

## 一、APP环境搭建

- 1、安装JDK，配置jdk环境变量
- 2、安装配置android sdk
- 3、Appium server安装



- 4、安装模拟器（夜神模拟器）



- 5、安装Python的第三方库 appium-python-client

## 二、Appium及Appium自动化测试原理

Appium跨平台、开源的app自动化测试框架，用来测试app应用程序，支持Android/iOS操作系统

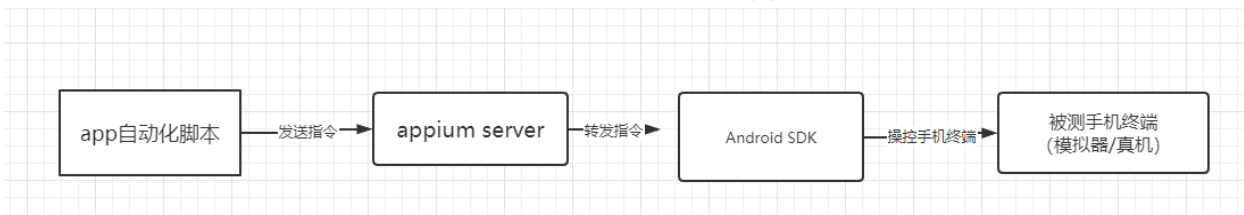
为什么通过代码操作不同的操作系统(Android/iOS)不同版本的手机终端的应用程序？

web自动测试：如何通过代码，实现操作pc端的浏览器进行对应的操作？

代码selenium中webdriver--xxx浏览器驱动----操控浏览器实现对应操作

app自动化测试：

代码-----？      操控手机端的应用程序，实现对应app自动化测试



## 三、Appium自动化配置项

不同系统不同版本可能用到不同的软件包/框架----》python脚本指定操作终端的这些参数设置

### [Desired Capabilities - Appium](#)

常用必备参数：

platformName 操作系统 (IOS/android) Android

platformVersion 版本 5.1.1

deviceName 设备名称 ? 127.0.0.1:62001

appPackage 包名 'com.tal.kaoyan'

appActivity 入口启动页面 'com.tal.kaoyan.ui.activity.SplashActivity'

## 四、常见ADB命令

### ADB命令:命令调试桥

1、连接模拟器 命令: 查看连接设备: adb devices 或者 连接设备: adb connect 127.0.0.1:62001

不同模拟器端口号不同: 雷神: 5555 夜神: 62001 mumu:7555 逍遥: 21503

### 2、获取包名及入口启动页面

aapt dump badging apk包路径 C:\Users\xingyao\Desktop\kaoyan3.1.0.apk

进入aapt.exe所在的路径 (D:\Andriod\_SDK\Andriod\_SDK\build-tools\29.0.3), 再输入命令

命令: aapt dump badging C:\Users\xingyao\Desktop\kaoyan3.1.0.apk

app 应用程序如何装到模拟器上

获取包名:

```
Microsoft Windows [版本 10.0.19042.1110]
(c) Microsoft Corporation. 保留所有权利。

D:\Andriod_SDK\Andriod_SDK\build-tools\29.0.3>aapt dump badging C:\Users\xingyao\Desktop\kaoyan3.1.0.apk
package: name='com.tal.kaoyan' versionCode='55' versionName='3.1.0' platformBuildVersionName='N'
targetSdkVersion:'21'
uses-permission: name='getui.permission.GetuiService.com.tal.kaoyan'
uses-permission: name='android.permission.SYSTEM_ALERT_WINDOW'
uses-permission: name='android.permission.INTERNET'
uses-permission: name='android.permission.WRITE_EXTERNAL_STORAGE'
uses-permission: name='android.permission.ACCESS_NETWORK_STATE'
uses-permission: name='android.permission.ACCESS_WIFI_STATE'
uses-permission: name='android.permission.BLUETOOTH'
uses-permission: name='android.permission.READ_EXTERNAL_STORAGE'
uses-permission: name='android.permission.MOUNT_UNMOUNT_FILESYSTEMS'
uses-permission: name='android.permission.GET_TASKS'
uses-permission: name='android.permission.CHANGE_WIFI_STATE'
uses-permission: name='android.permission.READ_PHONE_STATE'
uses-permission: name='android.permission.MANAGE_ACCOUNTS'
uses-permission: name='android.permission.GET_ACCOUNTS'
uses-permission: name='android.permission.CALL_PHONE'
uses-permission: name='android.permission.INTERACT_ACROSS_USERS_FULL'
uses-permission: name='android.permission.WAKE_LOCK'
uses-permission: name='android.permission.RECEIVE_BOOT_COMPLETED'
uses-permission: name='android.permission.VIBRATE'
uses-permission: name='android.permission.READ_LOGS'
```

包名

获取启动页面appActivity 的值

```
application-label-tr: 端端端端?
application-label-uk: 端端端端?
application-label-ur-PK: 端端端端?
application-label-uz-UZ: 端端端端?
application-label-vi: 端端端端?
application-label-zh: 端端端端?
application-label-zh-CN: 端端端端?
application-label-zh-HK: 端端端端?
application-label-zh-TW: 端端端端?
application-label-zu: 端端端端?
application-icon-120: res/drawable-ldpi-v4/ic_launcher.png'
application-icon-160: res/drawable-mdpi-v4/ic_launcher.png'
application-icon-240: res/drawable-hdpi-v4/ic_launcher.png'
application-icon-320: res/drawable-xhdpi-v4/ic_launcher.png'
application-icon-480: res/drawable-xxhdpi-v4/ic_launcher.png'
application-icon-640: res/drawable-xxhdpi-v4/ic_launcher.png'
application: label='端端端端? icon= res/drawable-mdpi-v4/ic_launcher.png'
launchable-activity: name='com.tal.kaoyan.ui.activity.SplashActivity' label=' icon='
feature-group: label
uses-feature: name='android.hardware.bluetooth'
uses-implied-feature: name='android.hardware.bluetooth' reason='requested android.permission.BLUETOOTH permission, requested android.permission.BLUETOOTH_ADMIN permission, and targetSdkVersion > 4'
uses-feature: name='android.hardware.camera'
uses-implied-feature: name='android.hardware.camera' reason='requested android.permission.CAMERA permission'
uses-feature: name='android.hardware.faketouch'
uses-implied-feature: name='android.hardware.faketouch' reason='default feature for all apps'
uses-feature: name='android.hardware.location'
uses-implied-feature: name='android.hardware.location' reason='requested android.permission.ACCESS_COARSE_LOCATION permission, requested android.permission.ACCESS_FINE_LOCATION permission, and requested android.permission.ACCESS_LOCATION_EXTRA_COMMANDS permission'
```

activity的值

## 五、第一app自动化脚本

启动被测app应用程序

python脚本发送http请求给到appium server , 实现操控app应用程序  
进行其他的操作,继续下一个http请求

## 六、查看详细日志文件

- 1、自动化脚本发送 http请求, 请求参数: 终端设置参数
  - 2、创建会话
  - 3、确认终端设备是否连接, 并且确认安卓的版本 确认设置其他参数与终端是否一致
  - 4、 appium会推送一个包'AppiumBootstrap.jar'
- AppiumBootstrap.jar:实现appium server与手机终端进行数据交互
- 5、下一个http请求