**概要设计说明书**

**《Swing学生成绩管理系统》**

**编写日期：2024.6.22**

**项目组：**

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **角色** |
| **202231060808** | **岳平** | **组长** |
| **202231060724** | **张磊** | **后端工程师** |
| **202231060732** | **刘如意** | **后端工程师** |
| **202231060725** | **刘定鹏** | **前端工程师** |
| **202231060730** | **钟云山** | **后端工程师** |
| **202231060731** | **秦铎洋** | **前端工程师** |

**修改日志**

|  |  |  |
| --- | --- | --- |
| 修改者 | 修改日期 | 备注说明 |
| 岳平 | 2024.6.22 | 完成基础说明 |
| 秦铎洋 | 2024.6.29 | 完成详细说明 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目 录**

[1. 引言 5](#_Toc469413310)

[1.1 编写目的 5](#_Toc469413311)

[1.2 定义 5](#_Toc469413312)

[1.3 参考资料 5](#_Toc469413313)

[2. 范围 5](#_Toc469413314)

[2.1 系统主要目标 5](#_Toc469413315)

[2.2 主要软件需求 5](#_Toc469413316)

[2.3 设计约束、限制 5](#_Toc469413317)

[3. 软件系统结构设计 6](#_Toc469413318)

[3.1 软件体系结构 6](#_Toc469413319)

[3.1.1 软件程序结构图 6](#_Toc469413320)

[3.1.2 模块描述 6](#_Toc469413321)

[3.2 功能需求追溯 7](#_Toc469413322)

[4. 数据设计 8](#_Toc469413323)

[5. 接口设计 8](#_Toc469413324)

[5.1 用户界面设计规则 8](#_Toc469413325)

[5.2 内部接口设计 8](#_Toc469413326)

[5.3 外部接口设计 8](#_Toc469413327)

[6. 出错处理设计 9](#_Toc469413328)

# 1.引言

## 编写目的

为软件开发团队提供一个高层次的技术文档，概述软件系统的架构和关键组件。它描述了系统的主要模块、它们之间的关系、接口定义、数据流以及主要的算法和处理流程。该文档用于指导详细设计阶段的工作，确保开发团队对系统的结构和功能有共同的理解。同时，概要设计说明书也是利益相关者沟通和决策的基础，有助于评估项目的可行性、风险和成本，确保项目按照既定目标和时间表推进。

## 定义

JSON：JavaScript 对象表示法，是一种轻量级的数据交换格式

XML：一种标记语言，用于描述数据的结构和语义

ORM：对象关系映射

SAML：安全声明标记语言，用于在不同的安全域之间交换认证和授权数据

LDAP：轻量目录访问协议

SSL/TLS：网络安全协议

## 参考资料

[1] 耿祥义、张跃平编著.《Java面向对象程序设计》.清华大学出版社，2015年

[2] 王家华.《软件工程》.东北大学出版社，2017年

[3] 魏善沛.《数据库系统概论》. 清华大学出版社，2016年

[4] 陈强.《精通Java开发技术》.清华大学出版社，2017年

# 范围

## 系统主要目标

提供一个自动化、集中化和用户友好的平台，用于高效地管理、录入、查询和分析学生成绩。系统将支持用户管理，数据维护，数据查询，报表打印等功能，同时确保信息的准确性和可访问性，以提高教育管理的效率和透明度

## 主要软件需求

高效的数据库管理以存储成绩数据；强大的用户界面以简化成绩录入和查询；精细的权限控制确保数据安全；灵活的报告生成工具以支持成绩分析；以及良好的系统兼容性和可扩展性，以适应不同教育环境和技术发展

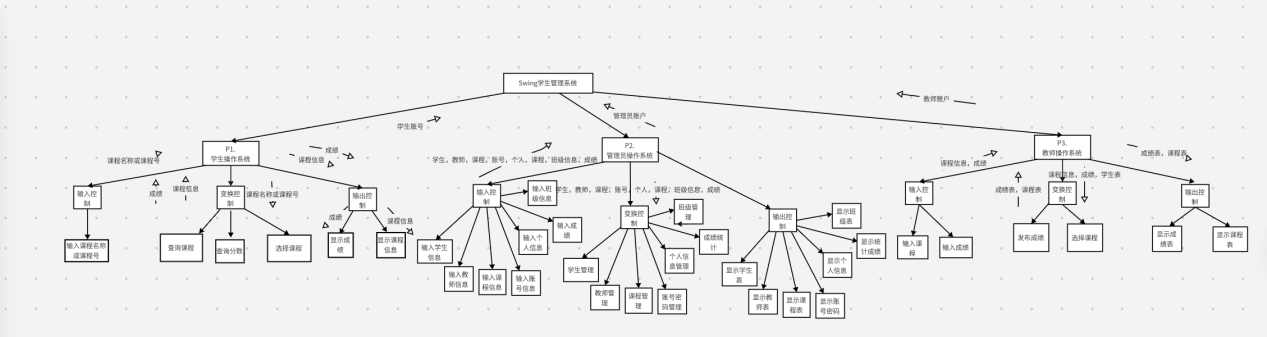
## 设计约束、限制

首先系统必须遵守相关的数据保护和隐私法规，确保学生信息的安全；其次系统需要兼容不同的操作系统、浏览器和设备，以支持广泛的用户访问；系统还应能够处理大量数据和高并发用户访问，保证响应速度和稳定性；设计也应考虑到不同用户群体的使用习惯，确保系统界面直观易用；项目预算也限制了系统开发的资源，包括人力、硬件和软件。同时系统需要保证时间效率、考虑资源和维护的可持续性、以及适应技术发展和用户需求的变化。这些限制要求系统设计必须平衡功能性、安全性、可用性和成本效益

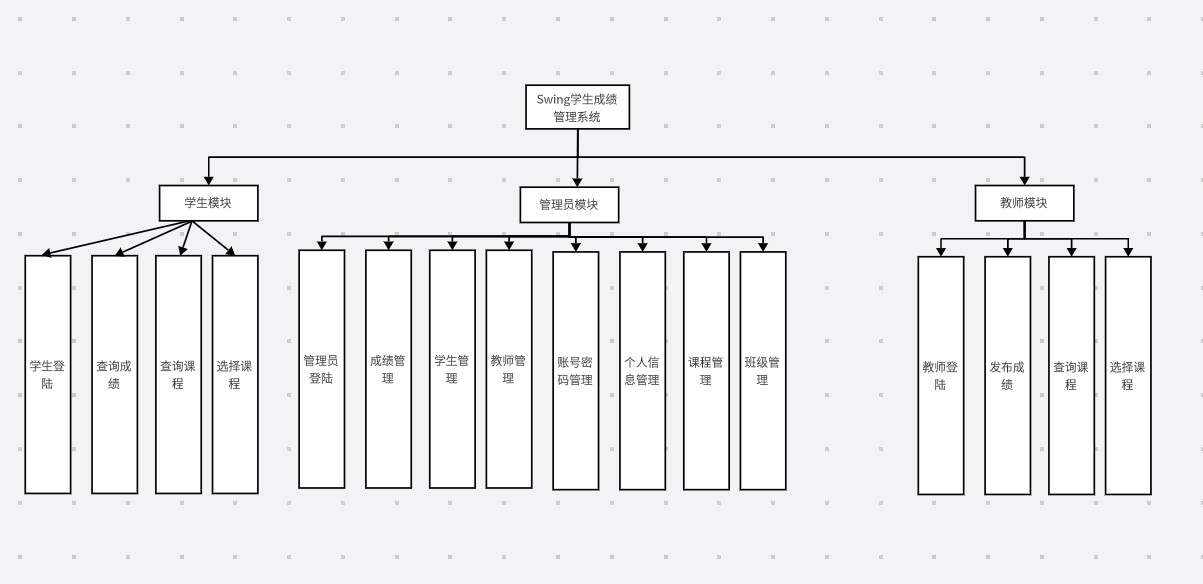
# 软件系统结构设计

## 软件体系结构

### 软件程序结构图



### 模块描述



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 模 块  名 称 | 学生模块 | | | 子系统名 称 | 成绩分析子系统等 | 系统  名称 | 学生成绩管理系统 |
| 接　口  说　明 | 输入 | 学生基本信息输入，成绩输入，教育背景输入等 | | | | | |
| 输出 | 学生基本信息输出，成绩报告输出，统计结果输出等 | | | | | |
| 功　能  说　明 | 可供学生登录，查询成绩，查询课程和选择课程等 | | | | | | |
| 运行环  境说明 | Intel Pentium 4 或更高版本的处理器，至少5GB内存，Windows 7 或更高版本的操作系统，IE 9 或更高版本浏览器等 | | | | | | |
| 调用关  系说明 | 调用模块 | | 数据库模块，用户认证模块，成绩管理模块等 | | | | |
| 被调用模块 | | 身份验证模块，教务管理模块等 | | | | |

## 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 模 块  名 称 | 管理员模块 | | | 子系统名 称 | 成绩管理子系统等 | 系统  名称 | 学生成绩管理系统 |
| 接　口  说　明 | 输入 | 管理员基本信息输入，学生成绩输入，课程输入等 | | | | | |
| 输出 | 修改管理员信息，删除用户等 | | | | | |
| 功　能  说　明 | 可供管理员登录，成绩管理，教师管理，课程管理等 | | | | | | |
| 运行环  境说明 | Intel Pentium 4 或更高版本的处理器，至少5GB内存，Windows 7 或更高版本的操作系统，IE 9 或更高版本浏览器等 | | | | | | |
| 调用关  系说明 | 调用模块 | | 用户管理模块，权限管理模块，数据管理模块等 | | | | |
| 被调用模块 | | 日志记录和监控等 | | | | |

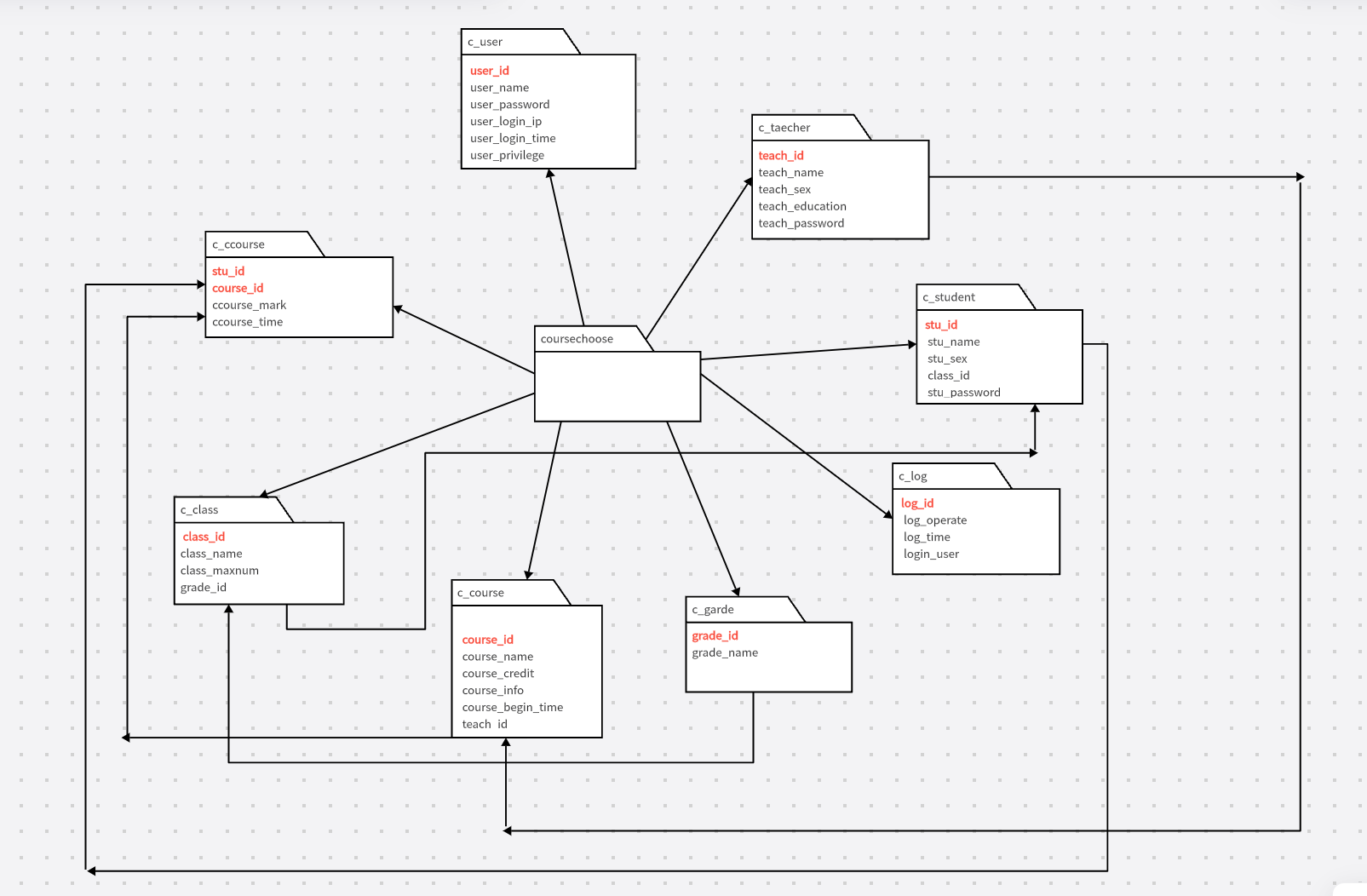
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 模 块  名 称 | 教师模块 | | | 子系统名 称 | 发布成绩子系统等 | 系统  名称 | 学生成绩管理系统 |
| 接　口  说　明 | 输入 | 教师基本信息输入等 | | | | | |
| 输出 | 课程统计结果输出等 | | | | | |
| 功　能  说　明 | 可供教师登录，查询课程信息，发布成绩等 | | | | | | |
| 运行环  境说明 | Intel Pentium 4 或更高版本的处理器，至少5GB内存，Windows 7 或更高版本的操作系统，IE 9 或更高版本浏览器等 | | | | | | |
| 调用关  系说明 | 调用模块 | | 学生信息管理模块，课程管理模块等 | | | | |
| 被调用模块 | | 管理员模块，选课模块等 | | | | |

## 功能需求追溯

|  |  |  |  |
| --- | --- | --- | --- |
|  | 学生模块 | 管理员模块 | 教师模块 |
| 学生选课 | √ |  |  |
| 课程查询 |  |  | √ |
| 教师成绩管理 |  |  | √ |
| 用户登录 | √ | √ | √ |
| 信息管理 |  | √ |  |

# 数据设计

（红色标注为主键）



# 接口设计

## 用户界面设计规则

确保界面直观且易于导航，采用清晰的布局和逻辑分组，以便用户快速找到所需功能。使用标准化的控件和一致的设计风格，以减少用户的学习成本。提供明确的反馈机制，如操作确认消息和错误提示，增强用户体验。确保界面适配不同分辨率和设备，提供响应式设计。遵循无障碍设计原则，使所有用户包括残障人士都能使用系统。保护用户数据安全，采用合适的认证和授权机制。最后，进行用户测试，收集反馈，不断迭代优化设计。

## 内部接口设计

软件内部模块间的接口安排需确保模块化设计和解耦合，以提高系统的可维护性和可扩展性。每个模块应有明确的功能定义和责任划分，通过定义清晰的API接口进行通信。接口应包括输入输出参数、数据类型、调用方法和返回值。使用标准化的数据交换格式如JSON或XML，确保数据一致性。同时，应实现错误处理机制，确保异常情况能够被捕捉并适当响应。接口文档应详尽，包括使用示例和版本控制信息，以便于开发和后续维护。此外，接口安全性设计也至关重要，需要通过认证和授权确保数据传输安全。

## 外部接口设计

用户界面：设计直观的用户界面，支持不同角色（如学生、教师、管理员）的交互，提供登录、数据输入、查询和报表查看等功能。

数据库接口：系统后端与数据库之间的接口应确保数据的安全性和完整性，使用SQL数据库，并实现ORM以简化数据操作。

硬件接口：如果系统需要与打印机、扫描仪等硬件设备交互，需定义硬件通信协议，如串行端口、USB或网络接口。

外部系统接口：对于需要与其他软件系统（如教务系统、图书馆系统）交换数据的情况，通过SAML实现集成。

网络接口：系统应支持通过Web服务或移动服务与外部网络通信，确保数据传输的安全性和效率。

安全接口：实现安全接口，包括用户认证（如LDAP）、数据加密（SSL/TLS）和安全日志记录。

系统维护接口：为系统管理员提供维护接口，包括系统监控、性能分析和故障排查工具。

文档和支持：提供详细的API文档、系统架构图和开发者指南，以便第三方开发者和维护人员理解和使用系统接口。

# 出错处理设计

E1：输入验证错误，补救措施：对所有用户输入进行严格的验证和过滤，确保只接受预期的输入格式，并对特殊字符进行转义或过滤

E2：数据库访问错误，补救措施：检查数据库连接配置和服务状态，优化查询语句，备份恢复等

E3：权限不足错误，补救措施：调整权限设置，使用root权限等

E4：系统资源错误，补救措施：增加虚拟内存，清理磁盘空间，系统还原等