

# | 个人项目总结与课程反思报告

## | 一、报告定位

本报告总结我在《软件工程》课程项目“衔筑云圈 (CoSphere)”中的个人参与经历与学习体会，聚焦于后端开发实践与团队协作，反思项目经验并提出个人成长与课程建议。

---

## | 二、核心内容

### | 1. 项目概述

- **个人职责：**

在团队中，我主要负责后端的用户模块和社交模块（即“修己斋”与“墨络轩”部分）的设计与开发工作，涵盖用户注册登录、资料修改、好友关系管理、动态等功能。同时参与接口文档编写、前后端联调、部分测试用例制定。

- **技术栈与工具：**

- 后端开发：Java + Spring Boot, MyBatis
  - 数据库：MySQL，部分使用Redis缓存
  - 工具：Postman、Git、GitHub、Navicat、飞书
  - 模块开发理念：注重模块解耦、接口清晰、异常处理完整，按照MVC模式规范设计Controller、Service、Mapper层。
- 

### | 2. 项目实施过程

## 需求与设计

我参与了用户与社交模块的功能拆分与UML建模，结合实际使用场景设计了：

- 用户体系：支持邮箱注册、登录、密码加密、用户信息展示与修改；
- 好友体系：包含好友申请、通过/拒绝、删除、双向好友关系表设计；
- 社交互动接口：为动态模块提供用户基本信息、好友列表与社交互动支持。

数据库方面，我参与设计了 `user`、`friendship` 等核心表，考虑性能与扩展性，使用联合索引优化好友查询效率。

## 开发与实现

- 用户模块：
  - 实现了基于JWT的Token登录验证机制，提升安全性；
  - 对用户数据做脱敏处理，保障隐私安全。
- 社交模块：
  - 实现了加好友请求机制，包括申请、接受、拒绝的三状态流转；
  - 实现了好友列表拉取与分组、附近的人初步逻辑；
  - 解决了双向好友关系建模问题。

## 联调与测试

- 使用飞书统一管理接口文档，配合前端进行联调，针对字段约定等做了多轮修正；
  - 使用Postman进行接口测试，编写涵盖正常流、异常流的用例，发现并修复了多个bug，如“重复添加好友未限制”、“状态未回滚”等；
-

### 3. 贡献与达成度

- **个人代码量贡献**：约占后端代码的30%，涵盖用户模块与社交模块全部接口及逻辑实现；
  - **功能达成度**：实现了预期的注册登录、好友管理、信息维护等功能，API接口稳定，顺利完成与动态模块、目标模块的集成对接。
  - **质量保障**：接口遵循RESTful规范，参数与异常处理统一，联调阶段bug较少，具备良好的扩展性与维护性。
- 

### 4. 遇到的问题与改进

- **事务问题复杂**：如任务完成失败需回滚，我查阅事务管理文档，调整为声明式事务方式，确保一致性。
  - **请求无状态问题**：结合JWT及ThreadLocal，封装请求上下文类BaseContext，解决了请求无状态的问题。
- 

### 5. 课程反思

- **个人成长**：  
通过本项目，我系统掌握了Spring Boot后端开发流程，尤其在用户认证、接口设计、数据库建模方面获得了实践经验。同时也提升了团队沟通能力和协作效率，在实际工作中更加注重接口契约和模块解耦。
- **工程认知提升**：  
理解了从“一个人写代码”到“多人协作写产品”之间的差距，对项目管理、接口规范、需求分析等非编码因素有了更深刻的认识。意识到规范、可维护、可测试的代码才是好代码。

- **课程建议：**

希望课程能在早期加入项目部署与运维相关内容，比如**Spring Boot**部署、日志分析、数据库迁移脚本等。同时希望鼓励更多使用真实第三方服务（如**OAuth**登录、图床、短信服务等），模拟商业项目更真实的开发流程。