|  |
| --- |
| UESTC |
| 软件详细设计 |
| 软件工程大作业 |

|  |
| --- |
| 王哲宇 2023090912005  2024-11-30 |

软件详细设计报告文档

[1. 引言 2](#_Toc183179476)

[1.1 编写目的 2](#_Toc183179477)

[1.2 项目风险 2](#_Toc183179478)

[1.3 文档约定 2](#_Toc183179479)

[1.4 预期读者和阅读建议 3](#_Toc183179480)

[2. 支撑环境 3](#_Toc183179481)

[2.1 开发工具、中间件以及数据库接口 3](#_Toc183179482)

[2.2 硬件环境 4](#_Toc183179483)

[2.3 多种支撑环境开发要点 5](#_Toc183179484)

[3. 部件详细设计 6](#_Toc183179485)

[4. 词汇表 15](#_Toc183179486)

# 引言

## 编写目的

本软件系统详细设计报告基于以下文档编写：

* **软件产品需求分析报告**：详述学生成绩管理系统的功能需求，包括学生、教师、管理员的操作范围与数据管理目标。
* **软件产品概要设计报告**：提供了系统的整体架构设计，明确采用 C/S 架构，定义了主要的模块和类的交互关系。

开发本系统的意义在于为高校提供高效、直观的学生成绩管理工具。通过系统化的管理，减少手工操作的繁琐，提高数据处理的准确性和效率。本报告详尽描述了系统的编码结构，明确其物理组成，为后续实现与维护提供参考。

本详细设计报告仅针对整个系统的核心功能模块，包括学生成绩查询与管理、教师课程管理、管理员用户权限管理等部分进行说明。

## 项目风险

在本软件开发项目中，可能存在以下风险，需明确承担者及其对应职责：

* **任务提出者**（如教务部门）：需明确功能需求并验证系统的开发成果是否符合实际需求。主要风险为需求描述不准确或变更频繁。
* **软件开发者**（开发团队）：负责系统的设计与实现。主要风险包括编码错误、性能问题、系统安全漏洞等。
* **产品使用者**（学生、教师、管理员）：需掌握系统使用方式，正确输入数据并反馈使用问题。主要风险为用户培训不足或输入数据不规范。

## 文档约定

为了确保本报告编写的规范性，采用以下约定：

* **部件编号方式**：模块按功能分为 StudentModule（学生模块）、TeacherModule（教师模块）和 AdminModule（管理员模块），方法按具体功能命名，如 addStudent()、setGrade()。
* **界面编号方式**：界面按用户类型编号，如学生界面 SP01，教师界面 TP01，管理员界面 AP01。
* **命名规范**：采用驼峰命名法（CamelCase），类名以大写字母开头（如 Student），变量名以小写字母开头（如 studentId）。
* **文件编码**：所有 Java 文件采用 GBK 编码。

## 预期读者和阅读建议

本软件系统详细设计报告针对以下预期读者：

* **开发人员**：关注系统模块的实现细节，可重点阅读模块设计与方法描述部分。
* **项目经理**：关注系统的整体设计与风险，可阅读引言与系统总体设计部分。
* **测试人员**：关注功能测试点与接口规范，可阅读系统接口设计与功能模块说明。
* **文档编写人员**：关注整体文档结构，可通读全文以掌握报告的组织逻辑。

文档其余部分结构如下：

* **系统总体设计**：介绍系统的架构及模块划分。
* **详细设计描述**：包含每个模块的类结构、方法设计及接口说明。
* **附录**：包括术语表、缩略语及参考文献等。

建议不同读者可以根据需求选择性阅读相关部分，以提高阅读效率。

# 支撑环境

## 开发工具、中间件以及数据库接口

* **开发工具**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **产品名称** | **发行厂商** | **版本号** | **补丁号版本号** | **语言或代码集** |
| Eclipse IDE | Eclipse 基金会 | 2023-06 R | N/A | Java (GBK 编码) |
| JDK | Oracle Corporation | 17.0.8 | N/A | Java 编程语言 |

**Eclipse IDE**（Integrated Development Environment，集成开发环境）是一个开放源代码的、基于Java的可扩展开发平台。Eclipse最初是由IBM公司开发的，用于Java语言的开发，但现在它已经发展成一个可以支持多种编程语言和平台的通用开发工具。Eclipse的核心功能相对有限，但通过插件可以扩展其功能。这使得Eclipse可以支持多种编程语言和工具，如C/C++、Python、PHP等。同时，Eclipse是Java开发的首选IDE之一，提供了对Java语言的深度支持，因此我们选择该工具进行开发

**JDK**（Java Development Kit，Java开发工具包）是Oracle公司提供的官方Java开发工具集，它包括了Java运行环境（JRE）和一些开发Java程序所需的工具和库。其中组件包括了Java编译器，用于将Java源代码编译成字节码文件；Java运行时环境（JRE），包括Java虚拟机（JVM）和Java类库，用于运行Java程序等，便于开发者使用Java进行开发。

* **数据库、接口与中间件**

本项目在实际设计中并未使用到数据库以及相关接口。未来可以使用的相关工具有：MySQL作为系统的后端数据库，存储学生、教师、课程及成绩信息；JDBC（Java Database Connectivity）接口用于 Java 应用程序与 MySQL 数据库之间的连接和数据操作；Apache Tomcat 用于系统部署和运行，主要负责处理客户端和服务器之间的通信，以及处理基于 JSP 的动态网页请求。

* **辅助开发工具**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **产品名称** | **发行厂商** | **版本号** | **补丁包版本号** | **语言或代码集** |
| Git | Git 开发团队 | |  | | --- | |  |   2.42.0 | N/A | N/A |
| Maven | Apache 软件基金会 | 3.9.5 | N/A | POM 配置文件(XML) |

Git 是一个开源的分布式版本控制系统，由 Linus Torvalds 创建，用于有效、高速地处理从小到大的项目版本管理。Git 是目前世界上最流行的版本控制系统，被广泛用于软件开发中，以便于代码的版本控制和多人协作。每个开发者的本地都保存有完整的代码仓库，包括代码、提交历史和分支等。此外，Git还支持快速切换的分支（Branch）和合并（Merge），方便开发者进行非线性开发。

**Maven** 是一个项目管理和构建自动化工具，主要服务于基于 Java 的项目。Maven 可以看作是一个项目管理和构建工具，它使用一个名为 POM（Project Object Model，项目对象模型）的 XML 文件来描述项目的构建过程、依赖关系等。它提供了自动处理项目依赖，包括依赖的下载、更新和版本控制的功能，使用 POM 文件定义项目结构和配置信息，通过插件和目标（Goals）来执行特定的构建任务，如编译、测试、打包等。

## 硬件环境

* **开发设备（工作站）**

|  |  |
| --- | --- |
| **参数** | **描述** |
| **机型** | Lenovo ThinkPad E14 |
| **主频** | 1.6 GHz (Intel Core i5-10210U，四核心处理器) |
| **内存容量** | 8 GB DDR4 |
| **磁盘容量** | 256 GB SSD |
| **特殊部件** | 无 |
| **操作系统** | Windows 10 Professional (64-bit) |
| **使用位置** | 开发团队使用，用于代码编写和调试 |
| **用途** | 运行 Eclipse IDE 进行代码开发与本地调试，测试基本功能实现 |

* **测试设备（笔记本）**

|  |  |
| --- | --- |
| **参数** | **描述** |
| **机型** | Dell Inspiron 15 3000 |
| **主频** | 2.1 GHz (AMD Ryzen 3 3250U，双核心处理器) |
| **内存容量** | 8 GB DDR4 |
| **磁盘容量** | 512 GB HDD |
| **特殊部件** | 无 |
| **操作系统** | Ubuntu 20.04 LTS (64-bit) |
| **使用位置** | 用于功能测试和基本性能验证 |
| **用途** | 测试开发完成的学生成绩管理系统功能，验证基本操作是否符合需求 |

* **普通运行设备**

|  |  |
| --- | --- |
| **参数** | **描述** |
| **机型** | Lenovo ThinkPad T14s Gen 3 |
| **主频** | 2.1 GHz (AMD Ryzen 7 PRO 6850U，8 核心处理器) |
| **内存容量** | 16 GB LPDDR4x |
| **磁盘容量** | 512 GB NVMe SSD |
| **特殊部件** | 指纹识别模块，TPM 2.0 芯片 |
| **操作系统** | Windows 10 Home (64-bit) |
| **使用位置** | 模拟最终用户环境，主要用于用户体验测试 |
| **用途** | 验证系统在普通硬件环境下的运行稳定性与操作流畅性 |

## 多种支撑环境开发要点

* **开发环境要点：**
  + **版本一致性：**所有开发设备需统一 Eclipse IDE 和 JDK 版本，确保项目代码在不同设备上兼容。
  + **模块化设计：**严格遵循 MVC 架构，将视图层、控制层和数据层解耦，方便跨环境迁移与扩展。
  + **数据库初始化：**为开发环境提供一套标准化的数据库脚本，用于快速初始化开发所需数据表。
* **测试要点：**
  + **跨平台测试：**系统需在 Windows 和 Linux 上运行测试，验证其跨平台兼容性。
  + **自动化测试**：通过 JUnit 编写单元测试，使用 Selenium 进行界面功能测试，覆盖关键功能模块。
  + **负载测试**：在测试环境中模拟多用户并发访问场景，验证系统性能与稳定性。
* **用户运行环境要点：**
  + **轻量化运行：**系统需对普通硬件环境进行优化，确保低配置设备上的运行流畅。
  + **安装说明**：提供详细的安装与部署文档，帮助用户快速配置 JRE、MySQL 和系统文件。
  + **错误提示友好**：增强系统异常处理能力，提示信息应简洁易懂，便于用户操作。

# 部件详细设计

**部件1：用户信息验证模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 001 | 部件名称 | CheckInfo |
| 所属子系统 | 用户管理子系统 | | |
| 部件调用者 | AddUser、DeleteUser、MainFrame | | |
| 部件被调用者 | 无 | | |
| 部件入口参数 | 用户类型（String）、用户编号（String）、密码（String） | | |
| 部件出口参数 | 验证结果（int），返回值含义：  1：用户编号和密码匹配  2：用户编号存在但密码不匹配  0：用户不存在 | | |
| 算法：遍历用户类型对应的文件，逐行读取并匹配用户编号和密码。 | | | |
| 流程描述：  1. 接收用户类型、编号和密码。  2. 根据用户类型选择对应的文件（如 student.txt）。  3. 遍历文件，逐行读取并拆分成字段数组。  4. 比较编号和密码，返回验证结果。 | | | |
| 表示形式 | 函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 响应时间 < 1 秒 |  |  |

**部件2：新增用户模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 002 | 部件名称 | AddUser |
| 所属子系统 | 用户管理子系统 | | |
| 部件调用者 | Administrator | | |
| 部件被调用者 | CheckInfo | | |
| 部件入口参数 | 用户信息（编号、姓名、性别、生日、学院、专业） | | |
| 部件出口参数 | 操作结果（boolean），标志用户是否成功添加 | | |
| 算法：验证用户是否存在，若不存在则追加至文件。 | | | |
| 流程描述：  1. 接收用户信息。  2. 调用 CheckInfo 验证用户是否已存在。  3. 若用户不存在，则将用户信息追加到对应的文件中。  4. 返回操作成功或失败标志。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 添加操作准确率 100%  2. 响应时间 < 2 秒 |  |  |

**部件3：删除用户模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 003 | 部件名称 | DeleteUser |
| 所属子系统 | 用户管理子系统 | | |
| 部件调用者 | Administrator | | |
| 部件被调用者 | CheckInfo | | |
| 部件入口参数 | 用户类型（String）、用户编号（String） | | |
| 部件出口参数 | 操作结果（boolean），标志用户是否成功删除 | | |
| 算法：验证用户是否存在，若存在则从文件中删除对应记录。 | | | |
| 流程描述：  1. 接收用户类型和编号。  2. 调用 CheckInfo 验证用户是否存在。  3. 若用户存在，删除文件中对应记录并保存修改。  4. 返回操作成功或失败标志。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 删除操作准确率 100%  2. 响应时间 < 3 秒 |  |  |

**部件4：查询用户信息模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 004 | 部件名称 | Info |
| 所属子系统 | 用户信息查询子系统 | | |
| 部件调用者 | 用户界面（如 StudentPanel） | | |
| 部件被调用者 | 无 | | |
| 部件入口参数 | 用户编号（String）、用户类型标志（int） | | |
| 部件出口参数 | 用户详细信息（JLabel） | | |
| 算法：根据用户编号查找用户信息并显示到界面。 | | | |
| 流程描述：  1. 接收用户编号和类型标志。  2. 根据标志选择用户信息文件（如 student.txt）。  3. 遍历文件，找到用户编号对应的记录。  4. 将用户信息解析后动态更新到界面。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 查询结果准确率 100%  2. 响应时间 < 2 秒 |  |  |

**部件5：成绩录入模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 005 | 部件名称 | GradeEnter |
| 所属子系统 | 成绩管理子系统 | | |
| 部件调用者 | TeacherPanel | | |
| 部件被调用者 | CheckInfo、文件操作模块 | | |
| 部件入口参数 | 课程编号（String）、学生编号（String）、成绩（float） | | |
| 部件出口参数 | 操作结果（boolean），标志成绩是否录入成功 | | |
| 算法：验证课程和学生是否存在，写入成绩到文件中。 | | | |
| 流程描述：  1. 接收课程编号和学生成绩。  2. 验证课程和学生是否存在。  3. 若验证通过，则将成绩写入成绩文件。  4. 返回成功或失败标志。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 成绩录入准确率 100%  2. 响应时间 < 3 秒 |  |  |

**部件6：课程查询模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 006 | 部件名称 | CourseView |
| 所属子系统 | 课程管理子系统 | | |
| 部件调用者 | 用户界面（如 StudentPanel、TeacherPanel） | | |
| 部件被调用者 | 文件操作模块 | | |
| 部件入口参数 | 用户编号（String）、查询标志（int） | | |
| 部件出口参数 | 课程信息列表（JTextArea） | | |
| 算法：根据用户类型和编号返回其相关的课程信息。 | | | |
| 流程描述：  1. 接收用户编号和查询标志（0：学生，1：教师）。  2. 根据标志选择查询逻辑：  • 学生：根据学号匹配课程学生文件，获取相关课程编号，再查询课程信息。  • 教师：直接查询教师编号对应的课程。  3. 返回课程信息列表并显示到界面。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 查询结果准确率 100%  2. 响应时间 < 3 秒 |  |  |

**部件7：成绩分段统计模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 007 | 部件名称 | GradeSort |
| 所属子系统 | 成绩管理子系统 | | |
| 部件调用者 | 用户界面（如 TeacherPanel） | | |
| 部件被调用者 | Course 模型类 | | |
| 部件入口参数 | 课程编号（String）、分段标准（float[]） | | |
| 部件出口参数 | 分段统计结果（int[]，每段学生数量） | | |
| 算法：按分段标准统计课程的学生成绩分布。 | | | |
| 流程描述：  1. 接收课程编号和分段标准。  2. 验证分段标准（如范围合理性）。  3. 根据课程编号获取学生成绩列表。  4. 根据分段标准统计成绩分布并返回结果。 | | | |
| 表示形式 | 函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 统计结果准确率 100%  2. 响应时间 < 5 秒 |  |  |

**部件8：新增课程模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 008 | 部件名称 | AddCourse |
| 所属子系统 | 课程管理子系统 | | |
| 部件调用者 | TeacherPanel | | |
| 部件被调用者 | 文件操作模块 | | |
| 部件入口参数 | 课程信息（编号、名称、学分、学时、教师编号、教师姓名） | | |
| 部件出口参数 | 操作结果（boolean），标志课程是否成功添加 | | |
| 算法：验证课程是否存在，若不存在则将课程信息写入文件。 | | | |
| 流程描述：  1. 接收课程信息。  2. 验证课程编号是否已存在。  3. 若课程不存在，则将课程信息写入课程文件，创建相关成绩文件。  4. 返回操作成功或失败标志。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 操作准确率 100%  2. 响应时间 < 3 秒 |  |  |

**部件9：用户抽象类**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 009 | 部件名称 | User |
| 所属子系统 | 数据模型子系统 | | |
| 部件调用者 | Student、Teacher、Administrator 子类 | | |
| 部件被调用者 | 无 | | |
| 部件入口参数 | 构造方法参数：用户编号（String）、密码（String）、姓名（String） | | |
| 部件出口参数 | 无 | | |
| 算法：无 | | | |
| 流程描述：  1. 定义用户编号、密码、姓名的属性。  2. 提供属性的 getter 和 setter 方法。  3. 提供无参和有参构造方法。  4. 作为抽象类供子类继承。 | | | |
| 表示形式 | 抽象类 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据封装性强，便于扩展  2. 保证子类继承一致性 |  |  |

**部件10：学生模型类**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 010 | 部件名称 | Student |
| 所属子系统 | 数据模型子系统 | | |
| 部件调用者 | 用户界面（如 StudentPanel） | | |
| 部件被调用者 | User（继承） | | |
| 部件入口参数 | 构造方法参数：用户编号、密码、姓名、性别、生日、学院、专业 | | |
| 部件出口参数 | 无 | | |
| 算法：无 | | | |
| 流程描述：1. 继承 User 类，扩展学生的特有属性：性别、生日、学院和专业。  2. 提供相应的 getter 和 setter 方法。  3. 提供无参和有参构造方法。 | | | |
| 表示形式 | 类 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据封装性强  2. 便于与用户界面交互 |  |  |

**部件11：教师模型类**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 011 | 部件名称 | Teacher |
| 所属子系统 | 数据模型子系统 | | |
| 部件调用者 | 用户界面（如 TeacherPanel） | | |
| 部件被调用者 | User（继承） | | |
| 部件入口参数 | 构造方法参数：用户编号、密码、姓名、性别、生日、学院、专业 | | |
| 部件出口参数 | 无 | | |
| 算法：无 | | | |
| 流程描述：1. 继承 User 类，扩展教师的特有属性：性别、生日、学院和专业。  2. 提供相应的 getter 和 setter 方法。  3. 提供无参和有参构造方法。 | | | |
| 表示形式 | 类 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据封装性强  2. 便于与用户界面交互 |  |  |

**部件12：管理员模型类**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 012 | 部件名称 | Administrator |
| 所属子系统 | 数据模型子系统 | | |
| 部件调用者 | 用户界面（如 AdministratorPanel） | | |
| 部件被调用者 | User（继承） | | |
| 部件入口参数 | 构造方法参数：用户编号、密码、姓名、性别、生日、学院、专业 | | |
| 部件出口参数 | 无 | | |
| 算法：无 | | | |
| 流程描述：1. 继承 User 类，扩展管理员的特有属性：性别、生日、学院和专业。  2. 提供相应的 getter 和 setter 方法。  3. 提供无参和有参构造方法。 | | | |
| 表示形式 | 类 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据封装性强  2. 便于与用户界面交互 |  |  |

**部件13：课程模型类**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 013 | 部件名称 | Course |
| 所属子系统 | 数据模型子系统 | | |
| 部件调用者 | CourseView、GradeSort、AddCourse | | |
| 部件被调用者 | 文件操作模块 | | |
| 部件入口参数 | 构造方法参数：课程编号、课程名称、教师编号、教师姓名、学分、学时、分段标准（可选） | | |
| 部件出口参数 | 无 | | |
| 算法：包括课程存在性验证、成绩分段统计、分段标准有效性检查等功能。 | | | |
| 流程描述：  1. 定义课程的基本属性：编号、名称、教师信息、学分、学时等。  2. 提供构造方法和对应的 getter、setter 方法。  3. 提供以下功能：   • **课程存在性验证**：检查课程编号是否已存在。   • **成绩分段统计**：根据成绩文件统计学生成绩分布情况。   • **分段标准有效性检查**：确保分段标准在 [0,100] 范围内，且分段顺序正确。 | | | |
| 表示形式 | 类 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据处理正确率 100%  2. 统计结果准确无误 |  |  |

**部件14：成绩分段统计功能**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 014 | 部件名称 | sortGrade |
| 所属子系统 | 成绩管理子系统 | | |
| 部件调用者 | GradeSort | | |
| 部件被调用者 | 文件操作模块 | | |
| 部件入口参数 | 无（使用课程对象中的分段标准和课程编号） | | |
| 部件出口参数 | 分段统计结果（int[]，每个分段人数） | | |
| 算法：根据课程分段标准统计成绩分布并返回结果。 | | | |
| 流程描述：  1. 遍历成绩文件，找到对应课程的成绩记录。  2. 根据课程的分段标准统计学生成绩分布。  3. 返回每个分段的学生人数。 | | | |
| 表示形式 | 方法 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 统计结果准确率 100%  2. 响应时间 < 2 秒 |  |  |

**部件15：课程验证功能**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 015 | 部件名称 | hasCourse 方法 |
| 所属子系统 | 课程管理子系统 | | |
| 部件调用者 | AddCourse、CourseView | | |
| 部件被调用者 | 文件操作模块 | | |
| 部件入口参数 | 无（使用课程对象中的课程编号） | | |
| 部件出口参数 | 验证结果（int）：1 表示课程存在，0 表示课程不存在 | | |
| 算法：遍历课程文件查找是否存在指定课程编号。 | | | |
| 流程描述：  1. 打开课程文件并逐行读取记录。  2. 检查是否存在匹配的课程编号。  3. 返回验证结果。 | | | |
| 表示形式 | 方法 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 验证准确率 100%  2. 响应时间 < 1 秒 |  |  |

**部件16：成绩分段标准验证功能**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 016 | 部件名称 | isValidate |
| 所属子系统 | 成绩管理子系统 | | |
| 部件调用者 | GradeSort | | |
| 部件被调用者 | 无 | | |
| 部件入口参数 | 无（使用课程对象中的分段标准） | | |
| 部件出口参数 | 验证结果（int）：1 表示验证失败，0 表示验证通过 | | |
| 算法：检查分段标准是否满足范围和顺序要求（如 pass < good < excellent）。 | | | |
| 流程描述：  1. 检查分段标准是否在 [0,100] 范围内。  2. 检查分段顺序是否符合逻辑：pass < good < excellent。  3. 返回验证结果。 | | | |
| 表示形式 | 方法 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 验证准确率 100%  2. 响应时间 < 1 秒 |  |  |

**部件17：登录主界面**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 017 | 部件名称 | MainFrame |
| 所属子系统 | 用户登录子系统 | | |
| 部件调用者 | 无（程序入口） | | |
| 部件被调用者 | StudentsPanel、TeachersPanel、AdministratorPanel、CheckInfo | | |
| 部件入口参数 | 用户编号（String）、密码（String）、用户类型（String） | | |
| 部件出口参数 | 登录验证结果，进入对应用户界面 | | |
| 算法：调用CheckInfo验证用户的编号和密码是否正确，若验证通过则进入对应角色的操作界面。 | | | |
| 流程描述：  1. 用户在界面中输入编号和密码，并选择用户类型（学生、教师、管理员）。  2. 调用CheckInfo模块验证用户信息。  3. 若验证通过，则根据用户类型进入对应操作界面（如学生进入StudentsPanel）。  4. 若验证失败，记录错误次数，超过5次则退出系统。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 登录验证准确率 100%  2. 响应时间 < 2 秒 |  |  |

**部件18：学生操作主界面**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 018 | 部件名称 | StudentsPanel |
| 所属子系统 | 学生管理子系统 | | |
| 部件调用者 | MainFrame | | |
| 部件被调用者 | Info、GradeInfo、CourseView、EditInfo | | |
| 部件入口参数 | 学生编号（String） | | |
| 部件出口参数 | 无 | | |
| 算法：通过按钮触发不同功能模块（如信息查询、成绩查询、课程查询、信息修改）。 | | | |
| 流程描述：  1. 学生登录后进入此界面。  2. 学生可选择以下操作：   • **信息查询**：调用Info模块显示学生基本信息。   • **成绩查询**：调用GradeInfo模块显示学生成绩。   • **课程查询**：调用CourseView模块显示学生相关课程信息。   • **信息修改**：调用EditInfo模块修改学生个人信息。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 各功能调用正确率 100%  2. 界面响应时间 < 2 秒 |  |  |

**部件19：教师操作主界面**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 019 | 部件名称 | TeachersPanel |
| 所属子系统 | 教师管理子系统 | | |
| 部件调用者 | MainFrame | | |
| 部件被调用者 | Info、GradeEnter、CourseView、EditInfo、AddCourse、SortGradeFrame | | |
| 部件入口参数 | 教师编号（String） | | |
| 部件出口参数 | 无 | | |
| 算法：通过按钮触发不同功能模块（如成绩录入、课程管理、分段统计等）。 | | | |
| 流程描述：  1. 教师登录后进入此界面。  2. 教师可选择以下操作：   • **信息查询**：调用Info模块显示教师基本信息。   • **成绩录入**：调用GradeEnter模块录入学生成绩。   • **课程查询**：调用CourseView模块显示教师教授的课程信息。   • **信息修改**：调用EditInfo模块修改教师个人信息   • **新增课程**：调用AddCourse模块添加新课程。   • **成绩分段统计**：调用SortGradeFrame模块进行课程成绩分段统计。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 各功能调用正确率 100%  2. 界面响应时间 < 2 秒 |  |  |

**部件20：管理员操作主界面**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 020 | 部件名称 | AdministratorPanel |
| 所属子系统 | 管理员管理子系统 | | |
| 部件调用者 | MainFrame | | |
| 部件被调用者 | AddUser、DeleteUser、EditInfo | | |
| 部件入口参数 | 管理员编号（String） | | |
| 部件出口参数 | 无 | | |
| 算法：通过按钮触发不同功能模块（如新增用户、删除用户、信息修改）。 | | | |
| 流程描述：  1. 管理员登录后进入此界面。  2. 管理员可选择以下操作：   • **新增用户**：调用AddUser模块新增学生、教师或其他管理员。   • **删除用户**：调用DeleteUser模块删除指定用户。   • **信息修改**：调用EditInfo模块修改管理员个人信息。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 各功能调用正确率 100%  2. 界面响应时间 < 2 秒 |  |  |

**部件21：成绩分段统计总界面**

|  |  |  |  |
| --- | --- | --- | --- |
| 部件编号 | 021 | 部件名称 | SortGradeFrame |
| 所属子系统 | 成绩管理子系统 | | |
| 部件调用者 | TeachersPanel | | |
| 部件被调用者 | GradeSort | | |
| 部件入口参数 | 课程编号（String），成绩分段标准（float[]） | | |
| 部件出口参数 | 分段统计结果（显示在界面上） | | |
| 算法：调用GradeSort模块进行成绩分段统计，并将结果显示在界面上。 | | | |
| 流程描述：  1. 教师在界面中输入课程编号和分段标准（及格、良好、优秀）。  2. 调用GradeSort模块验证课程和分段标准的有效性。  3. 若课程存在且标准正确，则进行统计并显示统计结果。  4. 若课程不存在或标准错误，则弹出提示信息。 | | | |
| 表示形式 | 界面+函数 | 运行环境 | Java 8，GBK 文件编码 |
| 性能要求 | 1. 数据统计正确率 100%  2. 界面响应时间 < 2 秒 |  |  |

# 词汇表

|  |  |
| --- | --- |
| 术语 | 定义 |
| **C/S 架构** | Client/Server 架构，客户端与服务器分离的一种系统架构，客户端负责用户界面和部分逻辑处理，服务器负责主要业务逻辑和数据存储。 |
| MVC 架构 | Model-View-Controller 架构，一种软件设计模式，将数据模型（Model）、用户界面（View）和控制逻辑（Controller）分离，便于扩展和维护。 |
| JDK | Java Development Kit，Java 开发工具包，包含 Java 编译器和运行环境，用于开发和运行 Java 程序。 |
| Eclipse IDE | 一个流行的集成开发环境，用于开发 Java 应用程序，支持插件扩展，提供代码编辑、调试、构建等功能。 |
| Git | 一种分布式版本控制系统，用于跟踪代码的更改和版本管理，支持多人协作开发。 |
| Maven | 一个项目管理和构建工具，用于管理 Java 项目的依赖关系和构建过程，通过 POM 文件定义项目结构和配置信息。 |
| JUnit | 一个用于 Java 程序单元测试的框架，支持自动化测试和测试用例组织。 |
| JDBC | Java Database Connectivity，Java 数据库连接，是 Java 应用程序与数据库交互的 API，用于执行 SQL 查询和数据操作。 |
| Apache Tomcat | 一个开源的 Java 应用服务器，用于运行基于 Java Servlet 和 JSP 的 Web 应用程序。 |
| MySQL | 一个开源的关系型数据库管理系统，用于存储和管理数据，通过 SQL 查询语言进行操作。 |
| TPM | Trusted Platform Module，可信平台模块，一种硬件安全芯片，用于增强系统的安全性。 |
| GBK 编码 | 国标汉字编码标准的一种，支持简体中文和繁体中文字符，常用于中文应用程序的开发。 |
| POM 文件 | Project Object Model，项目对象模型文件，Maven 项目的核心配置文件，用于定义项目的依赖关系、构建过程和插件使用。 |
| 学生模块（StudentModule） | 系统功能模块之一，负责处理学生相关的操作，包括信息查询、成绩查询、课程查询和信息修改等功能。 |
| 教师模块（TeacherModule） | 系统功能模块之一，负责处理教师相关的操作，包括课程管理、成绩录入、课程查询和分段统计等功能。 |
| 管理员模块（AdminModule） | 系统功能模块之一，负责处理管理员相关的操作，包括用户管理（新增、删除用户）和信息修改等功能。 |
| 学生面板（StudentsPanel） | 系统的学生端操作界面，用于学生登录后执行相关功能操作，如信息查询、课程查询等。 |
| 教师面板（TeachersPanel） | 系统的教师端操作界面，用于教师登录后执行相关功能操作，如成绩录入、课程管理等。 |
| 管理员面板（AdministratorPanel） | 系统的管理员操作界面，用于管理员登录后执行相关功能操作，如用户管理、信息修改等。 |
| CheckInfo 模块 | 用户信息验证模块，用于验证用户编号和密码是否匹配，确保用户身份的合法性。 |
| AddUser 模块 | 新增用户模块，用于管理员添加新用户（学生、教师、管理员），默认密码为 "123456"。 |
| DeleteUser 模块 | 删除用户模块，用于管理员删除指定用户（学生、教师、管理员）。 |
| EditInfo 模块 | 信息修改模块，用于用户更新个人信息（如姓名、性别、学院、专业等）。 |
| Info 模块 | 用户信息查询模块，用于根据用户编号和类型查询用户的基本信息并显示在界面上。 |
| GradeEnter 模块 | 成绩录入模块，用于教师录入学生的课程成绩，并保存到成绩文件中。 |
| CourseView 模块 | 课程查询模块，用于学生或教师查询课程信息。 |
| GradeSort 模块 | 成绩分段统计模块，用于教师按分段标准统计学生成绩的分布情况（如不及格、及格、良好、优秀）。 |
| AddCourse 模块 | 新增课程模块，用于教师添加新课程，并创建课程相关的文件（如成绩文件、课程学生文件）。 |
| SortGradeFrame 界面 | 成绩分段统计界面，教师输入课程编号和分段标准后，显示统计结果，包括不及格、及格、良好和优秀的学生人数。 |
| MainFrame 界面 | 系统登录主界面，用户通过此界面输入编号和密码选择登录类型（学生、教师、管理员）。 |
| 课程文件（course.txt） | 存储课程信息的文件，包括课程编号、课程名称、教师编号、教师姓名、学分和学时等。 |
| 成绩文件（grade.txt） | 存储学生成绩的文件，包括课程编号、课程名称、学生编号、学生姓名、成绩等。 |
| 用户文件（如 student.txt、teacher.txt、administrator.txt） | 存储不同用户信息的文件，包括编号、密码、姓名、性别、生日、学院和专业等。 |