

【在线银行系统】

——外汇子系统

测试 报告

组长：李浩宇

组员：李乐凡 林然 蔡佳伟

日期：2024/6/13

版本：Version 3.0

目录

目录

1.	引言.....	3
1.1.	编写目的.....	3
2.	模块功能测试.....	6
2.1.	模块说明.....	6
2.2.	外汇系统功能测试.....	7
3.	边界值与基路径测试.....	21
3.1	边界测试.....	21
3.2	基路径测试.....	25
4.	压力测试.....	26
4.1	测试简介.....	26
4.2.	控制.....	26
4.3.	输入.....	27
4.4.	测试结果.....	27
5.	其他模块接口测试.....	29
5.1	测试简介.....	29
5.2	测试对象说明.....	29
5.3	测试内容.....	29
6.	安全性测试.....	34
6.1.	URL攻击.....	34
6.2.	SQL注入攻击.....	34
6.3.	XSS跨站脚本攻击.....	35
7.	对外汇子系统功能的结论.....	37
7.1	用户模块.....	37
7.2	数据操作员模块.....	37
7.3	系统管理员模块.....	38
8.	分析摘要.....	38
8.1.	能力.....	38

1. 引言

1.1. 编写目的

本测试报告为在线银行系统-外汇子系统的测试报告，目的在于总结测试阶段出现的问题，检测系统是否达到预期功能目标。

本报告预期参考人员包括软件客户、项目经理、项目开发人员、软件质量分析人员、系统维护人员等。

1.2. 项目背景

在当今数字化时代，互联网与移动通信技术的飞速发展彻底改变了人们的生活方式和商业运作模式。金融服务行业，尤其是银行业，面临着前所未有的变革机遇与挑战。随着智能手机和平板电脑的普及，移动支付逐渐成为国人日常生活中不可或缺的一环，手机银行也成为了移动支付重要组成部分，消费者逐渐舍弃了前往传统的银行柜台进行银行操作，越来越倾向于使用移动设备或个人PC进行日常银行操作，如转账、账单支付、账户查询、投资理财等。客户期待随时随地获取金融服务，要求银行提供无缝、便捷的在线解决方案；云计算、大数据、人工智能、区块链等先进技术为在线银行系统提供了强大的技术支撑。这些技术能够提升系统处理能力，实现数据的深度分析，提供个性化服务，同时增强安全防护，减少欺诈风险。

研究现状：

中国银行app是中国银行开发的手机软件。2022年12月16日，中国银行正式发布手机银行8.0版。中国银行面向新市民、县域、养老、代发薪、少数民族、交通出行六大客群，打造专属服务体验。

国内各大银行的网页在线银行系统，旨在为用户提供安全便捷的银行服务。

1.3. 术语与缩写解释

缩写、术语及符号	解释
SOA 架构	面向服务的体系结构。
元数据 Metadata	描述数据的内容、质量、状况和其他有关特征的数据。
数据中心 Data Center	以各类数据为核心，依托成熟的存储、数据库、GIS、网络等技术，按照统一标准，建立的具有信息管理、分析、查询、统计及服务的一体化数据管理体系。

数据管理 Data Management	利用数据库、数据仓库、元数据和网络等技术，建立分布式、集中式或集中加分布式数据管理系统，开展数据接收、组织存储、运行维护、更新、共享交换等工作，实现对数据资源的有效组织和应用。
顺序图	顺序图也称为时序图，它描述了系统中，对象间通过消息 进行的交互，它强调了消息在时间轴上的先后顺序。常用 来描述用例的实现，它表明了由哪些对象，通过消息相互 协作来实现用例的功能，并标识了消息发生交互的先后顺 序。
状态图	状态图描绘一个系统或组件可能假设的状态，并且显示引起 或导致一个状态切换到另一个状态的事件或环境。
类图	类图描述了一个软件系统中的类、接口、关系以及它们之间 的静态结构。类图用于表示系统中的对象、类以及它们之间 的关系，它提供了一种直观、可视化的方式来表示系统的结 构
IPO图	IPO 是指结构化设计中变换型结构的输入（Input）、加工（Processing）、输出（Output）。IPO 图是对每个模块进行详细设计的工具，它是输入加工输出（INPUT PROCESS OUTPUT)图的简称。在系统的模块结构图形成过程中，产生了大量的模块，在进行详细设计时开发者应为每一个模块写一份说明，IPO 图就是用来说明每个模块的输入、输出数据和数据加工的重要工具。

1.4. 系统概述

在线银行系统是一个在线操作系统。我们需要实现一个银行系统，完成正常的网上银行所涵盖的所有功能，使得该软件能够正常使用并具有一定的可延展性。共有五个备选模块：1. 柜台操作子模块 2. 互联网个人银行子模块 3. 贷款子模块 4. 信用卡子模块 5. 外汇子模块。

我们小组要实现的子模块是外汇子模块，功能目标如下：

外汇业务和汇率管理：系统允许客户查询当前汇率，进行外币的买入和卖出交易，提供外币搜索功能，并模拟汇率的每日动态变化。此外，系统还提供数据操作员和系统管理员的入口，支持外币和汇率的调整，以及数据操作员账号的添加、修改、删除和权限管理。

交易历史显示：提供用户界面展示客户的交易历史记录，方便客户查看和管理自己的外汇交易信息。

系统管理和权限控制：支持系统管理员进行数据操作员账号的管理，包括添加、修

改、删除操作员账号，并管理数据操作员的权限，以确保系统的安全和数据的完整性。

具体功能介绍如下：

- i. 允许通过互联网进行外汇业务。允许客户查询当前汇率，买入和卖出外币。
- ii. 提供外币搜索功能
- iii. 提供界面显示客户的交易历史。
- iv. 需要简单模拟汇率的每日动态变化情况
- v. 允许数据操作员对外币和汇率进行调整，提供系统管理员入口，支持添加、修改、删除数据操作员账号，管理数据操作员权限。

1.5. 测试对象——外汇子系统说明

该项目的用户是在线银行系统的外汇兑换人员，包括用户，数据操作员，系统管理员。在线银行系统的外汇子系统中，用户可以进行外币兑换，搜索外币，查询外币汇率，查询交易历史等操作，数据操作员可以进行增添外币，修改外币汇率，删除外币等操作，系统管理员可以管理数据操作员权限，增减数据操作员等。

因此，本软件客户端的界面应尽量美观、直观、易懂，操作应尽量简单、容易上手、性能应尽量实时、准确。

我们外汇子系统模块实现了相应功能，并实现了与整个大组功能的集成。

1.6. 测试内容

测试名称	目的	内容
模块功能测试	检测各个模块的功能是否全部实现	根据《总体设计说明书》的要求针对各个模块进行功能测试，尽可能保证测试项覆盖所有功能和各种功能条件组合。
边界值测试	检测系统能否正常处理边界值	在一些存在边界值问题的数据里分别输入边界值，观察系统反应，检测系统的应对能力。
压力测试	测试系统的承受能力	对子系统进行超过规定性能指标的测试，包括系统能够在压力过程中避免明显的性能下降，以及在压力后的及时恢复。

模块接口测试	测试与其他模块的接口是否完好，能否最后实现集成测试	运行本子系统，观察数据库里的数据变化，检查子系统间的交互。
--------	---------------------------	-------------------------------

1.7. 测试设备

包括测试硬件设备和软件设备（浏览器、测试辅助工具）的说明

1.8. 测试进度安排

阶段	内容	时间
第一阶段 （预备阶段）	测试人员阅读本子系统的设计文档，熟悉各个功能模块所实现的具体功能，了解本子系统各个输入数据的边界情况。同时，寻找用于测试相关的工具。	6月2日-6月4日
第二阶段 （准备阶段）	测试人员根据各个功能模块的功能，编写测试用例，准备好测试所使用的输入数据。	6月5日-6月6日
第三阶段 （测试阶段）	测试人员针对已经开发出来的子系统，用测试用例对系统进行模块功能测试，找出本系统中存在的缺陷或错误。	6月7日-6月9日
第四阶段 （后期阶段）	本子系统编码人员根据测试阶段的结果调整程序代码，修复存在的缺陷或错误。	6月10日-6月12日

2. 模块功能测试

2.1. 模块说明

模块名称	外汇业务模块
模块简介	客户登录外汇系统，买入、卖出、查询汇率和查询交易历史

模块功能列表				
序号	一级功能		二级功能	
	功能名称	功能编号	功能名称	功能编号
1	客户登录外汇系统	Wb01	客户登录外汇系统	Wb0101
2	客户对外币进行操作	Wb02	客户买入外币	Wb0201
3	查询客户交易历史	Wb03	客户卖出外币	Wb0202
4	外币搜索功能	Wb04	客户查询当前外币汇率	Wb0203
5			客户查询交易历史	Wb0301
6			搜索外币	Wb0401

模块名称		外币、汇率变化功能模块		
模块简介		汇率每日自动变化和数据库操作员控制外币汇率		
模块功能列表				
序号	一级功能		二级功能	
	功能名称	功能编号	功能名称	功能编号
1	外币、汇率变化	Bh01	汇率每日自动变化	Bh0101
2	数据操作员登录	Bh02	数据操作员修改外币类型	Bh0201
3			数据操作员修改某外币汇率	Bh0202

模块名称		系统管理员模块		
模块简介		系统管理员控制数据操作员		
模块功能列表				
序号	一级功能		二级功能	
	功能名称	功能编号	功能名称	功能编号
1	系统管理员入口	Xt01	系统管理员登录	Xt0101
2	系统管理员控制数据操作员	Xt02	添加数据操作员账号	Xt0201
			修改数据操作员账号	Xt0202
			删除数据操作员账号	Xt0203
			调整数据操作员权限	Xt0204

2.2. 外汇系统功能测试

2.2.1 客户登录测试

2.2.1.1 测试用例

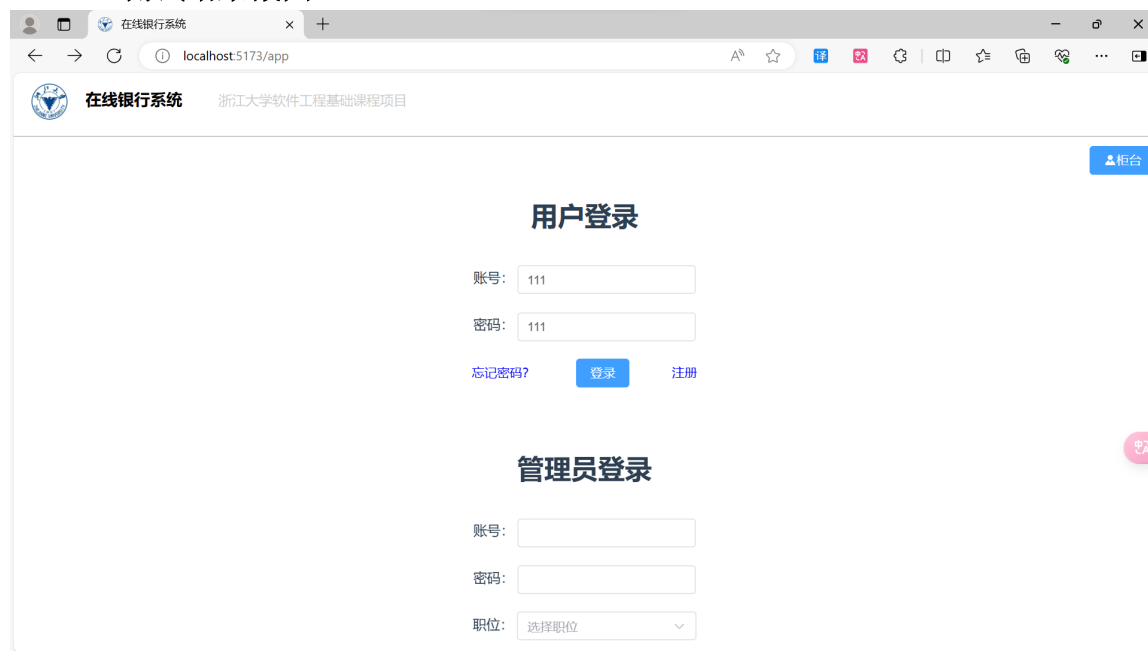
测试用例编号	Test221
测试功能	客户登录
测试标题	客户登录在线银行界面
重要级别	高
预置条件	系统内有该用户账号
输入	账号、密码
操作步骤	1. 进入主页面 2. 输入账号密码 3. 点击登录
预期输出	显示登录成功并进入主页/显示登录失败

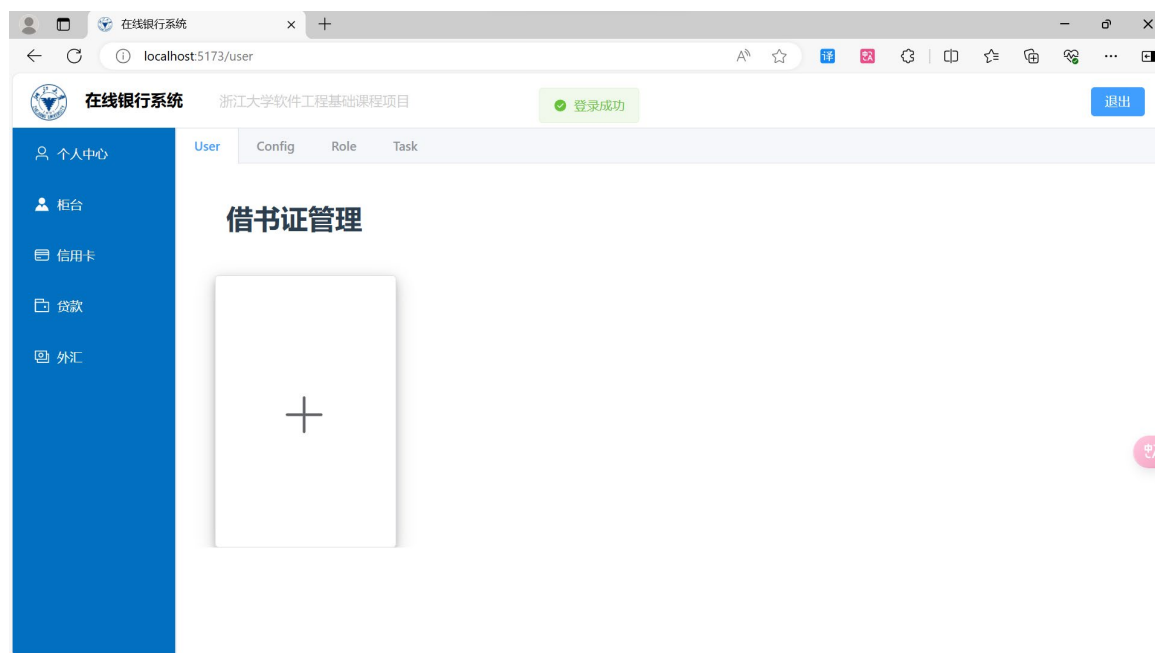
测试用例编号	实际输出	是否通过测试
Test2211	登录成功	是

2.2.1.2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.1.3测试结果截图





2.2.2 客户对外币进行操作测试

2.2.2.1 测试用例

测试用例编号	Test2221
测试功能	某银行卡买入外币
测试标题	该银行卡买入外币
重要级别	高
预置条件	客户已经登录账号
输入	买入的外币
操作步骤	1. 客户登录系统 2. 选择要买入的外币 3. 买入
预期输出	买入成功，该银行卡余额更新，交易记录更新/买入失败，提示余额不足

测试用例编号	实际输出	是否通过测试
Test2221	交易失败	是
Test2221——2	交易成功	是

2.2.2.2 测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.2.3 测试结果截图

交易外币

✕

外币名: Pound

汇率:8

交易类型:买入

交易量:20

+

-

银行卡

11

▼

交易外币

CNY

▼

取消

确认

✕ 交易失败

这里可以看到，交易失败是因为没有余额或是没有对应银行卡，交易成功以后，我们查询交易历史，发现了对应的交易，这里交易额是正数表示买入，负数表示卖出，可以查询对应时间段内的交易，也可以根据交易时间、额度等进行排序，代表我们实现了交易功能。

交易外币

✕

外币名: 英镑

汇率:11.084

交易类型:买入

交易量:10

+

-

银行卡

00000000000000000005

▼

交易外币

人民币

▼

111111

取消

确认

✓ 交易成功

首页

交易历史

汇率历史

银行卡:

00000000000000000005

起始时间: 2024 年 6 月 1 日

终止时间: 2024 年 6 月 22 日

查询

交易时间	货币1	交易额	货币2	交易额	卡号
20240614124413	人民币	17.1	该外币已删除	10	1232435
20240614125421	人民币	-71.36	该外币已删除	-10	1232435
20240614171859	人民币	0	英镑	0	1232435
20240614172415	人民币	110.84	英镑	10	1232435

浙江大学软件工程基础课程项目

退出

首页

交易历史

汇率历史

银行卡:

00000000000000000005

起始时间: 2024 年 6 月 1 日

终止时间: 2024 年 6 月 22 日

查询

交易时间	货币1	交易额	货币2	交易额	卡号
20240614125421	人民币	-71.36	该外币已删除	-10	1232435
20240614171859	人民币	0	英镑	0	1232435
20240614124413	人民币	17.1	该外币已删除	10	1232435
20240614172415	人民币	110.84	英镑	10	1232435

2.2.2.4 测试用例

测试用例编号	Test2222
测试功能	查询外币汇率
测试标题	某银行卡查询
重要级别	高
预置条件	客户已经登录账号
输入	需要查询的外币
操作步骤	1. 客户登录系统 2. 在搜索框内输入外币名 3. 查询
预期输出	弹出所查询外币的汇率

2.2.2.5测试结果分析

测试用例编号	实际输出	是否通过测试
Test2222	汇率	是

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.2.6测试结果截图

在线银行系统

浙江大学软件工程基础课程项目

个人中心

柜台

信用卡

贷款

外汇

首页

交易历史

汇率历史

Pound

查询

时间

外币名

当时汇率

20240607113153

Pound

8.5

2.2.3.1 测试用例

测试用例编号	Test2231
测试功能	查询某银行卡交易历史
测试标题	银行卡查询交易历史
重要级别	高
预置条件	客户已经登录账号
输入	银行卡id
操作步骤	1. 客户登录系统 2. 在搜索框内输入银行卡id和查询时间范围 3. 查询
预期输出	弹出所查询银行卡的交易历史

测试用例编号	实际输出	是否通过测试
Test2231	汇率	是

2.2.3
.2测

试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.3.3测试结果截图

浙江大学软件工程基础课程项目

退出

首页

交易历史

汇率历史

银行卡:

000000000000000000000000

起始时间: 2024年6月1日

终止时间: 2024年6月22日

查询

交易时间

货币1

交易额

货币2

交易额

卡号

20240614125421

人民币

-71.36

该外币已删除

-10

1232435

20240614171859

人民币

0

英镑

0

1232435

20240614124413

人民币

17.1

该外币已删除

10

1232435

20240614172415

人民币

110.84

英镑

10

1232435

首页 交易历史 汇率历史					
银行卡: 00000000000000000005					
起始时间: 2024 年 6 月 1 日					
终止时间: 2024 年 6 月 22 日					
查询					
交易时间	货币1	交易额	货币2	交易额	卡号
20240614124413	人民币	17.1	该外币已删除	10	1232435
20240614125421	人民币	-71.36	该外币已删除	-10	1232435
20240614171859	人民币	0	英镑	0	1232435
20240614172415	人民币	110.84	英镑	10	1232435

2.2.4.1 测试用例

测试用例编号	Test2241
测试功能	查询外币
测试标题	查询外币
重要级别	高
预置条件	客户已经登录账号
输入	外币名（部分也可以）
操作步骤	1. 客户登录系统 2. 在搜索框内输入外币名（部分） 3. 查询
预期输出	弹出所查询外币的信息

测试用例编号	实际输出	是否通过测试
Test224	外币信息	是

2.2.4.2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.4.3测试结果截图

首页 交易历史 汇率历史			
输入外币名			
外币名	汇率	涨幅	操作
Pound	8.5	0	买入 卖出
Dollar	7.3	0	买入 卖出
CNY	1	0	买入 卖出

首页

交易历史

汇率历史

Doll

外币名

汇率

涨幅

操作

Dollar

7.3

0

买入

卖出

首页

交易历史

汇率历史

olla

外币名

汇率

涨幅

操作

Dollar

7.3

0

买入

卖出

2.2.5.1 测试用例

测试用例编号	Test2251
测试功能	数据操作员修改外币汇率
测试标题	修改外币汇率
重要级别	高
预置条件	数据操作员登录且有修改外币汇率权限
输入	修改的汇率
操作步骤	1. 数据操作员登录系统 2. 在对应外币修改框内输入修改的汇率
预期输出	弹出修改后汇率和涨幅

测试用例编号	实际输出	是否通过测试
Test2251	汇率和涨幅	是

2.2.5.2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.5.3测试结果截图



2.2.5.4 测试用例

测试用例编号	Test2252
测试功能	数据操作员增加外币
测试标题	增加外币
重要级别	高
预置条件	数据操作员登录且有增加外币权限
输入	修改的汇率
操作步骤	1. 数据操作员登录系统 2. 在添加外币框内输入外币、汇率
预期输出	弹出新增的外币和汇率

测试用例编号	实际输出	是否通过测试
Test2252	外币和汇率	是

2.2.5.5 测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.5.6 测试结果截图



2. 2. 5. 7 测试用例

测试用例编号	Test2253
测试功能	自动更新外币汇率
测试标题	自动更新外币汇率
重要级别	高
预置条件	到自动更新汇率的时间且有外币
输入	修改的汇率
操作步骤	无需操作
预期输出	弹出修改后汇率和涨幅

测试用例编号	实际输出	是否通过测试
Test2253	汇率和涨幅	是

2. 2. 5. 8测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.5.9测试结果截图

如图所示，我们此时设置了每隔5秒更新一次外汇的汇率，实现了自动更新的功能

系统

浙江大学软件工程基础课程项目

首页

交易历史

汇率历史

输入外币名

外币名	汇率	涨幅
美元	5.805	0.051706308169598654
人民币	1	0

系统

浙江大学软件工程基础课程项目

首页

交易历史

汇率历史

输入外币名

外币名	汇率	涨幅
美元	5.831	0.4305890458146806
人民币	1	0

查询

时间	外币名	当时汇率
20240614100702	美元	7.024
20240614100707	美元	7.032
20240614100712	美元	7.079000000000001
20240614100717	美元	7.0729999999999995
20240614100722	美元	7.0760000000000005
20240614100727	美元	7.055
20240614100732	美元	7.03
20240614100737	美元	7.0120000000000005
20240614100742	美元	7.036
20240614100747	美元	7.037
20240614100752	美元	7.013
20240614100757	美元	7.017
20240614100802	美元	7.029000000000001
20240614100807	美元	7.007

2.2.5.10 测试用例

测试用例编号	Test2254
测试功能	对查询的外币排序
测试标题	查询外币

重要级别	高
预置条件	用户登录
输入	删除的外币
操作步骤	1. 用户登录系统 2. 对外币按不同指标进行排序
预期输出	弹出排序后的外币表

测试用例编号	实际输出	是否通过测试
Test2254	排序后外币界面	是

2. 2. 5. 11测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2. 2. 5. 12测试结果截图

输入外币名

外币名	汇率	涨幅	操作
CNY	1	0	买入 卖出
Dollar	7.3	0	买入 卖出
Pound	8.5	0	买入 卖出

外币名	汇率	涨幅	操作
Pound	8.5	0	买入 卖出
Dollar	7.3	0	买入 卖出
CNY	1	0	买入 卖出

外币名	汇率	涨幅	操作
CNY	2	100	买入 卖出
Dollar	7.3	0	买入 卖出
Pound	8	-5.88235294117647	买入 卖出

2. 2. 6. 1 测试用例

测试用例编号	Test2261
测试功能	添加数据操作员
测试标题	数据操作员添加
重要级别	高
预置条件	系统管理员登录
输入	添加的内容
操作步骤	1. 系统管理员登录系统 2. 添加信息
预期输出	弹出添加的数据操作员信息

测试用例编号	实际输出	是否通过测试
Test2261	数据操作员信息	是

2. 2. 6. 2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2. 2. 6. 3测试结果截图



新建成功

数据管理员列表

输入操作员账户

添加管理员

操作员序号	操作员账户	操作员权限
1	111	<input checked="" type="checkbox"/> 权限1 <input checked="" type="checkbox"/> 权限2 <input checked="" type="checkbox"/> 权限3

2.2.6.4 测试用例

测试用例编号	Test2262
测试功能	数据操作员权限调整
测试标题	数据操作员权限调整
重要级别	高
预置条件	系统管理员登录
输入	调整的权限
操作步骤	1. 系统管理员登录系统 2. 调整数据操作员权限 3. 确认
预期输出	弹出调整的数据操作员信息

测试用例编号	实际输出	是否通过测试
Test2262	数据操作员信息	是

2.2.6.5测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

2.2.6.6测试结果截图

输入操作员账户

添加管理员

确认更改

操作员序号	操作员账户	操作员权限
1	111	<input type="checkbox"/> 权限1 <input type="checkbox"/> 权限2 <input checked="" type="checkbox"/> 权限3

数据管理员列表

输入操作员账户

添加管理员

操作员序号	操作员账户	操作员权限
1	111	<input type="checkbox"/> 权限1 <input type="checkbox"/> 权限2 <input checked="" type="checkbox"/> 权限3

3. 边界值与基路径测试

3.1 边界测试

查询外币和历史，查询汇率，插入外币等操作是在线外汇子系统的重要组成部分，它们为用户提供了方便快捷的查询服务，需要具备高效、准确、可靠和安全的特点。这些功能的实现需要与多个数据源进行交互，因此系统的查询和插入操作功能应具有正确处理边界值输入的能力。

3.1.1.1 测试用例

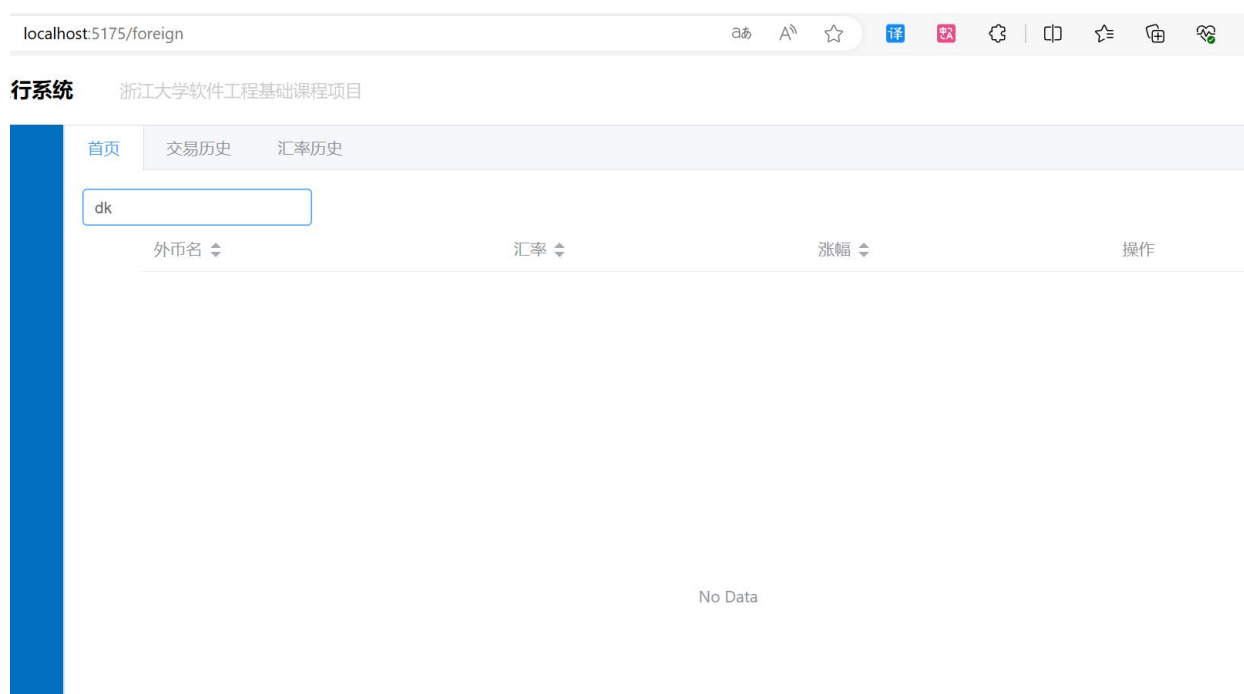
测试用例编号	Test3111
测试功能	用户外币查询
测试标题	用户查询不存在的外币
重要级别	高
预置条件	用户登录
输入	不存在的外币
操作步骤	1. 用户登录系统 2. 查询不存在的外币
预期输出	弹出没有东西

测试用例编号	实际输出	是否通过测试
Test3111	空	是

3.3.1.2 测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

3.3.1.3 测试结果截图



3.1.2.1 测试用例

测试用例编号	Test3121
测试功能	注册数据操作员
测试标题	系统管理员注册数据操作员少账号/少密码/都少
重要级别	高
预置条件	系统管理员登录
输入	信息
操作步骤	1. 系统管理员登录系统 2. 少输入数据操作员信息 3. 注册数据操作员
预期输出	注册失败

测试用例编号	实际输出	是否通过测试
Test3121	注册失败	是

3.1.2.2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

3.1.2.3测试结果截图

The screenshot displays a '新建管理员' (New Administrator) dialog box. The dialog has a title bar with a close button (X). Inside, there is a text input field containing '11', a password input field with the placeholder '输入操作员密码' (Enter operator password), and three checked checkboxes labeled '权限1' (Permission 1), '权限2' (Permission 2), and '权限3' (Permission 3). At the bottom right of the dialog are two buttons: '取消' (Cancel) and '确认' (Confirm). Below the dialog, a red error message box shows '× 新建失败' (New failed). Further down, there is a text input field containing '11111'. At the bottom of the page, there are two labels: '操作员账户' (Operator account) and '操作员权限' (Operator permissions).

新建管理员

11

输入操作员密码

☒权限1 ☒权限2 ☒权限3

取消 确认

× 新建失败

11111

操作员账户 操作员权限

新建管理员

输入操作员账户

输入操作员密码

☒权限1 ☒权限2 ☒权限3

取消 确认

✖ 新建失败

3.1.2.4 测试用例

测试用例编号	Test3122
测试功能	注册数据操作员
测试标题	系统管理员注册相同账户数据操作员
重要级别	高
预置条件	系统管理员登录
输入	信息
操作步骤	1. 系统管理员登录系统 2. 输入相同账户信息数据操作员 3. 注册数据操作员
预期输出	注册失败

测试用例编号	实际输出	是否通过测试
Test3122	注册失败	是

3.1.2.5测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

3.1.2.6测试结果截图

✖ 新建失败

3.1.3.1 测试用例

测试用例编号	Test3131
测试功能	数据操作员添加外币
测试标题	插入同名外币
重要级别	高
预置条件	数据操作员登录
输入	插入同名外币
操作步骤	1. 数据操作员登录系统 2. 数据操作员添加相同外币
预期输出	添加失败

测试用例编号	实际输出	24	是否通过测试
--------	------	----	--------

Test3131	失败添加	是
----------	------	---

3.1.3.2测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

3.1.3.3测试结果截图



浙江大学软件工程基础课程项目				✖ 新建失败
输入外币名				
添加外币				
外币名	汇率	涨幅	操作	
Pound	8	-5.88235294117647	更改汇率	删除
Dollar	7.3	0	更改汇率	删除
CNY	2	100	更改汇率	删除

3.2 基路径测试

外币查询功能是在线外汇子系统的重要组成部分，它们为用户提供了方便快捷的查询通道，需要具备高效、准确、可靠和安全的特点。因此，对查询功能的基路径测试是在线外汇子系统能够正常为用户提供服务的重要保障。

3.2.1.1 测试用例

测试用例编号	Test3211
--------	----------

测试功能	用户外币查询
测试标题	用户查询不输入
重要级别	高
预置条件	用户登录
输入	所有外币
操作步骤	1. 用户登录系统 2. 查询框内不输入外币
预期输出	弹出所有外币

测试用例编号	实际输出	是否通过测试
Test3211	所有外币	是

3.2.1.2 测试结果分析

功能测试点符合测试结果，由此可以推断功能可以正常使用，该模块功能已基本实现。

3.2.1.3 测试结果截图

首页

交易历史

汇率历史

輸入外币名

外币名	汇率	涨幅	操作
Pound	8.5	0	<div>买入</div> <div>卖出</div>
Dollar	7.3	0	<div>买入</div> <div>卖出</div>
CNY	1	0	<div>买入</div> <div>卖出</div>

4. 压力测试

4.1 测试简介

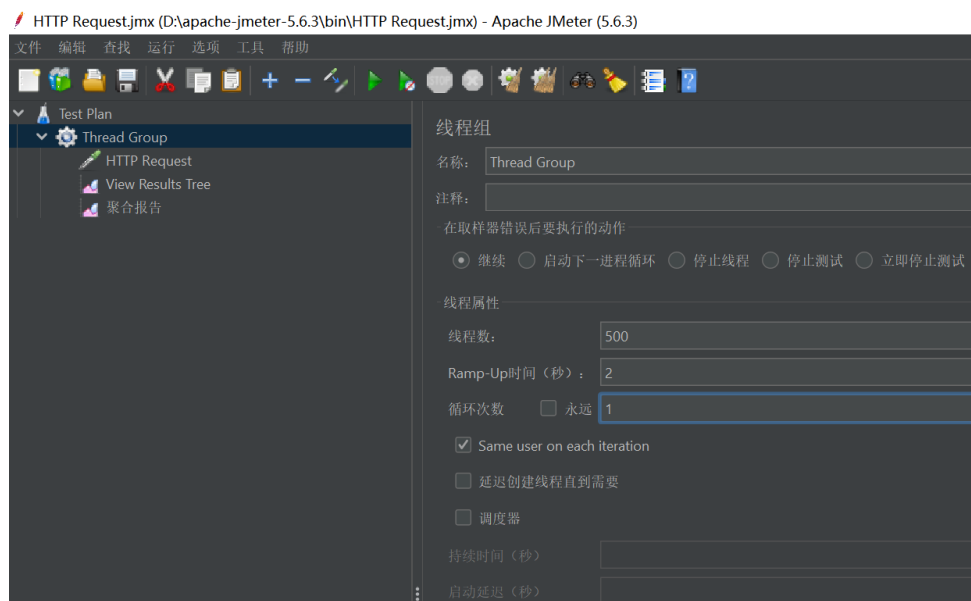
不同于功能测试，压力测试的重点不是正确性，而是系统的执行效率。本次压力测试的目的是测试在线银行系统外汇子系统的承载能力，主要包括系统对大量出错信息的处理；大规模用户同时发送请求时系统的最大负载能力、响应时间；系统对读取大量数据的响应等。

4.2. 控制

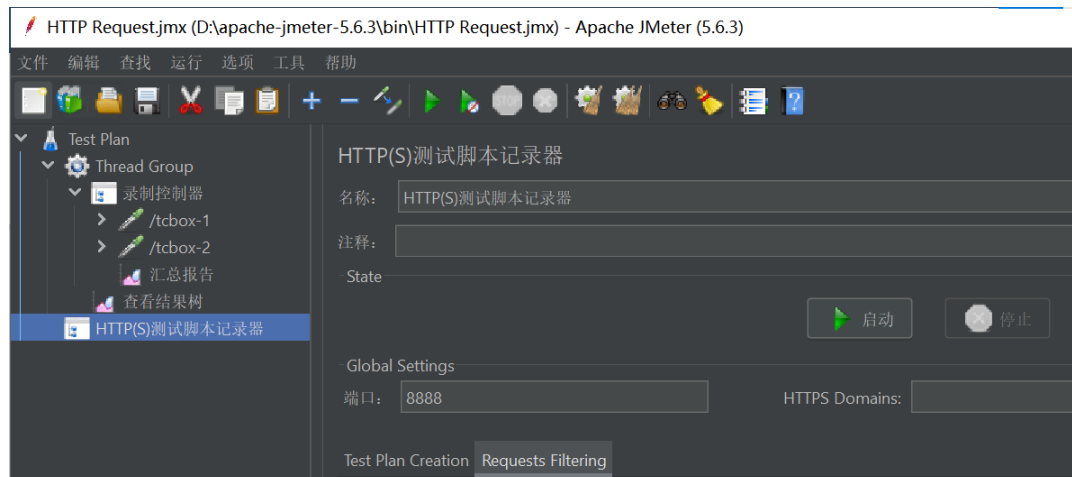
本次压力测试主要用到了一些自动化测试工具及对应的辅助工具，包括测试工具 Apache Jmeter 5.6.3。

4.3. 输入

①访问压力测试：选择HTTP Request压力测试，设置线程从500~3000不等，设置爬坡时间为2秒，设置循环次数为1。



②查询压力测试：采用HTTP(S)测试脚本记录器记录查询行为，然后复现3。选择设置线程从500~1800不等，设置爬坡时间为5秒，设置循环次数为1。



4.4. 测试结果

4.4.1. 测试结果数据

①访问压力测试：

课程作业网页支持两秒内2000次的HTTP请求，但是在线程升为2200次/两秒时，网页开始出现响应异常，异常率为6.86%。

外汇子系统模块的网页承载能力上限大概为2000~2200个用户/两秒，满足了课程要

求中500个用户在两秒内正常并发访问的要求。

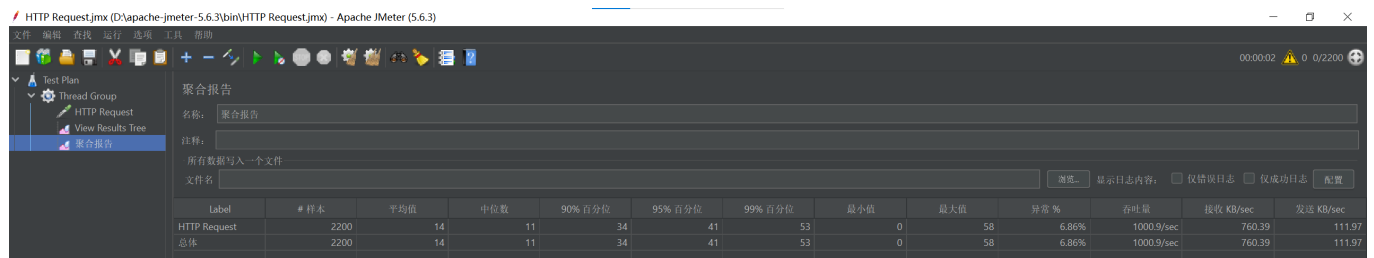
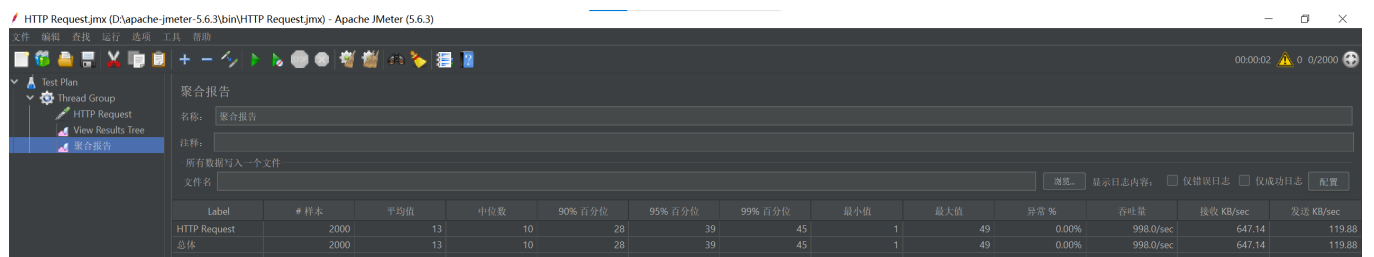
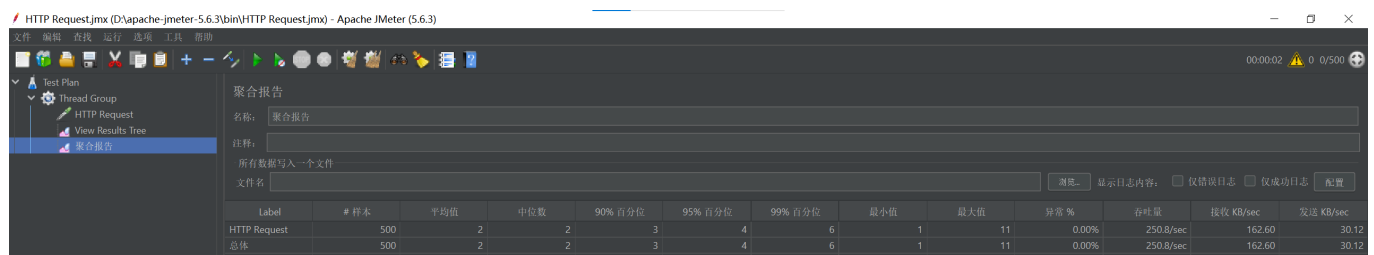
②访问压力测试：

课程作业网页支持五秒内1500次的HTTP请求，但是在线程升为1800~2000次/五秒时，实际运行时间会超过5秒（右上角感叹号左侧的时间）。

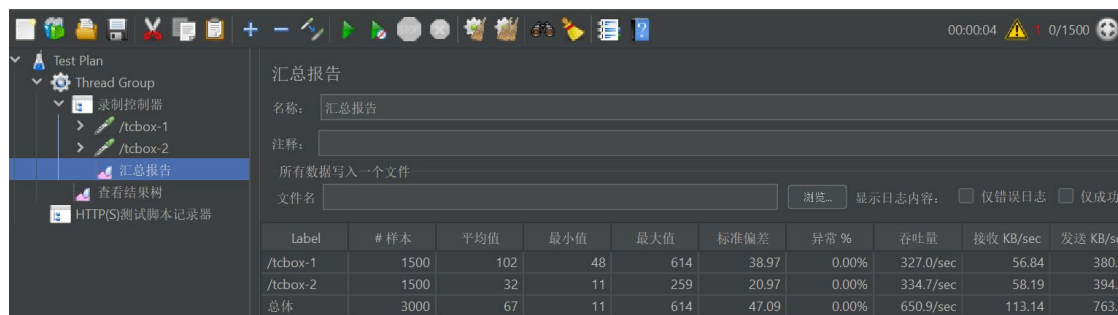
外汇子系统模块的网页承载能力上限大概为1500~1800个用户/五秒，满足了课程要求中500个用户在五秒内正常并发查询的要求。

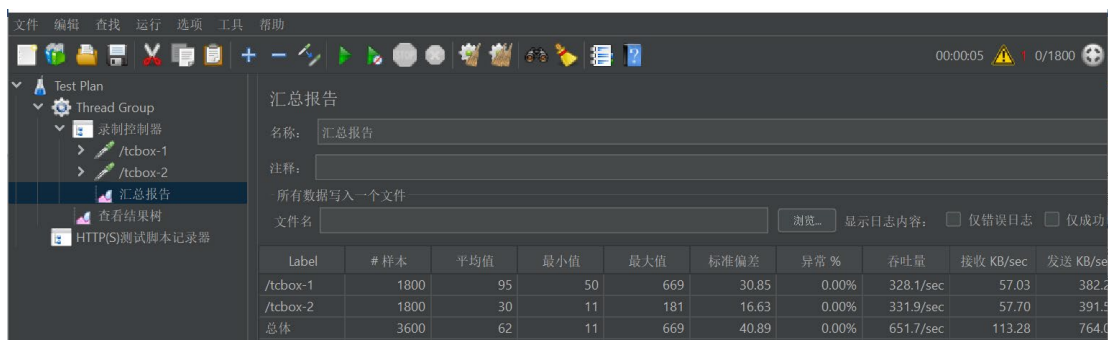
4.4.2. 测试数据截图

①访问压力测试：



② 查询压力测试：





The screenshot shows the JMeter Summary Report window. The left sidebar lists the test plan structure: Test Plan, Thread Group, /tcbox-1, /tcbox-2, Summary Report (selected), View Results Tree, and HTTP(S) Test Script Recorder. The main area displays the Summary Report for the selected Thread Group. It includes fields for Name (Summary Report), Comment, and a checkbox for 'All data written to a file'. Below this is a table with 10 columns: Label, # Samples, Average, Minimum, Maximum, Standard Deviation, Error %, Throughput, Received KB/sec, and Sent KB/sec. The table contains three rows: /tcbox-1, /tcbox-2, and the overall Total.

Label	# 样本	平均值	最小值	最大值	标准偏差	异常 %	吞吐量	接收 KB/sec	发送 KB/sec
/tcbox-1	1800	95	50	669	30.85	0.00%	328.1/sec	57.03	382.5
/tcbox-2	1800	30	11	181	16.63	0.00%	331.9/sec	57.70	391.5
总体	3600	62	11	669	40.89	0.00%	651.7/sec	113.28	764.0

4.4.3 测试结果分析

根据测试结果，本网站在 300 线程并发、10 轮不间断压力测试下，基本上承受住了绝大部分的访问请求，访问成功率很高。在多用户并发访问银行卡外汇子系统查询页面的场合，平均请求响应时间达到 5.501s，略微超出预期，但总体符合系统响应速度的要求。在多用户并发访问酒店 发布页面的场合，平均响应时间仅达 1.276s，说明系统在能够正确处理酒店发布请求的同时还能高效地完成发布任务。本次压力测试的结果基本符合预期，说明在线预订子系统能够满足软件的基本性能要求。

5. 其他模块接口测试

5.1 测试简介

本测试的目的主要是测试本外汇子系统与其他子系统的接口是否完好，能否最后实现集成测试。因为采用的是以数据为中心的体系结构风格，所以与其他子系统的接口测试最终主要集中在与数据库的接口测试上。项目具体信息存放在数据库中，前端通过后端接口从数据库中读取信息，以及从请求中获取登陆状态，实现与其他子系统的交互。本外汇子系统与其他子系统接口测试将主要测试与数据库的交互能否顺利进行以及用户登录信息能否正确从请求中获取。

5.2 测试对象说明

本测试的目的主要是测试本外汇子系统与其他子系统的接口是否完好，能否最后实现集成测试。因为采用的是以数据为中心的体系结构风格，所以与其他子系统的接口测试最终主要集中在与数据库的接口测试上。项目具体信息存放在数据库中，前端通过后端接口从数据库中读取信息，以及从请求中获取登陆状态，实现与其他子系统的交互。本外汇子系统与其他子系统接口测试将主要测试与数据库的交互能否顺利进行以及用户登录信息能否正确从请求中获取。

5.3 测试内容

5.3.1 数据操作员登录测试

在登录大组的银行管理系统后，使用系统管理员账号注册汇率数据操作员账号，成功登录。此外还提示密码错误，说明从数据库读取到了用户名数据。

The screenshot shows a web browser window with the URL `localhost:5173/adminlogin`. The page title is '在线银行系统' (Online Bank System) and the subtitle is '浙江大学软件工程基础课程项目' (Zhejiang University Software Engineering Basic Course Project). The left sidebar has two options: '用户登录' (User Login) and '管理员登录' (Admin Login). The main content area is titled '管理员登录' (Admin Login). The login form includes:

- 账号 (Account):
- 密码 (Password):
- 职位 (Position):
- 登录 (Login) button

A red error message '密码错误' (Password error) is displayed at the bottom of the page.

果程项目

✓ 登录成功

汇率 ◆

涨幅 ◆

5.3.2 数据操作员操作

登录上面注册的数据操作员账号后，增加外币和修改汇率，成功。

外币名	汇率	涨幅	操作
Meiyuan	8	0	<input type="button" value="更改汇率"/> <input type="button" value="删除"/>

localhost:5173/remitadmin

在线银行系统 浙江大学软件工程基础课程项目

系统管理员

输入外币名

添加外币

外币名

Meiyuan

汇率

涨幅

操作

更改汇率

删除

添加货币

货币名: ouyuan

汇率: 4

取消

确认

浙江大学软件工程基础课程项目

新建成功

外币名	汇率	涨幅	操作
Meiyuan	8	0	<input type="button" value="更改汇率"/> <input type="button" value="删除"/>
ouyuan	4	0	<input type="button" value="更改汇率"/> <input type="button" value="删除"/>

浙江大学软件工程基础课程项目

更改成功

外币名	汇率	涨幅	操作
Meiyuan	7	-12.5	<input type="button" value="更改汇率"/> <input type="button" value="删除"/>
ouyuan	4	0	<input type="button" value="更改汇率"/> <input type="button" value="删除"/>

5.3.3 数据操作员权限更改

登录整个大组的系统管理员账号后，修改数据操作员权限。成功。

更改成功

数据管理员列表

添加管理员

输入操作员账户

操作员序号	操作员账户	操作员权限
1	Caijiawei	<input type="checkbox"/> 添加 <input checked="" type="checkbox"/> 删除 <input checked="" type="checkbox"/> 修改
2	Liangtianshun	<input checked="" type="checkbox"/> 添加 <input checked="" type="checkbox"/> 删除 <input checked="" type="checkbox"/> 修改

5.3.4 用户操作

这里我们进行了用户的买入操作和卖出操作，我们选择的是前面的模块设置的银行卡，在用户登录以后，我们可以在银行卡界面选择他们已经设置好的银行卡。这里选用选择的方式是因为方便用户进行选择，避免记不住银行卡号的情况。

同时确保他们的银行卡里有余额，（没有余额或余额不足是无法买入的），在交易结束以后，会更新他们银行卡内的余额，说明我们与银行卡管理小组实施了很好的交互。

此外，我们还需要输入密码，我们对密码也进行了SHA-256加密同时加盐，是和我们大组的其他小组相同的操作，这确保了密码安全性的同时不会因为我们没有特殊处理密码造成买入的失败。

交易外币

✕

外币名: Pound

汇率:8

交易类型:买入

交易量:20

+

-

银行卡

11

交易外币

CNY

取消

确认

✕ 交易失败

这里可以看到，交易失败是因为没有余额或是没有对应银行卡，交易成功以后，我们查询交易历史，发现了对应的交易，这里交易额是正数表示买入，负数表示卖出，可

以查询对应时间段内的交易，也可以根据交易时间、额度等进行排序，代表我们实现了交易功能。

交易外币

外币名: 英镑

汇率:11.084

交易类型:买入

交易量:10

+

-

银行卡

00000000000000000005

交易外币

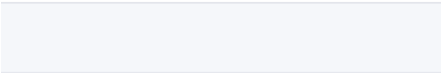
人民币

111111

取消

确认

交易成功



首页交易历史汇率历史

银行卡:

00000000000000000005

起始时间: 2024年6月1日

终止时间: 2024年6月22日

查询

交易时间	货币1	交易额	货币2	交易额	卡号
20240614124413	人民币	17.1	该外币已删除	10	1232435
20240614125421	人民币	-71.36	该外币已删除	-10	1232435
20240614171859	人民币	0	英镑	0	1232435
20240614172415	人民币	110.84	英镑	10	1232435

银行卡:

00000000000000000005

起始时间: 2024 年 6 月 1 日

终止时间: 2024 年 6 月 22 日

音詢

交易时间	货币1	交易额	货币2	交易额	卡号
20240614125421	人民币	-71.36	该外币已删除	-10	1232435
20240614171859	人民币	0	英镑	0	1232435
20240614124413	人民币	17.1	该外币已删除	10	1232435
20240614172415	人民币	110.84	英镑	10	1232435

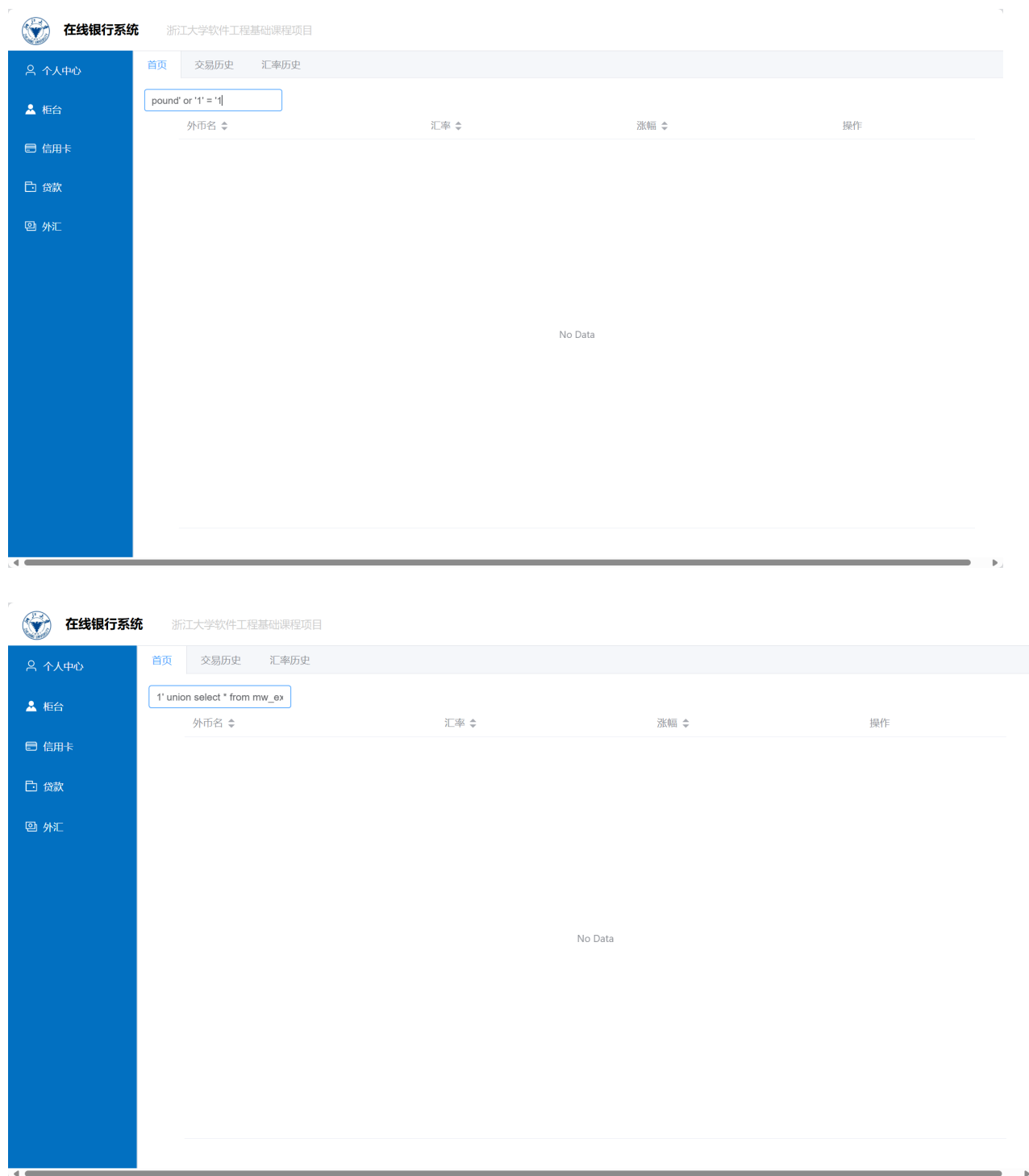
本系统的模块接口基本符合预期的测试结果，说明外汇子系统与其他子系统模块已经实现正常的交互，指定的功能得到基本的实现。这表明我们与其他系统的整合非常好，已经构成了一个相对完整的银行管理子系统，功能得到了顺利的实践。

6.1. URL攻击

外汇子系统为了保证数据传输的安全可靠性，在具体实现过程中极少使用URL来直接传递参数。本子系统仅在页面首次加载时需要通过URL进行数据的传递，其余情况均不涉及URL直接传递参数的情况。因此，不存在普通用户可以直接修改URL参数的情形。

此外，对于普通用户登陆情况下企图通过修改URL地址进入数据操作员或系统管理员界面的情况，系统将默认拒绝访问并返回原界面。

由于外汇子系统多处涉及用户的查询输入，在系统处理的过程均需要访问数据库获得相应的数据，因此避免SQL注入攻击是子系统正常、安全、可靠运行的基本需求。我们在该子系统的具体实现中考虑了SQL注入攻击的风险，并采取了一定的措施防止SQL注入攻击的发生。

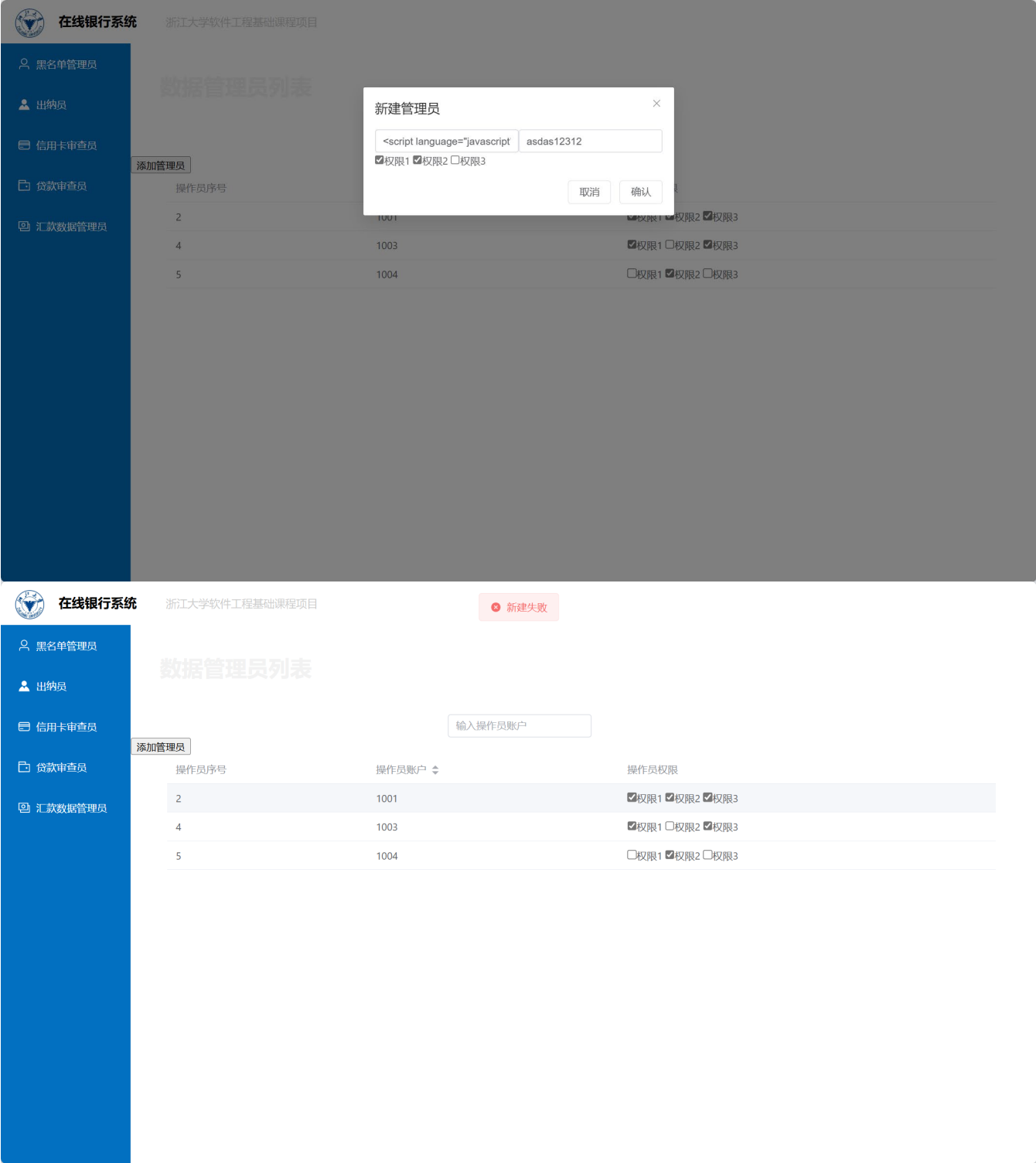


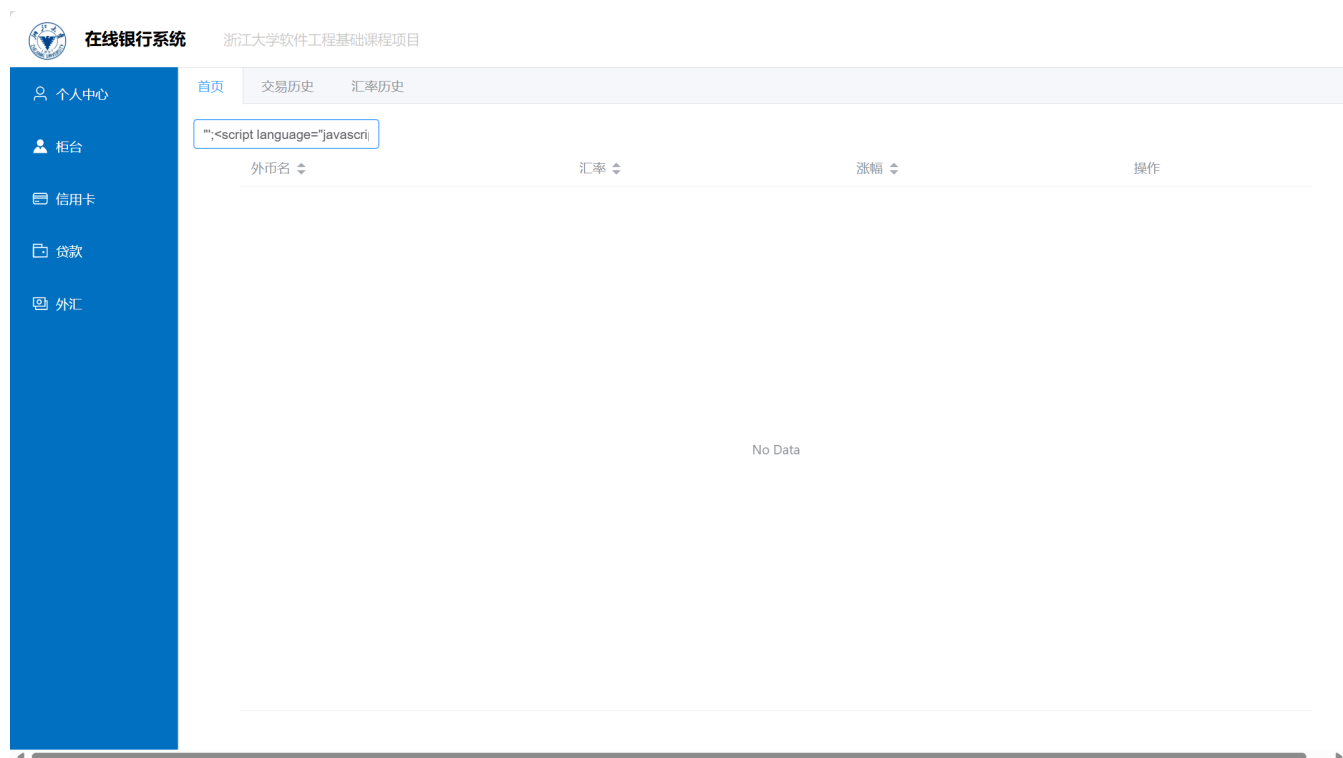
考虑到系统对数据库频繁的查找操作，安全隐患严重，因此我们采取类似 PreparedStatement 的机制操作 SQL 语句，从而达到有效防范 SQL 注入攻击的效果。同时，所有输入框会限制用户输入内容的长度，并过滤一些特殊字符，也可以起到防范 SQL 注入攻击的效果。

6.3. XSS 跨站脚本攻击

由于外汇子系统涉及用户查询外币交易历史，数据操作员添加、删除外币功能。因

此子系统对存储型 XSS 跨站脚本攻击做了相应的安全措施，以避免此类攻击的发生。例如，所有输入框会限制用户输入内容的长度，并过滤一些特殊字符，这样就可以起到防范XSS攻击的效果。





由结果可知，子系统将 JavaScript 脚本处理为普通文本，并没有执行脚本代码。说明子系统对 XSS 跨站脚本攻击有一定的抵抗能力。

7. 对外汇子系统功能的结论

7.1 用户模块

对于用户对在线银行系统外汇子模块的使用，我们在用户登录后提供给用户进行买入外币，查询交易历史，查询外币汇率等功能，让用户有一个愉快的使用体验。我们也没有忘记在买入外汇时和其他模块的交互，比如先检验银行卡的类型是信用卡还是储蓄卡，储蓄卡需要汇率换算后判断是否还有足够余额，购买后需要更新储蓄卡的余额。信用卡需要汇率换算后判断信用卡额度够不够，并且更新信用卡的欠款信息。这些与其他模块深度交互的部分，我们都完成得很好。

其中，在查询的交互方面，我们选择可以指定查询项目的排序顺序，在前端通过一个小箭头来指定，这符合用户的习惯，可以优化用户查询的体验。当查询条件为空时，系统将输出所有信息。这也使用户查询操作更加简单。

7.2 数据操作员模块

对于数据操作员的操作，我们给予了数据操作员不同的权限，这由系统管理员进行控制。这有效处理了数据操作员的越权操作，使得整个系统更加安全更加具有可持续性。此外，数据操作员还可以显式地看到汇率的变化，这也让数据操作员更清楚怎么更新汇

率。

7.3 系统管理员模块

在系统管理员的模块中，我们小组和别的小组是有很多部分功能的集成的，这无疑对我们的合作有很大的挑战，除了创建数据操作员，它还可以创建出纳员、信用卡管理员等等角色，但我们各个小组的合作十分完善，使得各个功能都实现的同时不会互相影响出bug。对于数据操作员的权限管理，我们也显式地采取了按钮的方式，只需要选择按钮，就可以更新权限。我觉得这也是很棒的设计。

8. 分析摘要

8.1. 能力

经过基本的模块功能测试、边界测试、基路径测试、压力测试、子系统模块接口测试后，本外汇子系统成功实现了所有功能需求，在线银行系统整体可以正常运行。作为在线银行系统的一个增值服务子系统，外汇子系统正确实现了xxx等功能。并且本子系统在安全方面也实现了一些措施以保证用户使用应用程序过程中的安全性，例如防止SQL注入、XSS跨站脚本攻击、URL攻击等。另外，本模块在压力测试中表现良好，基本达成了500人进行并发访问的要求。总而言之，外汇子系统根据需求总体说明书和设计总体说明书基本实现了所有功能需求，并且可供用户正常使用，并且在UI设计、用户交互、并发访问上表现良好。包括这篇测试报告，这是我们整个小组共同努力的结果，在此感谢所有小组成员的付出。

8.2. 缺陷和限制

由于软件工程基础大作业的时间紧张，我们在在线银行系统的外汇子系统的实现细节上仍然存在一系列问题和限制。例如网页总体的UI设计比较简陋，UI有待进行进一步美化工作。外汇子系统与用户之间的交互内容较为基本，虽然基本上满足了所有的功能需求，但是在与用户的交互细节方面仍停留在比较原始的阶段。在外汇的查询、增减外币相关操作，我们对用户的输入只进行了基本的检查，防止了一部分常见的安全攻击问题的发生，但实际上对用户的输入还需要做更加严格细致的语法检查。此外，外汇子系统后端数据库的数据表设计也比较简单，一些复杂的操作没有得到很好地实现。尽管如此，在基本的功能测试和压力测试下，我们小组的最终产品的表现还是十分优秀的，但仍有很大的提升空间。对整个银行系统整体上而言，在各个子系统进行交互和对接集成的过程中，由于时间紧迫，各个小组之间在项目开始初期缺乏沟通和统一的安排，这导致各个小组的进度难以统一，而且对于实现细节的把握上没有达成良好的一致性。

这导致在线银行系统的整体表现存在着一定的不足与缺陷。

8.3. 测试资源

在线支付系统在线预订子系统的测试由小组 4 人在若干台 PC 机上测试若干天后完成，由于时间仓促且经费有限，我们并没有能力购买统一的域名和服务器供在线支付系统运行。因此，小组成员在与其他小组成员的协商沟通中，最终选择在本地搭建前后端环境，并依靠校园局域网维持在线支付系统的正常运行并在一定程度上提供进行软件测试的平台。