

# 2025-2026-1 《Java 应用开发》期末大作业要求

## 应用架构性需求

- 采用 DAO 分层设计
- 自定义实现 Bean 容器和工厂，注入服务类对象和 DAO 类对象。
  - 要求使用**注解实现**，自定义注解名称为："Component\_姓名拼音"和"Autowired\_姓名拼音"。
    - Component 注解用于标注需要通过扫描被自动创建并管理的 Bean 对象
    - Autowired 注解用于实现构造函数注入
- 为业务层 Services 类生成动态代理。
  - 增加日志。日志格式：[时间] 完成了 [增删改查等] 操作。
  - 对于数据访问，增加事务支持。
- 支持数据持久化
  - 使用 JDBC 完成数据库访问，不能直接使用 Mybatis
  - 支持文件存储，使用 json 格式进行持久化。
  - 根据 Autowired 注解的内容在数据库和文件存储之间进行切换（提示，分别定义 File 和 DB 两种 DAO 的实现类）

组件	作用	对应架构要求
自定义注解	用于标记被管理的 Bean 和实现依赖注入。	Component 注解用于标注需要通过扫描被自动创建并管理的 Bean 对象 Autowired 注解用于实现构造函数注入
自定义容器	负责加载、管理和注入所有 Bean。	自定义实现 Bean 容器和工厂（基于注解方式）
DAO 层	实现数据的持久化操作。	采用 DAO 分层设计、支持数据持久化（JDBC /Json File）
Service 层	实现业务逻辑。 使用 Stream API 查询和统计	DAO 分层设计、动态代理生成（日志增强）
动态代理	为 Service 层增强横切关注点（如日志）。	为业务层 Services 类生成动态代理

## 应用功能性需求

- 系统用户登录
  - 用户身份包括教务管理人员、教师和学生。
  - 需要验证密码
- 学生管理
  - 至少包括 100 名学生信息，且必须含有本人姓名的学生，并在报告和演示中强调
  - 专业班级，至少包括 2 个专业 4 个班级。
  - 学生可以查询课程成绩
- 教师管理
  - 至少包括 5 名教师信息和 1 名教务管理人员。
- 课程管理
  - 至少包括 10 门课程信息，且必须含《Java 应用开发》课程
- 开课功能
  - 教学管理人员可以开课（要求选课表中具有开课学期的列，例如 2025-2026-1，任课教师，班级等信息）
- 任课教师课程成绩管理
  - 提供录入成绩的操作
  - **【选做】**可以批量导入成绩
- 相关查询（系统中至少包括 3 个班、30 名学生，3 名教师的 5 门课程的成绩）
  - 学生个人成绩查询
  - 查询指定教师教授的所有课程
  - 查询某班级某课程的成绩列表
  - 计算指定课程的平均成绩和及格率
  - 查询指定班级的学分绩点排名（ $\text{绩点} = (\text{成绩} - 50) / 10 * \text{学分}$ ）
  - 两次挂科达到 2 门及以上的学生名单

- 教师工作量分析：计算每位教师在当前学期所教的学生总人次（人次 = 所教班级人数\*所教课程门数），并按人次进行降序排名
- 学霸画像，找出满足以下条件的学生： 1. 所有课程成绩均不低于 85 分； 2. 至少修够 4 门课程
- 界面不限，命令行、GUI、Web 均可以
- 其他功能不限，例如数据导入、导出。
- 开发建议
  - 逐步完善，例如先实现学生管理，然后实现课程管理，最后再实现学生排课

建议三种角色功能划分如下

---

### 1. 教务管理人员 (Admin)

教务管理人员是系统的核心管理者，拥有最高权限，负责维护系统的基础数据和执行管理分析。

**基础功能（管理与维护）：**

- **登录：** 使用教务管理人员身份登录系统。
- **学生管理：** 负责所有学生信息的增、删、改、查（CRUD），包括管理学生所属的专业和班级。
- **【选做】教师管理：** 负责所有教师信息（以及其他教务人员账户）的增、删、改、查。
- **【选做】课程管理：** 负责系统中所有课程（如《Java 应用开发》）的增、删、改、查，包括设置课程名称、学分等。
- **开课功能：** 核心功能。为特定学期（如 2025-2026-1）指派课程、任课教师和授课班级，创建“开课实例”。

**查询与统计功能（全局数据分析）：**

- **全局查询：** 拥有最高查询权限，可以查询任意班级、任意课程、任意教师的成绩列表和授课情况。
- **学业预警：** 查询挂科情况，例如“两次挂科达到 2 门及以上的学生名单”，以便进行学业警告。
- **教学质量评估：** 计算任意指定课程的平均成绩和及格率。
- **学分绩点排名：** 查询任意指定班级的学分绩点排名。

- **教师工作量分析：** 统计所有教师的工作量（例如按学生总人次排名），用于绩效考核。
  - **学生画像分析：** 筛选“学霸画像”（例如所有课程不低于 85 分且修满 4 门课的学生），用于评奖评优。
- 

## 2. 教师 (Teacher)

教师是系统的核心使用者，主要负责与自己所教授课程相关的教学活动，特别是成绩录入。

**基础功能（教学核心）：**

- **登录：** 使用教师身份登录系统。
- **成绩录入：** 为自己所教授的班级和课程**录入学生成绩**。
- **【选做】成绩导入：** 选做 批量导入指定班级和课程的成绩。
- **成绩修改：** 在规定时间内修改已录入的成绩。

**查询与统计功能（个人数据相关）：**

- **查询个人课表：** 查询自己在本学期教授的所有课程和班级列表。
  - **查询班级成绩：** 查询自己所教班级的特定课程成绩列表（用于核对）。
  - **课程统计：** 计算自己所教课程的平均成绩和及格率。
- 

## 3. 学生 (Student)

学生是系统的主要用户，权限最低，主要集中在查询与自己学业相关的信息。

**基础功能（个人信息查询）：**

- **登录：** 使用学生身份登录系统。
- **查询个人成绩： 核心功能。** 查询自己所有已修课程的名称、学期、学分和最终成绩。

**查询与统计功能（自我评估）：**

- **查询个人绩点：**（隐含需求）查看自己的学分绩点（GPA）和在班级中的大致排名。

一种简化的表/类设计

表名	主要字段	对应实体
Users	id, username, password, role (Admin/Teacher/Student), name, className, majorName	用户/人员
Courses	id, courseName, credits, teacherId, className, semester, classSize	开课/课程
Grades	id, studentId, offeringId, score	成绩