# 1.引言

随着人们生活物质的不断积累,财富的增加，闲置商品交易蓝海正在形成，这个蓝海就是每年二手商品交易数据的增加。以休闲鱼APP为代表的二手贸易最为突出。依靠支付宝芝麻信用分迅速成为综合性二手交易app领域的领头羊,独占用率达到64.1%。本文将使用目前主流技术(Node.js/wapApp/mysql)构建一个二手商品交易app,为所学知识注入实战经验。

# 2.二手商品交易APP基本情况介绍

移动互联网的不断发展,在各行各业的应用,极大改变人们的生活习惯,手机买卖商品早已深入人心,成为现代人的生活习惯,随着人们购买的物品越来越多,导致家里的闲置物品越来越多,如不尽快处理闲置物品,势必会影响人们进一步消费。

根据闲鱼报告中国闲置市场规模保守预计达4000亿元, 90后是闲鱼使用率最频繁的团体，他们而且比所有用户的平均互动率高20％。 90后用户购买平均花费53分钟，与卖家交流次数高达15次.据此可以见二手商品交易市场的活力惊人,市场容量在不久的将来是巨大的,有理由相信,二手商品交易不可或缺,二手商品交易APP的研发很有必要。

# 3.二手商品交易APP编程技术介绍与分析

## 3.1 APP编程技术介绍

二手商品交易APP,需要有服务器语言(NODE)与(MYSQL)数据库对用户在客户端输入的内容进行处理展示,所以APP需要精心制作,使之页面美观,处理好与用户的交互,所有这些都离不开这些编程技术,在技术飞速发展的今天, 混合应用程序让java与网页编程走到了一起,这使得APP开发速度能与网页编程一样快速。

## 3.2 APP编程技术分析

混合应用程序指的是原始应用程序与web配合,原始应用它具有“良好的用户交互和良好的用户体验,Web应用具有快速开发和跨多平台, 混合应用程序 就是Web应用程序与原始应用程序之间的应用程序,目前用于网页和程序语言的混合支持应用程序有以下几种类型：

1,在混合开发中，多视图混合在本机视图和Web视图独立视图之间交替。 2012年的常见混合应用程序是本机视图和WebView交替的场景。 这种应用混合逻辑相对简单。 也就是说，如有必要，WebView作为单独的视图（活动）运行，相关的显示操作在WebView中完成。 移动应用程序本身通常是本机应用程序，而Web技术只是一种补充。 开发困难与本机应用程序基本相同。

2,单个视图混合在同一视图中，包括本机视图和Web视图。 它是一种覆盖（堆叠）的关系。 这种混合应用程序具有较高的开发成本，并且更难以开发，但体验更好。 例如，以百度search为代表的单视图混合移动应用可以实现足够的灵活性和更好的用户体验。

3,基于Web的移动应用程序的主要部分是Web视图，主要使用Web语言编写并分布在本机混合应用程序开发类型中。虽然这种类型的移动应用程序体验相对有缺陷，但总体开发难度可以大大降低，并且可以从根本上实现多个平台。基于Web的移动应用程序的用户体验主要取决于底层中间件的互操作性和跨平台。外国AppMobi，PhoneGap和WeX5，AppCan和Rexsee都是基于Web的移动应用中间件。 Rexsee不支持跨平台开发。除了appMobi和PhoneGap的基本功能外，混合型还通过扩展插件（插件）进行分发。除了插件机制外，AppCan还提供了大量的单视图混合界面，可以改善和补充与原始应用程序环境接近的基于Web的混合应用程序的不愉快体验。基于PhoneGap和Bootstrap等关键技术，WeX5可以优化性能，提供性能以及调用本机应用程序。 WeX5开发的应用程序是完全外围的，可以直接在多个前端环境中运行而无需修改。从分析可以看出，混合应用程序中的Web主机类型可以是最佳类型的混合应用程序，只要它可以解决用户体验不佳的问题。

4,多主体类型 - 灵活是这种支持基于Web的应用程序并支持基于本机的应用程序和混合开发模型的新开发模型。 例如:kerkee框架具有跨平台特性，用户体验，高性能，良好的可伸缩性，灵活性，易维护性，标准化，调试环境和跨域问题。 用户体验与原始应用程序相当。 在功能方面，开发人员可以随意扩展界面。

### 3.2.1 WapApp

wapApp是DCloud公司一个增强M站的解决方案，用于快速将现有的M站（也称为wap移动站，不同于PC网站）发布到App中，并通过DCloud wapApp框架进行简单配置和所需的编程。 可以完成改进的M站体验以实现本机应用程序的功能体验，然后作为原始安装包或流应用程序发布

wapApp特点:

为用户体验提供界面渲染速度和动画效果的原始渲染功能,提供丰富的本机系统功能（定位，共享，支付，推送等）以助力M站实现更多的功能

配置页面规则和json规则，低工作负载和低学习成本M站仅需稍作修改，改造成本极低

增强部分与前一个M工作站分开，M工作站进一步增强了业务逻辑，自动生成的应用程序包含更新的业务逻辑。

使用wap2app，开发人员可以在站M之前重用所有业务逻辑，只需要进行必要的强化，并且可以重用所有业务逻辑，例如注册，购物，搜索等。

开发人员需要开展的工作分为：

1 wapApp客户端配置工作程序

2对原M站进行体验性改造

**注意**这2块是分开解耦的。

在HBuilder 8.8.3中，新的wap2app项目将自动为项目生成所需的模板文件。 体验性改造工作主要在这里进行。。

当您选择将新的wapApp项目打包为原始应用程序或HBuilder中的应用程序时，wapApp项目下的此增强配置和编程文件将存储在原始App包中，并直接安装在手机的本地存储区域中。 安装。运行。这可以理解为C / S模式的客户端部分.Station M仍处于B / S模式，而不是将M站资源打包到应用程序包中，M站页面是从服务器在线加载的。

手机端实际运行时，wepApp的框架会自动把本地的强化配置和在线的M站融合在一起，在用户眼里是一个 App 的感觉。

混合APP有多种技术手段可以选择,而各种技术可谓各有千秋 ,在低端的Android手机上，DIV动画经常被卡住（特别是在图形列表的情况下）; DCloud通过Dual WebView解决了这个DIV拖动的流畅性问题; 完整的Web视图（子Web视图），使用本机动画的回弹动画，Web应用程序可以是单个视图 也可以使用多view,这给开发带来了灵活性,使得用户体验更好。

### 3.2.2 Vue

Vue框架诞生于2014年，其作者为中国人——尤雨溪。Vue用于构建交互式的Web界面的库，是一个构建数据驱动的Web界面渐进式框架，该框架遵循CMD规范，并且提供的设计模式为MVVM模式（Model->View->View-Model）和一个可组合的组合型组件系统，具有简单的、灵活的API(接口)。该框架继承了React的虚拟DOM技术和Angular的双向数据绑定技术，是一款较新的功能性框架。

特性:

1. 虚拟DOM（Virtual DOM），顾名思义，从字面上理解就是虚构的DOM树，当我们用传统的原生API或者jQuery去操作DOM时，浏览器会从构建DOM树开始从头到尾执行一遍流程。即使计算机硬件一直在更新迭代，但是操作真实DOM的代价仍旧很昂贵，真实的DOM节点，哪怕是一个最简单的div也包含很多属性，所以频繁的操作，会导致页面卡顿，影响用户的体验。为了解决这个浏览器性能问题，虚拟DOM（Virtual DOM）就被设计出来了，其核心算法是Diff算法。它会将一次操作过程中对真实DOM所有更新的diff内容保存到本地的一个js对象中，最终将这个js对象一次性attach到DOM树上，通知浏览器去执行绘制工作，避免了大量的无谓的计算量。
2. 双向数据绑定，在讲双向数据绑定前，我们要想说下单向数据绑定，单向数据绑定，就是把Model绑定到View上，当我们用JavaScript代码更新Model时，View就会自动更新了（Model-->View）。那么双向数据绑定就是，用户更新了View，Model的数据也会自动被更新（Model<-->View）。什么情况下用户可以更新View呢？举个最直接的例子，填写表单，当用户填写表单时，View的状态就被更新了，如果此时MVVM框架可以自动更新Model的状态，那么就相当于我们把Model和View做了双向数据绑定。其原理是我们要对input进行value 的属性绑定（v-bind：value=”...”）将Model中的变量绑定到View上（Model->View）以及当用户对input进行操作时，进行事件监听（v-on：input=”...”）将View上的更新传回Model中（View->Model）从而实现双向数据绑定，在Vue中，以上操作过于繁琐，便提供了v-model直接实现双向数据绑定的效果。

### 3.2.3 Node

Node是一个开发平台，可将JavaScript转换为服务器端语言的包裹语言，如PHP，Python，Perl，Ruby等。它于2009年5月由Ryan Dahl发布，其基本是Chrome V8引擎。

Node针对几种特殊用例进行了优化，并提供了一种替代API，使V8在非浏览器环境中更好地工作。 V8引擎运行Javascript非常快。

对与新技术，多问几个为什么总是好的。既然 PHP、Python、Java 都可以用来进行后端开发，为什么还要用 Node.js？

总的来说，Node.js 适合以下场景

1并发

与客户端不同，服务端开发者非常关心的一项数据是并发数，也就是这台服务器最多能支持多少个客户端的并发请求。早年的 C10K 问题就是讨论如何利用单台服务器支持 10K 并发数。当然随着软硬件性能的提高，目前 C10K 已经不再是问题，我们开始尝试解决 C10M 问题，即单台服务器如何处理百万级的并发。

从C10K开始，后端使用Apache服务器，但每次网络请求到达并在该子进程中运行PHP脚本时，它都会通过查询子进程来工作。 运行脚本后，结果将发送回客户端。

这使得不同进程可以不相互干扰，即使问题不影响整个服务器，但缺点也很明显。 过程是一个相对沉重的概念。，拥有自己的堆和栈，占用内存较多，一台服务器能运行的进程数量有上限，大约也就在几千左右。

Apache后来使用了FastCGI，它本质上是一个具有减少的进程创建开销的进程池，但没有有效地增加并发执行的数量。

多线程编程会带来各种麻烦，这一点想必程序员们都深有体会。如果不使用线程，还有两种解决方案，分别是使用协程(coroutine)和非阻塞 I/O。协程比线程轻得多，可以在同一个线程中执行多个协程，并且可以由程序员自己调度。这种技术在Go语言中被广泛使用。 而不是阻止I / O，Node.js使用非阻塞 I/O来处理高度并发的场景。

2 node的非阻塞 I/O

这里有两种类型的I / O.网络I / O和文件I / O.这些实际上非常相似。 I / O分为两个步骤首先，将文件（网络）的内容复制到缓冲区中此缓冲区位于操作系统专用存储区中。之后，缓冲区的内容被复制到用户程序的存储区。

为了阻止I / O，两个步骤都启动读取请求，准备缓冲区并阻止用户进程。非阻塞I / O实际轮询内核，缓冲区已准备就绪，否则继续执行其他操作。缓冲区就绪后，缓冲区的内容将复制到用户进程并实际被阻止。

I / O复用技术是指在单个线程中处理多个网络I / O，而选择和epoll通常被称为用于轮询所有套接字的函数。例如，Apache使用前者，Nginx和Node.js使用后者，但后者更有效。由于 I/O 多路复用实际上还是单线程的轮询，因此它也是一种非阻塞 I/O 的方案。

### 3.2.4 MySql

瑞典MySQL是一个关系数据库，关系数据库以行和列的形式存储数据，以便用户可以轻松地理解它。 这组行和列称为表，一组表构成数据库。 用户使用查询来检索数据库中的数据。 Query是一个SELECT语句，用于指定数据库中的行和列。

# 4.二手商品交易APP设计与实现

## 4.1 系统功能概述

本次设计的APP是实现一个二手商品交易APP,我们可以将使用该系统的人分为两类,一个是发布二手物品的,一个是购买二手物品的.APP功能划分如图4.1



图4.1

二手商品交易系统

商品交易系统是为用户提供二手商品销售和购买二手商品的系统,其中留言模块是为了使用户之间可以方便的交流.

用户信息资料管理

用户信息资料管理系统是APP的基础模块,app用户以此来维护自己的信息资料.

## 4.2 流程图

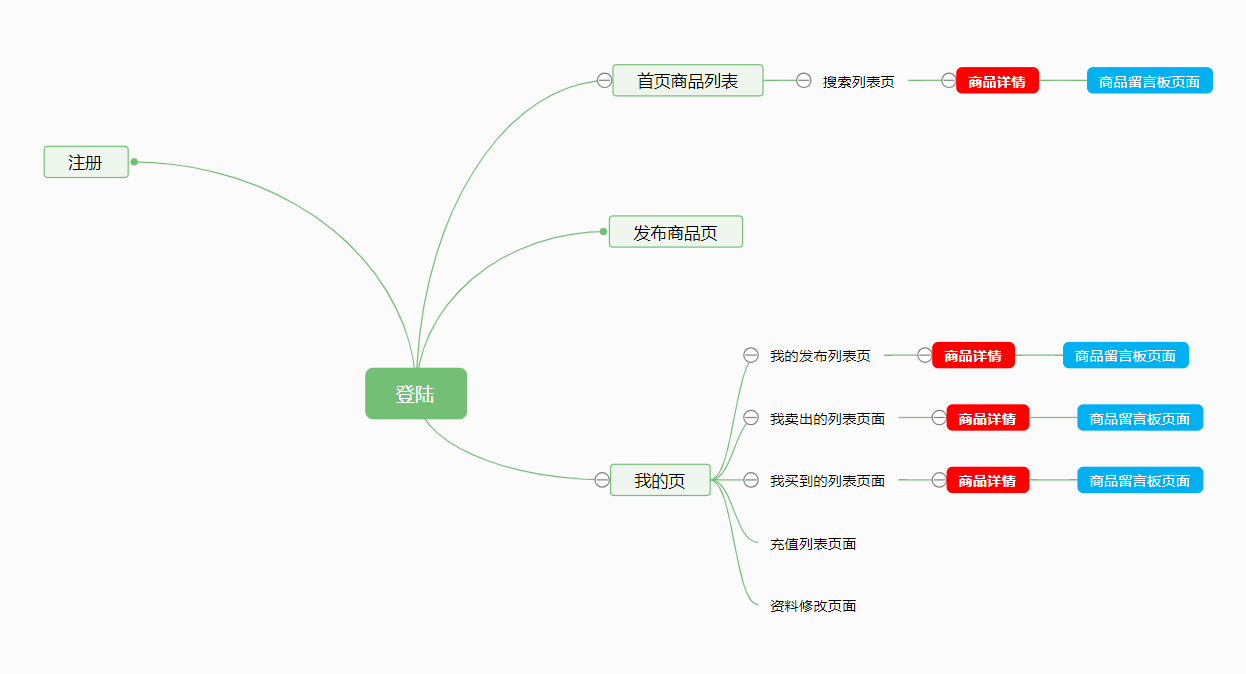


图4.2

## 4.3 数据库结构及实现

此次数据库总共3张表,表和表都有联系,表用途详情如下:

|  |  |
| --- | --- |
| 表名 | 说明 |
| Users | 保存用户账户密码,收货地址,身份证,名称,余额,头像信息 |
| leave\_words | 保存商品在售卖期间用户的留言消息 |
| commodity\_lists | 保存商品信息详情 |

表4.1

### 4.3.1 用户资料表Users

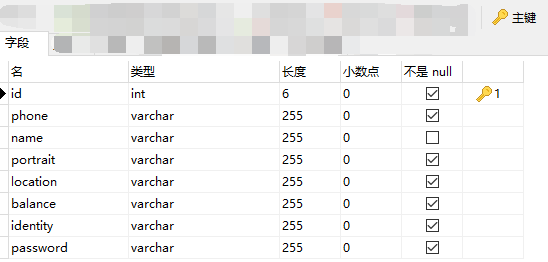
用户资料表users保存着用户的账户密码,收货地址,身份证,名称,余额,头像信息,结构如下图:

图 4.3

|  |  |
| --- | --- |
| 字段 | 说明 |
| Id | 数据唯一标识 |
| Phone | 注册手机号码 |
| Name | 昵称 |
| Portrait | 头像url |
| Location | 收货地址 |
| Balance | 余额 |
| Identity | 身份证 |
| password | 密码md5 |

表 4.2

### 4.3.2 商品资料表commodity\_lists

商品信息表保存商品的信息,交易状态,购买者和卖出者信息资料,结构如下图:

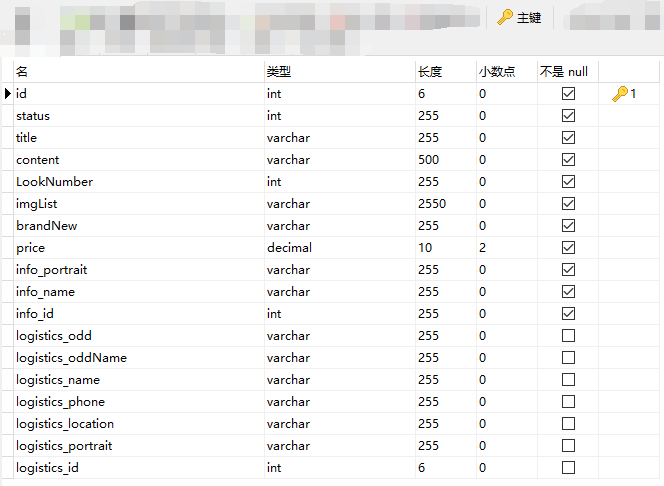


图4.4

|  |  |
| --- | --- |
| 字段 | 说明 |
| Id | 数据唯一标识 |
| status | 商品状态  商品发布  支付成功  已发货  取消发布  完成交易  异常订单 |
| Title | 商品标题 |
| Content | 商品内容介绍 |
| lookNumber | 浏览商品次数 |
| imgList | 商品图像详情[] |
| brandNew | 0不是全新1全新商品 |
| Price | 价格 |
| Info\_portrait | 发布者头像url |
| Info\_name | 发布者名字 |
| Info\_id | 发布者数据唯一标识 |
| Logistics\_odd | 快递号码 |
| Logistics\_oddName | 快递名称 |
| Logistics\_name | 购买者名字 |
| Logistics\_phone | 购买者手机 |
| Logistics\_location | 购买者收货地址 |
| Logistics\_portrait | 购买者头像 |
| Logistics\_id | 购买者数据唯一标识 |

表4.3

### 4.3.3 留言消息表leave\_words

用户的留言消息leave\_words表,保存用户对商品留言,结构如图下:

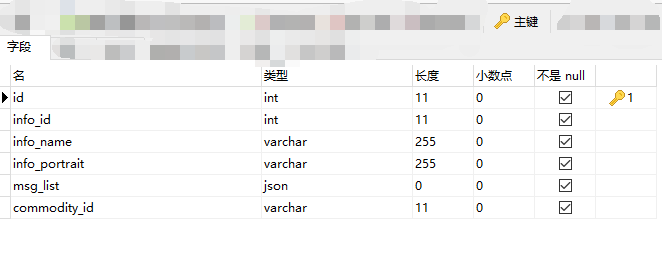


图4.5

|  |  |
| --- | --- |
| 字段 | 说明 |
| id | 数据唯一标识 |
| Info\_id | 用户数据id |
| Info\_name | 用户昵称 |
| Info\_portrait | 用户头像链接 |
| Msg\_list | 用户的留言数组列表[{msg:’’,time:’123123’}],msg是每次留言,time是留言时间戳 |
| Commodity\_id | 商品数据id |

表4.4

## 4.4 文件目录结构

### 4.4.1 WapApp http://dcloud-img.oss-cn-hangzhou.aliyuncs.com/guide/wap2app/wap2app-guide-02.png

图4.6

|  |  |
| --- | --- |
| 文件名/文件夹 | 说明 |
| Wapapp-example | 改文件夹用来存放图标引导图等资源 |
| sitemap.json | Wapapp json配置文件wapapp运行的大多数规则都是使用此文件完成的 |
| app.js | 为弥补sitemap的json的格式限制，提供的基于JavaScript基于JavaScript的配置扩展方案，以补偿sitemap的 json格式限制. |
| manifest.json | 工程打包配置，比如icon、splash图第三方SDK等 |
| client\_index.html | html静态页面，若首页有多选项卡，需在该文件中编程 |
| %APPID%.append.css | 本示例中 APPID 为 “\_W2A\_m.example.com” ，系统自动创建了一个 \_W2A\_m.example.com.append.css 文件；该 css 文件负责首页的样式 |

表4.5

### 4.4.2 Node

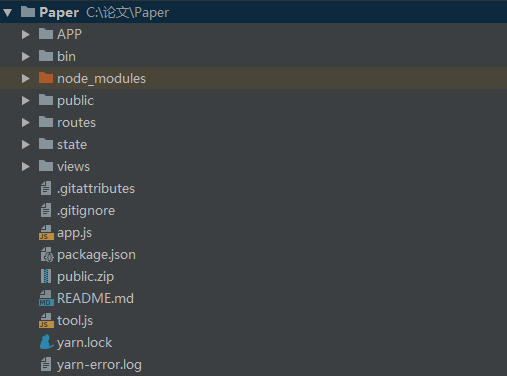


图 4.7

|  |  |
| --- | --- |
| 文件夹 | 用途 |
| APP | Vue项目打包后存放目录 |
| Bin | 项目入口文件 |
| Node\_modules | 项目依赖 |
| Public | 静态资源目录 |
| Router | 路由文件 |
| state | 状态管理 |
| Views | 视图文件 |

表 4.6

### 4.4.3 Vue

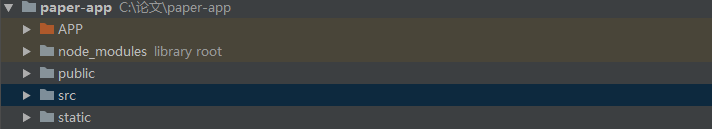


图4.8

|  |  |
| --- | --- |
| 文件夹名称 | 用途 |
| APP | 生产目录 |
| Node\_modules | 项目依赖 |
| Public/ Static | 静态资源目录 |
| Src | 源代码文件 |

表4.7

## 4.5 页面实现结果



图 4.9



图 4.10



图 4.11



图4.12



图 4.13

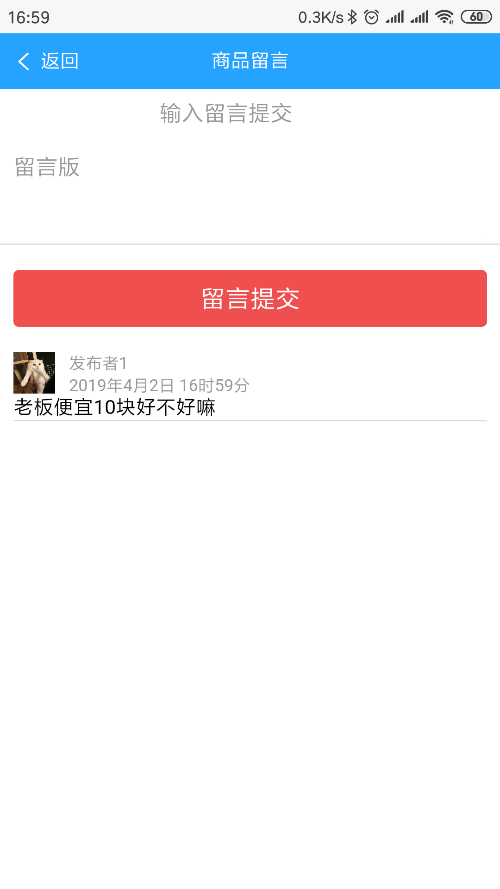


图4.14



图4.15



图 4.16



图 4.17

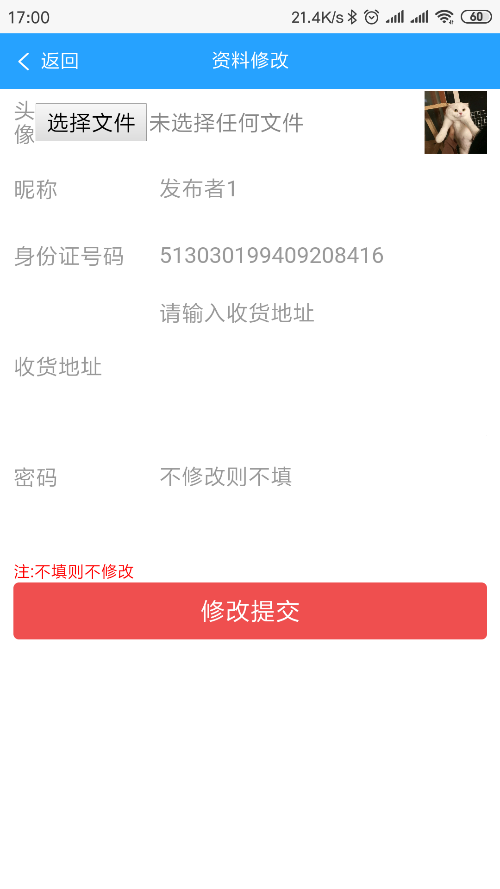


图4.18



图4.19

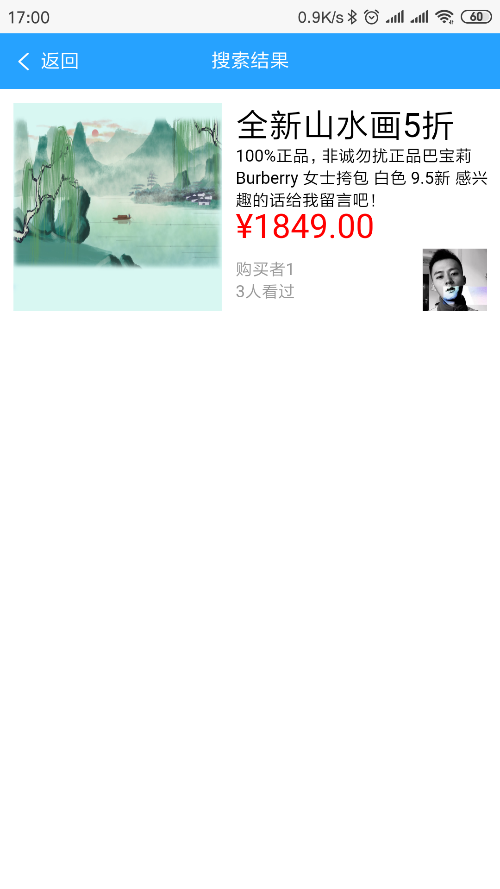


图4.20

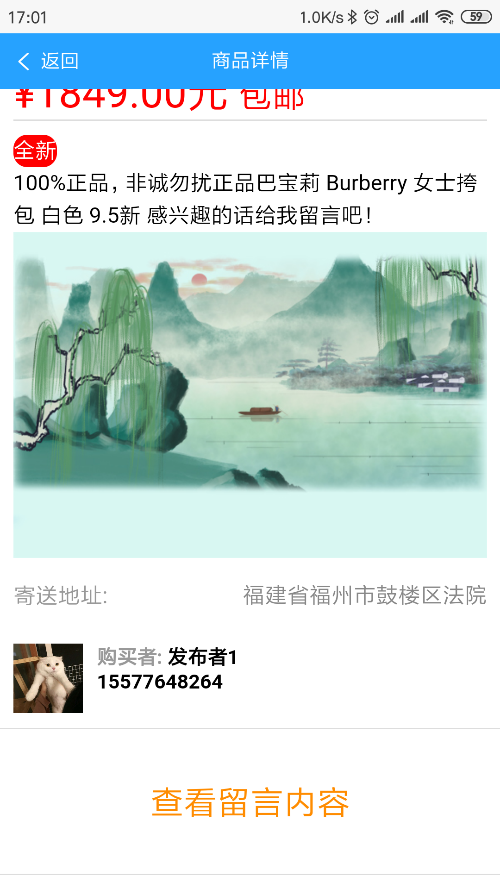


图4.21



图 4.22

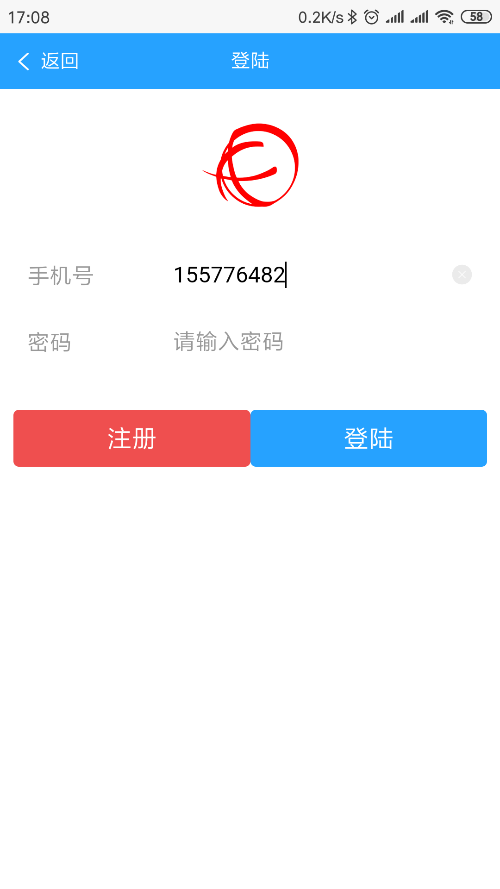


图4.23

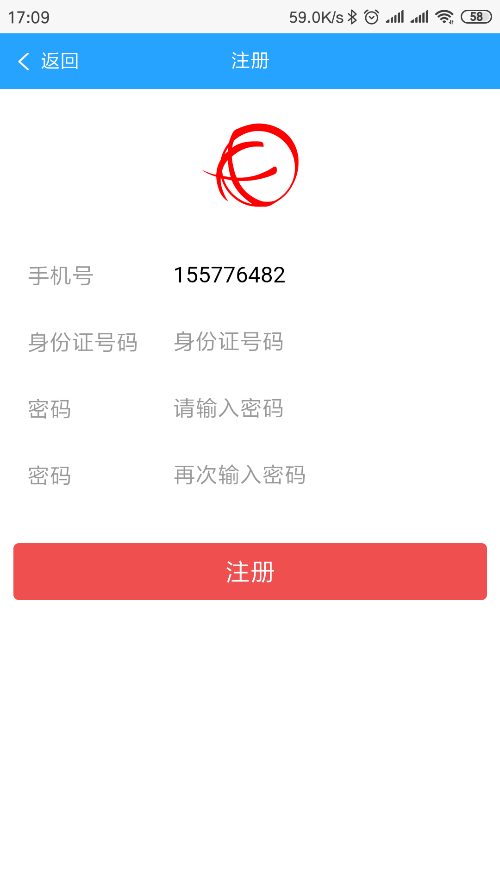


图 4.24



图 4.25

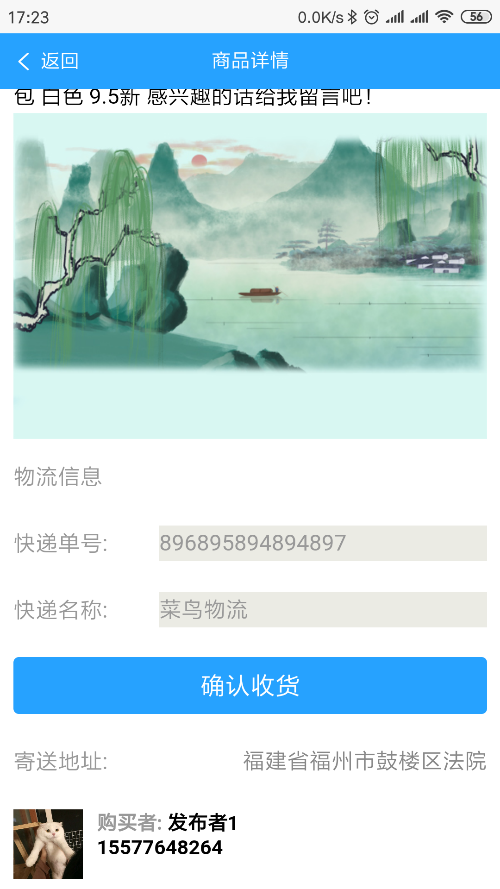


图 4.26

# 5设计开发种所遇到的困难和解决方案

设计和解决方案所面临的困难在设计和创作过程中是不可避免的，设计本身的整个过程不可避免地伴随着各种困难。完美并不令人满意，或多或少是生活问题，好坏，坏，自我毁灭，中途，不再关心。在这个APP设计中，起初它被批评为没有坚持和活跃，并因没有得到足够的重视而被批评，并且无法尽力进行这种设计。例如，虽然在这种设计中很难，因为图形文本的识别在颜色组合和实际应用的组合中是显而易见的或不显眼的，但是可以使用主题来展示创作设计的作品。我可以。和谐与和谐的感觉仍然非常困难，在整个主题颜色模式统一的前提下，有必要确保它不单调（例如，主页分类最初是一个平滑的圆圈）将表达）。但是，当与下一列组合时，它将非常不一致，因此在更改计数后，将使用平滑的圆角矩形，并且对整体主题完整性的影响将更小。我相信我可以解决任何问题，没有问题我无法解决。与同学交谈并寻求帮助、上网查阅有相关问题的资料,通过比较各种解决方案,最后终于得出一个最完美的解决方法。在字体的上面轻轻的覆盖一层半透明的黑色矩形框,这样,既不失其整体色调的统一性,又表现出了想要达到的视觉效果。这样类似的问题还有很多,但是有问题就会有解决,不管是多么困难的事,只要有心人愿意去做,愿意去解决,就会有解决的一天,事情也会有圆满的时候,求知的精神才是我们所需要也更要努力的。

# 6总结

此次设计感谢重庆大学为我提供一个锻炼自我、展示自我的平台,感谢对我的教育与培养,希望重庆大学未来的发展会越来越好。感谢我挚爱并且尊敬的老师,是你们任劳任怨的付出才成就如今我的成长。在这里,请允许我向在我大学的学习与生活中起到具有不可代替性的老师,表示深深的致敬,语言上的感激总是略显苍白无力,一声感谢也太过淡薄,内心的感激之情根本无法言说, 我的老师努力指导，有时甚至忙着为我们回答问题。 深夜，有时候很晚，这些精神深深打动了我们，并敦促我努力学习进步，成长到我想要的高度。 最后，我要衷心感谢我的同学们，谢谢你们在风暴中和我在一起,这份友谊和感情我会铭感于心,再次谢谢我亲爱的重庆大学!

**参考文献：**

[1]邵凌君. 少儿频道移动应用的设计与实现[D]. 华南理工大学, 2014.

[2]孙夏. MySQL数据库系统在教辅系统中的应用[D]. 西安科技大学, 2017.

[3]姚光艳. 去哪儿网酒店供应链系统的设计及实现[D]. 中国科学院大学(中国科学院工程管理与信息技术学院), 2017.

[4]刘杰. 基于扩展MVC的物品共享系统的设计与实现[D]. 北京邮电大学, 2017.

[5]王安卓. 基于iOS的O2O上门按摩服务应用软件的设计与实现[D]. 北京邮电大学, 2017.

[6]吴健. 基于并发控制机制的Web系统的开发技术研究[D]. 云南大学, 2015.

[7]贺祥. 基于Node.js平台的天线控制软件开发实例解析[D]. 国家新闻出版广电总局二九二台, 2016.

[8]吕昕. 基于Web的富客户端跨平台移动应用开发技术研究[D]. 云南大学, 2015.