

**《软件工程》项目报告**

**题目： NFTown**

|  |  |  |
| --- | --- | --- |
| **课程名称：** | **软件工程** | |
| **专业班级：** | **CS2007** | |
| **组 名：** | **310ba** | |
| **同组成员：** | **学号：** | **U202015502** |
|  | **姓名：** | **钟晟** |
|  | **学号：** | **U202015517** |
|  | **姓名：** | **邱攀攀** |
|  | **学号：** | **U202015515** |
|  | **姓名：** | **方一舟** |
|  | **学号：** | **U202090065** |
|  | **姓名：** | **杨少生** |
| **指导教师：** | **瞿彬彬** | |
| **报告日期：** | **2022/11/16** | |

**计算机科学与技术学院**

目录

**[任 务 书 2](#_Toc29604)**

[一 总体要求 2](#_Toc19741)

**[3. 根据需求分析进行总体设计、详细设计、编码与测试等。 2](#_Toc17715)**

[二 基本内容 2](#_Toc8989)

**[1 问题定义 3](#_Toc31691)**

[1.1 项目背景与意义 3](#_Toc2994)

[1.2 项目基本目标 4](#_Toc31387)

[1.3 可行性分析 4](#_Toc19636)

[1.4 人员管理和项目进度管理 4](#_Toc18911)

**[2 需求分析 6](#_Toc18474)**

[2.1 系统E-R图 6](#_Toc15187)

[2.2 UML相关需求分析图 6](#_Toc26991)

[2.3 原型系统设计 7](#_Toc4936)

**[3 概要设计和详细设计 9](#_Toc22626)**

[3.1 系统结构 9](#_Toc25491)

[3.2 类图等 10](#_Toc2869)

[3.3关键数据结构定义 10](#_Toc28321)

[3.4 关键算法设计 11](#_Toc21399)

[3.5 数据管理说明 11](#_Toc23667)

**[4 实现与测试 13](#_Toc10831)**

[4.1实现环境与代码管理 13](#_Toc6238)

[4.2 关键函数说明 13](#_Toc21628)

[4.3 测试计划和测试用例 14](#_Toc9647)

**[2.下拉刷新，如图4-3-3所示。 15](#_Toc6173)**

**[5. 单击图4-3-5中的“立即购买”，即可交易。 17](#_Toc27314)**

[4.3.3 功能测试 22](#_Toc8586)

**[5 总结 25](#_Toc11432)**

[5.1 用户反馈 25](#_Toc2567)

[5.1 全文总结 25](#_Toc3248)

**[6 体会 26](#_Toc12184)**

**[附录 27](#_Toc5292)**

# 任 务 书

## 一 总体要求

1. 综合运用软件工程的思想，协同完成一个软件项目的开发，掌软件工程相关的技术和方法；

2. 组成小组进行选题，通过调研完成项目的需求分析，并详细说明小组成员的分工、项目的时间管理等方面。

3. 根据需求分析进行总体设计、详细设计、编码与测试等。

## 二 基本内容

根据给出的题目任选一题，自行组队，设计与开发中软件过程必须包括：

**1. 问题概述、需求分析：**正确使用相关工具和方法说明所开发软件的问题定义和需求分析，比如NABCD模型，Microsoft Visio，StarUML等工具 (20%)；

**2. 原型系统设计、概要设计、详细设计**：主要说明所开发软件的架构、数据结构及主要算法设计，比如墨刀等工具（35%）；

**3. 编码与测试**：编码规范，运用码云等平台进行版本管理，设计测试计划和测试用例（30%）；

**4．功能创新**：与众不同、特别吸引用户的创新（10%）；

**5. 用户反馈**：包括用户的使用记录，照片，视频等（5%）。

1 问题定义

## 项目背景与意义

### N(Need 需求)你的创意解决了用户的什么需求?

NFT，全称为Non-Fungible Token，指非同质化通证，实质是区块链网络里具有唯一性特点的可信数字权益凭证，是一种可在区块链上记录和处理多维、复杂属性的数据对象。创作者可以通过NFT来售卖自己的作品，同时也可以很有效的避免盗版的产生，维护创作者的权益。同时，我们创建了一个UGC社区，大家可以在其中进行交易、交流。

### A(Approach 做法)你有什么招数来解决用户的痛苦或问题？

通过简洁明了的操作界面，丰富的商品种类，让元宇宙等宏大抽象的概念展现在人们面前，让人们更靠近商品层次，设计从生产到交易，从购入到售出的系统化的创作与交易平台。让人们以更传统的方式来接受新兴事物。

### B(Benefit 好处)你这个产品或服务会给用户带来什么好处？

其一，随着数字化发展迅猛，数字化虚拟产品越来越多，物质生活越来越富足，人们更加注重精神层面的需求。虚拟世界中的虚拟身份、游戏装备、皮肤等数字化资产迅速成为了人们精神需求中的顶流需求之一，但目前这些数字资产都有一个致命的缺陷，那就是中心化，机构平台几乎掌控了所有的控制权。想要将数字资产变成真正的私有资产，那就必须实现去中心化。NFT就是一个最好的解决方案，NFT是区块链上的有价编码，可以永久保存用户对数字资产的所有权。

其二，NFT具备的唯一性保证人们在区块链社区中拥有一个不可替代的身份。

### C(Competitors 竞争)你的产品有没有类似的竞争者，他们的产品怎么样？

在国外有着许多类似的项目，但是在国内，由于比特币等虚拟货币的不流通，大部分NFT产品均由官方来制作以及拍卖交易，而大部分用户只能通过网站或软件制作出自己的NFT而不能进行交易和流通，而我们的软件作为一个去中心化的平台，采用UGC的策略，建立一个交易社区，在当今国内市场上还是较为少见的。

### D(Delivery 推广)你如何推销你的产品？

通过在特定网站投放广告以及相应用户群体在社区中的转发分享来传播。

## 项目基本目标

基本实现一个NFT商城的功能，包括上传、出售、购买、展示等功能。

## 可行性分析

本软件的外观界面以及用户登录、NFT信息存储等后端功能的实现可行性较高，但是对于NFT上链等功能实现较为困难。最后对于支付和提现功能的实现，由于支付功能主要依托于国内主流平台，其申请支付的首要条件就是需要公司，对于现在的我们实现基本不可行。因此，我们计划做一个内测版本，其中的交易并不需要进行相应支付，支付功能可在未来有能力后再进行实现。

## 人员管理和项目进度管理

### 项目分工

|  |  |  |
| --- | --- | --- |
| **成员** | **负责模块** | **团队贡献度** |
| 邱攀攀 | NFT区块链部分开发 | 25% |
| 杨少生 | 后端开发 | 25% |
| 钟晟 | 前端开发 | 25% |
| 方一舟 | 美工设计，作图 | 25% |

**表X 项目分工**

* + 1. **项目进度安排**

2 需求分析

**2.1 系统E-R图**

我们小组在前期通过一系列的调研和分析，最后的系统E-R图如下图2-1所示：

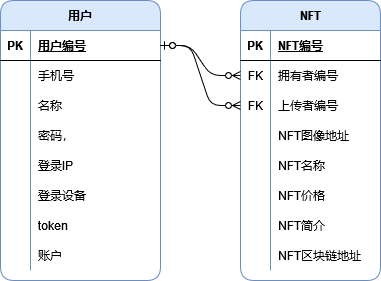


图2-1 系统E-R图

**2.2 UML相关需求分析图**

系统的用例图如图2-2所示：

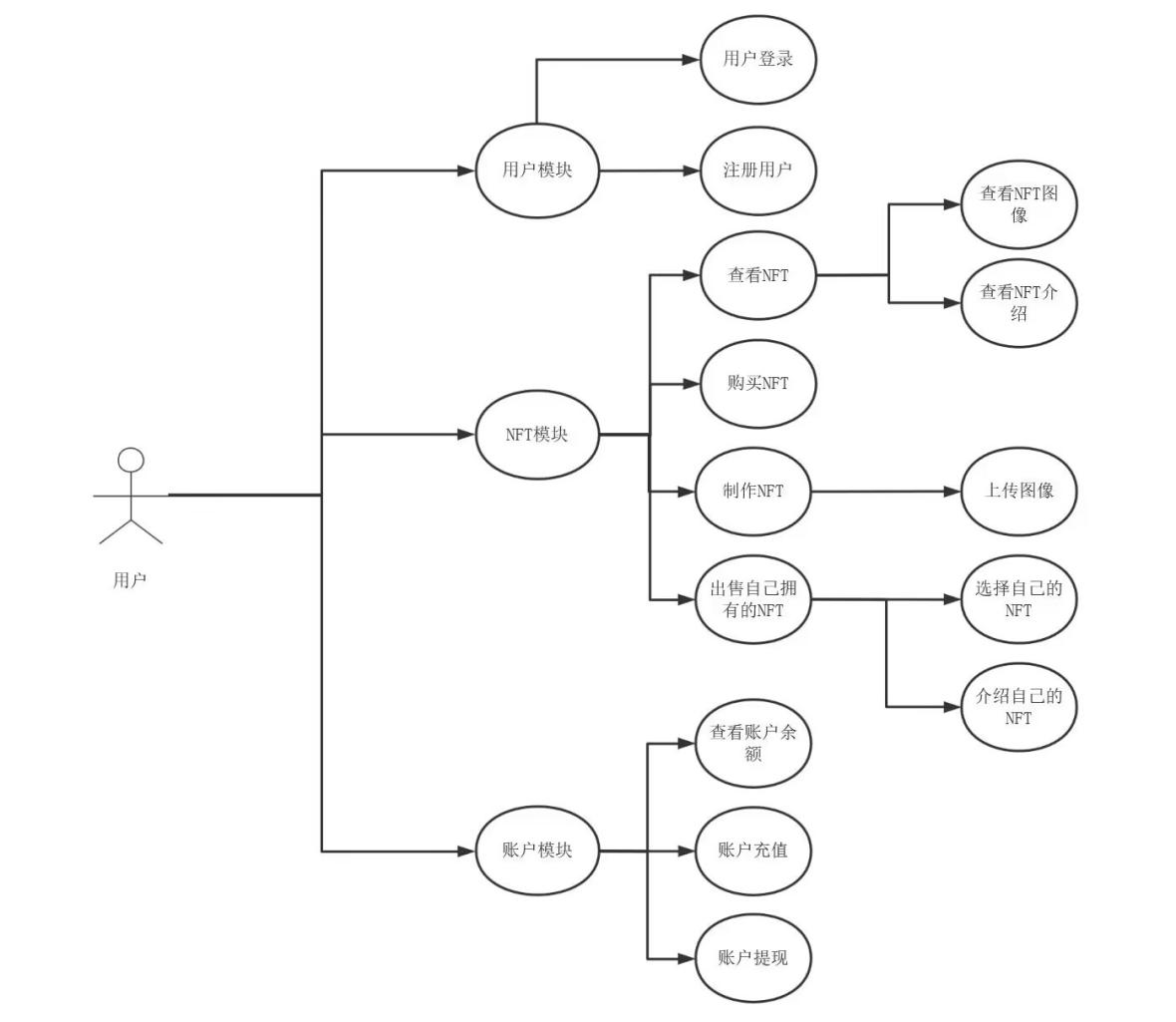


图2-2 系统UML需求分析图

**2.3 原型系统设计**

**2.3.1 设计工具**

墨刀

**2.3.2 设计思路**

计划开发一个安卓手机应用，因此初步设计八个页面，主页面用于展示NFT，详情页用于展示NFT详细信息，交易页用于展示热门NFT,搜索页用于对NFT进行搜索，上传和出售页用于上传和出售NFT,账户页用于用户主页，个人页用于展示个人收藏在售信息。

**2.3.3 设计成果**

原型系统效果详见网页：

*https://orgnext.modao.cc/app/F5gp5MXNrlju39ISuYFU3J*

部分页面效果图如下：

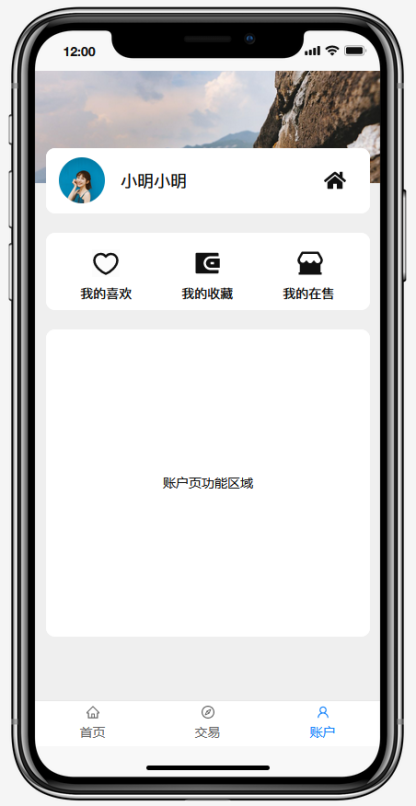
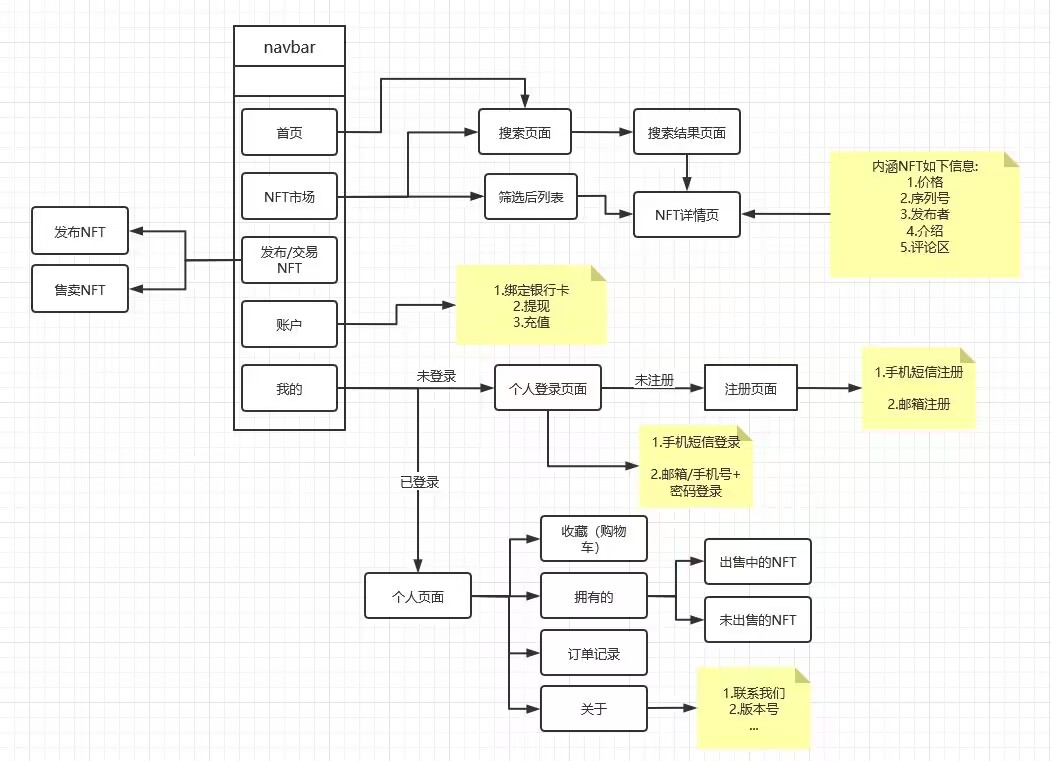


图2-3 部分原型系统页面图

# 3 概要设计和详细设计

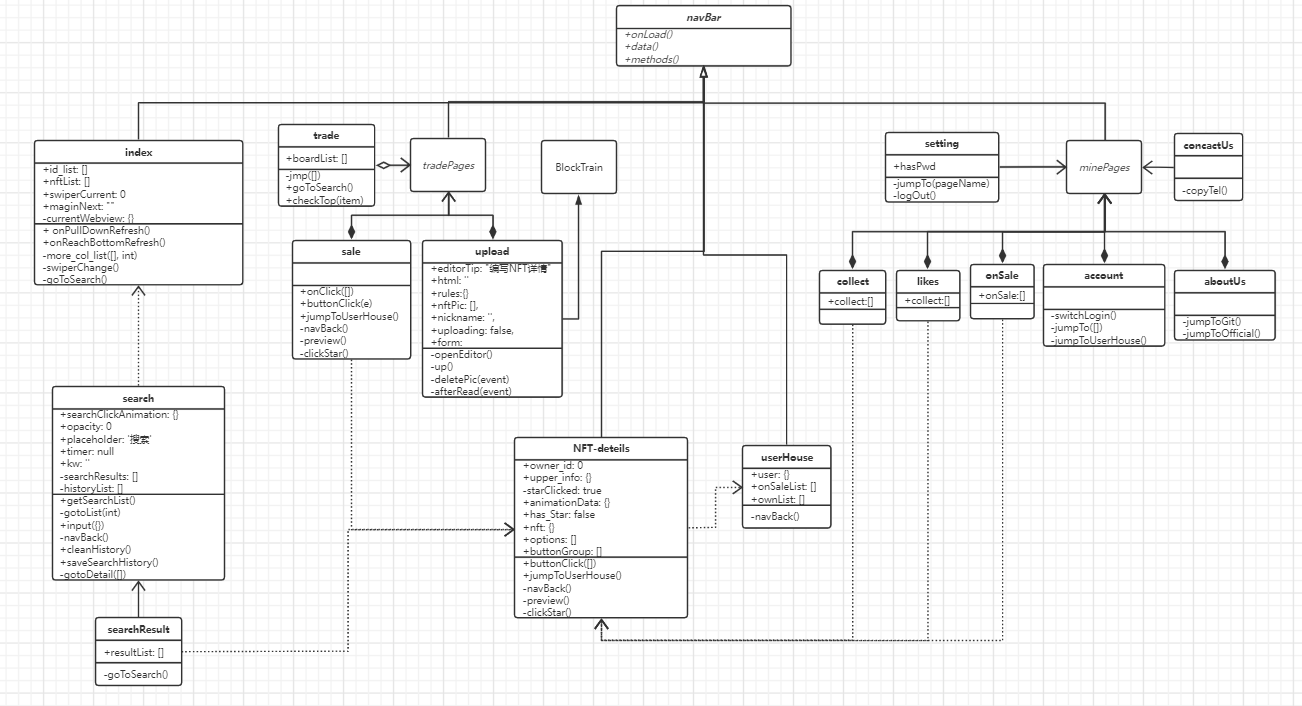
**3.1 系统结构**

这部分可根据用户需求，设计和规划一个系统，说明清楚系统应该有哪些功能模块，每个模块做什么。最后给出完整的系统结构图，以及相应的接口设计等。



* 注册：注册功能分为四个步骤，填入用户名、获取验证码、设置密码、用户信息的后台存储。
* 登陆：登陆功能采用了现代化的app登录设计模式，可以使用短信验证一键登录，也可以使用账号密码登录，经过与后端的用户信息对比成功才能成功登录。
* 浏览界面：在首页可以欣赏NFT库中数字藏品，支持下拉刷新和触底刷新以及搜索功能。可以点击藏品查看详细信息。遇到喜欢的藏品可以将其mark收藏。
* 生成NFT：在onLoad界面完善想要生成的NFT信息，在合理的报价以及信息填充后，加入区块链的使用信息，最终完成电子藏品的生成。
* 在线交易：在onSale界面欣赏在售数字藏品，发现理想的数字藏品后可以直接进行交易，交易内容在链上广播，该NFT后台持有信息更新

**3.2 类图等**



运用类图等方法说明系统的设计。

**3.3关键数据结构定义**

这部分要写的：（1）首先描述系统中要处理那些数据，每种类型的数据包括哪些数据项，每个数据项的数据类型；（2) 描述这多种数据在系统中如何关联，可通过图直观的说明这多种数据间的关联。

该项目使用的数据结构主要为Array，Queue和Tree。

首先定义了藏品结点NFT，定义如下

Struct NFT{

Char\* \_id;//在数据库中的唯一标识

Upper\* upper;//指针指向发行者的详情界面

Owner\* owner;//指针指向拥有者的详情界面

NFTdetails\* detail;//指针指向该数字藏品的详情信息界面

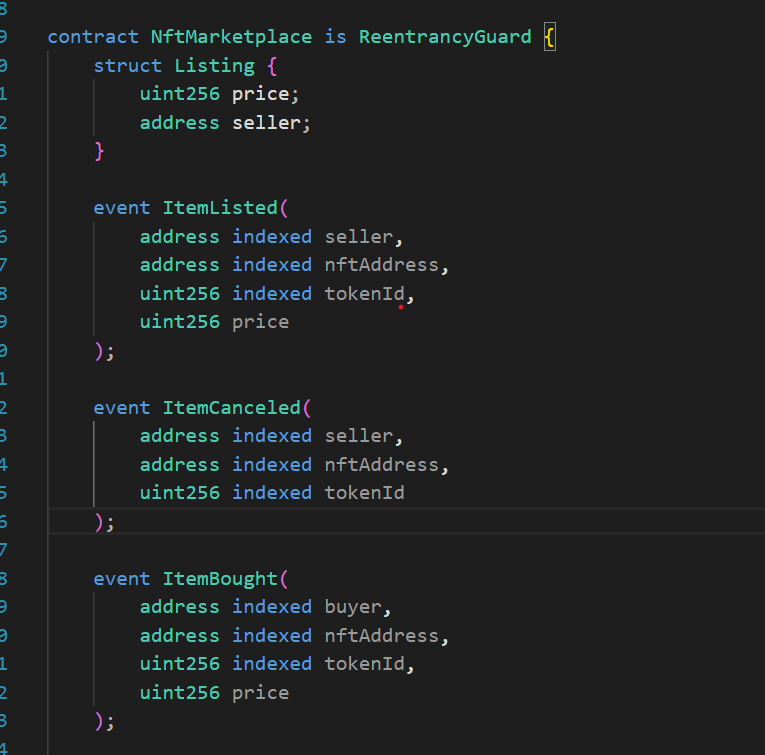
}

同时维护一个数组用于保存各个结点的信息，该队列可以用于渲染like、trade、index界面的信息，通过NFT.detail.like和NFT.detail.price进行推荐以及排序等操作。队列的结构则主要体现在upload和trade页面。上传和交易NFT时通过push和pop操作可以很便捷快速地完成数据库内数据的增删。

树的结构主要用于对数据库的查询上，相较于数组的查询有了很大的效率上的提升。

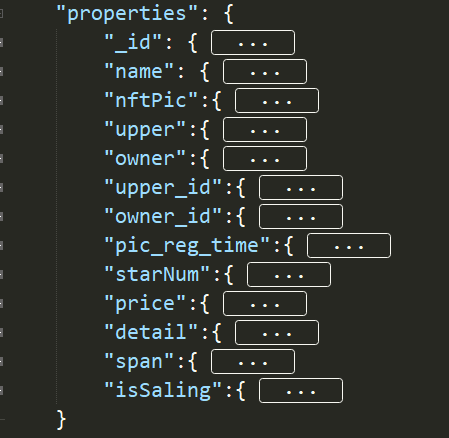
**3.4 关键算法设计**

此处主要是需要在solidity之中编写智能合约的时候尽量的在交易发生的时候使用较少的gas，所以需要设计较为高效的数据存储结构，以及简洁的交易程序。



**3.5 数据管理说明**

获取链上的信息，将上链成功的NFT详细信息（如图所示）保存到数据库中，如此便在后台完成了一件数字藏品的生成。而在交易模块则会涉及到藏品的所属问题。成功交易后交易双方的账户信息将会改变，主要改变内容为（如图所示）藏品列表和账户余额（因为是该软件自定义的虚拟货币，因而不具备法偿性和强制性等货币属性）





# 4 实现与测试

**4.1实现环境与代码管理**

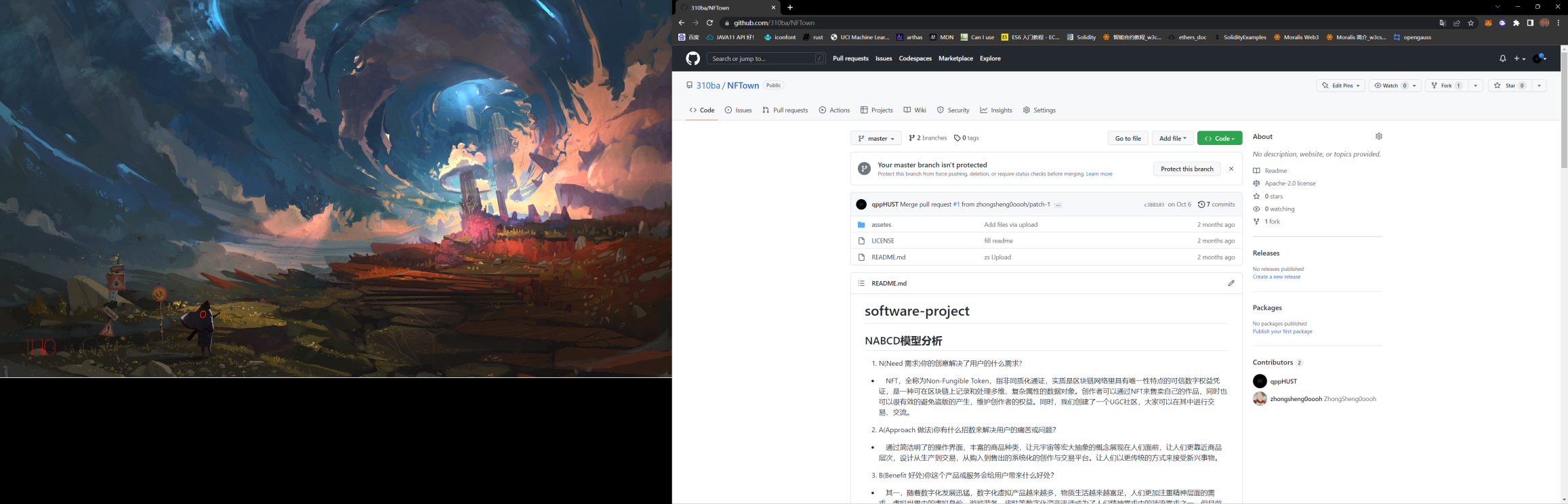
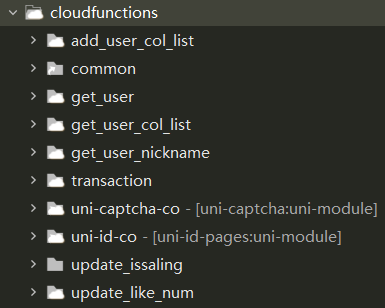


图4-1 码云平台代码签入示例

**4.2 关键函数说明**

程序代码部分在这里不需要给出来，只需要叙述清楚在系统中包括哪些函数，各函数的说明，如何利用这些函数实现系统各模块的功能，以及函数间的调用关系（可用图表示出来）。程序详见附录。



(1).首先是在数据库中封装的一些的云函数。

1. function add\_user\_col\_list(usr\_id, pic\_id)//usr\_id是用户id，pic\_id是数字藏品id，用于上传数字藏品时更新数据库。
2. Function transaction(usr\_id,pic\_id)//usr\_id是用户id，pic\_id是数字藏品id，封装的交易函数。
3. Update\_issaling(pic\_id)//pic\_id是数字藏品id，用于更新数字藏品的出售状态。
4. Update\_like\_num(pic\_id)//pic\_id是数字藏品id，用于更新数字藏品的收藏次数。
5. .还有一些其他的关键函数和函数在代码中的实现说明如下：
6. more\_col\_list(id\_list, flag)//id\_list是界面上加载的数字藏品队列，flag是用于判断操作是上拉触底还是下拉刷新。该函数用于主页的展示。
7. mapState('m\_nft', ['nft'])// 将 m\_nft 模块中的 nft映射到当前页面中使用

**4.3 测试计划和测试用例**

手动测试，模拟用户登录，发布，交易nft藏品

**4.3.1 操作测试**

1.由桌面应用图标（图4-3-1）点击进入应用，如图4-3-2所示。



图4-3-1 桌面应用图标



图4-3-2 打开效果

2.下拉刷新，如图4-3-3所示。

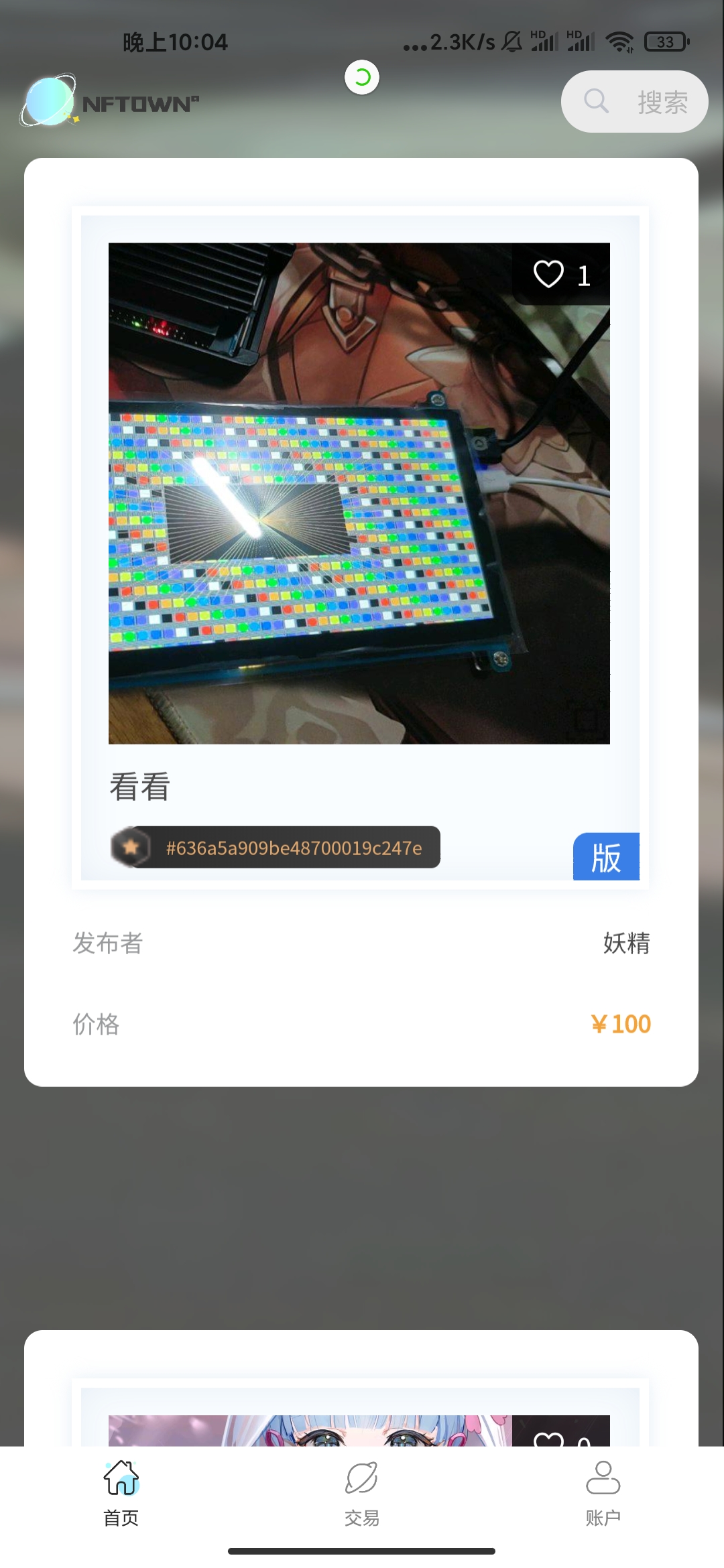


图4-3-3 下拉刷新

1. 单击图4-3-3中的“交易”，进入NFT上传以及交易界面。如图4-3-4所示。



图4-3-4 交易界面

1. 单击图4-3-4中的数字藏品，进入该数字藏品的详细界面。如图4-3-5所示。
2. 单击图4-3-5中的“立即购买”，即可交易。



图4-3-5 NFT详细界面

1. 单击图4-3-4中的“上传NFT”，进入上传界面，填写完信息后点击上传。如图4-3-6所示。



图4-3-6 上传NFT

1. 单击图4-3-4中的“出售NFT”，进入出售界面（图4-3-7），点击数字藏品可进行出售操作，如图4-3-8所示。

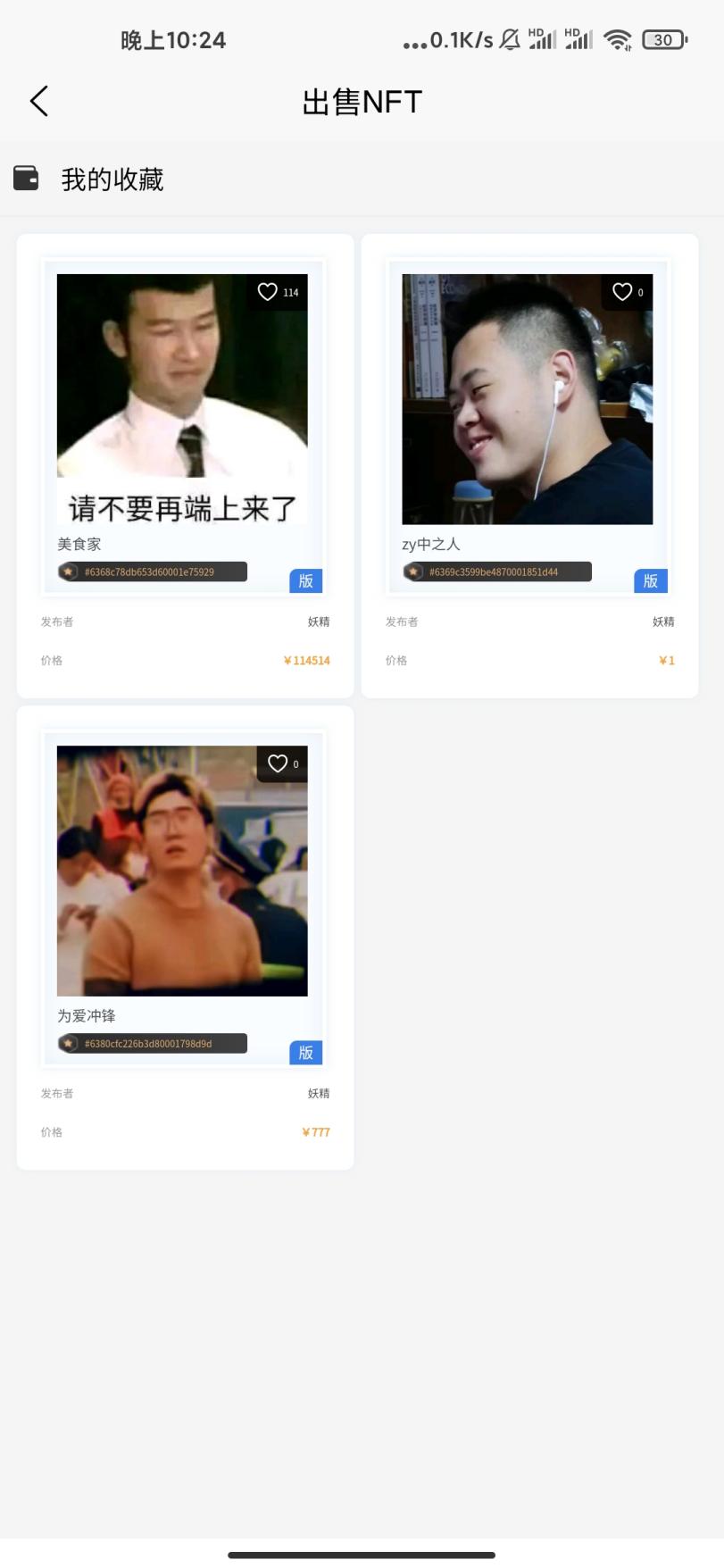


图4-3-7 选择库中数字藏品界面



图4-3-8 选择出售界面

1. 单击图4-3-3中的“账户”，进入用户信息界面。如图4-3-9所示。

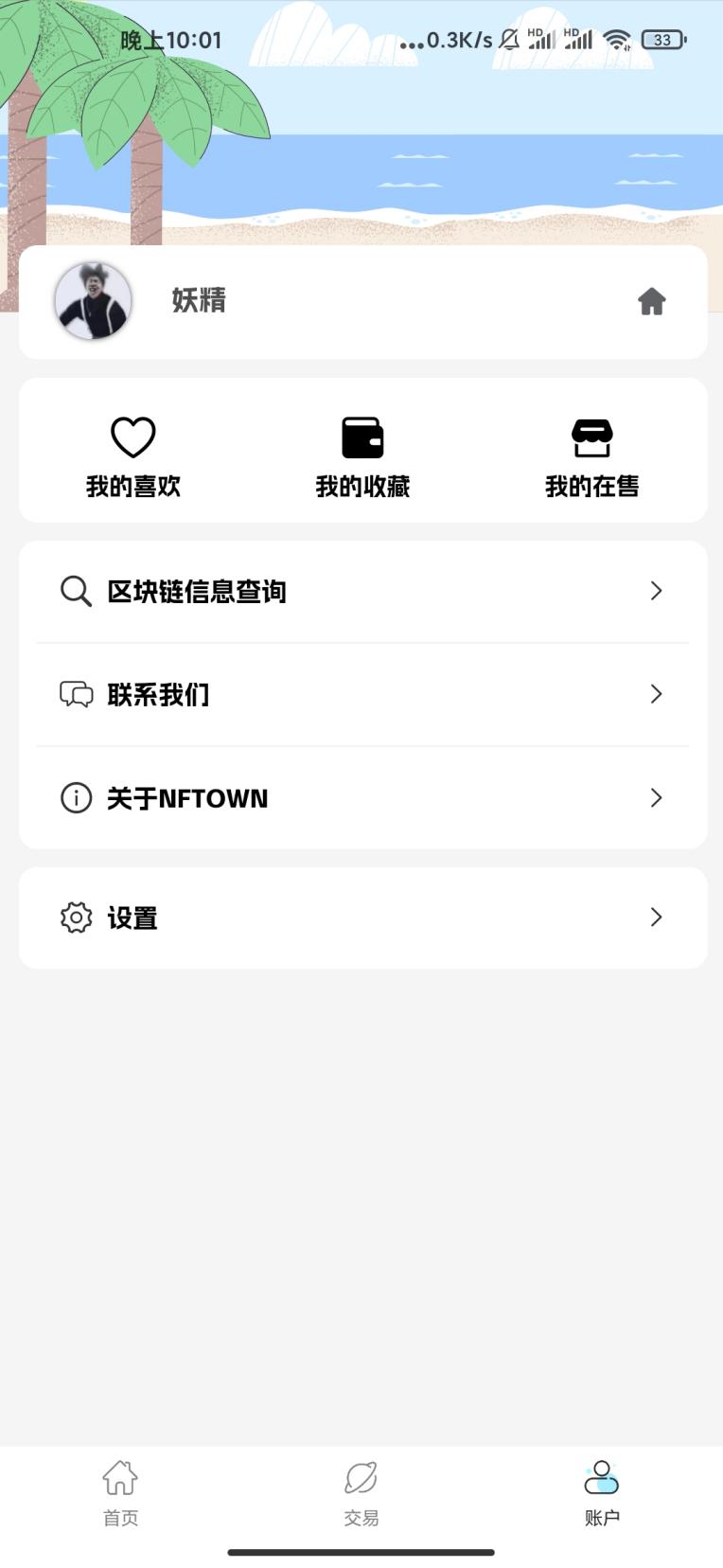


图4-3-9 用户信息界面

**4.3.3 功能测试**

功能测试包括首页刷新，数字藏品上链等模块的操作测试，测试结果如表4-3-2所示。

表4-3-2 功能测试表

|  |  |  |
| --- | --- | --- |
| 功能A | 首页刷新 | |
| 用例目的 | 测试上拉下滑的页面加载情况以及全部加载情况 | |
| 动作 | 期望输出 | 测试情况 |
| 上拉触底刷新 | 刷新出新的数字藏品 | Screenshot_2022-11-25-22-40-11-047_uni.UNI3FF32C6 |
| 下滑刷新 | 刷新出新的数字藏品 | Screenshot_2022-11-25-22-04-05-344_uni.UNI3FF32C6 |

**4.4 结果分析**

运行结果正常，符合我们的设计预期，同时交互正常，体验尚可。

# 5 总结

**5.1 用户反馈**

包括用户的使用记录，照片，视频等。

**5.1 全文总结**

全文总结：

1.通过多渠道向在校学生以及居民推荐本软件并发放使用一周的权限；

2.向使用了此软件的学生和居民分发调查问卷，收集他们的三个体验关键词，对此产品的看法和认为它所具有的优缺点，并收集了他们对此产品设计上的建议，以年龄，性别，职业等不同方向对收集到的反馈进行归类，制作分类图；

3.以后将搭建一个平台端启动的用户反馈机制，可以自动通过整理，归类，过滤等一系列操作，得到一些客观公正偏评价式的描述，另外，还通过一些沟通途径向用户侧反向反馈，用来增进用户的产品参与感，同时鼓励用户再次反馈，并通过邮件形式，短信通知形式，电话访问形式等沟通渠道对用户表示感谢，打造一个健康可持续的用户-平台沟通闭环。

# 6 体会

**关于前端设计的体会**：

有关的VUE进行前端开发的知识在上学期学过，但是并不算精通，画出来的页面也不是很好看。通过这次软件工程的开发经历，让我对前端编写代码更为熟练，前面认真画出的界面也比较好看，但是后期由于事情较多，比较忙，导致时间不太够，后续页面的开发就不如前面页面那么精致，算是比较遗憾。不过通过这次软件工程课程，我还是收获颇多，希望以后可以再接再厉。有关web的知识在学习python爬虫时担当涉猎但是不系统与完整，这次担任了小组的前端设计师之后，我系统的学习了一下web开发的有关知识，包括html，css，JavaScript三大web语言。也取得了不错的效果。就是开发阶段比较枯燥，遇到过不少困难与阻力，不同浏览器之间的显示效果经常会有不同，所以在协调调配上花费了不少精力，也在UI设计了有些许妥协，但是好在最后的成品还是不错的。赢得了周围用户的不错反馈。

**关于后端开发的体会**：

这是我第一次真正意义上的完成的软件后台开发。在前端页面的基础上完成数据库的编写维护以及页面间和页面内的跳转、交互逻辑。由于这两方面的知识在项目初期时是相当匮乏的，也缺乏相应的实战经验，于是我只能抓紧时间自学数据库以及web开发的相关知识，后面在此基础上又学习了uniCloud的开发，总算是在ddl前一周完成了后端的基本逻辑的实现。最后一点时间在和负责前端的同学交流后完成对软件的优化、润色以及打包。这次开发经历带给我良多收益，其一便是收获了软件开发的经验，让我对软件开发的认识，尤其是后端开发有了进一步的提升，深刻理解了前端和后端的交互流程，不再是那种朦朦胧胧的感觉了。其二是将所学的知识融会贯通，开发期间无数次的debug经历提高了我的coding能力，学习并实践的过程总是令人愉悦，这次经历极大地提高了我的自信心。遗憾也是有的，如果时间更加充裕的话可以将这个软件完成地更加优秀，大概这是对处女作寄予厚望的心理作祟。

**关于NFT区块链部分开发的体会**：

之前没有接触过这方面的开发，所以开始只能直接去学习，国内没有现成的教程可以学习，所以只能去油管上面找，英文的教程看起来很吃力，不过幸好还是肝完了，数字藏品只是人们在区块链之上衍生出来的一个方面，感觉只是一帮投机者想出来的另一个圈钱的方法，其本质.....感觉没有技术，但是区块链本身是很有意义以及值得学习的。通过这次学习帮我揭开了区块链以及大火的nft的神秘面纱，让我知道了什么是区块链，以及数字藏品的产生形式，交易形式。但是碍于我国的现有国情，写出来的东西也没办法用，因为根本没办法接入到公网，自己的私网也无法搭建，因为国内直接不能用区块链钱包，其实参考国内的数字藏品平台，基本上都是有名无实的，有一些甚至会教用户如何翻墙使用，甚至教用户如何翻墙申请metamask账号......所以说还是有很多遗憾，没有做的比较好。

**关于美工设计，作图的体会**：

有关web的知识在学习python爬虫时担当涉猎但是不系统与完整，这次担任了小组的前端设计师之后，我系统的学习了一下web开发的有关知识，包括html，css，JavaScript三大web语言。也取得了不错的效果。就是开发阶段比较枯燥，遇到过不少困难与阻力，不同浏览器之间的显示效果经常会有不同，所以在协调调配上花费了不少精力，也在UI设计了有些许妥协，但是好在最后的成品还是不错的。赢得了周围用户的不错反馈。