# 全国大学生“区块链+”应用比赛

设计与开发文档

作品编号：\_\_\_\_\_\_\_\_\_\_\_\_\_

作品名称：\_\_\_\_\_\_\_\_\_\_\_\_\_

作者：\_\_\_\_\_\_\_\_\_\_\_\_\_

填写日期：\_\_\_\_\_\_\_\_\_\_\_\_\_

# 需求分析

**开发背景**

随着随着数字技术的进步和互联网的发展，数字内容的生产和传播变得前所未有的便捷，电子书、数字音乐、流媒体视频等数字作品逐渐取代了传统的实体作品。这种变化带来了新的版权挑战，数字内容的非法复制和传播、盗版、版权侵权等问题的频发。传统版权保护弊端也随着市场的不断发展显现出来。

传统版权存证方式容易被篡改、复制且成本高昂，亟需更高效的解决方案。区块链技术的不可篡改性、去中心化和透明性，能够为版权存证提供更加安全和可靠的保护机制。因此，开发基于区块链的版权存证系统成为必要。

**竞品分析**

当前市场上已有类似的区块链版权存证产品，如Mycelia和Ascribe等。这些产品多关注于特定领域，如音乐或数字艺术。本项目的目标是开发一个更通用的版权存证系统，对标对象为Mycelia。

|  |  |  |
| --- | --- | --- |
| **维度** | **本作品** | **Mycelia** |
| **应用场景** | 泛数字内容版权保护 | 音乐版权保护 |
| **主要功能** | 版权确权、存证、授权管理 | 音乐版权确权与授权 |
| **用户群体** | 广泛的数字内容创作者 | 音乐创作者 |
| **技术基础** | 多链兼容的区块链架构 | 以太坊区块链 |
| **扩展性** | 高扩展性，支持多种内容格式 | 中等扩展性，集中于音乐 |
| **智能合约** | 全自动化授权与交易功能 | 基础授权管理 |

**面向用户**

本作品面向广泛的数字内容创作者、版权持有人、法律顾问以及数字内容平台运营者。通过简化版权确权、存证和管理流程，提供高效、透明的版权保护和管理工具。

**主要功能**

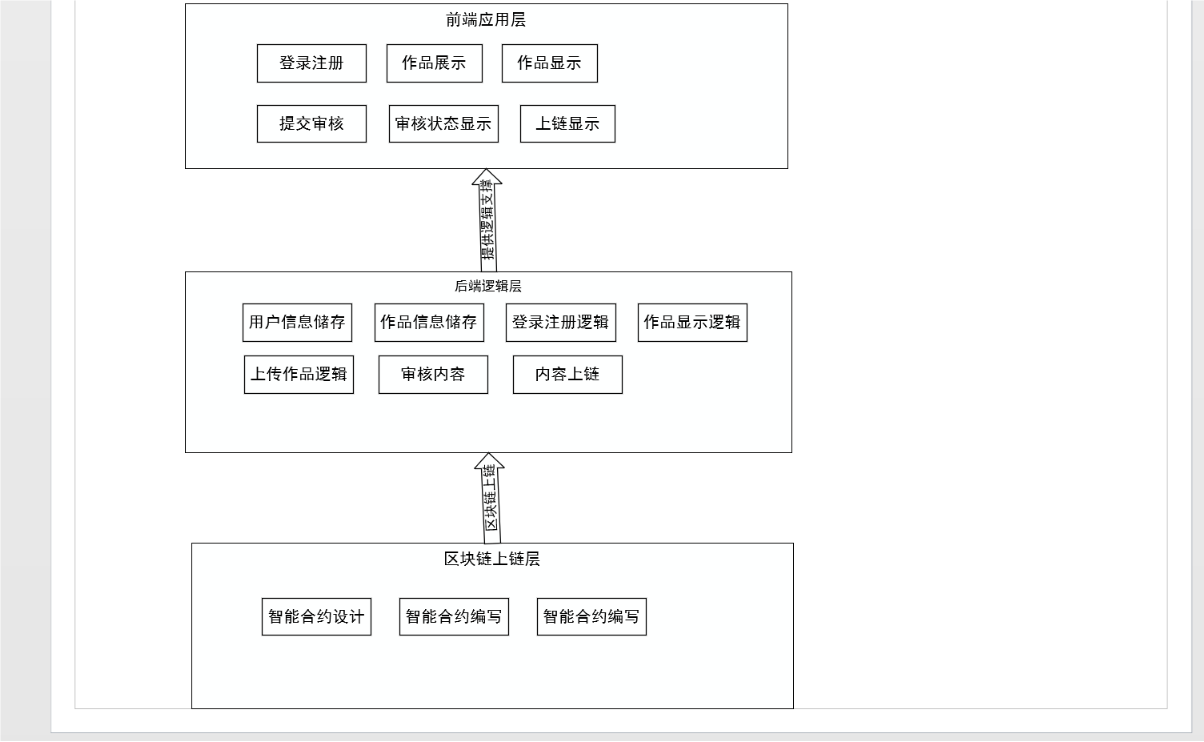
* **版权确权**：通过区块链记录原创作品的所有权及时间戳。
* **存证管理**：存储并验证版权信息，确保不可篡改。
* **授权与交易**：通过智能合约自动化版权授权和交易流程。
* **溯源能力**：提供版权信息的全链条溯源功能。

**主要性能**

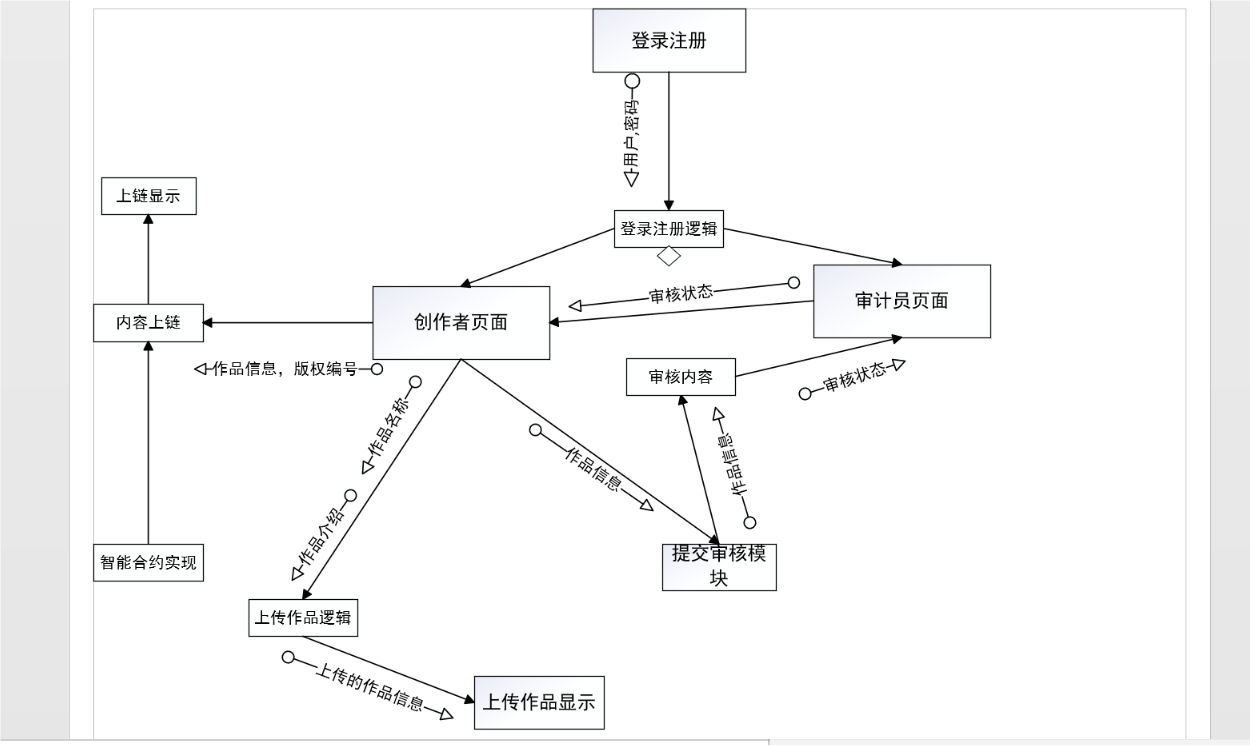
* **安全性**：基于区块链的分布式架构，确保数据安全性。
* **可扩展性**：支持多种数字内容格式，适用于不同规模的用户需求。
* **效率**：采用优化的智能合约和共识机制，提高交易和确认速度。

# 第二章概要设计

模块层次结构



调用关系



模块接口

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | showWorks | | | | | | |
| 说明 | 显示当前创作者的作品信息 | | | | | | |
| 调用路径 | http://localhost:8080/works/creator/show | | | | | | |
| 调用参数 | @RequestParam("id") Long creatorId | | | | | | |
| 返回参数 | worksService.showAll(creatorId) | | | | | | |
| 输入参数 | | | | | | | |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| id | | | Long | | | 是 | 用户的id |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前用户的作品信息 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | getAllWorks | | | | | | |
| 说明 | 显示所有创作者的作品信息 | | | | | | |
| 调用路径 | <http://localhost:8080/works/creator/all> | | | | | | |
| 调用参数 | 无 | | | | | | |
| 返回参数 | worksService.getAllWorksWithCreatorName() | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| 无 | | | 无 | | | 否 | 无 |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | upload | | | | | | |
| 说明 | 内容创作者上传作品接口 | | | | | | |
| 调用路径 | <http://localhost:8080/works/creator/upload> | | | | | | |
| 调用参数 | @RequestBody UploadForm uploadForm | | | | | | |
| 返回参数 | worksService.uploadWork(uploadForm) | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| uploadForm | | | UploadForm | | | 是 | 无 |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |
| 名称 | submitCopyrightApplication | | | | | | |
| 说明 | 内容创作者提交申请版权接口 | | | | | | |
| 调用路径 | http://localhost:8080/works/creator/applicationTrue | | | | | | |
| 调用参数 | @RequestParam("workId") Long workId | | | | | | |
| 返回参数 | worksService.submitCopyrightApplication(workId); | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| workId | | | Long | | | 是 | 作品的id |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | | 自定义 |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |

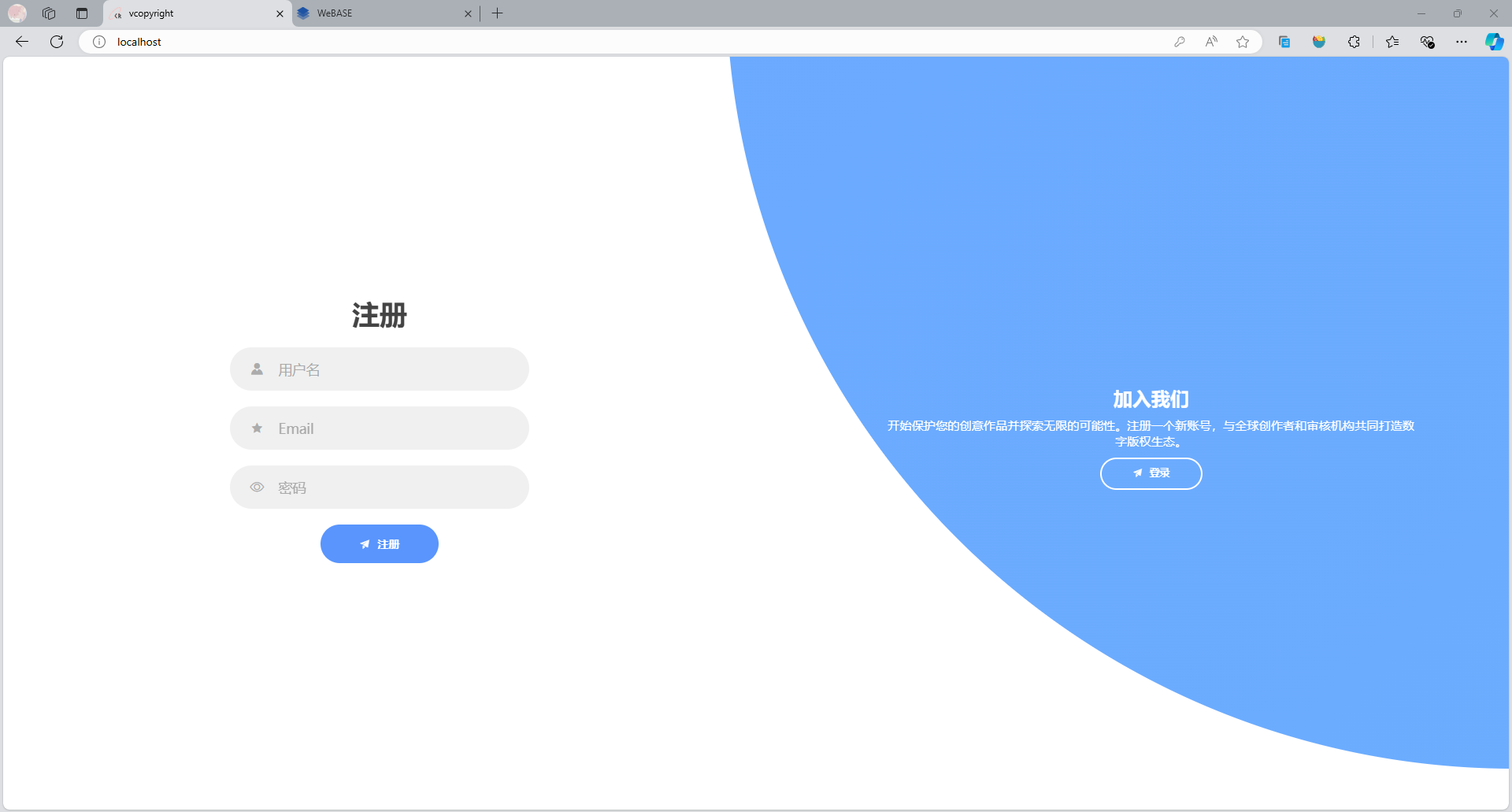
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | reviewCopyrightApplication | | | | | | |
| 说明 | 审核机构审核版权接口 | | | | | | |
| 调用路径 | http://localhost:8080/works/creator/reviewCopyright | | | | | | |
| 调用参数 | @RequestParam("workId") Long workId, @RequestParam("approval") boolean approval | | | | | | |
| 返回参数 | worksService.reviewCopyrightApplication(workId, approval) | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| workId | | | Long | | | 是 | 作品的id |
| approval | | | boolean | | | 是 | 是否同意 |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | login | | | | | | |
| 说明 | 用户登录 | | | | | | |
| 调用路径 | <http://localhost:8080/users>/login | | | | | | |
| 调用参数 | @RequestBody LoginForm loginForm | | | | | | |
| 返回参数 | usersService.login(loginForm) | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| loginForm | | | LoginForm | | | 是 | 邮箱地址，密码 |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | register | | | | | | |
| 说明 | 用户注册 | | | | | | |
| 调用路径 | <http://localhost:8080/users>/register | | | | | | |
| 调用参数 | @RequestBody RegisterFrom registerFrom | | | | | | |
| 返回参数 | usersService.register(registerFrom) | | | | | | |
| 输入参数 | | | | | | |  |
| 参数名称 | | | 参数类型 | | | 是否必填 | 说明 |
| registerFrom | | | RegisterFrom | | | 是 | 用户名，密码，邮箱地址 |
| 输出参数（返回参数） | | | | | | | 说明 |
| 返回状态码 | | code | | string | 是 | | 状态码 |
| 返回消息 | | msg | | string | 是 | |  |
| 数据内容 | | data | | object | 是 | | 当前的作品信息 |

# 人机界面

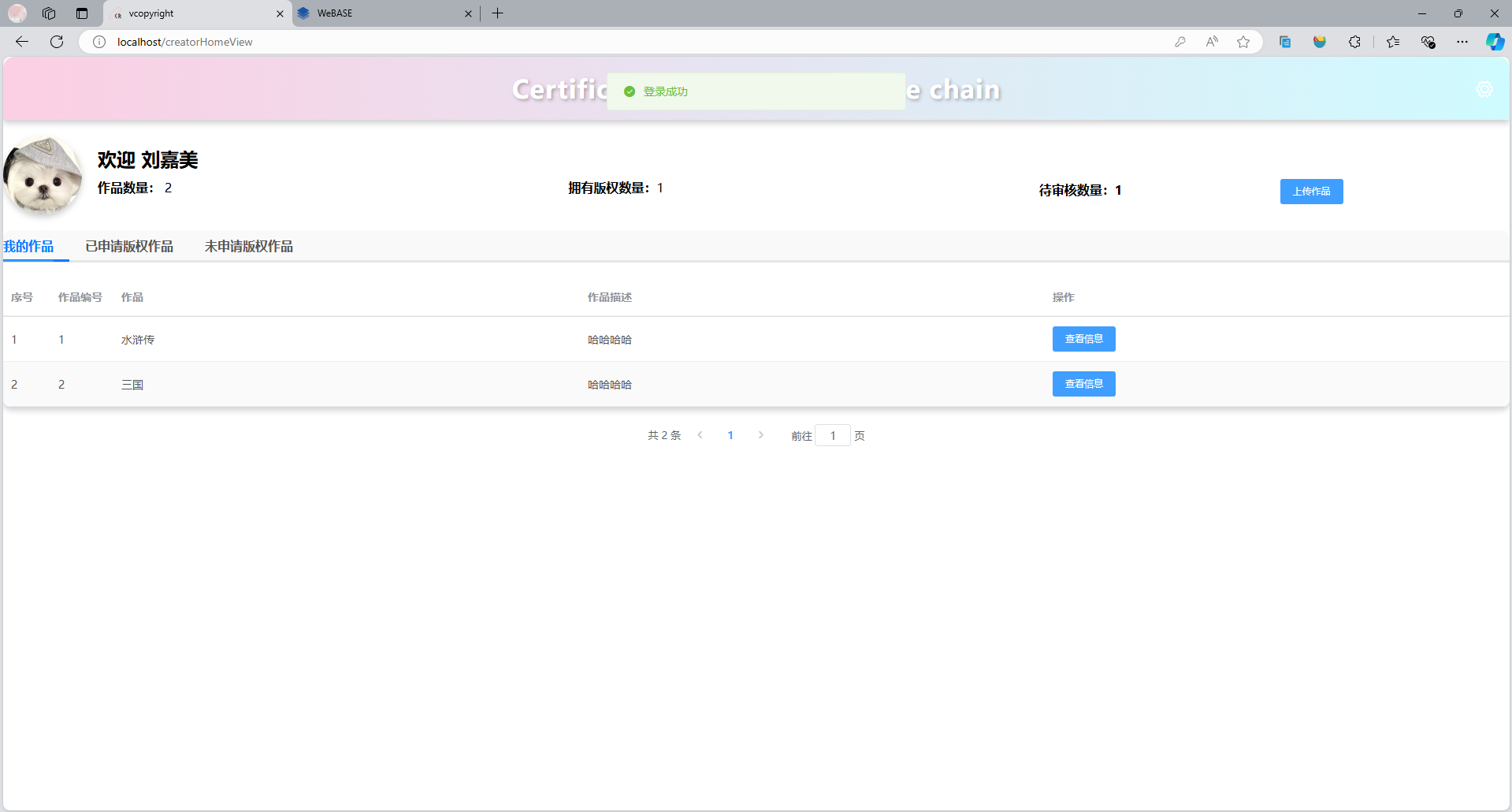
注册界面



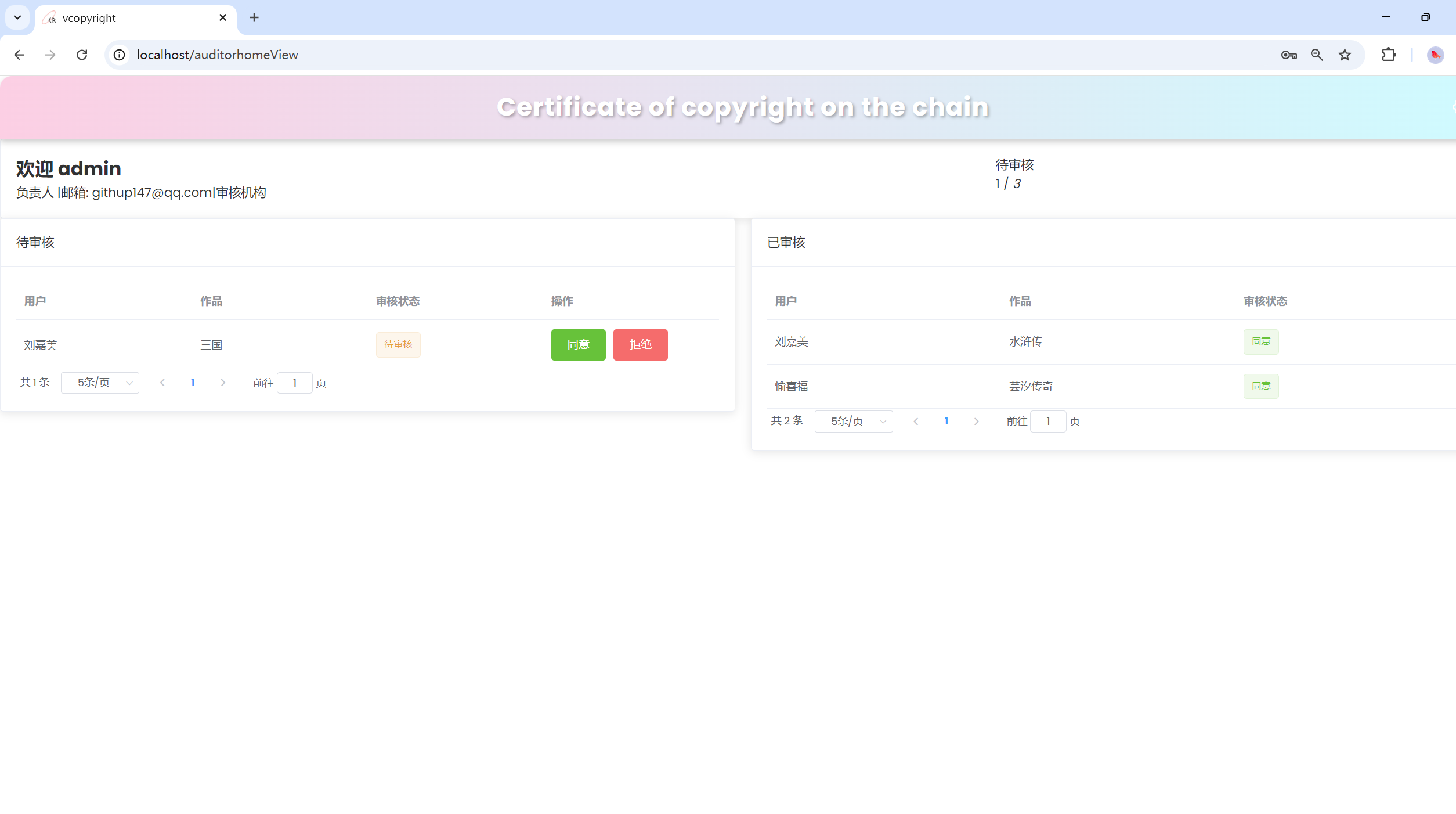
登录界面



用户界面



审核员界面



# 第三章 详细设计

链基础环境

1. 系统环境和依赖库

* **Ubuntu 22.04.4**：这是一个基于Debian的Linux发行版，提供了稳定且安全的操作系统环境。它适合运行各种应用和服务，包括区块链相关应用。
* **openssl, curl, wget, git, nginx, dos2unix**：这些工具对于日常的系统管理、文件传输、版本控制和Web服务部署至关重要。例如，openssl用于安全的加密和认证，curl和wget用于文件下载，git用于版本控制，nginx作为高性能的Web服务器，dos2unix用于处理文件编码差异。

2. Java SDK

* **Java（Oracle JDK 8 至14）**：Java是构建FISCO-BCOS联盟链应用的主要语言。Java 8至14提供了不同的功能集和性能优化，具体选择版本取决于应用的需求和兼容性。

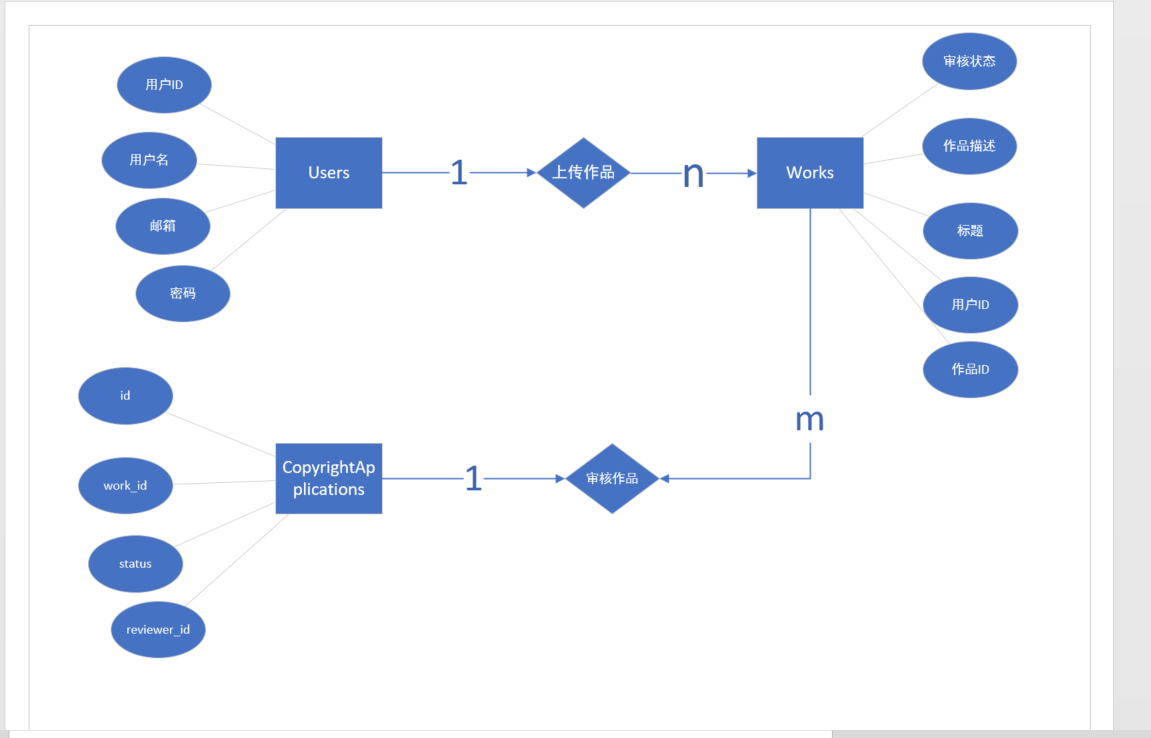
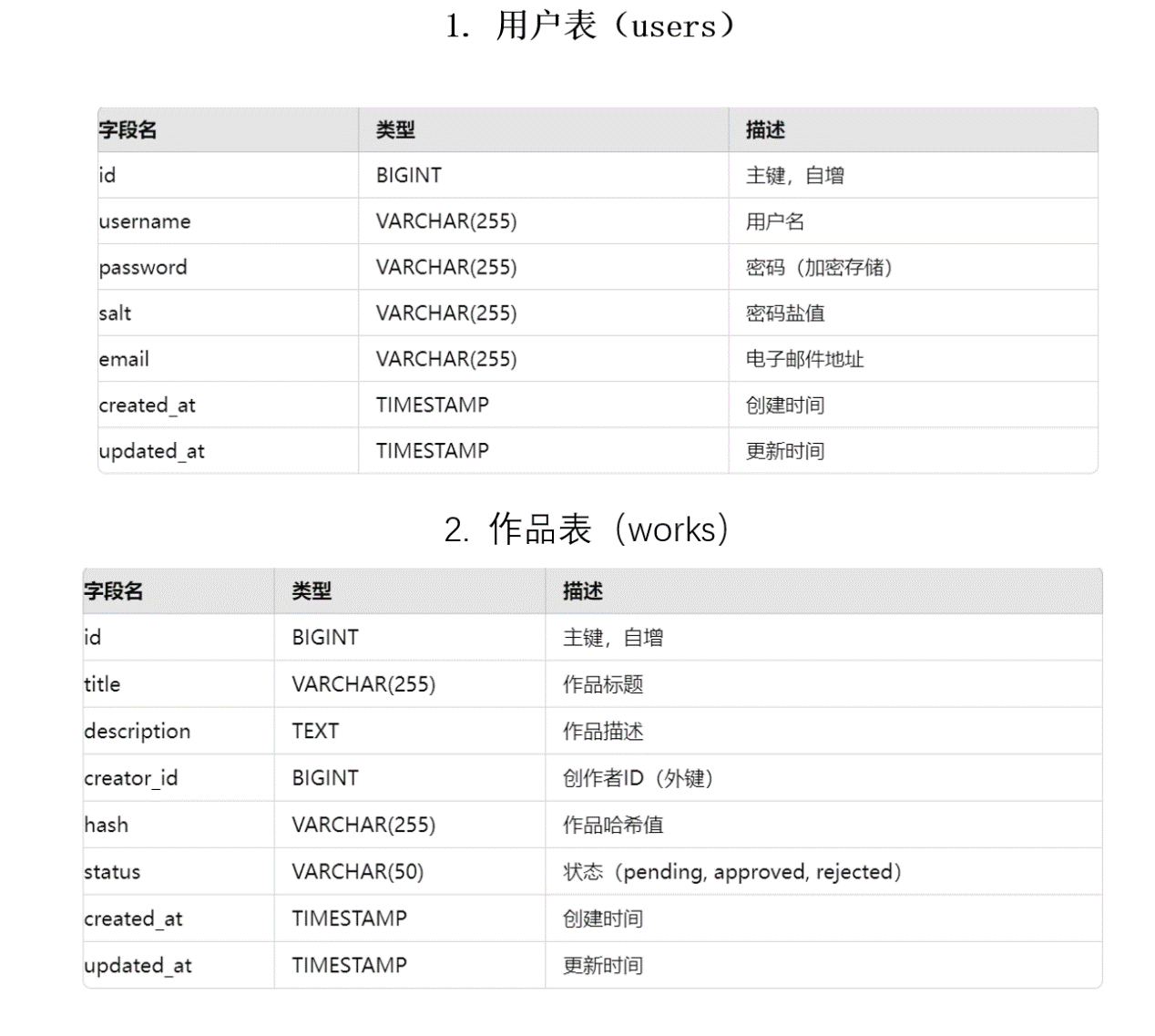
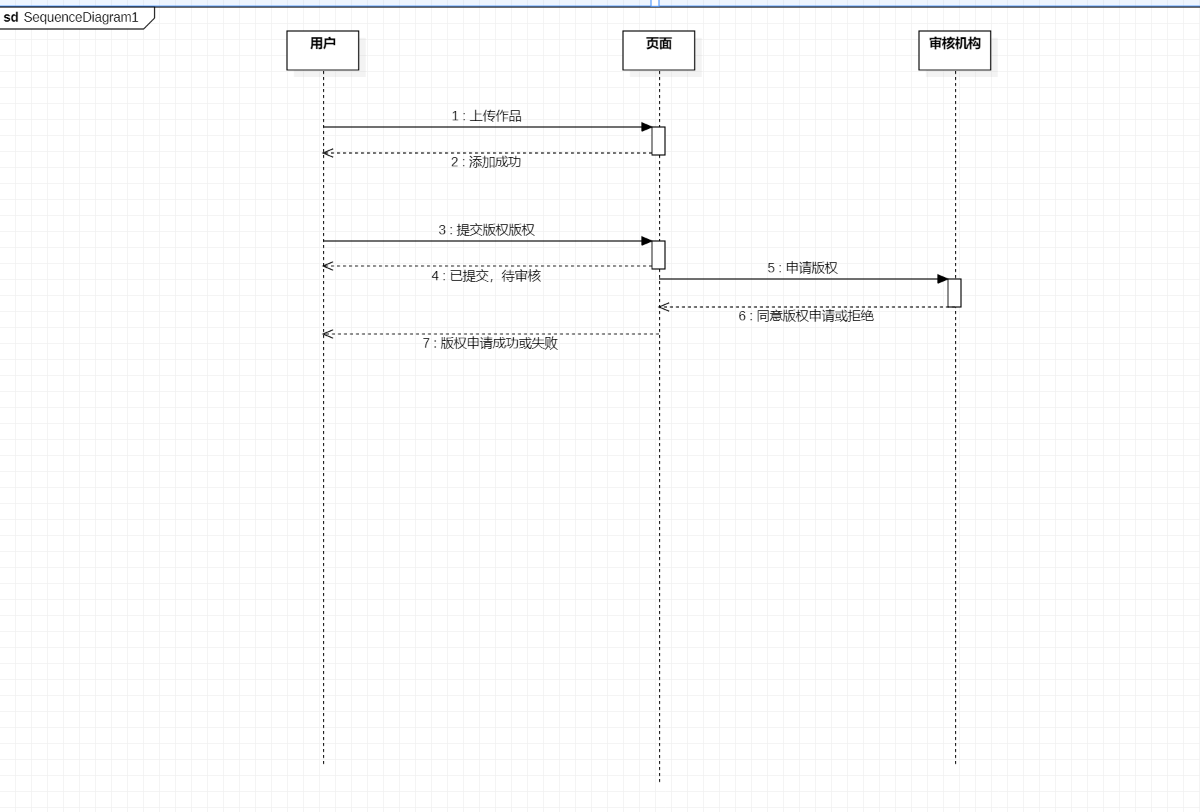
3. FISCO-BCOS联盟链

* **单群组4节点**：表示系统由4个节点组成，这些节点共同维护区块链网络的完整性和一致性。每个节点都运行FISCO-BCOS的Java SDK，负责数据的存储、验证和传播。
* **控制台创建**：这通常涉及到使用FISCO-BCOS提供的工具或API来初始化和管理链的配置，包括添加节点、设置权限、部署智能合约等。

4. WEBASE管理平台

* **MySQL（MySQL-5.6及以上）**：作为数据库服务，MySQL用于存储链的元数据、交易历史、智能合约状态等信息。选择MySQL-5.6及以上版本，确保了兼容性和性能。
* **Python（Python3.6及以上）**：Python作为脚本语言，用于开发管理平台的后端逻辑，如数据处理、API接口、自动化任务等。Python3.6及以上版本提供了丰富的库支持，如PyMySQL用于MySQL数据库的Python接口。
* **PyMySQL**：这是一个Python的MySQL数据库驱动，允许Python脚本与MySQL数据库进行交互，执行SQL查询、插入数据等操作。

数据库设计



# 第四章 测试报告

# 测试工作的经过

|  |  |  |
| --- | --- | --- |
| 时间 | 工作类容 | 参与人员 |
| 2024-9-1 | 登录注册的功能测试 | 唐洪银 |
| 2024-9-2 | 创作者页面上传功能测试 | 唐洪银 |
| 2024-9-2 | 创作者页面显示功能测试 | 唐洪银 |
| 2024-9-3 | 审计员页面功能测试 | 唐洪银 |

# 测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 时间 | 功能 | 用例 | 结果 | BUG |
| 2024-9-1 | 注册功能 | 11 | 通过：7，失败：4 | 3 |
| 2024-9-1 | 登录功能 | 6 | 通过：2，失败：4 | 1 |
| 2024-9-2 | 创作者上传功能 | 8 | 通过：2，失败：6 | 2 |
| 2024-9-2 | 创作者显示功能 | 2 | 通过：2，失败：0 | 1 |
| 2024-9-3 | 审计员功能 | 2 | 通过：2，失败：0 | 0 |

# 质量评估

登录注册基本功能完整，有的缺陷，影响已有的功能使用

注册功能首先是没有确认密码选项，其次密码在注册时不可查看，会造成注册时的密码与登录时的密码不一致。

创作者页面中有一个BUG一般缺陷，在后续的使用中可能会出现影响  
创作者页面上传作品后，没有及时时刷新页面。

因为时间原因，没有进行兼容性测试，只在Microsoft Edge浏览器上进行了测试，其他浏览器可能会有质量问题

# 改进建议 1.在创作者上传作品后可以及时刷新页面，保证用户可以及时查看是否上传成功。 2.测试用例编写过程中有BUG遗漏的情况，可以在测试过程中适当的进行交叉测试。

# 第五章 安装与使用