**IAT框架设计**

# 1背景

## 1.1 项目背景

在移动平台服务端接口测试覆盖度为零的情况下，根据服务端接口的特点，以及升级更新的速度较快等，需要开发此框架来实施服务端接口的自动化测试。

## 接口测试

接口测试属于灰盒测试范畴，通常不需要了解接口底层的实现逻辑，但需要测试人员能够使用代码的方式来调用接口。接口测试主要用例测试接口的功能以及接口返回数据的正确性。根据接口测试的复杂度接口测试分为两种。即单一接口测试，以及多接口组合功能测试。由于接口测试是通过代码调用的方式完成，而且接口测试与前端UI属于松耦合（或无耦合）因此通过自动化手段将极大提高测试效率以及回归测试的复用率。本文中提到的接口测试主要是指基于http,https，rpc协议的web接口。

## 适用性分析

移动平台大部分以http接口方式提供服务，通过前台App调用接口方式实现功能。同时大部分接口功能，以及表现形式稳定，对于前台变化敏感度较低。基于上述接口测试的特点，认为移动平台项目非常适合接口层级的自动化测试。

# 2 IAT框架

## 2.1 IAT介绍

IAT是Interface Automation Testing的简称。通过热插拔的方式支持http,rpc,soap类协议的web接口测试。框架支持单一接口，多接口组合测试，支持用户通过自定义方法实现精确验证结果的需求。

## 2.2 框架特点

* 提供多种接口测试方式。即单一接口测试，多接口业务流程测试。目前多见的为单一接口的测试。
* 根据用户需求不同，不同的接口测试方式，用例开发难易度不同。
* 用例开发门槛低，用户只需要将接口用例数据填入格式化文件即可自动通过工具生成用例。
* 对于高级需求，框架提供自定义配置包括数据构造，精确匹配测试结果等。
* 框架对于不同域名下的相同接口支持自定义配置，只需要简单修改测试平台配置即可轻松将用例应用在不同平台上。
* 框架对于不同协议接口的支持，近乎无缝连接。
* 框架支持可配置

## 2.3框架整体组成

测试准备

测试数据

单元测试代码

用例描述文件

测试执行器

框架核心

Httpunit

* 单元测试代码：由代码生成工具根据用例描述文件自动生成，一个Testmehod为一个用例。
* 测试数据：格式化存储测试用例需要的数据
* 用例描述文件：用于定义用例需要调用那些接口，那些方法以及数据预制等信息
* 用例执行器：根据单元测试传递的参数，查找，解析用例文件并执行。
* 框架核心：包含操作用例描述文件，测试数据文件，以及数据构造，接口调用，结果验证等功能

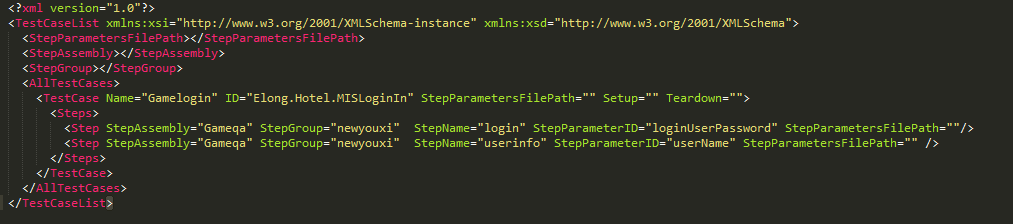
## 2.3 用例执行流程图



## 2.4单一接口用例数据结构

|  |  |  |
| --- | --- | --- |
| **字段名称** | **字段说明** | **可选值** |
| ID | 唯一即可 |  |
| DomainName | 域名可配置 |  |
| Path | 域名后参数前的部分 |  |
| ParameterName | 增加一个参数请增加一列 |  |
| ExpectResult |  | |  | | --- | |  | |  | |  | |
| AssertType |  | |  | | --- | | Equal | | Contains | | Custom | |
| AssertMethod |  | 自定义断言方式需要提供断言方法 |
| SetupType |  | |  | | --- | | SQL | | DBID | | Custom，  SelfInvoke，  依赖关系 | |
| SetupContext |  | |  | | --- | | SQL语句 | | 数据构造系统ID | | 数据预制方法(需要提供数据参数ID),  自调用方式需要提供数据的同时需要提供一个方法名称，  需要提供依赖用例ID | |
| TearDownType |  | |  | | --- | | SQL | | DBID | | Custom  SelfInvoke | |
| TearDownContext |  | |  | | --- | | SQL语句 | | 数据构造系统ID | | 数据预制方法, | |

**2.5 组合接口测试用例结构**



**2.6 组合接口参数存储结构**

