
版次:

《软件工程基础》大作业

对《成绩管理系统》的开发文档（模拟）

信软学院软件工程课程组

2023 年 10 月

本文档的版本记录

本文件的版本变动情况如下：

版次	修改原因	修改内容	修改人	日期
1	完善系统信息	在测试环节增加对系统 BUG 的描述	成棋伟	2025.1.1
2	完善格式	对 4 部分报告格式进行修订，统一字体样式，增加了图表中的标题行，删除模板文字,增加了目录	成棋伟	2025.1.3

人员分工

学号	姓名	主要贡献	角色
2022130102013	成棋伟	需求分析报告编写; Junit 测试代码部署; 部分 BUG 查找; 统篇格式修订	组长
2022130101018	商江松	概要设计报告编写; 概要设计图表绘制	组员
2022130102001	刘冠显	详细设计报告编写; 详细设计图表绘制	组员
2022130102023	王沿喆	软件测试报告编写; 部分 BUG 查找	组员

目录

本文档的版本记录	II
人员分工	III
目录	IV
附录 A 软件需求分析报告文档模板	1
1. 引言	1
2. 综合描述	2
3. 外部接口需求	3
4. 系统用例设计	6
5. 系统功能需求	9
6. 其它非功能需求	15
7. 分析模型	15
附录 B 软件概要设计报告文档模板	28
1. 设计概述	28
2. 系统逻辑设计	29
3. 技术设计	33
4. 进度计划	35
附录 C 软件详细设计报告文档模板	36
1. 引言	36
2. 部件（或类、方法）详细设计	36
3. 界面详细设计	62
附录 D 软件测试报告文档模板	66
1. 引言	66

2. 单元测试	67
3. 集成测试	71
4. 验收测试	76
5 系统中存在的 BUG	84

附录 A 软件需求分析报告文档模板

1. 引言

本报告主要分析成绩管理系统（以下简称“系统”）的需求，包括系统的目标、功能、用户角色等部分。通过深入分析用户需求和场景，初步确定产品功能，确保系统设计能够满足所有相关方的需求，并为后续系统设计、开发及测试提供依据。

1.1 编写目的

本报告目的在于介绍成绩管理系统的基本情况，包括但不限于系统概述、功能介绍、接口设计以及需求建模等。通过相关分析，本报告将为系统开发提供明确的指导，确保最终产品能够满足特定用户的实际需求，达到高效管理教师、学生、管理员以及对应课程和成绩信息的目的，在其应用领域中发挥应有的作用。

1.2 软件预期用户

1.2.1 教师

教师可以通过本系统进行个人信息维护、课程信息管理（包括开设课程、添加课程学生）以及实现学生的成绩信息管理等。

1.2.2 学生

学生可以通过本系统进行个人信息维护、查询课程信息以及查询个人课程成绩等。

1.2.3 管理员

管理员可以通过本系统参与用户账号维护、管理学生信息以及管理教师信息等。

1.3 预期读者和阅读建议

- **用户：**建议优先阅读原型图的解释，熟悉系统界面；其次阅读系统用例以及系统功能需求，了解整个管理系统的基本功能及使用；
- **开发人员：**建议优先阅读原型图解释、外部接口以及用例描述，明确软件的功能需求，并参考非功能需求以及参考文献了解软件的开发技术；
- **项目经理：**建议优先阅读产品定位、主要功能概述以及系统功能需求等内容，以便了解系统整体情况和功能价值，从而管理整个系统的进度和资源分配；
- **测试人员：**建议阅读系统的用例描述以及功能需求部分，为后续的软件测试部分活动做铺垫；
- **文档编写人员：**建议对整个需求分析文档仔细阅读，在此基础上不断补充完善系统的需求分析报告；
- **其他利益相关人员：**建议选择性地阅读功能概述、外部接口需求、系统用例等部分，了解系统大致需求和设计的功能。

1.4 产品范围

1.4.1 产品概述

本成绩管理系统旨在为高校等教育机构提供一个全面、高效的平台，用于管理学生的课程成绩等相关数据。

1.4.2 产品功能

- **用户登录：**包括学生、教师和管理员的个人登录；
- **用户信息管理：**包括学生、教师和管理员信息的增加、删除以及修改；
- **课程信息管理：**教师可进行开课并登记课程信息，学生和教师可以查询课程

信息：

- **成绩信息管理：**包括成绩录入、查询和统计；
- **用户界面：**提供直观简易的用户界面，支持不同角色操作。

1.5 文档约定

- **正文风格：**宋体、五号
- **一级标题：**黑体、四号
- **二级标题：**黑体、小四
- **三级标题：**黑体、五号

1.6 术语与缩写解释

缩写、术语	解 释
GUI	Graphical User Interface，图形用户界面
GBK	汉字内码拓展规范

1.7 参考文献

- [1] 罗杰 S. 普莱斯曼等著，郑人杰，马素霞等译. 软件工程：实践者的研究方法[M]. 北京：机械工业出版社，2016. 12.
- [2] 侯捷. Word 排版艺术[M]. 北京：电子工业出版社，2004. 10.
- [3] 成绩管理系统说明文档-方峻版
- [4] 参考：软件项目报告样例文档-成电微记（修复）
- [5] 参考：健康码管理系统案例
- [6] 软件项目报告模板（2024 版）

2. 综合描述

2.1 产品的定位

本成绩管理系统是一个新型、自主型产品，聚焦于高校等教育机构对学生成绩、课程等信息的管理需求，目标用户主要有学生、教师以及教务系统管理员。整个系统涵盖了学生、教师、管理员的信息管理，同时能够对课程、成绩等信息进行管理和分析。系统最终预期是能够为教育机构提供一个良好、便捷的成绩系统管理平台。

2.2 产品的主要功能概述

- **基本信息管理**

对于本系统目标用户群体（学生、教师和管理员），主要功能涉及到用户登录、增加用户信息、修改用户信息以及删除用户信息，具体内容可见下表：

表 2-1 基本信息管理功能表

学生	个人信息查询	个人信息维护	
教师	个人信息查询	个人信息维护	
管理员	个人信息维护	增加用户条目	删除用户条目

● 课程信息管理

对于教师这一用户群体，主要涉及查询全部开课、开设新的课程；对于学生这一用户群体，则是查询个人课程，具体内容如下表所示：

表 2-2 课程信息管理功能表

学生	查询个人课程	
教师	查询全部开课	开设新的课程

● 成绩信息管理

对于已经开设的课程，本系统还支持对学生对应课程成绩的信息管理。比如教师可以登记成绩信息、统计成绩分段；而学生则可以查询课程成绩，具体内容如下表：

表 2-3 成绩信息管理功能表

学生	查询课程成绩	
教师	登记成绩信息	统计成绩分段

2.3 运行环境

- **操作系统和版本：**推荐 Windows7 及以上版本，或者其它可行的操作系统；
- **软件环境：**推荐使用 Eclipse，并设置字符编码为 GBK 格式。

2.4 设计和实现上的限制

本系统预期采用的编程语言为 Java，编码方式为 GBK 格式，交互界面为 Java GUI，界面语言预期为简体中文，运行软件平台推荐为 Eclipse，使用其它标准或方法设计实现可能会导致部分功能无法实现。

3. 外部接口需求

本部分主要描述成绩管理系统的外部接口以及对应需求。由于系统设计较为简单，暂未涉及外部软件的组件以及额外通讯功能相关需求，以下着重分析用户界面需求。

3.1 用户界面需求

3.1.1 账户登录界面

运行软件后首先展示账户登陆界面，界面主要内容描述如下表所示：

表 3-1 账户登陆界面

界面部件	对应的描述
ID 号输入	用于输入用户特有的账号
密码输入	用于输入对应 ID 账号的密码
端口选择	根据不同用户身份（学生、教师和管理员）可以选择不同端口登录，从而进入不同主界面
登录按钮	单击“登录”，若账户以及对应密码相匹配，进入后续操作主界面；若不匹配，则显示“无此用户，或者密码输入错误”

对应的用户登录活动图大致如下图：

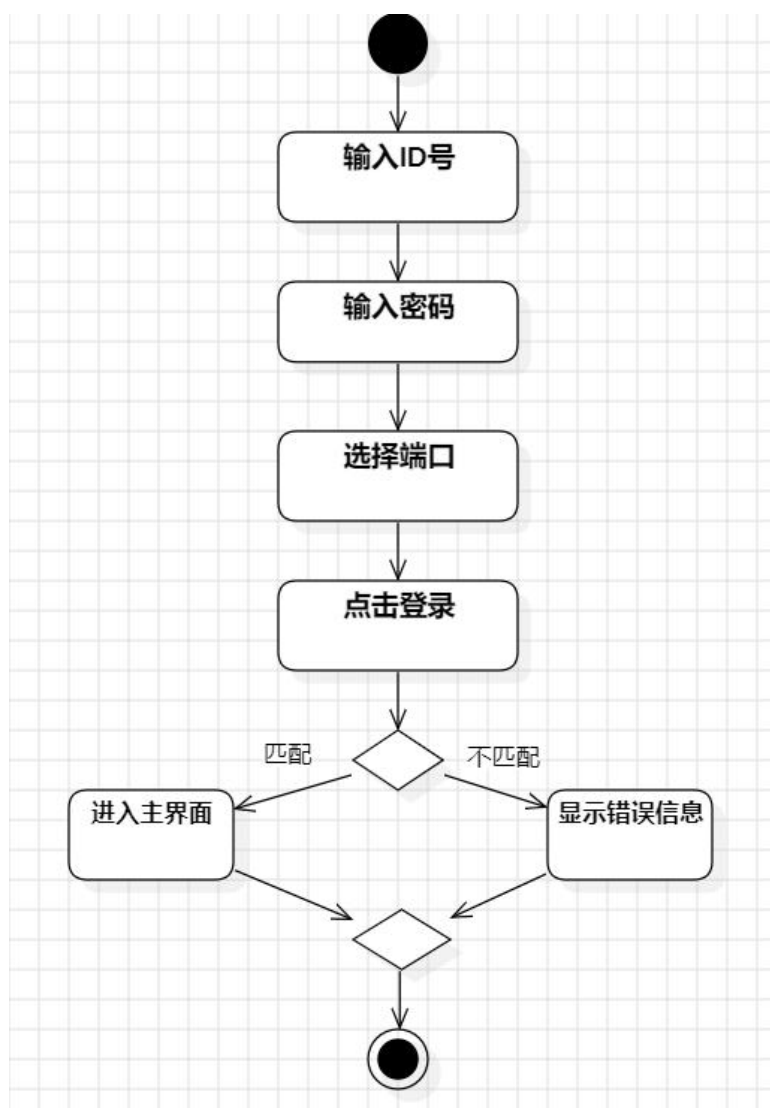


图 3-1 登录活动图

3.1.2 学生端主界面

选择学生身份登录后进入该主界面，学生端主界面主要信息如下表所示：

表 3-2 学生端主界面描述

界面部件	对应的描述
信息查询	单击后进入学生个人信息查询功能界面
成绩查询	单击后进入学生课程成绩查询功能界面
课程查询	单击后进入学生课程信息查询功能界面
修改信息	单击后进入学生个人信息修改功能界面

此外，通过界面右上角的叉除标识，学生可以在操作完成后退出界面。

3.1.3 教师端主界面

教师可以在选择身份并登录后进入该主界面，教师端的界面主要信息如下表所示：

表 3-2 教师端主界面描述

界面部件	对应的描述
信息查询	单击后进入教师个人信息查询功能界面
修改信息	单击后进入修改教师个人信息功能界面
开课	单击后教师可以选择增设课程，填写对应信息
全部课程	单击后进入教师课程信息查询功能界面
成绩登录	单击后教师可以选择登记相关课程成绩
成绩统计	单击后进入课程成绩统计功能界面

此外，通过界面右上角叉除标识，教师可以在操作完成后退出界面。

3.1.4 系统管理员端主界面

管理员可以在成功验证登录后进入该界面，界面主要信息如下表所示：

表 3-3 管理员端主界面描述

界面部件	对应的描述
修改信息	单击后进入管理员个人信息修改功能界面
增加用户	单击后进入管理员增加用户功能界面
修改信息	单击后进入管理员个人信息修改功能界面

此外，通过界面右上角叉除标识，学生可以在操作完成后退出界面。

3.1.5 其它功能界面

对于学生、教师和管理员三种用户，在登录对应账号之后，针对信息查询、修改等功能相关的界面描述如下：

- **个人信息查询界面：**包括学生端和教师端，显示账号 ID、名字、性别、生日、学院、系别；
- **学生端成绩查询界面：**每个课程按照课程号排序，依次显示课程号、课程名、

教师工号、教师姓名、学号、学生姓名以及成绩；

- **个人信息修改界面：**包括学生、教师和管理员，显示姓名、性别、学院、专业和密码，修改完成后可以单击“提交”，保存信息；
- **教师开课界面：**界面中要求教师填写的信息栏有：课程号、课程名、学分、学时、教师名称以及教师号；
- **教师端成绩录入界面：**界面要求先输入课程号，再提交学生学号、成绩以及姓名等信息；
- **教师端成绩统计界面：**界面要求先设置成绩统计分数标准，再展示不及格人数、及格人数、良好人数以及优秀人数信息；
- **管理员端添加用户界面：**界面包括输入添加信息，含姓名、账号、性别、生日、学院、用户角色以及专业等，之后单击“提交”保存；
- **管理员端删除用户界面：**界面包括输入删除用户的账号以及选择对应角色，之后单击“提交”执行。

3.2 硬件接口

考虑到数据的备份以及人机交互要求，以下为学生成绩管理系统中使用到的硬件相关服务：

- 键盘、鼠标的输入
- 读取、写入文本文件
- 读取、写入电脑硬盘

4. 系统用例设计

本部分主要涉及系统用例的定义，更加详细的描述将在下节的“系统功能需求”部分。

4.1 总体用例设计

系统功能主要包括账号、信息、课程以及成绩的部分，对应用户以及用例如下图所示：

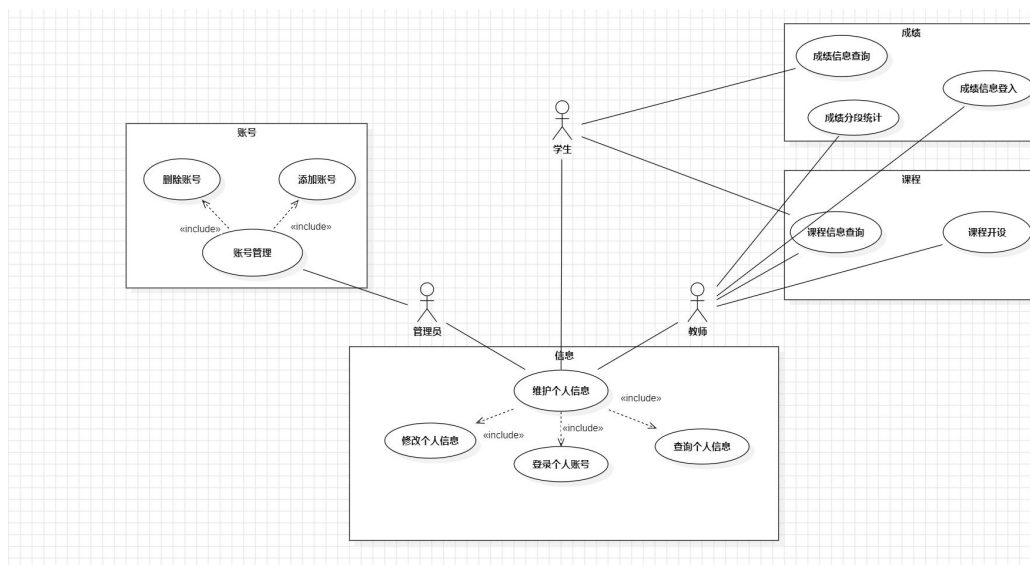


图 4-1 总体用例设计图

4.2 子用例设计

4.2.1 个人信息

用户能通过本系统维护个人信息，包括登录个人账号、查询和修改个人信息。具体用例图如下所示：

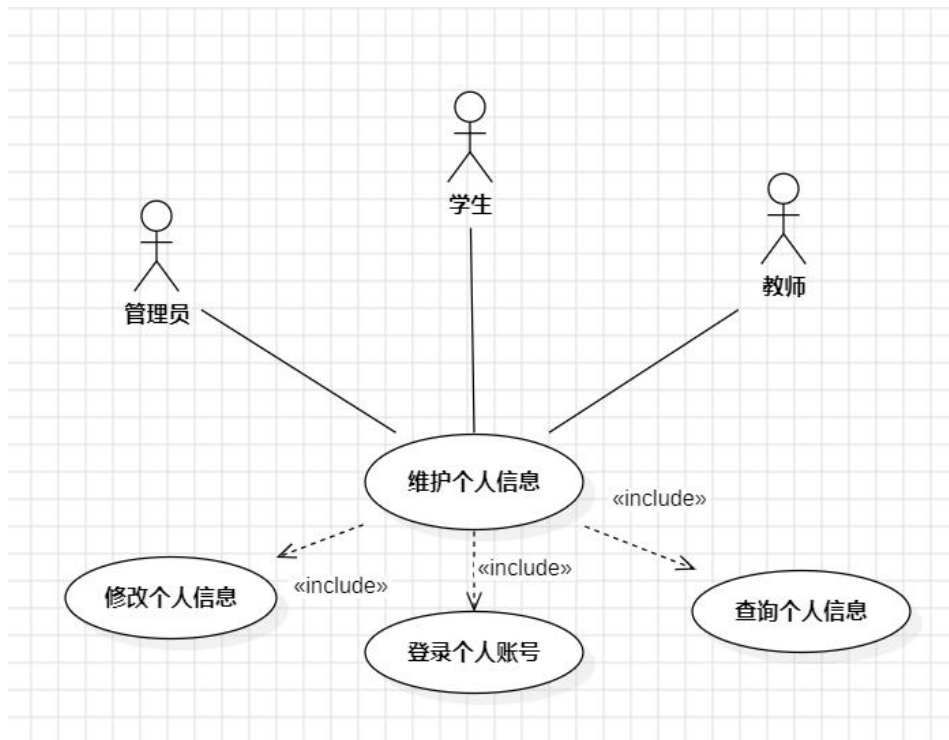


图 4-2 维护个人信息子用例图

4.2.2 账号管理

管理员可以通过本系统进行账号管理，包含删除账号和添加账号，具体用例图如下所示：

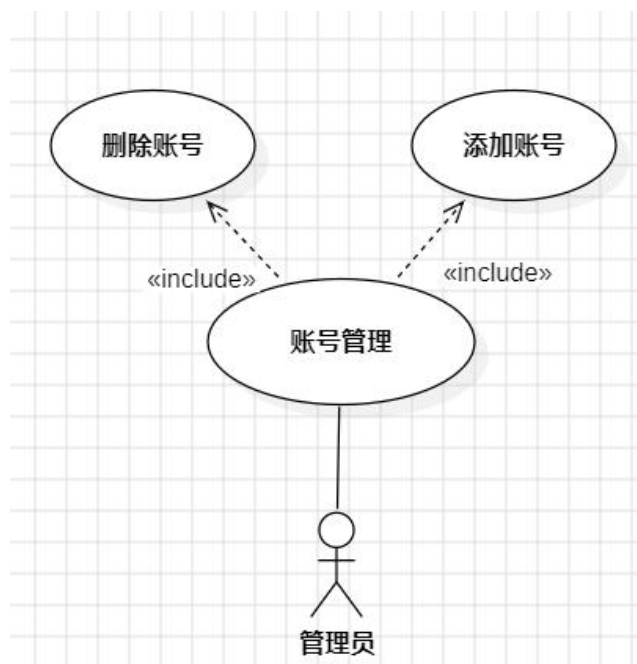


图 4-3 账号管理子用例图

4.2.3 课程相关

通过本系统，教师能够进行课程开设，学生和教师也能够查询课程信息，对应用例图如下所示：

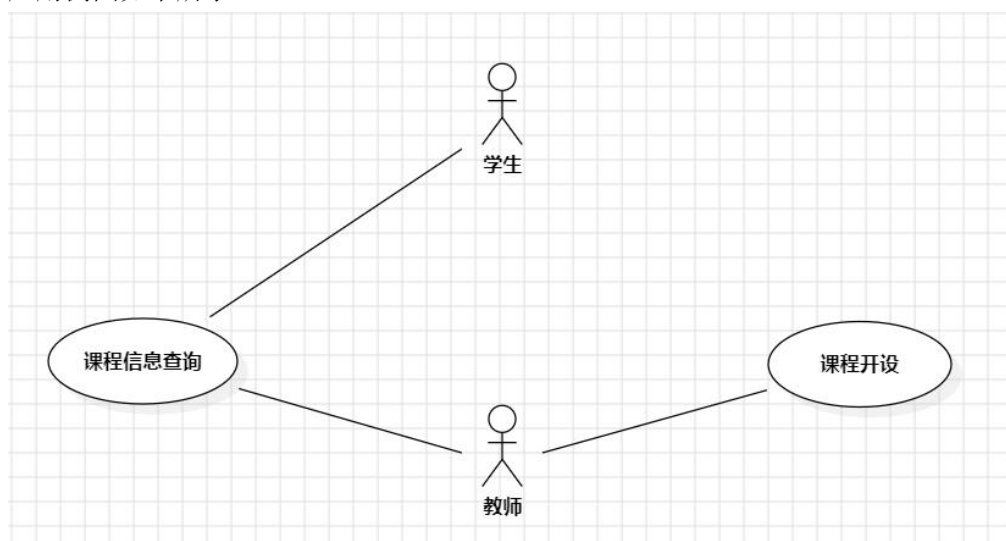


图 4-4 课程相关子用例图

4.2.4 成绩信息

通过本系统，教师能够进行课程成绩信息登入和成绩分段统计，学生能够查询相关课程成绩，具体用例图如下：

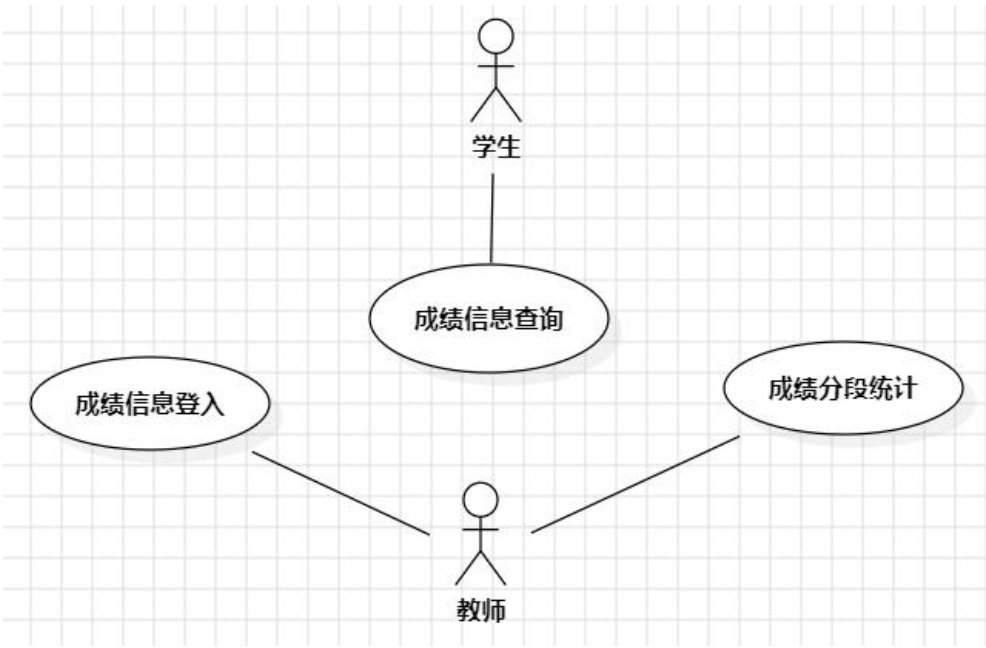


图 4-5 成绩信息子用例图

5. 系统功能需求

5.1 用例描述

本部分主要以表格形式对上一节“系统用例设计”中涉及到的活动用例进行详细解释。

5.1.1 登陆个人账号

活动用例“登录个人账号”的具体描述如下表所示：

表 5-1 “登录个人账号”用例说明

用例名	登录个人账号
简述	用户可以在登录界面进行登录
主要参与者	学生用户、教师用户、管理员用户
情景目标	用户成功登录
前提条件	程序启动，进入登录界面
触发事件	用户点击程序启动按钮
基本时间流	1. 用户点击程序启动按钮，启动该用例 2. 系统显示用户登录页面 3. 用户输入登录信息，点击提交 4. 登录成功，跳转至用户功能页面
异常处理	若用户输入修改信息格式不合法，返回至基本流步骤 3
使用频率	高

后置条件	信息修改完成跳转至用户功能界面
------	-----------------

5.1.2 添加账号

活动用例“添加账号”的具体描述如下表所示：

表 5-2 “添加账号”用例说明

用例名	添加账号
简述	用户可以在添加用户模块上添加新用户
主要参与者	系统管理员用户
情景目标	添加新用户
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击添加用户按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例 2. 用户输入添加新用户的信息并提交 3. 系统存储相关信息并提示添加成功 4. 用户关闭展示页面 5. 用例结束,回到功能界面
异常处理	若用户输入信息不合法，返回值基本步骤流 2
使用频率	高
后置条件	用户添加新用户完毕后跳转至管理员功能界面

5.1.3 删除账号

活动用例“删除账号”的具体描述如下表所示：

表 5-3 “删除账号”用例说明

用例名	删除账号
简述	用户可以在删除用户模块上删除用户
主要参与者	系统管理员用户
情景目标	删除用户
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击删除用户按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例 2. 用户输入删除用户的账号并提交 3. 系统搜索删除用户的账号并返回提示

	4. 用户关闭提示界面 5. 用例结束，跳转至功能界面
异常处理	
使用频率	低
后置条件	用户添加新用户完毕后跳转至管理员功能界面

5.1.4 查询个人信息

活动用例“查询个人信息”的具体描述如下表所示：

表 5-4 “查询个人信息”用例说明

用例名	查询个人信息
简述	用户可以在查询信息模块上查询自己的个人信息
主要参与者	学生用户、教师用户、管理员用户
情景目标	查询系统中的个人信息
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击查询信息按钮
基本时间流	1. 用户登录完毕，点击按钮，启动该用例 2. 系统显示用户信息展示页面 3. 展示用户信息完成，用户关闭信息页面 4. 用例结束，跳转至功能界面
异常处理	无
使用频率	中
后置条件	信息查询完成跳转至用户功能界面

5.1.5 修改个人信息

活动用例“修改个人信息”的具体描述如下表所示：

表 5-5 “修改个人信息”用例说明

用例名	修改个人信息
简述	用户可以在修改信息模块上修改个人信息
主要参与者	学生用户、教师用户、管理员用户
情景目标	修改系统中的个人信息
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击信息修改按钮

基本时间流	1. 用户登录完毕，点击按钮，启动该用例 2. 系统显示信息修改填写页面 3. 用户输入修改后的信息，点击提交 4. 系统存储修改后的用户信息 5. 修改完成，提示修改成功，跳转至功能界面
异常处理	若用户输入修改信息格式不合法，返回至基本流步骤 3
使用频率	中
后置条件	信息修改完成跳转至用户功能界面

5.1.6 课程信息查询

活动用例“课程信息查询”的具体描述如下表所示：

表 5-6 “课程信息查询”用例说明

用例名	课程信息查询
简述	用户可以在课程查询模块上查询课程信息
主要参与者	学生用户、教师用户
情景目标	课程查询
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击课程查询按钮
基本时间流	1. 用户登录完毕，点击按钮，启动该用例 2. 系统根据用户信息搜索数据中与用户相关的课程 3. 系统展示用户相关课程 4. 展示用户课程完成，用户关闭展示页面 5. 用例结束，跳转至功能界面
异常处理	若教师输入的科目不存在，返回至基本流步骤 2
使用频率	高
后置条件	用户课程查询用例完成跳转至用户功能界面

5.1.7 课程开设

活动用例“课程开设”的具体描述如下表所示：

表 5-7 “课程开设”用例说明

用例名	课程开设
简述	用户可以在开课模块上开设新课程

主要参与者	教师用户
情景目标	开设新课程
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击开课按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例 2. 用户输入新开设课程的信息并提交 3. 系统存储相关信息并提示添加成功 4. 用户关闭展示页面 5.用例结束,回到功能界面
异常处理	<ol style="list-style-type: none"> 1. 若有输入行为空，返回至基本流步骤 2 2. 若无输入行为空但有信息冲突，返回至基本流步骤 2
使用频率	低
后置条件	用户使用课程修改功能完成跳转至教师功能界面

5.1.8 成绩信息登入

活动用例“成绩信息登入”的具体描述如下表所示：

表 5-8 “成绩信息登入”用例说明

用例名	成绩信息登入
简述	用户可以在成绩录入模块上录入学生的课程成绩
主要参与者	教师用户
情景目标	录入成绩
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击成绩录入按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例 2. 用户输入录入成绩的科目并提交 3. 系统搜索该科目的成绩并展示成绩输入界面 4. 用户输入科目成绩并提交 5. 系统将该科目成绩存储 6. 系统提示存储成功

	7. 用例结束，跳转至教师功能界面
异常处理	若教师输入的科目不存在，返回至基本流步骤 2
使用频率	高
后置条件	教师成绩录入完成跳转至教师功能界面

5.1.9 成绩信息查询

活动用例“成绩信息查询”的具体描述如下表所示：

表 5-9 “成绩信息查询”用例说明

用例名	成绩信息查询
简述	用户可以在成绩查询模块上查询自己的成绩
主要参与者	学生用户
情景目标	查询成绩
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击成绩查询按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例 2. 用户输入查询成绩的课程号并提交 3. 系统搜索该科目的成绩并展示成绩页面 4. 展示用户成绩完成，用户关闭成绩展示页面 5. 用例结束，跳转至功能界面
异常处理	若用户输入的科目不存在，返回至基本流步骤 2
使用频率	高
后置条件	学生查询成绩完成跳转至学生功能界面

5.1.10 成绩分段统计

活动用例“成绩分段统计”的具体描述如下表所示：

表 5-10 “成绩分段统计”用例说明

用例名	成绩统计
简述	用户可以在成绩统计模块上统计学生的成绩情况
主要参与者	教师用户
情景目标	统计成绩
前提条件	程序启动，用户成功登录，进入功能界面
触发事件	用户点击成绩统计按钮
基本时间流	<ol style="list-style-type: none"> 1. 用户登录完毕，点击按钮，启动该用例

	2. 用户输入需要排序成绩的科目名字并提交 3. 系统搜索该科目并展示成绩排序分级输入页面 4. 用户填写成绩排序分级并提交 5. 系统根据提交的成绩排序标准对成绩排序并展示 6. 用户关闭展示页面 7. 用例结束，回到功能界面
异常处理	1. 若用户输入的科目不存在，返回至基本流步骤 2 2. 若用户输入的成绩分数段不合法，返回至基本流步骤
使用频率	低
后置条件	教师查询成绩完成跳转至教师功能界面

6. 其它非功能需求

本部分主要介绍系统的非功能需求，如安全性需求、软件质量需求等。

6.1 安全性需求

用户必须通过安全的身份认证机制登录系统, 通过 ID 号和密码进行区别于账号类型的登录, 在连续多次登录失败后还应有登录锁定功能; 账号只能通过管理员进行管理, 添加或删除。

6.2 软件质量需求

本系统采用模块化设计，以确保代码的清晰度和可读性。同时，提供详尽的文档，使用户能够轻松地理解系统的功能。系统的功能简单明了，易于使用，无需过多学习。此外，在后续设计实现系统时，程序还应提供完善的说明文档，以使用户学习、使用和确认系统成品所包含的功能。

7. 分析模型

本部分主要通过活动图、类图、顺序图等方式，对系统和用户可能涉及到的功能、行为进行建模，力求清晰直观。

7.1 活动图

7.1.1 登录个人账号

用户登录个人账号的活动流程如下所示：

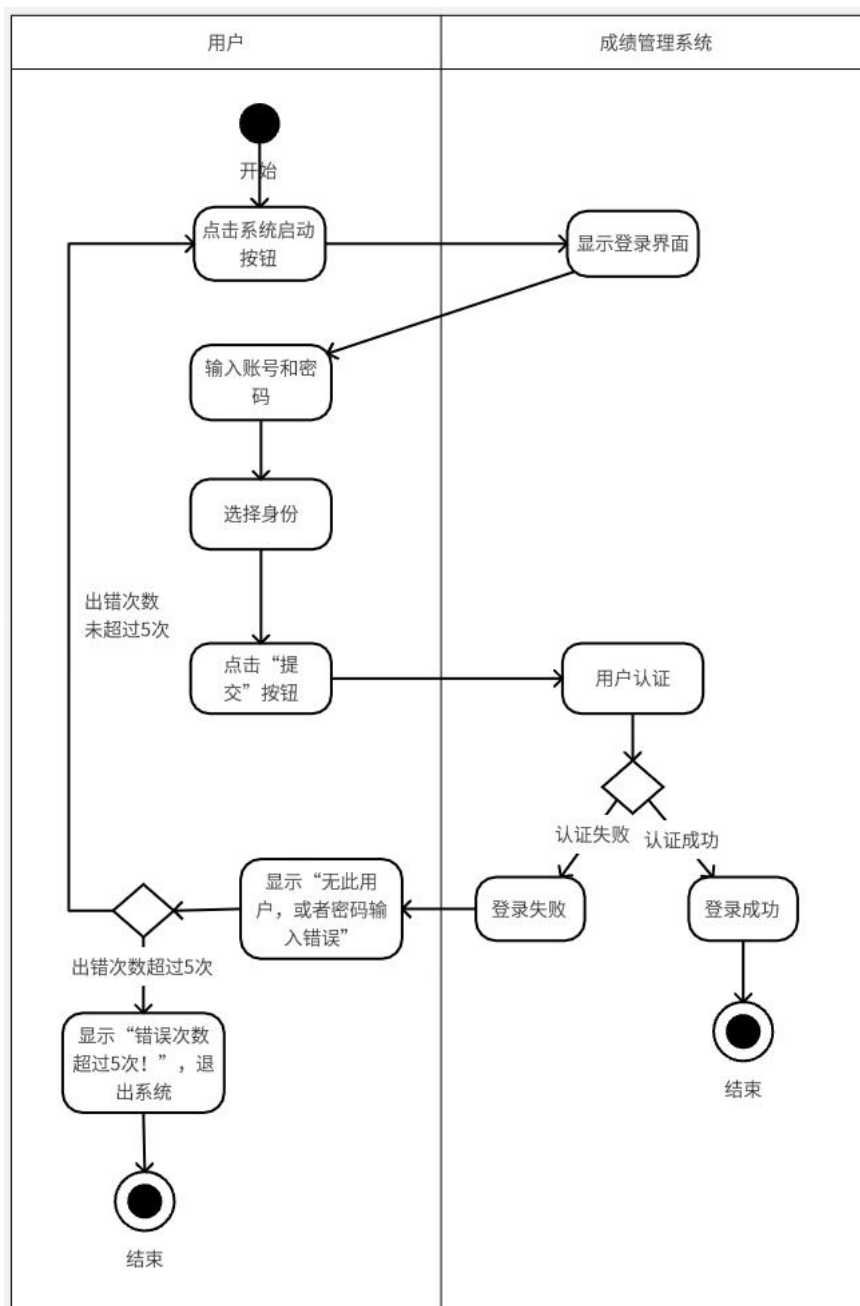


图 7-1 登录个人账号活动图

7.1.2 添加账号

用户添加账号的活动流程如下所示：

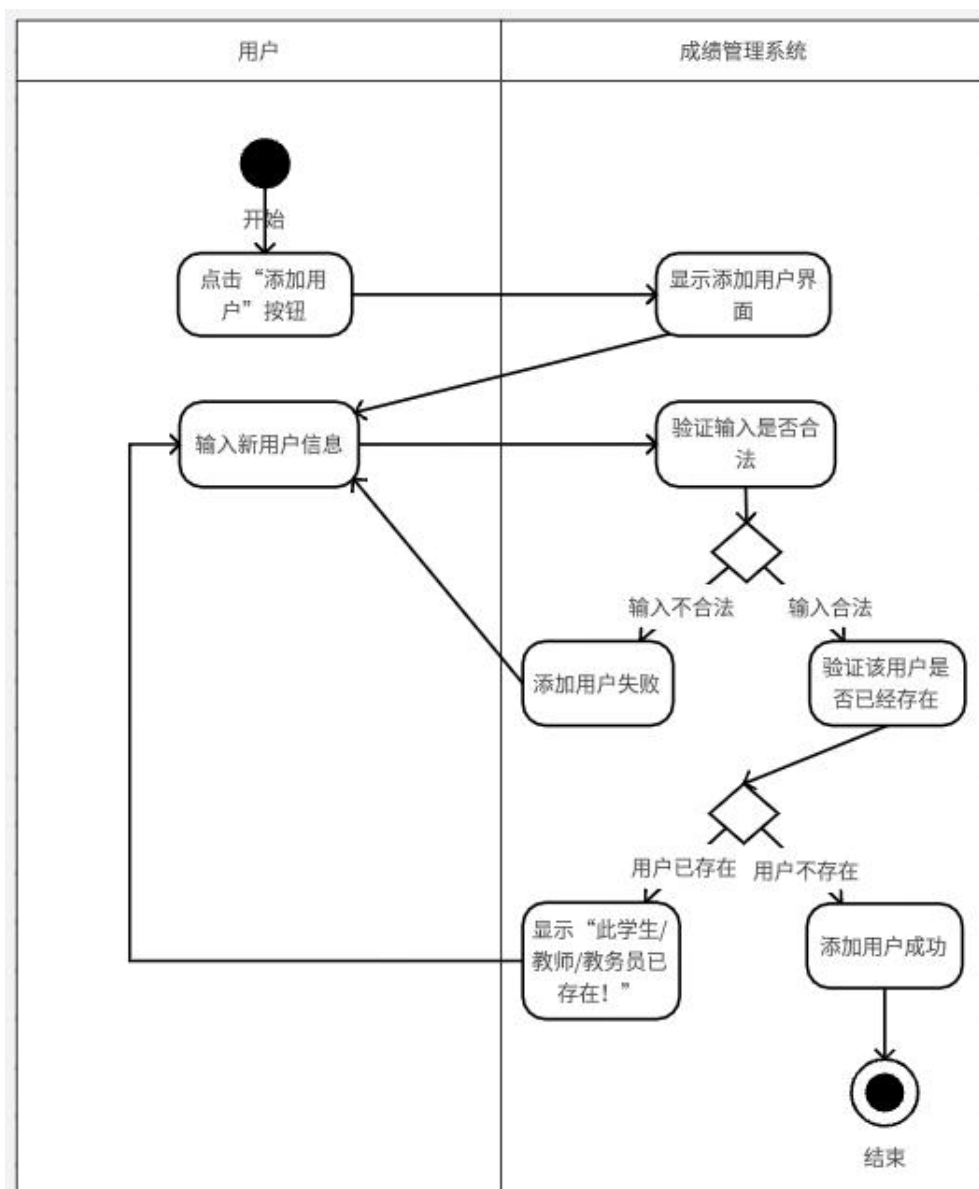


图 7-2 添加账号活动图

7.1.3 删除账号

用户删除账号的活动流程如下所示：

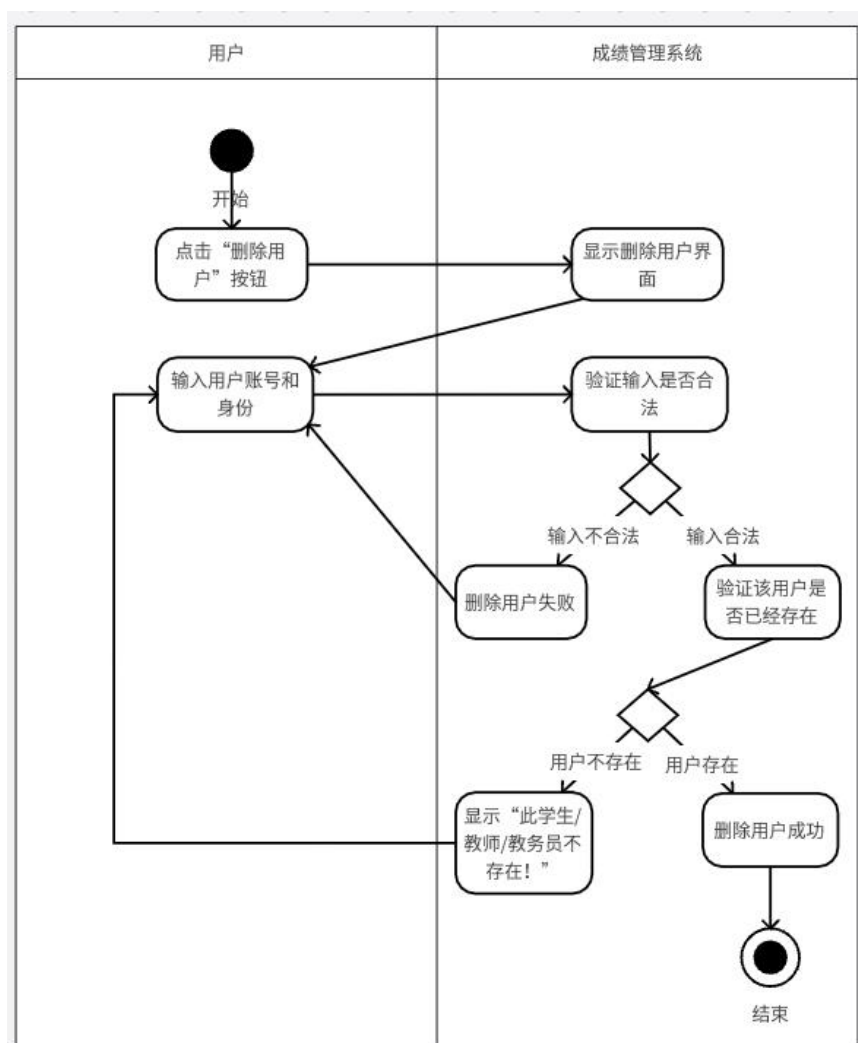


图 7-3 删除账号活动图

7.1.4 查询个人信息

用户查询个人信息的活动流程如下图所示：

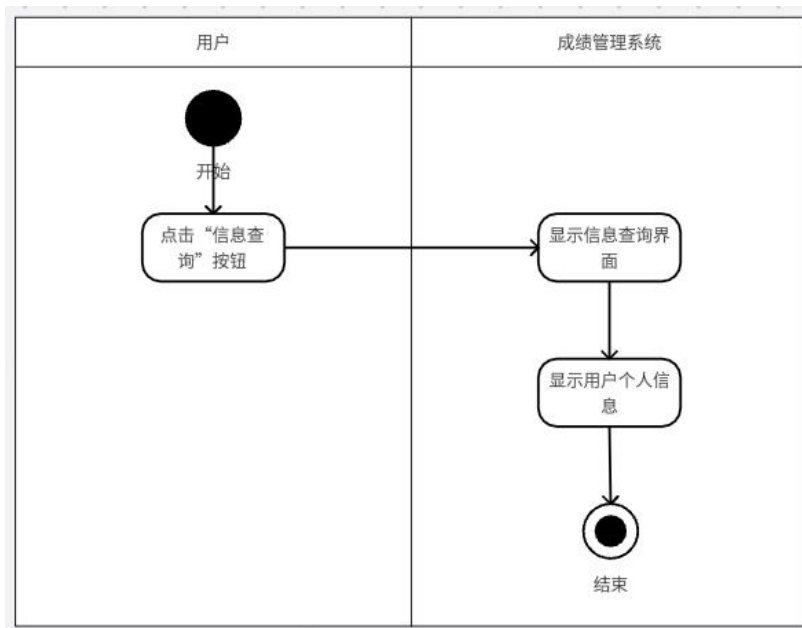


图 7-4 查询个人信息活动图

7.1.5 修改个人信息

用户修改个人信息的活动流程如下图所示：

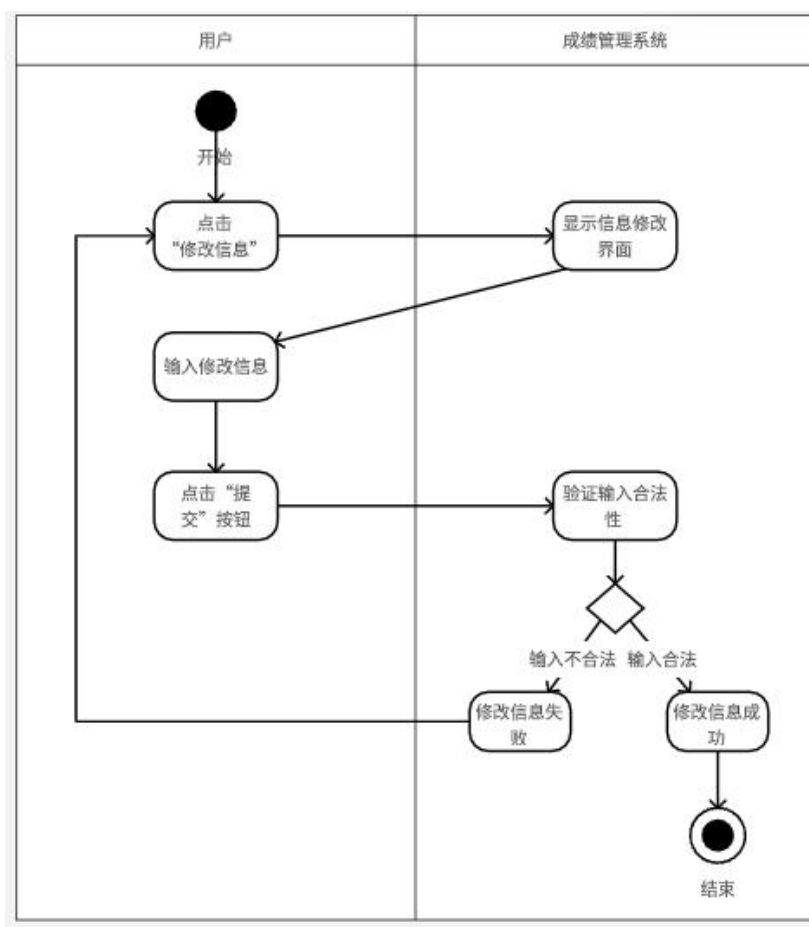


图 7-5 修改个人信息活动图

7.1.6 课程信息查询

用户查询课程信息的活动流程如下图所示：

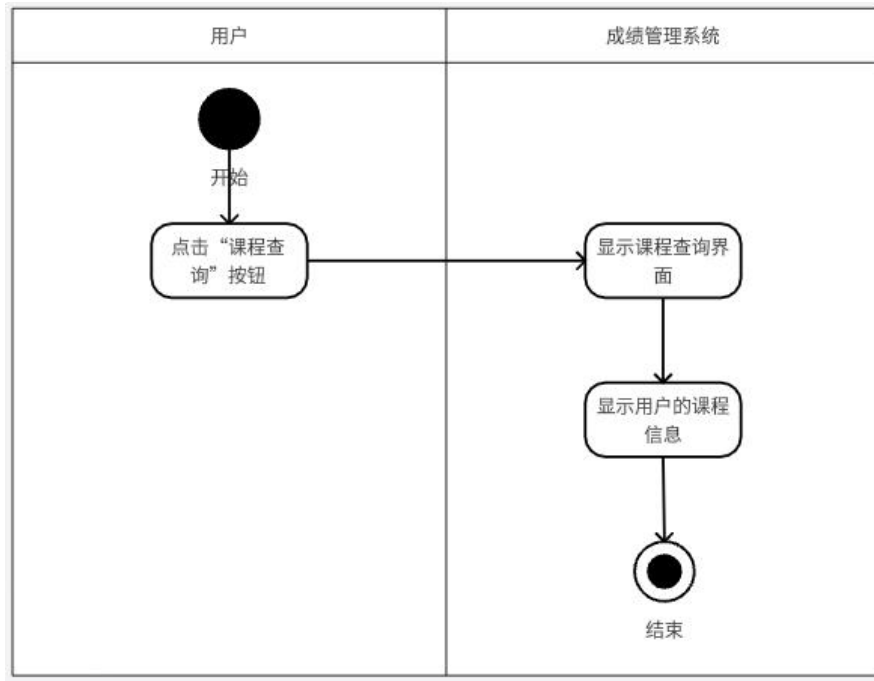


图 7-6 查询课程信息活动图

7.1.7 课程开设

用户进行课程开设的活动流程如下图所示：

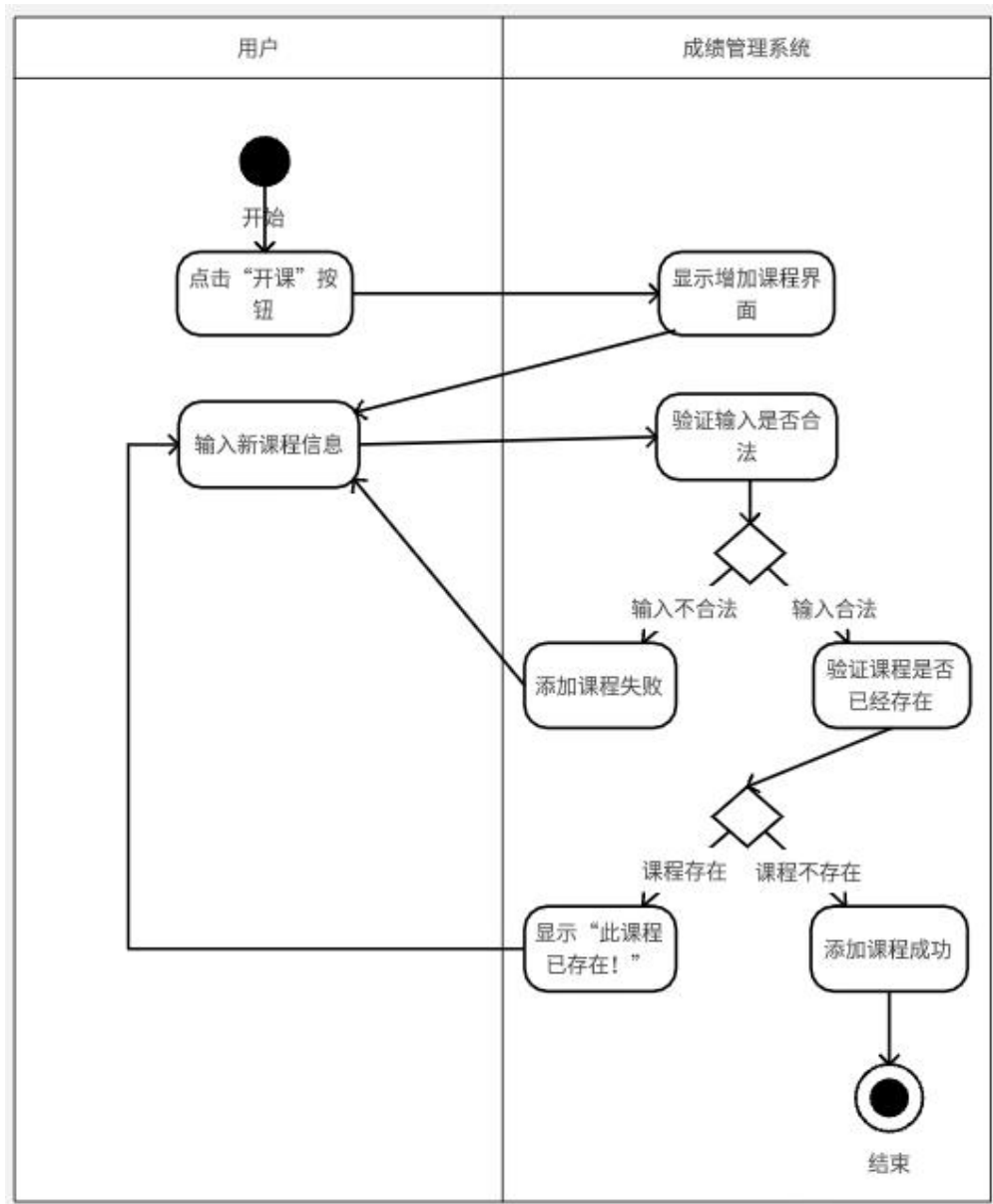


图 7-7 课程开设活动图

7.1.8 成绩信息登入

用户进行成绩信息登入的活动流程如下图所示：

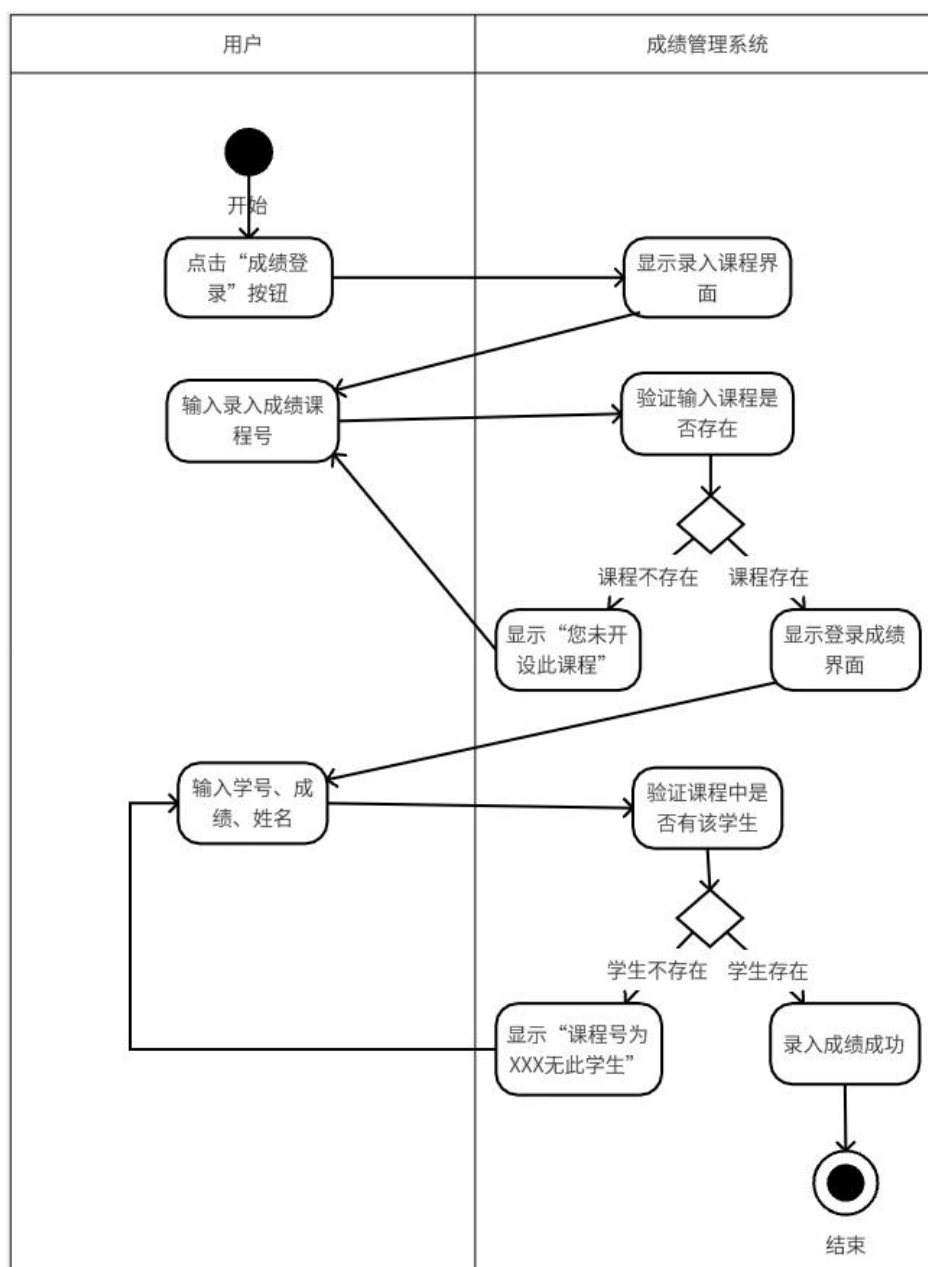


图 7-8 成绩信息登入活动图

7.1.9 成绩信息查询

用户进行成绩信息查询的活动流程如下图所示：

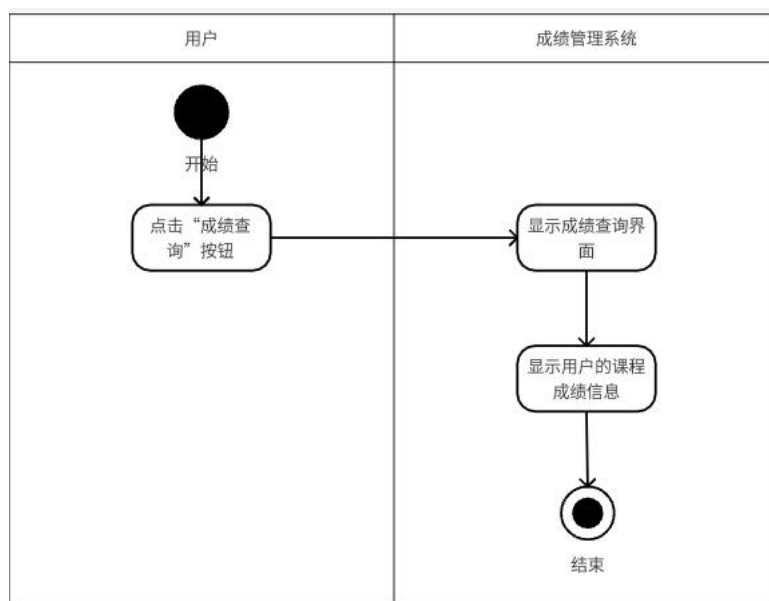


图 7-9 成绩信息查询活动图

7.1.10 成绩分段统计

用户进行成绩分段统计的活动流程如下图所示：

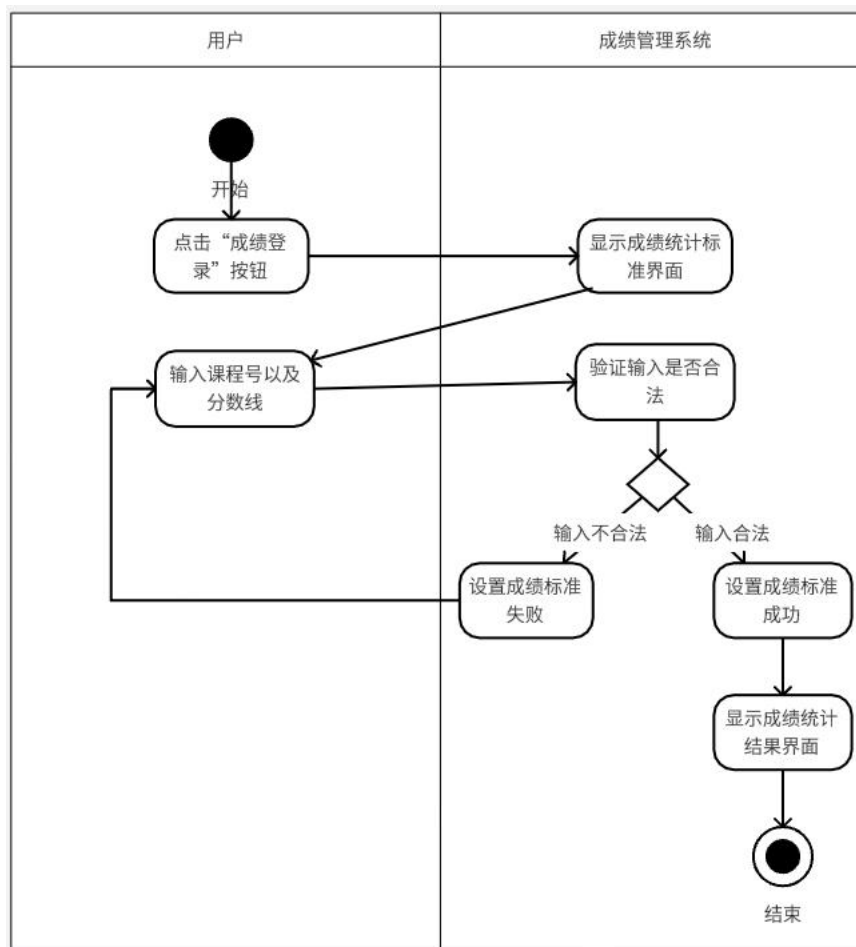


图 7-10 成绩分段统计活动图

7.2 类图

根据本系统功能设计，系统参与者主要有学生、教师和管理员，主要实体包括学生、教师、管理员、账号、课程以及成绩，结合以上信息建模得到总体类图如下：

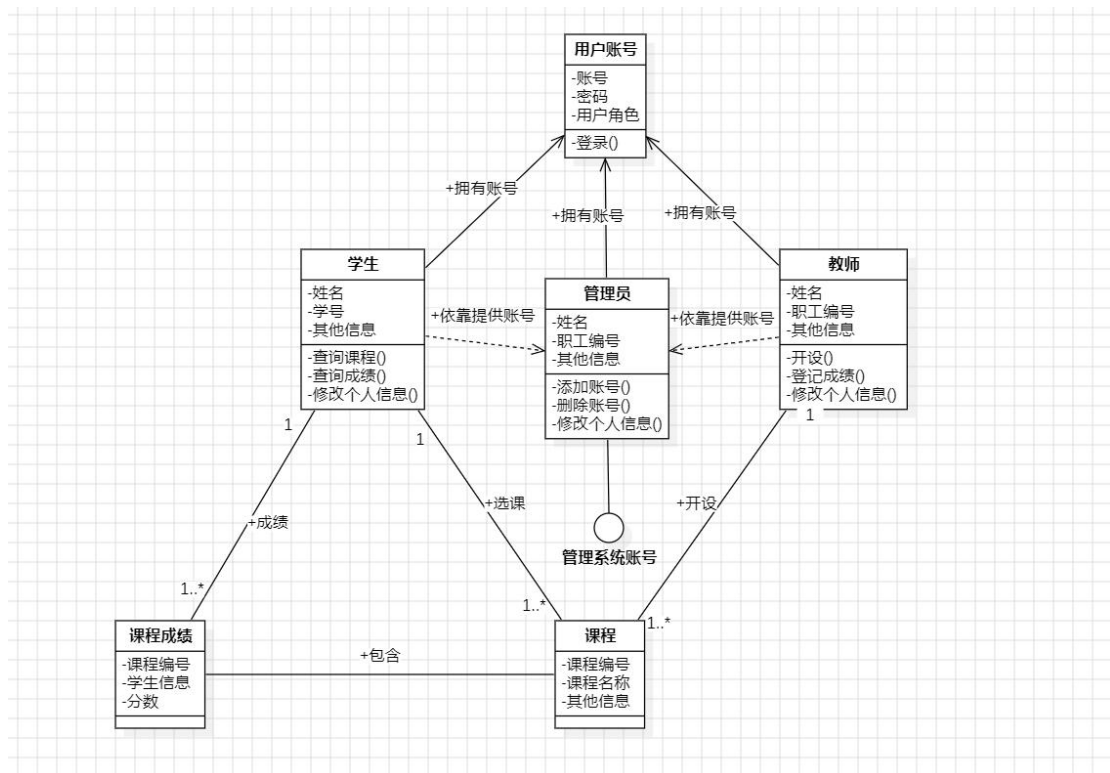


图 7-11 系统分析类图

7.3 顺序图

本部分主要对用户与系统交互活动过程中部分典型流程进行建模分析，包括用户登录、信息修改以及成绩录入。

7.3.1 用户登录

在用户登录本系统时，对应的时序活动如下图所示：

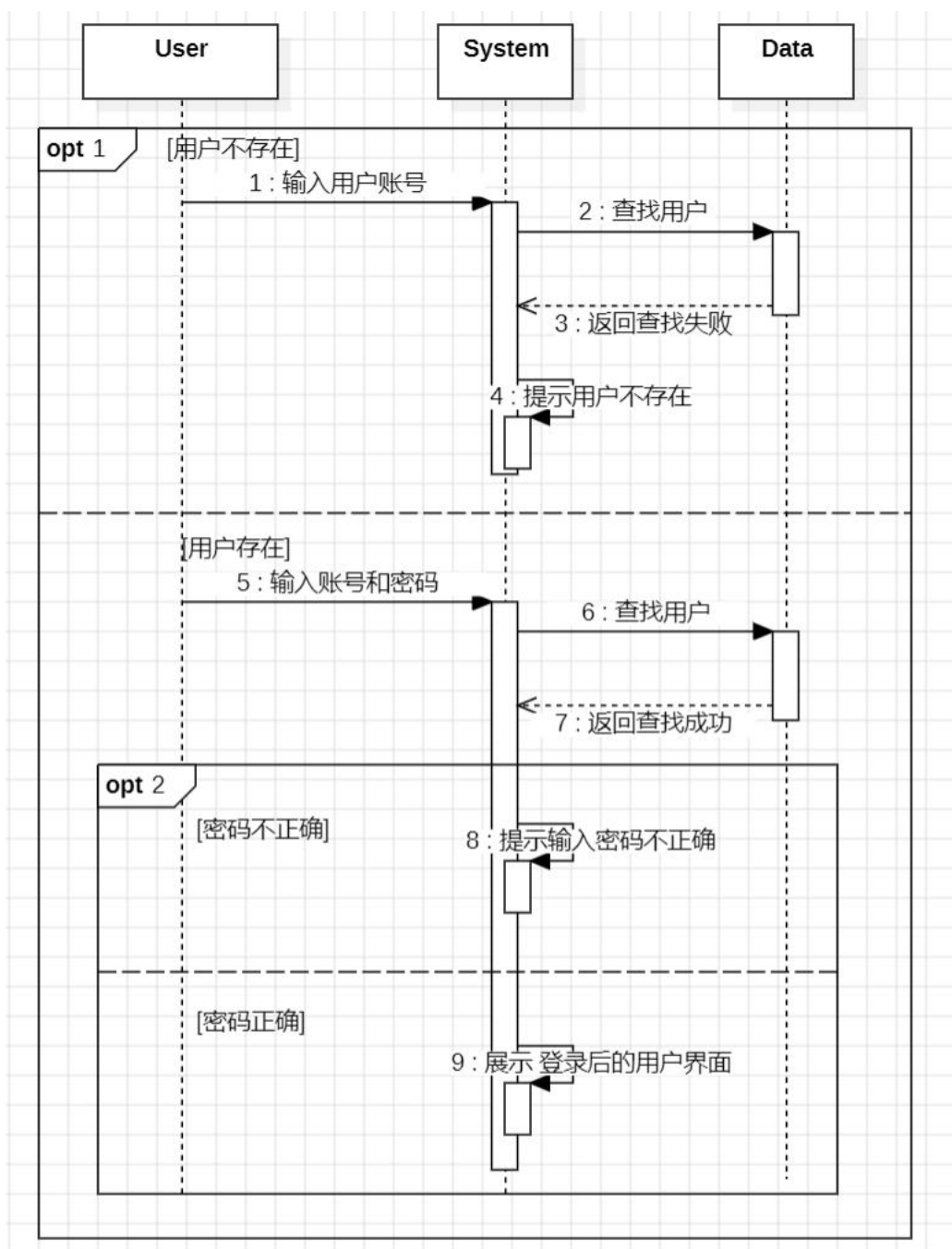


图 7-12 用户登录活动顺序图

7.3.2 信息修改

在用户修改个人信息时，对应的时序活动如下图所示：

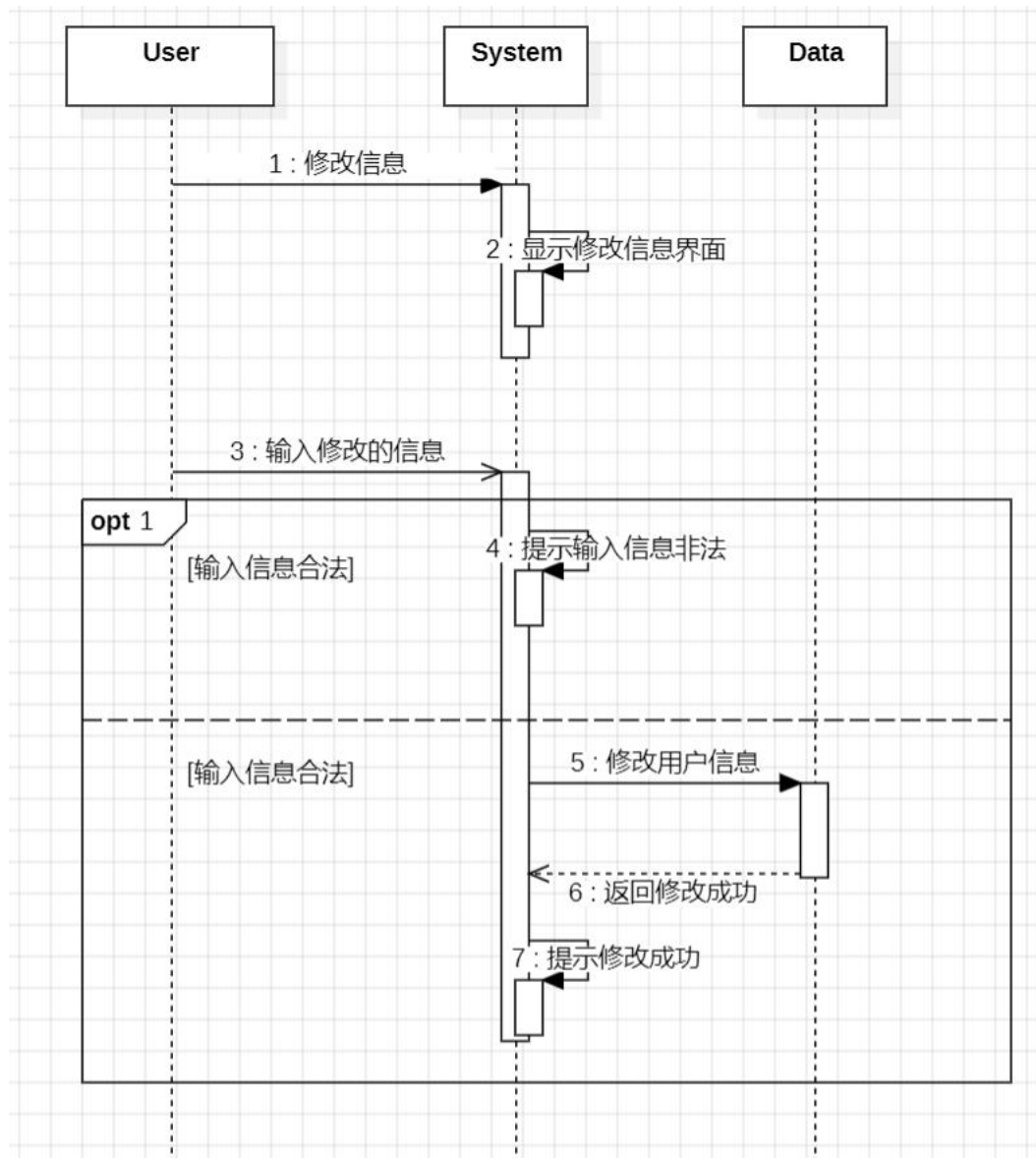


图 7-13 用户修改信息活动顺序图

7.3.3 成绩录入

在用户录入成绩信息时，对应的时序活动如下图所示：

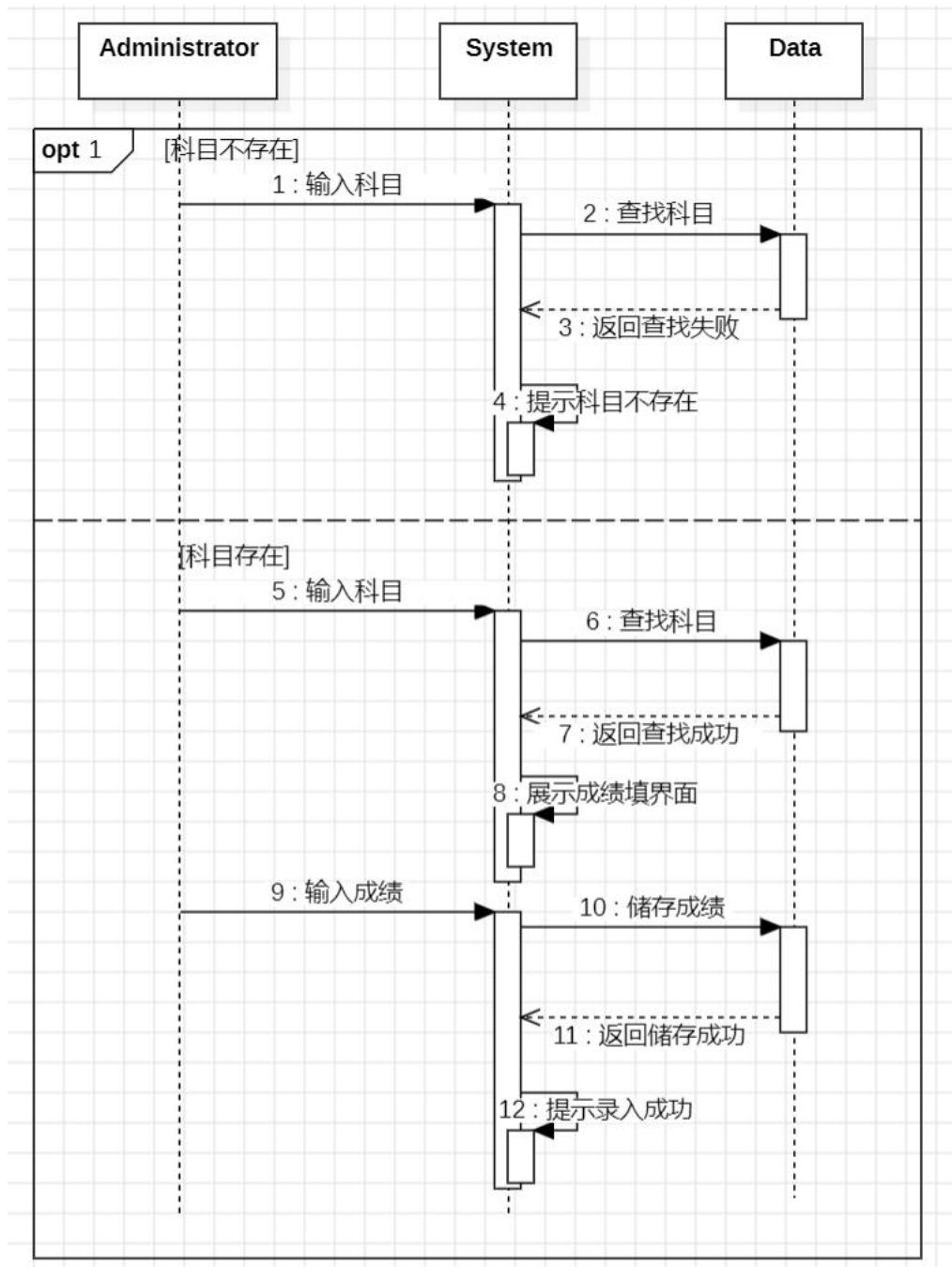


图 7-14 用户成绩录入活动顺序图

附录 B 软件概要设计报告文档模板

1. 设计概述

1.1 设计原则和设计要求

- **命名规则：**小驼峰命名法
- **模块独立性原则：**根据功能划分为相对独立的多个组件。将系统划分为学生模块、教师模块和管理员模块，每个模块负责相关的任务，实现模块间的低耦合和高内聚。
- **边界设计原则：**对可能的输入值设计多个测试用例，使这些用例分别满足语句覆盖、分支覆盖、条件覆盖以及条件组合覆盖，确保边界设计的稳定与准确。
- **必须的安全措施：**采取必要的安全措施来保护所保存的数据，包括访问控制、加密、备份和恢复等。
- **安全性和保密原则：**确保未经授权的人员无法访问、修改或删除系统中的数据，包括访问控制、加密、备份和恢复、审计和监控等。
- **系统灵活性要求：**系统需要具有一定的灵活性，以适应不同的业务需求 and 变化，包括可扩展性、可配置性和可定制性。
- **系统易操作性要求：**系统需要具备良好的用户界面和易于使用的功能，以便用户能够轻松地使用系统，包括易用性、可用性和可访问性等。
- **系统可维护性要求：**系统需要具备良好的可维护性，以便及时修复故障和进行升级，包括可测试性、可维护性和可升级性等。

1.2 限制和约束

- **技术条件：**掌握 Java 基础语法以及面向对象的设计相关技术
- **资金状况：**无
- **开发环境（包括工具和平台）：**Windows7 或其它可行操作系统，JDK8 及后续版本，Eclipse, Staruml
- **时间限制：**无
- **系统目标：**实现成绩管理系统相关数据的存储以及学生用户的个人信息维护、课程信息的查询、成绩信息的查询，教师的个人信息维护、课程信息管理（包括开设新课程、添加课程学生）、成绩信息管理以及教务管理员：用户帐号维护、学生信息管理，教师信息管理等功能。

1.3 项目风险

本部分主要具体说明本软件开发项目的全部风险承担者，以及各自在本阶段所需要承担的主要风险，首要风险承担者包括：

- **任务提出者：**学校相关部门，在本阶段的主要风险有由于需求定义不清晰、需求变更频繁、项目目标不明确等问题，所导致开发团队无法准确理解需求、项目进度延误、预算超支以及最终交付的软件无法满足期望。
- **软件开发人员：**软件开发人员，在本阶段的主要风险有由于技术难题、开发资源不足、人员变动、沟通和协作问题等问题，所导致开发过程中的延迟、质量问题以及团队效率下降。
- **产品使用者：**学生，教师，教务管理员。在本阶段的主要风险有由于用户体验不佳、功能需求不满足、安全性问题等问题，所导致产品无法得到广泛接受和应用，

用户对产品不满意，甚至可能引发用户的投诉和纠纷。

1.4 支撑环境

1.4.1 开发工具、中间件以及数据库接口

- 开发工具包：JDK8 及后续版本
- IDE:Eclipse（2024.9）
- 开发语言：Java 语言
- 涉及 Java 包：java.awt.event，java.io 包，Java Util 包

1.4.2 硬件环境

- 机型：笔记本或台式主机
- 主频：无特殊要求
- 内存容量：无特殊要求
- 输入输出设备：键盘，鼠标
- 操作系统：Windows11 操作系统或其它可行系统

2. 系统逻辑设计

2.1 系统组织设计

系统对应的子系统设计如下表所示：

表 2-1 子系统构成

子系统编号	英文名称	中文名称	业务职能	备注
1	LoadFrame	登录	负责用户的登录功能	无
2	Information Frame	信息	负责用户的个人信息管理功能	无
3	CourseFrame	课程	负责课程信息管理功能	无
4	GradeFrame	成绩	负责课程成绩信息的管理功能	无
5	UserFrame	用户	负责用户的管理功能	无

2.2 系统结构（或功能）设计

2.2.1 系统用例模块映射表

对系统各个子系统的设计映射表如下所示。

表 2-2 登录子系统映射表

子系统编号：1 子系统英文名称：LoadFrame 子系统中文名称：登录						
用例编号	系统用例英文名称	系统用例中文名称	操作功能	调用对象	被调用对象	备注
1	MainFrame	系统主界面	用户登录的主界面	无	学生、教师、管理员操作主界面	不同的用户登录后会进入不同的界面
2	StudentPanel	学生操作主界面	学生进行操作的界面	MainFrame	无	无
3	TeacherPanel	教师操作主界面	教师进行操作的界面	MainFrame	无	无
4	Administrator	管理员操作主界面	管理员进行操作的界面	MainFrame	无	无

表 2-3 信息子系统映射表

子系统编号：2 子系统英文名称：InformationFrame 子系统中文名称：信息						
用例编号	系统用例英文名称	系统用例中文名称	操作功能	调用对象	被调用对象	备注
1	CheckInfo	验证用户信息	验证用户信息	登录主界面	无	无
2	Info	查询个人信息	用户查询个人信息	学生/教师端主页面	个人信息查询功能界面	无
3	EditInfo	修改信息	用户修改信息	学生/教师/系统管理员端主页面	修改个人信息功能界面	无

表 2-4 课程子系统映射表

子系统编号：3 子系统英文名称：CourseFrame 子系统中文名称：课程						
用例编号	系统用例 英文名称	系统用例 中文名称	操作功能	调用对象	被调用 对象	备注
1	CourseView	查询课程	学生和教师查询课程	学生端/ 教师端主 页面	课程查询 功能界面	无
2	AddCourse	增加课程	教师增加 新课程	教师端主 页面	开课功能 界面	无

表 2-5 成绩子系统映射表

子系统编号：4 子系统英文名称：GradeFrame 子系统中文名称：成绩						
用例编号	系统用例 英文名称	系统用例 中文名称	操作功能	调用对象	被调用 对象	备注
1	GradeInfo	查询成绩	学生查询 成绩	学生端主 页面	成绩查询 功能界面	无
2	GradeEnter	添加成绩	教师输入 课程成绩	教师端主 页面	开课功能 界面	无
3	GradeSort	统计成绩	教师统计 成绩并将 其分类	教师端主 页面	成绩统计 功能	无

表 2-6 用户子系统映射表

子系统编号：5 子系统英文名称：UserFrame 子系统中文名称：用户						
用例编号	系统用例 英文名称	系统用例 中文名称	操作功能	调用对象	被调用 对象	备注
1	AddUser	添加用户	管理员增 加用户	系统管理 员端主页 面	添加用户 功能界面	无
2	DeleteUser	删除用户	管理员删 除用户	系统管理 员端主页 面	删除用户 功能界面	无

2.2.2 系统体系结构图或功能模块图

使用层次图对系统体系结构描述如下所示：

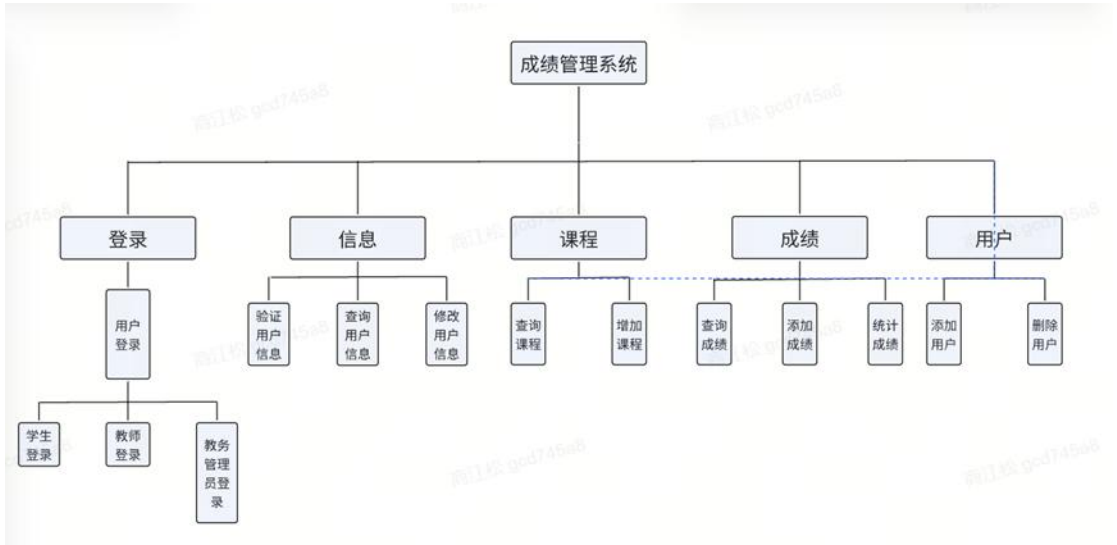


图 2-1 系统体系结构描述图

2.3 系统接口设计表

2.3.1 系统接口表

系统中相关接口名称、类型和性质的描述如下列表所示：

表 2-7 登录接口

子系统编号：1				
子系统英文名称：LoadFrame				
子系统中文名称：登录				
接口编号	接口名称	接口类型	接口性质	备注
001	登录	前端接口	输入	无
说明：用户登录				

表 2-8 信息接口

子系统编号：2				
子系统英文名称：InformationFrame				
子系统中文名称：信息				
接口编号	接口名称	接口类型	接口性质	备注
002	查询	前端接口	双向	无
003	修改	数据端接口	输入	无
说明：用户对信息的查询和修改				

表 2-9 课程接口

子系统编号：3 子系统英文名称：CourseFrame 子系统中文名称：课程				
接口编号	接口名称	接口类型	接口性质	备注
004	查询课程	前端接口	双向	无
005	添加课程	数据端接口	输入	无
说明：用户对课程的查询和添加				

表 2-10 成绩接口

子系统编号：4 子系统英文名称：GradeFrame 子系统中文名称：成绩				
接口编号	接口名称	接口类型	接口性质	备注
006	查询	前端接口	双向	无
007	添加	数据端接口	输入	无
008	统计	前端接口	双向	无
说明：用户对成绩的查询，添加和统计				

表 2-11 用户接口

子系统编号：5 子系统英文名称：UserFrame 子系统中文名称：用户				
接口编号	接口名称	接口类型	接口性质	备注
009	添加	数据端接口	输入	无
010	删除	数据端接口	输入	无
说明：管理员对用户的添加和删除				

3. 技术设计

3.1 系统开发技术说明表

关于系统中涉及到的开发技术说明如下列表格所示：

表 3-1 登录子系统技术说明

子系统编号：1						
子系统英文名称：LoadFrame						
子系统中文名称：登录						
技术编号	开发技术	开发技术	处理功能	系统特性编号		备注
	英文名称	中文名称				
001	Java	Java 语言	编写程序	001		无
002	Awtevent	java AWT 事件处理	实现对登录的响应	001		无
说明：应用于用户登录事件						

表 3-2 信息子系统技术说明

子系统编号：2						
子系统英文名称：InformationFrame						
子系统中文名称：信息						
技术编号	开发技术	开发技术	处理功能	系统特性编号		备注
	英文名称	中文名称				
001	Java	Java 语言	编写程序	002		无
002	Awtevent	java AWT 事件处理	实现用户验证	002		无
003	BufferedReader	BufferedReader	实现查询与修改信息时的输入	001	002	无
说明：应用于用户信息修改						

表 3-3 课程子系统技术说明

子系统编号：3						
子系统英文名称：CourseFrame						
子系统中文名称：课程						
技术编号	开发技术	开发技术	处理功能	系统特性编号		备注
	英文名称	中文名称				
001	Java	Java 语言	编写程序	003		无
002	Awtevent	java AWT 事件处理	实现对添加课程的响应	003		无
003	BufferedReader	BufferedReader	读取多个字符，实现对课程的输入	003		无
说明：应用于教师开课						

表 3-4 成绩子系统技术说明

子系统编号：4						
子系统英文名称：GradeFrame						
子系统中文名称：成绩						
技术编号	开发技术	开发技术	处理功能	系统特性编号		备注
	英文名称	中文名称				
001	Java	Java 语言	编写程序	004		无
002	Awtevent	java AWT 事件处理	实现对成绩输入的响应	004		无
说明：应用于教师登入学生课程成绩						

表 3-5 用户子系统技术说明

子系统编号：5						
子系统英文名称：UserFrame						
子系统中文名称：用户						
技术编号	开发技术	开发技术	处理功能	系统特性编号		备注
	英文名称	中文名称				
001	Java	Java 语言	编写程序	005		无
002	Awtevent	java AWT 事件处理	实现对用户添加与删除的响应	005	006	无
说明：应用于管理员增删用户						

3.2 开发技术应用说明

- **Java 语言：**即 Java 语言编程技术。
- **Awtevent：**实现程序对某些操作的响应，如按钮点击、鼠标操作、键盘输入等场景下的应用。
- **BufferedReader：**从字符输入流中读取文本，缓冲字符以提供字符、数组和行的高效读取。使用 BufferedReader 可以提高读取大量数据的性能，因为每次读取可以获得多个字符，而不是每次读取一个字符。

4. 进度计划

本小节主要对系统实际项目开发的进度和安排进行规划，具体时间和人员配置的规划如下表所示：

表 4-1 系统进度计划表

进度计划			
子系统	预备时间（天）	人员配备（人）	备注
信息	3	1	无
课程	2	1	无
成绩	3	1	无
用户	2	1	无
登录	2	4	待前四个模块完成后，一起完成主界面以及各个分界面

附录 C 软件详细设计报告文档模板

1. 引言

本文档是成绩管理系统的软件详细设计报告，旨在为开发团队、测试团队以及项目相关人员提供对系统设计的全面理解。通过详细描述系统的各个部件、界面、类和方法，本报告为后续的编码、测试和维护工作提供了清晰的指导。

本报告采用结构化的方式编写，分为多个章节，分别描述系统的部件设计、类设计以及界面设计。每个部分均通过表格、流程图和文字说明相结合的方式，确保信息的准确性和易读性。

1.1 文档约定

为了确保文档的一致性和可读性，本报告采用以下编写约定：

1) 部件编号方式

部件编号采用三位数字，格式为 001、002 等。

部件编号按顺序递增，便于快速定位。

2) 界面编号方式

界面编号采用三位数字，格式为 001、002 等。

界面编号按顺序递增，便于快速定位。

3) 文字格式

- 正文风格：宋体、五号；
- 一级标题：黑体、四号
- 二级标题：黑体、小四
- 三级标题：黑体、五号

2. 部件（或类、方法）详细设计

本部分主要对系统实际开发设计过程中各个部件的详细组成进行描述，具体描述如下列表格所示：

表 2-1 成绩分段设计

部件编号	001	部件名称	GradeSort
所属子系统	成绩管理子系统		
部件调用者	SortGradeFrame (部件编号: 002)		
部件被调用者	Course (部件编号: 003)		
部件入口参数	courseId: String - 课程编号 pass: float - 及格分数线 good: float - 良好分数线 excellent: float - 优秀分数线		
部件出口参数	int[] - 返回一个包含四个整数的数组, 分别表示不及格、及格、良好和优秀的学生人数。		
算法	通过调用 Course 类的 sortGrade() 方法, 统计不同分数段的学生人数。		
流程描述	1. 接收课程编号和分数线的输入。 2. 调用 Course 类的 sortGrade() 方法, 获取不同分数段的学生人数。 3. 返回统计结果。		
表示形式	类	运行环境	
性能要求	响应时间: 统计操作应快速完成。 灵活性: 应允许教师灵活查询成绩信息。 精度: 统计结果必须准确无误。		

表 2-2 成绩分段界面设计

部件编号	002	部件名称	SortGradeFrame
所属子系统	成绩管理子系统		
部件调用者	TeachersPanel (部件编号: 004)		
部件被调用者	GradeSort (部件编号: 001)		
部件入口参数	无		
部件出口参数	无		
算法	通过用户输入课程编号和分数线, 调用 GradeSort 类进行成绩统计, 并显示结果。		
流程描述	1. 显示输入界面, 用户输入课程编号和分数线。 2. 调用 GradeSort 类进行成绩统计。 3. 显示统计结果。		
表示形式	类	运行环境	
性能要求	响应时间: 界面操作应快速响应。		

表 2-3 课程系统设计

部件编号	003	部件名称	Course
所属子系统	课程管理子系统		
部件调用者	GradeSort （部件编号：001）		
部件被调用者	无		
部件入口参数	courseId: String - 课程编号 pass: float - 及格分数线 good: float - 良好分数线 excellent: float - 优秀分数线		
部件出口参数	int[] - 返回一个包含四个整数的数组，分别表示不及格、及格、良好和优秀的学生人数。		
算法	读取成绩文件，统计不同分数段的学生人数。		
流程描述	1. 根据课程编号查找对应的成绩文件。 2. 读取成绩文件，统计不同分数段的学生人数。 3. 返回统计结果。		
表示形式	类	运行环境	
性能要求	响应时间：文件读取和统计操作应快速完成。 精度：统计结果必须准确无误。		

表 2-4 教师界面设计

部件编号	004	部件名称	TeachersPanel
所属子系统	教师管理子系统		
部件调用者	MainFrame （部件编号：005）		
部件被调用者	SortGradeFrame （部件编号：002）		
部件入口参数	idd: String - 教师编号		
部件出口参数	无		
算法	提供教师操作界面，包括成绩统计、课程查询等功能。		
流程描述	显示教师操作界面。 根据用户选择，调用相应的功能模块。		
表示形式	类	运行环境	
性能要求	响应时间：界面操作应快速完成。		

表 2-5 主界面设计

部件编号	005	部件名称	MainFrame
所属子系统	用户登录子系统		
部件调用者	无		
部件被调用者	TeachersPanel （部件编号：004） StudentsPanel （部件编号：006） AdministratorPanel （部件编号：007）		
部件入口参数	无		
部件出口参数	无		
算法	提供用户登录界面，根据用户类型跳转到相应的操作界面。		
流程描述	显示登录界面，用户输入账号和密码。 根据用户类型，跳转到相应的操作界面。		
表示形式	类	运行环境	
性能要求	响应时间：登录操作快速完成。		

表 2-6 学生界面设计

部件编号	006	部件名称	StudentsPanel
所属子系统	学生管理子系统		
部件调用者	MainFrame （部件编号：005）		
部件被调用者	Info （部件编号：008） GradeInfo （部件编号：009） CourseView （部件编号：010） EditInfo （部件编号：011）		
部件入口参数	id: String - 学生编号		
部件出口参数	无		
算法	提供学生操作界面，包括信息查询、成绩查询、课程查询等功能。		
流程描述	显示学生操作界面。 根据用户选择，调用相应的功能模块。		
表示形式	类	运行环境	
性能要求	响应时间：界面操作应快速响应。		

表 2-7 信息系统设计

部件编号	007	部件名称	Info
所属子系统	用户信息管理子系统		
部件调用者	StudentsPanel （部件编号：006） TeachersPanel （部件编号：004）		
部件被调用者	无		
部件入口参数	id: String - 用户编号 flag: int - 用户类型标志（0：教师，1：学生）		
部件出口参数	无		
算法	根据用户编号和类型，查询并显示用户信息。		
流程描述	根据用户编号和类型，查询用户信息。 显示用户信息。		
表示形式	类	运行环境	
性能要求	响应时间：查询操作应快速响应。 精度：显示的信息必须准确无误。		

表 2-8 管理员界面设计

部件编号	008	部件名称	AdministratorPanel
所属子系统	管理员管理子系统		
部件调用者	MainFrame （部件编号：005）		
部件被调用者	AddUser （部件编号：012） DeleteUser （部件编号：013） EditInfo （部件编号：011）		
部件入口参数	idd: String - 管理员编号		
部件出口参数	无		
算法	提供管理员操作界面，包括添加用户、删除用户、修改信息等功能。		
流程描述	显示管理员操作界面。 根据用户选择，调用相应的功能模块。		
表示形式	类	运行环境	
性能要求	响应时间：界面操作应快速响应。		

表 2-9 成绩信息设计

部件编号	009	部件名称	GradeInfo
所属子系统	成绩管理子系统		
部件调用者	StudentsPanel (部件编号: 006)		
部件被调用者	无		
部件入口参数	id: String - 学生编号		
部件出口参数	无		
算法	根据学生编号, 查询并显示该学生的所有成绩。		
流程描述	根据学生编号, 查询成绩文件。 显示该学生的所有成绩。		
表示形式	类	运行环境	
性能要求	响应时间: 查询操作应快速响应。 精度: 显示的成绩必须准确无误。		

表 2-10 课程浏览设计

部件编号	010	部件名称	CourseView
所属子系统	课程管理子系统		
部件调用者	StudentsPanel (部件编号: 006) TeachersPanel (部件编号: 004)		
部件被调用者	无		
部件入口参数	id: String - 用户编号 flag: int - 用户类型标志 (0: 学生, 1: 教师)		
部件出口参数	无		
算法	根据用户编号和类型, 查询并显示相关课程信息。		
流程描述	根据用户编号和类型, 查询课程信息。 显示课程信息。		
表示形式	类	运行环境	
性能要求	响应时间: 查询操作应快速响应。 精度: 显示的课程信息必须准确无误。		

表 2-11 编辑信息设计

部件编号	011	部件名称	EditInfo
所属子系统	用户信息管理子系统		
部件调用者	StudentsPanel （部件编号：006） TeachersPanel （部件编号：004） AdministratorPanel （部件编号：007）		
部件被调用者	无		
部件入口参数	id: String - 用户编号 flag: int - 用户类型标志（0：学生，1：教师，3：管理员）		
部件出口参数	无		
算法	根据用户编号和类型，提供修改用户信息的界面。		
流程描述	根据用户编号和类型，显示用户信息。 用户修改信息后，保存修改。		
表示形式	类	运行环境	
性能要求	响应时间：保存操作应快速响应。 精度：修改后的信息必须准确无误。		

表 2-12 增设用户设计

部件编号	012	部件名称	AddUser
所属子系统	用户管理子系统		
部件调用者	AdministratorPanel （部件编号：007）		
部件被调用者	无		
部件入口参数	无		
部件出口参数	无		
算法	提供添加用户的界面，支持添加学生、教师和管理员。		
流程描述	显示添加用户界面。 用户输入信息后，保存新用户信息。		
表示形式	类	运行环境	
性能要求	响应时间：保存操作应快速响应。 精度：添加的用户信息必须准确无误。		

表 2-13 删除用户设计

部件编号	013	部件名称	DeleteUser
所属子系统	用户管理子系统		
部件调用者	AdministratorPanel （部件编号：007）		
部件被调用者	无		
部件入口参数	无		
部件出口参数	无		
算法	提供删除用户的界面，支持删除学生、教师和管理员。		
流程描述	显示删除用户界面。 用户选择要删除的用户后，执行删除操作。		
表示形式	类	运行环境	
性能要求	响应时间：删除操作应快速响应。 精度：删除的用户信息必须准确无误。		

2.1 类数据成员说明

2.1.1 成员变量列表

1. GradeSort 类 - controller/GradeSort.java

用于统计课程成绩的类。

表 2-14 GradeSort 类变量

序号	变量	类型	含义	修饰符
1	courseId	String	课程编号	private
2	pass	float	及格分数线	private
3	good	float	良好分数线	private
4	excellent	float	优秀分数线	private
5	course	Course	课程对象	private

2. Course 类 - model/Course.java

表示课程的类，包含课程的基本信息和成绩统计方法。

表 2-15 Course 类变量

序号	变量	类型	含义	修饰符
1	courseId	String	课程编号	private
2	courseName	String	课程名称	private
3	teacherId	String	教师编号	private
4	teacherName	String	教师姓名	private
5	credit	String	学分	private
6	hour	String	学时	private
7	fail	float	不及格分数线	private
8	pass	float	及格分数线	private

序号	变量	类型	含义	修饰符
9	good	float	良好分数线	private
10	excellent	float	优秀分数线	private

3. Administrator 类 - model/Administrator.java

表示系统管理员的类，继承自 User 类。

表 2-16 Administrator 类变量

序号	变量	类型	含义	修饰符
1	sex	String	性别	private
2	birthday	String	生日	private
3	institute	String	学院	private
4	major	String	专业	private

4. User 类 - model/User.java

表示用户的抽象类，包含用户的基本信息。

表 2-17 User 类变量

序号	变量	类型	含义	修饰符
1	id	String	用户编号	private
2	pwd	String	密码	private
3	name	String	姓名	private

5. Student 类 - model/Student.java

表示学生的类，继承自 User 类。

表 2-18 Student 类变量

序号	变量	类型	含义	修饰符
1	sex	String	性别	private
2	birthday	String	生日	private
3	institute	String	学院	private
4	major	String	专业	private

6. Teacher 类 - model/Teacher.java

表示教师的类，继承自 User 类。

表 2-19 Teacher 类变量

序号	变量	类型	含义	修饰符
1	sex	String	性别	private
2	birthday	String	生日	private
3	institute	String	学院	private
4	major	String	专业	private

7. TeachersPanel 类 - view/TeachersPanel.java

教师操作界面的类，提供教师的功能入口。

表 2-20 TeachersPanel 类变量

序号	变量	类型	含义	修饰符
1	idd	String	教师编号	private
2	contain	JPanel	面板容器	private
3	infoButton	JButton	信息查询按钮	private
4	gradeButton	JButton	成绩录入按钮	private
5	courseButton	JButton	课程查询按钮	private
6	editButton	JButton	修改信息按钮	private
7	courseView	JButton	开课按钮	private
8	sortGrade	JButton	成绩统计按钮	private

8. SortGradeFrame 类 - view/SortGradeFrame.java

成绩统计界面的类，用于输入课程编号和分数线。

表 2-21 SortGradeFrame 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	id	JLabel	课程号标签	private
3	pass	JLabel	及格标签	private
4	good	JLabel	良好标签	private
5	excellent	JLabel	优秀标签	private
6	idt	JTextField	课程号输入框	private
7	passt	JTextField	及格输入框	private
8	goodt	JTextField	良好输入框	private
9	excellentt	JTextField	优秀输入框	private
10	submit	JButton	提交按钮	private
11	result	int[]	统计结果	private

9. MainFrame 类 - view/MainFrame.java

系统主界面的类，提供用户登录功能。

表 2-22 MainFrame 类变量

序号	变量	类型	含义	修饰符
1	idTextField	JTextField	账号输入框	private
2	passwdTextField	JPasswordField	密码输入框	private
3	idLabel	JLabel	账号标签	private
4	passwdLabel	JLabel	密码标签	private
5	chooice	Choice	用户类型选择框	private
6	logon	JButton	登录按钮	private

序号	变量	类型	含义	修饰符
7	contain	JPanel	面板容器	private
8	count	int	登录尝试次数	private

10. AdministratorPanel 类 - view/AdministratorPanel.java

管理员操作界面的类，提供管理员的功能入口。

表 2-23 AdministratorPanel 类变量

序号	变量	类型	含义	修饰符
1	deleteUser	JButton	删除用户按钮	private
2	addUser	JButton	添加用户按钮	private
3	selfInfo	JButton	修改信息按钮	private
4	contain	JPanel	面板容器	private
5	idd	String	管理员编号	private

11. StudentsPanel 类 - view/StudentsPanel.java

学生操作界面的类，提供学生的功能入口。

表 2-24 StudentsPanel 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	id	String	学生编号	private
3	infoButton	JButton	信息查询按钮	private
4	gradeButton	JButton	成绩查询按钮	private
5	courseButton	JButton	课程查询按钮	private
6	editButton	JButton	修改信息按钮	private

12. Info 类 - controller/Info.java

用户信息查询界面的类，用于显示用户信息。

表 2-25 Info 类变量

序号	变量	类型	含义	修饰符
1	idLabel	JLabel	账号标签	private
2	nameLabel	JLabel	姓名标签	private
3	genderLabel	JLabel	性别标签	private
4	birLabel	JLabel	生日标签	private
5	insLabel	JLabel	学院标签	private
6	majorLabel	JLabel	专业标签	private
7	stuInfoJPanel	JPanel	信息面板	private
8	id	String	用户编号	private
9	name	String	姓名	private
10	pwd	String	密码	private

序号	变量	类型	含义	修饰符
11	gender	String	性别	private
12	birthday	String	生日	private
13	institute	String	学院	private
14	major	String	专业	private
15	stu	Student	学生对象	private
16	t	Teacher	教师对象	private

13. GradeInfo 类 - controller/GradeInfo.java

成绩查询界面的类，用于显示学生成绩。

表 2-26 GradeInfo 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	list	JTextArea	成绩列表	private
3	id	String	学生编号	private
4	courseid	String	课程编号	private
5	coursename	String	课程名称	private
6	teacherid	String	教师编号	private
7	teachername	String	教师姓名	private
8	studentid	String	学生编号	private
9	studentname	String	学生姓名	private
10	grade	String	成绩	private

14. CourseView 类 - controller/CourseView.java

课程查询界面的类，用于显示课程信息。

表 2-27 CourseView 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	list	JTextArea	课程列表	private

15. EditInfo 类 - controller/EditInfo.java

用户信息修改界面的类，用于修改用户信息。

表 2-28 EditInfo 类变量

序号	变量	类型	含义	修饰符
1	id	String	用户编号	private
2	contain	JPanel	面板容器	private
3	submit	JButton	提交按钮	private
4	name	JLabel	姓名标签	private
5	inst	JLabel	学院标签	private

序号	变量	类型	含义	修饰符
6	birth	JLabel	生日标签	private
7	pass1	JLabel	新密码标签	private
8	pass2	JLabel	确认密码标签	private
9	major	JLabel	专业标签	private
10	namet	JTextField	姓名输入框	private
11	instt	JTextField	学院输入框	private
12	birtht	JTextField	生日输入框	private
13	pass1t	JPasswordField	新密码输入框	private
14	pass2t	JPasswordField	确认密码输入框	private
15	major t	JTextField	专业输入框	private
16	check1	Checkbox	男性选择框	private
17	check2	Checkbox	女性选择框	private
18	group	CheckboxGroup	性别选择组	private
19	flag	int	用户类型标志	private

16. AddUser 类 - controller/AddUser.java

添加用户界面的类，用于添加新用户。

表 2-29 AddUser 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	id	JLabel	编号标签	private
3	name	JLabel	姓名标签	private
4	birthday	JLabel	生日标签	private
5	institute	JLabel	学院标签	private
6	major	JLabel	专业标签	private
7	idt	JTextField	编号输入框	private
8	namet	JTextField	姓名输入框	private
9	birthdayt	JTextField	生日输入框	private
10	institutet	JTextField	学院输入框	private
11	major t	JTextField	专业输入框	private
12	check1	Checkbox	男性选择框	private
13	check2	Checkbox	女性选择框	private
14	group	CheckboxGroup	性别选择组	private
15	submit	JButton	提交按钮	private
16	chooice	Choice	用户类型选择框	private
17	file	String	文件路径	private

17. DeleteUser 类 - controller/DeleteUser.java

删除用户界面的类，用于删除用户。

表 2-30 DeleteUser 类变量

序号	变量	类型	含义	修饰符
1	contain	JPanel	面板容器	private
2	id	JLabel	编号标签	private
3	idt	JTextField	编号输入框	private
4	chooice	Choice	用户类型选择框	private
5	submit	JButton	提交按钮	private
6	file	String	文件路径	private

2.2 类方法成员的签名、算法和使用说明

2.2.1 成员方法/函数列表（示例）

1. GradeSort 类

表 2-31 GradeSort 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	GradeSort	String courseId, float pass, float good, float excellent	void	构造函数，初始化成绩统计对象	public
2	hasCourse	无	int	检查课程是否存在	public
3	sortGrade	无	int[]	统计不同分数段的学生人数	public
4	isValidate	无	int	验证成绩标准是否合法	public

2. Course 类

表 2-32 GradeSort 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	Course	String courseId, float pass, float good, float excellent	void	构造函数，初始化课程对象	public
2	Course	String courseId, String courseName, String teacherId, String	void	构造函数，初始化课程对象	public

		teacherName, String credit, String hour			
3	Course	String courseId, String courseName, String teacherId, String teacherName, float fail, float pass, float good, float excellent	void	构造函数，初始化 课程对象	public
4	getCourseId	无	String	获取课程编号	public
5	setCourseId	String courseId	void	设置课程编号	public
6	getCourseName	无	String	获取课程名称	public
7	setCourseName	String courseName	void	设置课程名称	public
8	getTeacherId	无	String	获取教师编号	public
9	setTeacherId	String teacherId	void	设置教师编号	public
10	getTeacherName	无	String	获取教师姓名	public
11	setTeacherName	String teacherName	void	设置教师姓名	public
12	getCredit	无	String	获取学分	public
13	setCredit	String credit	void	设置学分	public
14	getHour	无	String	获取学时	public
15	setHour	String hour	void	设置学时	public
16	getFail	无	float	获取不及格分数线	public
17	setFail	float fail	void	设置不及格分数线	public
18	getPass	无	float	获取及格分数线	public
19	setPass	float pass	void	设置及格分数线	public
20	getGood	无	float	获取良好分数线	public
21	setGood	float good	void	设置良好分数线	public
22	getExcellent	无	float	获取优秀分数线	public
23	setExcellent	float excellent	void	设置优秀分数线	public
24	hasCourse	无	int	检查课程是否存在	public
25	sortGrade	无	int[]	统计不同分数段的 学生人数	public

26	isValidate	无	int	验证成绩标准是否合法	public
----	------------	---	-----	------------	--------

3. Administrator 类

表 2-33 GradeSort 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	Administrator	无	void	默认构造函数	public
2	Administrator	String id, String pwd, String name, String sex, String birthday, String institute, String major	void	构造函数，初始化 管理员对象	public
3	getSex	无	String	获取性别	public
4	setSex	String sex	void	设置性别	public
5	getBirthday	无	String	获取生日	public
6	setBirthday	String birthday	void	设置生日	public
7	getInstitute	无	String	获取学院	public
8	setInstitute	String institute	void	设置学院	public
9	getMajor	无	String	获取专业	public
10	setMajor	String major	void	设置专业	public

4. User 类

表 2-34 User 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	User	无	void	默认构造函数	public
2	User	String id, String pwd, String name	void	构造函数，初始化 用户对象	public
3	getId	无	String	获取用户编号	public
4	setId	String id	void	设置用户编号	public
5	getPwd	无	String	获取密码	public
6	setPwd	String pwd	void	设置密码	public
7	getName	无	String	获取姓名	public
8	setName	String name	void	设置姓名	public

5. Student 类

表 2-35 Student 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	Student	无	void	默认构造函数	public
2	Student	String id, String pwd, String name, String sex, String birthday, String institute, String major	void	构造函数，初始化学 生对象	public
3	getSex	无	String	获取性别	public
4	setSex	String sex	void	设置性别	public
5	getBirthday	无	String	获取生日	public
6	setBirthday	String birthday	void	设置生日	public
7	getInstitute	无	String	获取学院	public
8	setInstitute	String institute	void	设置学院	public
9	getMajor	无	String	获取专业	public
10	setMajor	String major	void	设置专业	public

6. Teacher 类

表 2-36 Teacher 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	Teacher	无	void	默认构造函数	public
2	Teacher	String id, String pwd, String name, String sex, String birthday, String institute, String major	void	构造函数，初始化教 师对象	public
3	getSex	无	String	获取性别	public
4	setSex	String sex	void	设置性别	public

5	getBirthday	无	String	获取生日	public
6	setBirthday	String birthday	void	设置生日	public
7	getInstitute	无	String	获取学院	public
8	setInstitute	String institute	void	设置学院	public
9	getMajor	无	String	获取专业	public
10	setMajor	String major	void	设置专业	public

7. TeachersPanel 类

表 2-37 TeachersPanel 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	TeachersPanel	String id	void	构造函数，初始化教师操作界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

8. SortGradeFrame 类

表 2-38 SortGradeFrame 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	SortGradeFrame	无	void	构造函数，初始化成绩统计界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	showResult	无	void	显示统计结果	private

9. MainFrame 类

表 2-39 MainFrame 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	MainFrame	无	void	构造函数，初始化主界面	public
2	actionPerformed	ActionEvent e	void	处理登录按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

10. AdministratorPanel 类

表 2-40 AdministratorPanel 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	AdministratorPanel	String id	void	构造函数，初始化管理员操作界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

11. StudentsPanel 类

表 2-41 StudentsPanel 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	StudentsPanel	String id	void	构造函数，初始化学生操作界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

12. Info 类

表 2-42 Info 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	Info	String id, int flag	void	构造函数，初始化用户信息界面	public
2	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

13. GradeInfo 类

表 2-43 GradeInfo 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	GradeInfo	String id	void	构造函数，初始化成绩查询界面	public

14. CourseView 类

表 2-44 CourseView 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	CourseView	String id, int flag	void	构造函数，初始化课程查询界面	public

15. EditInfo 类

表 2-45 EditInfo 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	EditInfo	String id, int flag	void	构造函数，初始化 信息修改界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

16. AddUser 类

表 2-46 AddUser 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	AddUser	无	void	构造函数，初始化 添加用户界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

17. DeleteUser 类

表 2-47 DeleteUser 类函数

序号	方法名	参数列表	返回类型	用途	修饰符
1	DeleteUser	无	void	构造函数，初始化 删除用户界面	public
2	actionPerformed	ActionEvent e	void	处理按钮点击事件	public
3	processWindowEvent	WindowEvent e	void	处理窗口关闭事件	protected

2.2.2 成员方法实现细节

1. GradeSort 类的 sortGrade() 方法

用途：统计不同分数段的学生人数。

- 1) 调用 Course 类的 sortGrade() 方法，获取成绩文件路径。
- 2) 读取成绩文件，逐行遍历每一条成绩记录。
- 3) 根据成绩判断其所属的分数段（不及格、及格、良好、优秀）。
- 4) 统计每个分数段的学生人数。
- 5) 返回一个包含四个整数的数组，分别表示不及格、及格、良好和优秀的学生人数。



图 2-1 sortGrade() 方法流程 1

2. Course 类的 sortGrade() 方法

用途：统计不同分数段的学生人数。

- 1) 根据课程编号查找对应的成绩文件。
- 2) 读取成绩文件，逐行遍历每一条成绩记录。
- 3) 根据成绩判断其所属的分数段（不及格、及格、良好、优秀）。
- 4) 统计每个分数段的学生人数。
- 5) 返回一个包含四个整数的数组，分别表示不及格、及格、良好和优秀的学生人数。

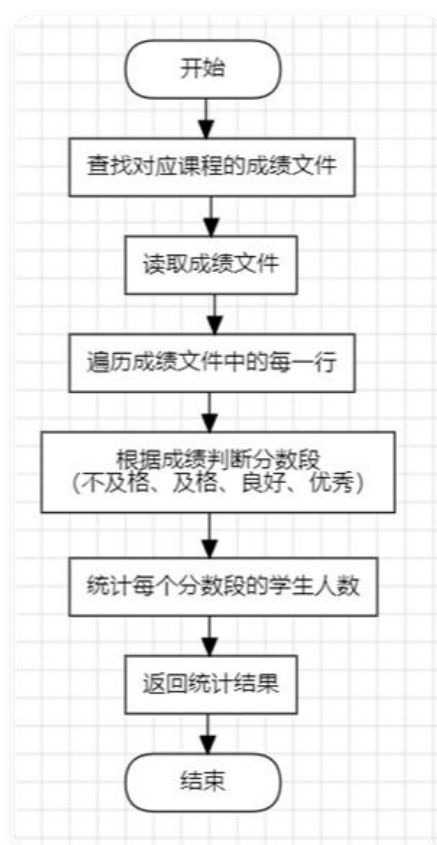


图 2-2 sortGrade() 方法流程 2

3. TeachersPanel 类的 actionPerformed() 方法

用途：处理教师操作界面中的按钮点击事件。

- 1) 根据用户点击的按钮，判断需要执行的操作。
- 2) 如果是“信息查询”按钮，调用 Info 类显示教师信息。
- 3) 如果是“成绩录入”按钮，调用 GradeEnter 类进入成绩录入界面。
- 4) 如果是“课程查询”按钮，调用 CourseView 类显示课程信息。
- 5) 如果是“修改信息”按钮，调用 EditInfo 类进入信息修改界面。
- 6) 如果是“开课”按钮，调用 AddCourse 类进入开课界面。
- 7) 如果是“成绩统计”按钮，调用 SortGradeFrame 类进入成绩统计界面。

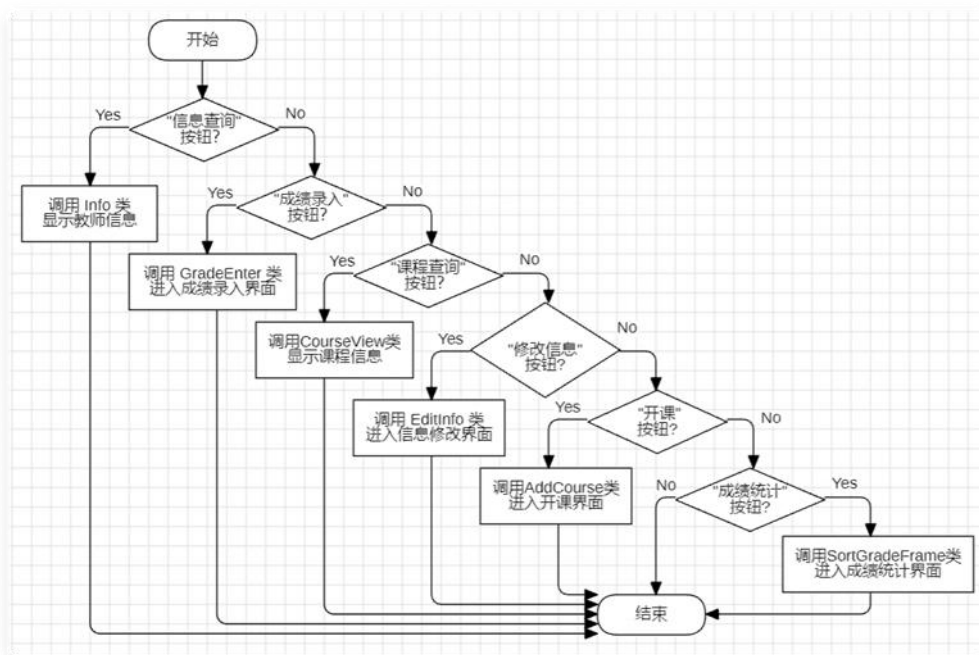


图 2-3 actionPerformed() 方法流程 1

4. SortGradeFrame 类的 actionPerformed() 方法

用途：处理成绩统计界面中的按钮点击事件。

- 1) 判断用户点击的按钮是否为“提交”按钮。
- 2) 如果点击的是“提交”按钮，检查输入框是否为空。
- 3) 如果输入为空，显示错误提示。
- 4) 如果输入不为空，调用 GradeSort 类验证成绩标准是否合法。
- 5) 如果验证失败，显示错误提示。
- 6) 如果验证成功，调用 GradeSort 类统计成绩，并显示统计结果。

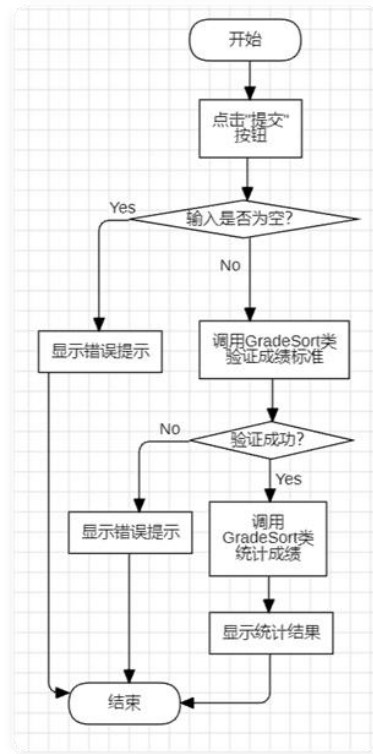


图 2-4 actionPerformed() 方法流程 2

5. MainFrame 类的 actionPerformed() 方法

用途：处理主界面中的登录按钮点击事件。

- 1) 根据用户选择的用户类型（学生、教师、系统管理员），调用 CheckInfo 类验证账号和密码。
- 2) 如果验证成功，根据用户类型跳转到相应的操作界面（StudentsPanel、TeachersPanel 或 AdministratorPanel）。
- 3) 如果验证失败，显示错误提示。

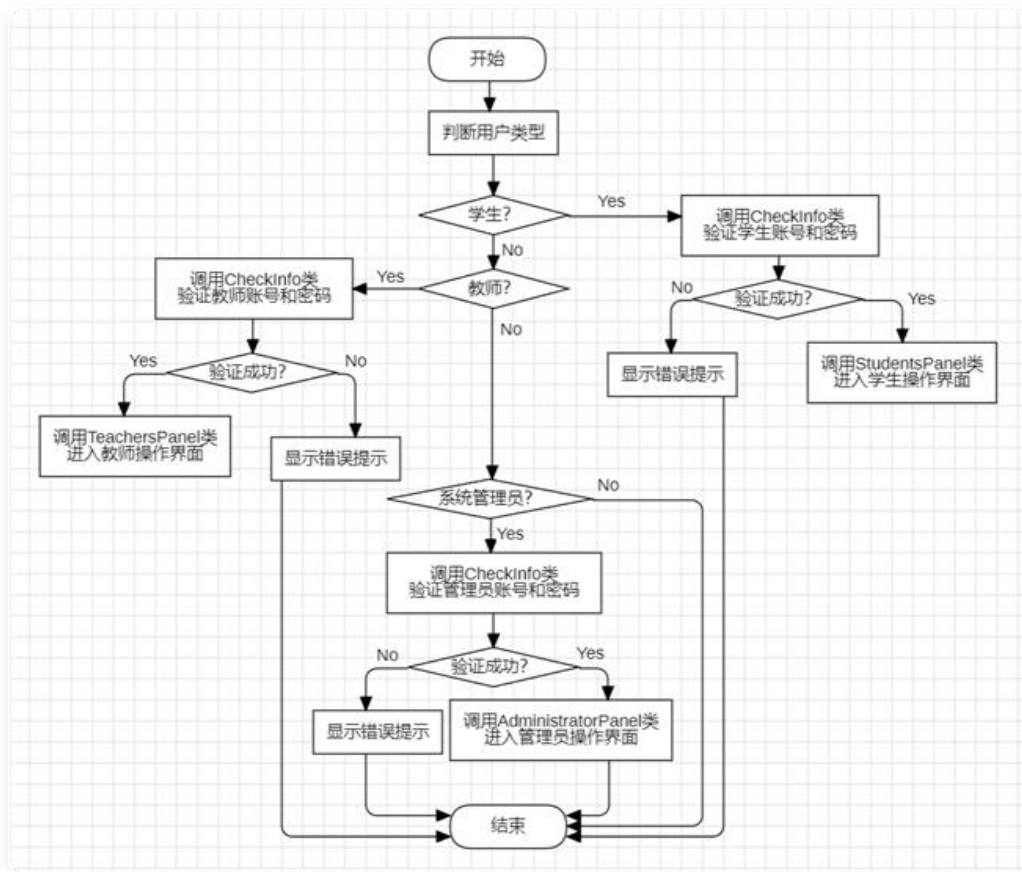


图 2-5 actionPerformed() 方法流程 3

6. EditInfo 类的 actionPerformed() 方法

用途：处理信息修改界面中的按钮点击事件。

- 1) 判断用户点击的按钮是否为“提交”按钮。
- 2) 如果点击的是“提交”按钮，检查输入框是否为空。
- 3) 如果输入为空，显示错误提示。
- 4) 如果输入不为空，检查新密码和确认密码是否一致。
- 5) 如果不一致，显示错误提示。
- 6) 如果一致，保存修改后的信息，并显示成功提示。

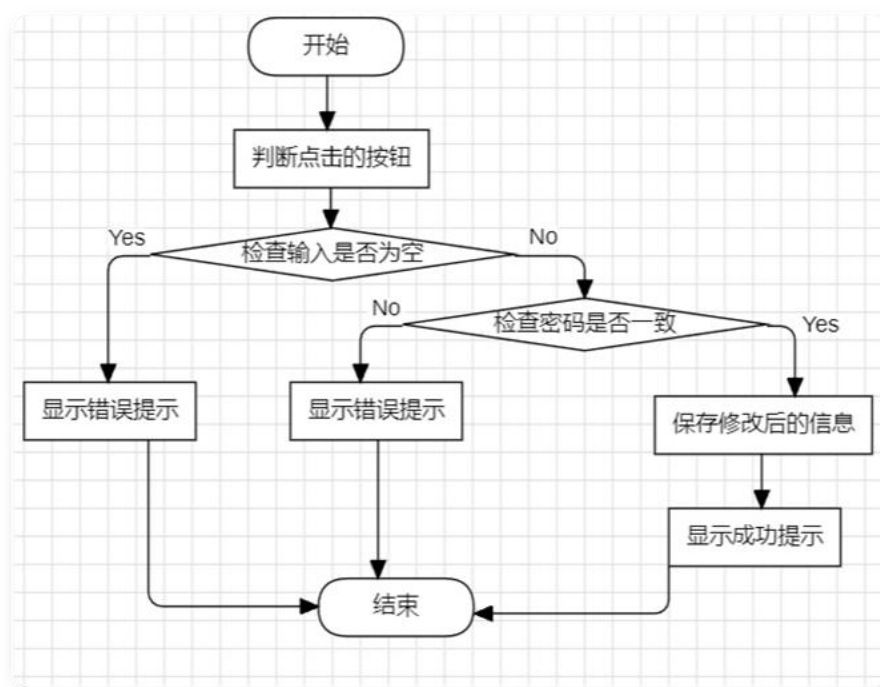


图 2-6 actionPerformed() 方法流程 4

7. AddUser 类的 actionPerformed() 方法

用途：处理添加用户界面中的按钮点击事件。

- 1) 判断用户点击的按钮是否为“提交”按钮。
- 2) 如果点击的是“提交”按钮，检查输入框是否为空。
- 3) 如果输入为空，显示错误提示。
- 4) 如果输入不为空，根据用户类型（学生、教师、管理员）添加用户。
- 5) 如果添加成功，显示成功提示。
- 6) 如果添加失败，显示错误提示。

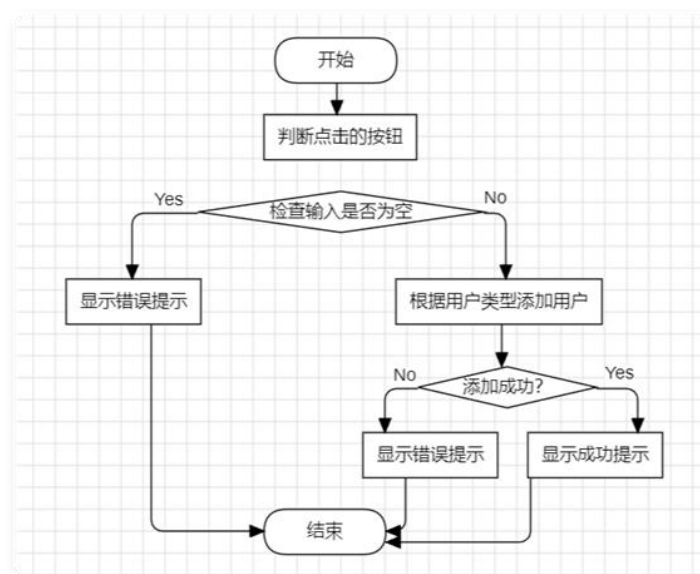


图 2-7 actionPerformed() 方法流程 5

3. 界面详细设计

3.1 MainFrame 界面

表 3-1 MainFrame 界面

界面编号	001	部件名称	MainFrame
界面性质	用户登录界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
idTextField	账号输入框	用户输入账号	
passwdTextField	密码输入框	用户输入密码	
chooice	用户类型选择框	选择用户类型（学生、教师、管理员）	

3.2 TeachersPanel 界面

表 3-2 TeachersPanel 界面

界面编号	002	部件名称	TeachersPanel
界面性质	教师操作界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
infoButton	信息查询按钮	点击后显示教师信息	
gradeButton	成绩录入按钮	点击后进入成绩录入界面	
courseButton	课程查询按钮	点击后显示课程信息	
editButton	修改信息按钮	点击后进入信息修改界面	
courseView	开课按钮	点击后进入开课界面	
sortGrade	成绩统计按钮	点击后进入成绩统计界面	

3.3 StudentsPanel 界面

表 3-3 StudentsPanel 界面

界面编号	003	部件名称	StudentsPanel
界面性质	学生操作界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
infoButton	信息查询按钮	点击后显示学生信息	

gradeButton	成绩查询按钮	点击后显示学生成绩
courseButton	课程查询按钮	点击后显示学生课程
editButton	修改信息按钮	点击后进入信息修改界面

3.4 AdministratorPanel 界面

表 3-4 AdministratorPanel 界面

界面编号	004	部件名称	AdministratorPanel
界面性质	管理员操作界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
addUser	添加用户按钮	点击后进入添加用户界面	
deleteUser	删除用户按钮	点击后进入删除用户界面	
selfInfo	修改信息按钮	点击后进入信息修改界面	

3.5 SortGradeFrame 界面

表 3-5 SortGradeFrame 界面

界面编号	005	部件名称	SortGradeFrame
界面性质	成绩统计界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
idt	课程号输入框	输入课程编号	
passt	及格分数线输入框	输入及格分数线	
goodt	良好分数线输入框	输入良好分数线	
excellentt	优秀分数线输入框	输入优秀分数线	
submit	提交按钮	点击后统计成绩并显示结果	

3.6 Info 界面

表 3-6 Info 界面

界面编号	006	部件名称	Info
界面性质	用户信息查询界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			

参数名	内容	说明
idLabel	账号标签	显示用户账号
nameLabel	姓名标签	显示用户姓名
genderLabel	性别标签	显示用户性别
birLabel	生日标签	显示用户生日
insLabel	学院标签	显示用户学院
majorLabel	专业标签	显示用户专业

3.7 GradeInfo 界面

表 3-7 GradeInfo 界面

界面编号	007	部件名称	GradeInfo
界面性质	成绩查询界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
list	成绩列表	显示学生的所有成绩	

3.8 CourseView 界面

表 3-8 CourseView 界面

界面编号	008	部件名称	CourseView
界面性质	课程查询界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
list	成绩列表	显示学生的所有成绩	

3.9 EditInfo 界面

表 3-9 EditInfo 界面

界面编号	009	部件名称	EditInfo
界面性质	信息修改界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
namet	姓名输入框	输入用户姓名	

instt	学院输入框	输入用户学院
birtht	生日输入框	输入用户生日
pass1t	新密码输入框	输入新密码
pass2t	确认密码输入框	确认新密码
majort	专业输入框	输入用户专业
submit	提交按钮	点击后保存修改后的信息

3.10 AddUser 界面

表 3-10 AddUser 界面

界面编号	010	部件名称	AddUser
界面性质	添加用户界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
idt	编号输入框	输入用户编号	
namet	姓名输入框	输入用户姓名	
birthdayt	生日输入框	输入用户生日	
institutet	学院输入框	输入用户学院	
majort	专业输入框	输入用户专业	
chooice	用户类型选择框	选择用户类型（学生、教师、管理员）	
submit	提交按钮	点击后添加新用户	

3.11 DeleteUser 界面

图 3-11 DeleteUser 界面

界面编号	011	部件名称	DeleteUser
界面性质	删除用户界面	界面介质	图形用户界面（GUI）
表示形式：窗口（JFrame）			
界面参数			
参数名	内容	说明	
idt	编号输入框	输入用户编号	
chooice	用户类型选择框	选择用户类型（学生、教师、管理员）	
submit	提交按钮	点击后删除用户	

附录 D 软件测试报告文档模板

1. 引言

本报告描述了对成绩管理系统的软件测试结果。成绩管理系统是一个用于管理学生成绩的软件应用程序。测试工作的目的是验证系统的功能和性能是否达到预期的要求，并识别潜在的缺陷和问题。

1.1 编写目的

本软件测试用例报告主要帮助预期读者对象，熟悉和了解这份文档所编写的目的、存在风险及产品范围。

1.2 项目背景

成绩管理系统的开发是为了解决以前通过手工来记录和管理学生成绩的繁琐和低效问题、以及过去学校通过纸质档案记录学生成绩所出现的数据丢失、错误和难以统计分析的情况。同时也让学生能快速查询自己的成绩。

为了提高学校成绩管理的效率和准确性，学校成绩管理系统应运而生。该系统利用 JAVA 程序，以电子化的方式存储和管理学生成绩数据。通过该系统，老师、管理员可以轻松记录、查询和更新学生成绩。

1.3 系统简介

成绩管理系统系统为教学类应用，多用于学校的官方平台，来帮助学校进行课程信息和成绩信息的管理，也便于学生和老师等个体用户有效地查询信息。

系统的主要功能有：信息查询、成绩查询、课程查询、修改信息、开课、录入成绩、增加用户、删除用户、分段统计。

1.4 术语和缩写词

缩写、术语	解释
黑盒测试	在完全不考虑程序内部结构和内部特性的情况下，在程序接口进行测试，
白盒测试	清楚盒子内部的东西以及里面是如何运作的情况下进行程序测试
等价类划分	将程序所有可能的输入数据（有效的和无效的）划分成若干个等价类。

1.5 读者对象

- **开发人员：**开发人员需要阅读测试报告来了解系统在各个方面的测试结果和问题，

以便进行修复和改进。

- **项目经理：**项目经理需要了解软件测试的情况，以评估项目的进展和风险，并做出相应的决策，如是否发布软件、是否进行修复和改进等。
- **客户：**客户可以根据测试报告来评估软件是否满足他们的需求和期望，并提供反馈和建议。

1.6 参考资料

[1] 成绩管理系统说明文档-方峻版

[2] 参考：软件项目报告样例文档-成电微记（修复）

[3] 软件项目报告模板（2024 版）

2. 单元测试

2.1 被测对象（单元）的介绍

- CheckInfo 类位于 controller 包中，主要负责验证用户（学生、教师、管理员）的登录信息。该类通过读取存储在文件中的用户信息，验证用户输入的 ID 和密码是否正确，并返回相应的验证结果

2.2 测试范围与目的

- 测试范围：本次单元测试的范围仅限于 controller 包中的 CheckInfo 类，具体测试对象为 CheckInfo 类的 isMember 方法。测试范围包括以下内容：
 - 功能测试：验证 isMember 方法能够正确比较用户 ID 和密码，并返回相应的验证结果。
 - 路径覆盖：通过基本路径测试法，覆盖 isMember 方法的所有可能路径，包括：
 - 用户 ID 和密码匹配。
 - 用户 ID 存在但密码不匹配。
 - 用户 ID 不存在，继续读取。
 - ID 不存在，无匹配的 ID
 - 边界条件：验证 isMember 方法在处理边界条件（如空输入、空文件等）时的行为是否符合预期。
- 测试目的
本次单元测试的主要目的是验证 CheckInfo 类的 isMember 方法的功能正确性和鲁棒性，确保该方法能够正确处理用户登录信息的验证，并满足以下目标：
 - 功能正确性：验证 isMember 方法能够根据用户类型、用户 ID 和密码，正确返回验证结果。确保方法能够正确处理用户 ID 和密码的匹配情况，返回 1（验证成功）、2（用户存在但密码错误）或 0（用户不存在）。
 - 路径覆盖：通过基本路径测试法，确保 isMember 方法的所有可能路径都被覆盖，包括正常路径和异常路径。确保方法在不同输入条件下的行为符合预期。
 - 异常处理：验证 isMember 方法在文件不存在或文件格式错误时，能够正确处理异常情况，避免系统崩溃或返回错误结果。确保方法在异常情况下能够返回合理的

默认值或抛出适当的异常。

2.3 测试环境与测试辅助工具的描述

- 测试环境：Win11、Eclipse

2.4 基本路径覆盖法设计单元测试用例

结合程序已有代码，得到的流程图如下所示：

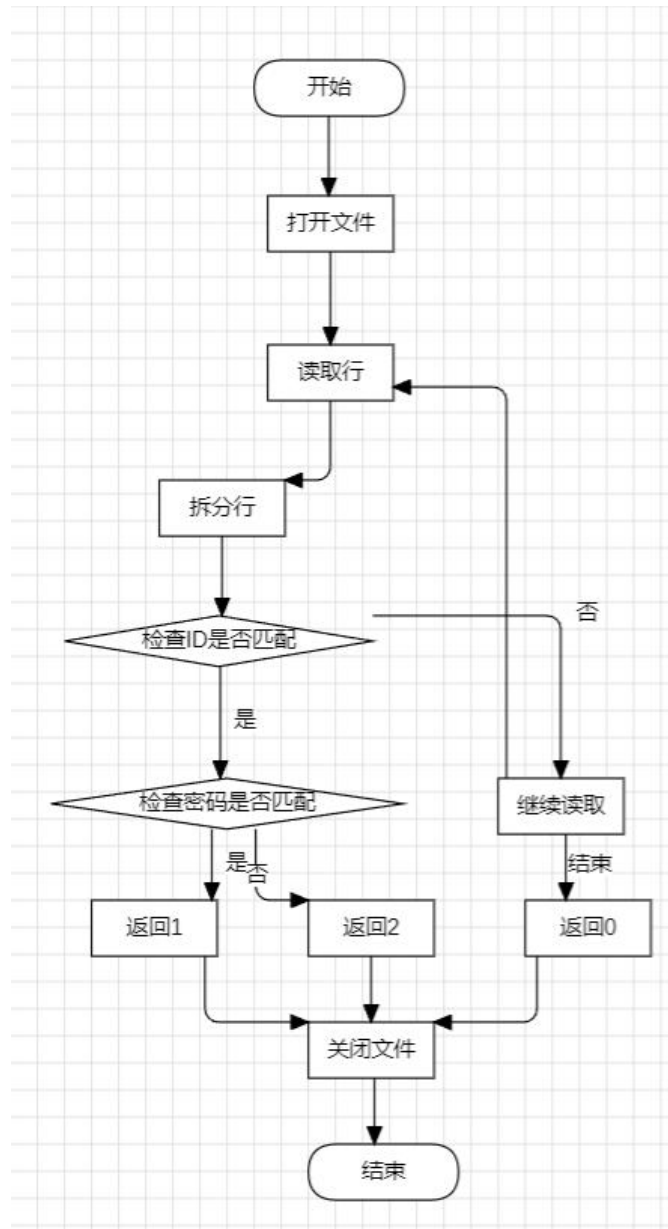


图 2-1 单元测试流程图

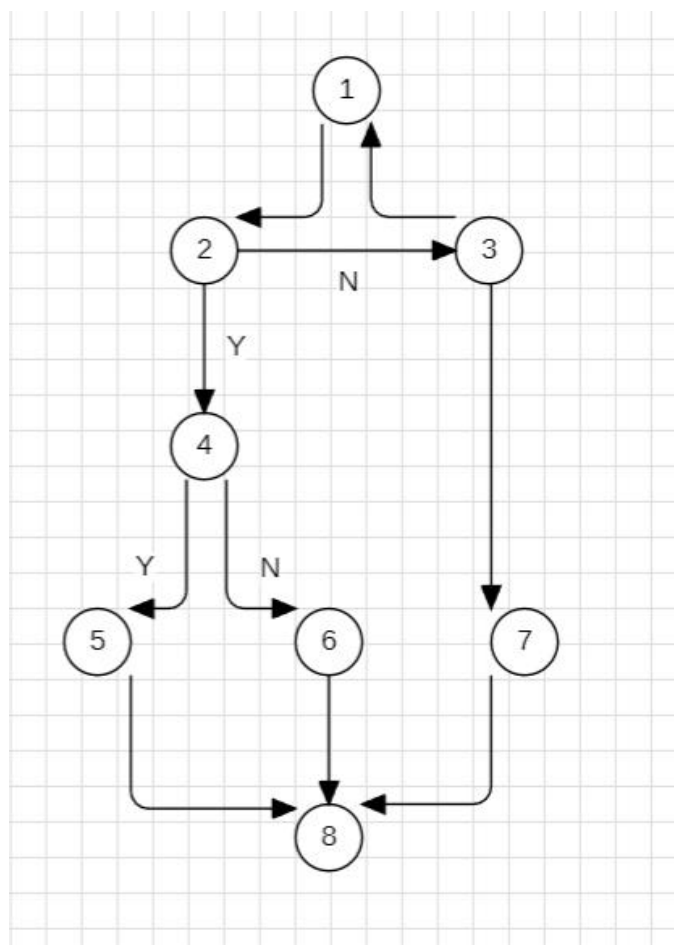


图 2-2 单元测试流图

通过计算，得到环路复杂度 $V(G)=E(10)-N(8)+2=4$ ，对应的基本路径覆盖测试用例需要至少 4 组，如下表所示：

表 2-1 基本路径测试用例设计

	第 1 组	第 2 组	第 3 组	第 4 组
用例描述	ID 存在， 密码匹配	ID 存在， 密码不匹配	ID 不存在， 继续读取	ID 不存在， 无匹配的 ID
用例设置	ID:a PWD:123456	ID:a PWD:02468	ID:b PWD:123456	ID:c PWD:13579
预期结果	登录成功，进入 对应用户界面	登录失败， 显示错误信息	登录失败， 显示错误信息	登录失败， 显示错误信息

2.5 测试记录和分析

- 对于第 1 组用例，输入界面和登陆后显示界面如下图：

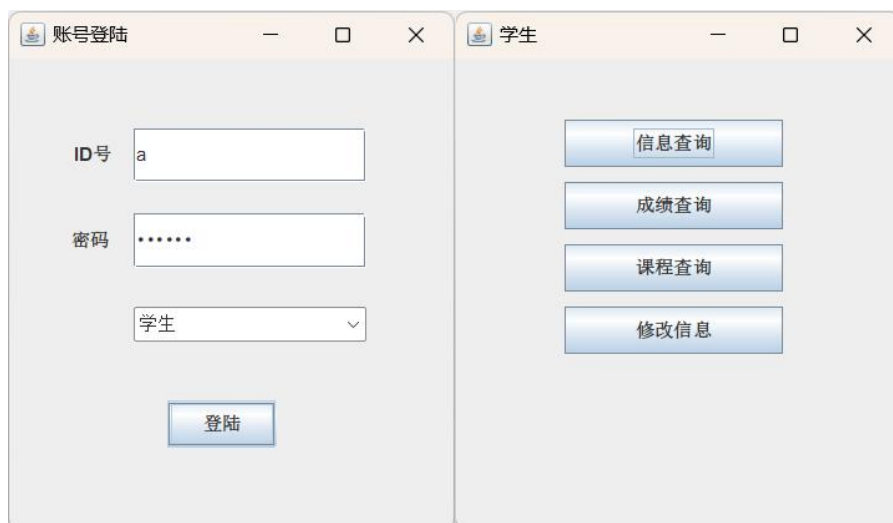


图 2-3 测试用例一结果

- 对于第 2 组用例，输入界面和登陆后显示界面如下图：

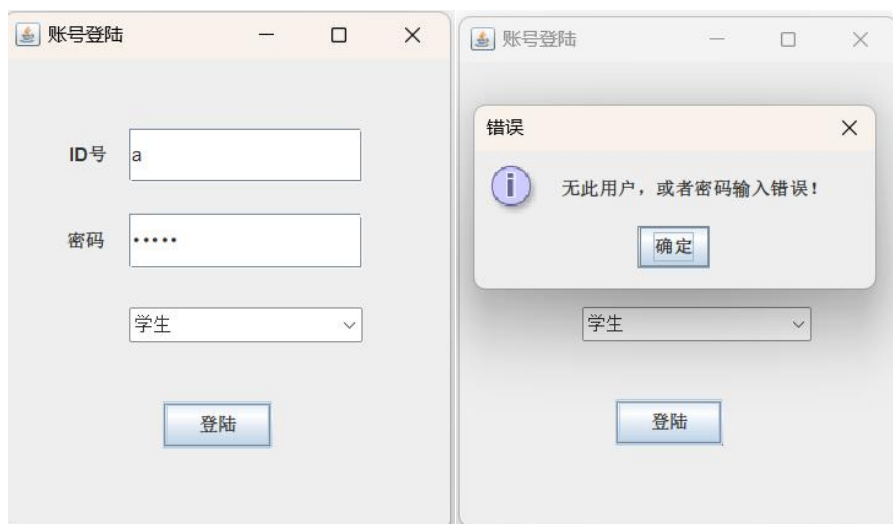


图 2-4 测试用例二结果

- 对于第 3 组用例，输入界面和登陆后显示界面如下图：

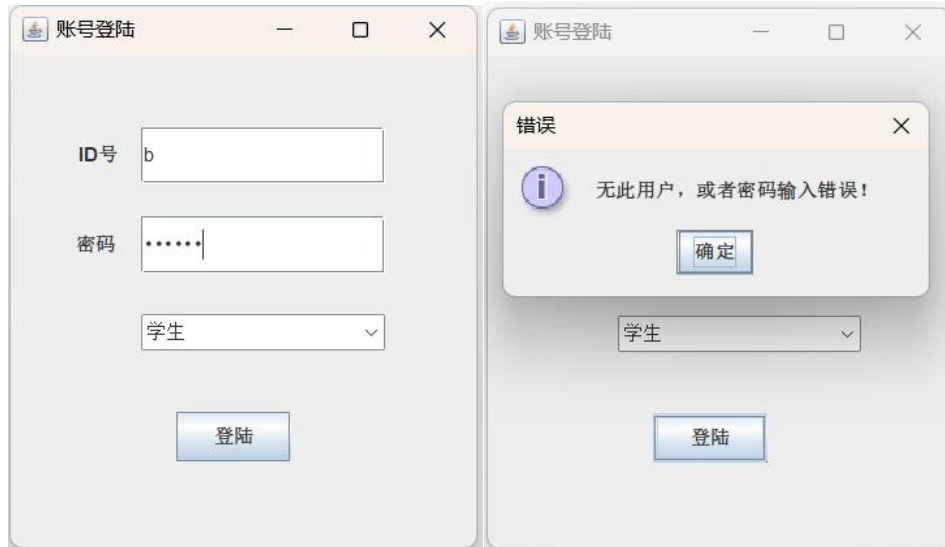


图 2-5 测试用例三结果

- 对于第 4 组用例，输入界面和登陆后显示界面如下图：

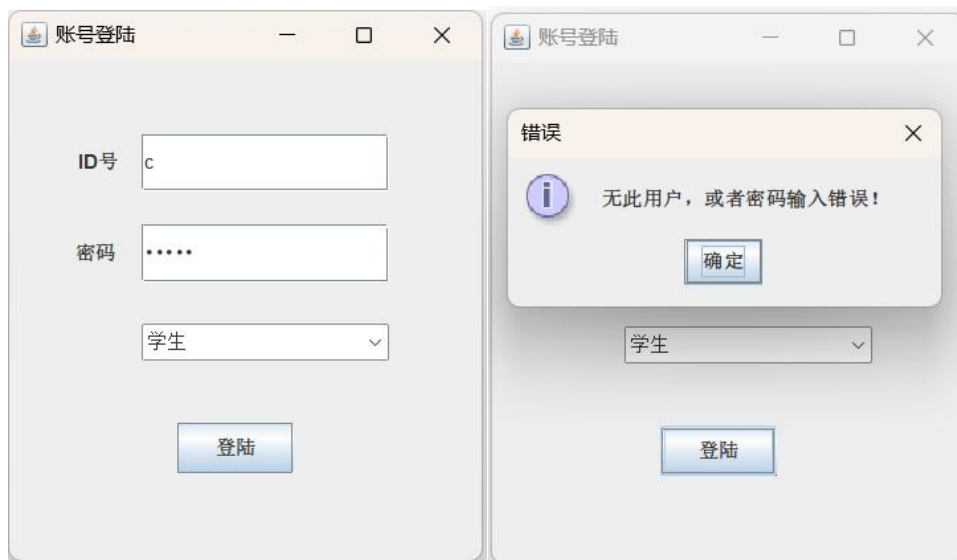


图 2-6 测试用例四结果

3. 集成测试

3.1 被测试对象的介绍

- 包概述

model 包是系统的核心数据模型包，包含了系统中主要的数据库实体类。这些类用于表示系统中的用户、课程、成绩等核心数据，并提供了对这些数据的操作和管理功能。

- 被测试类

User 类：抽象类，表示系统中的用户，包含用户的基本信息（ID、密码、姓名）。

Student 类：继承自 User 类，表示学生用户，包含学生的性别、生日、学院、专业等

信息。

Teacher 类：继承自 User 类，表示教师用户，包含教师的性别、生日、学院、专业等信息。

Administrator 类：继承自 User 类，表示管理员用户，包含管理员的性别、生日、学院、专业等信息。

Course 类：表示课程信息，包含课程号、课程名、教师 ID、教师姓名、学分、学时等信息，并提供对课程的操作方法。

3.2 测试范围与目的

● 测试范围

本次集成测试的范围是 `model` 包中的类，主要测试这些类之间的交互和数据传递。具体测试范围包括以下内容：

◆ **类继承关系：**验证 Student、Teacher、Administrator 类是否正确继承自 User 类。确保子类能够正确访问和修改父类的属性。

◆ **数据一致性：**验证 Course 类中的课程信息是否能够正确存储和检索。确保数据在不同类之间的传递和处理是准确的。

◆ **方法调用：**验证 Course 类中的方法（如 `hasCourse`、`sortGrade`、`isValidate`）是否能够正确调用并返回预期结果。确保方法在不同类之间的调用是准确的。

◆ **异常处理：**验证 Course 类在处理异常情况（如成绩标准不合法）时的行为是否符合预期。确保类在异常情况下能够返回合理的默认值或抛出适当的异常。

◆ **类间交互：**验证 `model` 包中的类之间的交互是否正确，确保数据在不同类之间的传递和处理是准确的。确保类之间的依赖关系和数据传递是合理的。

● 测试目的

本次集成测试的主要目的是验证 `model` 包中的类之间的交互和数据传递是否正确，确保这些类能够协同工作，并满足以下目标：

◆ **功能正确性：**验证 `model` 包中的类能够正确实现其功能，确保数据在不同类之间的传递和处理是准确的。确保类之间的交互符合需求规格说明书和设计文档的要求。

◆ **数据一致性：**验证 `model` 包中的类能够正确存储和检索数据，确保数据的一致性。确保数据在不同类之间的传递和处理是准确的。

◆ **方法调用：**验证 `model` 包中的类中的方法能够正确调用并返回预期结果。确保方法在不同类之间的调用是准确的。

◆ **异常处理：**验证 `model` 包中的类在处理异常情况时的行为是否符合预期。确保类在异常情况下能够返回合理的默认值或抛出适当的异常。

◆ **类间交互：**验证 `model` 包中的类之间的交互是否正确，确保数据在不同类之间的传递和处理是准确的。确保类之间的依赖关系和数据传递是合理的。

◆ **代码质量：**通过集成测试，发现并修复 `model` 包中的潜在缺陷，提高代码的质量和可靠性。确保类的实现符合编码规范和最佳实践。

3.3 测试环境与测试辅助工具的描述

测试环境：Win11、Eclipse

辅助工具：Junit4, 使用 junit 编写测试脚本在 Eclipse 中运行检测判断测试用例是否达到预期结果

3.4 集成策略与驱动/桩测试模块的设计

3.4.1 User 类属性设置和获取测试

- ◆ **测试目的：**测试 User 类构造方法 User() 能否正确设置 id、pwd、name 属性；
- ◆ **测试步骤：**创建继承 User 类的 ConcreteUser 子类（抽象类 User 无法直接实例化）；实例化 ConcreteUser 类，利用断言判断属性是否正确设置。
- ◆ **测试脚本：**如下图所示：

```

1 package model;
2
3 import static org.junit.Assert.*;
4 import org.junit.Test;
5
6 public class UserTest {
7
8     @Test
9     public void testUserAttributes() {
10         ConcreteUser user = new ConcreteUser("001", "password", "John Doe");
11         assertEquals("001", user.getId()); // 验证用户ID是否为"001"
12         assertEquals("password", user.getPwd()); // 验证用户密码是否为"password"
13         assertEquals("John Doe", user.getName()); // 验证用户名是否为"John Doe"
14     }
15 }
16

```

图 3-1 Junit 测试脚本 UserTest.java

- ◆ **预期结果及分析：**若无异常抛出，构造方法 User() 属性设置测试通过；若抛出异常，测试不通过。

3.4.2 User 类子类属性和继承测试

- ◆ **测试目的：**测试 User 类的子类 Administrator、Student、Teacher 是否正确继承了 User 的属性和方法，是否添加了自己的特定属性和方法。
- ◆ **测试步骤：**对应 Administrator、Student、Teacher 三个子类编写测试脚本，利用断言判断属性是否正确设置。
- ◆ **测试脚本：**片段如下图所示：

```

1 package model;
2
3 import static org.junit.Assert.*;
4
5
6 public class UserSubclassTest {
7
8     public void testAdministrator() {}
9
10    public void testStudent() {}
11
12    public void testTeacher() {}
13 }
14

```

图 3-2 Junit 测试脚本 UserSubclassTest.java

3.4.3 加入 Course 类的 Model 包集成测试

- ◆ **测试目的：**测试 User 类以及子类 Administrator、Student、Teacher 的方法和属性；测试 Course 类的构造方法及属性。
- ◆ **测试步骤：**对应 User 类、Administrator 类、Student 类、Teacher 类以及 Course 类编写测试脚本，利用断言判断代码是否正确。
- ◆ **测试脚本：**片段如下图所示：

```
1 package model;
2
3 import static org.junit.Assert.*;
4
5
6
7 public class ModelIntegrationTest {
8
9
10     public void testAdministrator() {}
11
12
13
14
15
16
17
18
19
20
21
22     public void testStudent() {}
23
24
25
26
27
28
29
30
31
32
33
34     public void testTeacher() {}
35
36
37
38
39
40
41
42
43
44
45
46     public void testCourseConstruction() {}
47
48
49
50
51
52
53
54
55
56
57
58 }
59
```

图 3-3 Junit 测试脚本 ModelIntegrationTest.java

3.5 测试记录和分析

3.5.1 User 类属性设置和获取测试

对于 User 类属性以及相关方法的测试结果如下，测试用例通过，表示 User 类相关属性和方法无误，在调用时能够正常执行初始化。



图 3-4 User 类的测试脚本运行结果

3.5.2 User 类子类属性和继承测试

对于 User 类的子类属性以及相关方法的测试结果如下，测试用例通过，表示子类 Administrator、Student、Teacher 相关属性和方法无误，正确继承了相关属性和方法，在调用时能够正常执行初始化。

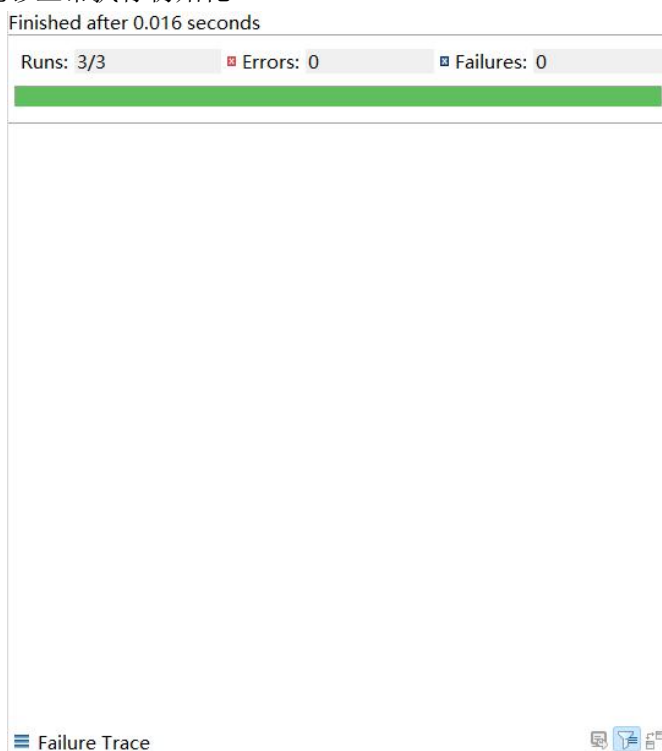


图 3-5 User 类子类的测试脚本运行结果

3.5.3 加入 Course 类的 Model 包集成测试

整个 Model 包的集成测试结果如图所示，测试用例通过，表示包中类 User、Administrator、Student、Teacher、Course 相关属性和方法无误，相互之间有正确的继承和被继承关系，在调用时能够正常执行初始化。

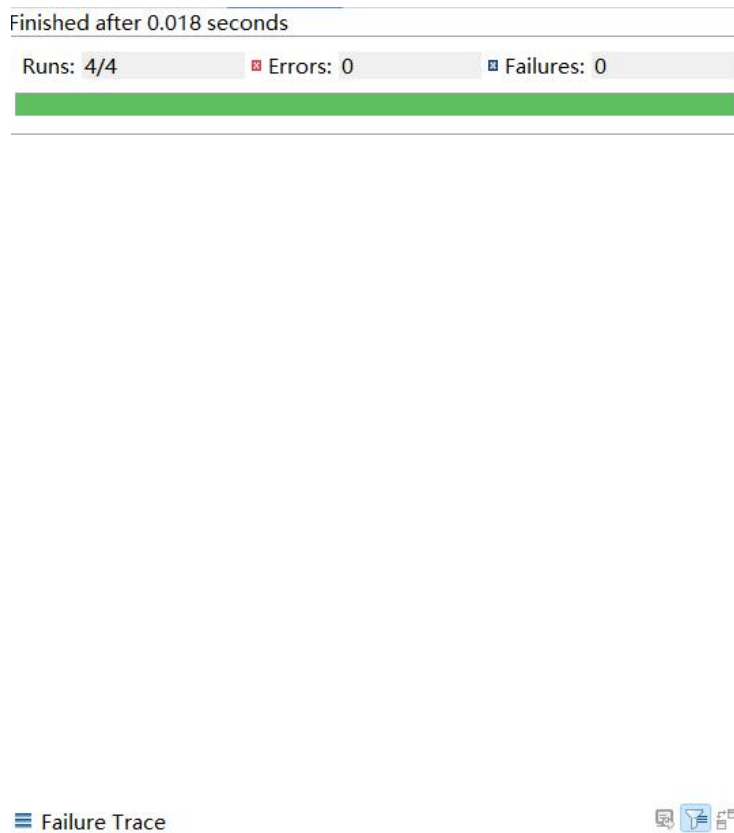


图 3-6 Model 包中所有类的集成测试脚本运行结果

4. 验收测试

4.1 被测试对象的介绍

4.1.1 角色概述

- ◆ 教师角色是学校管理系统中的一个重要用户角色，主要负责课程管理、学生成绩录入、个人信息维护等操作。教师通过系统可以方便地管理所教授的课程，录入和查询学生成绩，并维护自己的个人信息。

4.1.2 主要功能

- ◆ **登录系统：**教师使用唯一的教师 ID 和密码登录系统。
- ◆ **查询个人信息：**教师可以查看自己的个人信息，包括姓名、性别、生日、学院、专业等。
- ◆ **修改个人信息：**教师可以修改自己的个人信息，如姓名、性别、生日、学院、专业等。
- ◆ **增加课程：**教师可以添加新的课程，包括课程号、课程名、学分、学时等信息。
- ◆ **录入学生成绩：**教师可以为所教授课程的学生录入成绩。
- ◆ **查询所教授的课程：**教师可以查看自己所教授的所有课程信息。
- ◆ **统计课程成绩：**教师可以统计所教授课程的学生成绩分布情况，如不及格、

及格、良好、优秀的人数。

4.2 测试范围与目的

对上述功能模块的验证

4.3 测试环境与测试辅助工具的描述

Win11、Eclipse

4.4 黑盒测试策略设计验收测试用例

4.4.1 等价类划分

■ 登录系统等价类划分：

- ◆ 有效输入：正确的教师 ID 和密码
- ◆ 无效输入：错误的教师 ID 或密码

测试用例：见表 4-1

表 4-1 登录系统等价类划分

测试用例	对应等价类	预期结果
ID: t 密码: 123456	正确的教师 ID 和密码	登陆成功，显示登录后功能界面
ID: te 密码: 123456	登录失败	提示“无此用户或密码错误”
ID: te 密码: 654321	登录失败	提示“无此用户或密码错误”

■ 修改个人信息等价类划分：

- ◆ 有效输入：合法的个人信息（如姓名、性别、生日等）
- ◆ 无效输入：非法的个人信息（如生日格式错误、性别为空等）

测试用例：见表 4-2

表 4-2 修改个人信息等价类划分

测试用例	对应等价类	预期结果
Tom male 2020.1.1 computer software	合法的个人信息	成功修改
male 2020.1.1 computer software	非法的个人信息	提示“信息不能为空”
Tom male computer software	非法的个人信息	提示“信息不能为空”
Tom male 2020.1.1 software	非法的个人信息	提示“信息不能为空”
Tom male 2020.1.1 computer	非法的个人信息	提示“信息不能为空”

■ 增加课程等价类划分：

- ◆ 有效输入：合法的课程信息（如课程号、课程名、学分、学时等）
- ◆ 无效输入：非法的课程信息（如课程号重复、学分为负数等）

测试用例：见表 4-3

表 4-3 增加课程等价类划分

测试用例	对应等价类	预期结果
3 C++ 2 48 t Tom	合法的课程信息	成功增加课程
1 C++ 2 48 t Tom	非法的课程信息	提示“此课程已经存在”

■ 录入学生成绩等价类划分：

- ◆ 有效输入：合法的学生成绩
- ◆ 无效输入：非法的学生成绩

测试用例：见表 4-4

表 4-4 录入学生成绩等价类划分

测试用例	对应等价类	预期结果
课程号：1 学号：a 成绩：66 姓名：jack	合法的学生成绩	成功录入
课程号：3 学号：a 成绩：66 姓名：jack	非法的学生成绩	提示“无此学生”

■ 统计课程成绩等价类划分：

- ◆ 有效输入：合法的成绩标准（如及格、良好、优秀）
- ◆ 无效输入：非法的成绩标准

测试用例：见表 4-5

表 4-5 统计课程成绩等价类划分

输入条件	有效等价类	无效等价类
及格分数线范围	$0 \leq \text{及格分数线} < \text{良好分数线}$ (1)	$0 > \text{及格分数线}$ (2)， $\text{及格分数线} > \text{良好分数线}$ (3)
良好分数线范围	$\text{及格分数线} < \text{良好分数线} < \text{优秀分数线}$ (4)	$\text{良好分数线} > \text{优秀分数线}$ (5)
优秀分数线范围	$\text{良好分数线} < \text{优秀分数线}$	$\text{优秀分数线} > 100$ (7)

	<=100 (6)	
--	-----------	--

4.4.2 边界值分析

■ 统计课程成绩边界值分析:

及格分数线: 上点: 0 和良好分数线的值; 内点: 0 和良好分数线之间的任何一点; 离点: -1, 良好分数线的值-1, 测试用例为: {-1, 0, 0 到良好分数线之间的值, 良好分数线的值-1, 良好分数线}

良好分数线: 上点: 及格分数线的值和优秀分数线的值; 内点: 及格分数线的值和优秀分数线的值之间的任何一点; 离点: 及格分数线的值+1, 优秀分数线的值-1, 测试用例为: {及格分数线, 及格分数线+1, 及格分数线和优秀分数线之间的值, 优秀分数线-1, 优秀分数线}

优秀分数线: 上点: 良好分数线的值和 100; 内点: 良好分数线的值和 100 之间的任何一点; 离点: 良好分数线的值+1, 101, 测试用例为: {良好分数线的值, 良好分数线的值+1, 良好分数线和 100 之间的值, 100, 101}

测试用例: 见表 4-6

表 4-6 统计课程成绩边界值分析

用例	及格分数线	良好分数线	优秀分数线	预期输出
1	60	80	90	符合要求 (覆盖(1)(4)(6))
2	-2	80	90	错误提示 (覆盖(2))
3	80	60	90	错误提示 (覆盖(3))
4	60	90	80	错误提示 (覆盖(5))
5	60	80	120	错误提示 (覆盖(7))
6	-1	80	90	错误提示
7	0	80	90	符合要求
8	79	80	90	符合要求
9	80	80	90	错误提示
10	60	60	90	错误提示
11	60	61	90	符合要求
12	60	89	90	符合要求
13	60	90	90	错误提示
14	60	80	80	错误提示
15	60	80	81	符合要求
16	60	80	100	符合要求
17	60	80	101	错误要求

4.5 测试记录和分析

4.5.1 登陆系统的等价类划分

◆ 对于第 1 组用例，输入界面和登陆后显示界面如下图：

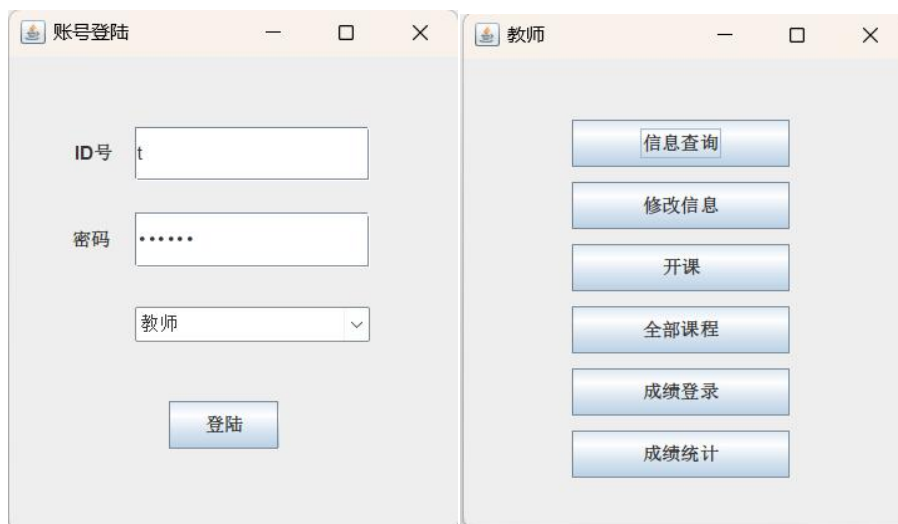


图 4-1 登录系统等价类划分用例一测试结果

◆ 对于第 2 组用例，输入界面和登陆后显示界面如下图：

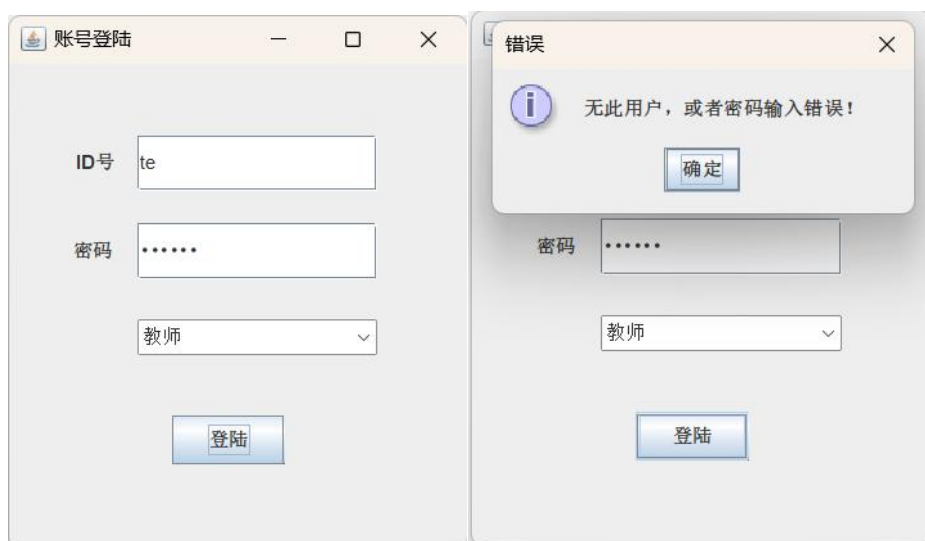


图 4-2 登录系统等价类划分用例二测试结果

◆ 对于第 3 组用例，输入界面和登陆后显示界面如下图：

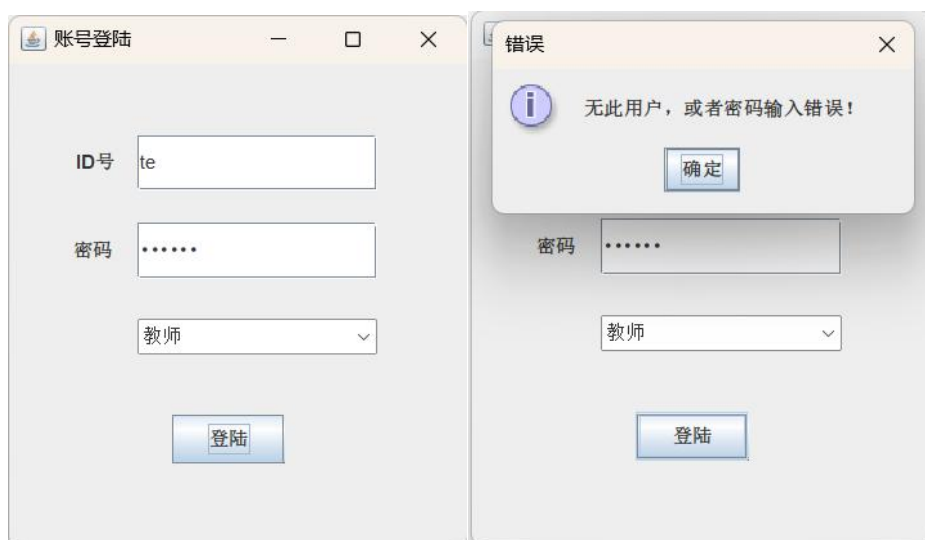


图 4-3 登录系统等价类划分用例三测试结果

4.5.2 修改信息的等价类划分

- ◆ 对于用例一，测试结果如下图，表示修改成功

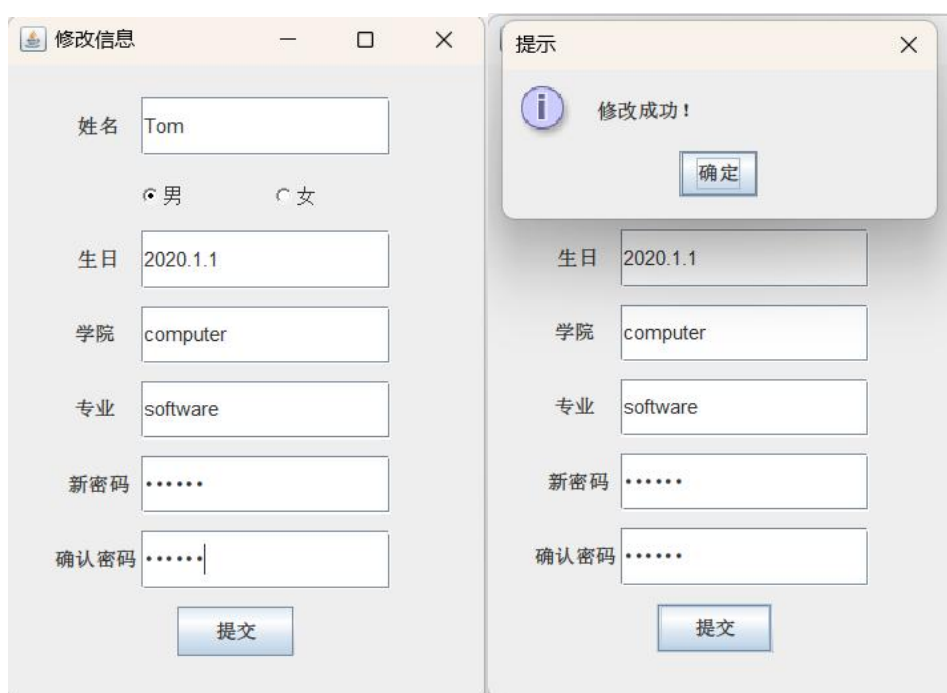


图 4-4 修改信息等价类划分用例一测试结果

- ◆ 对于用例二、三、四五，测试结果如下图，表示修改失败



图 4-5 修改信息等价类划分用例报错结果

4.5.3 增加课程的等价类划分

- ◆ 对于用例一，测试结果如下图，表示开课成功

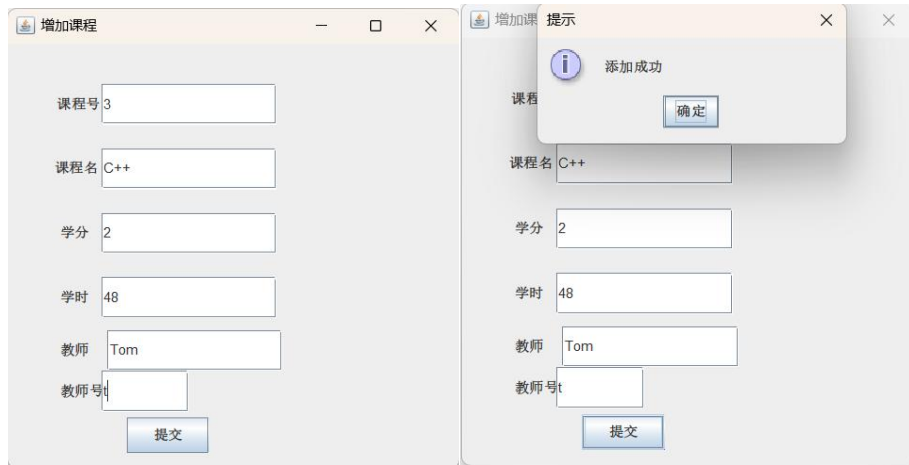


图 4-6 增加课程等价类划分用例一测试结果

- ◆ 对于用例二，测试结果如下图，表示开课失败

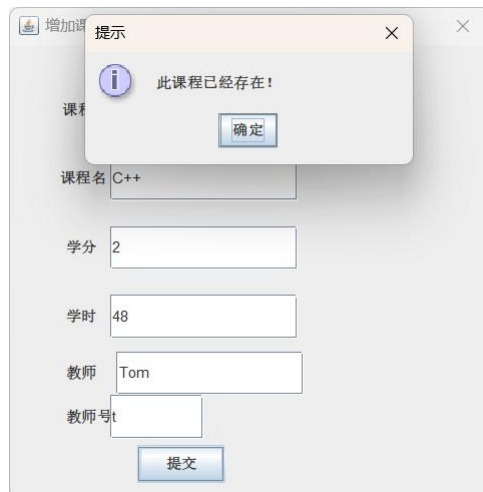


图 4-7 增加课程等价类划分用例二测试结果

4.5.4 录入学生成绩等价类划分

- ◆ 对于用例一，测试结果如下图，表示成绩录入成功

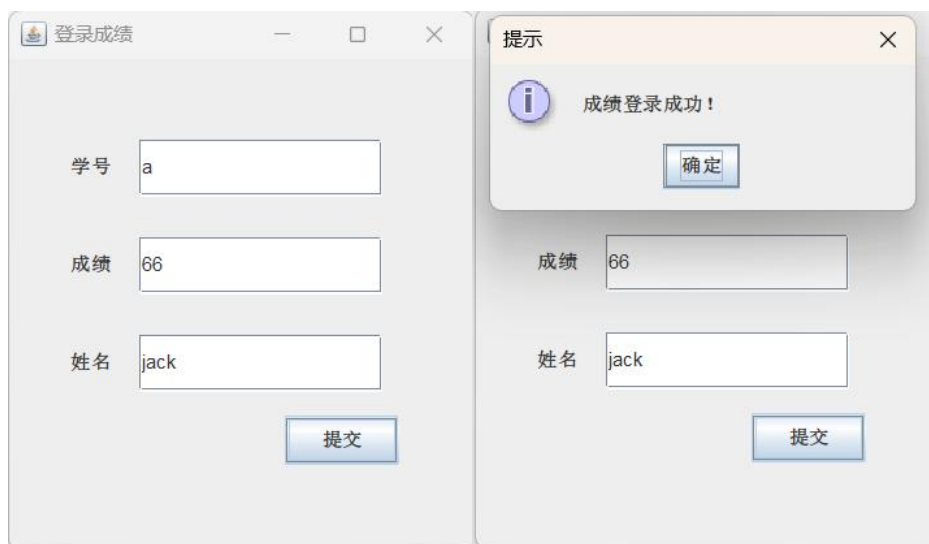


图 4-8 成绩录入等价类划分用例一测试结果

- ◆ 对于用例二，测试结果如下图，表示成绩录入失败



图 4-9 成绩录入等价类划分用例二测试结果

4.5.5 统计课程成绩等价类划分

- ◆ 对于用例 1、7、8、11、12、15、16，进入如下图的“成绩统计结果”界面，表示课程成绩统计成功

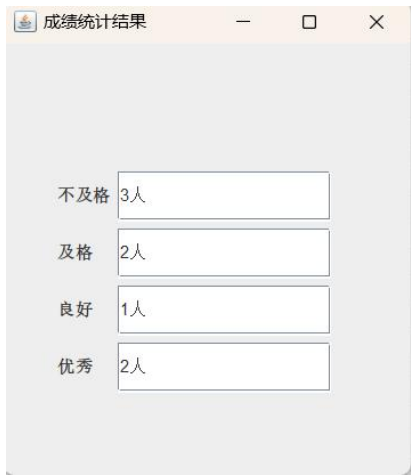


图 4-10 统计课程成绩边界值划分用例正确结果

- ◆ 对于用例 2、3、4、5、6、9、10、13、14、17，出现提示“成绩标准设置错误”，表示课程成绩统计失败



图 4-11 统计课程成绩边界值划分用例错误结果

5 系统中存在的 BUG

■ 教师增设课程中的数据保存

在教师填写正确开课信息并成功开后，数据保存格式与已有课程不一致，如下图增设课程 5 后数据顺序与前 4 门出现冲突

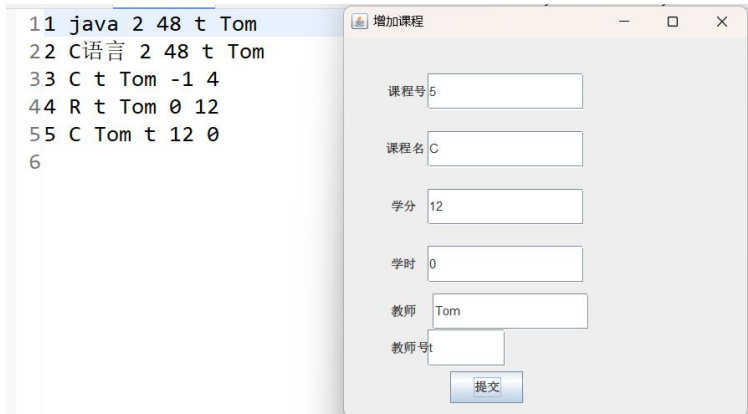


图 5-1 开设课程中数据保存 BUG

■ 学生修改信息中的数据保存

在学生填写正确个人信息并成功修改系列为 software se 后，再次查询的个人信息与保存信息不一致，如下图查询后系列与保存数据出现冲突

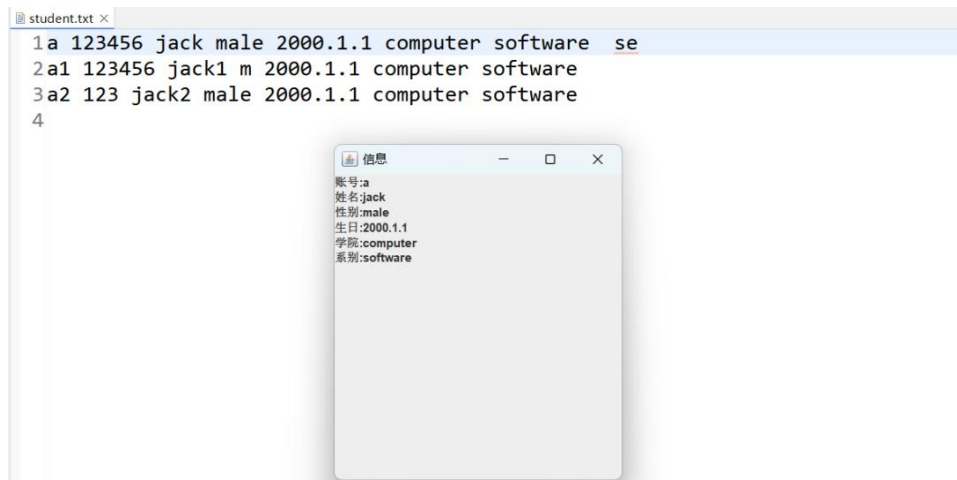


图 5-2 修改信息中数据保存 BUG