**软件测试计划**



项目名称：表白墙网站开发

专业班级： 软件工程1903

小组成员：吴联想、王义博、郑航舰

指导教师： 杨枨老师

二零二一年十一月十一日

说明

1.《软件测试计划》STP)描述对计算机软件配置项CSCI,系统或子系统进行合格性测试的计划安排。内容包括进行测试的环境,测试工作的标识及测试工作的时间安排等。

2．通常每个项目只有一个STP,使得需方能够对合格性测试计划的充分性作出评估。

**版本记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 修订日期 | 版本/状态 | 修订人 | 备注 |
| 01 | 2021.11.11 | 0.1 | 郑航舰 | 初稿 |
| 02 | 2021.11.11 | 0.2 | 吴联想 | 主体内容编写 |
| 03 | 2021.12.12 | 0.3 | 郑航舰 | 部分单元测试图片插入 |
| 04 | 2021.12.15 | 0.4 | 王义博 | 登录注册个人中心单元测试 |
| 05 | 2021.12.22 | 0.5 | 郑航舰 | 测试用例 |
| 06 | 2021.12.29 | 0.6 | 郑航舰 | 集成、系统测试修订 |
| 07 |  |  |  |  |
| 08 |  |  |  |  |
| 09 |  |  |  |  |
| 10 |  |  |  |  |

**目录**

软件测试计划

目录

1.引言

1.1标识

1.2系统概述

1.3文档概述

1.4与其他计划的关系

1.5基线

2引用文件

3 软件测试环境

3.x(测试现场名称)

3.x.1软件项

3.x.2 硬件及固件项

3.x.3 其他材料

3.x.4 所有权种类、需方权利与许可证

3.x.5 安装、测试与控制

3.x.6 参与组织

3.x.7 人员

3.x.8 定向计划

3.x.9 要执行的测试

4 计划

4.1总体设计

4.1.1测试级

4.1.2 测试类别

4.1.3 一般测试条件

4.1.4 测试过程

4.1.5 数据记录归约和分析

4.2 计划执行的测试

4.2.x （被测试项）

4.2.x.y （测试的项目唯一标识符）

4.3 测试用例

4.4 集成测试

4.5 系统测试

5 测试进度表

6 需求的可追踪性

7.评价

7.1评价准则

7.2数据处理

7.3结论

8注解

附录

1 引言

1.1标识

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ ] 正式发布  [√] 正在修改 | 文件标识： | SE2021-G11-软件测试说明 |
| 当前版本： | 0.6 |
| 作者： | 吴联想、王义博、郑航舰 |
| 完成日期： | 2021-12-29 |

1.2系统概述

项目由G11小组在软件工程课程下提出，根据杨枨老师提出的建议，我们要求：用户注册、发帖（文字、图片）、删帖、看帖、评论、点赞、楼中楼、消息反馈、表白墙排序（根据发帖日期、点赞数）功能；管理员查收反馈信息、用户管理（查、删、重置密码、禁言）、置顶、精华帖的设置等功能，我们将争取做到精简界面，为用户带来良好的应用体验。

项目的目标是依照项目的要求，开发出表白墙网站，在开发的过程中学习软件工程课程内容，实现软件工程课程目标，通过软件工程课的评审，拿到软件工程课的学分。

1.3文档概述

本文档书写的目的在于对即将完成的G11小组的校园表白墙网站进行测试，在软件开发的早期阶段考虑测试问题，促使软件设计人员在设计时注意提高软件的可测试性，确定本项目软件测试的目标和方法，确定测试策略、测试方案、预期的测试结果和测试进度计划。

1.4与其他计划的关系

本计划是总体总体设计计划的一部分

1.5基线

软件需求规格说明

2引用文件

1.《计算机软件需求规格说明规范》（GB/T9385-2008）

2．《表白墙网站项目开发计划书》

3.《表白墙网站项目开发可行性分析报告》

4.《软件工程导论》（张海藩、牟永敏编著第六版）

5.《计算机软件产品开发文件编制指南》(GB8567—88）

3 软件测试环境

3.x(测试现场名称)

3.x.1软件项

IntelliJ IDEA 2021.2.3

3.x.2 硬件及固件项

笔记本电脑

3.x.3 其他材料

无

3.x.4 所有权种类、需方权利与许可证

30天体验

3.x.5 安装、测试与控制

HP Unified Functional Testing

Junity

3.x.6 参与组织

G11项目小组

3.x.7 人员

吴联想、王义博、郑航舰

3.x.8 定向计划

无

3.x.9 要执行的测试

需求测试、单元测试、集成测试

4 计划

4.1总体设计

4.1.1测试级

系统级

4.1.2 测试类别

需求测试、单元测试、集成测试

4.1.3 一般测试条件

每个测试应包括额定值、最大值和最小值；每个类型的测试都应该使用真实数据

4.1.4 测试过程

渐进测试，先测试每个模块的功能，再测试各个模块功能的接口

4.1.5 数据记录归约和分析

数据记录使用word记录结果并进行分析

4.2 计划执行的测试

4.2.1

封禁测试：

先在数据库中自定义一些用户，然后在管理系统中对这些用户执行封禁操作，观察这些被封禁的用户是否正常使用表白墙功能。

预期结果：被封禁的用户无法使用表白墙功能。

4.2.2

修改测试：

先在数据库中自定义一些用户，然后在管理系统中对这些用户的相关数据（如密码）等进行修改，然后在数据库中观察这些用户对应的值是否发生改变。

预期结果：这些用户对应的值是否发生改变

4.2.3

删除测试（管理员）：

先在数据库中自定义一些表白墙，然后在管理系统中对这些表白墙进行删除，然后在数据库中观察这些表白墙是否发生改变。

预期结果：表白墙被删除，用户无法在主页上看到被删除的表白墙。

4.2.4

精选测试（管理员）：

先在数据库中自定义一些表白墙，然后在管理系统中对这些表白墙进行精选，然后观察这些表白墙有没有在精选栏中出现。

预期结果：表白墙在精选栏中出现，并且普通栏中也可以看到那些表白墙。

4.2.5

浏览测试（管理员）：

浏览测试分为两个部分：

1. 让用户发表白墙，测试管理员系统中能否看到表白墙

预期效果：管理员能看到用户发的表白墙

1. 用户发多个表白墙，测试管理员是否能通过上下翻页来浏览所有的表白墙

预期结果：管理员能看到所有用户发的所有表白墙

4.2.6

发布测试（用户）：

1. 用户发布符合要求的表白墙

预期结果：提示发布成功，在浏览界面上能看到发布出来的表白墙。

1. 用户发帖正文内容超出字数限制

预期结果：提示字数超出限制，发布失败并且浏览界面上看不到发出来的表白墙。

1. 用户发帖标题字数超出限制

预期结果：提示字数超出限制，发布失败并且浏览界面上看不到发出来的表白墙。

4.2.7

删除测试（用户）：

用户可以在个人界面上查看自己发的表白墙并且删除表白墙。

预期结果：删除后的表白墙不会现实在浏览界面上。

4.2.8

浏览测试（用户）：

用户可以正常浏览发布的表白墙和精选表白墙。

预期结果：同上。

4.2.9

收藏测试（用户）：

用户可以在主页的正常浏览中点击收藏按钮收藏自己想要看的表白墙。

预期结果：收藏的表白墙能够在用户个人中心的“收藏”中看到表白墙。

4.2.10

注册测试：

1. 注册时输入密码不合规

预期结果：提示密码输入不合规并且注册失败。

1. 注册时输入学号错误

预期结果：真正的学号邮箱里不会收到发出来的邮件。

1. 输入全部合规

预期结果：注册成功并跳转到登录界面。

4.2.11

登录测试：

1. 输入错误的账号

预期结果：提示账号输入错误，登录失败

1. 输入错误的密码

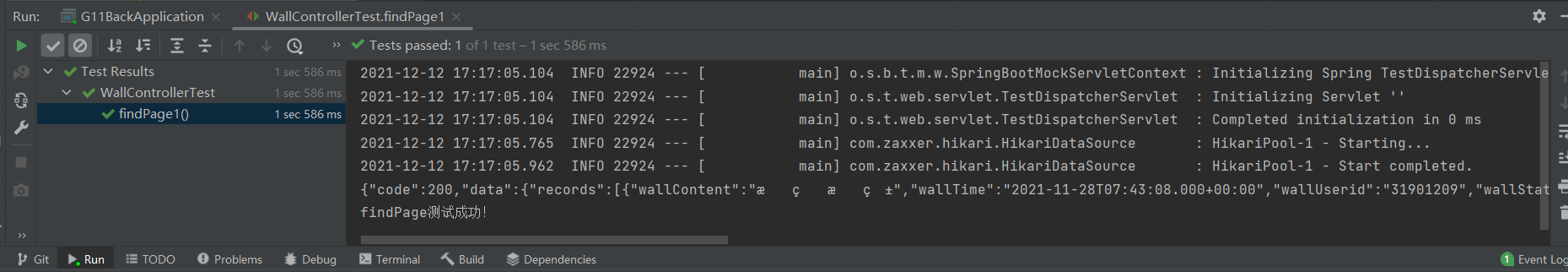
预期结果：提示密码输入错误，登录失败

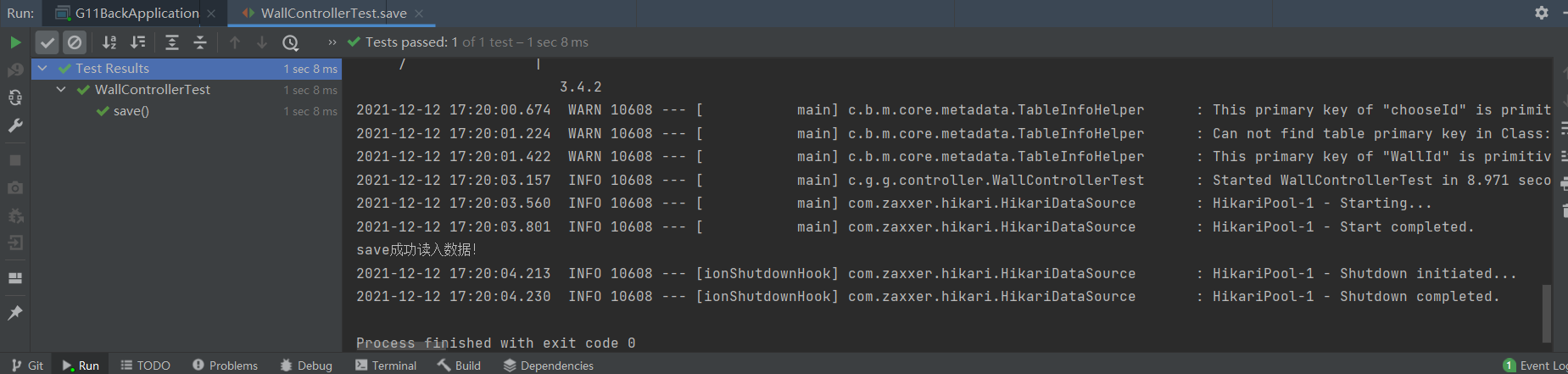
1. 账号密码输入正确

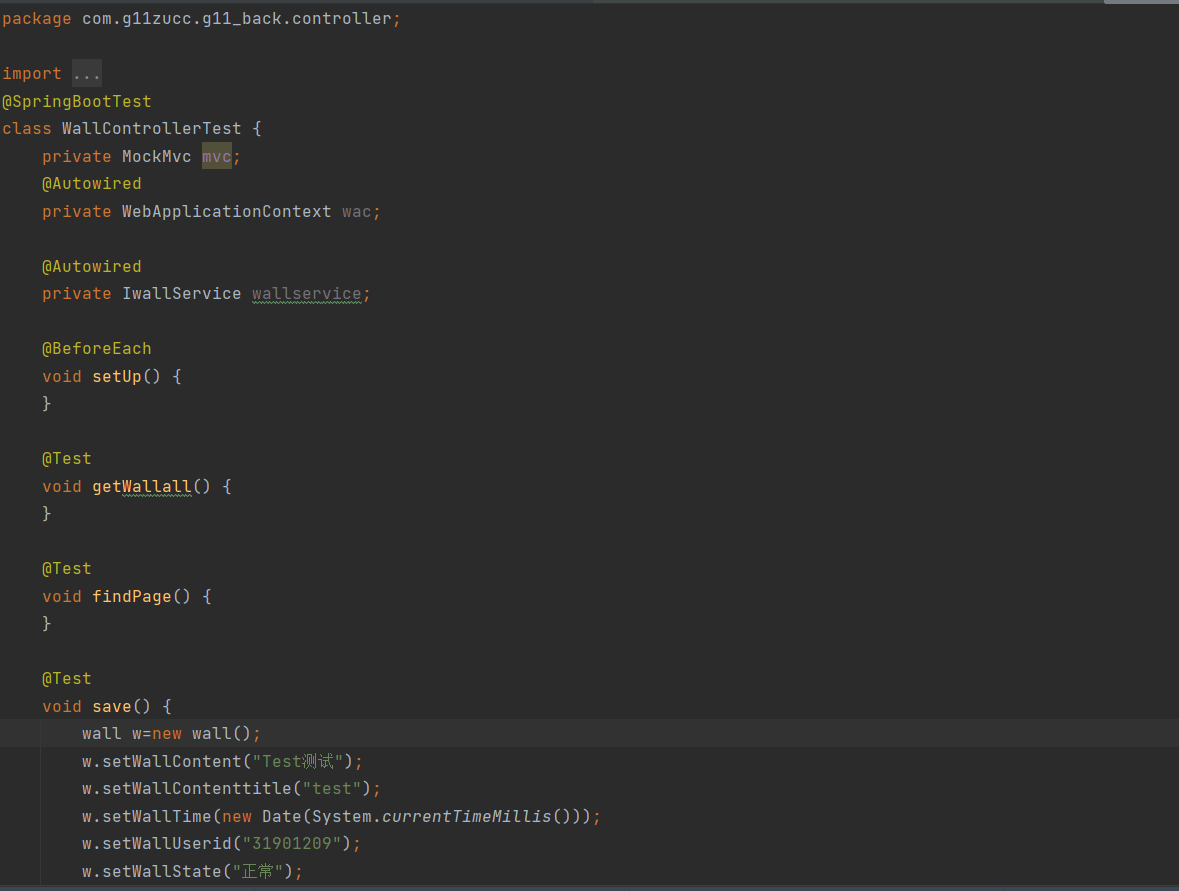
预期结果：提示登录成功，进入主页面。

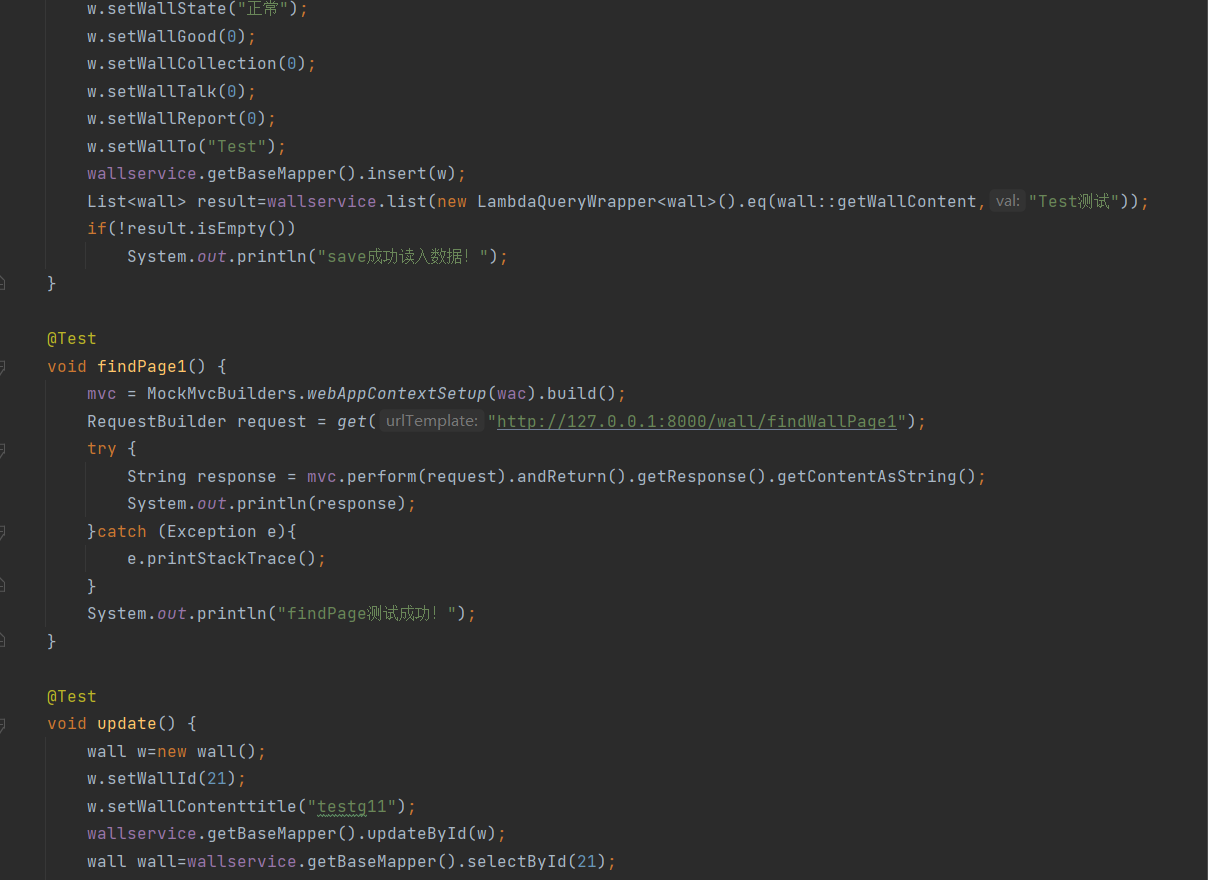
4.3 测试用例

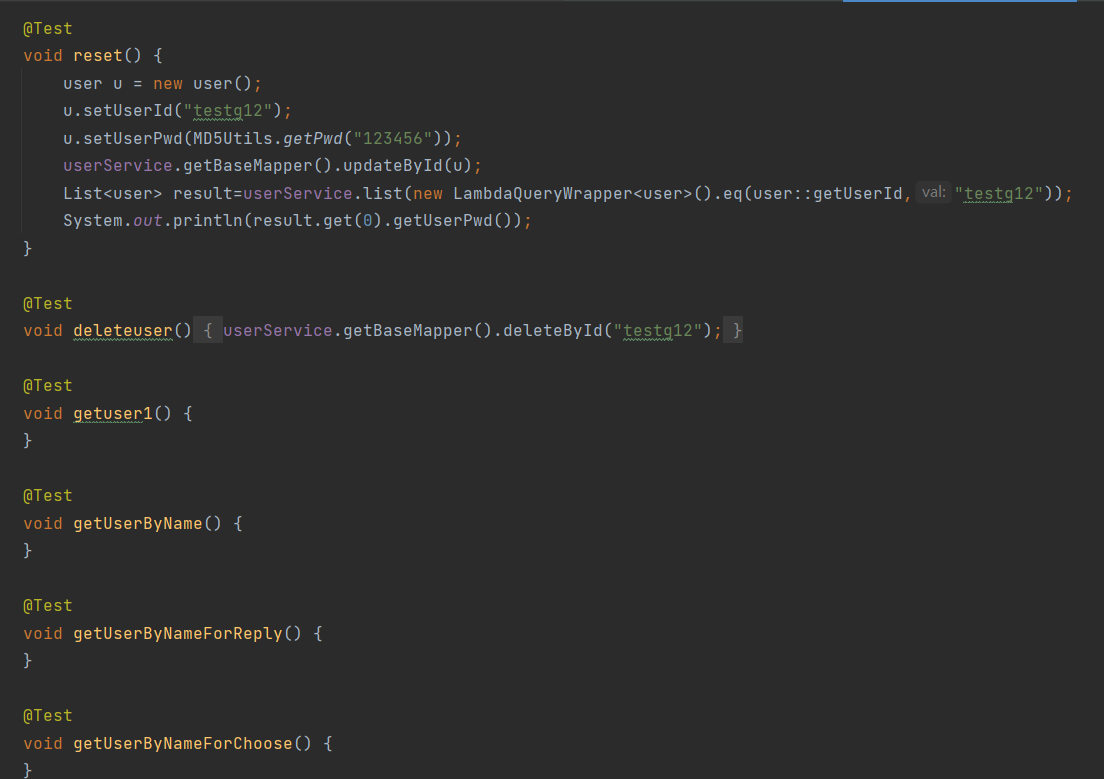
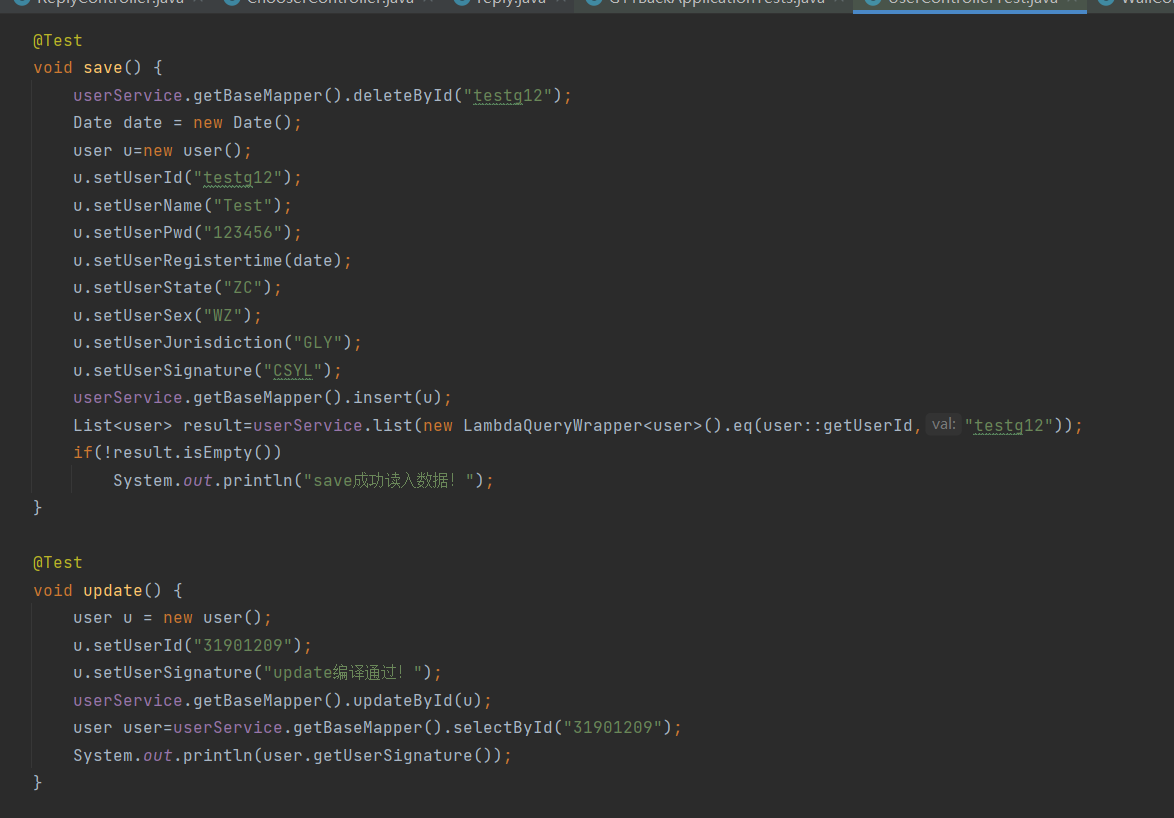
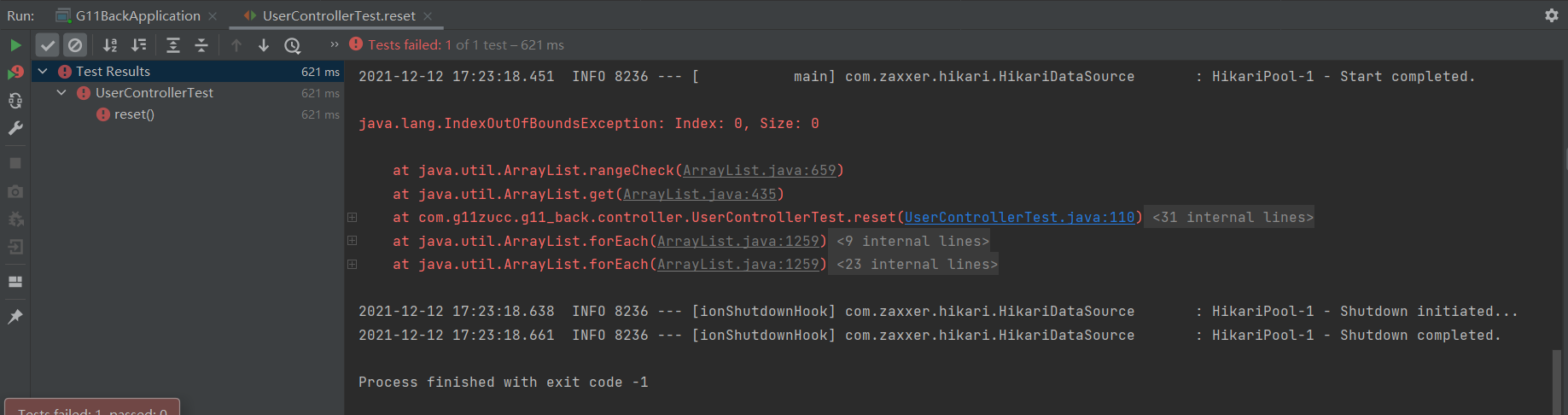
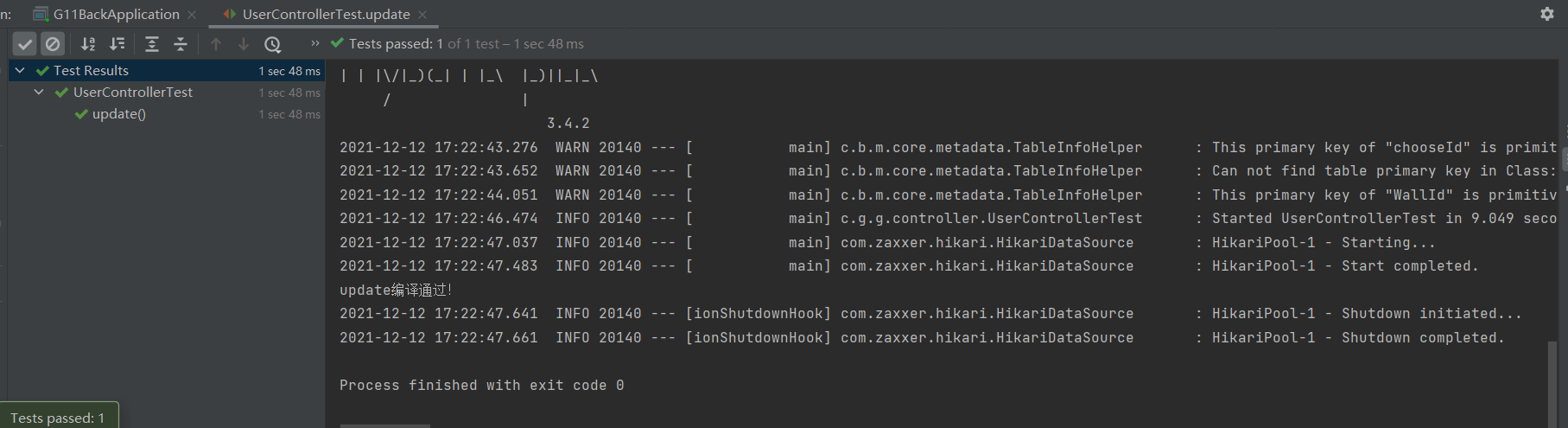
WallTest:



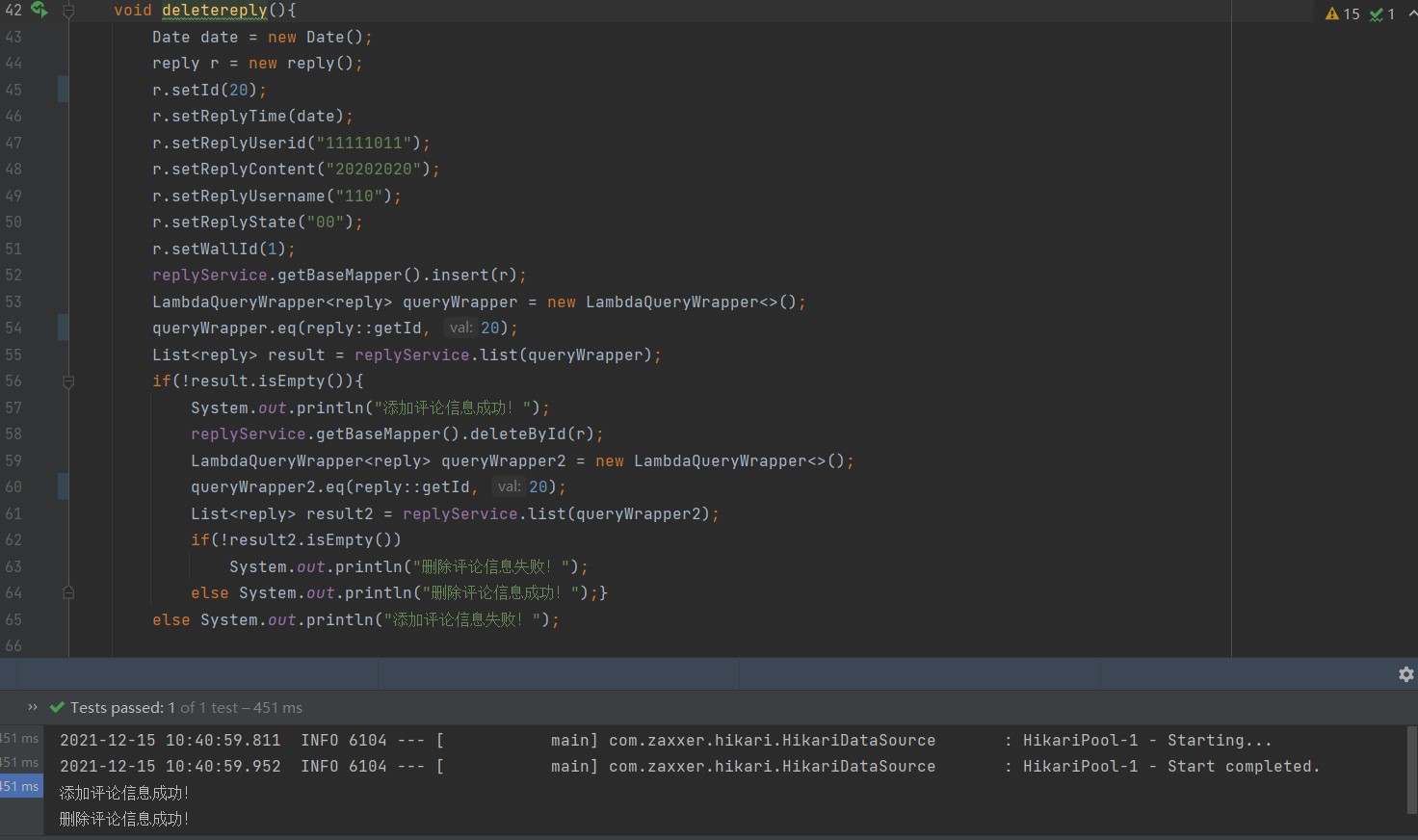




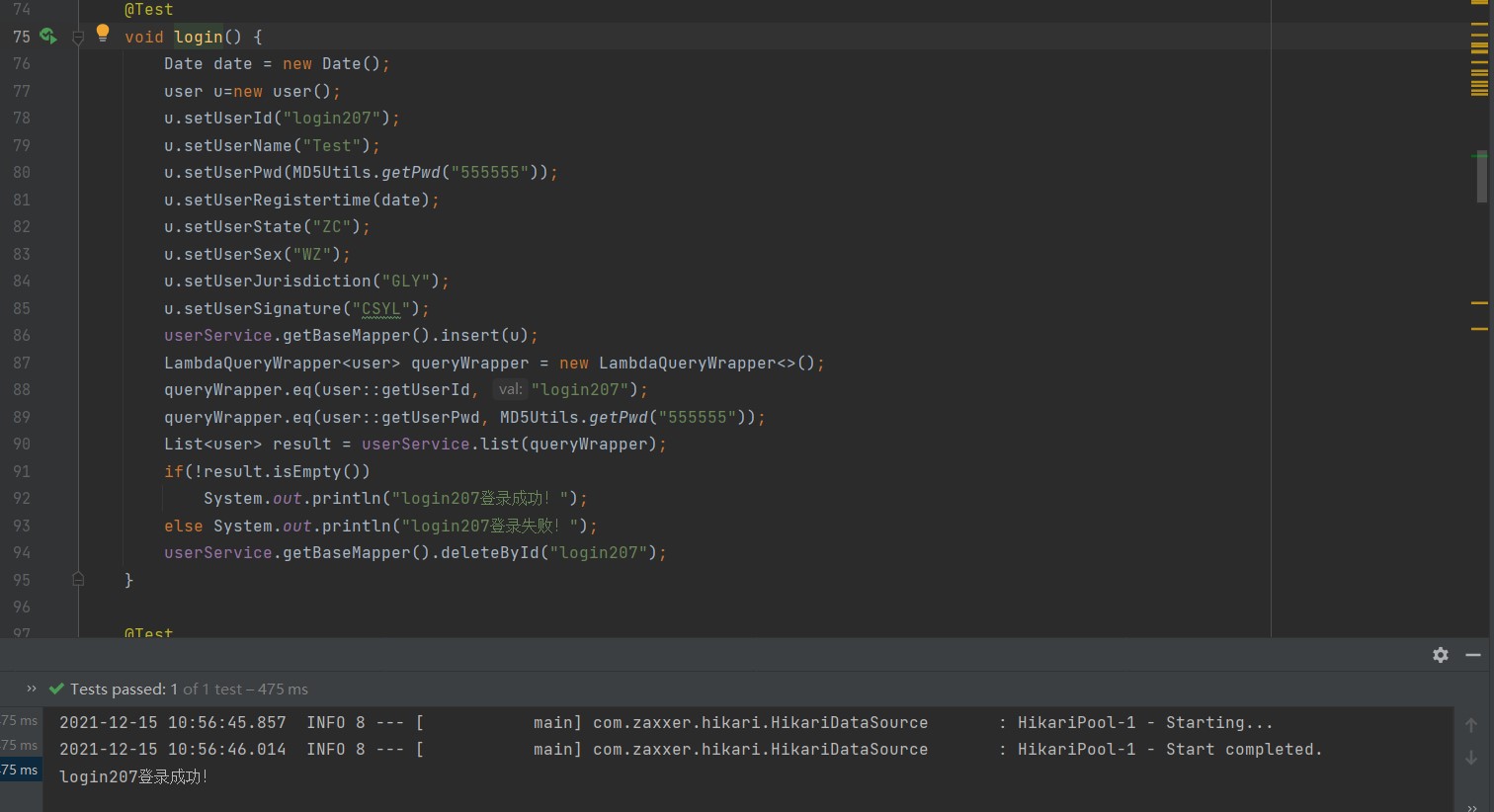
UserTest:



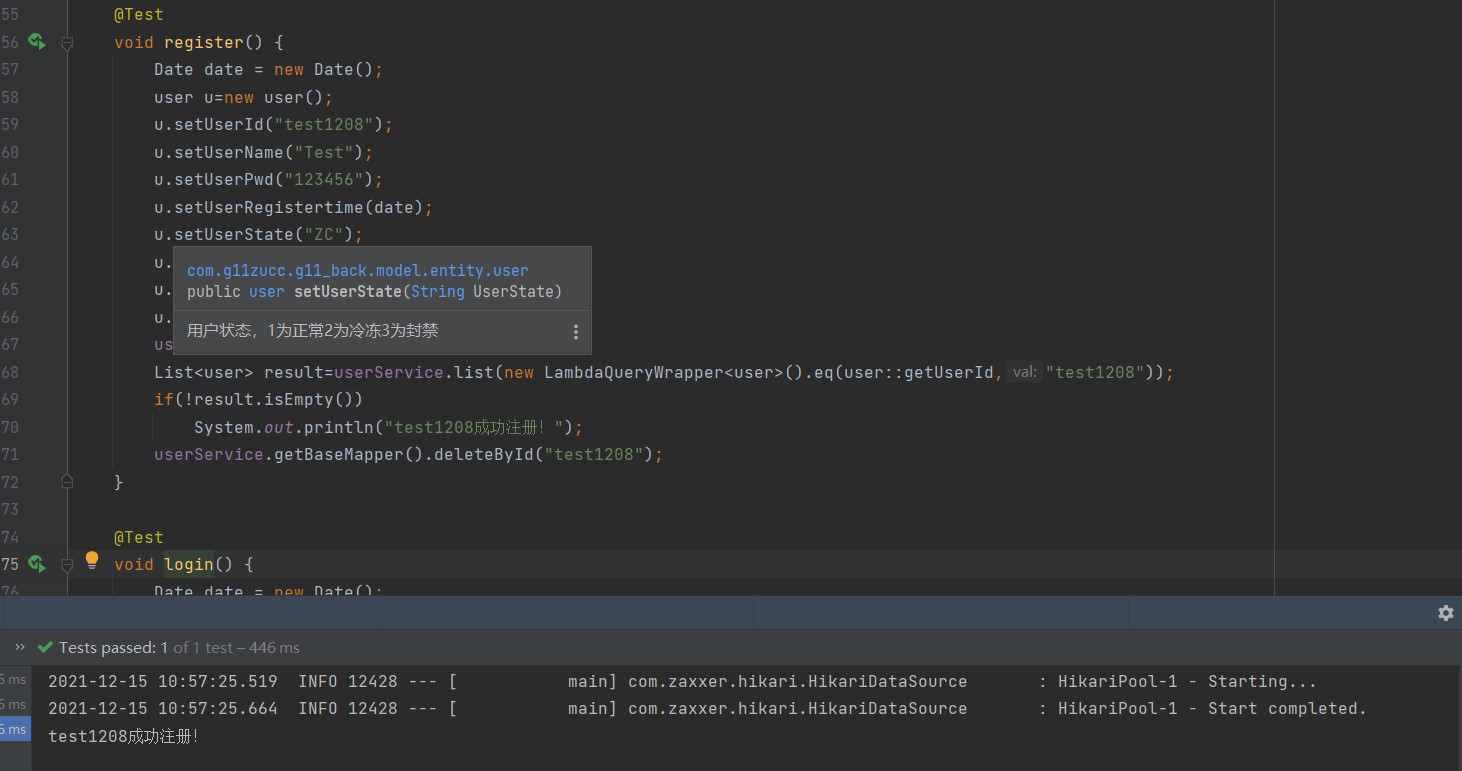
删除评论单元测试:



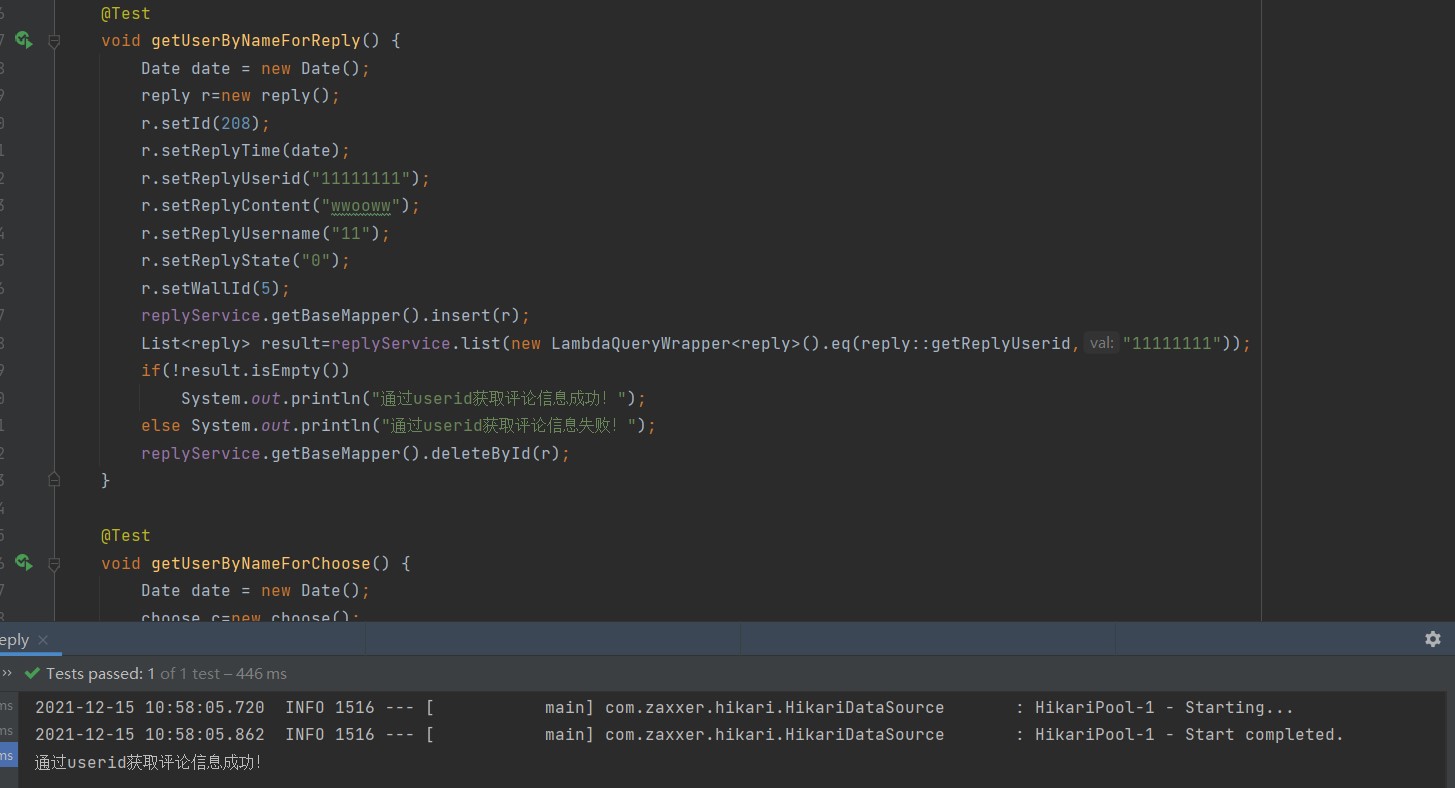
登录单元测试:



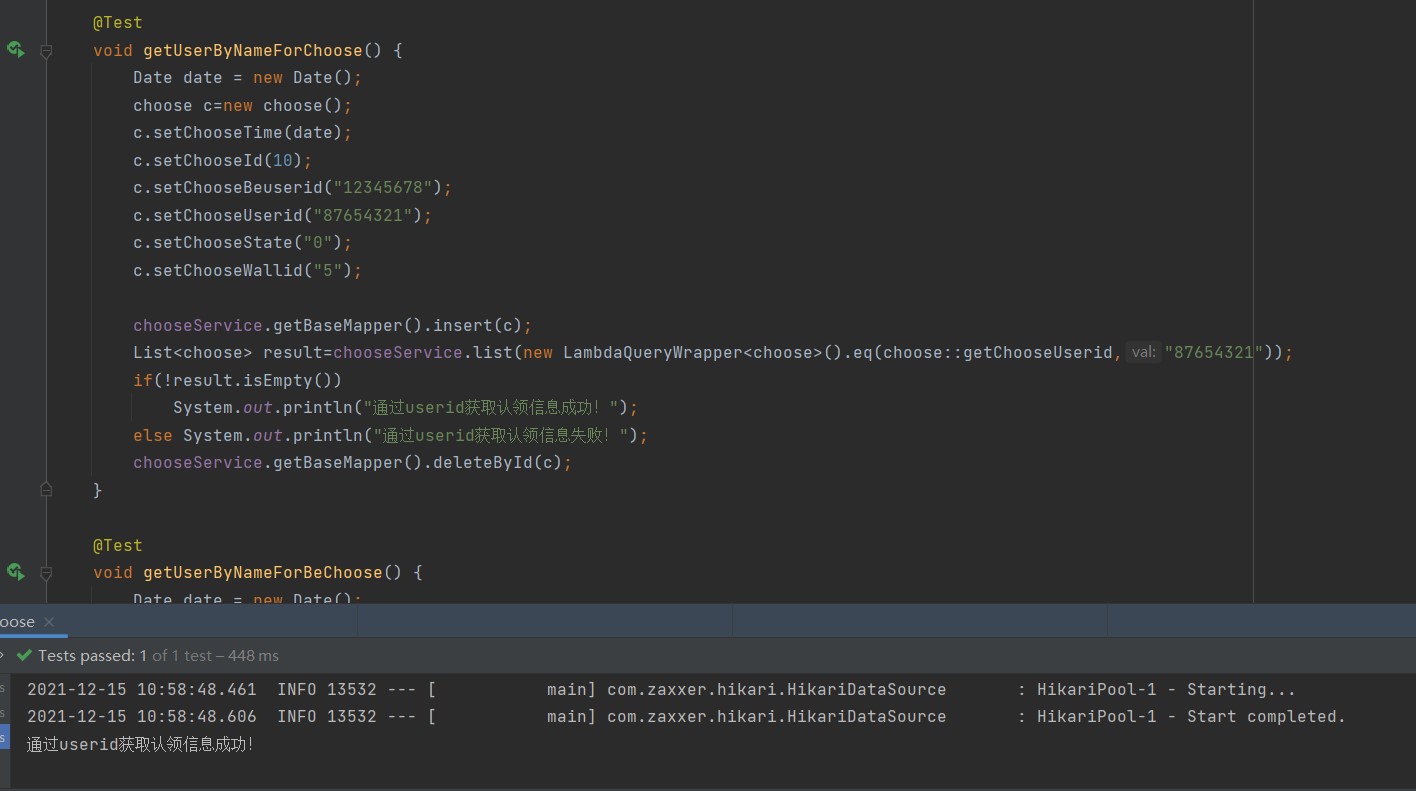
注册单元测试:



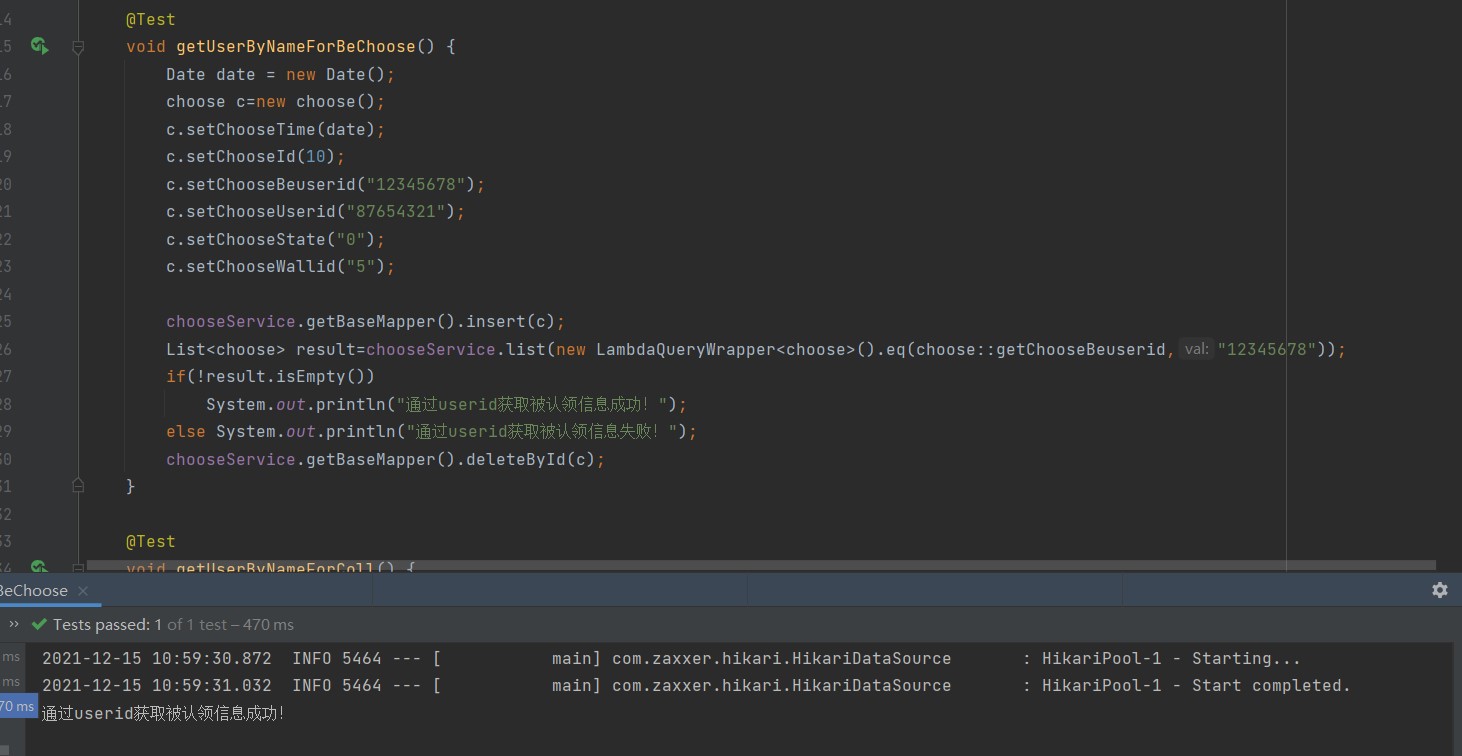
获取特定用户评论单元测试:



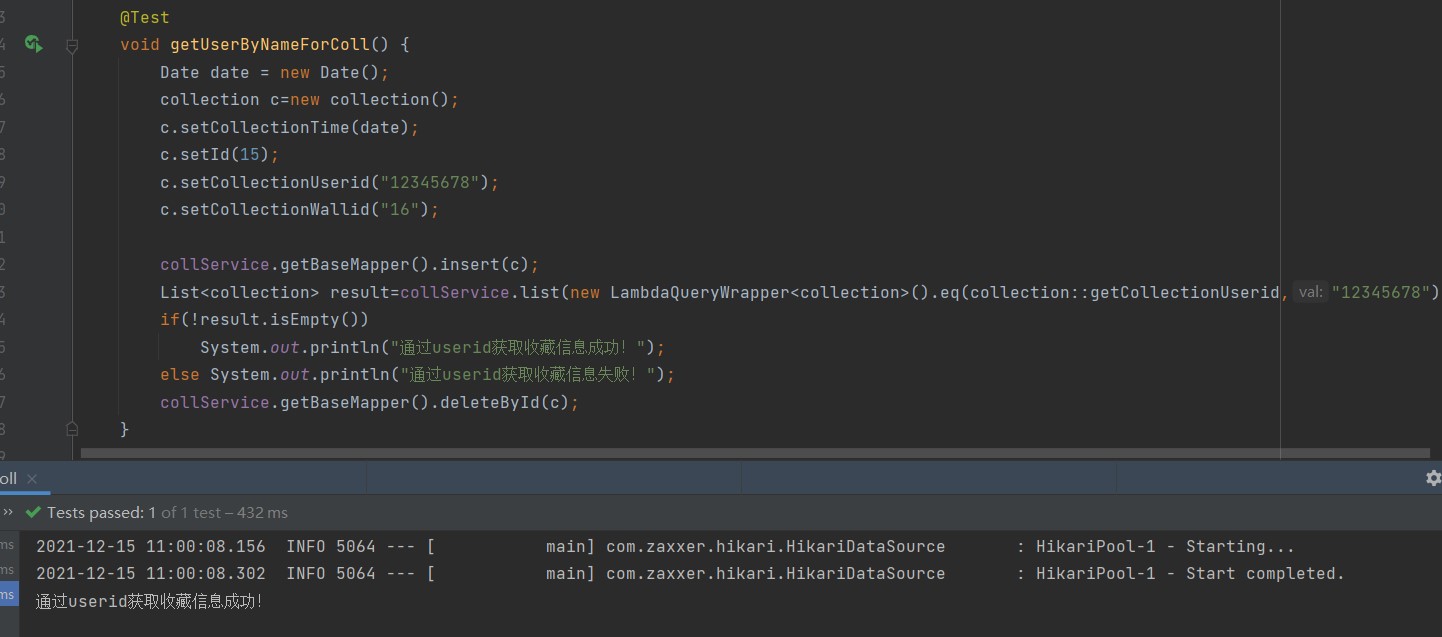
获取特定用户认领情况单元测试:



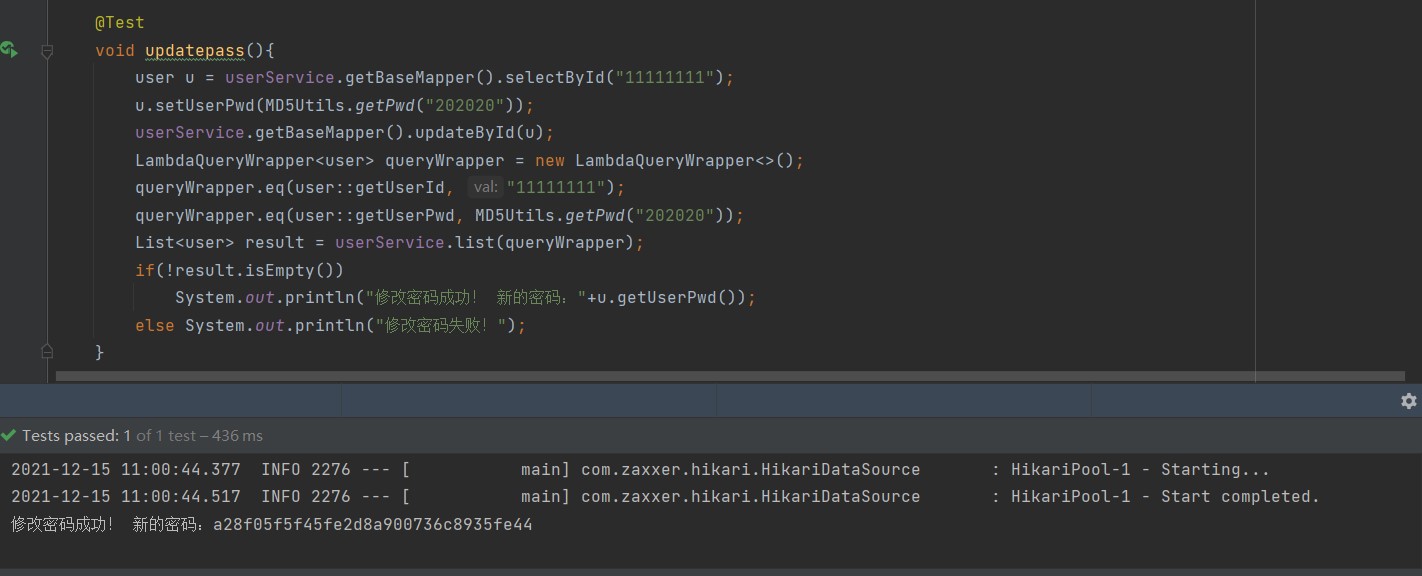
获取特定用户被认领情况单元测试:



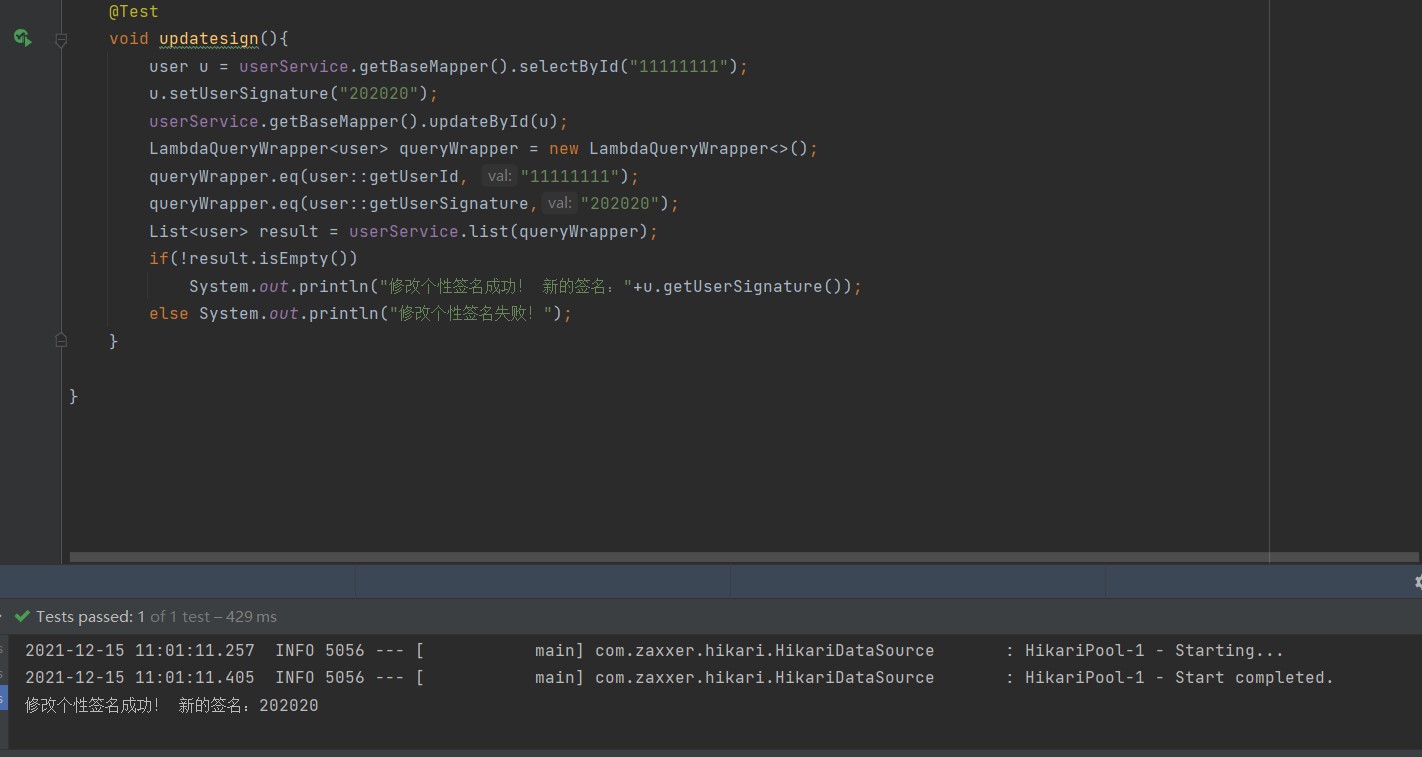
获取特定用户收藏单元测试:



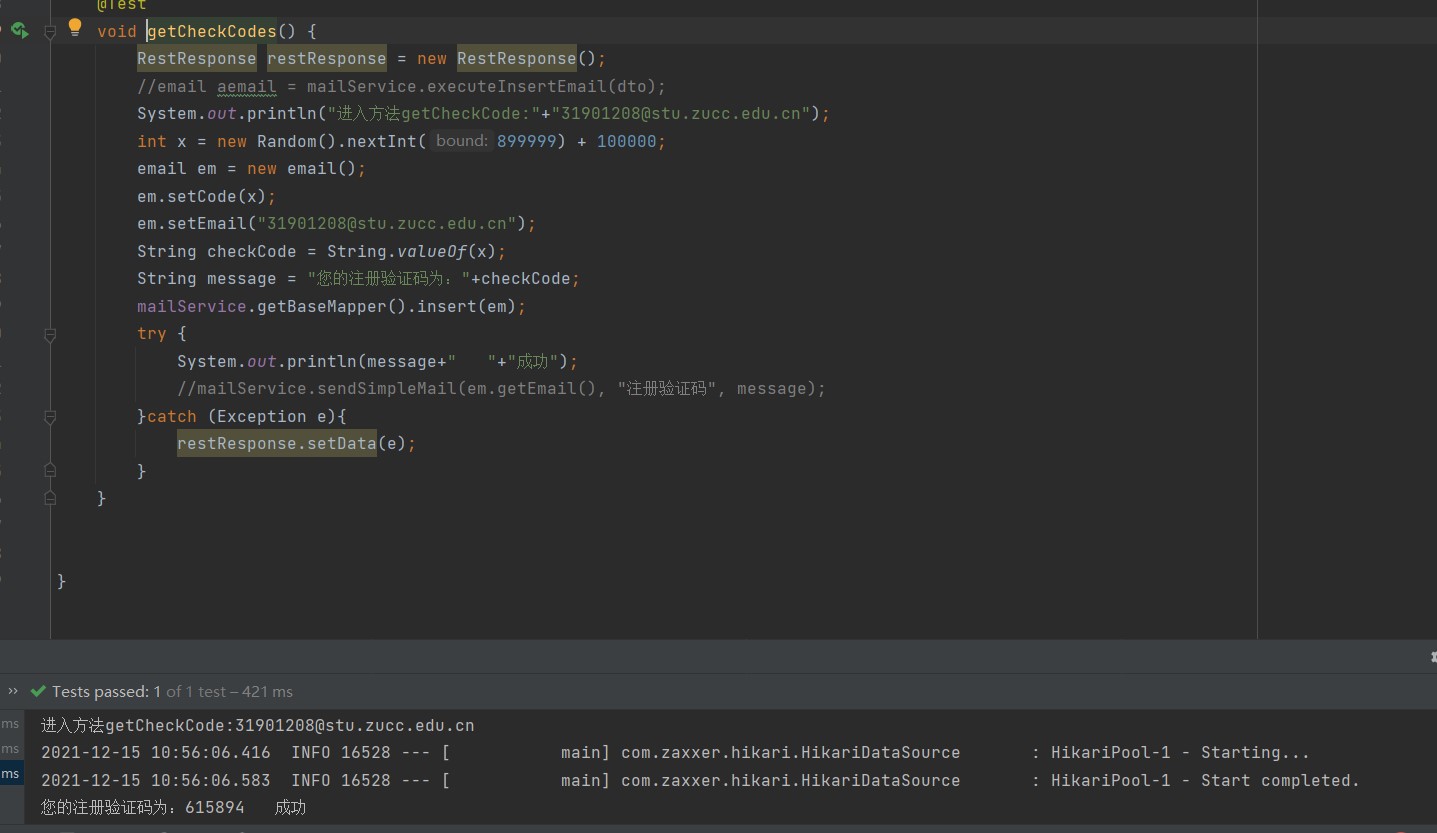
修改密码:



修改个性签名:



邮箱验证码单元测试:



UserControllerTest

1. Save(user)方法

描述：在数据库中插入一个user类型数据

路径：/user/save

输入：一个user类型数据（UserId=“testg12”，UserName=”Test”,UserPwd=”123456”,

UserState=”ZC”,UserSex=”WZ”,UserJurisdiction=”GLY”,UserSignature=”CSYL”）

输出：输出一句话“save成功读入数据！”

1. Update(user)方法

描述：在数据库中传入一个带有UserId的user数据，将数据库中原有的数据替换成该user数据

路径：/user/update

输入：一个user类型数据（UserId=”31901209”,UserSignature=”update编译通过”）3.

输出：查找“31901209”的数据时，输出这条数据的个性签名“update编译通过”

1. FindPage1(pageNum，pageSize,search)方法

描述:根据传入的参数分页输出user数据

路径：/user/findPage

输入：pageNum=1，pageSize=10，search=“”

输出：输出数据库中所有的user数据

1. register()方法

描述：注册

路径：/user/ register

输入：一个user类型数据（UserId=“test1208”，UserName=”Test”,UserPwd=”123456”,

UserState=”ZC”,UserSex=”WZ”,UserJurisdiction=”GLY”,UserSignature=”CSYL”）

输出：输出一句话“test1208成功注册！”

1. login（）方法

描述：登录

路径：/user/login

输入：用户ID（UserId=“login207”）和用户密码（UserPwd=“555555”）

输出：输出一句话“login207登录成功！”

1. getUserByNameForReply（）方法

描述：获取特定用户的评论

路径：/user/ reply/{username}

输入：用户ID（username）

输出：输出一句话“通过userid获取评论信息成功！”

1. getUserByNameForColl（）方法

描述：获取特定用户的收藏

路径：/user/ coll/{username}

输入：用户ID（username）

输出：输出一句话“通过userid获取收藏信息成功！”

1. getUserByNameForchoose（）方法

描述：获取特定用户的认领

路径：/user/ choose/{username}

输入：用户ID（username）

输出：输出一句话“通过userid获取认领信息成功！”

1. getUserByNameForBechoose（）方法

描述：获取特定用户的被认领情况

路径：/user/ bechoose/{username}

输入：用户ID（username）

输出：输出一句话“通过userid获取被认领信息成功！”

1. updatepass（）方法

描述：修改密码

路径：/user/ updatepass

输入：用户ID（UserId=“202020”）和用户密码（UserPwd=“555555”）

输出：输出一句话“修改密码成功！ 新的密码：555555”

1. updatesign（）方法

描述：修改个性签名

路径：/user/ updatesign

输入：用户ID（UserId=“202020”）和用户个性签名（updatesign =“202020”）

输出：输出一句话“修改个性签名成功！ 新的签名：202020”

WallControllerTest

1. Save(wall)方法

描述：在数据库中插入一个wall类型数据

路径：/wall/save

输入：一个wall类型数据（WallContent="Test测试“，WallContenttitle="test"，WallTime=new Date(System.*currentTimeMillis*())，  
WallUserid="31901209"  
WallState="正常"  
WallGood=0  
WallCollection=0  
WallTalk=0  
WallReport=0  
WallTo="Test"

）

输出：输出一句话“save成功读入数据！”

1. Update(wall)方法

描述：在数据库中传入一个带有WallId的wall数据，将数据库中原有的数据替换成该wall数据

路径：/wall/update

输入：一个user类型数据（wallId=21,WallContenttitle=”testg11”）

输出：查找“21”的wall数据时，输出这条数据的表白标题“testg11”

1. FindPage1(pageNum，pageSize,search)方法

描述:根据传入的参数分页输出wall数据

路径：/wall/findPage

输入：pageNum=1，pageSize=10，search=“”

输出：输出数据库中所有的wall数据

1. findWallPage (pageNum，pageSize,search)方法

描述:根据传入的参数分页输出wall数据

路径：/wall/findWallPage

输入：pageNum=1，pageSize=1，search=“正常”

输出：输出数据库中满足调教的所有的wall数据，并打印一句话“findPage测试成功！”

1. findWallPage 6(pageNum，pageSize,search)方法

描述:根据传入的参数按点赞数分页输出wall数据

路径：/wall/findWallPage6

输入：pageNum=1，pageSize=1，search=“正常”

输出：输出数据库中满足调教的所有的wall数据，并打印一句话“findPage测试成功！”

1. findWallPage 9(pageNum，pageSize,search)方法

描述:根据传入的参数按收藏数分页输出wall数据

路径：/wall/findWallPage9

输入：pageNum=1，pageSize=1，search=“正常”

输出：输出数据库中满足调教的所有的wall数据，并打印一句话“findPage测试成功！”

1. findHotWallPage (pageNum，pageSize,search)方法

描述:根据传入的参数按热度分页输出wall数据

路径：/wall/findHotWallPage

输入： wall w = new wall();

w.setWallGood(0);

w.setWallContent("test");

w.setWallId(1);

w.setWallTime(new Date(System.currentTimeMillis()));

w.setWallContenttitle("test");

w.setWallCollection(0);

w.setWallTalk(0);

w.setWallTo("asdasdasd");

w.setWallReport(0);

w.setWallUserid("notexist");

传入w

输出：输出数据库中满足调教的所有的wall数据，并打印一句话“findPage测试成功！”

1. insertWall（）方法

描述:根据传入的参数按热度分页输出wall数据

路径：/wall/ insertWall

输入：pageNum=1，pageSize=1，search=“正常”

输出：打印一句话“wall插入成功！！”

FeedbackControllerTest

1. insertFeedback (wall)方法

描述：在数据库中插入一个feedback类型数据

路径：/feedback/insertFeedback

输入：一个feedback类型数据（FeedbackTime=new Date(System.*currentTimeMillis*());  
FeedbackContent="这是测试"  
FeedbackState="正常"  
FeedbackId=1  
FeedbackTitle="飞时达"  
FeedbackUserid="31901209"

）

输出：“feedback插入成功“

CollectionControllerTest

1. insertCollection (collection)方法

描述：在数据库中插入一个collection类型数据

路径：/ collection /insertCollection

输入：一个reply类型数据

collection c = new collection();

c.setId(1);

c.setCollectionUserid("31901209");

c.setCollectionWallid("5");

c.setCollectionTime(new Date(System.currentTimeMillis()));

collectionservice.getBaseMapper().insert(c);）

输出：“collection插入成功！！“

1. deletecoll（）方法

描述：删除收藏

路径：/reply/ deletecoll

输入：评论ID（Id=“18”）

输出：输出一句话“删除收藏信息成功！”

# MailControllerTest

1. getCheckCodes()方法

描述：获取邮箱验证码并发送邮件

路径：/mail/ getCheckCode

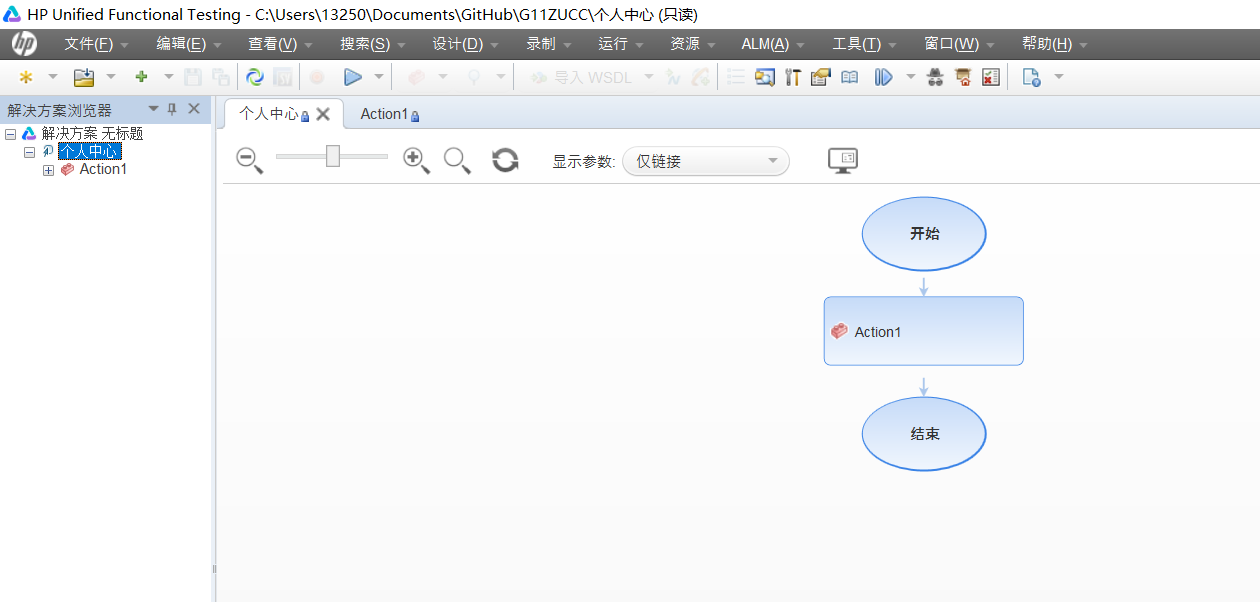
输入：邮箱（email=“1456034259@qq.com”）

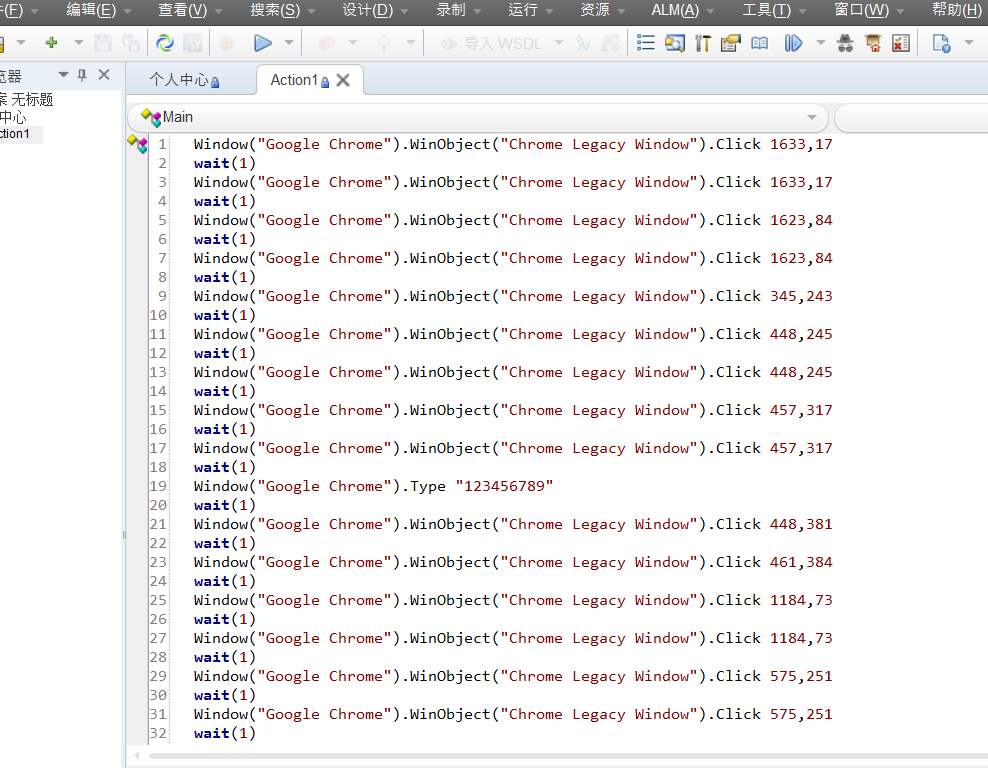
输出：输出一句话"您的注册验证码为：xxxxxx"

4.4 集成测试

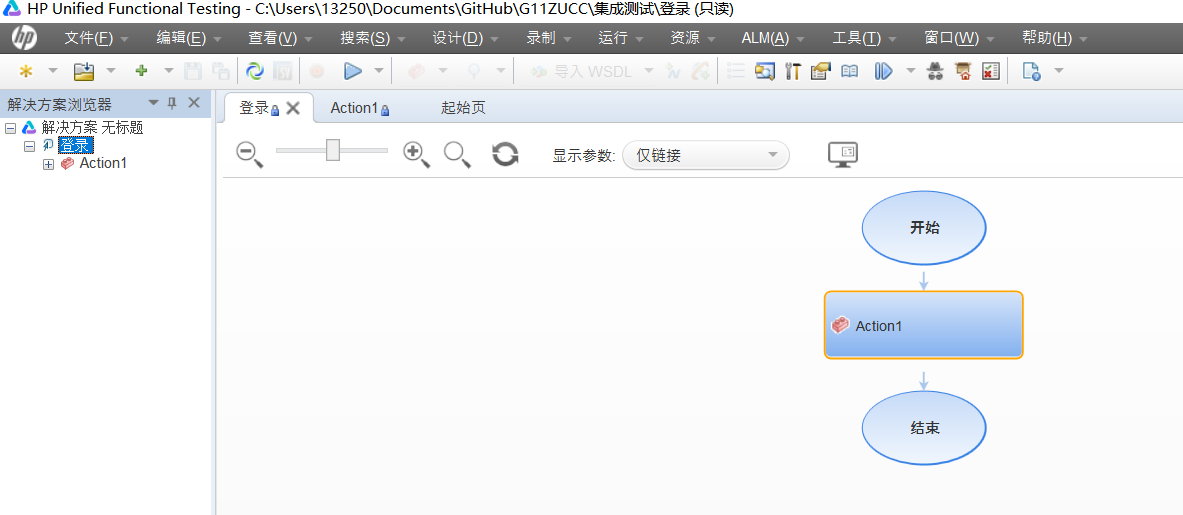
集成测试采用HP Unified Functional Testing工具进行测试，下面是集成测试部分截图，详细内容见HP Unified Functional Testing工具。

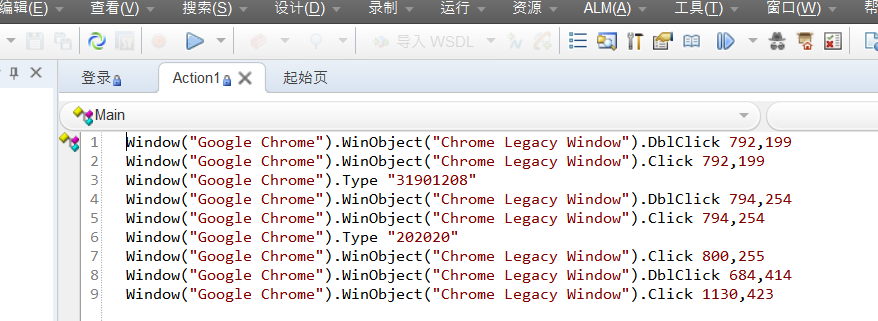
**个人中心**



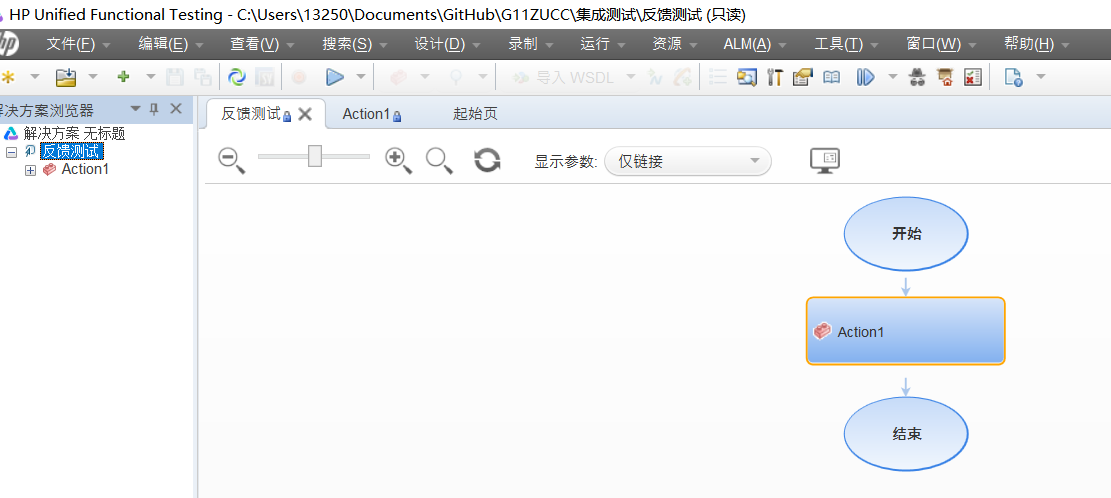


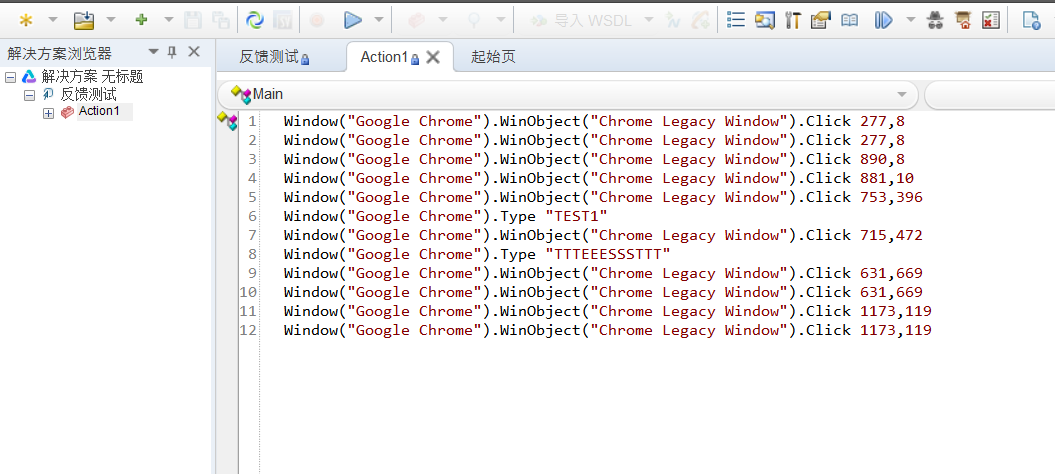
**登录**



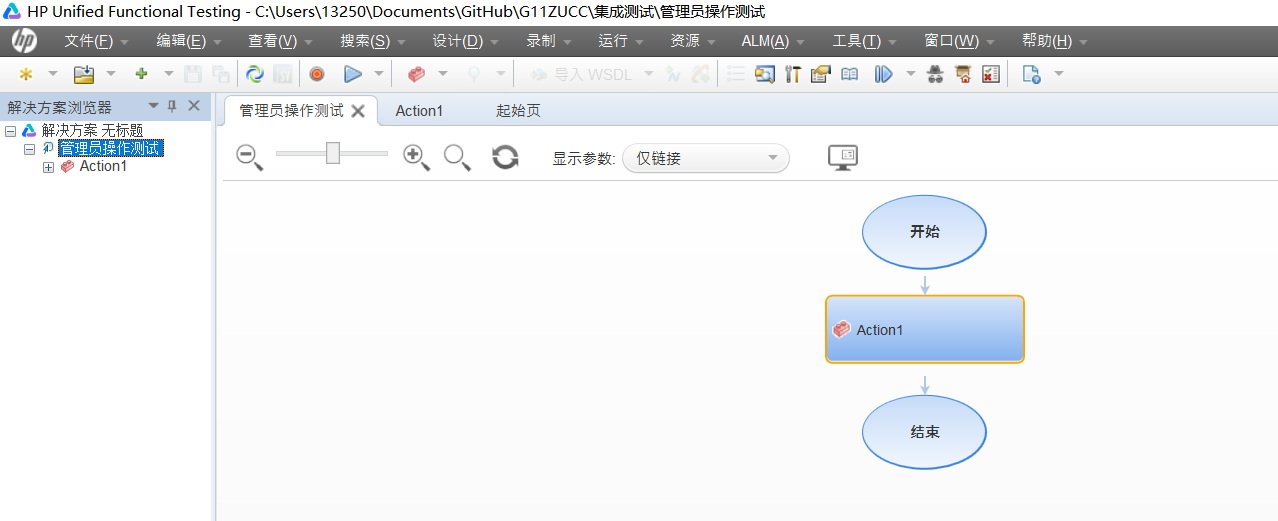


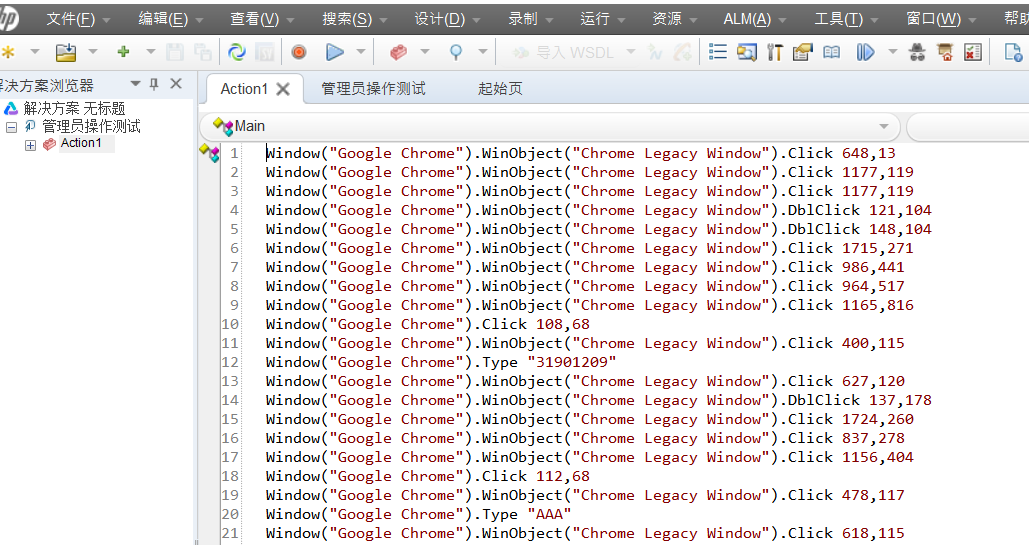
**反馈**



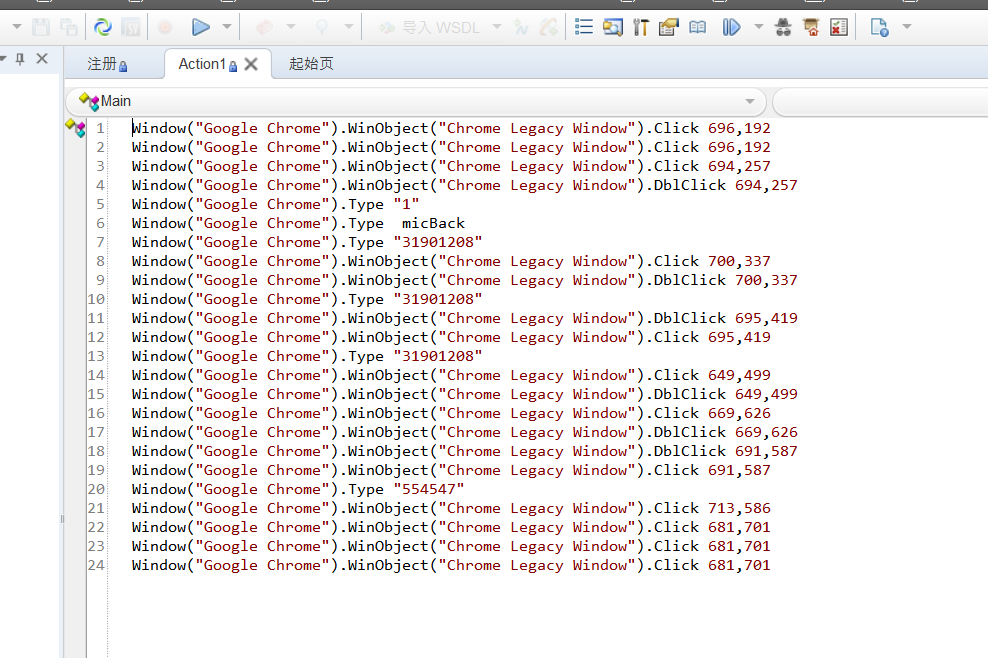
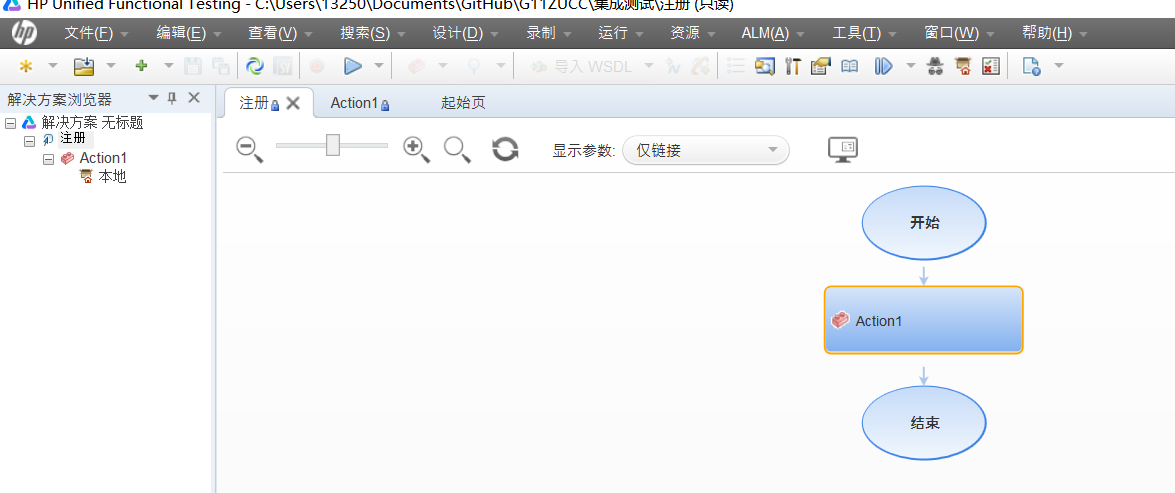


**管理员操作**



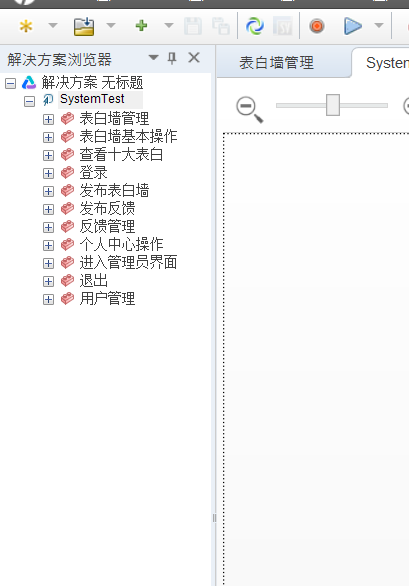


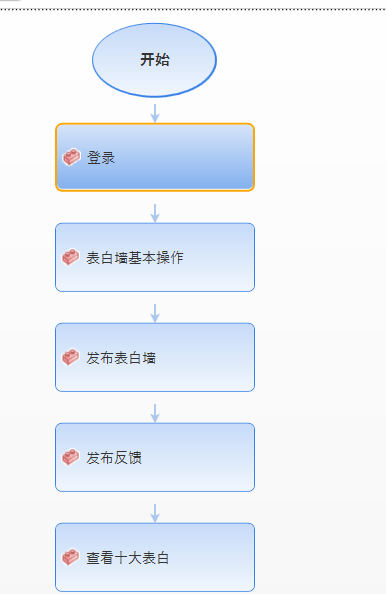
**注册**

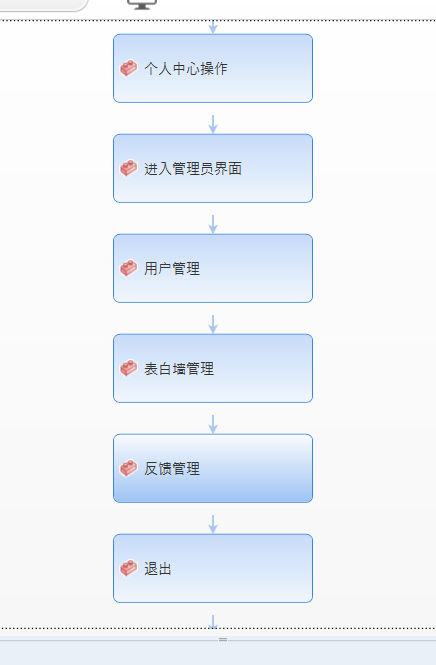


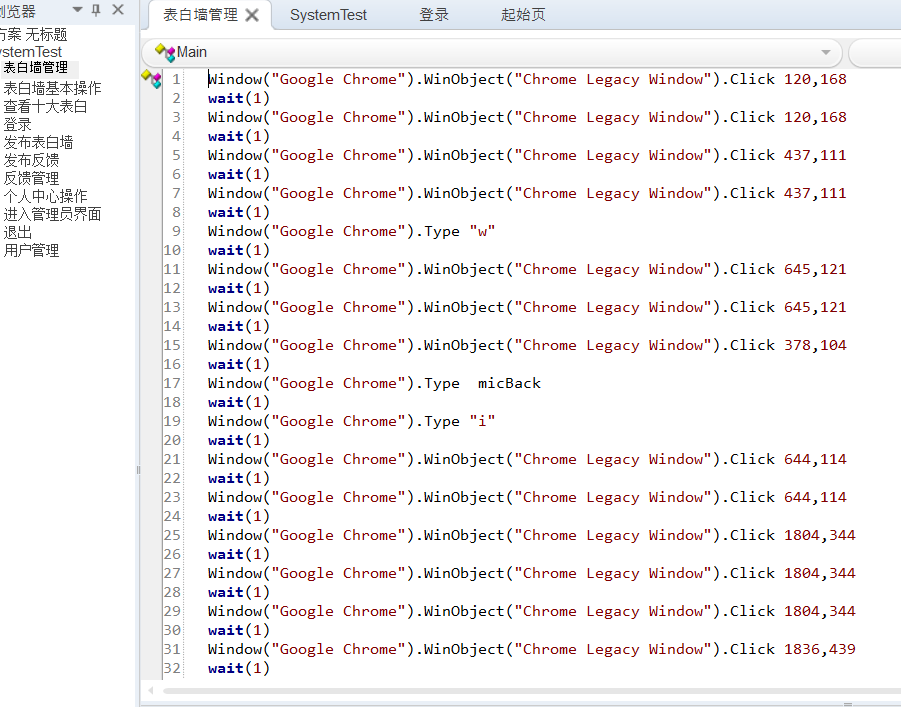
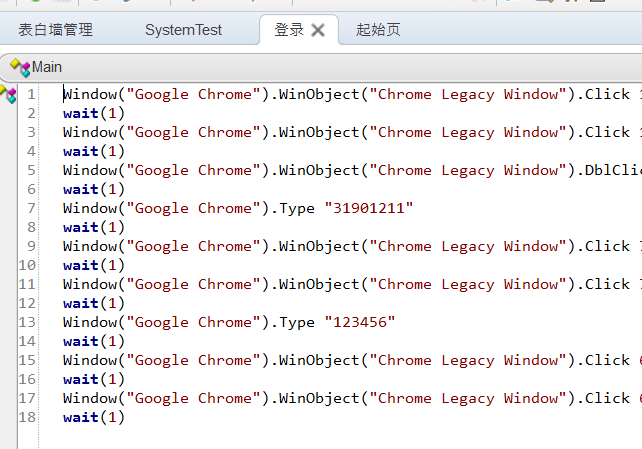
4.5 系统测试

系统测试采用HP Unified Functional Testing工具进行测试，下面是系统测试截图，详细内容见HP Unified Functional Testing工具。







5 测试进度表

已完成测试

6 需求的可追踪性

本计划的需求可以追踪，有可追踪性

7.评价

7.1评价准则

校园表白墙的主要功能能够实现并且正常运行。

7.2数据处理

使用MySql处理数据

7.3结论

本项目的测试阶段能够正常进行并得到一个相对完整的系统

8注解

附录