

**团队项目实训报告**

|  |  |
| --- | --- |
| **实训名称：** | **微众银行区块链项目实训** |
| **组号：** | **第四组** |
| **项目名：** | **NTF交易市场系统** |
| **组长姓名：** | **刘斯韩** |
| **学生专业：** | **软件工程** |
| **开课学期：** | **2020-2021-02** |

**华南理工大学软件学院**

**2021年07月12日**

**团队项目实训报告各个子报告目录**

1. [小组基本情况介绍 3](#_Toc44497440)
2. [NTF交易市场系统项目需求规格说明书 5](#_Toc44497441)
3. [ntf交易市场系统概要设计说明书 21](#_Toc44497442)
4. [ntf交易市场系统详细设计说明书 26](#_Toc44497443)
5. [ntf交易市场系统测试报告 40](#_Toc44497444)
6. [ntf交易市场系统项目总结 46](#_Toc44497445)

# **小组基本情况介绍**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 组号： | | **第四组** | | | | | |
| 项目名： | | **NTF交易市场系统** | | | | | |
| 项目地址： | |  | | | | | |
| 组长  姓名： | | 刘斯韩 | | 学号： | | 201830670344 | |
| 班级： | | 一班 | | 专业： | | 软件工程（卓越） | |
| 主要职责（分小点写清楚项目中承担的角色和承担的主要工作任务）： | | | | | | | |
| 1. 负责Sodility合约开发 2. 负责后端java平台开发 3. 负责详细设计报告 | | | | | | | |
| 组员1  姓名： | | 洪煜鑫 | | 学号： | | 201830670115 | |
| 班级： | | 一班 | | 专业： | | 软件工程（卓越） | |
| 主要职责（分小点写清楚项目中承担的角色和承担的主要工作任务）： | | | | | | | |
| 1. 负责前端主要页面的设计与开发 2. 负责小组基本情况和总结 | | | | | | | |
| 组员2  姓名： | | 贾涵溪 | | 学号： | | 201830670160 | |
| 班级： | | 一班 | | 专业： | | 软件工程（卓越） | |
| 主要职责（分小点写清楚项目中承担的角色和承担的主要工作任务）： | | | | | | | |
| 1. 用户登录和注册功能 2. 生成新账户地址和加载合约实例 | | | | | | | |
| 组员3  姓名： | | 苏泓嘉 | | 学号： | | 201830670474 | |
| 班级： | | 一班 | | 专业： | | 软件工程（卓越） | |
| 主要职责（分小点写清楚项目中承担的角色和承担的主要工作任务）： | | | | | | | |
| 1.进行系统概要设计，并撰写系统概要设计文档； 2.前端用户登录和注册账号页面部分的开发； | | | | | | | |
| 组员3  姓名： | | 潘宣达 | | 学号： | | 201830670436 | |
| 班级： | | 一班 | | 专业： | | 软件工程（卓越） | |
| 主要职责（分小点写清楚项目中承担的角色和承担的主要工作任务）： | | | | | | | |
| 1. 数据库与需求规格设计 2. 数据库分别与前后端相连接 | | | | | | | |

# **NFT交易市场系统项目需求规格说明书**

**目录**

[**1. 概述 7**](#_Toc44497804)

[**1.1. 编写目的 7**](#_Toc44497805)

[**1.2. 项目背景 7**](#_Toc44497806)

[**2. 系统功能需求 7**](#_Toc44497807)

[**2.1. 系统功能架构： 7**](#_Toc44497808)

[**2.2. 登录 7**](#_Toc44497809)

[**2.3. 用户注册 9**](#_Toc44497810)

[**2.4. 账户充值 10**](#_Toc44497811)

[**2.5. 浏览市场 11**](#_Toc44497812)

[**2.6. 购买交易 12**](#_Toc44497813)

[**2.7. 发布交易 14**](#_Toc44497814)

[**2.8. 管理个人信息 15**](#_Toc44497815)

[**2.9. 查看历史交易记录 17**](#_Toc44497816)

[**2.10. 发起退款 18**](#_Toc44497817)

[**2.11. 管理员查看所有交易 19**](#_Toc44497818)

[**3. 系统非功能需求 20**](#_Toc44497819)

[**4. 系统接口：外部系统接口 21**](#_Toc44497820)

1. **概述**
   1. **编写目的**

本设计说明书简单阐明了NFT交易市场系统的基本设计思想、基本功能、模块划分以及模块间接口，便于各模块开发人员能更好地了解该系统的基本情况及各模块详细功能。

* 1. **项目背景**

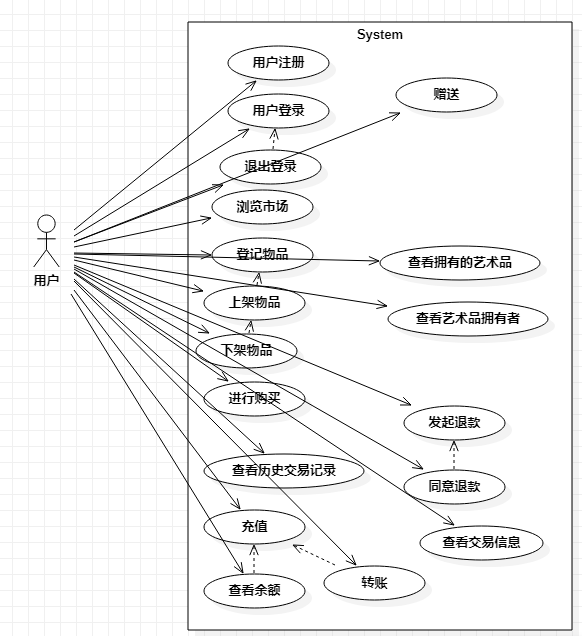
区块链技术的出现有机会深刻改变人类生活，NFT 概念的火爆让区块链常年高喊的落地应用口号变成现实。当下社会，物理存在、虚拟身份和数字实体之间的关系变得更加分散和普及。截止到 2021 年初，NFT 及其配套设施 Gas 使用量已占以太坊全网络的 14％。

非同质化代币又称NFT是专有的数字资产。因此，没有不可替代的令牌是相同的。当我们向用户发送1个BTC并收到1个BTC时，对于我们来说没有任何变化。那是因为每个硬币都是相同的，这使其成为可替代的数字资产。NFT就像达芬奇（Leonardo da Vinci）的《蒙娜丽莎》（Mona Lisa）。它是独特的，这就是使其变得有价值的原因。全世界没有其他和它一样的画作。

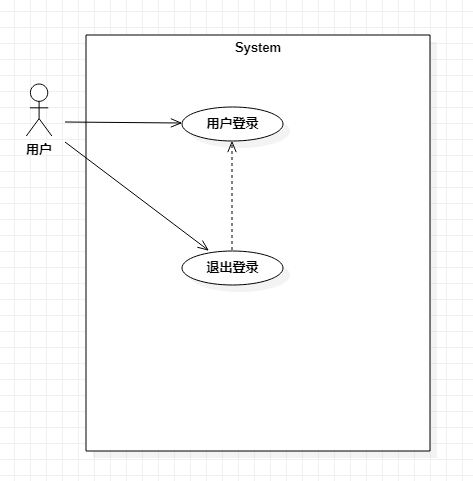
在此背景下，NFT交易市场的需求正在提高。二手商品交易市场，是基于一种全新消费文化和消费理念，伴随着互联网的高速发展，快速、便利地为消费者提供高性价比的二手物品的平台，既能满足人们的生活需要，又可以减少资源浪费。这之中，又以数字艺术品最为有实用意义

本NFT交易市场系统可以提供一个供用户自由发布和购置二手商品的平台。用户在系统中可以进行注册用户，发布商品，完成交易，查看交易等操作。本系统后台是基于FISCO-BCOS的区块链系统，在保证稳定性和高效性的同时能够提供足够的安全性，并可以保护用户的隐私和交易的安全性。

1. **系统功能需求**
   1. **系统功能架构：**



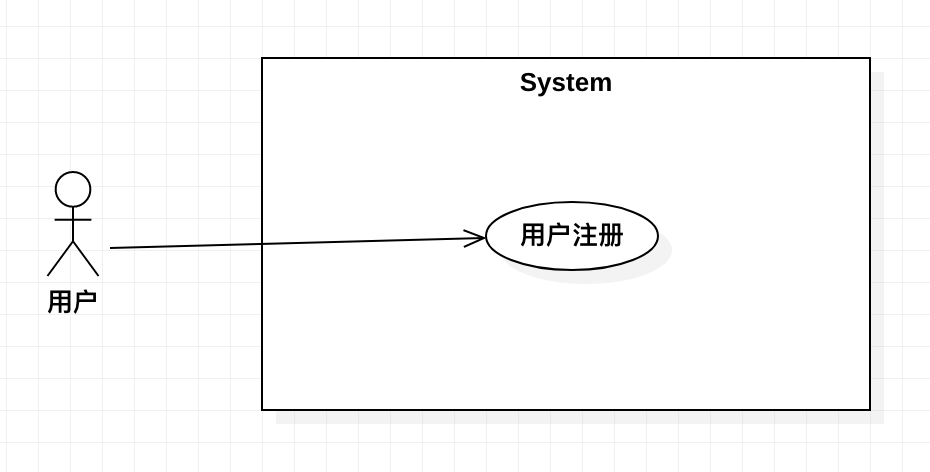
* 1. **登录**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户可以登陆自己的NFT交易系统账户。 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户未登录NFT交易系统 |
| 用例后置条件 | 成功登录系统 |
| 基本事件流 | 用户打开NFT交易系统  系统显示欢迎界面，并要求输入账户名和密码  用户输入账户名和密码  系统验证登录信息，允许用户登录  系统根据不同用户类型启动不同的主界面 |
| 备选事件流1 | 用户打开二手物品交易系统  系统显示欢迎界面，并要求输入账户名和密码  用户输入账户名和密码  系统显示用户名错误或密码错误的提示信息  用户可以重新输入登录信息，也可以选择结束该用例 |
| 备选事件流2 | 用户登录系统，进入首页面  用户点击退出登录按钮，退出账户登录  返回初始界面 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求： |
| 用例相关业务数据 | 用户名、密码、账号身份 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

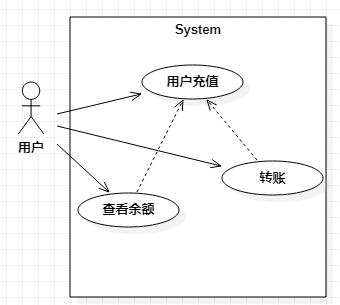
* 1. **用户注册**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户可以注册NFT交易系统账户。 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户未使用过NFT交易系统 |
| 用例后置条件 | 成功注册账户 |
| 基本事件流 | 用户打开NFT交易系统  系统显示欢迎界面，用户点击注册按钮进入注册界面  用户输入注册信息  系统验证登录信息，用户自动登录 |
| 备选事件流1 | 用户打开NFT交易系统  系统显示欢迎界面，用户点击注册按钮进入注册界面  用户输入注册信息非法  系统显示注册信息有误的提示信息  用户可以重新输入注册信息，也可以选择结束该用例 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库加密 |
| 用例相关业务数据 | 用户名、密码、账号身份 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

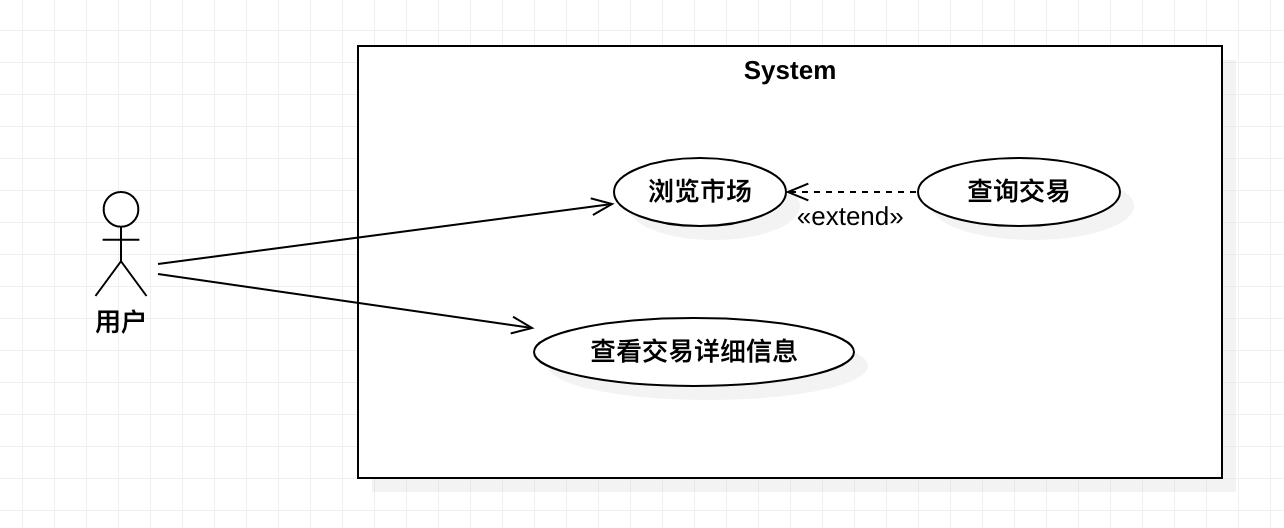
* 1. **账户充值**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户向账户中充值，用于交易 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户登录NFT交易系统 |
| 用例后置条件 | 账户中充值成功 |
| 基本事件流 | 用户进入市场  点击充值按钮，输入充值数额  系统显示充值成功 |
| 备选事件流1 | 用户进入市场  点击充值按钮，输入充值数额  输入数额错误，系统显示充值失败  用户可以选择重新输入数额，也可以选择结束用例 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 账号地址，账户余额 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

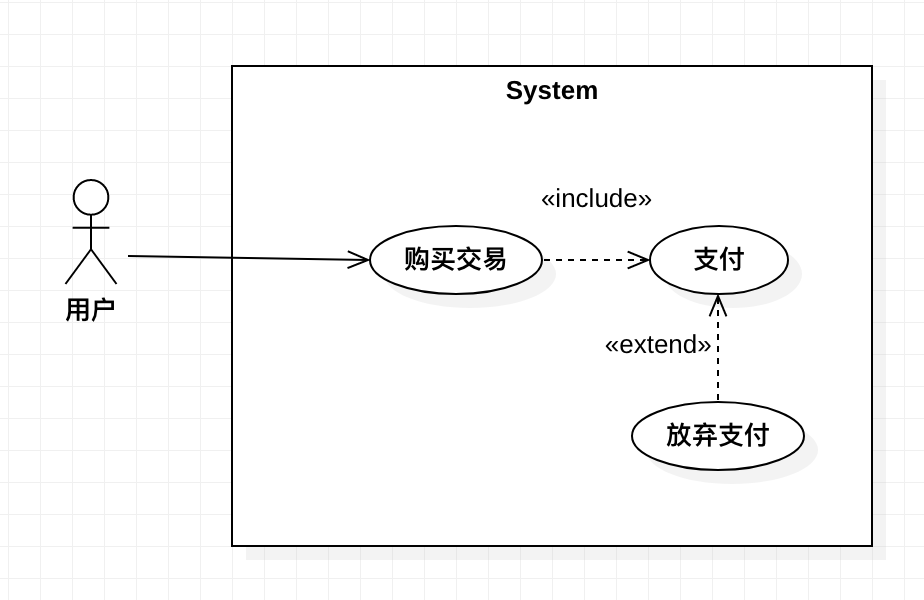
* 1. **浏览市场**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户可以浏览NFT市场，选择感兴趣的交易 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户登录NFT交易系统 |
| 用例后置条件 | 用户浏览市场，并查看到交易的详细信息 |
| 基本事件流 | 用户进入市场  点击市场中的交易进入详情页面 |
| 备选事件流1 | 用户进入市场  输入关键字搜索相关已发布的交易  点击交易进入详情页面 |
| 备选事件流2 | 用户进入市场  输入关键字搜索相关已发布的交易  没有搜索到相关的交易  用户可以重新输入关键字搜索，也可以选择结束该用例 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 交易号，交易详细信息 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

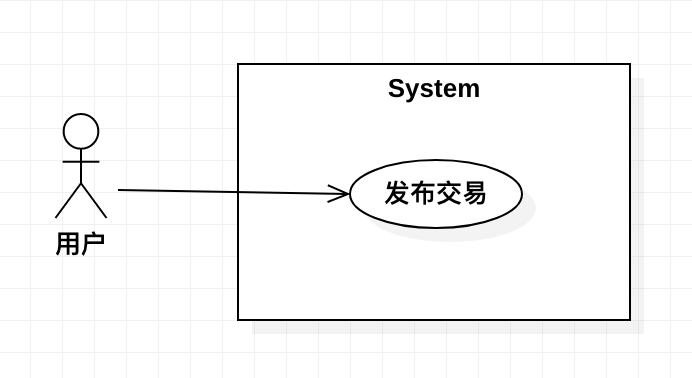
* 1. **购买物品**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户进入交易详情页后可以选择购买交易 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户已经进入交易详情页 |
| 用例后置条件 | 成功购买交易 |
| 基本事件流 | 用户点击购买按钮，进入交易支付界面  点击支付按钮，输入密码  系统显示成功购买，钱款转账成功 |
| 备选事件流1 | 用户点击购买按钮，进入交易支付界面  点击支付按钮，输入密码  账户余额不足，放弃支付 |
| 备选事件流2 | 用户点击购买按钮，进入交易支付界面  点击取消按钮，放弃支付  用户可选择重新购买，也可以选择结束该用例 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 交易号，用户地址，用户余额，用户密码 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

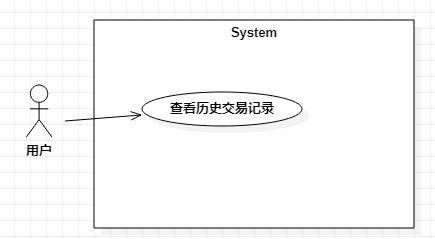
* 1. **发布交易**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户可以发布交易，出售商品。 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户登录NFT交易系统 |
| 用例后置条件 | 用户成功发布交易，市场中增加一个交易 |
| 基本事件流 | 用户点击发布交易按钮  填写出售物品的描述、图片、定价等信息  系统显示交易发布成功 |
| 备选事件流1 | 用户点击发布交易按钮  填写出售物品的描述、图片、定价等信息  用户点击取消发布按钮，结束用例 |
| 备选事件流2 | 用户点击发布交易按钮  填写出售物品的描述、图片、定价等信息  交易信息填写不符合要求  用户可以重新填写，也可以选择结束该用例 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 交易号，交易描述，交易价格，交易状态，用户id |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

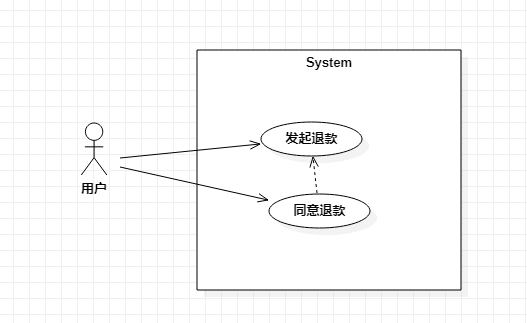
* 1. **查看历史交易记录**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户查看与自己有关的交易（包括已完成的和正在出售的）。 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户NFT交易系统 |
| 用例后置条件 | 用户交易被修改或删除，或用户查看自己的历史交易。 |
| 基本事件流 | 用户点击历史交易按钮  用户选择一个还未交易成功的交易  点击修改按钮，修改商品信息  系统显示交易信息已更新，在市场中可以看到最新的信息。 |
| 备选事件流2 | 用户点击历史交易按钮  用户选择一个交易  用户查看交易信息  用户点击返回按钮，退出用例。 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 用户地址，交易号，交易描述信息，交易价格，交易状态。 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

* 1. **发起退款**
     1. **用例图**



* + 1. **用例描述**

|  |  |
| --- | --- |
| 用例简要描述 | 用户申请退款，经过对方同意后完成退款。 |
| 用例角色 | 用户 |
| 用例前置条件 | 用户有已完成的交易 |
| 用例后置条件 | 交易退款成功，钱款恢复到原来账户 |
| 基本事件流 | 用户点击退款按钮  选择需要退款的交易  提交卖家同意  对方通过，交易款项退回到买家账户 |
| 备选事件流1 | 用户点击退款按钮  选择需要退款的交易  提交卖家同意  卖家不通过，交易不做修改，用例结束 |
| 用例非功能性需求 | 性能需求：  响应时间：小于0.1秒  可用性需求 ：  正常运行时间:99.9%。  并发需求：  能支持至少百个并发请求。  安全性需求：  数据库进行加密 |
| 用例相关业务数据 | 买家地址，卖家地址，账户余额，交易号 |
| 用例相关人（参与讨论的人员） | 刘斯韩、洪煜鑫、潘宣达、苏泓嘉、贾涵溪 |

1. **系统非功能需求**

|  |  |
| --- | --- |
| 性能需求 | 系统能够适应400个用户，平均每个会话持续8分钟；  生成的web页面，通过速率100KBps在10秒内可以全部下载 |
| 可用性需求 | 正常运行时间:99.9%。 |
| 并发需求 | 能支持至少百个并发请求。 |
| 安全性需求 | 所有涉及信息的操作必须登录后才能完成；  系统只允许用户查看自己的信息，而不能查看其他用户的资料；  数据库进行加密。 |

1. **系统接口：外部系统接口**

|  |  |
| --- | --- |
| 用户界面 | 界面风格：本系统采用的是图形用户界面，界面简单且搭配合理。  界面布局：系统页面较为合理，给人一种成熟稳重的感觉。  界面操作：页面上的按钮和文本框合理设计，让用户的使用更加方便快捷。  界面内容：界面设置导航，消息显示为弹出式。 |
| 硬件接口 | 软件之间交流的数据：用户账户上的信息，如已完成交易等。  通信协议：HTTP协议 |
| 软件接口 | 数据存储：采用FISCO BCOS 区块链  操作系统：开发系统为Ubuntu 16.04  工具：IDEA等  数据访问：Web3j SDK |
| 通信接口 | 通过浏览器访问 |
| 故障处理 | 登录失败或连接失败：尝试检查本地网络连接，并重启浏览器。 |

# **NFT交易市场系统概要设计说明书**

目录

[**1. 概述 24**](#_Toc44884412)

[**1.1. 系统简述 24**](#_Toc44884413)

[**1.2. 软件设计目标 24**](#_Toc44884414)

[**1.3. 参考资料 24**](#_Toc44884415)

[**1.4. 修订版本记录 24**](#_Toc44884416)

[**2. 术语表 25**](#_Toc44884417)

[**3. 设计概述 25**](#_Toc44884418)

[**3.1. 系统结构设计 25**](#_Toc44884419)

[**3.1.1. 系统物理架构 25**](#_Toc44884420)

[**3.1.2. 系统逻辑架构 25**](#_Toc44884421)

[**3.1.3. 系统数据模型 26**](#_Toc44884422)

[**3.2. 系统接口设计 27**](#_Toc44884423)

[**3.3. 系统数据字典 27**](#_Toc44884424)

[**3.4. 系统智能合约设计 28**](#_Toc44884425)

[**3.4.1. 主要solidity函数及功能 28**](#_Toc44884426)

1. **概述**
   1. **系统简述**

随着文化产业的兴起，市面上摄影、绘画以及音乐等艺术作品的出现规模越发庞大。而计算机技术和互联网的发展则让艺术作品更多地趋向电子化。无论是使用电脑软件创作出来的插画、音乐，还是发布在线上的摄影作品、视频创作，都是电子艺术作品最典型的例子。

本质上电子艺术作品和实体艺术作品一样都是属于创作者独有的资产，他人若是有使用需求，则必须向其创作者进行购买获得其所有权或者进行租借获得其使用权。而目前由于市场上对该领域的监管尚未完善，电子艺术作品的作者难以维护自己对作品的知识产权，他人轻易就可以对电子艺术作品进行盗用，对创作者带来权益上的损害。

在此背景下，一个能有效对电子艺术作品的知识产权进行权威认证的平台则显得必不可少。作者拥有了对自己作品所有权的权威认证后，就可以更好地进行维权防止他人的肆意盗用。这种平台对填补电子艺术作品的知识产权保护上的漏洞，有着非常大的积极作用。

这也是本平台开发的初衷，通过FISCO BCOS区块链技术，平台可以有效地对知识产权的所有权进行权威认证。由于区块链可溯源、防篡改的特性，电子艺术作品作为一种可交易的资产，在链上也可以查到其所有权的所有交易转让记录。

* 1. **软件设计目标**

a) 提供一个基本的交易平台，可以进行资产的交易以及个人账户的建立和维护操作：创建账户、查询余额、查看名下资产等；

b) 较高的可靠性。该系统应提供可靠的交易服务，并且为交易日志信息和资产信息提供保护机制，防止对资产所有权以及交易记录的恶意篡改；

c) 良好的可维护性。若系统出现故障应该能够快速地被修复。此外该系统也能较容易地添加新的功能来帮助市场的运营。

* 1. **参考资料**

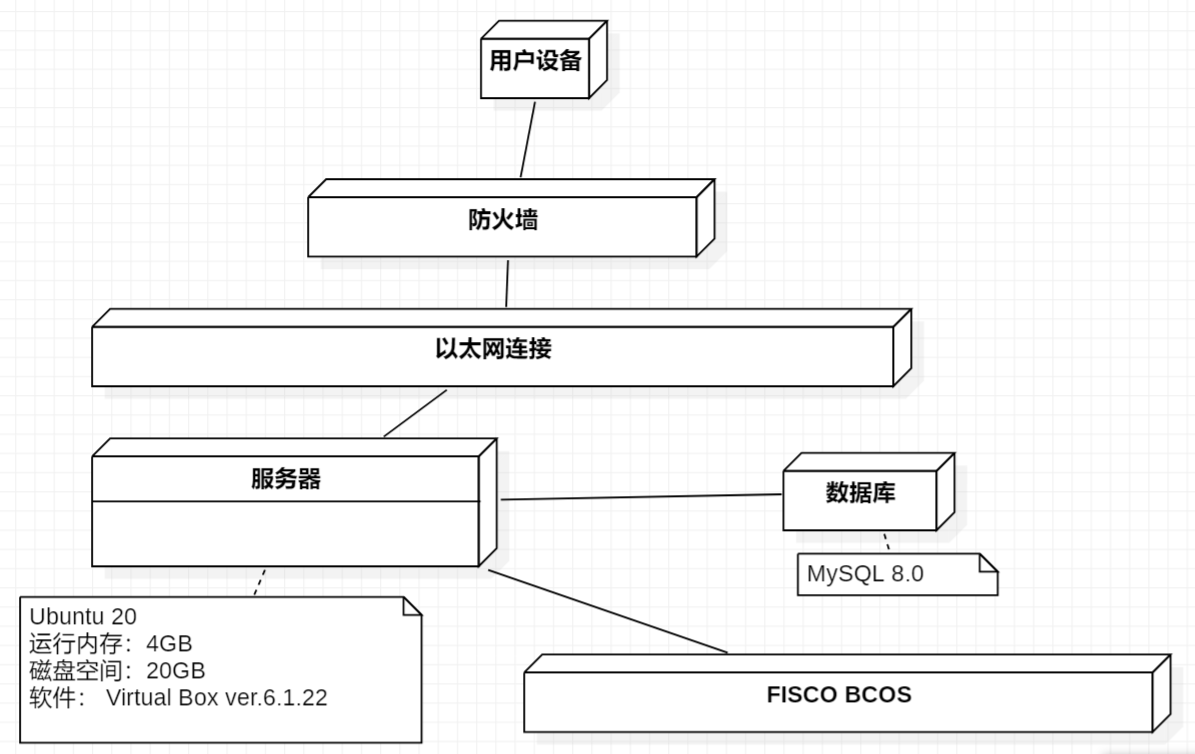
1. 《IT项目管理》
2. 《数据库系统概念》
3. 《软件工程》
4. 百度百科
   1. **修订版本记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2020.7.8 | v1.0 | 完成概要设计说明书的编写 | 苏泓嘉 |

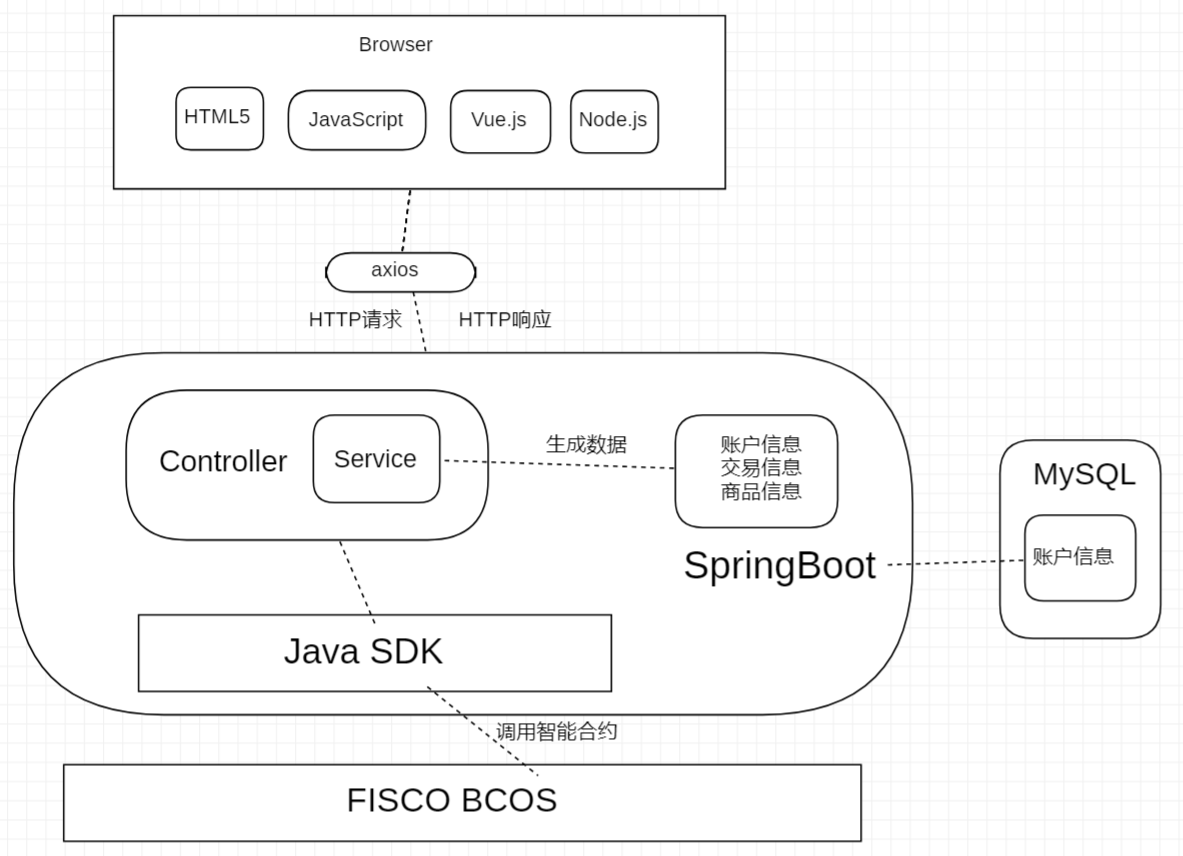
1. **术语表**

|  |  |  |
| --- | --- | --- |
| 术语 | 解释 | 备注 |
| 项目 | 软件工程中的一个项目，由一个项目经理和若干个普通成员组成。 | 项目是本项目的基本单位 |
| 任务 | 项目中包含的若干任务，任务由项目经理布置和指派人员。任务分为4层，最后一层无法再包含子任务。 |  |
| 功能 | 软件在正确部署后，可向用户提供的服务，也是软件开发工作的主要内容 |  |

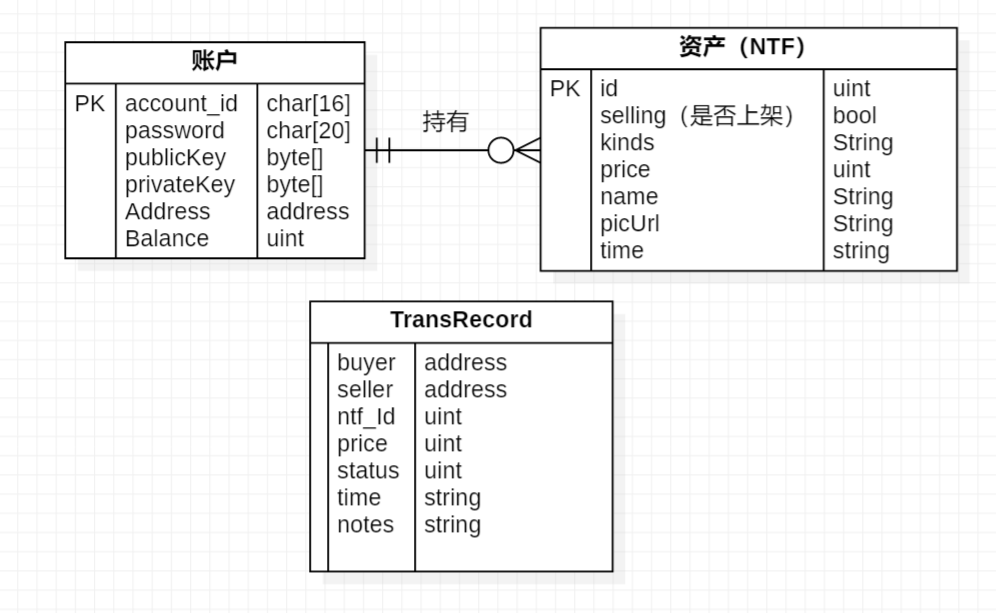
1. **设计概述**
   1. **系统结构设计**
      1. **系统物理架构**

****

* + 1. **系统逻辑架构**



* + 1. **系统数据模型**
       1. **E-R图**



* + - 1. **数据库表**

表1 账户信息表 (User）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 能否为空 | 描述 | 备注 |
| account\_id | VARCHAR(16) | 否 | 用户名 |  |
| password | VARCHAR(20) | 否 | 用户密码 |  |
| publicKey | byte[] | 是 | 链账户公钥 |  |
| publicKey | byte[] | 是 | 链账户私钥 |  |
| Balance | INT | 否 | 账户余额 |  |
| address | VARCHAR(32) | 是 | 链账户地址 |  |

* 1. **系统接口设计**

|  |  |  |
| --- | --- | --- |
| 所属 | 接口 | 功能 |
| 注册模块 | 注册并登录 | 前端发送用户信息，后端注册一个链账户并绑定用户的账号和密码，并用该账户登录平台 |
| 登录模块 | 登陆 | 前端发送用户名和密码，后端执行登陆 |
| 交易模块 | 发布交易 | 前端发送买卖方id，交易描述，图片路径等信息，后端执行交易发布 |
| 查看交易 | 前端发送交易id，后端执行查找交易并返回信息 |
| 转账 | 前端发送金额和买卖方账户，后端执行账户之间的转账 |
| 申请退款 | 前端发送请求，后端执行退款流程相关操作 |
| 信息查询模块 | 查询余额 | 前端发送查询请求，后端返回个人账户余额 |
| 新建Ntf | 前端发送Ntf信息，后端根据信息在账户名下新建Ntf资产 |
| 下架ntf | 前端发送id，后端根据信息将该Ntf资产变为不可交易（下架） |
| 查询链账户 | 前端发送查询请求，后端返回个人链账户地址 |
| 查询名下ntf | 前端发送查询请求，后端返回个人名下所有Ntf资产列表 |
| 查询在售ntf | 前端发送查询请求，后端返回个人名下所有可交易的Ntf资产列表 |
| 查询ntf信息 | 前端发送id，后端返回该id对应的Ntf资产信息 |
| 查询个人交易记录 | 前端发送查询请求，后端返回个人参与的所有交易记录列表 |

* 1. **系统数据字典**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据名称 | 数据业务需求描述 | 业务数据类型 | 所属表格 | 对应字段 | 字段类型 |
| 用户账号 | 不超过15字符 | 字符串 | User | username | VARCHAR(16) |
| 密码 | 6-16位字符(加密) | 字符串 | User | password | VARCHAR(20) |
| 账户余额 | 用户当前余额 | INT | User | balance | INTEGER |
| 商品名字 | 不超过30字符 | 字符串 | Order | name | VARCHAR(32) |
| 商品描述 | 不超过200字符 | 字符串 | Order | description | VARCHAR(200) |
| 商品价格 | 商品价格 | INT | Order | price | INTEGER |
| 交易状态 | 分为未售出，已售出，已下架，仲裁中4种状态 | INT | Order | status | INTEGER |

* 1. **系统智能合约设计**
     1. **主要solidity函数及功能**

|  |  |
| --- | --- |
| **函数名称** | **功能** |
| **transfer** | 交易转账 |
| **newAccount** | 新建账户 |
| **myBalance** | 查询余额 |
| **myAddress** | 查询链账户地址 |
| **newNtf** | 新建Ntf资产 |
| **sellNtf** | 上架资产 |
| **offSale** | 下架商品 |
| **getMyNtfs** | 获取名下所有资产 |
| **makePurchase** | 创建一笔交易 |
| **getMyRecords** | 获取个人交易记录 |
| **getRecordById** | 获取特定交易信息 |
| **requestArbitration** | 向区块链发送退款请求 |
| **approveArbitration** | 接受退款请求，在链上回滚交易 |
| **denyArbitration** | 拒绝退款请求 |

# **NFT交易市场系统详细设计说明书**

**目录**

[**1. 需求概述 29**](#_Toc44884518)

[**2. 需求分析 29**](#_Toc44884519)

[**2.1. 宏观分析 29**](#_Toc44884520)

[**2.2. 国内平台分析 30**](#_Toc44884521)

[**2.3. 用户分析 30**](#_Toc44884522)

[**2.4. 竞争态势分析 32**](#_Toc44884523)

[**2.5. 市场分析 32**](#_Toc44884524)

[**3. 现状总结 33**](#_Toc44884525)

[**3.1. 行业可行性 33**](#_Toc44884526)

[**3.2. 未来发展方向 33**](#_Toc44884527)

[**4. 用户角色分析 34**](#_Toc44884528)

[**4.1. 角色 34**](#_Toc44884529)

[**4.2. 参与业务 34**](#_Toc44884530)

[**5. 功能性需求 35**](#_Toc44884531)

[**6. 非功能性需求 36**](#_Toc44884532)

[**7. 系统设计 36**](#_Toc44884533)

[**7.1. 基本架构设计 37**](#_Toc44884535)

[**7.2. 功能实现函数设计 39**](#_Toc44884536)

[**8. 后台设计 42**](#_Toc44884537)

[**8.1. 环境设置 42**](#_Toc44884538)

[**8.2. 后端架构 42**](#_Toc44884539)

[**9. 界面设计 43**](#_Toc44884540)

1. **需求概述**

随著网际网路时代的猛爆性成长，区块链架构资产无疑已经是本世代最被人注目的投资项目之一，其中包括各式各样的虚拟货币，相信所有人都已经略有所闻。近期，一种名为「NFT」的资产在全球艺术界急速窜升，成为了无论是艺术品爱好者、投资家或是艺术家都密切关注的区块链资产。举例来说，Twitter 首席执行官杰克多西（ Jack Dorsey ）就正以 NFT 拍卖他人生的首条贴文、特斯拉CEO 伊隆马斯克（Elon Musk） 的女友也透过 NFT 售出了价值 600 万美元的数位艺术品，以及现年 39 岁的数位艺术家Beeple （Mike Winkelmann） 以 6,900 万美元的价格拍卖了他的纯数位艺术作品的 NFT。

在此背景下，NTF物品交易市场的需求正在提高。NTF交易市场，是基于一种全新消费文化和消费理念，NTF交易是指消费者将非同质化代币NTF，通过网络平台将其作为一种新的商品进行交易流通的行为。非同质化代币，即NFT，则是唯一的、不可拆分的token，如加密猫或token化的数字门票。这些区别就像带有编号的人民币，却不会存在编号相同的人民币。

本NTF交易市场系统可以提供一个用户自由创建，交易NTF的平台，用户可以进行注册用户，发布商品，上下架商品，完成交易，查看订单，是否退货等操作。本交易系统后台采用java语言，基于FISCO-BCOS的区块链系统，可以保证稳定性和高效性的同时能够提供足够的安全性，且对用户的隐私有更好的保障。



图表 1 NTF案例（加密猫）

1. **需求分析**
   1. **宏观分析**
      1. **NFT 是可以真正拥有和控管的数位资产的媒介。**

在当今的世界，虚拟资产正在占据越来越重要的地位，如图片、音乐或是电玩游戏，而现在人们缺乏证明自己虚拟资产所有权的方法，目前的数位传递本质上都只是复制分享，并非真正的价值转移。NTF的出现给这个问题提供了一种解决的方法。艺术品的买主没有通过NTF交易获得实体的艺术品，无论是存放在 USB 或是光碟中等等都没有，而是获得了这一令人垂涎的数位艺术品无可质疑的所有权。

* + 1. **中国消费者购买力正在增强，同时虚拟类物品消费不断上升，为NFT交易行业提供利好机会**

近年来，中国社会虚拟消费品零售总额不断增长，这表明国民的消费购买力正在增强，同时对许多数位内容创作者和相关收藏者来说，他们则是将 NFT 视为一项更加安全且长期的投资。NFT 的本质是数位的，从而消除了某些物理的限制，例如对物理上时间和空间对项目的耗损。随著世界变得越来越数位化，人们在网路上花费的时间越来越多，获得更多的数位资产将会更加普遍。另外，许多人 也将 NFT 视为是更可靠的正品验证证明，这给在线NFT交易行业的发展提供机会。

* 1. **目前NFT平台分析**

NFT交易市场在世界上是长期存在的。成立于 2018 年的 OpenSea 是一个基于区块链的全球数字物品买卖市场。Nifty Gateway 是一个受监管的、非同质化代币（NFT）的货币市场，利用 Gemini 的后端基础设施建立了美元兑换平台。发布于 2016 年的 MakersPlace 是一个基于区块链技术的、为艺术家和创作者提供全方位服务的平台，用于铸造代表其作品的 NFT。Rarible 是 NFT 发行与交易平台，创建和出售由区块链保护的数字藏品。



图表 2多家NFT平台资料

* 1. **用户分析**

人们对于资产的拥有从完全的物质资产转向了部分物质部分数字的资产，且数字资产在某些情况下具有很大的价值，例如，知名音乐人Grimes，即特斯拉CEO埃隆·马斯克的女友，在首次NFT拍卖8件代币化艺术品后，20分钟内就创造了580万美元的收益。这不得不让大家对于这种创造财富的能力有点好奇。因此NFT交易 在未来应拥有很大的前景。

图表 3 知名NFT平台OpenSea

* 1. **竞争态势分析**

1. 供应商：创建并上架NFT商品的用户。
2. 购买者：购买NFT商品的用户。
3. 潜在进入者：虚拟物品商，垂直品类社区及电商平台，
4. 替代品：电商平台，其它NFT交易平台
5. 同行竞争：OpenSea，Nifty Gateway，MakersPlace，Rarible等。
   1. **市场分析**

市场阶段一般分为导入期、成熟和衰落期。近年来NFT交易产品刚刚上线，可以说，还没有引起太大的波动，但已经吸引了部分人的关注。其中的原因除了部分NTF本身存在着很高的价值，还有这种新的交易模式让人们感受到了更多可能性。我认为NFT交易市场正在渐入佳境。

1. **现状总结**
   1. **行业可行性**

根据前面的分析，NFT交易市场的发展仍然处于探索期，且当前环境对其发展十分有利；金融机构支持，经济环境适合，市场不断下沉扩大，互联网技术普及以及新技术成熟应用。因此在NFT交易细分场景下的垂直领域仍然又很大机会进入。通过对NFT交易用户群体的研究分析，用户基础依然非常庞大，这近一步印证NFT交易市场颇具潜力。

* 1. **未来发展方向**

在很多人的印象中，NFT交易的模式还相当新颖，但现阶段的NFT交易平台，已经十分成熟。无论是分享经济的模式拓展，还是电商市场的深耕，抑或是跳出个体买卖，进入企业端NFT交易的广阔空间，或者与公益及环保领域的深入融合，都为NFT交易的未来，带来无限的可能性。

1. **用户角色分析**
   1. **角色**

用户角色从整体来看，为普通用户，无管理员监管。

* + 1. **普通用户**

这里包含了买卖双方。如从调研来看，用户的主要需求在于交易的管理，以及交易售后问题。用户能够在我们的平台上发布交易，能够在市场上浏览上架的物品、进行购买，同时在必要时能够对交易进行申诉。

* 1. **参与业务**

普通用户：创建物品，查看物品，上架下架物品、浏览物品、购买物品、充值、管理个人信息、查看个人历史记录；

在系统中，用户设计为只有一种普通用户。以下为各种类型的用户对应的功能，其中，普通用户可以在市场种查看未售出交易。此外，考虑到普通用户会查询历史交易，在实际使用时，根据交易中的身份来进行已售出、已下架的访问许可。



1. **功能性需求**

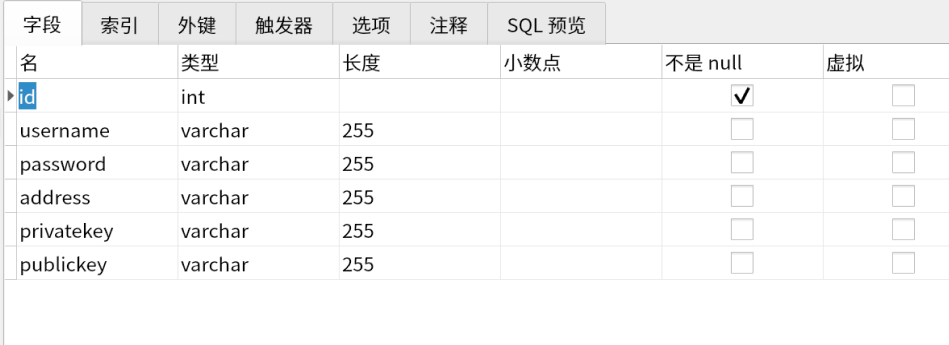
|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 用户角色 | 描述 | 备注 |
| 账号注册 | 普通用户 | 用户输入用户名，密码，地址进行账户注册 |  |
| 登录 | 普通用户 | 各类用户使用自己的密码进行登录。 |  |
| 市场交易查询 | 任何用户 | 任何用户都可在市场输入物品的名称、选择金额范围和地址进行查询。 |  |
| 市场交易申请 | 普通用户 | 用户在市场查询物品后，点击购买，填写地址等信息后付款即可购买物品。 |  |
| 账号充值 | 普通用户 | 用户登录后，输入充值金额即可给账户充值。 |  |
| 交易记录查询 | 普通用户 | 普通用户登录并输入交易可查看交易的详细记录。 |  |
| 用户信息查看 | 普通用户 | 用户登录后，可查看自身用户余额等信息。 |  |
| 历史交易查询 | 普通用户 | 用户登陆后，可在个人信息的历史交易中，查看自身参与过的交易。 |  |
|  |  |  |  |

1. **非功能性需求**

|  |  |
| --- | --- |
| 性能需求 | 1.系统能够适应400个用户，平均每个会话持续8分钟；  2.生成的web页面，通过速率100KBps在10秒内可以全部下载 |
| 可用性需求 | 正常运行时间:99.9%。 |
| 并发需求 | 能支持至少百个并发请求。 |
| 安全性需求 | 1.所有涉及信息的操作必须登录后才能完成；  2.系统只允许用户查看自己的信息，而不能查看其他用户的资料；  3.数据库进行加密。 |

1. **系统设计**
   1. **基本架构设计**
      1. **基本组件设计**

参考用户角色分析，用户为普通用户。普通用户进行注册，生成User实例。

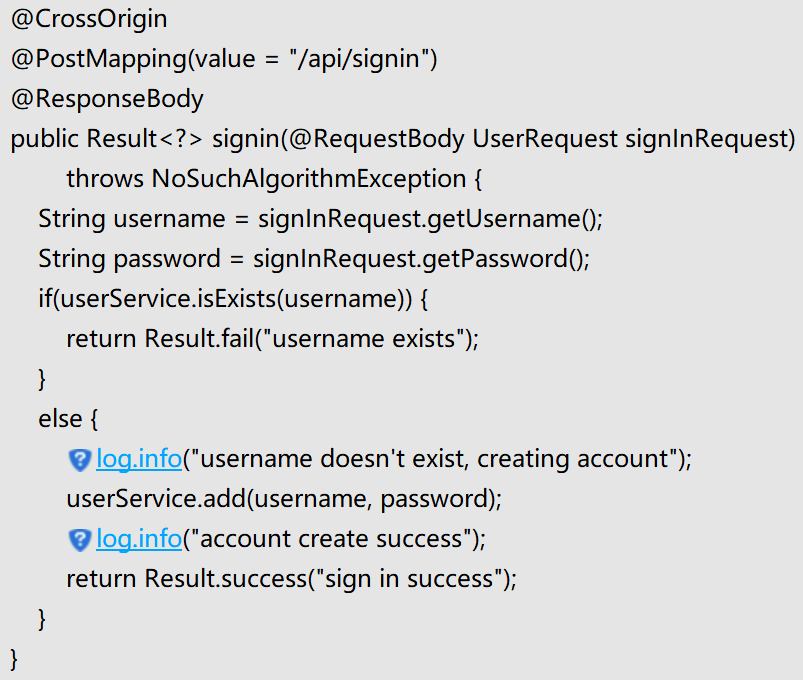


* + 1. **基本注册功能设计**

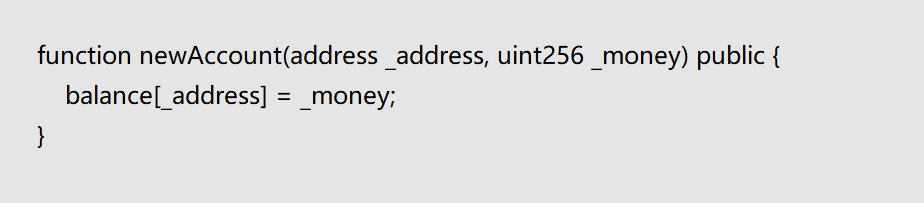
1用户首次注册通过调用signin，输入相关信息，生成唯一对应的address



前端调用函数

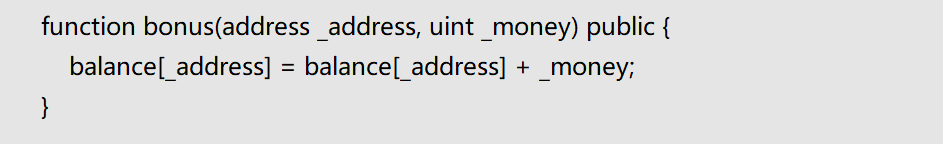


后端提供的接口



在合约中初始化该账户

2 用户在充值账户时需要调用bonus函数，函数从后台接收username以及用户输入的金额，将充值的金额储存进user中

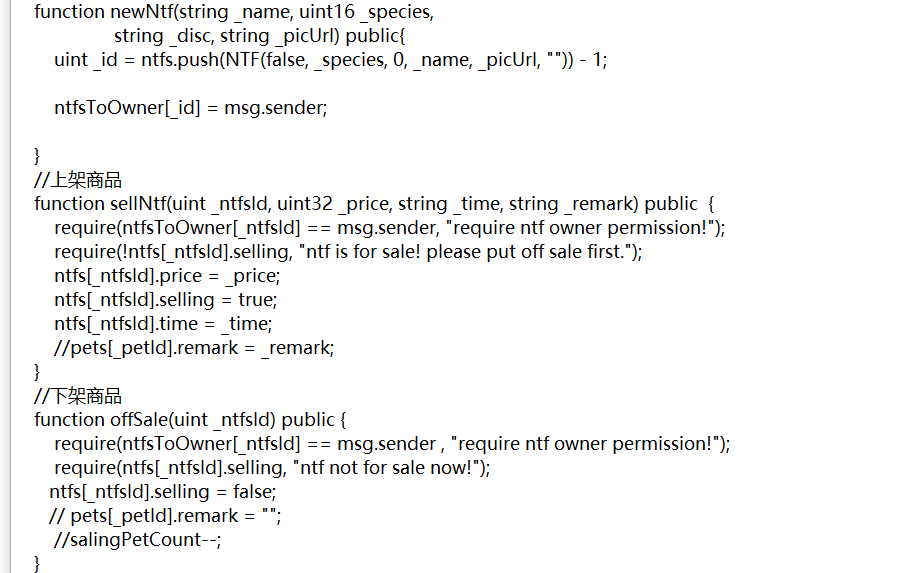


* 1. **功能实现函数设计**
     1. **主要功能函数设计**

buyNtf实现了购买商品的功能



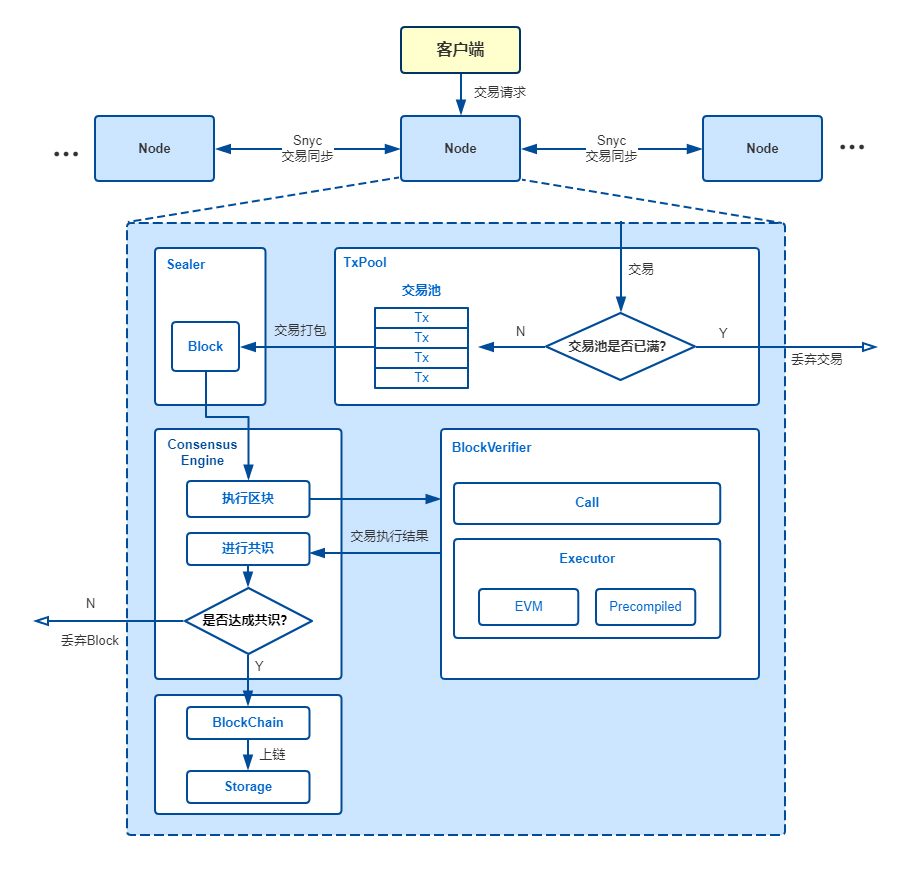
以下为生成新Nft，上下架Nft的函数



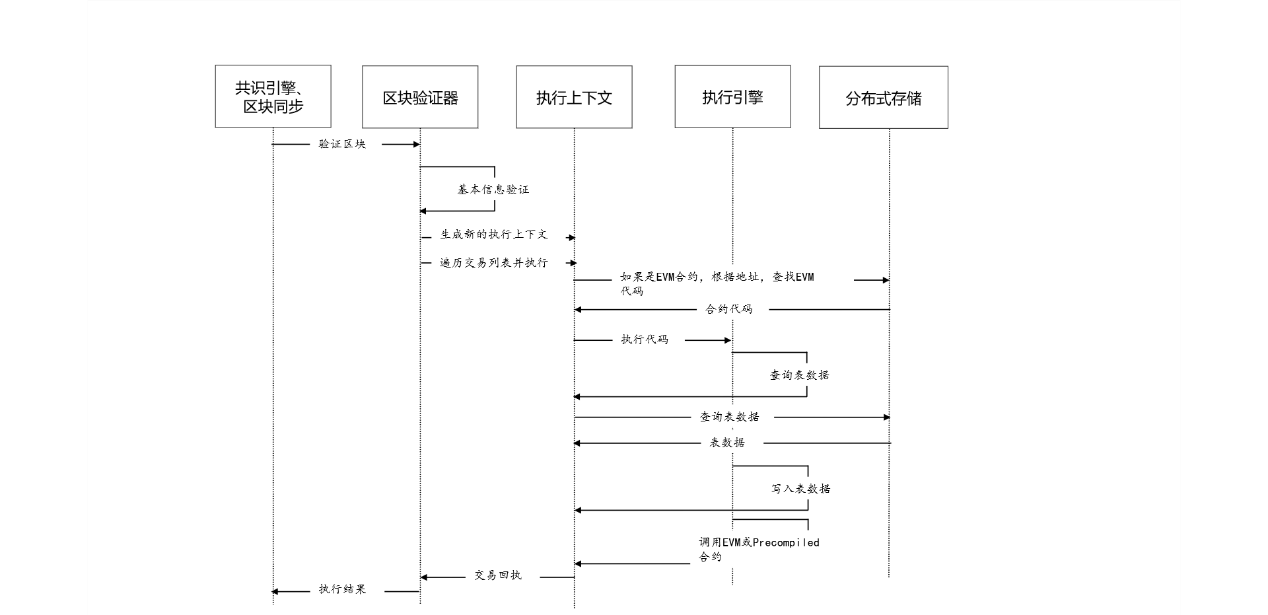
* 1. **区块链设计**
     1. **区块链结构**

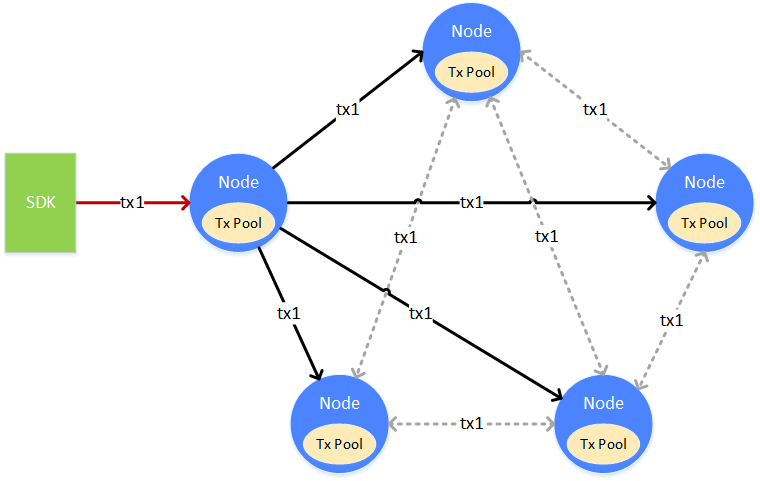
本系统采用单链四节点双群组的结构，分别是一个节点组成的仲裁群组和三个节点构成的用户群组。节点间的交易需验证一致性，使用pbft共识机制。

用户向节点发起RPC请求以发起交易，节点收到交易后将交易附加到交易池中，打包器不断从交易池中取出交易并通过一定条件触发将取出交易打包为区块。生成区块后，由共识引擎进行验证及共识，验证区块无误且节点间达成共识后，将区块上链。当节点通过同步模块从其他节点处下载缺失的区块时，会同样对区块进行执行及验证。



* + 1. **数据传输**





一笔交易（tx1），从客户端上发往某个节点，节点在接收到交易后，会将交易放入自身的交易池（Tx Pool）中供共识去打包。与此同时，节点会将交易广播给其它的节点，其它节点收到交易后，也会将交易放到自身的交易池中。交易在发送的过程中，会有丢失的情况，为了能让交易尽可能的到达所有的节点，收到广播过来交易的节点，会根据一定的策略，选择其它的节点，再进行一次广播。

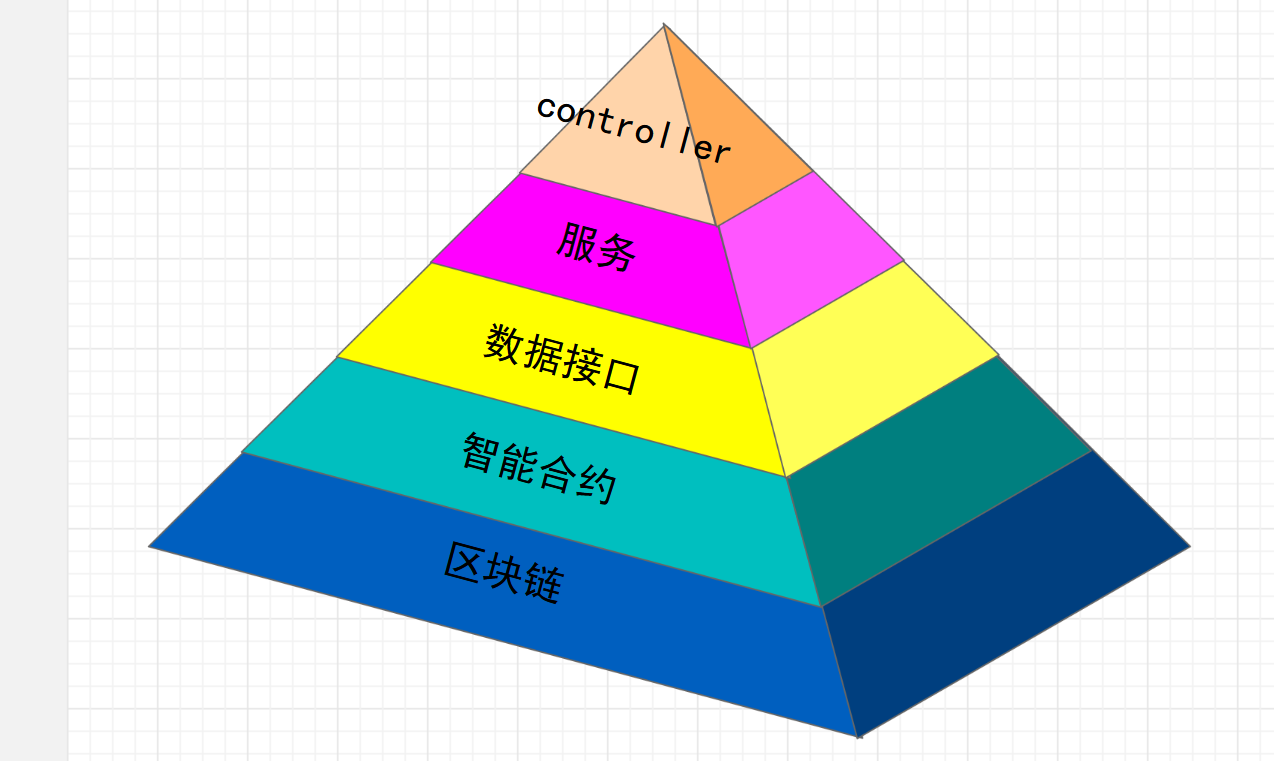
1. **后台设计**
   1. **环境设置**

Gradle：4.0.0

Jdk：13

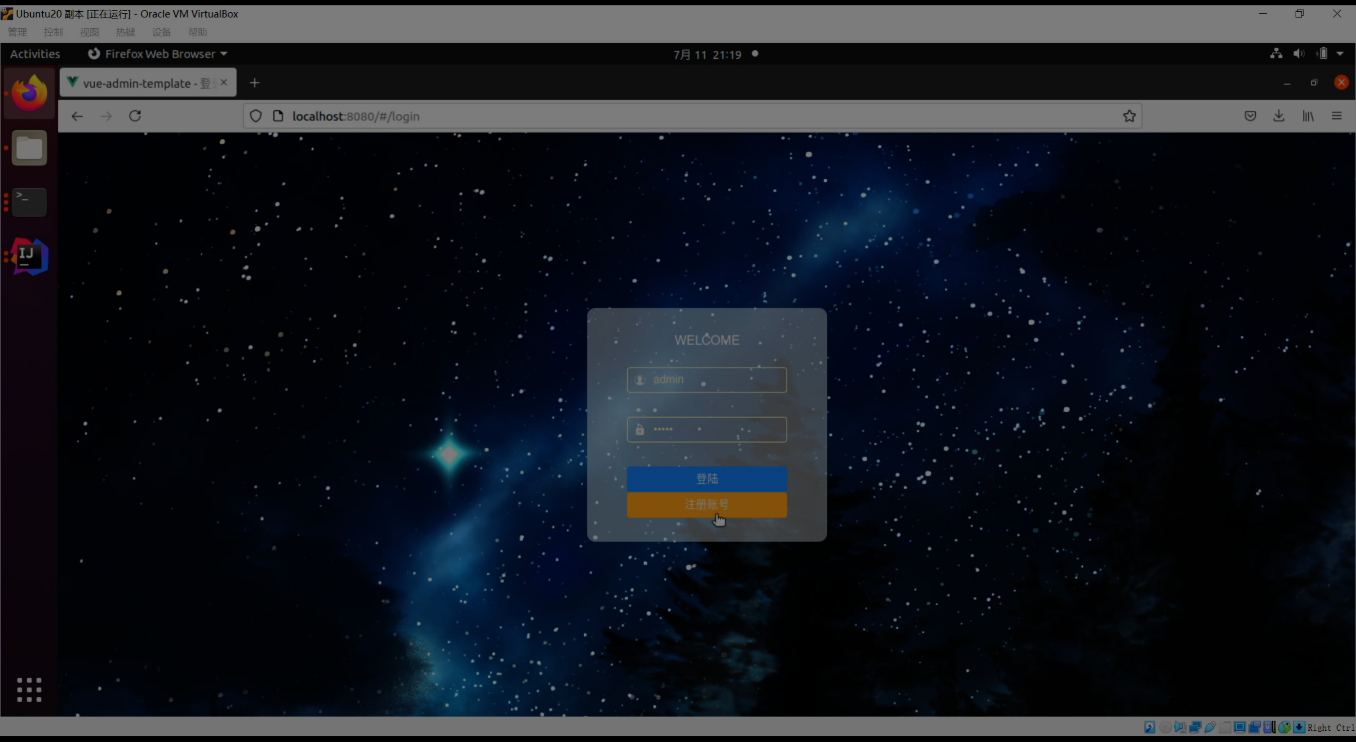
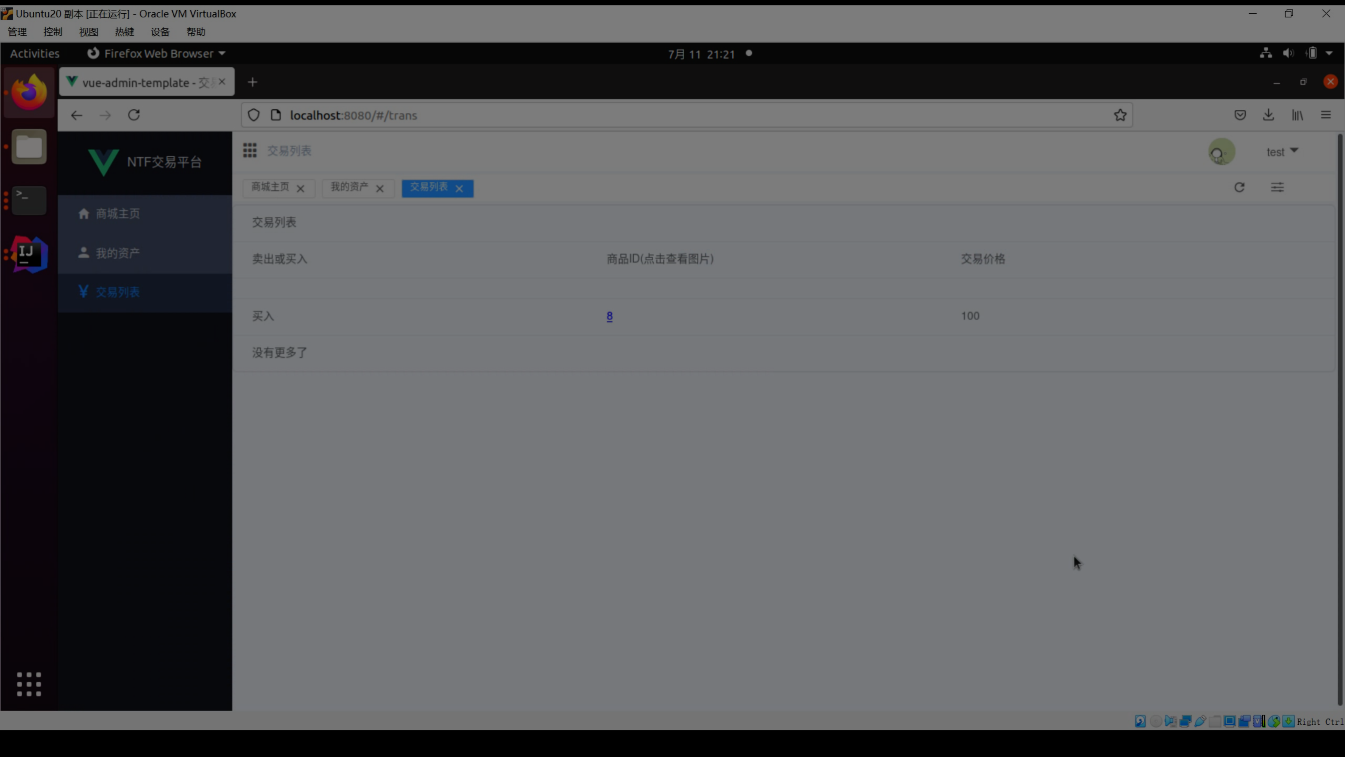
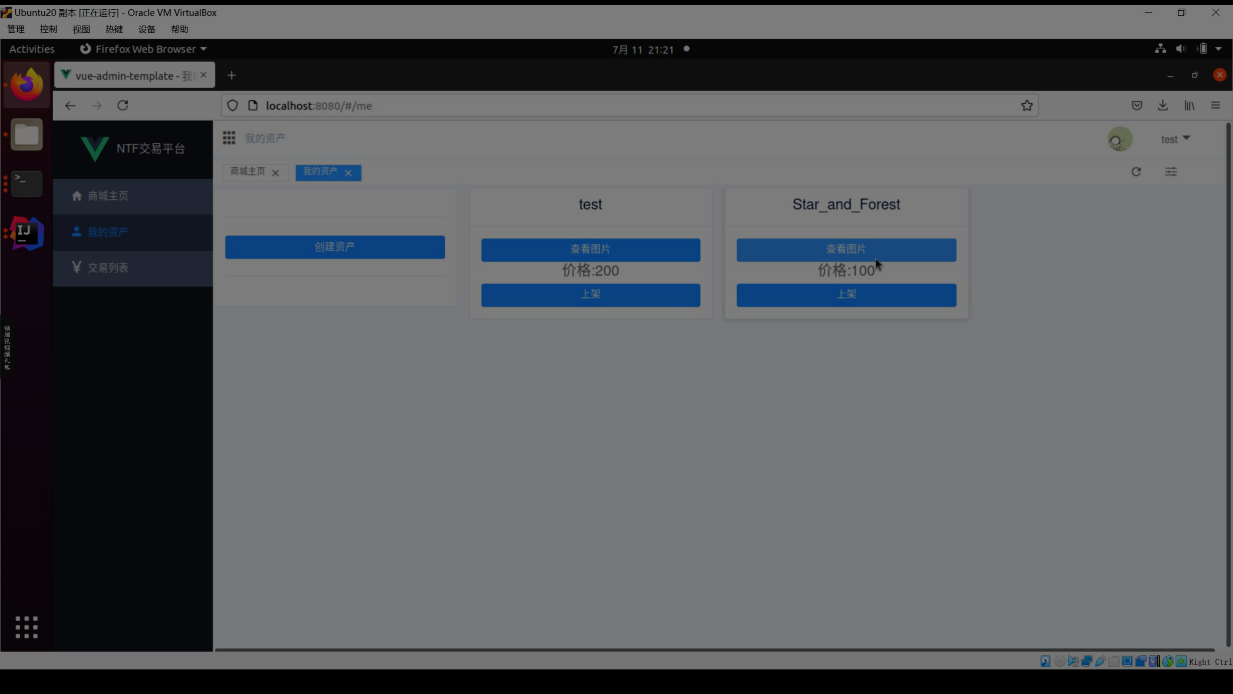
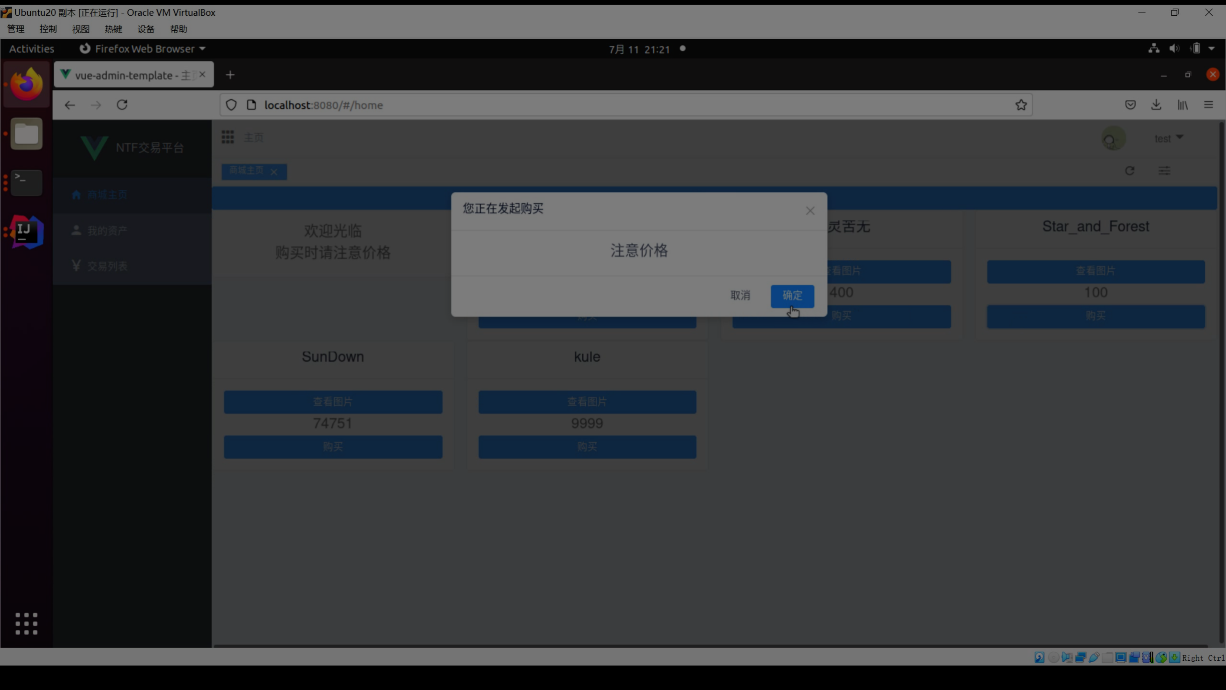
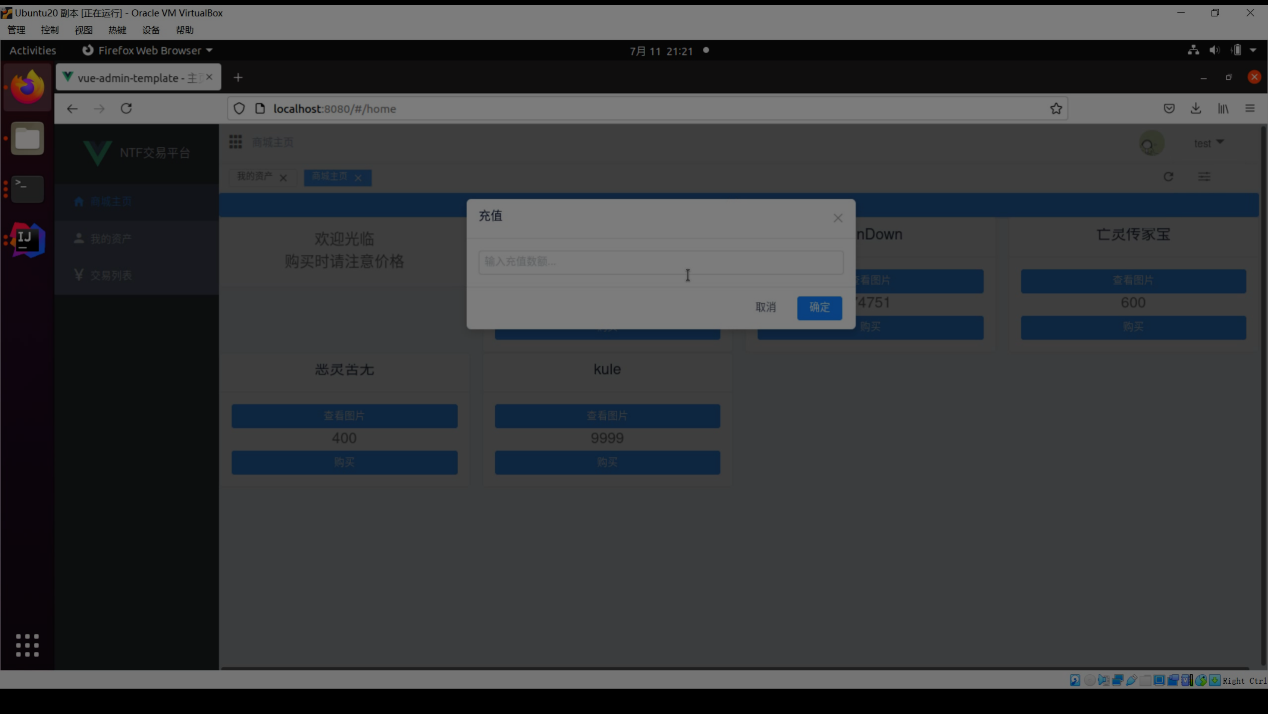
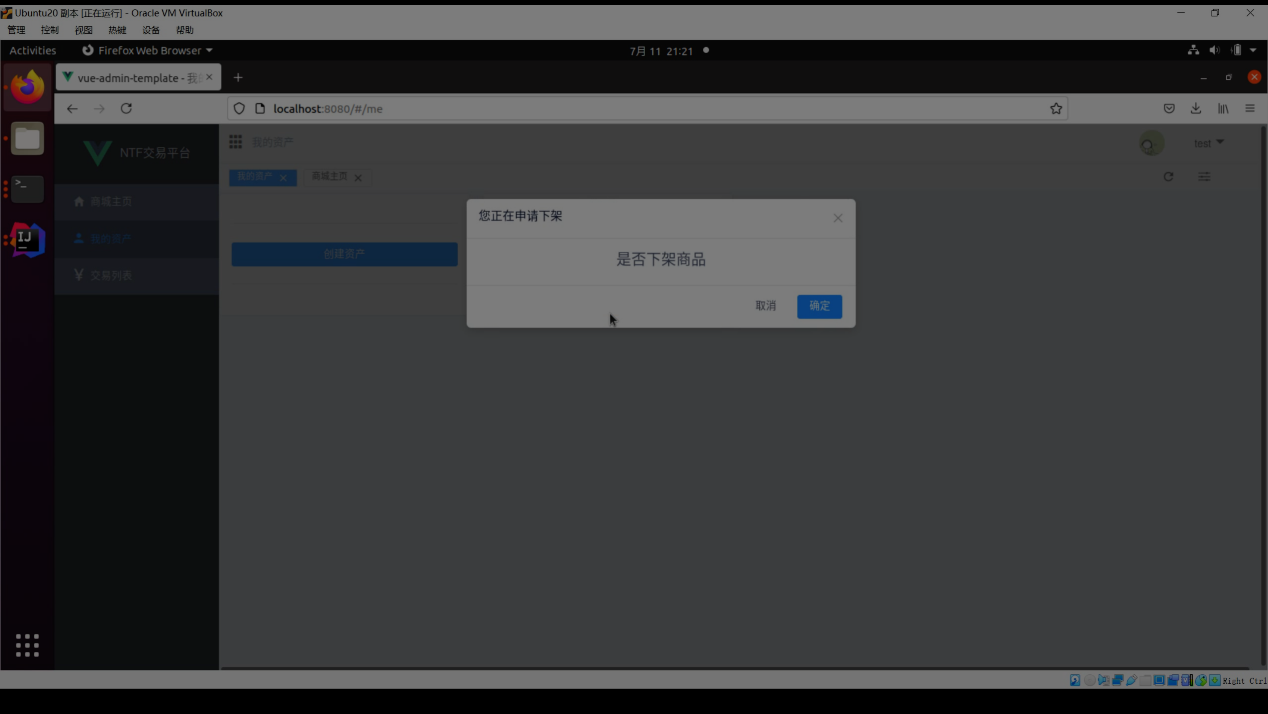
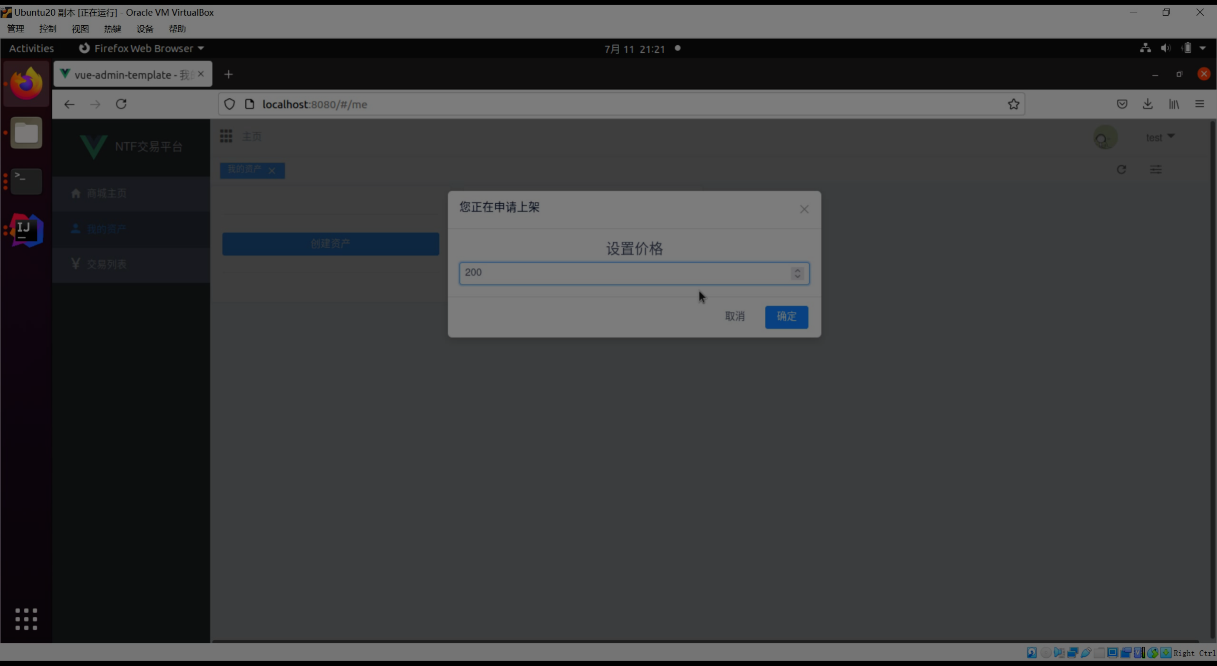
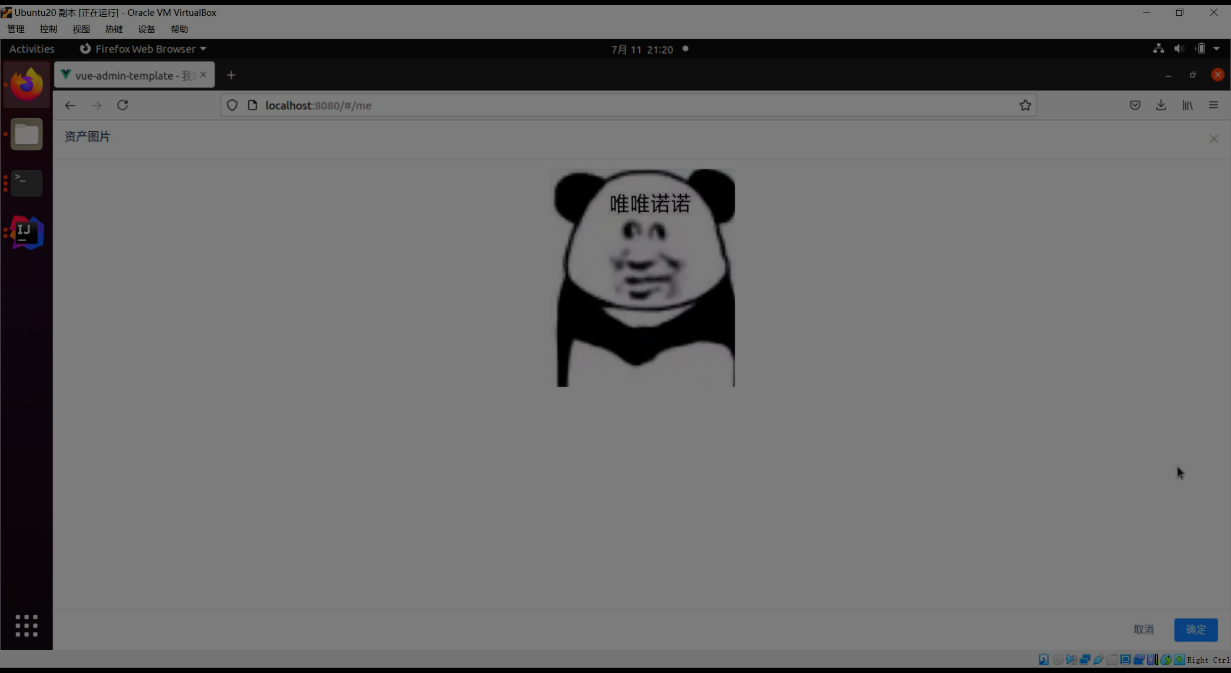
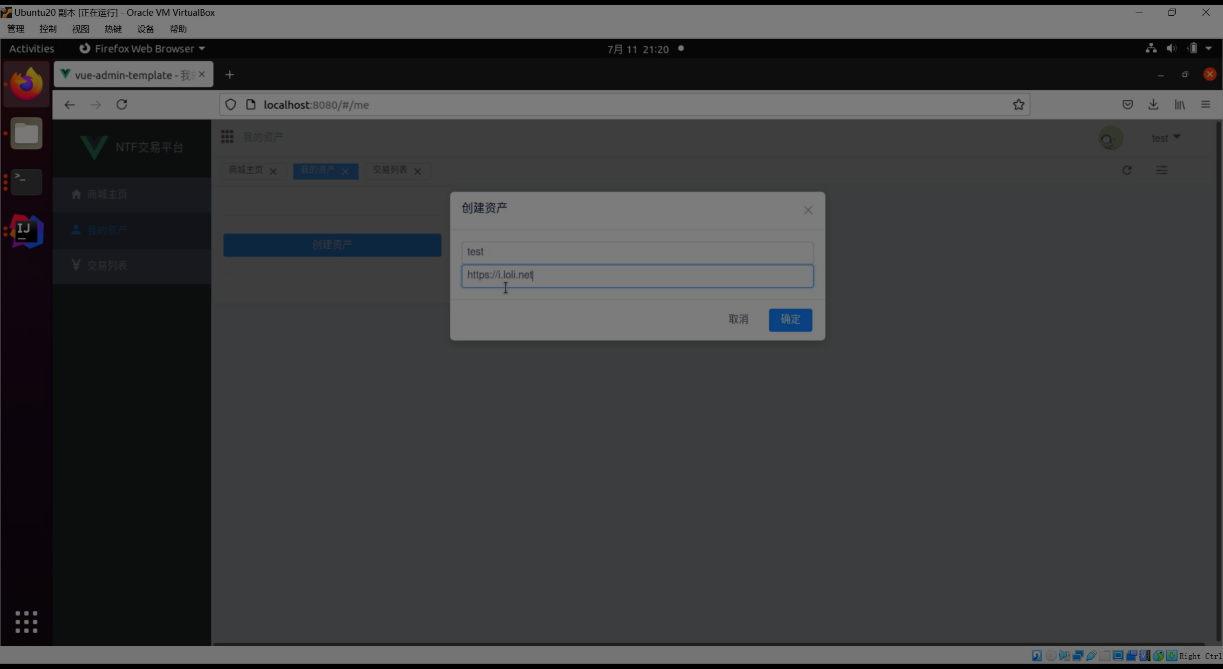
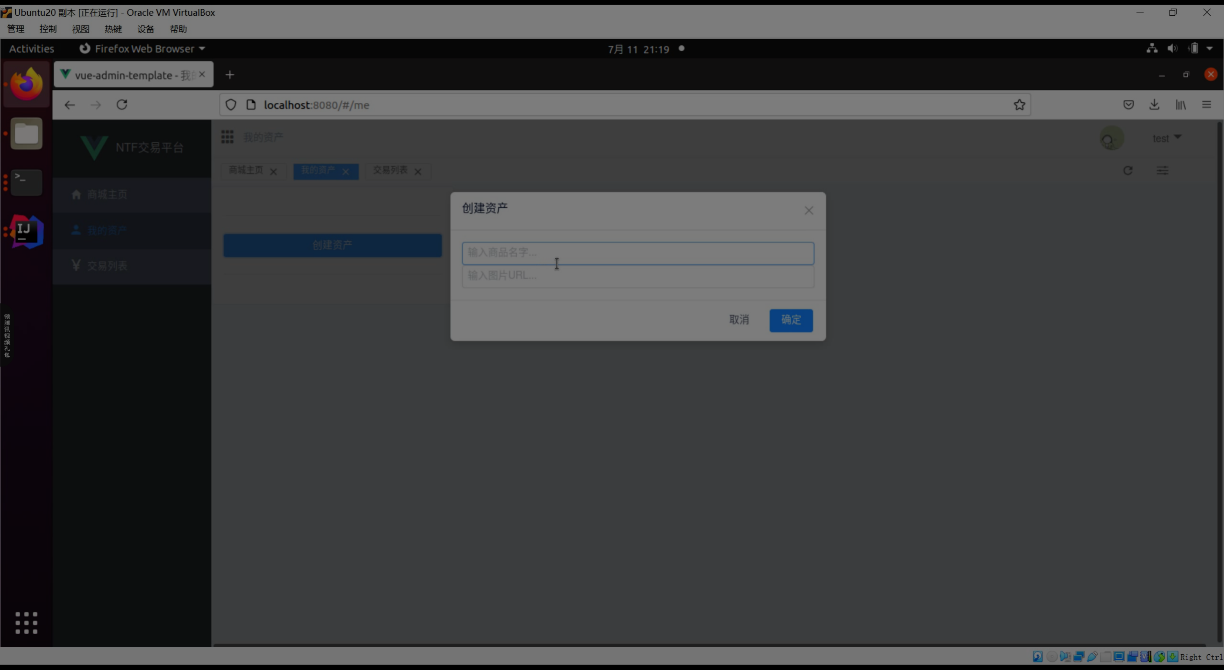
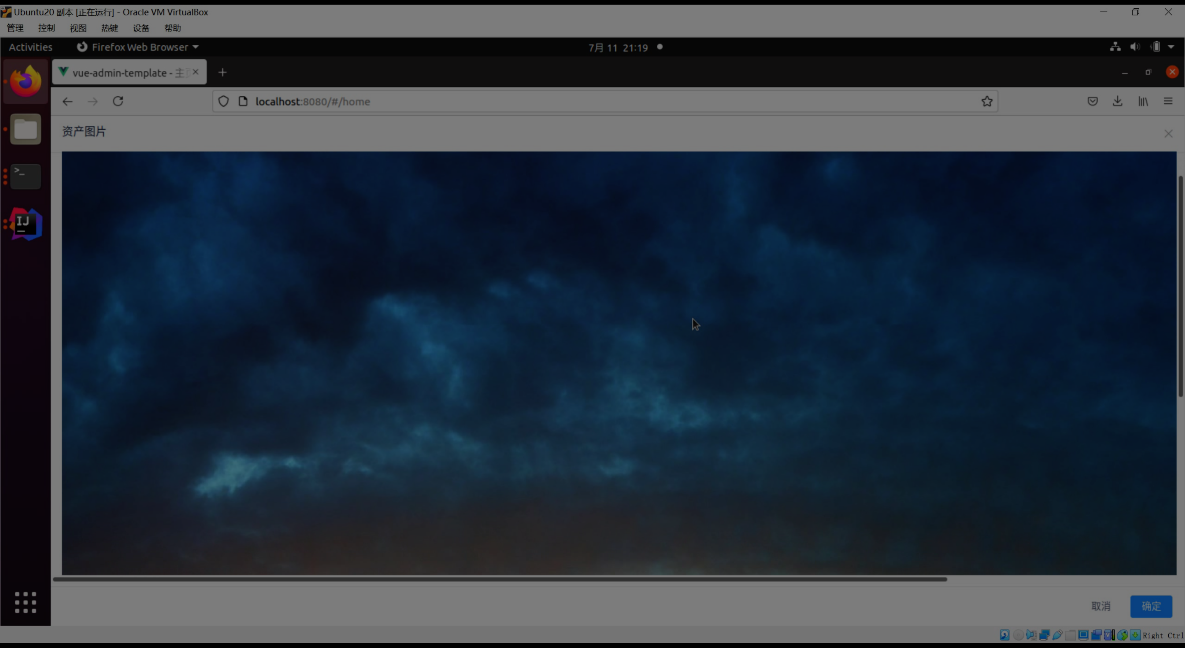
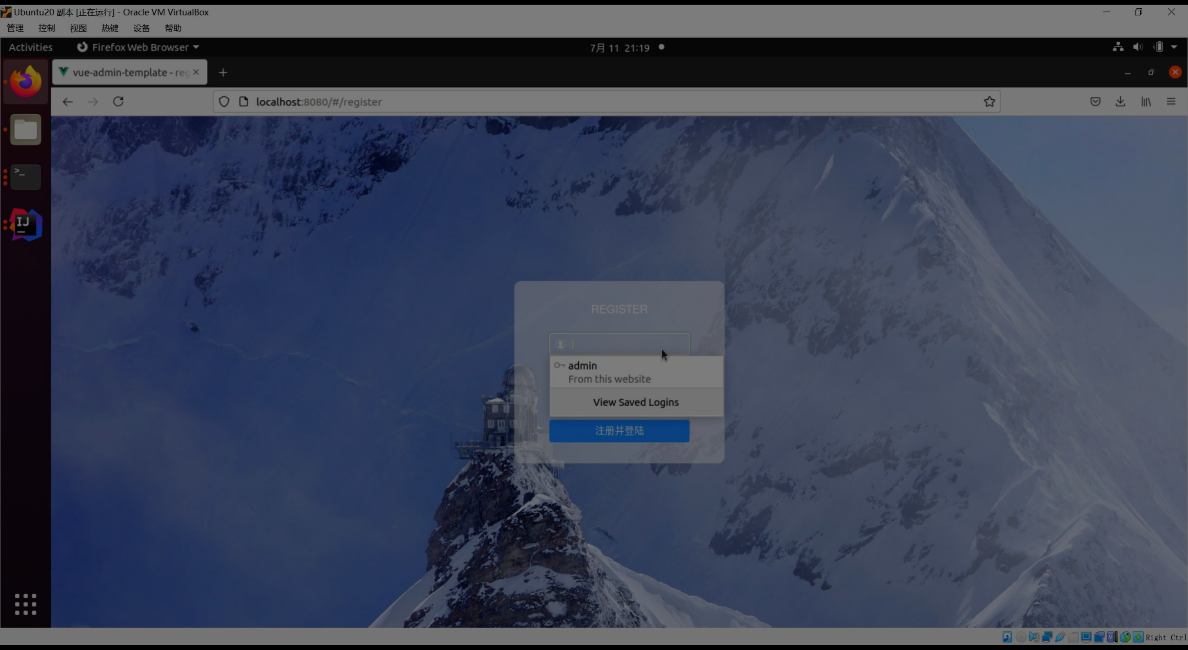
IDE : IntelliJ IDEA

* 1. **后端架构**

1. 

后端的整体架构大约分为3层：数据接口、服务层和Controller层，结构如上图所示。

1. 数据接口：在智能合约转化成java文件以后，对其暴露的函数接口进行封装，使接口更加美观，方便服务层进行调用。
2. 服务层：利用数据接口实现需求，使其可以被Controller层所调用
3. Controller：响应前端的请求，并返回所需的数据
4. **界面设计**



# **NFT交易市场系统测试报告**

**目录**

[**1. 引言 43**](#_Toc44498494)

[**1.1. 编写目的 43**](#_Toc44498495)

[**1.2. 项目背景与系统简介 44**](#_Toc44498496)

[**1.3. 术语 44**](#_Toc44498497)

[**2. 参考资料 44**](#_Toc44498498)

[**3. 测试概要 44**](#_Toc44498499)

[**3.1. 测试的概要介绍 44**](#_Toc44498500)

[**3.2. 用例设计方法 44**](#_Toc44498501)

[**3.3. 测试环境与配置 44**](#_Toc44498502)

[**4. 测试结果与缺陷分析 45**](#_Toc44498503)

[**4.1. 测试组织 45**](#_Toc44498504)

[**4.2. 测试时间 45**](#_Toc44498505)

[**4.3. 测试版本 45**](#_Toc44498506)

[**4.4. 覆盖分析 45**](#_Toc44498507)

[**4.5. 缺陷分析 49**](#_Toc44498508)

[**4.6. 残留缺陷和未解决的问题 49**](#_Toc44498509)

[**5. 测试建议与结论 49**](#_Toc44498510)

[**5.1. 建议 49**](#_Toc44498511)

[**5.2. 测试结论 49**](#_Toc44498512)

# 简介

## 编写目的

本文档为基于区块链的ntf交易系统的测试报告，目的在于总结测试阶段的测试情况以及分析测试结果，描述系统是否符合设计需求，是否已实现预期的功能目标。

## 参考资料

a. 需求规格说明书

b. 概要设计说明书

c. 详细设计说明书

## 测试范围

测试主要根据需求规格说明书以及相应的文档进行测试，包括功能测试和性能测试。

参加测试的主要功能包括：用户注册、用户登录、商城显示商品列表，用户上架商品，购买商品，浏览自己的资产等。

# 测试内容

## 测试类型

|  |  |  |  |
| --- | --- | --- | --- |
| 测试类型 | 测试内容 | 测试目的 | 所用的测试工具和方法 |
| 功能测试 | 1.用户登录和注册  2.用户上传和上架自己的艺术品  3.用户购买已上架的艺术品 | 核实所有功能均已正常实现，符合需求规格说明书的要求 | 采用黑盒测试，使用场景法设计用例，进行手工测试 |
| 单元测试 | solidity合约中的方法 | 确保solidity合约能够实现核心功能 | 由合约开发者进行，使用webase-front部署合约并调用 |
| Java后台接口 | 核实后端提供的接口符合描述，功能正确 | 由后端开发者进行，使用postman调用接口，测试正确的接口录入接口文档 |
| 前端页面按钮 | 核实网站页面显示和跳转的功能无误 | 由前端开发人员进行，人工点击进行测试 |

## 功能测试设计

### 用户注册并登录

|  |  |
| --- | --- |
| 基本流1 | (1)用户点击注册——(2)输入账户和密码——(3)注册成功——（4）跳转到商城页面 |
| 备选流1-1 | 输入的用户名已经存在 |
| 备选流1-2 | 输入含有非法字符 |

### 用户登录

|  |  |
| --- | --- |
| 基本流2 | （1）用户点击注册——（2）输入账户和密码——（3）注册成功——（4）跳转到商城页面 |
| 备选流2-1 | 输入的用户名不存在 |
| 备选流2-2 | 输入的密码错误 |

### 用户上架艺术品

|  |  |
| --- | --- |
| 基本流3 | （1）用户登陆商城——（2）点击我的资产——（3）创建资产——（4）输入图片url——（5）资产创建成功——（6）点击上架资产——（7）输入价格——(8)上架成功 |
| 备选流3-1 | 上传的图片url无法解析 |
| 备选流3-2 | 用户点击上架将已经拥有的资产上架 |

### 用户购买艺术品

|  |  |
| --- | --- |
| 基本流4-1 | （1）用户登陆商城——（2）选择商品列表中的商品——（3）点击购买——（4）确认购买——（5）购买成功 |
| 备选流4-1 | 用户账户余额不足 |
| 备选流4-2 | 商品已被别的用户购买 |

### 场景设计

|  |  |  |
| --- | --- | --- |
| 场景1-注册并登录 | 基本流1 |  |
| 场景2-注册的账户名重复 | 基本流1 | 备选流1-1 |
| 场景3-输入含有非法字符 | 基本流1 | 备选流1-2 |
| 场景4-成功登录 | 基本流2 |  |
| 场景5-用户名不存在 | 基本流2 | 备选流2-1 |
| 场景6-密码错误 | 基本流2 | 备选流2-2 |
| 场景7-创建并上架艺术品 | 基本流3 |  |
| 场景8-无法解析图片 | 基本流3 | 备选流3-1 |
| 场景9-上架已有资产 | 基本流3 | 备选流3-2 |
| 场景10-购买艺术品 | 基本流4 |  |
| 场景11-账户余额不足 | 基本流4 | 备选流4-1 |
| 场景12-商品已被购买 | 基本流4 | 备选流4-2 |

### 测试用例设计

|  |  |  |
| --- | --- | --- |
| 测试用例编号 | 场景 | 预期结果 |
| 1 | 场景1-注册并登录 | 成功登录，跳转到商城 |
| 2 | 场景2-注册的账户名重复 | 提示错误，重新填写 |
| 3 | 场景3-输入含有非法字符 | 提示错误，重新填写 |
| 4 | 场景4-成功登录 | 成功登录，跳转到商城 |
| 5 | 场景5-用户名不存在 | 提示用户名或密码错误 |
| 6 | 场景6-密码错误 | 提示用户名或密码错误 |
| 7 | 场景7-创建并上架艺术品 | 上架成功 |
| 8 | 场景8-无法解析图片 | 创建资产失败，回到基本流步骤3 |
| 9 | 场景9-上架已有资产 | 跳过基本流步骤3、4、5 |
| 10 | 场景10-购买艺术品 | 购买成功，可在“我的资产”中找到购买的资产 |
| 11 | 场景11-账户余额不足 | 提示账户余额不足，购买失败 |
| 12 | 场景12-商品已被购买 | 提示购买失败，商品已下架 |

# 测试执行情况及结果

## 测试环境

Java SDK：openJdk13

数据库：mysql 8.0.25

部署环境：Ubuntu 20.03 LTS

## 测试结果

功能测试用例覆盖率为100%，用例通过率为100%，测试未检查到系统缺陷。

# **NFT交易市场系统项目总结**

本次区块链项目实训中，我们组的课题为基于FISCO-BCOS的NFT交易市场系统。这个项目与之前做过项目的最大区别便是在后端需要将数据发送到FISCO-BCOS节点处理与储存。由于大家都是在实训过程中才开始接触并学习使用FISCO区块链系统，对Java SDK也并不熟悉，组内也没有人有前端开发经验，加之由于环境配置问题每周的个人任务并不轻松，我们的项目工作迟迟没有进展。

我们虽然早早确认了项目的基本框架：前端使用VUE开发网页，后端采用Springboot开发工具，以合约和SDK相结合的方式与FISCO的交互。但由于我们缺乏相关的知识，项目前期基本上每个组员都在学习自己负责部分相关的内容，直到最后一周才真正开始实现。

在开发过程中，由于对基于Springboot和VUE的前后端分离式系统开发的经验较少，我们上网查阅了许多资料，并相互交流，最终我们成功在9天内完成了所有代码的编写。虽然仍然有部分预想的功能未能实现，但相对而言已经完成了一个能用的系统了。

这次实训时间较为紧迫，知识掌握也不够充分，但我们组依然能够顺利地按照项目章程的规划如期完成，体现了大家的团队合作精神。我们不仅学习到了区块链的相关知识，了解了它在实际项目中的运用，还对Springboot系统的开发更加熟练。在之后的实践中，我们将继续对这个系统进行完善，争取开发出一个高性能的区块链系统。

|  |
| --- |
| **团队项目实训报告成绩评定：** **分** |