



南昌航空大学

个人实训报告

学 院： 软 件 学 院

专 业： 软件工程 班 级： 232018

学 号： 23201837 姓 名： 李茂田

实训题目： 学生信息管理系统

实训时间： 2024-7-5 ~ 2022-7-25

实训地点： 软件学院实训中心 D304

指导教师： 贺桑迪

软件实训任务书

实训项目选题		学生信息管理系统	
项目组长	李茂田	项目组成员	熊锋, 邢沂炜
实训具体要求： 一、项目目标 系统支持教师进行成绩录入、修改、删除和查询操作，确保数据的准确性和及时性。同时，学生可以通过系统方便地查询自己的考试成绩，帮助他们更好地了解和评估自己的学习表现。建立一个 C / S 系统，能够根据不同角色的权限，分别为学生、教师和教务处管理员提供功能，包括学生信息查询、发表留言、学生成绩录入、考试成绩查询以及学籍调整等操作。			
二、项目功能 管理员：进行所有功能操作 教师：进行成绩录入、修改和删除操作 学生：进行查询自己考试成绩操作			
三、项目要求 遵循 MVC 框架，统一 JDK1.8，使用 Swing 图形化界面，各成员分头行动，最后使用 IDEA 的 code with me 进行代码整合。 首要任务是实现学生信息管理、课程安排和成绩管理的基础功能。这些功能是系统的核心。特别地，我们对系统的安全性给予了高度重视，尤其是针对 SQL 注入等常见安全漏洞。如果项目进度允许，我们将着手开发自动排课功能。			
实训进度计划			
序号	起止日期	工 作 内 容	
1	7-5~7-7	基础知识补强：包括 Java 的基本语法、关键字和语法规则，数据类型，流程控制，异常处理 try-catch 块、throw 和 throws 关键字的使用和字符串操作 统一 Java 编程工具和开发环境：采用 JDK 8，并利用 Swing 进行开发。学习基础的 Java Swing 容器，如 JFrame 和 JPanel 等窗口的代码实现，包括它们的创建、布局和事件处理的具体编写方式	

2	7-8~7-10	学生管理系统调研：对学生管理系统进行基本需求分析，如学生信息管理、课程安排、成绩管理等功能；分析市场上已有的类似系统的功能和特点分析。考虑系统的扩展性和可维护性，以及开发和维护的成本评估，根据前人的开发经验得出可行性方案和大概的软件构型。
3	7-11~7-12	探究学生信息管理系统的需求规格：确立系统功能和非功能性需求，以及用户需求。系统功能要求包括学生信息管理、课程管理、成绩管理和课程安排。非功能性需求关注系统性能、安全性、可靠性、易用性和扩展性。用户需求考虑用户角色和权限设置和界面设计偏好。需求规格还需考虑法律合规性、系统集成和数据保护。
4	7-13~7-14	设计学生信息管理系统：画流程图和 E-R 图，帮助明确系统的工作流程和数据结构。流程图展示了用户操作的顺序和系统处理的逻辑，确保每个功能模块的流程清晰、合理。流程图中应详细标注每个步骤的输入输出、决策点和处理环节，以便于开发人员理解和实现。E-R 图用于描述系统中数据实体之间的关系。在学生信息管理系统中，E-R 图明确学生、课程、成绩等实体，以及它们之间的关系，如学生选课、成绩录入等。 通过详细明了的图表帮助我们进行下一步开发。
5	7-15~7-19	进行实际编码:遵循 MVC 框架，各成员分头行动，最后使用 IDEA 的 code with me 进行代码整合 中期评估，导师提出修改意见，团队随后进行软件调试，增加新的功能
6	7-20~7-21	进行已有代码的软件测试，适当放弃部分无法实现的功能，撰写测试报告
7	7-22~7-24	软件最终测试，撰写个人实训报告和团队实训报告，准备答辩
项目参考资料		
1. IceHowe. "Student Management System." GitHub, n.d., https://github.com/IceHowe/student_management_system . 2. Liujiang240556. " StudentGradeMangerSystem " GitHub, n.d., https://github.com/Liujiang240556/StudentGradeMangerSystem 3. 张明东, 戴丹丹. "Java EE 下学生信息管理系统的设计与实现." 集宁师范学院, n.d. Web. [2024-7-9]. 4. 李志学. "基于 C/S 架构的潍坊科技学院学生信息管理系统." 中国海洋大学, n.d. Web. [2024-7-9].		

目录

1 项目概述	2
1.1 项目背景及意义	2
1.2 项目设计目标	2
1.3 项目可行性分析	2
2 承担任务	3
2.1 小组介绍	3
2.2 个人承担任务	3
3 负责的需求模块	4
3.1 功能需求	4
3.2 软件的总体结构	5
3.3 个人承担概要设计	5
4 负责的详细设计模块	6
4.1 用户登录注册模块详细以及主框架设计	6
4.2 成绩管理模块详细设计	9
4.3 操作日记模块详细设计	10
5 负责的编码	11
5.1 用户登录注册和主框架模块核心代码	11
5.2 成绩管理模块核心代码	12
5.3 操作日记模块核心代码	14
6 考虑社会，健康，安全，法律等因素	14
7 实训总结	15
7.1 项目工作总结	15
7.2 项目设计成果总结	15
7.3 存在的不足及下一步工作	15
7.4 个人在项目开发中的主要工作成果及不足之处	16
8 参考文献	16
附录	17
附录 1	17

1 项目概述

1.1 项目背景及意义

项目背景：随着社会不断发展，高校学生日常管理涉及的各类数据信息越来越多，随之而来的管理水平也需要得到不断提升。与使用传统的人工手段完成学生日常管理任务和数据记录相比，暴露出来的问题越来越明显：例如工作效率低、容易重复出错、信息检索速度慢，以及教师不能充分掌握班级或学生的平均学习活动情况等，不利于监督学校正常的日常教学管理工作。

项目意义：

提高管理效率：随着高校学生数量的增加，传统的手工管理方式已经无法满足日益增长的管理需求。通过开发学生信息管理系统，可以实现学生信息的数字化管理，提高数据处理效率和准确性，从而有效减轻管理人员的工作负担。

优化数据管理：学生信息管理系统能够提供多样化的查询和筛选功能，支持按照不同条件对学生数据进行分类和管理，如按班级、年级、课程等。这有助于教育机构更好地组织和分析学生数据，为教育决策提供有力支持。

促进教育公平：学生信息管理系统通过信息技术打破了地域限制，让更多学生能够享受到相对均衡的教育资源和服务。这有助于减少因地域和资源差异导致的教育差距，促进教育公平。

适应社会进步：随着技术的发展和需求的变化，学生信息管理系统需要不断完善和升级功能，以更好地适应社会进步和教育发展的新形势。这包括利用云计算和大数据技术提升数据处理和存储的效率，减少学校的 IT 投入和管理负担。

1.2 项目设计目标

系统支持教师进行成绩录入、修改、删除和查询操作，确保数据的准确性和及时性。同时，学生可以通过系统方便地查询自己的考试成绩，帮助他们更好地了解和评估自己的学习表现。建立一个 C/S 系统，能够根据不同角色的权限，分别为学生、教师和教务处管理员提供功能，包括学生信息查询、发表留言、学生成绩录入、考试成绩查询以及学籍调整等操作。

遵循 MVC 框架，统一 JDK1.8，使用 Swing 图形化界面，各成员分头行动，最后使用 IDEA 的 code with me 进行代码整合。首要任务是实现学生信息管理、课程安排和成绩管理的基础功能。这些功能是系统的核心。特别地，我们对系统的安全性给予了高度重视，尤其是针对 SQL 注入等常见安全漏洞。如果项目进度允许，我们将着手开发自动排课功能。

1.3 项目可行性分析

经济可行性分析：

1. 成本与投资回报

学生管理系统的建设和实施通常涉及初期投资，包括软件开发、硬件设备、培训和技术支持等方面的成本。这些成本需要与长期运营中节省的成本和效益进行比较。例如，节省的人力成本、纸质文档处理成本、信息管理和报告成本等。

2. 长期运营成本

除了初始投资，学生管理系统还需要考虑长期运营成本，包括系统维护、更新升级、技术支持和人员培训等方面的费用。这些费用应与系统带来的效率提升和服务质量改善相平衡。

3. 效率提升与成本节约

学生管理系统的主要经济收益之一是提升工作效率和服务质量，从而降低管理成本。通过自动化和电子化管理流程，减少了手工处理和纸质文档的使用，节省了大量的时间和人力资源成本。

技术可行性分析

1. 技术架构与设计

学生管理系统的技术架构应该能够支持系统的所有功能需求，如学生信息管理、课程安排、成绩录入与管理、通知发布等。良好的技术架构能够提供稳定的系统性能和扩展性，以应对未来

可能的需求变化和功能扩展。

2. 数据管理与安全

系统需要有效地管理大量的学生数据和敏感信息，包括个人信息、成绩、课程安排等。因此，数据安全是至关重要的考虑因素。系统应该采用适当的数据加密、权限控制、备份和恢复机制，以确保数据的机密性、完整性和可用性。

3. 用户界面与用户体验

良好的用户界面设计能够提升系统的易用性和用户接受度。系统应该考虑用户的操作习惯和界面设计原则，使用户能够轻松地进行操作、查询信息和执行任务。

法律可行性分析：

1. 数据隐私与保护

学生管理系统处理大量敏感数据，如个人身份信息、家庭背景、成绩等。因此，系统设计必须符合数据保护和隐私保护法律法规，如《个人信息保护法》等。必须确保数据仅用于合法目的，并且有适当的安全措施来防止未经授权的访问和泄露。

2 承担任务

2.1 小组介绍

学生姓名	学号	职责分工
李茂田	23201837	组长，需求分析，后端开发，架构设计，前端开发
熊锋	23201826	模型构建，数据库设计，数据处理
邢沂炜	23201812	界面设计、前端开发、配置管理

2.2 个人承担任务

个人职责包括使用 Java Swing 技术实现用户界面的按钮跳转和菜单栏设计。我负责后端逻辑的编写，确保与数据库的高效交互，同时根据用户的具体需求，设计和实现所需数据的展示方式，以确保信息的高可读性和易用性。此外，我还负责了整个系统功能代码的全面测试和调试工作。作为组长，还包括了协调组内成员分工，根据工作进度安排组员工作与娱乐的平衡。

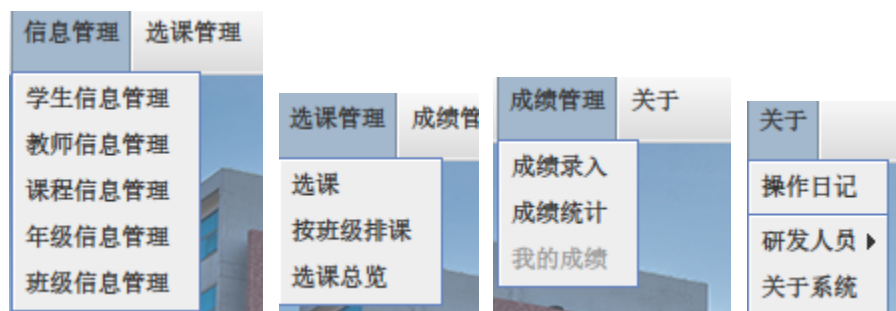
具体来看，我的作用包括了编写按班级批量排课功能，对一个班的成绩自动绘制区间柱状图功能，查看操作日记功能和快捷查询我的全部成绩功能。

测试与 debug 方面，优化了用户修改密码流程，从只有管理员能修改密码改为所有用户都可修改密码，另外在修改密码之后会自动退出，重返登录页面；解决了修改密码后跳转登录却不会关闭主系统页面的 bug；解决了录入和修改学生的学号只能+1 递增，而不能手动录入的 bug；解决了成绩统计和成绩柱状图两个窗口无法同步操作，比如无法自动同步关闭的 bug。

3 负责的需求模块

系统支持教师进行成绩录入、修改、删除和查询操作，确保数据的准确性和及时性。同时，学生可以通过系统方便地查询自己的考试成绩，帮助他们更好地了解 and 评估自己的学习表现。建立一个 C/S 系统，能够根据不同角色的权限，分别为学生、教师和教务处管理员提供功能，包括学生信息查询、发表留言、学生成绩录入、考试成绩查询以及学籍调整等操作。

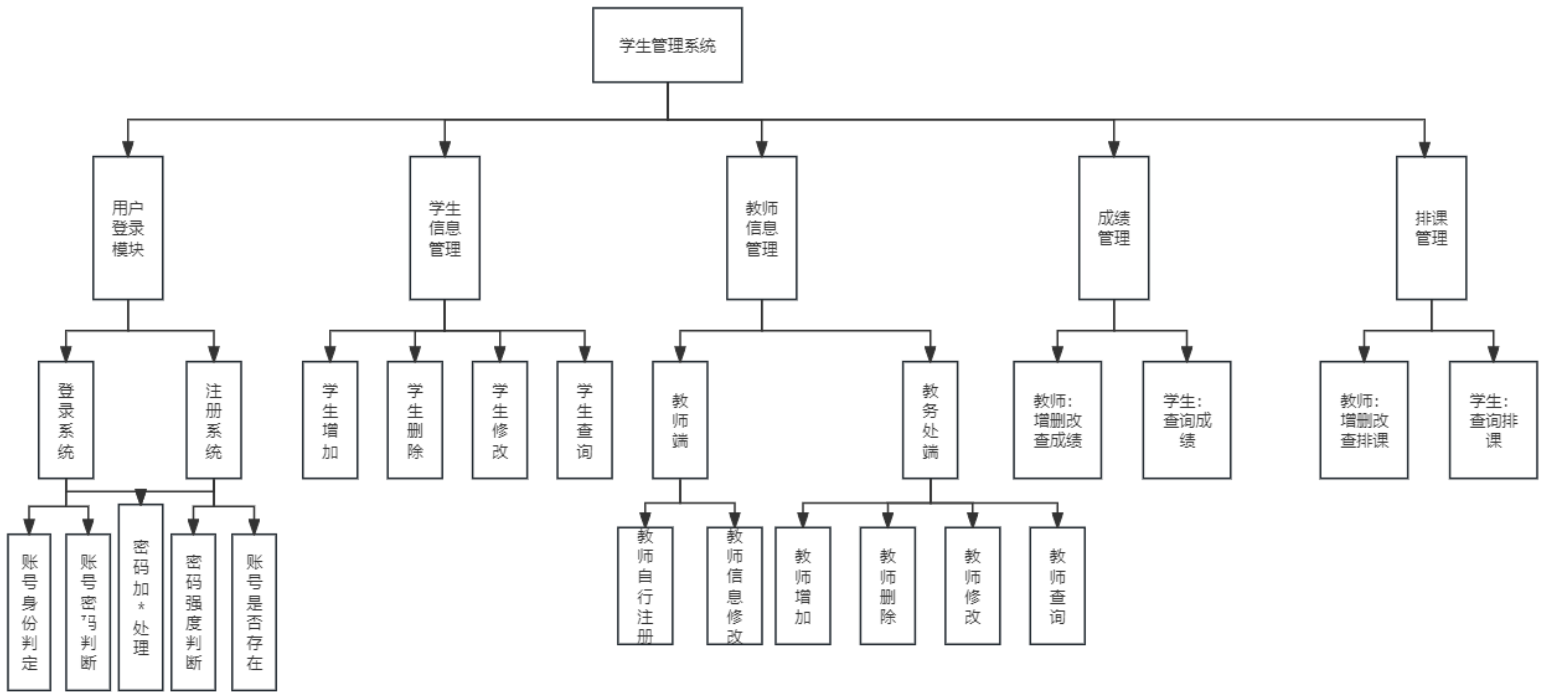
具体来看，分别有用户登录注册，修改密码功能；学生，教师，课程，年级和班级管理功能和主框架设计；单词选课，教务处老师按班级意见排课，总览排课功能；成绩录入，成绩统计，查询我的成绩功能；查看操作日记功能。



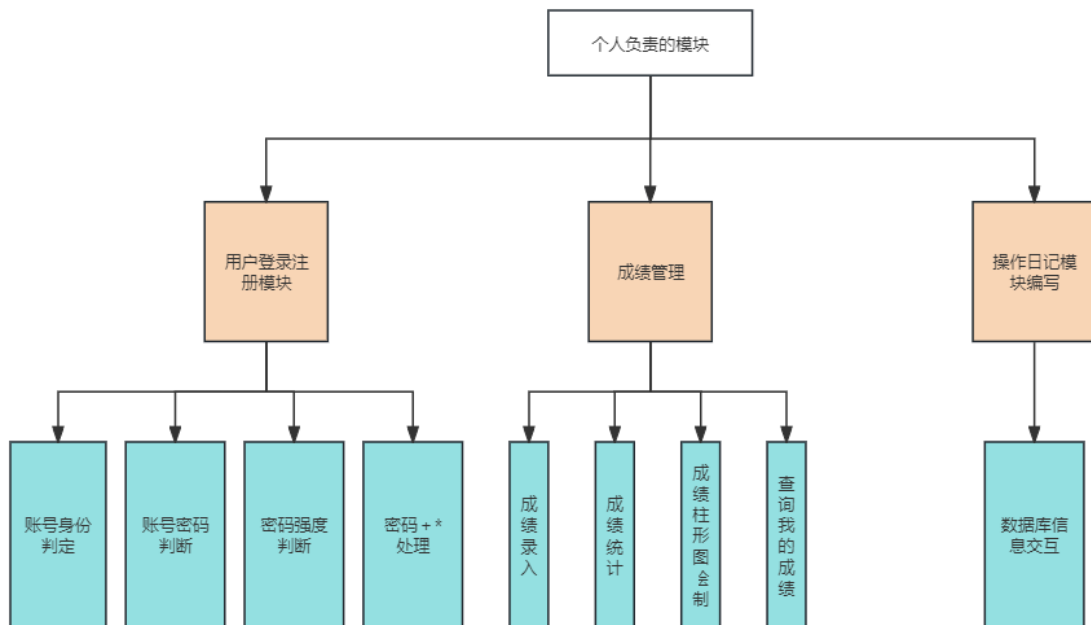
3.1 功能需求

经项目小组需求分析，该系统应该具备用户登录注册模块，学生信息管理，教师信息管理，成绩管理，排课管理这几个主要功能模块，其中用户登录注册界面以及主框架模块，成绩管理模块和操作日记模块编写工作由本人承担。具体功能需求如下所示。

3.2 软件的总体结构



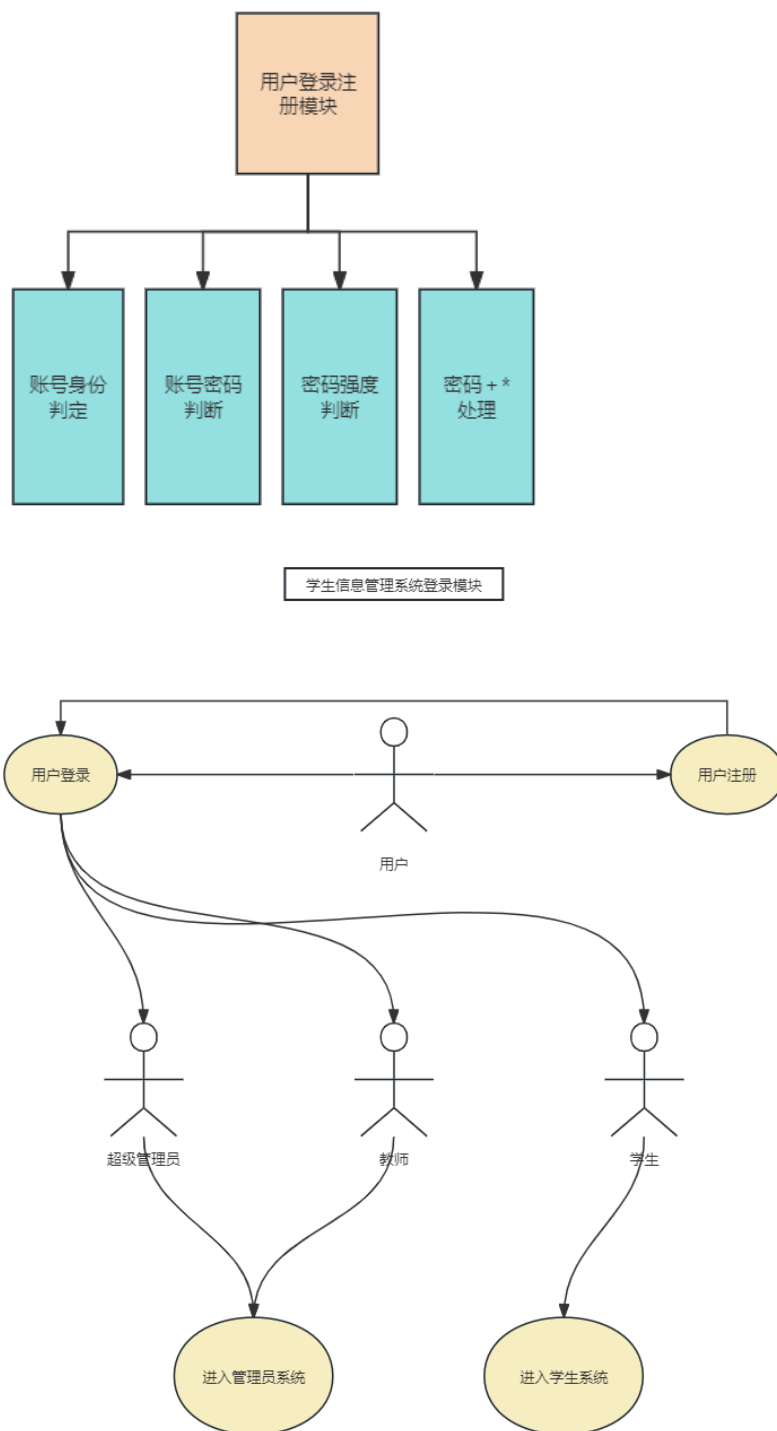
3.3 个人承担概要设计



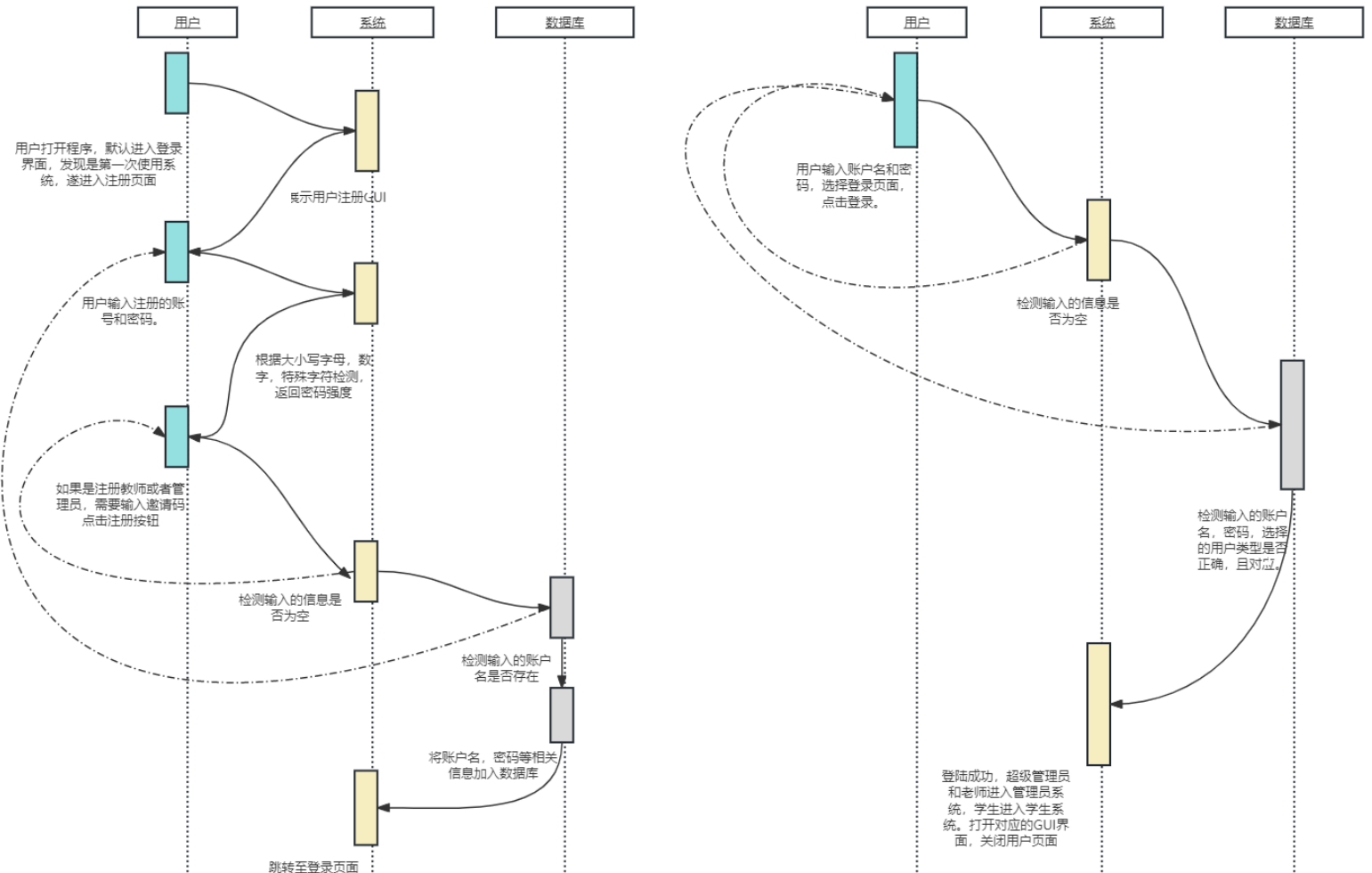
4 负责的详细设计模块

4.1 用户登录注册模块详细以及主框架设计

主要功能如图所示



学生信息管理系统注册后登录时序图



用户登录时序图如上设计

用户登录界面采用 NCHU 校园图片作为背景，包括用户名和密码输入框，以及账户类型多选框。密码经过加密处理并显示。用户可以通过键入 Enter 键或点击确定按钮来进行密码验证操作。选择不同的用户类型将连接到对应身份的存储账户名和密码的数据库，并进行身份信息验证。登录界面的具体图如下所示。

另外，为防止 SQL 注入，使用 PreparedStament 语句防止入侵系统。



主界面包括工具栏和北京时间动态显示功能。根据上一步登录人的身份，系统将禁用部分选项。具体图如下。



4.2 成绩管理模块详细设计

- (1) 成绩录入：系统根据多选框选择的信息与数据库进行交互，显示某学生的当前分数，如果没有打分则显示 0 分。可在当前页面进行学生打分或者更改。

已选课程成绩录入

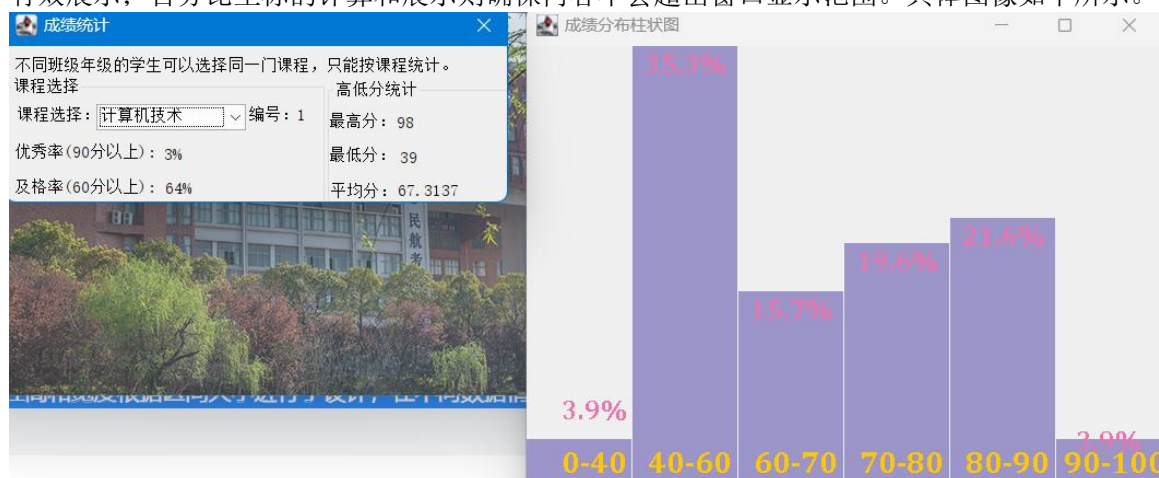
年级选择：一年级 班级选择：核弹技术07-1

选课学生：陈罗志 已修课程：计算机技术 ☐ 开启过滤

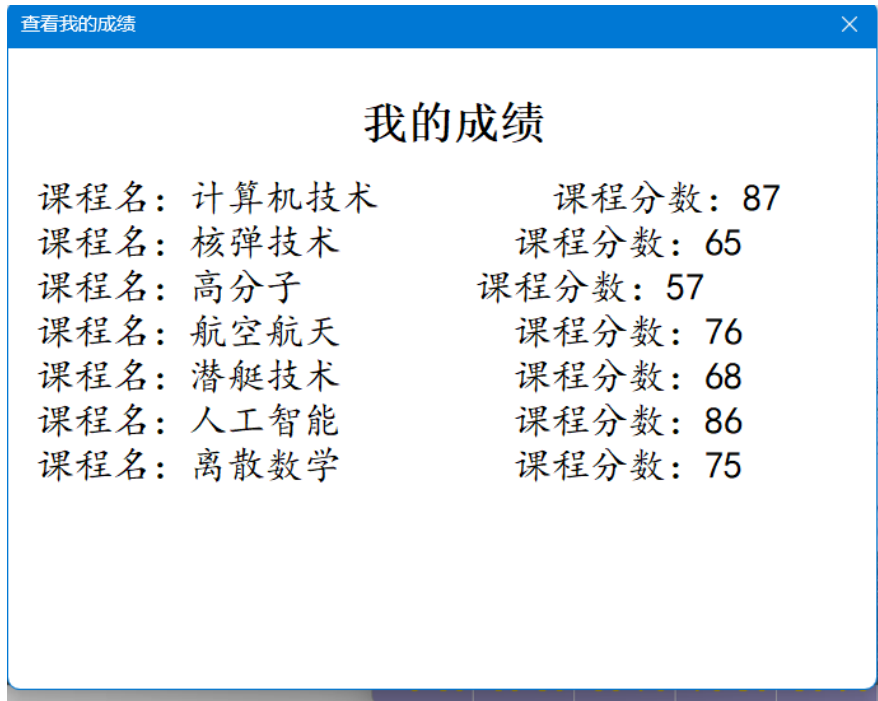
87 确定录入

注：“开启过滤”只显示学生成绩还没被录入的课程。

- (2) 成绩统计：点击此按钮将弹出两个窗口，设计了 X 轴对齐以确保界面美观。第一个窗口展示课程的优秀率、及格率、最高分、最低分和平均分；第二个窗口展示成绩分布的柱状图，将学生成绩分布分为 5 个区间，并呈现为正态分布图。这些功能方便教师了解教学情况，优化课堂教学策略。此外，柱状图的柱高和宽度根据区间大小进行了设计，在不同数据情况下均能有效展示；百分比坐标的计算和展示则确保内容不会超出窗口显示范围。具体图像如下所示。



- (2) 一键查询我的成绩：仅供学生使用，快捷展示自己的所有成绩。



4.3 操作日记模块详细设计

操作日志用于记录用户成功登录的用户名和登录时间等信息。当发生错误更改成绩或学生入侵系统的情况时，可以追查到相关责任人，以确保系统的安全性。界面图如下所示。



5 负责的编码

5.1 用户登录注册和主框架模块核心代码

(1) 用户登录注册

```
public static String storeUserName = null; // 登录用户名
public static String storeUserPassword = null; // 登录密码
static int storeUserId; // 登录用户名
static boolean RELOAD = true; // 重新登陆标记
static int login_user_type; // 0 表示管理员, 1 表示老师, 2 表示学生
    使用 static 类似全局变量, 方便后续操作使用用户名, 密码和身份信息。
```

```
if ("".equals(user)) {
    JOptionPane.showMessageDialog(null, "用户名不能为空");
    return;
}
if ("".equals(password)) {
    JOptionPane.showMessageDialog(null, "密码不能为空");
    return;
}
```

输入为空的验证, 以及弹窗提示

```
if (login_user_type == 0) {}
else if (login_user_type == 1) {}
else if (login_user_type == 2) {}
    根据不同用户类型, 进入不同的界面, 再执行不同的操作。
```

(2) 主框架窗口

```
new Thread(() -> {
    while (true) {
        String time = new
java.text.SimpleDateFormat("HH:mm:ss").format(new
java.util.Date());
        label_time.setText(time);
        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}).start();
```

新建线程, 并行操作, 不影响或者说阻碍其他的代码运行更新时钟的时间。

```
if(Login.login_user_type==1){
    //老师类
```



```

jMenuItem_user_manage.setEnabled(false);
jMenu_information.setEnabled(false);
jMenuItem_log.setEnabled(false);
jMenuItem_lookMyScore.setEnabled(false);
}

```

根据登录身份信息的不同，禁用掉一些功能。

```

public class btnListener implements ActionListener {
    //鼠标事件
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == jMenuItem_relogin) {
            dispose(); //dispose() 方法释放组件占用的资源
            Login login = new Login(); // 调用无参数构造方法
            login.setVisible(true);

```

按钮的实现，新建一个自定义类，实现 ActionListener 接口，让他成为按钮的监听类。

5.2 成绩管理模块核心代码

(1)成绩录入

```

void flashTextArea() {
    int stu_id =
student_lists.get(jComboBox_stu.getSelectedIndex())
                .getStu_id();
    jLabel_stu_id.setText(Integer.toString(stu_id));

    CCourseDao ccd = new CCourseDao();
    List<CCourseModel> lists = ccd.getListsByStuId(stu_id);
    jTextArea_ccourse.setText("");
    for(int i=0;i<lists.size();i++){
        int course_id = lists.get(i).getCourse_id();
        //根据 id 查找对应的课程名称
        String course_name =
ccd.getNameByCourseId(course_id);
        jTextArea_ccourse.append("课程编号: " + course_id+"
"+"课程名称: "+course_name);
        jTextArea_ccourse.append("\n");
    }
}

```

每一次更改多选框的内容，都会执行 flashTextArea 操作，刷新多选框的文本，同步信息。

(2) 成绩统计 柱形图绘制

// 绘制每个柱子

```

for (int i = 0; i < data.length; i++) {

```

```

g.setColor(new Color(155, 149, 201));
//g.setColor(Color.BLUE);
// 计算柱子的位置和高度
int x = i * barWidth;
int barHeight = (int) ((double) data[i] / maxValue *
height); // 缩放数据以适应面板高度
int y = height - barHeight; // 计算柱子顶部位置
// 绘制柱子
g.fillRect(x, y, barWidth - 1, barHeight);
// 添加数据的下标
if (data[i]>0){
    String str=null;
    String percentage=null;
    switch (i){
        case 0:str="0-40";break;
        case 1:str="40-60";break;
        case 2:str="60-70";break;
        case 3:str="70-80";break;
        case 4:str="80-90";break;
        case 5:str="90-100";break;
    }
    根据数据的大小动态缩放数据以适应面板高度。

double temp=data[i]/(double)amount*100;
percentage=String.format("%.1f" ,temp)+ "%";
int pWidth = fm.stringWidth(percentage);
g.drawString(str, x + (barWidth - labelWidth) / 2, 390);
//y 为柱子顶部底的高度
g.setFont(new Font("Cambria", Font.BOLD, 25));
y+=fm.stringWidth(percentage);
if (y-50<=0){
}else {
    y-=50;
}
g.setColor(new Color(241,115,172));
g.drawString(percentage, x + (barWidth - pWidth) / 2, y);
百分比字符串要判定显示的位置,防止因为显示位置超出界面而造成无法显示的情况。

```

(3) 查询我的成绩

```

JTextArea_ccourse = new JTextArea();
CCourseDao cCourseDao = new CCourseDao();
List<StuCourseMarkModel> lists =
cCourseDao.getCCourseMarkCourseNameByStuId(studentID);
for(int i=0;i<lists.size();i++){
    String courseName = lists.get(i).getCourseName();
    int mark=lists.get(i).getCourseMark();
}

```



```

        jTextArea_ccourse.append("课程名: " + courseName +
        "+"课程分数: "+mark);
        jTextArea_ccourse.append("\n");
    }
    jTextArea_ccourse.setFont(new Font("楷体", Font.PLAIN, 30));
    jTextArea_ccourse.setEditable(false);

```

根据与数据库交互得到信息，对 jTextArea 扩展，并且设置其不可编辑，设置合适的字体大小以方便展示。

5.3 操作日记模块核心代码

```

LogDao ld = new LogDao();
List<LogModel> lists = ld.getLogLists();
String heads[] = { "用户 ID", "操作", "操作时间"};
model = new DefaultTableModel(null, heads);
model.setRowCount(lists.size());
try {
    for (int i = 0; i < lists.size(); i++) {
        String[] times =
lists.get(i).getLog_time().toString().split(" ");
        model.setValueAt(lists.get(i).getLogin_user(),
i, 0);
        model.setValueAt(lists.get(i).getOperate(), i,
1);
        //model.setValueAt(times[0], i, 2); //只显示年月日
        model.setValueAt(lists.get(i).getLog_time(), i,
2); //显示日期+时间
    }
}
catch (Exception e) {
}

```

从数据库获取信息，显示用户 ID,操作和操作时间。

6 考虑社会，健康，安全，法律等因素

系统的设计应当注重保护学生的个人信息，遵循隐私法律和政策，确保数据的安全性和保密性，防止未经授权的访问和数据泄露。安全方面，系统必须有严格的访问控制和身份验证机制，确保只有授权的人员才能访问和修改敏感信息。系统的架构应具备弹性，以应对潜在的安全威胁和网络攻击。法律方面，则需遵守国家和地区的相关法律法规，特别是关于数据保护、个人隐私和信息安全的法律要求，确保系统的合法性和可持续性。

7 实训总结

7.1 项目工作总结

在本次实训中，我们小组负责开发的学生信息管理系统项目，经过团队成员的共同努力，最终基本达到了项目的预期目标。我们成功实现了包括登录界面设计、学生信息管理，教师信息管理，成绩管理，操作日记等核心功能。尽管在开发过程中遇到了一些技术难题和挑战，但通过团队内部的沟通、资料查阅和老师的指导，我们最终克服了这些困难。

7.2 项目设计成果总结

在本次我们小组负责的学生信息管理系统最终基本达到了项目的预期，完成了包括登录界面的设计，可以进行登录界面设计、学生信息管理，教师信息管理，成绩管理，操作日记等核心功能。虽然还是有一瑕疵，但是基本功能点还是都实现了。

具体来看，项目的设计成果主要体现在以下几个方面：

用户界面设计：我们设计了直观、易用的用户界面，确保了用户能够轻松地进行操作和管理。

功能实现：我们实现了系统的核心功能，包括用户登录注册、信息管理、成绩管理、操作日记管理等，满足了用户的基本需求。

安全性：我们重视系统的安全性，采取了多种措施防止 SQL 注入等安全漏洞，确保了数据的安全性和隐私性。包括成绩修改等高危操作也有日记记录，若有用户入侵系统，可以及时查看操作日记，回退数据。

7.3 存在的不足及下一步工作

尽管项目取得了一定的成果，但我们也认识到存在一些不足之处：

功能完善度：部分功能模块的完善度还有待提高，如自动排课功能尚未实现。

用户体验：用户界面的美观性和易用性还有提升空间，需要进一步优化。可以在主框架界面增加更多的信息显示。

性能优化：在数据库查询中，有大量重复查询 ID 的操作，在数据量较多的情况下，可能会出现延迟，卡顿的情况。未来准备重构实体类的设计，调节学生，教师，与课程间的关系，以降低数据所占的内存空间。

边界值和特殊值检查：在打分时，输入浮点数会存在几无法打分，也没有相关的提示框出现的情况，容易造成用户误解；另外，还可以打负分，不符合实际；

数据库字段设计不合理：学号为了处理方便，设计的是 `int` 形，如果遇到研究室，博士生的学生信息处理，他们的学号为了区分方便，常常有 Y,B 等字符，如 Y100012,B10021,如遇到此类情况，则该系统无法处理。分数目前为 `int` 形，不精确，应该改为浮点型，以符合现实情况，比如 89.3 的平均分，77 的考试成绩，再取 40%，60% 的比例，再计算总分为 81.92，结尾取 81，展示给用户，这样数据更为精确。

7.4 个人在项目开发中的主要工作成果及不足之处

成果：

个人在项目中负责登录注册界面以及主框架模块，成绩管理模块和操作日记模块编写。作为组长，还包括了协调组内成员分工，根据工作进度安排组员工作与娱乐的平衡。

测试与 Debug 方面，优化了用户修改密码流程，从只有管理员能修改密码改为所有用户都可修改密码，另外在修改密码之后会自动退出，重返登录页面；解决了修改密码后跳转登录却不会关闭主系统页面的 Bug；解决了录入和修改学生的学号只能+1 递增，而不能手动录入的 Bug；解决了成绩统计和成绩柱状图两个窗口无法同步操作，比如无法自动同步关闭的 Bug。

不足：

首先，在团队协作方面，我没有有效安排团队成员的任务，导致了无法“并发”协作的不足，只能采取了“并行”的工作方式。这种情况下，我在前后端编写上需要频繁地等待负责数据库的同事完成他们的编码工作，影响了整体的开发效率和进度。

其次，我在时间安排上也存在不合理的情况，部分原计划的扩展功能由于时间不足而未能实现。这反映出我在项目计划和时间管理方面需要更好的规划和执行能力，以确保项目能够按时交付，并且能够实现预期的所有功能和扩展需求。

除此之外，还有其他方面的不足，比如可能在代码质量的细节处理上有所疏忽，或者在需求分析阶段的沟通和理解不够清晰，导致后续开发中出现了一些调整和修正。

8 参考文献

【1】 IceHowe. "Student Management System." GitHub, n.d., https://github.com/IceHowe/student_management_system.

【2】 Liujiang240556. " StudentGradeMangerSystem " GitHub, n.d., <https://github.com/Liujiang240556/StudentGradeMangerSystem>

【3】 张明东, 戴丹丹. "Java EE 下学生信息管理系统的设计与实现." 集宁师范学院, n.d. Web. [2024-7-9].

【4】 李志学. "基于 C/S 架构的潍坊科技学院学生信息管理系统." 中国海洋大学, n.d. Web. [2024-7-9].

附录

附录 1

1. 学生管理系统调研报告
2. 学生信息管理系统需求规格说明书
3. 学生信息管理系统系统设计
4. 学生信息管理系统测试报告