

武汉理工大学

《现代软件工程学》
团队报告

目录

知响社交圈项目报告.....	2
1 开发报告.....	2
1.1 项目概述.....	2
1.1.1 项目简介.....	2
1.1.2 项目背景.....	2
1.1.3 项目意义.....	2
1.2 工作成果.....	3
1.2.1 需求分析.....	3
1.2.2 软件建模.....	3
1.2.3 软件设计.....	6
1.2.4 编码和测试.....	7
2 团队软件过程.....	7
2.1 团队目标.....	7
2.1.1 技术目标.....	7
2.1.2 产品目标.....	8
2.1.3 团队成长目标.....	8
2.2 团队分工.....	9
2.2.1 项目管理与测试 - 胡姗.....	9
2.2.2 UI设计与产品规划 - 方骏.....	9
2.2.3 技术架构与核心开发 - 王文超.....	10
2.2.4 数据库设计与开发 - 黄川.....	10
2.3 项目周计划与实现情况.....	10
3 项目展示.....	12
3.1 展示成果.....	12
3.2 Q&A.....	14
安装说明.....	15
默认账号.....	15

知响社交圈项目报告

1 开发报告

1.1 项目概述

1.1.1 项目简介

知享社交圈是一个创新的论坛社区平台，旨在构建一个知识共享和互助的在线环境。该平台鼓励用户分享专业知识、经验和见解，同时寻求他人的帮助和建议。我们特别重视为用户提供一个安全、匿名的交流空间，以促进知识的自由流通和思想的碰撞。为了维护社区的质量和专业化，我们采用了邀请制注册方式，确保每位成员都能为社区带来价值。

项目使用的技术栈如 Table 1：

Table 1 技术栈

技术领域	技术/框架/库	版本
后端	Spring Boot、MyBatis	2.2.6
前端	Bootstrap、jQuery	3.3.6、2.0
模板引擎	Thymeleaf	-
数据库	MySQL	-
其他	Druid连接池	-

1.1.2 项目背景

在快速变化的现代社会中，知识的更新和信息的流通速度日益加快。人们越来越需要一个平台，能够快速获取所需信息，分享个人见解，并与他人进行深入交流。知享社交圈应运而生，旨在满足这一需求，提供一个集知识分享、问题解答和经验交流于一体的社交平台。

1.1.3 项目意义

知享社交圈不仅为用户提供了一个交流思想和解决问题的空间，还通过匿名机制保护了用户的隐私，鼓励更开放和真诚的交流。此外，平台的邀请制注册策略有助于吸引和聚集行业内的专业人士，提升社区内容的专业性和可靠性。

1.2 工作成果

1.2.1 需求分析

系统界面简洁直观，易于使用，并且具有较高的性能与可扩展性。主要包括以下几种功能：

用户管理：支持用户的注册(邀请码机制)、登录验证、个人信息修改、权限管理等。

帖子管理：支持用户发布、编辑、删除帖子，其他用户可以进行评论。

文件管理：提供文件上传、文件下载、文件删除、容量控制、权限管理等网盘功能。

管理员功能：支持论坛内容的管理、用户管理、帖子管理、邀请码管理等。

1.2.2 软件建模

用例图：

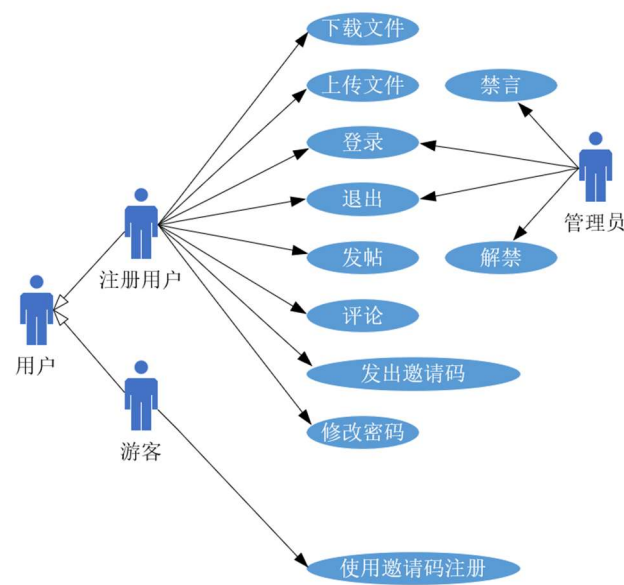


Figure 1 用例图

类图抽象：

Table 2 类图抽象

用例	实体类	控制类
注册用户、管理员	User	User Controller
发帖	Post	Post Controller
回复	Reply	Reply Controller
上传文件、下载文件	File	Net disk Controller
生成邀请码、注册	Invite code	
登录、退出	Page	

类图:

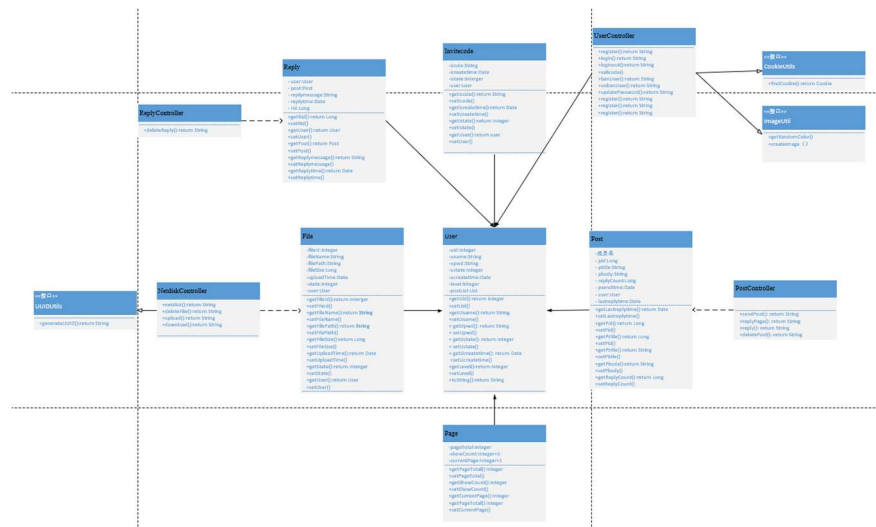


Figure 2 类图

时序图:

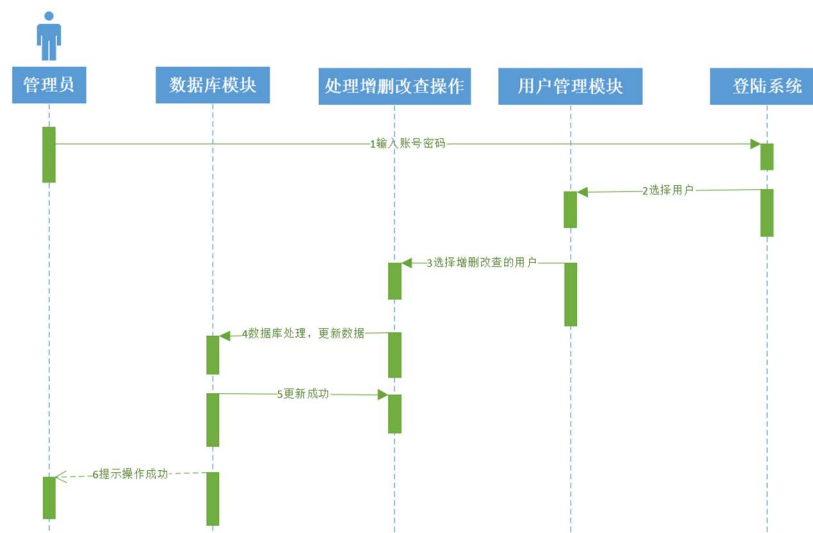


Figure 3 管理员时序图

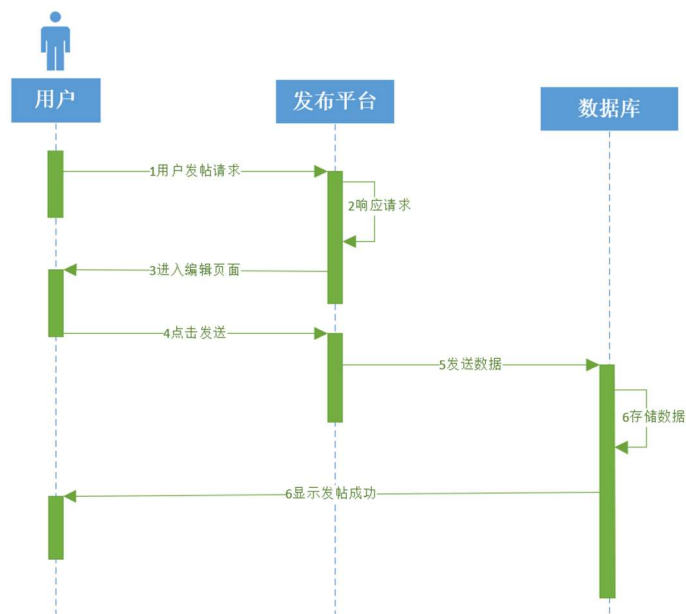


Figure 4 发帖时序图

对象图:

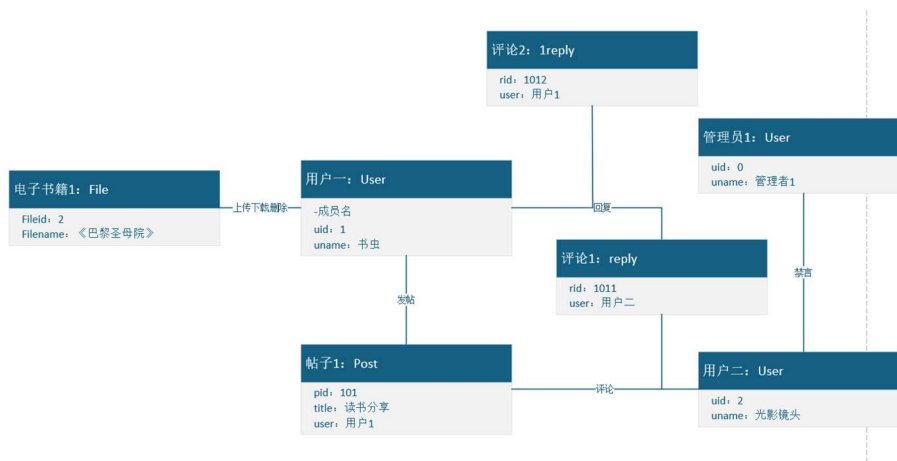


Figure 5 对象图

流程图:

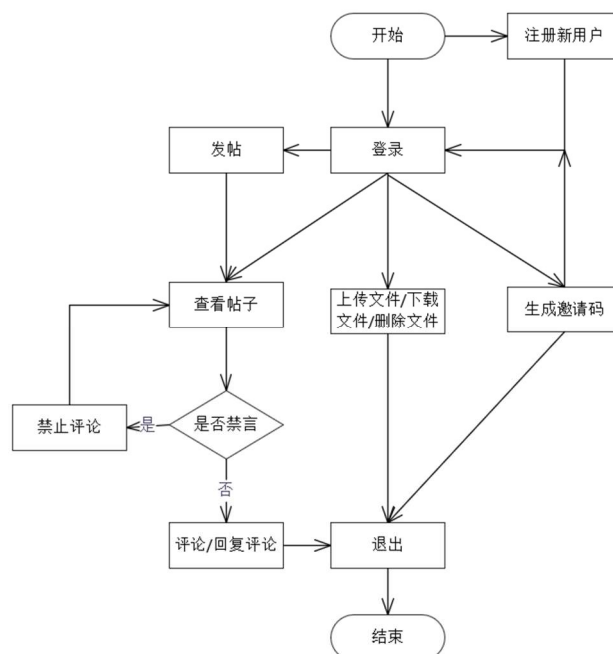


Figure 6 流程图

数据流图:

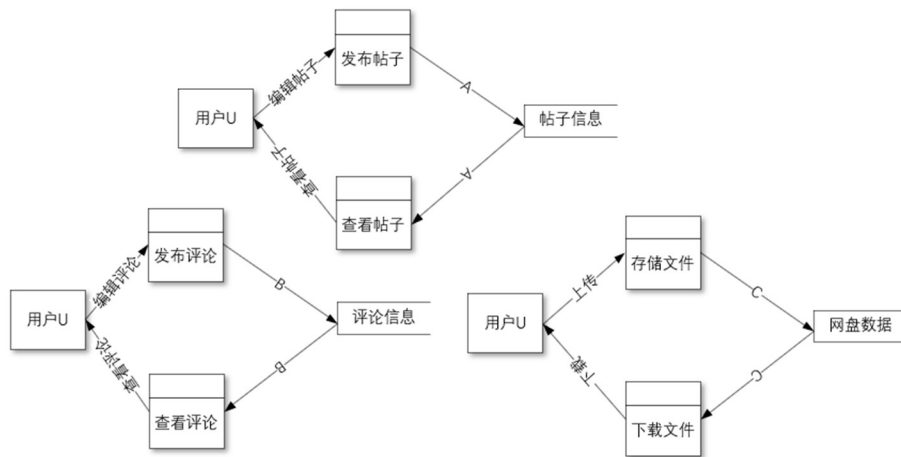


Figure 7 数据流图

1.2.3 软件设计

架构设计: - 采用标准三层架构: - 表现层(Controller) - 业务层(Service) - 数据访问层(Mapper)

数据库设计: - 主要实体: User(用户)、Post(帖子)、Reply(回复)、File(文件)、Invitecode(邀请码)

UI 设计展示了系统的界面布局 and 用户交互。页面的设计考虑用户的使用习惯，保持界面简洁、易于操作。以下为本项目的主要前端设计:



Figure 8 应用主页



Figure 8 登录和注册页面



Figure 9 网盘页面

接口设计: RESTful 风格 API - 统一响应格式 - 安全认证机制

1.2.4 编码和测试

编码规范: - 遵循阿里巴巴 Java 开发规范 - 统一异常处理 - 详细代码注释

测试策略: - 单元测试 - 接口测试 - 系统测试

2 团队软件过程

我们的团队目标分为三个维度：技术目标、产品目标和团队成长目标。旨在构建一个技术先进、用户友好且安全可靠的知享社交圈平台。

2.1 团队目标

2.1.1 技术目标

深入理解 Spring Boot 框架的核心特性：

我们深入学习了 Spring Boot 框架的核心特性，包括但不限于自动配置、起步依赖、模块化设计、依赖注入以及面向切面编程。通过深入理解这些特性，才能

够构建更高效、更灵活的应用程序，并充分利用 Spring 生态系统的强大功能。

掌握 MyBatis 的动态 SQL、缓存机制和性能优化：

为了提高数据访问层的性能和灵活性，我们学习了 MyBatis 的动态 SQL 生成、二级缓存机制以及查询优化技术。这些功能帮助我们实现更高效地处理复杂查询，减少数据库负载，提升系统响应速度。

提升前端技术栈应用能力，熟练运用 Bootstrap 和 jQuery：

前端团队学习了 Bootstrap 框架和 jQuery 库，利用这些工具快速构建响应式、美观的用户界面。

建立完整的测试体系：

我们学习并建立包括单元测试、集成测试和性能测试在内的完整测试体系，确保系统在各个层面都能达到高标准的质量要求。

2.1.2 产品目标

（1）打造安全可靠的知识分享社区

为了保证社区的高质量，我们引入邀请码注册机制，确保新用户具有一定的专业背景和社区贡献意愿。这将有助于维护社区的积极氛围和知识分享的深度。此外，我们建立一套高效的内容审核流程，包括自动审核和人工审核相结合的方式，确保社区内容的准确性和合规性。

（2）提供优质的用户体验

遵循“少即是多”的设计理念，我们确保用户界面简洁、直观，操作流程清晰。未来我们也将不断优化界面设计，确保用户可以在 3 次点击内到达任何功能页面。

为了提供流畅的用户体验，我们将优化前端加载和后端处理流程，确保系统响应迅速，页面加载时间控制在 2 秒以内。这将显著提升用户的操作满意度。

我们将对数据库进行优化，包括索引优化、查询优化和缓存策略，确保数据库查询响应时间控制在 100ms 以内，进一步提升系统的整体性能。

（3）构建完整的社区生态

构建完整的用户权限体系，实现基于角色的访问控制：我们设计一套灵活的用户权限体系，支持基于角色的访问控制，确保不同用户在社区中的权限和职责明确。此外，我们将建立用户积分和等级体系，激励用户积极参与社区互动。

集成文件分享功能，打造安全可靠的文件管理系统：我们将集成一个功能完善的文件管理系统，支持多种格式文件的上传、下载和管理。通过文件权限控制和安全传输机制，确保用户文件的安全性和可靠性。

2.1.3 团队成长目标

（1）熟练使用敏捷开发方法

采用敏捷开发方法，提高迭代速度：我们采用敏捷开发方法，Scrum，通过短周期的迭代快速交付功能，提高项目的灵活性和响应速度。同时，我们定期进行回顾会议，持续改进开发流程。

加强需求管理，提高需求分析能力：未来提高需求分析和任务分解能力，确保每个需求都能被准确理解并有效实现。学习采用用户故事、用例等工具，确保需求的全面性和可追踪性。

（2）提高需求分析和任务分解能力

有效分解任务，确保开发进度：我们采用任务分解技术，将复杂的需求分解为多个小任务，确保每个任务都能在预定时间内完成。通过任务分解提高团队的开发效率和项目管理能力。

（3）加强团队协作和沟通效率

建立高效的沟通机制：通过高效的沟通机制，确保团队成员之间的信息畅通。通过定期的团队会议、在线协作工具和文档管理来减少信息孤岛，提高团队协作效率。

促进团队成员之间的技术交流和知识共享：我们定期组织技术分享会和培训（每周一次），促进团队成员之间的技术交流和知识共享。

2.2 团队分工

2.2.1 项目管理与测试 - 胡姗

职责范围：

- 项目整体规划与进度管理
- 任务分配与进度跟踪
- 测试计划制定与执行
- 质量保证与问题跟踪

具体工作：

胡姗作为项目经理，首先制定了详细的项目计划表，将项目分为需求分析、设计开发、测试部署等多个阶段。并且每周组织团队会议，检查项目进度并及时调整计划。在测试方面，还设计了完整的测试用例集，覆盖了所有核心功能，并亲自执行测试工作。如果发现问题，与开发团队密切配合，确保问题得到及时解决。

2.2.2 UI 设计与产品规划 - 方骏

职责范围：

- 用户界面设计
- 用户体验优化
- 产品需求分析
- 功能规划与迭代

具体工作：

在 UI 设计过程中，方骏首先进行了用户研究，创建了用户画像和使用场景。基于这些分析，他使用 Figma 设计了整套 UI 界面原型，并与团队成员多次讨论

优化。还特别关注了用户体验，确保界面简洁直观，操作流程顺畅。

2.2.3 技术架构与核心开发 - 王文超

职责范围：

- 系统架构设计
- 前端开发实现
- 核心功能开发
- 性能优化

具体工作：

王文超首先设计了基于 Spring Boot 的整体技术架构，制定了详细的类图和组件关系图。在前端开发中，他选择了 Bootstrap 和 jQuery 框架，实现了响应式布局和流畅的交互体验。还重点开发了用户认证、帖子管理等核心模块，并实现了前后端数据交互。通过引入缓存机制和代码优化，将系统响应时间控制在预期范围内。同时还建立了代码审查机制，确保代码质量符合规范。

2.2.4 数据库设计与开发 - 黄川

职责范围：

- 数据库架构设计
- 后端接口开发
- 数据存储优化
- 安全性保障

具体工作：

在数据库设计阶段，黄川采用 ER 图设计了整个数据库结构，建立了完整的表关系。我使用 MyBatis 框架实现了数据访问层，开发了高效的 SQL 语句和存储过程。为了提高查询性能，他建立了适当的索引，并通过分页机制优化大数据量查询。在安全方面，实现了数据加密存储和访问权限控制，并建立了定期备份机制。

2.3 项目周计划与实现情况

第一周

计划实现：

- 需求分析与文档编写
- 技术架构设计
- 数据库设计
- UI 原型设计

实际实现：

第一周我们完成了预期的所有任务。首先召开需求分析会议，由方骏主导完成了用户需求调研和功能规划，形成了详细的需求文档。在技术选型上，王文超建议采用 Spring Boot + MyBatis 的主架构，并完成了系统架构设计文档。黄川完

成了数据库表设计，包括用户表、帖子表、评论表等核心表结构的设计与评审。方骏通过 Figma 工具完成了系统 UI 原型设计，经过团队多轮讨论后确定了最终版本。整体进度符合预期，为后续开发打下了良好基础。

第二周

计划实现：

- 开发环境搭建
- 基础框架整合
- 数据库实现
- 基础功能开发

实际实现：

第二周我们主要完成了项目基础架构的搭建。王文超首先创建了 Spring Boot 项目，完成了 MyBatis、Thymeleaf 等框架的整合。在此过程中遇到了版本兼容性问题，经过调试最终解决。黄川完成了数据库表的创建和基础数据的导入，并实现了基础的 DAO 层代码。同时，我们建立了 Git 版本控制系统，制定了代码规范和提交规范。项目骨架搭建完成后，开始了用户模块的基础功能开发，包括注册和登录界面的实现。"

第三、四周

计划实现：

- 用户管理模块
- 帖子管理模块
- 评论功能
- 权限控制

实际实现：

这两周是核心功能开发阶段。首先完成了用户管理模块，包括基于 Spring Security 的权限控制系统实现。在帖子管理模块中，实现了帖子的 CRUD 操作和分页展示。评论功能的开发过程中遇到了多级评论展示的技术难题，通过递归算法最终解决。期间发现了几个数据库查询性能问题，黄川通过优化 SQL 和添加适当的索引进行了改善。方骏持续优化界面交互体验，使整体操作更加流畅。"

第五周

计划实现：

- 网盘功能
- 文件管理
- 性能优化
- 缓存实现

实际实现：

第五周主要聚焦于网盘功能的开发。我们实现了文件上传下载功能，通过配置文件设置了每个用户 2G 的默认空间限制。在文件存储方面，采用了分级目录

存储方案，避免单一目录下文件过多的问题。同时引入了 Druid 连接池优化数据库访问性能，并实现了简单的缓存机制提升系统响应速度。期间遇到了大文件上传的性能问题，通过分片上传技术成功解决。"

第六周

计划实现：

- 系统测试
- 问题修复
- 系统部署
- 文档完善

实际实现：

最后一周主要进行系统测试和部署。胡姗设计了详细的测试用例，覆盖了所有核心功能。测试过程中发现了一些边界条件下的 bug，开发团队及时进行了修复。完成测试后，我们部署了生产环境，进行了数据迁移和系统调优。同时完成了用户手册和技术文档的编写。最后进行了项目总结会议，复盘了整个开发过程，并制定了后续的维护计划。所有工作按期完成，系统运行稳定。

3 项目展示

3.1 展示成果

首页：



Figure 10 首页

登录注册功能：

登录

用户

用户名

密码

密码

验证

验证码

看不清，点击切换

关闭

登录

注册

用户昵称

用户名

用户密码

密码

确认密码

确认密码

邀请代码

邀请代码

验证代码

验证码

看不清，点击切换

关闭

注册

Figure 11 登录注册功能

管理员后台页面：

知享社交圈

后台

网盘

发帖

[admin]个人管理

退出

邀请码管理

邀请码列表

生成邀请码

邀请码	生成时间	激活状态	用户	操作
INV10001XYZ	2024-11-10 23:00:00	已激活	hushan	删除
INV10002ABC	2024-11-12 00:00:00	已激活	知识分享者A	删除
0D95369CC3134287B5F24E8699B5481B	2024-12-04 10:39:06	未激活		删除
170E08AFE1944536BEC2F760905D139B	2024-12-04 12:09:56	未激活		删除
93422A0FE65544F7B147B4560101FEE3	2024-12-04 10:38:54	未激活		删除
INV10003DEF	2024-11-12 20:00:00	未激活		删除

用户管理

用户列表

用户id	用户名	创建时间	状态	操作
1	admin	2020-08-12 18:57:45	正常	
2	匿名用户2	2024-11-02 19:00:00	正常	
3	匿名用户3	2024-11-03 20:00:00	禁言	
4	知识分享者A	2024-11-05 22:00:00	正常	
5	小白用户	2024-11-06 23:00:00	禁言	解禁
6	匿名用户1	2024-11-01 18:00:00	正常	

Figure 12 管理员后台页面

网盘功能界面：

知享社交圈

后台

网盘

发帖

[admin]个人管理

退出

网盘 剩余空间 1.95116895530Gb

文件列表

上传文件

文件名	文件大小	上传时间	
前端开发入门.mp4	50.00000000 MB	2024-11-14 19:00:00	下载 删除
Log.txt	0.00298977 MB	2024-11-15 17:14:02	下载 删除

Figure 13 网盘功能界面

发帖功能界面：

知享社交圈

后台

网盘

发帖

[admin]个人管理

退出

网盘 剩余空间 1.95116895530Gb

文件列表

上传文件

文件名	
前端开发入门.mp4	下载 删除
Log.txt	下载 删除

发布帖子

帖子标题

帖子内容

关闭

发送

Figure 14 发帖功能界面 1

发帖功能界面：

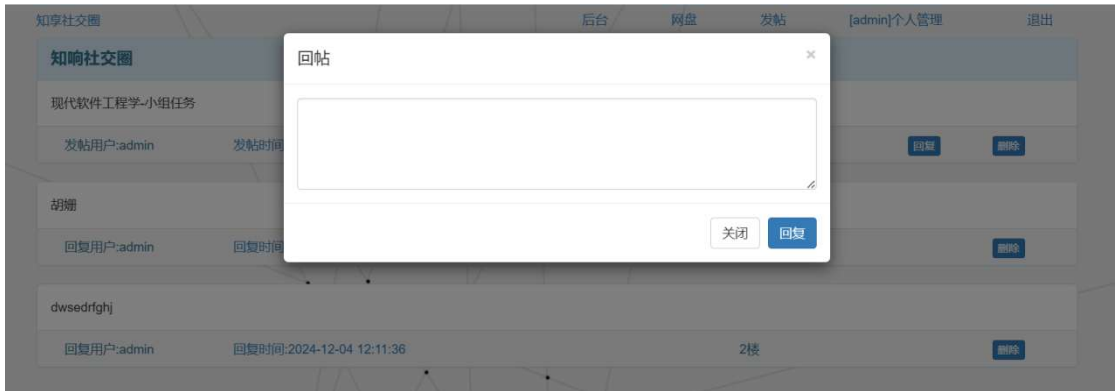


Figure 14 发帖功能界面 2

3.2 Q&A

Group 16

Q1: Git 工具在你们整个项目开发过程中发挥了哪些作用？你们是如何使用它的？

A1: 在我们的项目开发中，Git 主要用于版本控制、团队协作和代码备份。通过分支管理，每个成员独立开发模块，完成后通过 PR 进行代码审查并合并到主分支。我们定期提交代码，记录变更历史，借助 GitHub 实现代码共享和备份。同时，通过 Git 的回滚功能快速修复问题，配合敏捷开发流程高效推进项目进度。

Group 17

Q2: 可以详细讲讲邀请码功能的设计逻辑，比如如何生成、失效机制。

A2: 邀请码功能通过随机生成一串唯一字符串并存储在数据库中，绑定生成者，并设置有效期和使用状态。验证时校验邀请码的存在性、有效性和使用状态，确保权限匹配后完成绑定或注册操作。失效机制包括时间限制、使用次数限制，保障系统安全性与效率。

Group 18.

Q3: 你们项目的测试工作是如何开展的，主要做了哪些部分的测试？

A3: 我们的开发阶段和测试阶段是同时进行的，针对每个功能模块，开发人员在编写代码时就完成各个单元的单元测试，最后整合后统一进行接口测试、系统测试，同时也进行了一些界面的前端测试。

安装说明

1 环境要求

- JDK 1.8+
- MySQL 5.7+
- Maven 3.0+

2 安装步骤

- 导入数据库(bbs.sql)
- 修改配置文件
- 运行

`mvn spring-boot:run`

3 默认账号

管理员邀请码: INV10001XYZ