校园卡管理系统架构设计评估与改进

一、逻辑视图评估（类设计）

1. 当前问题​​

①类职责混乱：用户类同时处理身份认证、消费记录等。

②继承结构不合理：管理员/学生采用继承关系导致代码重复。

③数据保护不足：关键字段如密码未封装，直接暴露属性。

2. 解决方案​​

①采用组合模式重构：

创建独立的AuthenticationService处理登录认证，用户类仅保留基础属性，通过关联关系使用其他服务

②引入策略模式：

支付方式（现金/扫码）通过PaymentStrategy接口实现，交易记录使用组合关系而非继承。

二、进程视图问题与改进

1. 当前问题​​

①进程通信不安全：共享内存未加锁，日志服务丢包

2. 解决方案​​

①通信机制优化：

进程间通信改用gRPC协议

②熔断机制：

数据库访问添加重试策略（指数退避）

三、开发视图问题与改进

1. 当前问题​​

①模块边界模糊：数据库操作分散在20+个类

②构建效率低下：全量编译耗时过长

③版本管理混乱：数据库脚本与代码不同步

2. 解决方案​​

①分层重构：

（1）数据访问层（统一SQL操作）（2）接口层（REST/Qt界面）

②构建优化：

核心模块拆分为动态库、实现增量编译（CMake配置）

③版本控制：

数据库迁移纳入Git管理、使用Flyway维护脚本版本

四、物理视图问题与改进

1. 当前问题​​

①单点故障：数据库无备份

②扩展困难：硬件不支持水平扩展

2. 解决方案​​

①高可用部署：数据库主从复制（MySQL Group Replication）

②容器化准备：

制作Docker镜像

编写Kubernetes部署描述文件