自动化测试需求规格说明书

# 引言

开发自动化测试脚本，目的是为了减轻手工测试的工作强度，加快测试进度。从而使得测试人员可以有更多的精力去做其他更加重要的工作。

设计初期实现单线程跑完多个测试用例。

# 运行环境及配置

采用python编写的自动化脚本支持跨平台操作。

需要安装：Python2.7（不支持3.0）、NodeJS、Selenium、Appium、maven

# 开发环境

pycharm 、aptana、notepad++、sublime。

# 框架概要

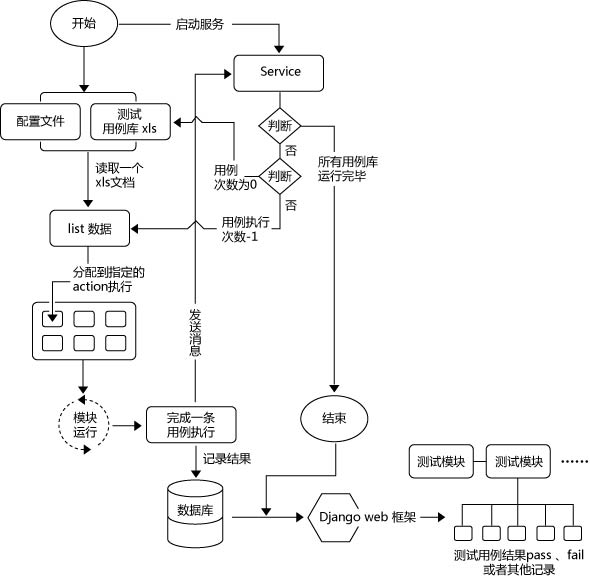
自动化测试脚本在运行过程中，通过监控线程，对整个测试用例的运行状态进行监控。在获知某条用例运行完毕后，重新读取数据分配任务。

每个测试用例的action，都继承自一个TaskBase的基类，便于监控线程的全局控制。

测试用例配置采用excel，降低测试人员的执行难度，它由脚本初始化时读取到list中，再通过以上的action来运行测试任务。

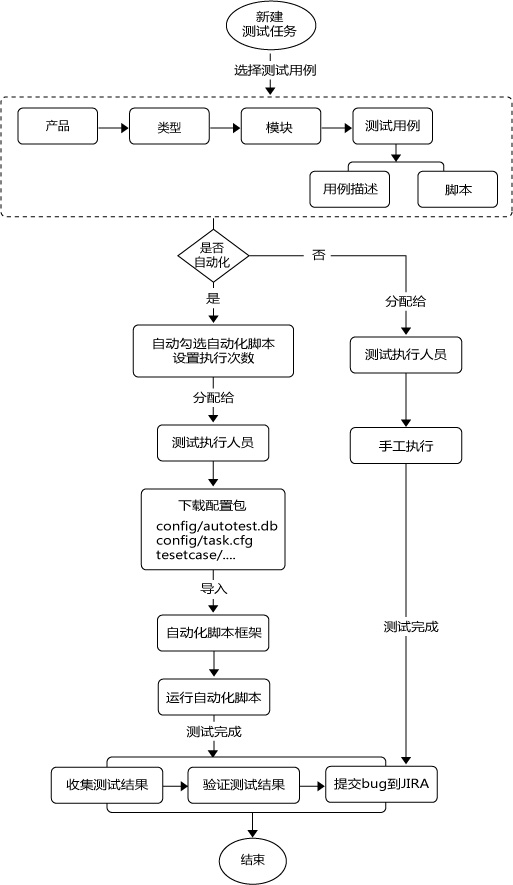
数据库采用sqlite，测试完成后，即放入到测试结果web服务中，并对原有的数据库按日期备份，这样可以防止数据库不断的变的庞大。

# 自动化脚本框架流程图



# 测试任务执行流程

测试任务可以由不同的角色发起



# 框架功能细分

## action-用例执行包

该包内包含了执行测试用例所有需的类，该类通过继承TaskBase来实现。

通过读取testsuite 文件夹内的测试人员配置的测试用例（exlce文档），来执行具体的测试任务。

## Core-功能核心包

包括service、任务管理、任务基类

### service

负责监视当前测试用例的运行状态，接收测试用例finish消息，并分配任务的执行。

其中分配任务有随机分配和按顺序分配。

def receiveMsg(caseId,caseStatus):

### 任务管理类

包含测试用例读取、存储、查找、分配等。

|  |  |  |
| --- | --- | --- |
| 方法名 | 数值类型 | 描述 |
| readXlsToDb | Void | 从xls文件从读取数据 |
| findCase | TestCase | 根据id从数据库查找一条Case |
| dispatchCase | void | 分配case执行 |
|  |  |  |

### 任务基类

所有运行case的 Task都需要通过继承这个基类来实现。

|  |  |  |
| --- | --- | --- |
| 属性名 | 数值类型 | 描述 |
| isRunning | Boolean | Case的运行状态 |
| caseName | String | Case的名称 |
| caseId | Int | 在sqlite中存储的id值 |
| caseLoop | Int | 执行次数，每执行完一次递减 |
|  |  |  |
|  |  |  |

方法：

|  |  |  |
| --- | --- | --- |
| 方法名 | 返回类型 | 描述 |
| start() | Void | 开始Case运行 |
| stop() | Void | 停止Case运行，某些情况下需要强行终止 |
| finish() | Void | 本次Case运行结束，等待下一次分配。 |
| sendMsg(caseId,status) | Void | 发出自身的运行状态消息，由finish() 调用后触发，该消息会被service.py 监测到。 |
| ........ |  |  |

## 公用方法

通用的操作类集合包，方便开发人员调用

|  |  |
| --- | --- |
| 功能名称 |  |
| fileUtils.py |  |
| httpUtils.py |  |
|  |  |

# 第三方测试工具包的选择

第三方测试工具包可以使得这套自动化脚本的开发难度大大降低，我们没有精力也无须去重复造轮子。经过这几天的筛选，重点考察了以下几套：

## Selenium

对web ui的自动化测试选择之一, 需要注意它不能判断网页加载完成，需要自己去判断网页超时等。Selenium测试直接运行在浏览器中，就像真正的用户在操作一样。测试与浏览器的兼容性——测试你的应用程序看是否能够很好得工作在不同浏览器和操作系统之上。

支持自动录制动作和自动生成 .Net、Java、Python等不同语言的测试脚本。

结论：可以录制脚本，可以大大减少手工编写测试用例代码。

## Sikuli

可以实现任意类型的UI（GUI）测试，轻松实现跨平台测试，并可以解决web测试中的浏览器兼容性测试问题。

输出的报告无法获取。

## Monkeyrunner

对android apk在真机或模拟器下，通过monkeyrunner + python 进行自动化测试。

不能输入文本

## Robotium

只有对android的测试。

## Appium

支持iOS和Android创建功能自动化测试用例。相比其他的自动化测试工具，Appium测试由于调用了Selenium的client库使其可以使用任意的语言，包括Python、Ruby、Node.js,Objective，java等。

结论：比较符合一套自动化框架，通用语不同的平台的产品测试。

注： 初步决定使用selenium和Appium 来辅助实现自动化测试脚本的实现，其中Appium由于额外支持IOS，所以优先选择。

# 测试结果呈现

采用Django web框架，把生成的测试结果sqlite db，放入到数据库目录，以往的数据库按日期名备份。

前期只有对输出的错误结果快速定位。

将来可以发展到对测试结果的统计分析，图表对比等。