

从业刷（金融业考学习网站）测试报告

杨帆 陈鹏宇 黄鑫豪 齐家乐

1. 简介

1.1. 编写目的

本测试报告为“从业刷”的测试报告，本文档用于记录测试过程，总结测试情况，分析测试数据，描述系统是否符合需求，是否达到预期功能目标，并归纳测试工作进行过程中暴露的问题与遗留的风险，给出相应的测试建议。

1.2. 项目背景

1.2.1 项目名称:

从业刷（金融业考学习网站）

1.2.2 项目简介:

本项目以黄金、银行、证券、期货从业资格考试为主体，通过在线练习、及时纠错、复习资料、在线课堂等多种方式，全方位指导备考人群科学备考，合理学习，加强对中国金融市场的理解，增强从业人员的知识储备进而提高从业人员质量。

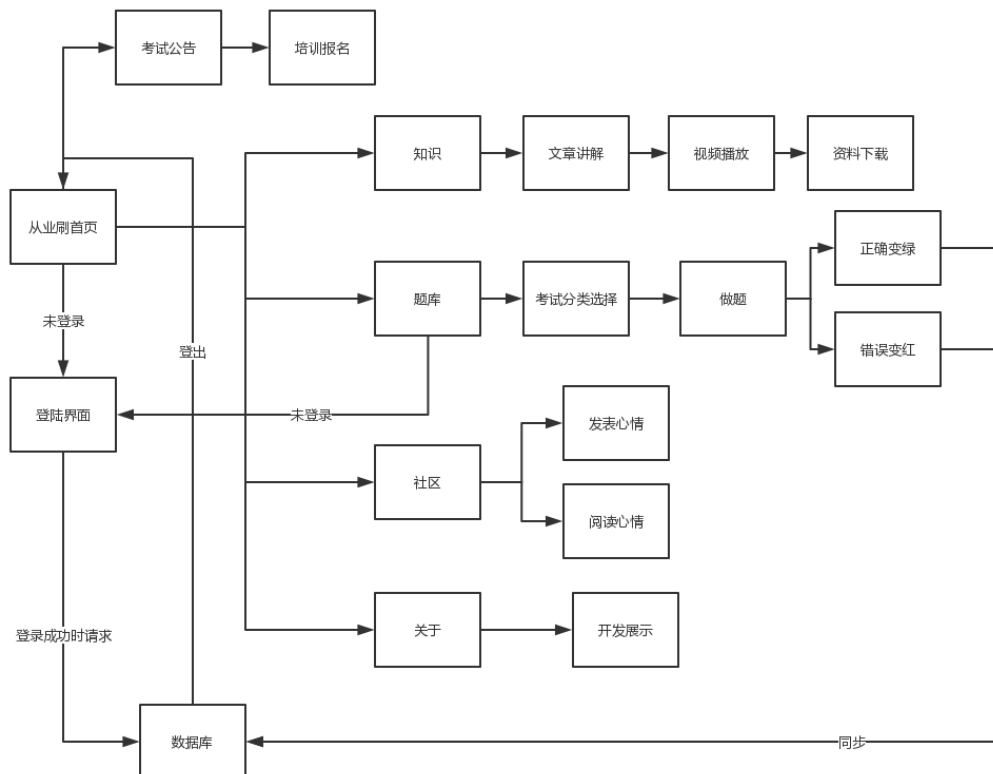
1.2.3 主要功能

基础知识学习（跳转外链）

习题练习

留言讨论

1.2.4 逻辑流程图



1.3. 参考资料

课堂第 8 讲相关 Slides

《软件工程：面对对象和传统的方法》第 6 章

2. 测试概要

2.1 测试环境及工具

测试操作系统：

Windows 10 教育版 64 位 (10.0 版本 17134)

测试浏览器：

Google Chrome 74.0.3729.169 (正式版本) (64 位)

Firefox 67.0 (64 位)

Edge 42.17134.1.10

Internet Explorer 11.765.17134.0

测试工具：

Selenium IDE 3.8.1

WireShark

2.2 测试范围

测试主要根据项目代码及相应文档进行系统测试，包括功能测试、安全性和访问控制测试、用户界面测试以及兼容性测试。

2.3 测试类型

测试类型	测试内容	测试目的	所用测试方法
功能测试	用户登录、注册信息、社区论坛、链接跳转、答题训练、错题回顾	核实所有功能均已正常实现，即可按用户的需求使用系统 1、业务流程检验：各个业务流程能够满足用户需求，用户使用不会产生疑问 2、数据准确：各数据输入输出时系统计算准确	采用黑盒测试，使用边界值测试、等价类划分等测试方法，测试包括手工测试和利用 Selenium 工具进行测试
用户界面测试	友好性、易用性、合理性、一致性、正确性	核实网站风格符合可接受标准，能够保证用户界面友好性、易操作性，符合用户操作习惯	手工测试
安全性和访问控制测试	注册、登录 应用层通信协议安全问题 通过修改 URL 非法访问	应用程序级别的安全性：核实用户只能操作其所拥有权限能操作的功能 统级别的安全性：	黑盒测试、手工测试

		核实只能具备系统访问权限的用户才能访问系统	
兼容性测试	用不同浏览器登录系统	核实系统在不同的软件中运行稳定	黑盒测试、手工测试

3. 测试结果

3.1 功能测试

3.1.1 测试单元：注册模块

输入等价类	测试用例	期望结果	实际结果
无效等价类	用户名：空 密码：非空	注册不成功	弹出出错信息
	用户名：非空 密码：空	注册不成功	弹出出错信息
	用户名：空 密码：空	注册不成功	弹出出错信息
	用户名：已存在 密码：非空	注册不成功	弹出出错信息
有效等价类	用户名：正确 密码：正确	注册成功	进入登录页

3.1.2 测试单元：登录模块

输入等价类	测试用例	期望结果	实际结果
无效等价类	用户名：空 密码：非空	登录不成功	弹出出错信息
	用户名：非空 密码：空	登录不成功	弹出出错信息
	用户名：空 密码：空	登录不成功	弹出出错信息
	用户名：正确 密码：不正确	登录不成功	弹出出错信息
	用户名：不正确 密码：正确	登录不成功	弹出出错信息
有效等价类	用户名：正确 密码：正确	登录成功	进入主页

3.1.3 测试单元：做题模块

输入等价类	测试用例	期望结果	实际结果
无效等价类	选择错误选项	提示错误	背景变红
有效等价类	选择正确选项	提示正确	背景变绿

3.1.4 测试单元：留言模块

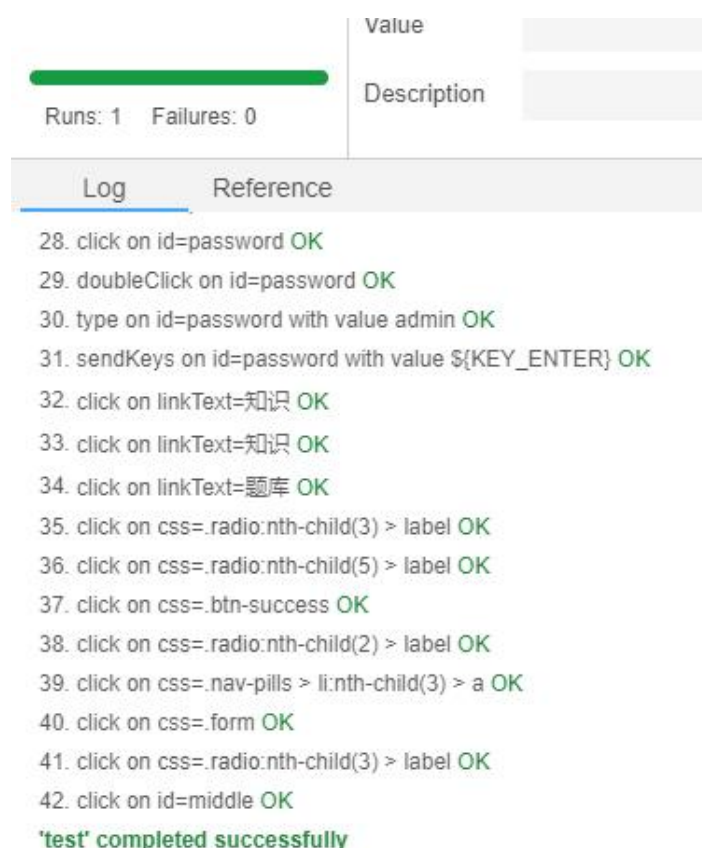
输入等价类	测试用例	期望结果	实际结果
无效等价类	未登陆	不能留言	没有留言栏
有效等价类	已登陆	可以成功留言	留言成功

3.1.5 测试单元：基础功能

顶部导航栏页面切换	正常
社区留言	正常
考点速递、在线课程（超链接功能）	正常
下载资料	正常
切换题目	正常
切换考试科目	正常
退出登陆状态	正常

3.1.6 使用 Selenium 进行功能测试

为减少手工测试中过多的重复操作所耗时间，在测试中也使用了 Selenium IDE 工具完成了部分重复性工作。页面截图如下：



3.2 兼容性测试

在 IE11, EDGE42, Chrome74, Firefox67 四款浏览器上进行全部测试，未发现兼容性问题，一般情况下显示效果良好。

发现问题：

题库页在浏览器较窄时显示不美观，组件位置出现偏移。

3.3 用户界面测试

发现问题:

未实现用户信息页，在社区页面点击用户名后会跳转回网站首页。

不能记录做题历史数据。

期货从业页无法点击右边题号切换。

显示题目总数不对。

在知识页的子栏目下方切换页面会报错。

3.4 安全性测试

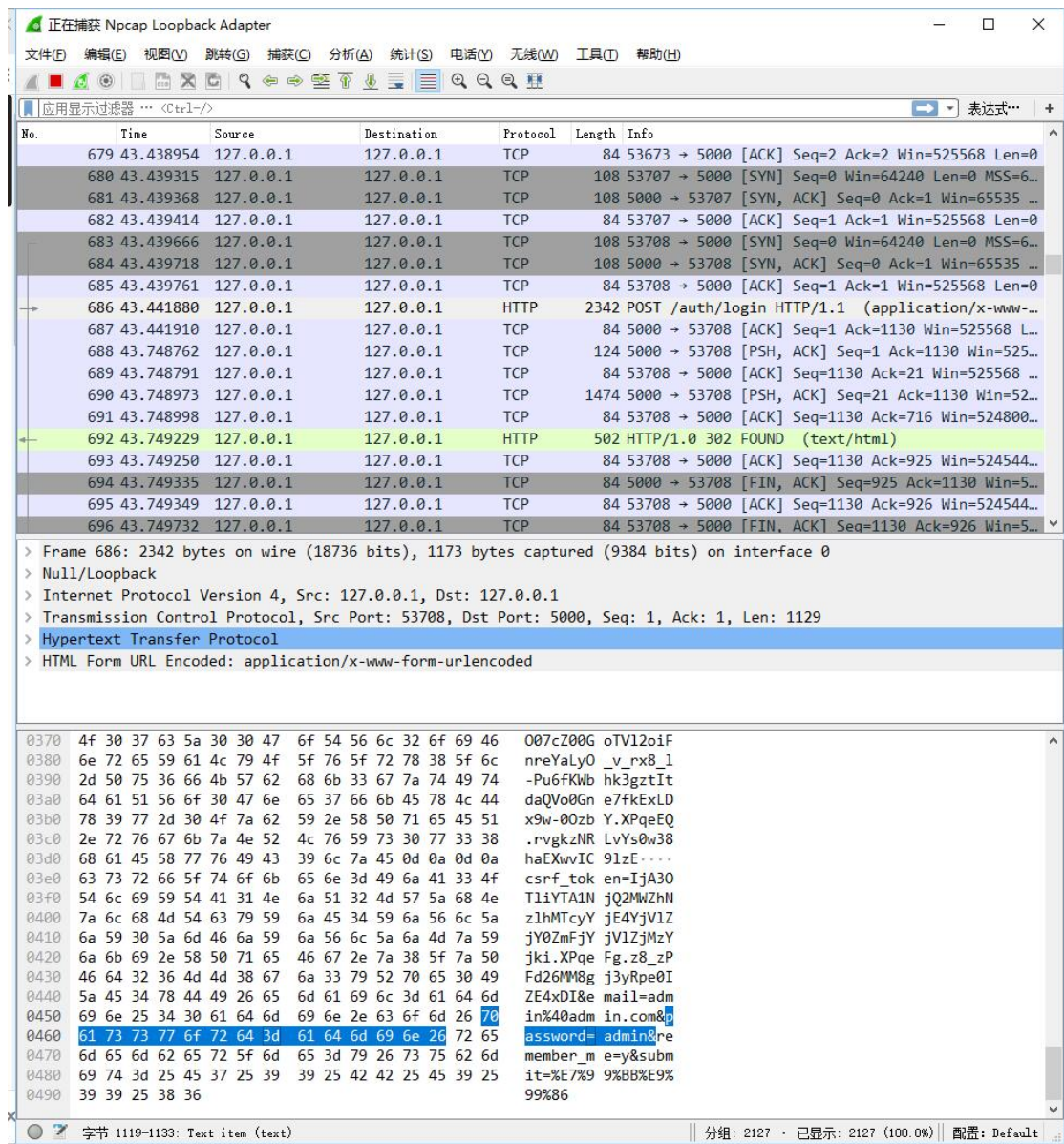
注册、登录功能在逻辑上安全。

修改为错误 URL 时，系统会报 “Internal Server Error”，无法非法访问。安全。

发现问题:

不真正检验注册者是否拥有该邮箱，而直接以邮箱为帐号登录。

网站采用 http 协议而非 https，帐号密码以明文传输，可以通过 wireshark 等工具窃听到密码，易被网络攻击者攻击。(见下图，可看到帐号为 admin@admin.com 密码为 admin)



3.5 白盒测试

由于本项目为 Web 项目，主要代码中分支语句并不多，且功能相互独立，在本次白盒测试的过程中，确保了所有独立路径至少被使用一次，覆盖了软件所有的逻辑路径。

RegistrationForm() 函数内：

```
def validate_email(self, field):  
    if User.query.filter_by(email=field.data).first():  
        raise ValidationError('该邮箱已经被注册过了')  
  
def validate_username(self, field):  
    if User.query.filter_by(username=field.data).first():  
        raise ValidationError('这个用户名已经被使用了')
```

测试过程：分别使用已用过的邮箱，用户名和未用过的邮箱用户名尝试注册，前者失败，后者成功，通过测试用例。

Login() 函数中：

```
if form.validate_on_submit():  
    user = User.query.filter_by(email=form.email.data).first()  
    if user is not None and  
user.verify_password(form.password.data):  
        login_user(user, form.remember_me.data)  
        return redirect(request.args.get('next') or  
url_for('index'))  
    flash('Invalid username or password.')
```

测试过程：分别使用空密码，错误密码，正确密码尝试登陆，前两者失败，后者成功，通过测试用例。

Register() 函数中：

```
if form.validate_on_submit():  
    user = User(email=form.email.data,  
                username=form.username.data,  
                password=form.password.data)  
    db.session.add(user)  
    db.session.commit()  
    flash('You can now login.')
```

```
return redirect(url_for('login'))  
return render_template('auth/register.html', form=form,  
current_time=datetime.utcnow())
```

测试过程：输入满足要求则注册成功，跳转至登陆页面，输入不满足要求则继续留在注册页，通过测试用例。

3.6 缺陷统计与分析

测试类型	严重影响系统运行的错误	功能缺陷,影响系统运行	不影响但应该修改	合理的建议	总数
功能测试	0	0	0	0	0
安全性测试	0	1	1	0	2
兼容性测试	0	0	1	0	1
用户页面测试	0	0	3	2	5

4. 结论

4.1 测试结论

“从业刷”网站虽然仍存在部分问题，但已达到交付标准，系统功能较为完善，运行较为稳定，系统鲁棒性较优，符合设计要求，该网站能够满足用户的需求，可以上线使用。

4.2 呈现的问题

呈现的问题主要集中在安全性上，特别是使用 http 而非 https 的问题，但在将本系统部署于本机而非服务器端的情况下，不会有严重的问题。

页面一些使用较少的功能存在部分小问题，不影响主要功能的使用，后续开发过程中可予以修改。