<LETMEKNOW通知app>

测试用例

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <日/月/年> | <x.x> | <详细信息> | <姓名> |
| <1/9/2017> | <1.0> | <定义测试用例> | <景致> |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 概述 4

1. 测试功能性需求
2. 测试非功能性需求 4

测试用例

# 简介

本文档记录了对整个工程（包括安卓端，windows管理员端，服务器端）的功能正确性和性能的测

试。

## 目的

保证工程各部分以及整体整合的正确性。确保工程的各项功能都能够运转。考察工程在固定硬件设施和网络条件下的性能。

## 范围

测试用例涵盖了整个工程（包括安卓端，windows管理员端，服务器端）的功能性需求和非功能性需求。

测试将依照本文档进行，测试报告会受到本文档影响。

## 概述

本文档第二小节为各个测试用例的具体呈现，以功能性和非功能性需求、不同客户端和服务器端、

不同用户的操作为分类依据。

# 测试功能性需求

**2.1 安卓端操作**

1. **登录**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 1. admin, 123456（已有用户，正确的用户名密码） 2. admin, null（已有用户，不输入密码） 3. admin, 654321（已有用户，正确的用户名和错误的密码） 4. null, 666666（不输入用户名，随意输入密码） 5. null, null（不输入用户名和密码） 6. testuser1, imtestuser1（登入一个圈主用户，正确的用户名密码） 7. testuser2, imtestuser2（登入一个通知方用户，正确的用户名密码） 8. testuser3, imtestuser3（登入一个既不是圈主又不是通知方的用户，正确的用户名密码） | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 用户输入登录信息并点击登录按钮 | 控制用户输入的用户名密码是否为空，是否与数据库中的记录匹配 | 如果用户输入的用户名或密码为空，则提示用户该域不能为空  如果用户名和密码匹配成功，则登陆成功，并跳转向主界面  否则，登录不成功，提示用户错误的用户名或密码 |
|  | | |
| 后置条件： | 若用户登录成功，服务器端用token记录下用户状态为登录，否则不变 | |

1. **注册**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 用户在登录界面点击注册按钮 | |
|  | | |
| 测试输入 | 1. admin, 123456, 123456, [123@qq.com,](mailto:123@qq.com,) DongChuan Str.（尝试注册已有用户名） 2. testuser4,null, null, null, null（信息填写不完整） 3. testuser4, imtestuser4, imtetuser4, [123@qq.com,](mailto:123@qq.com,) DongChuan Str.（注册新用户，信息填写完整，两次输入密码一致） 4. testuser4, imtestuser4, imtetuser6, [123@qq.com,](mailto:123@qq.com,) DongChuan Str.（注册新用户，信息填写完整，两次输入密码不一致） | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 用户点击注册的按钮 | 无 | 跳转至注册界面 |
| 用户输入注册信息并点击注册按钮 | 控制用户输入不同的注册信息 | 若用户输入的注册信息填写不完整，注册失败，提示用户请填写完整信息  若用户两次输入的密码不一致，注册失败，提示用户检查  若用户填写的用户名已经被注册，注册失败，提示用户重名  否则，注册成功，服务器更改数据库加入新的用户数据，将用户导向登录界面 |
|  | | |
| 后置条件： | 无 | |

1. **加入圈子**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 扫描二维码加入圈子 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 控制用户扫描已加入的圈子的二维码或为加入的圈子的二维码 | 若用户本来不在该圈子中，扫描后服务器收到请求，更改数据库将该用户加入圈子中  若用户本来在该圈子中，扫描后无事发生，状态不改变 |
|  | | |
| 后置条件： | 之后用户浏览自己已加入的圈子是将会看到该圈子 | |

1. **退出圈子**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 用户至少加入了一个圈子 | |
|  | | |
| 测试输入 | 在圈子界面点击退出圈子选项 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | 服务器收到请求，更改数据库将该用户从圈子中删除 |
|  | | |
| 后置条件： | 之后用户浏览自己加入的圈子将不会看到该圈子 | |

1. **修改圈子信息**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 该用户为该圈的圈主 | |
|  | | |
| 测试输入 | 在圈子界面点击修改圈子信息，输入修改的信息，并点击确定 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 在圈子界面点击修改圈子信息 | 无 | 跳转到修改圈子信息界面 |
| 填入修改的圈子信息并点击确定 | 无 | 服务器收到请求，更改数据库储存新的圈子信息 |
|  | | |
| 后置条件： | 无 | |

1. **踢出成员**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 该用户为该圈的圈主 | |
|  | | |
| 测试输入 | 在圈子界面点击踢出，从成员列表中选择成员，并点击确定 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 在圈子界面点击踢出 | 无 | 向服务器请求成员列表，跳转到成员列表界面并显示 |
| 选择成员并确定 | 无 | 服务器收到请求，更改数据库将被选择的成员踢出该圈子 |
|  | | |
| 后置条件： | 无 | |

1. **发送通知**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 输入一条通知  “请尽快缴纳学费”  由admin用户发给testuser1 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | Testuser1用户收到推送 |
|  | | |
| 后置条件： | 无 | |

1. **回复通知**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 其他用户向该用户发送了通知 | |
|  | | |
| 测试输入 | 无 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 收到通知的用户点击确认按钮后通知方的状态 |  | 服务器收到请求，更改数据库改变该通知的状态。通知方再向服务器查询的时候能得到已经回复的状态和相对应的回执 |
|  | | |
| 后置条件： |  | |

1. **删除通知**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 无 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 用户选择通知并点击删除按钮 | 无 | 客户端本地储存的通知数据被删除，服务器收到请求，将对应通知的状态设为deleted |
|  | | |
| 后置条件： | 该通知将不再出现在通知列表中 | |

1. **查看用户详情**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 点击用户头像查看详情 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | 从本地缓存读取用户信息并展示 |
|  | | |
| 后置条件： | 无 | |

1. **查看圈子详情**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 该用户至少加入了一个圈子 | |
|  | | |
| 测试输入 | 在圈子列表中点击一个圈子 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | 服务器收到请求，并返回圈子名称、头像等信息给客户端。 |
|  | | |
| 后置条件： | 无 | |

**2.2 电脑端操作**

1. **登录**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 无 | |
|  | | |
| 测试输入 | 1. admin, 123456（管理员用户，正确的用户名密码） 2. admin, null（管理员用户，不输入密码） 3. admin, 654321（管理员用户，正确的用户名和错误的密码） 4. null, 666666（不输入用户名，随意输入密码） 5. null, null（不输入用户名和密码） 6. testuser1, imtestuser1（正确的用户名和密码，但不是管理员用户） | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 用户输入登录信息并点击登录按钮 | 控制用户输入的用户名密码是否为空，是否与数据库中的记录匹配 | 1.2.3.4.5同安卓端登录。6，登录失败，弹出对话框“该用户没有管理员权限” |
|  | | |
| 后置条件： | 无 | |

1. **查找并封禁用户**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 管理员已登录 | |
|  | | |
| 测试输入 | 输入用户名“testuser3”查找用户并封禁操作 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 查找并封禁用户 | 无 | 向服务器发送请求将该用户的状态改为封禁 |
|  | | |
| 后置条件： | 无 | |

1. **查找并封禁圈子**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 管理员已登录 | |
|  | | |
| 测试输入 | 输入名为“group1”的圈子并封禁操作 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 查找并封禁圈子 | 无 | 想服务器发送请求将该圈子的状态改为封禁 |
|  | | |
| 后置条件： |  | |

1. **查找通知**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 管理员已登录 | |
|  | | |
| 测试输入 | 输入内容包含“中文的内容”的通知并查找 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | 列出所有包含“中文的内容”的通知 |
|  | | |
| 后置条件： | 无 | |

1. **删除通知**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 管理员已登录，查找通知并查看详情 | |
|  | | |
| 测试输入 | 无 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 无 | 向服务器发送请求，从数据库中删除该条通知 |
|  | | |
| 后置条件： | 无 | |

1. **处理申请**

|  |  |  |
| --- | --- | --- |
| 前置条件： | 管理员已登录 | |
|  | | |
| 测试输入 | 用户点击处理申请按钮 | |
|  | | |
| 观察点 | 控制点 | 预期结果 |
| 无 | 管理员选择通过或者不通过 | 若通过，则向服务器发送请求，对应的用户或者圈子的状态发生改变 |
|  | | |
| 后置条件： | 无 | |

1. **测试非功能性需求**

**3.1易用性测试**

|  |  |  |
| --- | --- | --- |
| 测试标题 | 测试描述 | 测试结果 |
| 整体风格一致性 | 查看app中android端和管理员端各自是否有一致的风格。界面排布、配色是否统一。 | 通过 |
| UI易用性 | 查看app中android端和管理员端各自的每张界面是否简洁明了，便于用户使用。 | 通过 |
| 操作易用性 | 查看android端的每个功能的实现流程是否用户友好，会不会使用户产生困惑，有没有歧义。 | 通过 |
| 是否过度渲染 | 查看android端的每张activity是否因为层叠过多过度渲染而导致滑动卡顿。 | 通过 |

**3.2服务器性能测试**

(1)测试内容:大量并发访问+数据库索引+数据库写

(2)测试目标:数据库在高并发条件下的写入效率

(3)测试接口:/admin/users/activateUser/{id}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 请求数 | 并发数 | 平均处理时间 | 总处理时长 | 平均等待时间 |
| 1000 | 1000 | 0.491ms | 0.491s | 491ms |
| 10000 | 1000 | 0.299ms | 2.989s | 299ms |
| 25000 | 1000 | 0.264ms | 6.598s | 264ms |