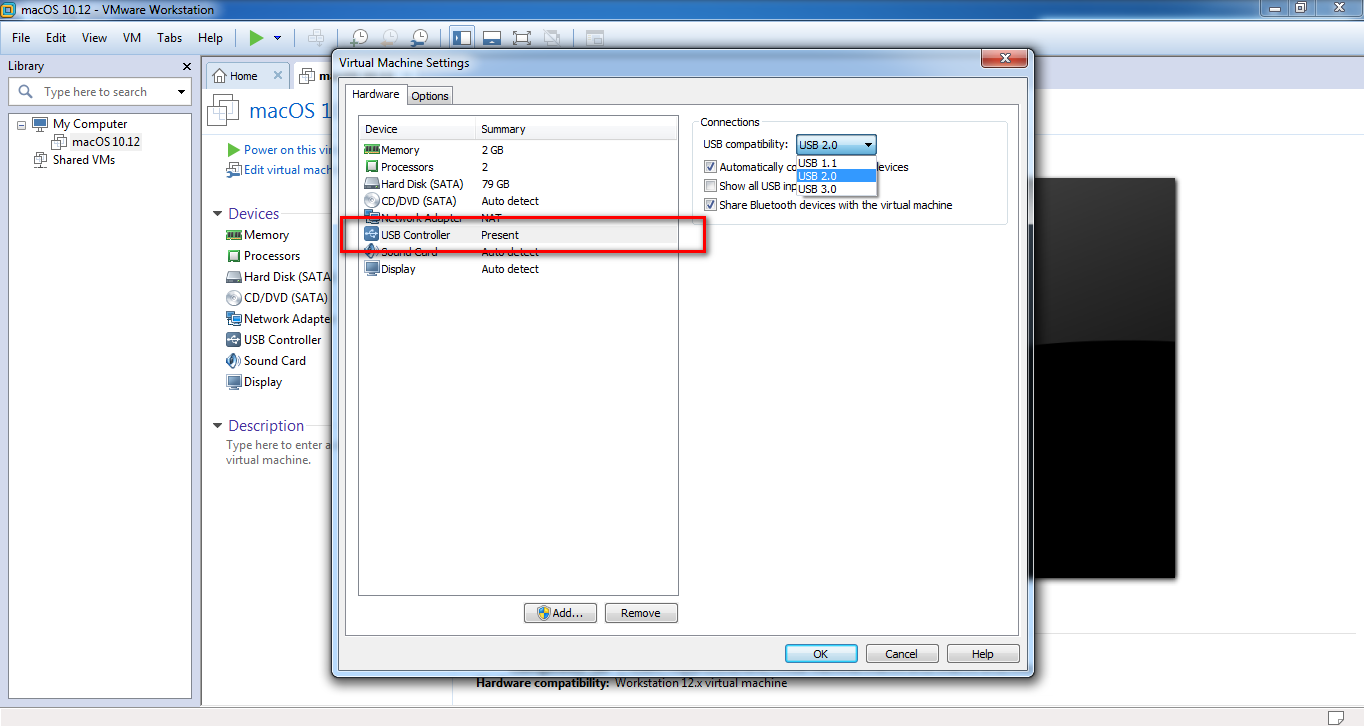




* Mount file macOS 10.12 Sierra (full setting image)

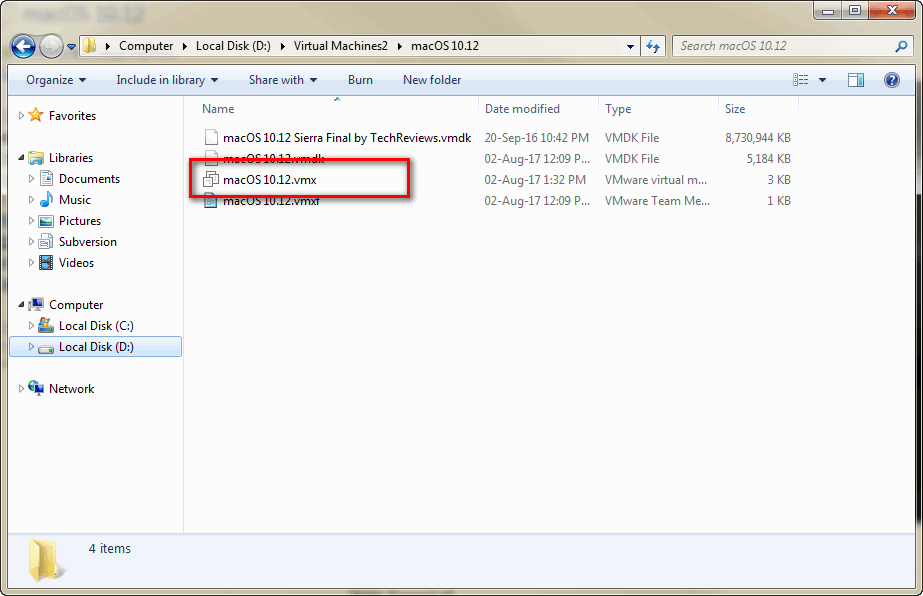


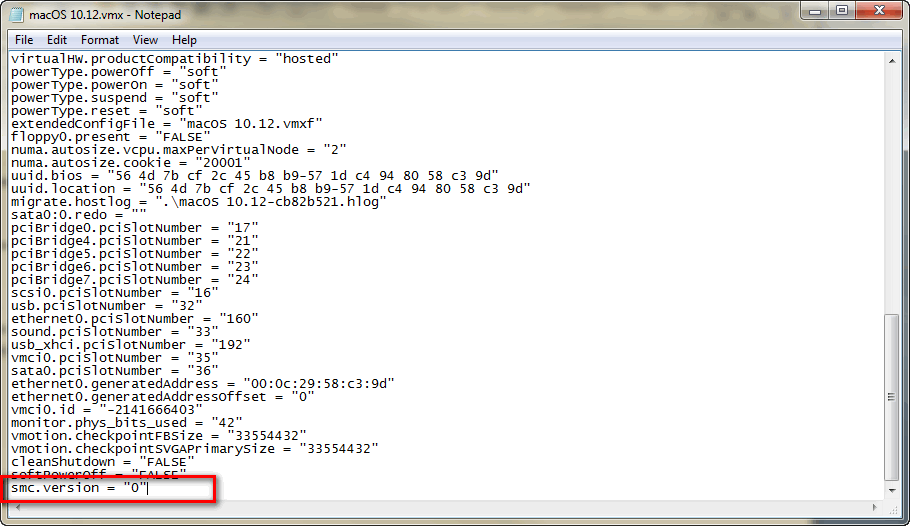
* Change settings USB controller -> USB 2.0



**Step3: Edit VMX file in folder VMware Virtual Machine**

* Edit VMX file by add -> smc.version = "0"





Reference (20/8/2017): <https://techsviewer.com/install-macos-sierra-vmware-windows/>

**How to install Robot Framework on Mac OSX**

**Configuration for Mac OSX**

1. **Python**

* Install python 2.7.x (h[ttps://www.python.org/](https://www.python.org/))

1. **Robot Framework + Appium (type command in terminal)**

* Install robotframework
* pip install robotframework
* pip install robotframework-selenium2library
* pip install robotframework-appiumlibrary
* pip install robotframework-excellibrary
* Install beautifulsoup&xlsxWriter Library for Robot Framework via Command line. Install beautifulsoup Library use pull data from Report.html
* pip install beautifulsoup4
* pip install xlsxWriter
* Install psutil module **# Update on 03/04/2017**
* pip install psutil
* Install Lock File for use in take virtual device **# Update on 02/12/2016**
* pip install lockfile

1. **Install Application**

* Install Xcode app store
* Install Snail SVN

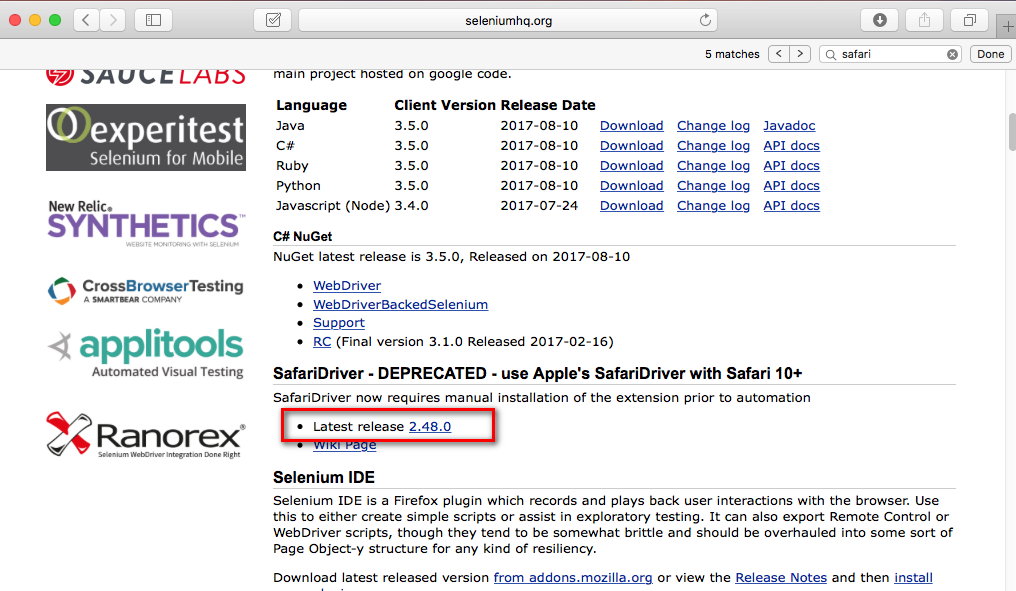
1. **Installation**

* Install brew behind the proxy or using Wifi
* ruby -e "$(curl –fsSL <https://raw.githubusercontent.com/Homebrew/install/master/install>)"
* Install Appium using Node
* brew install node
* npm install -g appium
* npm install wd # get appium client
* Start Appium (check if the appium service has started, check with the below command & the socket needs to be in Listening mode.)
* appium & # start appium
* netstat -an | grep 4723
* Set path for XcodeBuild
* sudo xcode-select –s /Applications/Xcode.app/Contents/Developer/
* Install tools
* brew install ideviceinstaller
* brew install carthage
* npm install -g ios-deploy
* npm install -g deviceconsole
* sudo gem install xcpretty
* brew install libimobiledevice --HEAD
* Check Appium for iOS
* npm install –g appium-doctor
* appium-doctor --ios

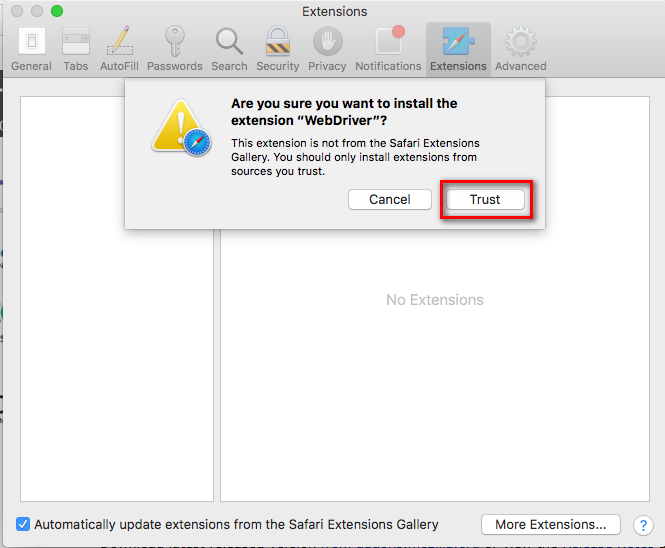
**Setting configuration for Robot framework**

1. **Config for browser Safari**

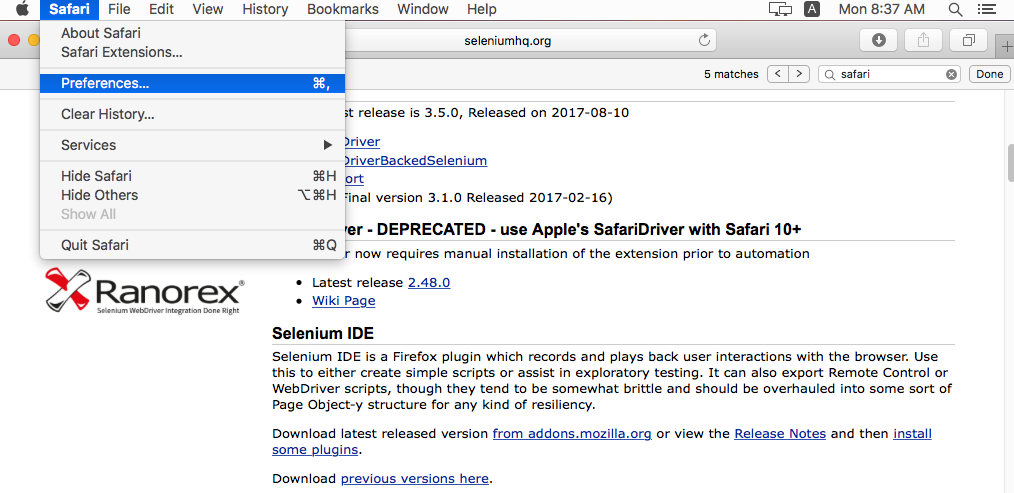
* Download selenium webdriver for browser safari from <http://www.seleniumhq.org/download/>



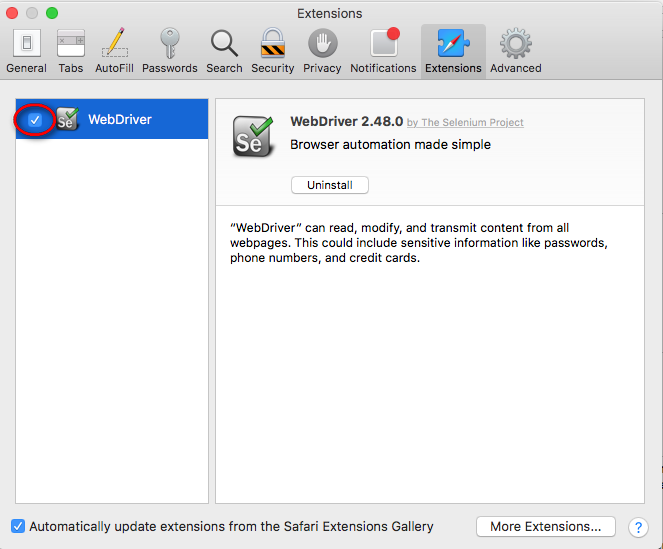
* Install and select Trust



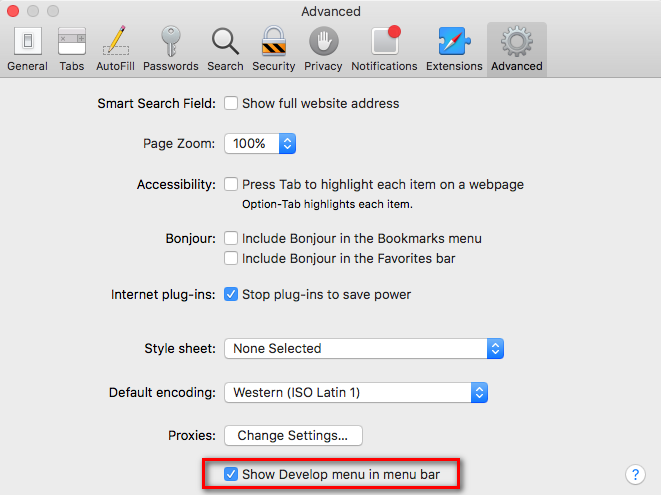
* Select menu on browser > Safari > Preferences…



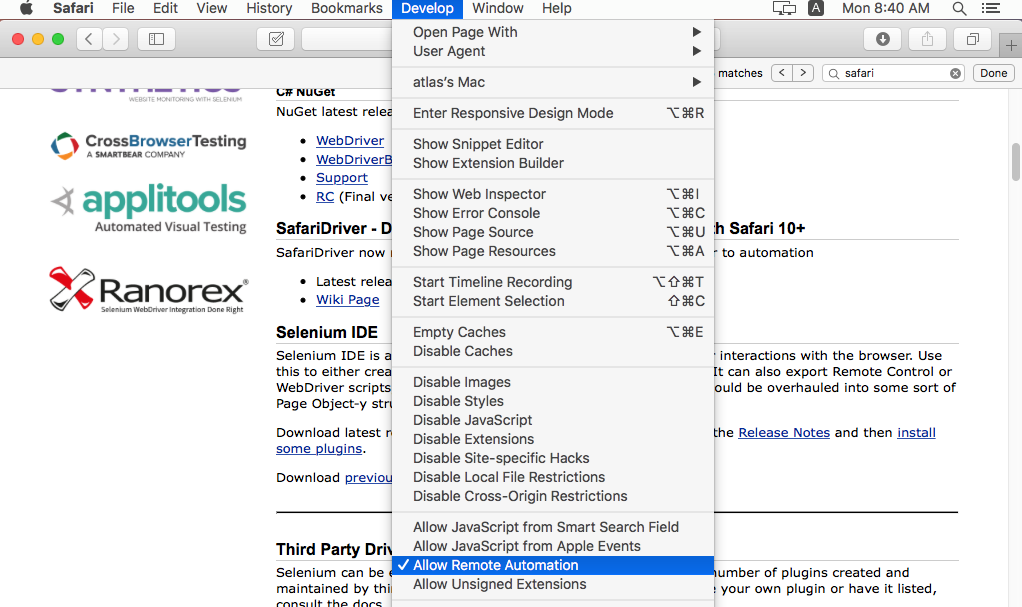
* Check WebDriver > enabled



* Set enable Show Develop menu in menu bar

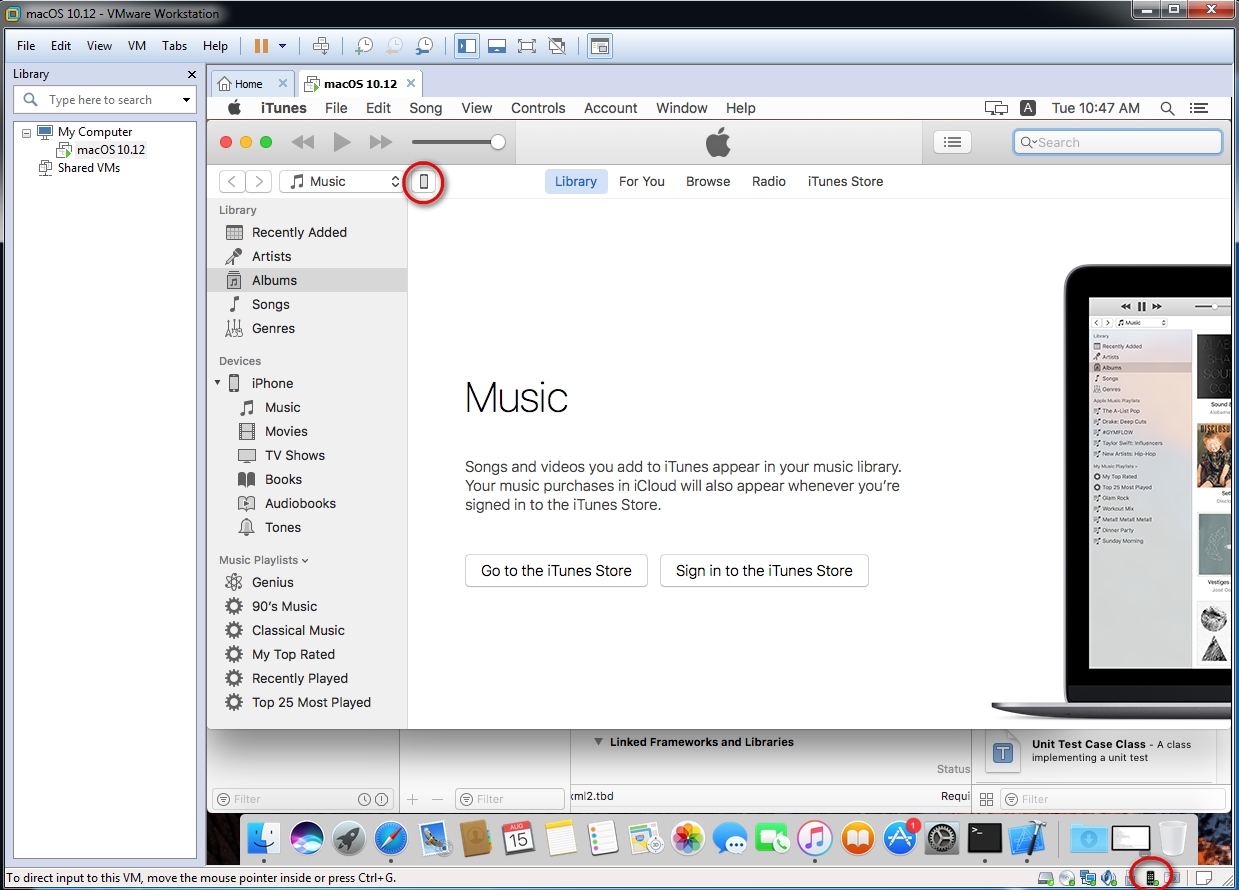


* Select menu on browser > Develop > Allow Remote Automation (enable)

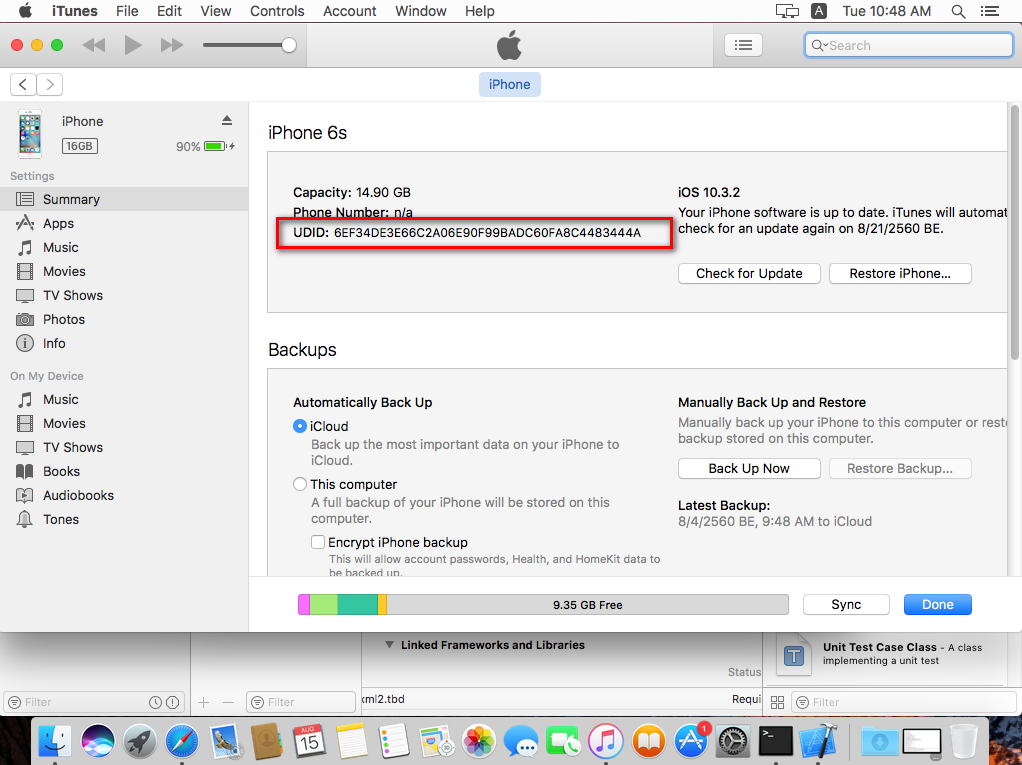


1. **Config for iPhone**

* **Check connect iPhone on VMware**
* Check connect device USB of VMware (below right)
* Select application iTune and select icon mobile



* Select Serial Number for change to UDID



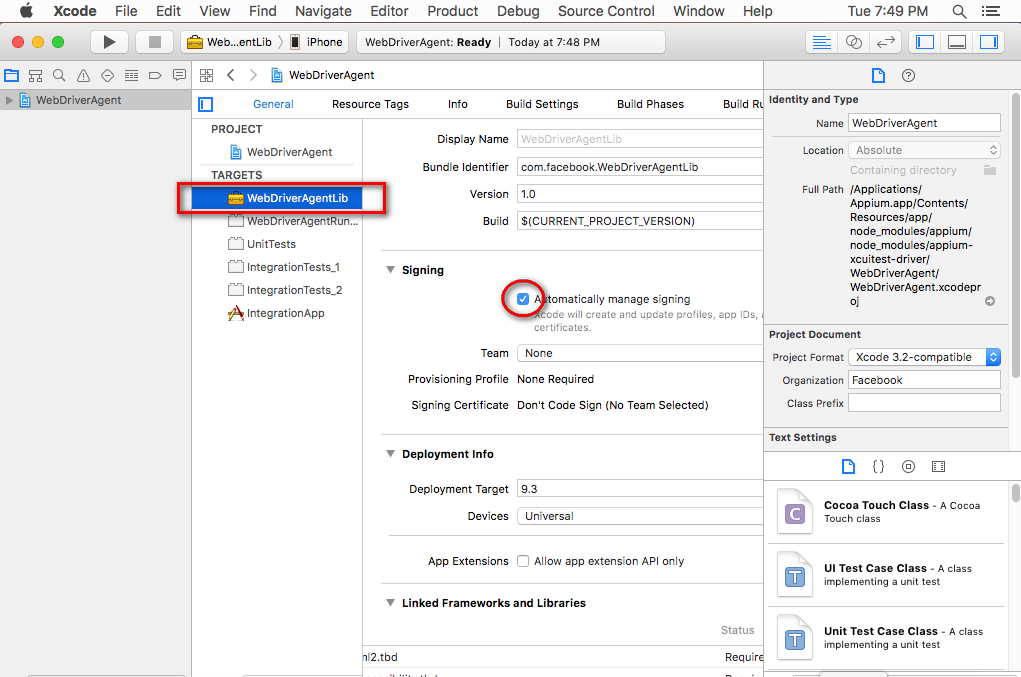
* **Set appium terminal (command)**
* cd /usr/local/lib/node\_modules/appium/node\_modules/appium-xcuitest-driver/WebDriverAgent/
* ./Scripts/bootstrap.sh
* open -a Xcode WebDriverAgent.xcodeproj
* **Set appium GUI**
* Download Appium GUI
* Move Appium to Application path

**(command)**

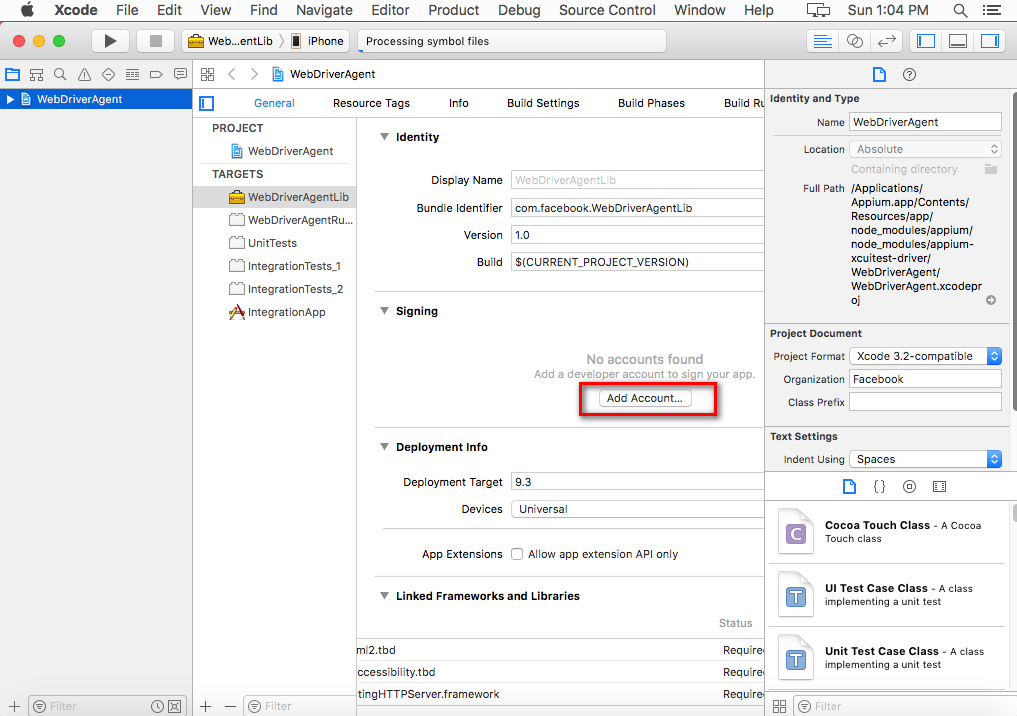
* cd /Applications/Appium.app/Contents/Resources/app/node\_modules/appium/node\_modules/appium-xcuitest-driver/WebDriverAgent
* bash Scripts/bootstrap.sh
* open -a Xcode WebDriverAgent.xcodeproj

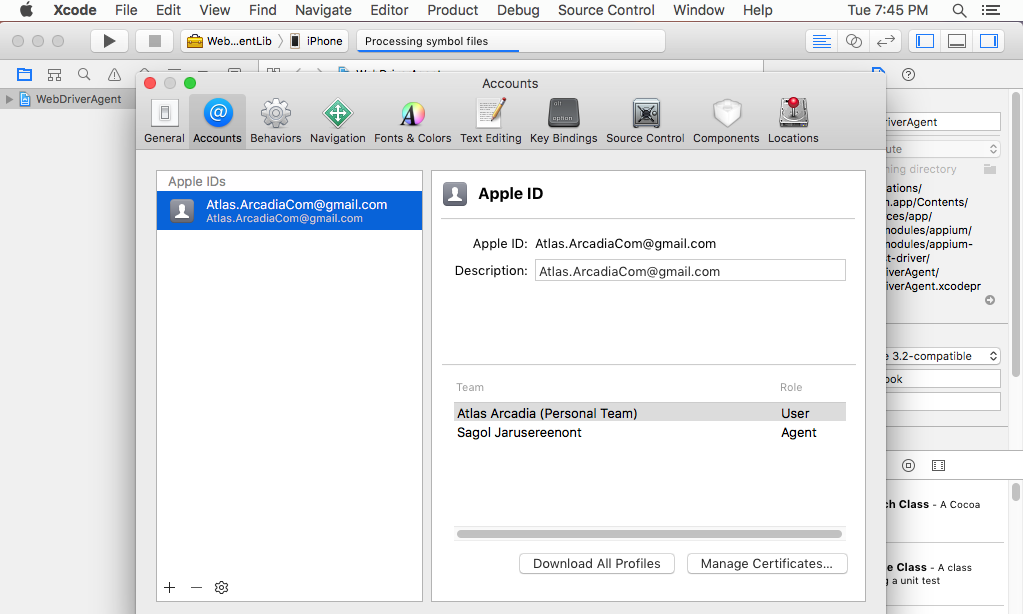
**Set Xcode for build iPhone (set iPhone for run automated via Appium)**

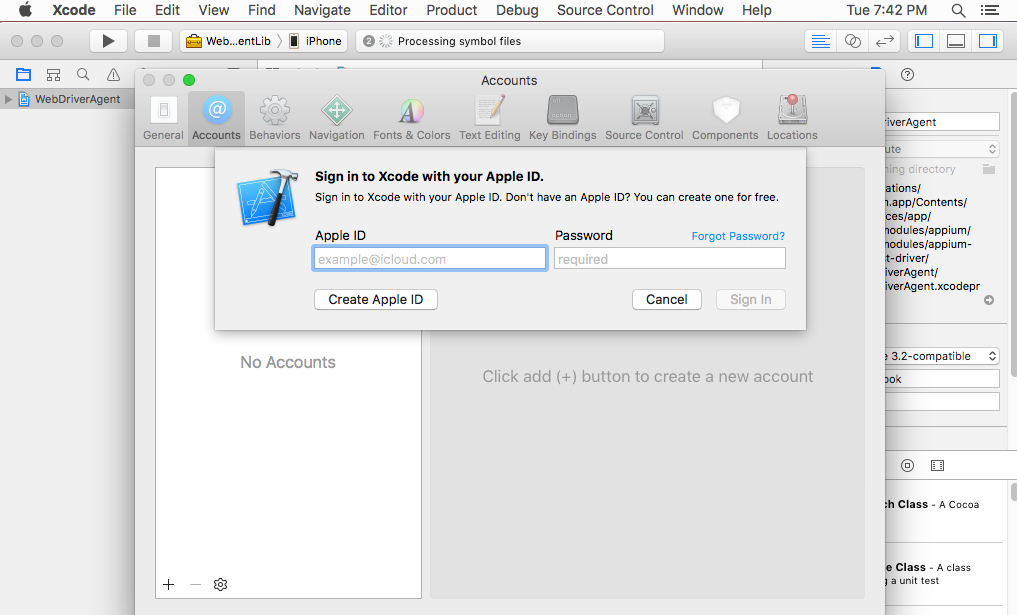
* Select WebDroverAgentLib > enable Automatically manage signing



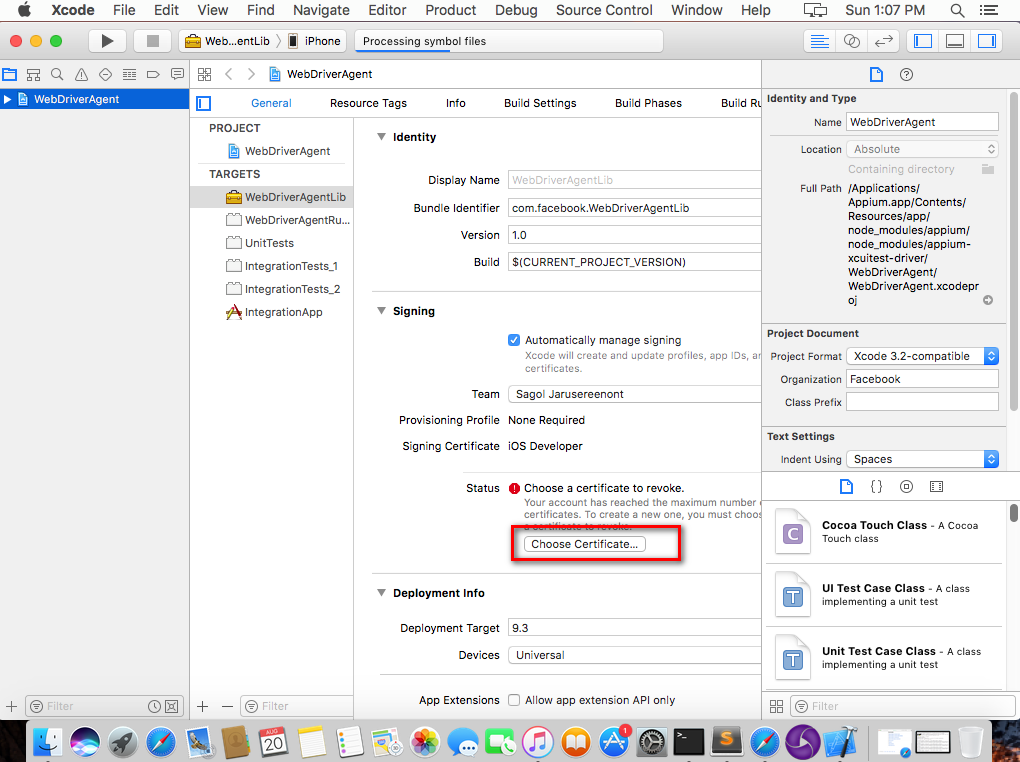
* Select Add Account…

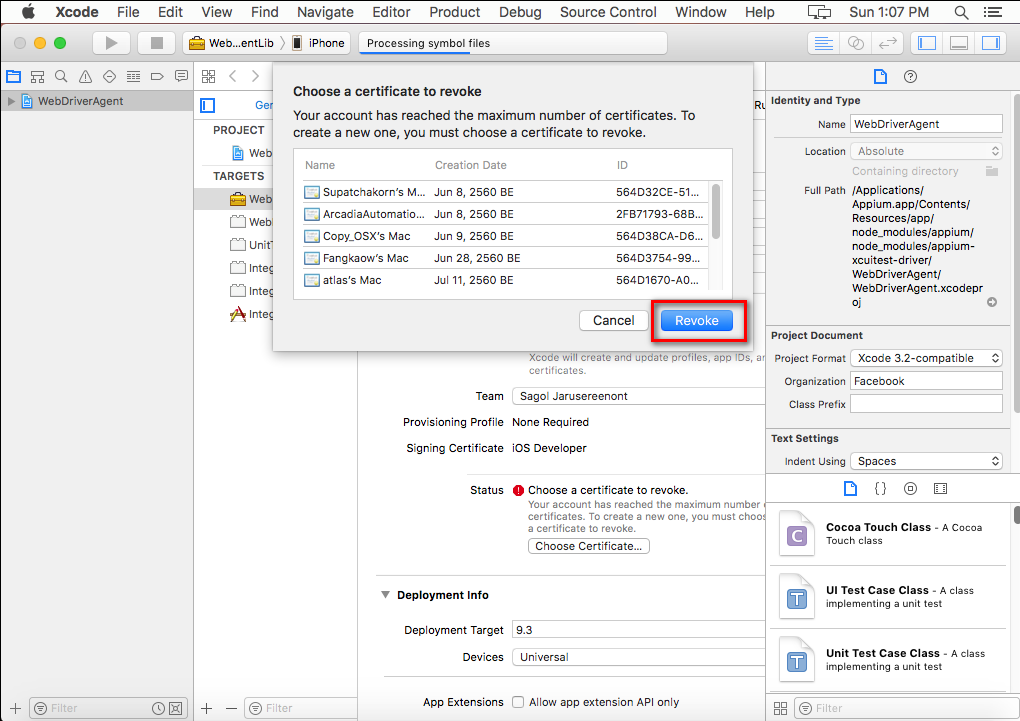


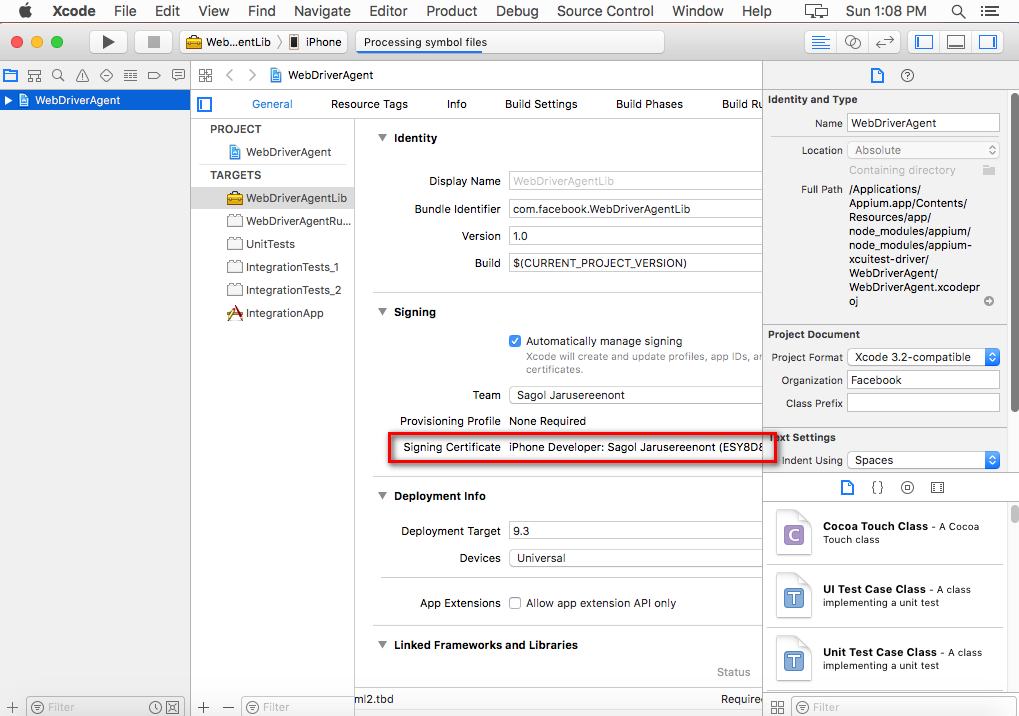




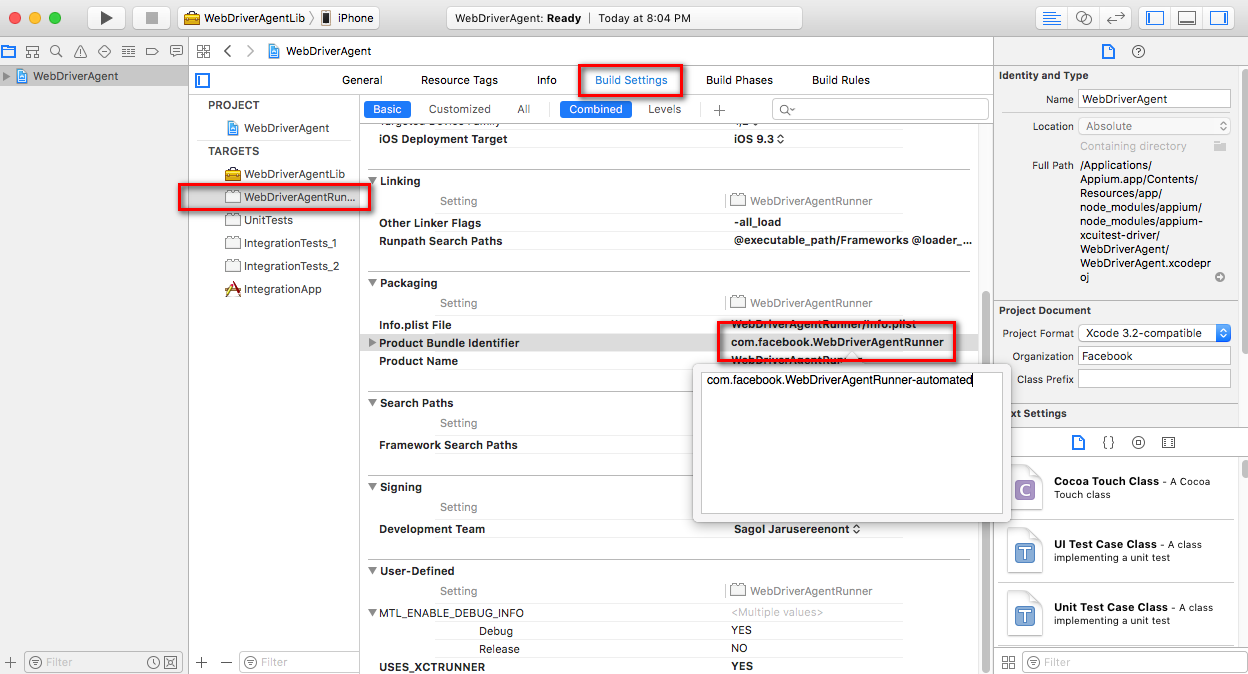
* Select Choose Certificate for Revoke certificate…



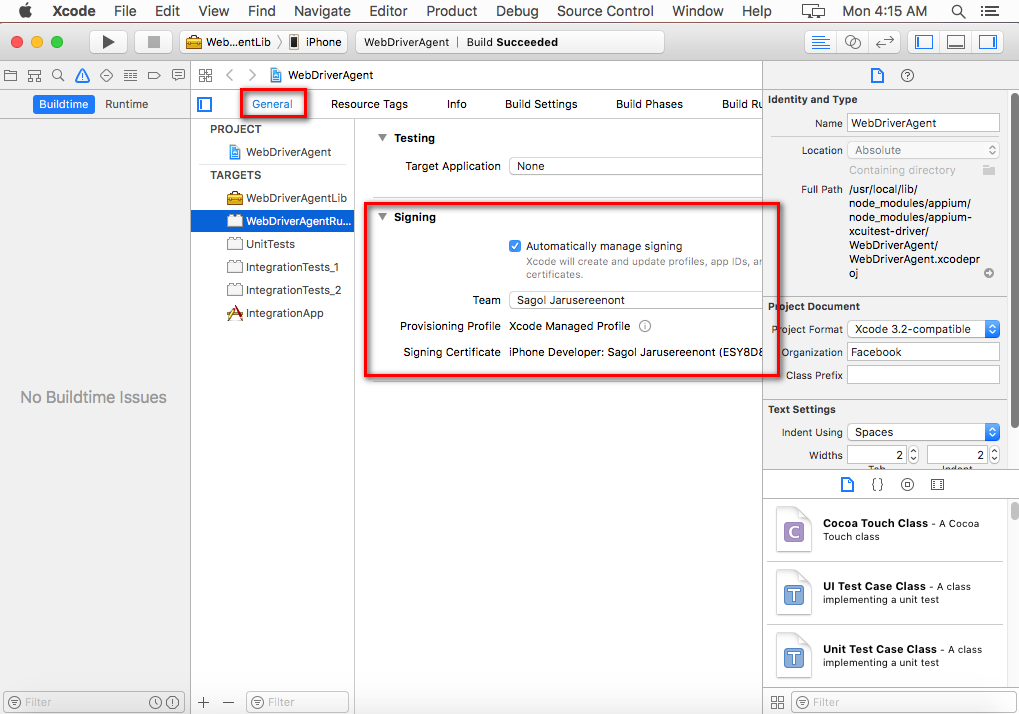




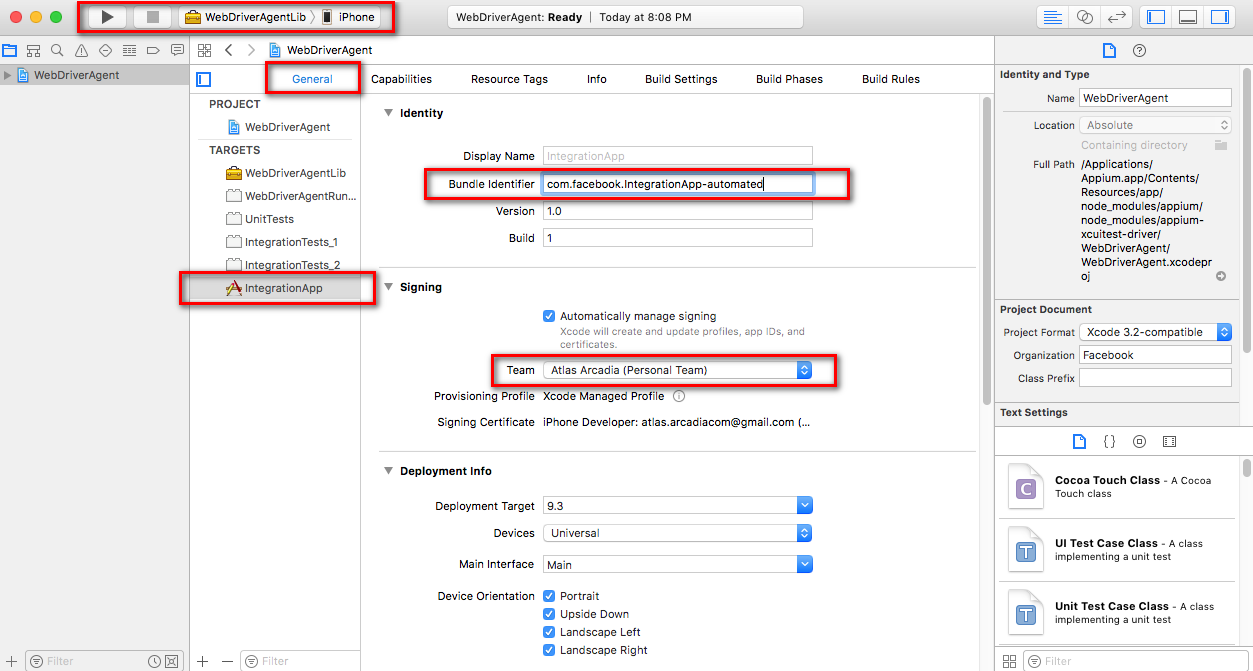
* Select WebDriverAgentRun… > Build Settings
* Edit new Product Bundle Identifier (add –automated behind name)



* Select WebDriverAgentRun… > General
* Check enable Automatically manage signing and set Team already



* Select IntegrationApp > General
* Check Bundle Identifier (add –automated behind name)
* Check enable Automatically manage signing and set Team already



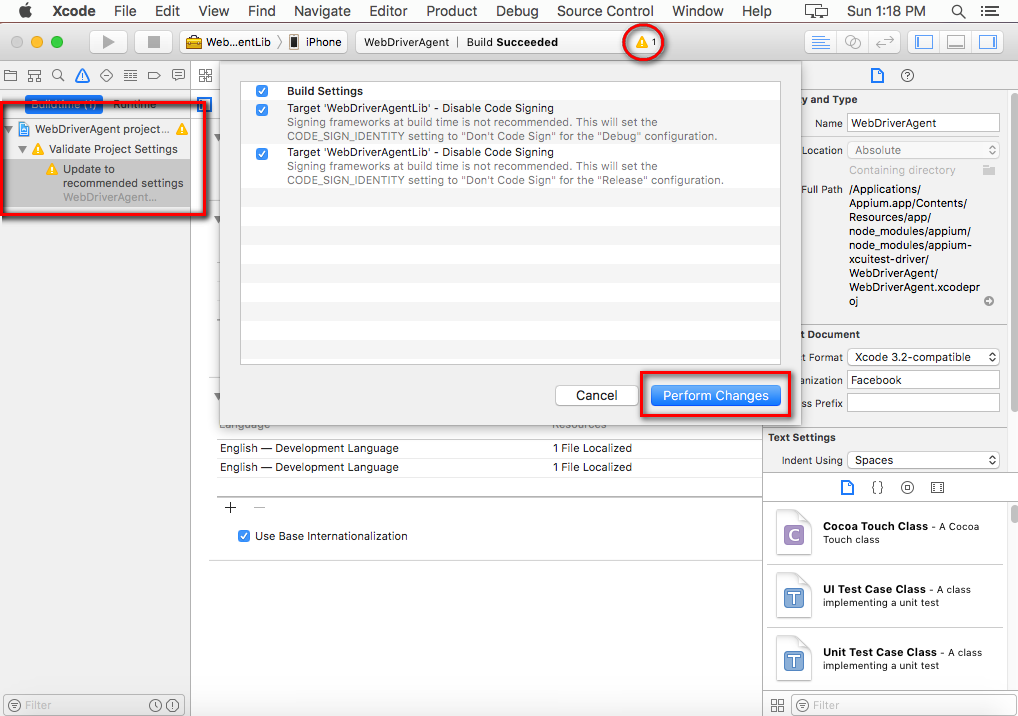
* Select WebDriverAgentLib > iPhone (name device) and test Build

Reference (20/8/2017): <http://testnblog.com/ios-automation-with-appium-1-6-robot-framework/>

**(Optional) Build failed or run on appium error xcode**

* Open File WebDriverAgent..xcodeproj
* Setting again and build

(If it found warning, select Perform Changes and rebuild)

****

**How to set proxy local (command)**

* Set proxy terminal

export http\_proxy=PROXY:PORT

* Set proxy node

npm config set proxy PROXY:PORT

npm config set https-proxy PROXY:PORT