## 说明

1. 本次的Appium搭建环境是在MAC,IOS,Appium,Python
2. 安装Appium的两种方式是，第一种官方已经编译好的安装包的方式（含GUI窗体），第二种使用在线命令的方式安装（无GUI窗体）速度快
3. Appium使用在线命令安装的方式，优点：

* 及时使用最新发布的版本
* 利于持续集成

## 系统要求及说明

1. macOS 10.11 或更高
2. brew需要最新版本的 Xcode 或Xcode Command Line
3. IOS9.3及以下使用 Automation 作为底层技术，在iOS9.3出现了XCUITest，并且在iOS10时抛弃了 Automation ，所以Appium在iOS9.3及以下和iOS10的配置方法不同，本文档搭建的是IOS9.3以上的系统

## 安装xcode

安装xcode,我安装的是9.0

## 安装Homebrew

macOS上的软件使用Homebrew进行安装，使用brew安装过的软件，不再需要sudo命令。打开终端输入：

/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/

## 安装Nodejs

brew install node

检查是否安装成功

node -v  
*# v6.9.1*npm -v  
*# 3.10.8*

## 安装cnpm

由于npm源在国外，我们使用npm安装库时会比较慢，可以使用taobao的镜像源

npm install -g cnpm --registry=https://registry.npm.taobao.org

检查安装是否成功

cnpm -v  
*# 4.4.0*

以后均使用cnpm 命令代替npm

## 安装Python

安装Python3

brew install python3

检测安装成功

python3 -V  
*# Python 3.6.0*pip3 -V  
*# pip 9.0.1 from /usr/local/lib/python3.6/site-packages (python 3.6)*

如果系统版本之前安装过python3导致安装后输入python3没有找到命令，可以尝试brew link --overwrite python3重新链接

## 安装appium server

appium server本质上就是一个Nodejs应用，我们可以使用cnpm对其进行安装，安装完毕后就可以使用命令行启动

cnpm install -g appium

## 安装appium doctor

可以使用appium-doctor --ios来确认安装环境是否完成

bogon:~ shark.shi$ appium-doctor --ios  
info AppiumDoctor Appium Doctor v.1.4.3  
info AppiumDoctor ### Diagnostic starting ###  
info AppiumDoctor ✔ The Node.js binary was found at: /usr/local/bin/node  
info AppiumDoctor ✔ Node version is 7.6.0  
info AppiumDoctor ✔ Xcode is installed at: /Applications/Xcode.app/Contents/Developer  
info AppiumDoctor ✔ Xcode Command Line Tools are installed.  
info AppiumDoctor ✔ DevToolsSecurity is enabled.  
info AppiumDoctor ✔ The Authorization DB is set up properly.  
info AppiumDoctor ✔ Carthage was found at: /usr/local/bin/carthage  
info AppiumDoctor ✔ HOME is set to: /Users/shark.shi  
info AppiumDoctor ### Diagnostic completed, no fix needed. ###  
info AppiumDoctor   
info AppiumDoctor Everything looks good, bye!  
info AppiumDoctor

## 安装Appium Client

pip3 install Appium-Python-Client

## 推荐IDE

[PyCharm](http://www.jetbrains.com/pycharm/)

安装简单，调试方便

## 安装相关依赖

brew install libimobiledevice --HEAD #真机和iOS手机通讯使用  
brew install ideviceinstaller # 安装app使用

在iOS10需要使用 ios-deploy

cnpm install -g ios-deploy

对真机需要xcpretty

gem install xcpretty

gem 安装速度慢可以更换镜像源

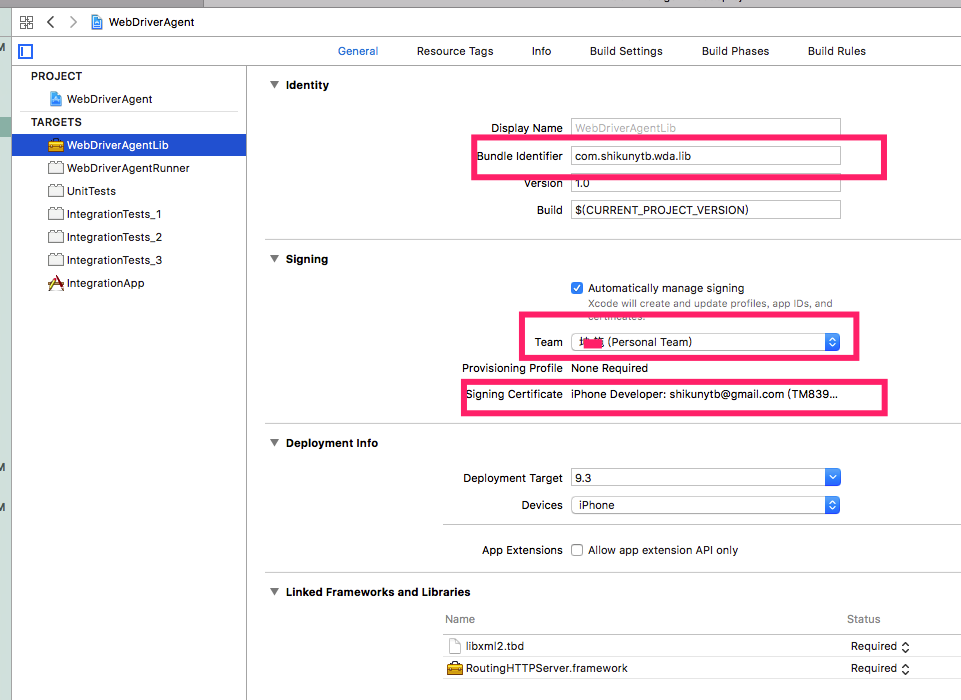
gem sources --add https://gems.ruby-china.org/ --remove https://rubygems.org/

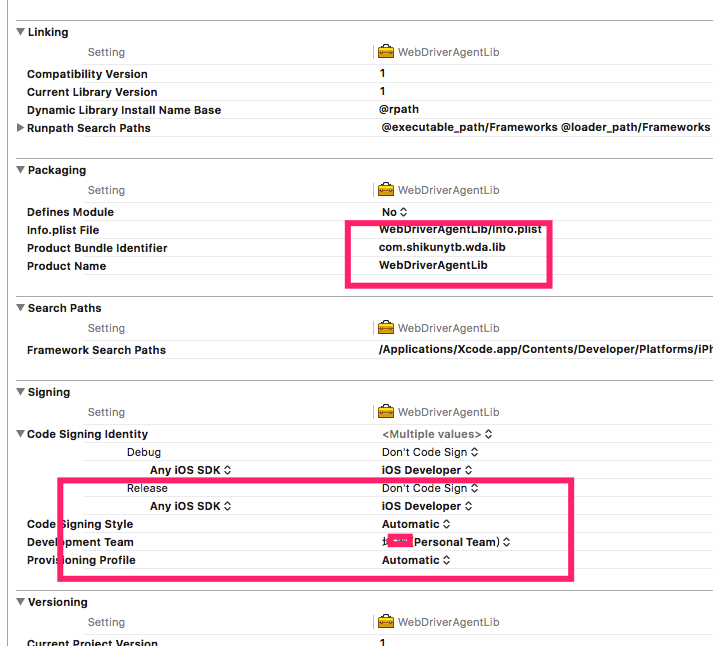
## 安装WebDriverAgent

1. 主要用来查看页面，定位元素，现在使用的是XCUITest，Appium使用的模块是appium-xcuitest-driver，其中引用了Facebook提供的WDA方案来驱动iOS的测试。 装Appium Desktop的时候，它里面带了一个WebDriverAgent，但是这个自带的是有问题的！会造成不能使用Inspector，从Facebook那里自己clone自己编译:

cd ~  
git clone https://github.com/facebook/WebDriverAgent.git  
cd WebDriverAgent  
mkdir - p Resources / WebDriverAgent.bundle  
./Scripts / bootstrap.sh # 开始下载并编译 编译不应该报错  
cd /usr/local/lib/node\_modules/appium/node\_modules/appium-xcuitest-driver/  
rm - rf WebDriverAgent # 把自带的删掉  
ln - s ~ / WebDriverAgent WebDriverAgent # 用facebook的原版替换回去

2. 重签名。进入到安装Homebrew目录，点击WebDriverAgent.xcodeproj，更改 WebDriverAgent 的BundleID ，原来是com.facebook.WebDriverAgentLib，我建议把facebook改为你的名字或公司的名字，改完这个之后发现还是提示有错误，最后在Build Settings里的Product Bundle Identifier里再修改一下。注意，这里可以使用个人账号，我在使用是可以正常编译通过；主要是修改WebDriverAgentLib，WebDriverAgentRunner，IntergrationApp中的tem,bundleID等，编译时那么报错就改哪个





3. 安装到虚拟机，直接点击product-test选择虚拟机即可，可以看到桌面上多来是那个app,其中一个时webdriverAgent,一个是自带的测试app

4.用命令的方式启动,进入到WebDriverAgent目录：

xcodebuild -project WebDriverAgent.xcodeproj -scheme WebDriverAgentRunner -destination id='udid' test

5. 获取udid可以用下面到命令，或者xcode也能查看

ideviceinfo -k UniqueDeviceID

6.执行xcodebuild命令后 而iphone多了一个WebDriverAgentRunner的app。手动启动那个app的时候

[[](https://testerhome.com/uploads/photo/2017/076c53abe8adf2aaceba1386aaaea457.png!large)](https://testerhome.com/uploads/photo/2017/076c53abe8adf2aaceba1386aaaea457.png!large)

  原因是我们手机并并没有信任这个apple id的开发者，到设置-通用-设备管理（描述文件） 信任你的apple id就可以了

## USB support

实测过程中，xcodebuild安装wd 到真机后，在pc上并不能用http://localhost:8100/inspector,访问查看页面，并且还会造成一直连接不上appium server,解决方案，具体查看此[链接](https://github.com/facebook/WebDriverAgent/wiki/USB-support)

$ brew install usbmuxd  
# Start proxy  
$ iproxy 8100 8100

## 实例

### 启动appium server

直接输入appium启动server

bogon:appium shark.shi$ appium  
[Appium] Welcome to Appium v1.7.1  
[Appium] Appium REST http interface listener started on 0.0.0.0:4723

### 客户端代码

运行代码看测试效果

from appium import webdriver  
def run():  
 desired\_caps = {}  
 desired\_caps['automationName'] = 'XCUITest' # Xcode8.2以上无UIAutomation,需使用XCUITest  
 desired\_caps['platformName'] = 'iOS'  
 desired\_caps['platformVersion'] = '9.3.3'  
 desired\_caps['deviceName'] = 'iPhone 6 Plus'  
 desired\_caps['bundleId'] = 'com.huawei.works'  
 desired\_caps['udid'] = 'fb5991a499fb96bde153923f4b9d61c68fbeee42'  
 desired\_caps['newCommandTimeout'] = 3600 # 1 hour  
 remote = "http://127.0.0.1:" + "4723" + "/wd/hub"  
 driver = webdriver.Remote(remote, desired\_caps)

driver.find\_element\_by\_accessibility\_id("知识").click()

### 获取页面元素

***注意，PC和手机网络要在同一个网络，最好是PC分享一个热点给手机使用，这样就能调试真机***

