<Memory>

测试用例

android

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2016/9/8 | 1.0 | 对api的测试 | 李轶 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. 简介 5

1.1 目的 5

1.2 范围 5

1.3 定义、首字母缩写词和缩略语 5

1.4 参考资料 5

1.5 概述 5

2. 单元测试 5

2.1 Add book 5

2.2 Search music 6

2.3 ScanCapture 6

2.4 Main refresh 6

2.5 Choose photo 6

2.6 Crop photo 6

2.7 DataBase 6

3. URI-API集成测试 UserBasic 7

3.1 /login POST 7

3.2 /logout POST 7

3.3 /identity/reg POST 7

3.4 /identity/detail/{username} GET 7

3.5 /identity/profile/{username} GET 7

3.6 /identity/profile POST 7

4. URI-API集成测试 Resource 7

4.1 /resources GET 7

4.2 /resources/latest GET 7

4.3 /resource/friends GET 7

4.4 /resources/latest/{timestamp} GET 7

4.5 /resources/friends/{timestamp} GET 7

4.6 /resources/self/{timestamp} GET 7

4.7 /resources POST 7

4.8 others 7

4.9 /resources/{resource\_id}/words GET/POST 7

4.10 /resources/{resource\_id}/image GET/POST 7

4.11 others 7

5. URI-API 集成测试 Friend 8

5.1 /friends GET 8

5.2 /friends/{username} DELETE 8

5.3 /friends/apply GET 8

5.4 /friends/apply/{username} POST 8

5.5 /friends/make GET 8

5.6 /friends/make/{username} POST/DELETE 8

6. URI-API 集成测试 Mark Music And Book 8

6.1 /resources/{resource\_id}/marks GET 8

6.2 /resources/{resource\_id}/marks/{timestamp} GET 8

6.3 /resources/resource\_id}/marks POST/GET/DELETE 8

6.4 /resources/{resource\_id}/music/{hashcode} POST 8

6.5 /resources/{resource\_id}/book/{ISBN} POST 8

7. URI-API 集成测试 Others 8

7.1 /statistics/visitors, ./resources/new, ./friends/new, ./resources GET 8

7.2 /recommend/user GET 8

测试用例

# 简介

本测试用例文档，主要囊括了安卓端(android end)的测试用例代码及其描述。通过对每一条URI-API的测试，来确定其可用性。

## 目的

本文档旨在保证安卓端的可用性，并在测试结果的基础上给出建议。

## 范围

本文档主要包括安卓端的代码测试，包括单元测试以及集成测试。

## 定义、首字母缩写词和缩略语

|  |  |
| --- | --- |
| **文档用词** | **含义** |
| uri | 指代某一个服务API所对应的uri |

## 参考资料

1. API文档： PART 1-UserBasic.json

PART 2-Resource.json

PART 3-Friend.json

PART 4-Mark.json

PART 5-MusicAndBook.json

PART 6-Tool.json

## 概述

本文档通过对每一个测试用例的描述或者输入代码，配合期待的输出值或者相应表现，来形成一个测试用例。测试人员通过对比来生成测试报告。测试用例的部分以产生所有输出信息为原则，完成边界值以及等价类覆盖。

# 单元测试

## Add book

独立功能测试

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 1. isbn |
| 观察点 | 1. search()函数调用 |
| 预期结果 | 获取到json书籍信息 |

## Search music

独立功能测试

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 1. music name |
| 观察点 | 1. search()函数调用 |
| 预期结果 | 获取到json音乐信息 |

## ScanCapture

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 扫描条形码 |
| 观察点 | Scan返回内容 |
| 预期结果 | 获得条形码内容 |

## Main refresh

上拉刷新

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 指定加载内容 |
| 预期结果 | 显示出指定内容，其它卡片内容不变 |

下拉刷新

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 指定加载内容 |
| 预期结果 | 显示出指定内容，其它卡片内容不变 |

翻页刷新

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 指定加载内容 |
| 预期结果 | 显示出指定内容，其它卡片内容不变 |

## Choose photo

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 手机选择照片 |
| 观察点 | Println() |
| 预期结果 | 返回照片路径 |

## Crop photo

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 获取到原图片文件 |
| 测试输入 | 原图片bitmap  指定比例/任意比例裁剪框 |
| 观察点 | Crop save() |
| 预期结果 | 显示并保存裁剪图片 |

## DataBase

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 打开主页面刷新 |
| 观察点 | 主页面显示 |
| 预期结果 | 获得上次浏览内容 |

# URI-API集成测试 UserBasic

本部分的内容，以及之后的URI-API集成测试部分，均以一个URI+http方法为一个模块进行测试用例的书写

## /login POST

登录信息正确或者错误

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录或者已经登录 |
| 测试输入 | {  **"name"**:**"Sturmfy"**,  **"password"**:**"1234"** } |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"Login Success with old user name1"** }  or  {  **"fail"**:**"User Not Exist Or Password Not Match"** } |
| 后置条件 | 成功 |

## /logout POST

已经登录情况下，登出

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"Log out success"** } |

未登录情况下，登出

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"No User Logged"** } |

## /identity/reg POST

注册（无论登录与否）

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录或者已经登录 |
| 测试输入 | {  **"name"**:**"Sturmfy"**,  **"password"**:**"1234"**,  **"role"**: 1,  **"email"**:**"123@email.com"**,  **"age"**:**"20"**,  **"content"**:**"A cute boy ready for love"** } |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"user exists"** }  or {  **"success"**:**"Register Success!"** }  如果登录的情况下注册成功，自动切换为该用户 |

注册非法输入测试

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录或者已经登录 |
| 测试输入 | {  **"name"**:**"Sturmfy"**,  **"password"**:**"1234"**,  **"role"**: 1,  **"email"**:**"123 "**,  **"age"**:**"20"**,  **"content"**:**"A cute boy ready for love"** }  {  **"name"**:**""**,  **"password"**:**"1234"**,  **"role"**: 1,  **"email"**:**"123 "**,  **"age"**:**"20"**,  **"content"**:**"A cute boy ready for love"** } |
| 观察点 | 手机端提示 |
| 预期结果 | 输入为空，请重新输入  邮箱不存在，请重新输入 |

## /identity/detail/{username} GET

登录情况下查询自己的信息

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 自己的username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"name"**:**"Sturmfy"**,  **"role"**: 1,  **"email"**:**"123@email.com"**,  **"age"**:**"20"**,  **"content"**:**"A cute boy ready for love"** }  不包括密码的所有详细信息 |

登录情况查询他人信息，或者未登录查询他人信息

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或没有登录 |
| 测试输入 | 他人的username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"name"**:**"Sturmfy"**,  **"age"**:**"20"**,  **"content"**:**"A cute boy ready for love"** }  简单信息 |

## /identity/profile/{username} GET

获取用户的头像

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或没有登录 |
| 测试输入 | 某个人的username |
| 观察点 | 控制台输出 |
| 预期结果 | 图片文件流，可以直接用html代码或者其他文件流管道对接。显示的图片没有损坏错误 |

用户不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或没有登录 |
| 测试输入 | 某个不存在的人的username |
| 观察点 | 控制台输出 |
| 预期结果 | 默认的图片文件流，可以直接用html代码或者其他文件流管道对接。显示的图片没有损坏错误 |

## /identity/profile POST

上传

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 图片文件流 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"image upload succeeded"** } |

上传，未登录

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录 |
| 测试输入 | 图片文件流 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"Something's wrong"** } |

上传，文件大小超过限制

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 图片文件流 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"Something's wrong"** } |

# URI-API集成测试 Resource

## /resources GET

登录，获取

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"Sturmfy"**,  **"status"**:**"1"** }  {  **"id"**:72,  **"name"**:**"Sturmfy"**,  **"memory\_id"**:1,  **"image\_id"**:2,  **"status"**:**'1'** } ] |

未登录，获取

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"User Not Logged"** } |

## /resources/latest GET

获取最新可以可浏览的资源

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 无 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402  } ] |

## /resource/friends GET

获取到朋友的最新的可以浏览的信息

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402  } ] |

未登录

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"User Not Logged"** } |

## /resources/latest/{timestamp} GET

获取时间点之前的resource（之前没有）

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 久远的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [] |

正常的时间

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 正常的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402  } ] |

未登录：见前面 /resources/latest GET

## /resources/friends/{timestamp} GET

获取时间点之前的resource（之前没有）

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 久远的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [] |

正常的时间

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 正常的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402  } ] |

未登录：见前面 /resources/friends/latest GET

## /resources/self/{timestamp} GET

获取时间点之前的resource（之前没有）

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 久远的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [] |

正常的时间

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 正常的时间 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402  } ] |

未登录：见前面 /resources/self/latest GET

## /resources POST

创建一个resource（正常）

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | {  “emotion”:1,  “type”:2  } |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"id"**:1,  **"name"**:**"name1"**,  **"timestamp"**:1468909652402 } |

创建一个resource 未登录

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 未登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"User Not Logged"** } |

创建，内容类型错误

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | {  “emotion”:1,  “type”:2  } |
| 观察点 | 控制台输出 |
| 预期结果 | 控制台报错 |

## others

改变resource的公开属性

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | /resource/{resource\_id}/public  /resource/{resource\_id}/private |
| 观察点 | 控制台输出 |
| 预期结果 | 返回成功信息 |

改变resource公开属性，但是resource不存在或者未登录

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或者未登录 |
| 测试输入 | /resource/{resource\_id}/public  /resource/{resource\_id}/private  resource\_id不存在 |
| 观察点 | 控制台输出 |
| 预期结果 | 返回失败信息 |

## /resources/{resource\_id}/words GET/POST

获取resource的word

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"id"**:2,  **"content"**:**"life is so different here. when I feel the life before** , I indeed get my life.**",  "timestamp"**:1248794565 } |

创建或者更改resource的word

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id  {  **"content"**:**"I wanna change my life"** } |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"failed for some reason"** }  or {  **"success"**:**"create succeeded"** } |

创建或获取resource的word，但没有登录或者resource不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id  以及对应的输入数据（见上） |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"failed for some reason"** } |

## /resources/{resource\_id}/image GET/POST

获取resource的image

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id |
| 观察点 | 控制台输出 |
| 预期结果 | 图片文件流 |

创建或者更改resource的image

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id  图片文件流 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"Something's wrong"** }  or {  **"success"**:**"image upload succeeded"** } |

创建resource的image，但没有登录或者resource不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或未登录 |
| 测试输入 | resource\_id  以及对应的输入数据（见上） |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"failed for some reason"** } |

获取resource的image，但是没有登录或resource不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或未登录 |
| 测试输入 | resource\_id  以及对应的输入数据（见上） |
| 观察点 | 控制台输出 |
| 预期结果 | 获得默认的图片的图片流 |

## others

创建tag

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id  tag\_name  /resources/resource\_id/tags/{tag\_name} |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"succeeded"** } |

获取tags

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id  tag\_name  /resources/resource\_id/tags GET |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"name"**:**"heart"** }  {  **"name"**:**"love"** }  {  **"name"**:**"gay"** } ] |

未登录或者发生错误

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录或未登录 |
| 测试输入 | resource\_id  tag\_name  /resources/resource\_id/tags GET  or  /resources/resource\_id/tags/{tag\_name} POST |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"failed for some reason"** } |

# URI-API 集成测试 Friend

## /friends GET

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"myName"**:**"Sturmfy"**,  **"frName"**:**"name1"** } ] |

## /friends/{username} DELETE

存在用户

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"success"** } |

用户不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"fail"** } |

## /friends/apply GET

获取所有的申请

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"applyer"**:**"Sturmfy"**,  **"applyee"**:**"name1"**,  **"status"**:**"-"** }  {  **"applyer"**:**"Sturmfy"**,  **"applyee"**:**"name2"**,  **"status"**:**"y"** }  {  **"applyer"**:**"Sturmfy"**,  **"applyee"**:**"name3"**,  **"status"**:**"n"** } ] |

## /friends/apply/{username} POST

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"success"** } |

## /friends/make GET

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 无 |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"applyer"**:**"Sturmfy"**,  **"applyee"**:**"name1"**,  **"status"**:**"-"** } ] |

## /friends/make/{username} POST/DELETE

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"success"** } |

申请人不在申请列表

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | username |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"fail"**:**"failed"** } |

## 未登录错误

当未登录的时候控制台一律输出

{

“fail”:”User Not Logged”

}

# URI-API 集成测试 Mark Music And Book

## /resources/{resource\_id}/marks GET

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:12,  **"item\_id"**:235,  **"this\_name"**:**"name1"**,  **"content"**:**"how do you feel about this?"**,  **"time"**:1228242156224,  **"mark\_on\_id"**:**null** }  {  ...  } ] |

resource\_id 不存在

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 错误的resource\_id |
| 观察点 | 控制台输出 |
| 预期结果 | []为空 |

## /resources/{resource\_id}/marks/{timestamp} GET

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id， timestamp |
| 观察点 | 控制台输出 |
| 预期结果 | [  {  **"id"**:12,  **"item\_id"**:235,  **"this\_name"**:**"name1"**,  **"content"**:**"how do you feel about this?"**,  **"time"**:1228242156224,  **"mark\_on\_id"**:**null** }  {  ...  } ] |

久远的timestamp

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id, timestamp |
| 观察点 | 控制台输出 |
| 预期结果 | []为空 |

## /resources/{resource\_id}/music/{hashcode} POST

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id, hashcode |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"7aed5f321596a3d"** } |

## /resources/{resource\_id}/book/{ISBN} POST

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | resource\_id, ISBN |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"success"**:**"978-123-7-1234-1234"** } {  **"fail"**:**"Invalid ISBN Format"** } |

# URI-API 集成测试 Others

## /statistics/visitors, ./resources/new, ./friends/new, ./resources GET

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 直接获取 |
| 观察点 | 控制台输出 |
| 预期结果 | {  **"amount"**:**"1068"** } |

## /recommend/user GET

要求已经登录

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 测试输入 | 直接获取 |
| 观察点 | 控制台输出 |
| 预期结果 | [  “a”,  “b”  ] |

# 权限控制测试

## 资源获取

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录，并且用不同身份登录 |
| 测试输入 | 1. 分别创建各自的private资源 2. 分别获取 |
| 观察点 | 返回输出 |
| 预期结果 | 不会有对方的private资源 |

## 推荐获取

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录，并且用不同身份登录 |
| 测试输入 | 1. 分别创建各自的public资源  2. 分别获取推荐 |
| 观察点 | 返回输出 |
| 预期结果 | 获取的推荐用户不完全一致 |

## 跨域修改

|  |  |
| --- | --- |
| 项目 | 描述或伪代码 |
| 前置条件 | 已经登录 |
| 测试输入 | 1. 通过API对不属于自己的resource进行修改 2. 对不属于自己的private资源进行查看 |
| 观察点 | 控制台输出 |
| 预期结果 | 无修改权限，被拒绝，给出拒绝信息 |