|  |
| --- |
| 上海大智慧股份有限公司 |
| 大智慧移动  自动化使用说明书 |
| 2016-12-30 |
|  |
|  |

**修订记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本 | 章节 | 修订内容 | 修订日期 | 作者 | 批准人 |
| 1.0 | 1 | 首稿 | 2016.12.20 | 鲁砂 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 1测试环境搭建

## 1.1 工具获取

* Java开发工具jdk  
  <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
* Java集成开发环境intellij idea  
  <https://www.jetbrains.com/idea/>
* Android sdk   
  <https://android-sdk.en.softonic.com/>
* Appium 服务端

<http://appium.io/>

## 1.2 环境配置

**依据自己本机环境,下面为参考**

* JAVA\_HOME=C:\Program Files\Java\jdk1.7.0\_79
* ANDROID\_HOME =D:\Program\sdk

# 2工程结构描述

* Android: 安卓平台测试用例,公共函数库,测试资源
* Ios: 苹果平台测试用例,公共函数库,测试资源
* Driver: 驱动测试用例,appium,
* Entry: 测试报告,测试日志,配置文件

# 3工程管理

大智慧移动自动化平台采用maven进行项目管理

具体使用可参考: <http://maven.apache.org/>

# 4用例模版

**#if** (**${**PACKAGE\_NAME**}** && **${**PACKAGE\_NAME**}** != "")package **${**PACKAGE\_NAME**}**;**#end**import gw.com.cn.testcase.DZHBaseTestCase;  
import org.testng.annotations.AfterMethod;  
import org.testng.annotations.BeforeMethod;  
import org.testng.annotations.Test;  
  
**#parse**("File Header.java")  
public class **${**NAME**}** extends DZHBaseTestCase {  
  
 @BeforeMethod  
 public void setUp() {  
 super.setUp();  
 }  
  
 @Test  
 public void testStep() {  
 super.testStep();  
 }  
  
 @AfterMethod  
 public void tearDown() {  
 super.tearDown();  
 }  
}

**建议一个测试用例只包含一个测试场景**

* Setup 测试预置条件
* Teststep 具体测试步骤
* TearDown 测试清理操作

# 5测试用例配置

将编写好的测试用例配置到classes节点下

具体配置可参考: <http://testng.org/doc/documentation-main.html>

<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >  
<suite name="dzhSuite">  
 <listeners>  
 <listener class-name="gw.com.cn.CustomListener"/>  
 <listener class-name="org.uncommons.reportng.DzhHTMLReporter"/>  
 <listener class-name="org.uncommons.reportng.JUnitXMLReporter"/>  
 </listeners>  
 <test name="dzhSettingsTest">  
 <classes>  
 <class name="gw.com.cn.testcase.settings.DZH\_TestCase\_Settings\_001"/>  
 </classes>  
 </test>  
 <test name="dzhNewsTest">  
 <classes>  
 <class name="gw.com.cn.testcase.news.DZH\_TestCase\_News\_001"/>  
 </classes>  
 </test>  
 <test name="dzhSelfSelection">  
 <classes>  
 <class name="gw.com.cn.testcase.selfSelection.DZH\_TestCase\_SelfSelection\_001"/>  
 </classes>  
 </test>  
</suite>

# 6平台配置

{  
 "address": "http://127.0.0.1",  
 "ftpPath": "10.15.54.198",  
 "port": "4723",  
 "sdkPath": "D:\\Program\\sdk",  
 "appPath": "C:\\dzh.apk",  
 "downloadAppFromServer": **false**,  
 "replaceExistingApp" : **false**,  
 "devices": [  
 {  
 "deviceBrand": "google",  
 "brandSeries": "nexus6p",  
 "deviceType": "master",  
 "deviceName": "ENU7N16709000499",  
 "platformName": "Android",  
 "platformVersion": "25",  
 "automationName": "Appium",  
 "sessionTimeout": "120",  
 "language": "zh\_CN",  
 "locale": "CN"  
 },  
 {  
 "deviceBrand": "meizu",  
 "brandSeries": "note1",  
 "deviceType": "slave",  
 "deviceName": "304d1165ec0baf8f",  
 "platformName": "Android",  
 "platformVersion": "23",  
 "automationName": "Appium",  
 "sessionTimeout": "120",  
 "language": "zh\_CN",  
 "locale": "CN"  
 }  
 ]  
}

* address: appium服务端地址
* ftpPath: 大智慧移动应用ftp地址
* sdkPath: android sdk路径
* appPath: 本地应用路径
* downloadAppFromServer: 是否从服务下载最新应用
* replaceExistingApp: 是否替换手机上的应用
* deviceBrand 手机品牌
* brandSeries 品牌系类
* deviceType: 主设备或者辅设备
* deviceName: 设备名称
* platformName: 平台名称(ios,android)
* platformVersion: 平台版本(android20)
* automationName: 自动化工具的名称(Appium or Selendroid)

# 7日志配置

可以通过打开entry模块下的logs\ dzhLog.log查看执行日志

Notes: 调试环境下可将debug打开,可以在日志中查看post信息

### set log levels ###  
log4j.rootLogger =debug,std,logFile  
  
### 输出到控制台 ###  
log4j.appender.std = org.apache.log4j.ConsoleAppender  
log4j.appender.std.Target = System.out  
log4j.appender.std.layout = org.apache.log4j.PatternLayout  
log4j.appender.std.layout.ConversionPattern = %d{yyyy-MM-dd HH:mm:ss,SSS} %5p (%F:%L) - %m%n  
#log4j.appender.std.layout.ConversionPattern = %d{yyyy-MM-dd HH:mm:ss,SSS} %5p [%t] (%F:%L) - %m%n  
  
### 日志文件输出 ###  
log4j.appender.logFile=org.apache.log4j.FileAppender  
log4j.appender.logFile.Threshold=DEBUG  
log4j.appender.logFile.ImmediateFlush=true  
log4j.appender.logFile.Append=true  
log4j.appender.logFile.File=./logs/dzhLog.log  
log4j.appender.logFile.layout=org.apache.log4j.PatternLayout  
log4j.appender.logFile.layout.ConversionPattern = %d{yyyy-MM-dd HH:mm:ss,SSS} %5p (%F:%L) - %m%n

# 8运行测试

配置entry模块下gw.com.cn.DzhStart文件

Notes：

* json文件下配置的设备类型和设备名称必须与连接到电脑的一致
* json文件下配置的设备系统名称和系统版本必须与连接到电脑的一致

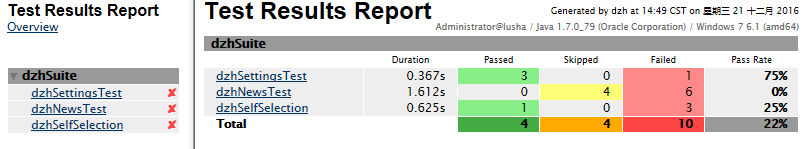
public class DzhStart {  
  
 @Test  
 public void test(){  
 DZHRunner.*runDZHCases*("\\src\\main\\resources\\conf\\master.json", "\\src\\main\\resources\\caseConf\\allCases.xml");  
 }

｝

# 9 结果查看

通过打开entry模块下的dzhReport\html\index.html文件查看报告

Overviw:



Detail:

