

浙江大學城市學院
ZHEJIANG UNIVERSITY CITY COLLEGE

毕业设计（论文）



题 目 易换小程序设计与实现

姓 名 孙祎捷

学 号 31501080

专业班级 计算机 1501 班

所在学院 计算学院

指导教师（职称） 霍梅梅（副教授）

二〇一九 年 五 月 十八 日

易换小程序设计与实现

【摘要】 跳蚤市场在全国各高校内屡见不鲜，但由于时间和空间上的局限性，物品的流通率并不是很高。随着电子商务的迅速发展，跳蚤市场已成为电商的一个交易内容。作为一种新兴的电子产物，微信成为了学生生活中不可或缺的工具，因此以微信平台为媒介建立小范围的二手交易平台更容易为人接受。

由此，针对传统校园二手商品市场交易方式存在的缺陷，设计了一个基于微信小程序的校园二手商品交易平台。平台的开发在微信开发者工具里进行，使用了微信小程序开发框架和 JavaScript 语言来搭建界面并实现功能；后端直接采用 Bmob 后端云提供的云数据库服务来搭建数据库。程序分为商品管理、用户信息管理、消息管理三个模块，主要功能有商品发布、修改、搜索、查询、定位、购买等。程序的界面设计简洁友好，利于用户使用。

【关键词】 校园，二手商品，云数据库，微信小程序

Design and Implementation of YiHuan WeChat Mini-Program

【Abstract】 The flea market is quite normal in universities across the country, but due to time and space limitations, the circulation rate of goods is not high. With the rapid development of e-commerce, the flea market has become a transaction content of e-commerce. As an emerging electronic product, WeChat has become an indispensable tool in students' life. Therefore, it is easier to accept a small-scale second-hand trading platform based on the WeChat platform.

In view of the shortcomings of traditional campus second-hand commodity market transactions, a campus second-hand item trading platform based on WeChat mini-program was designed. The development of the platform is carried out in the WeChat developer tool, using the WeChat applet development framework and JavaScript language to build the interface and implement the function; the back-end directly uses the cloud database service provided by the Bmob backend cloud to build the database. The program is divided into three modules: commodity management, user information management, and message management. The main functions include product release, modification, search, query, location, and purchase. The interface design of the program is simple and friendly, which is beneficial to users.

【 Key Words 】 Campus , second-hand goods, cloud database , WeChat Mini-Program

目录

| | |
|--------------------------|----|
| 第 1 章 绪论..... | 1 |
| 1.1 研究的背景与意义..... | 1 |
| 1.1.1 研究的背景..... | 1 |
| 1.1.2 程序开发的意义..... | 2 |
| 1.2 研究的主要内容..... | 3 |
| 第 2 章 程序的开发环境及技术介绍..... | 4 |
| 2.1 程序开发环境介绍..... | 4 |
| 2.1.1 Bmob 后端云-云数据库..... | 4 |
| 2.1.2 微信开发者工具..... | 4 |
| 2.2 程序中涉及的开发技术介绍..... | 5 |
| 第 3 章 程序总体设计..... | 6 |
| 3.1 程序的整体框架和模块功能设计..... | 6 |
| 3.1.1 用户管理模块..... | 6 |
| 3.1.2 商品管理模块..... | 7 |
| 3.1.3 消息管理模块..... | 7 |
| 3.2 系统的流程设计..... | 8 |
| 3.3 系统的数据库设计..... | 9 |
| 第 4 章 程序功能详细设计与开发..... | 14 |
| 4.1 用户管理模块详细设计..... | 14 |
| 4.1.1 用户一键登录功能..... | 14 |
| 4.1.2 用户模块其它功能..... | 16 |
| 4.2 商品管理模块详细设计..... | 17 |
| 4.2.1 商品首页展示..... | 17 |
| 4.2.2 商品发布、修改功能..... | 19 |
| 4.2.3 商品搜索..... | 20 |
| 4.2.4 商品详情页..... | 22 |
| 4.2.5 地图模式..... | 23 |
| 4.2.6 商品收藏..... | 25 |
| 4.2.7 商品订购..... | 26 |
| 4.3 消息管理模块详细设计..... | 27 |
| 第 5 章 系统测试..... | 29 |
| 5.1 系统测试原则..... | 29 |

| | |
|------------------|----|
| 5.2 主要的测试用例..... | 29 |
| 结论..... | 31 |
| 参考文献..... | 32 |
| 致谢..... | 34 |

图目录

| | |
|-------------------------------|----|
| 图 3.1 系统功能结构图..... | 6 |
| 图 3.2 系统主要流程图..... | 9 |
| 图 3.3 数据库表汇总图..... | 10 |
| 图 3.4 反馈表 feedInfo 字段实例图..... | 12 |
| 图 4.1 授权界面效果图..... | 16 |
| 图 4.2 个人信息页面图..... | 16 |
| 图 4.3 用户收藏商品及订单列表..... | 17 |
| 图 4.5 首页布局图..... | 18 |
| 图 4.6 导航栏布局图..... | 18 |
| 图 4.7 商品发布界面图..... | 20 |
| 图 4.8 商品搜索界面..... | 22 |
| 图 4.9 商品详情界面图..... | 23 |
| 图 4.10 地图模式图..... | 24 |
| 图 4.11 地图选择模式图..... | 24 |
| 图 4.12 消息通知示例图..... | 26 |
| 图 4.13 我的消息列表图..... | 28 |

表目录

| | |
|----------------------|----|
| 表 3.1 用户表..... | 10 |
| 表 3.2 发布商品表..... | 11 |
| 表 3.3 留言表..... | 11 |
| 表 3.4 反馈表..... | 12 |
| 表 3.5 消息表..... | 12 |
| 表 3.6 订单表..... | 13 |
| 表 5.1 测试用例及测试结果..... | 30 |

第 1 章 绪论

1.1 研究的背景与意义

随着现代信息技术的迅猛发展和普及，基于互联网的商品交易现象已经非常普遍，基础环境的成熟与需求欲望的增长推动着电子商务与商务网站建设的不断发展[1]。

1.1.1 研究的背景

由中国电子商务研究中心发布的《2018 年（上）中国 B2B 电子商务市场数据监测报告》显示，该年上半年中国电子商务交易规模已达 11.2 万亿元，同比增长 14.2%。在经济发展总体放缓的大背景下，国家围绕着供给侧改革及“互联网+”推出了系列相关政策，成为了 B2B 电商发展的契机[2]。另据 2015 年由全球著名市场调研公司尼尔森公司发布的中国电子商务行业发展“杭州指数”白皮书显示，18-35 岁年龄段的网购群体占据了总网购人数的 69%，本科及以上学历的网购群体比例达到了 79% [3]。这说明在网购人群中，接受过高等教育的青年群体是网购的主力群体。而在这一主体中有相当大一部分是目前的在校大学生。

对于现在的很多在校大学生来说，其购买能力相较之前已经有了很大的提高，能够比较随心地购买自己喜欢的商品。大学生作为特殊的群体，消费欲望强烈，追求时尚潮流，不可避免地产生众多闲置物品[4]。随着各类电商平台的迅速发展，大学生在购物过程中也出现了较多盲目消费、冲动消费的现象。购买的部分商品使用不久后便被闲置；但商品并没有失去它本身的价值，久置后基本都会以丢弃的方式被处理，造成较大的浪费。对于一些经济水平有限的学生来说，也能够以较低廉的价格买到心仪的商品。随着可持续发展和绿色发展观念的不断深入，大学生节约资源的意识也在不断提高，环保观念的流行使大学生购买二手商品成为越来越常态化的事情。

1.1.2 程序开发的意义

以校园跳蚤市场活动等为代表的二手商品交易方式，不仅有助于大学生培养勤俭节约的意识、发扬艰苦奋斗的优良传统，还有助于引导大学生理性消费，帮助大学生树立自主创业的观念，促进校园文化的和谐发展，因此在校园内很受推崇，是校园文化的重要组成部分。跳蚤市场作为高校内的一种经济现象，具有其他市场所没有的特点，如市场主体稳定性、商品廉价性、商品比例变动性以及“地摊式”经营主导性[5]。传统的跳蚤市场，一般一年进行一次，主要在毕业季展开，其交易商品种类繁多，但由于未形成有效规模，持续时间短暂且管理不规范，从而降低了获利空间[6]；购买者在短期内无法快速定位到自己需要的商品，很大程度上降低了交易的效率。而“互联网+”形式的二手商品交易平台，能够及时有序地展示较多的二手商品出售信息，让交易不受场地限制且种类更加丰富多样[7]，方便购买者查找，从而保证二手商品信息的流通性，使二手商品交易率和资源利用效率都大幅提升。

随着产品周期变短和人们购买能力增强，社会上闲置物品逐渐增加，人们对在线二手商品交易的需求会逐渐增大[8]。如今，市面上也有许多二手商品交易网站。如 58 同城、爱回收（电子设备回收平台）、安居客(二手房房产软件)、人人车（二手车交易平台）等。比较完善的二手商品交易平台主要是以交易大件物品（如汽车、家用电器等）或价值较为高昂的商品（如名牌包包、珠宝、电子产品等）为主。由于校园二手商品交易对象以学生为主，使得交易商品种类存在着明显的“学生”特点，具体表现在：书籍、学习资料在总交易额中占了较大的比重，所售商品主要以实用化的小商品为主[9]。但针对校园二手商品交易设计的平台还不多见。目前在各大校园内并没有很好的符合校园特点的网上交易平台，学生大多还是借助一些网络聊天交友平台发布想要转让的二手商品[10]。这种方式较传统的跳蚤市场更具有灵活性，但还是远远不够满足需求。

从第三方移动数据服务平台 TalkingData 于 2018 年推出的一份微信小程序调查报告来看，自 2017 年 1 月 9 日微信小程序正式上线后，微信不断地开放平台能力，极有耐心地在为小程序逐步加温。在争议与关注中，小程序在 2018 年年初做到了日活 1.7 亿、月活 4.3 亿，参与开发者 100 多万[11]。小程序无需安装、“用

完即走”的理念能够很好地满足用户需求且节省手机内存。微信现在作为移动终端的典型代表，使用人数在最近的几年内急剧增长，势头非常猛[12]。相较于传统网站和 App 形式，以微信小程序为载体构建的二手商品交易平台具有很大的潜在市场覆盖范围，可以在很大程度上克服时间和空间上的限制[13]，从而提高二手商品交易的效率。

1.2 研究的主要内容

本课题拟设计一个基于微信小程序的校园二手物品交易平台。应用力求简洁、美观、实用，让使用者可以快速上手。用户使用微信账号直接授权一键登录程序，并自动使用微信昵称、头像注册账号，登录后可以对用户个人信息进行修改。进入程序主界面后会自动定位用户位置。登录用户能够在平台上浏览他人发布的物品信息，也能对感兴趣的物品进行收藏，方便下次快速定位。借助商品分类和搜索功能快速搜索到用户需要的物品。排序规则可以选择同校（距离）优先，按热度排序等。购买商品后，系统要生成相应的消息提示和订单记录。

第 2 章 程序的开发环境及技术介绍

2.1 程序开发环境介绍

2.1.1 Bmob 后端云-云数据库

Bmob 后端云专注于为移动应用提供后端云服务，帮助开发者免去了几乎所有的服务器端编码的工作量，能够成倍降低开发成本和减少开发时间。它提供了可视化的云端数据表设计界面，并且支持十种不同数据类型的存储，让开发者能够轻松建立数据库和数据表。该平台支持微信小程序的开发，只要在 Bmob 控制台配置了小程序的密钥和安全域名，即可无障碍使用。

在 Bmob 上创建每个新表，都会自动创建几个系统字段（固定属性）：`objectId`（String）、`createdAt`（Date）、`updatedAt`（Date）。`objectId` 是每个表的固定主键，当然也可以自己增加新的唯一键；`createdAt` 和 `updatedAt` 两个字段分别是记录的创建时间和记录最近一次的更新时间，在增加或修改记录时会自动设置创建时间和更改时间，无需自己进行记录和更改。这三个属性在后面的数据库设计章节里将不再进行一一说明和展示。

2.1.2 微信开发者工具

为帮助开发者简单且高效地开发、调试微信小程序，微信团队在原有的公众号网页调试工具的基础上，推出了全新的微信开发者工具，集成了公众号网页调试和小程序调试两种开发模式。使用小程序调试，开发者可以完成小程序页面的开发调试、代码查看和编辑、小程序预览和发布等功能。该开发工具中的模拟器为开发者提供了多种不同的电子设备，当然用户也可以添加自定义的设备来调试小程序，以解决程序在不同分辨率机型上的适配问题。

2.2 程序中涉及的开发技术介绍

微信客户端为小程序提供宿主环境，使其可以完成很多普通网页无法实现的功能。整个小程序框架系统由逻辑层和视图层两部分组成。小程序提供了自己的视图层描述语言 WXML (WeiXin Markup Language) 和 WXSS (WeiXin Style Sheets)，以及基于 JavaScript 的逻辑层框架，并在视图层与逻辑层间提供了数据传输和事件系统，让开发者能够专注于数据与逻辑。框架管理了整个小程序的页面路由，可以做到页面之间的无缝切换，并给以页面完整的生命周期。开发者需要做的只是将页面的数据、方法、生命周期函数注册到框架中，其他的一切复杂操作都交由框架处理。

WXML 是该框架设计的一套标签语言，结合基础组件与事件系统，可以构建出页面的大体结构。它与 HTML 的不同之处在于有些标签名不同、多了些 `wx:if` 这样的属性以及 `{{ }}` 这样的表达式。

WXSS 是一套样式语言，用于描述 WXML 的组件样式。为了适应广大的前端开发者，WXSS 具备 CSS 的大部分特性。同时为了更适应微信小程序的开发，WXSS 对 CSS 进行了两方面的扩充。扩充的两个特性分别是：1) 新的尺寸单位 `rpx`，根据屏幕宽度进行自适应；2) 样式导入，使用 `@import` 语句导入外联样式表，其后跟上需要导入的外联样式表的相对路径，用分号表示语句的结束。

WXS (WeiXin Script) 是小程序的一套脚本语言，它与 JavaScript 是不同的语言，有自己的语法。本系统设计过程中还是主要以 JavaScript 语言为主。JavaScript 语言支持模型-视图-控制器 (MVC) 体系结构。该体系结构维护可读代码并清晰地分离程序代码的各个部分[14]。

腾讯位置服务为微信小程序提供了基础的标点能力、线和圆的绘制接口等地图组件和位置展示、地图选点等地图 API 位置服务能力支持，使得开发者可以自由地开发自己的微信小程序产品。在此基础上，腾讯位置服务微信小程序 JavaScript SDK 是专为小程序开发者提供的 LBS 数据服务工具包，可以在小程序中调用腾讯位置服务的 POI 检索、关键词输入提示、地址解析、逆地址解析、行政区划和距离计算等数据服务，使小程序更加强大。本程序中主要使用到距离计算、地点搜索、逆地址解析等功能。

第3章 程序总体设计

3.1 程序的整体框架和模块功能设计

易换小程序的主要功能分为三大模块：用户管理模块、商品管理模块以及消息管理模块。用户管理模块主要有：用户的个人信息管理、用户发布的商品及收藏商品的管理。商品管理模块主要功能是展示用户发布的商品列表、商品发布、商品详情、商品检索和分类查询等。消息管理模块是一些重要消息的处理操作。图 3.1 展示了本程序基本的功能结构。

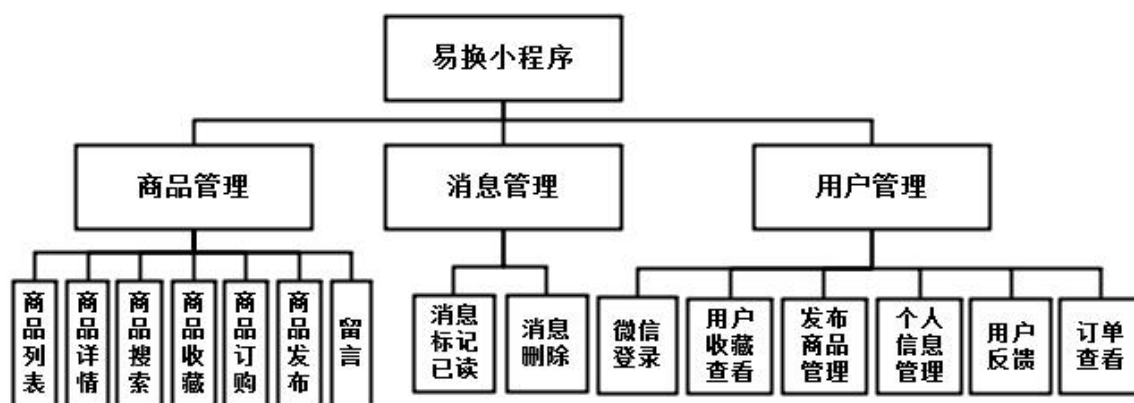


图 3.1 系统功能结构图

3.1.1 用户管理模块

为方便用户的使用，采用了一键授权登录的形式，获得授权后自动为初次使用程序的用户创建账号。

个人信息管理功能支持用户编辑修改自己的昵称、头像、学校等信息。

在个人发布管理中用户可查看自己所有的发布记录，并点击跳转进入详情页面，对其做出修改操作。用户收藏里记录了登录用户收藏过的商品，方便用户下次快速定位到自己感兴趣的商品。订单管理里列出了用户购买过的商品订单以及用户卖出的商品订单。

另有一个用户反馈功能，使用户可对本程序进行意见、建议、缺陷等的反馈。

3.1.2 商品管理模块

商品管理模块是本程序的重要模块。

商品发布时可以输入标题、描述、价格、地址以及要上传商品的相关图片。

首页里包括热门商品的轮播展示和最新发布商品的列表展示。为了较简洁明了的显示信息，应当在列表里展示出发布人信息、发布时间、转让商品信息及图片、价格等。

商品搜索模块是通过关键词查找相关的商品，方便用户快速找到需要的商品。为了更好的用户体验，可以记录并显示最近的热门搜索、用户的搜索历史等，并对搜索记录进行删除、清空操作。搜索页面还可以选择商品类别，根据用户选择的分类加载指定类目的商品。商品还可选择排序方式，主要有“默认排序”、“离我最近”、“人气最高”三种排序方式。

在商品详情模块里，用户发布时上传的商品图片通过轮播方式展示，商品的详细信息以及发布者的部分信息也都有展示。点击页面中的地址可进入地图模式查看详细位置；点击收藏即可将该商品加入用户自己的收藏列表，在已收藏状态下点击即为取消收藏。另外，该界面还有一个留言板区域，用户可在此处留言，发表对商品的看法等。

针对发布者与其他用户，进入详情页时显示的界面不完全一样。发布者在该页面右下角会看到一个“去修改”的按钮；而其他用户则是看到“我想买”的按钮，确认下单即可预订商品。

3.1.3 消息管理模块

一个用户作为卖家，若其发布的商品被其他用户收藏或是有人在商品详情页里留言时，需要收到通知，以便及时进行关注和回复。同时，用户订购了商品后，也需要有消息提醒发布人进行处理，从而保证交易顺利进行，才能有效地提高交易成功率。用户下单支付后，也会在微信里收到订单通知。

3.2 系统的流程设计

易换小程序的使用流程如下：用户首先进入授权界面，点击微信授权按钮，同意授权后，进入程序主界面。用户在首页可以看到热门商品或活动的轮播展示、以及最新发布商品的列表展示，列表会显示商品的大致信息。

用户可以点击悬浮的搜索图标，进入搜索界面，通过搜索关键词和商品分类筛选商品，也可在该页面选择商品列表的排序方式。

用户点击首页商品列表（或个人收藏列表、个人发布列表等）中的对应一个商品，即会跳转到该商品的详情界面。在详情界面中，用户可以查看商品的完整信息，点击商品地址进入地图模式查看详细位置；用户还可对商品进行收藏操作，收藏后用户可进入个人中心-我的收藏查看收藏的全部商品。用户还可在详情界面留言板进行留言。

点击商品发布，填写商品相关信息、定位选址，即可发布新的商品，在个人中心-我的发布里可查看用户发布的商品列表，页面上还有一个悬浮按钮，点击就直接跳转到商品发布页面。

系统的流程图如图 3.2 所示。

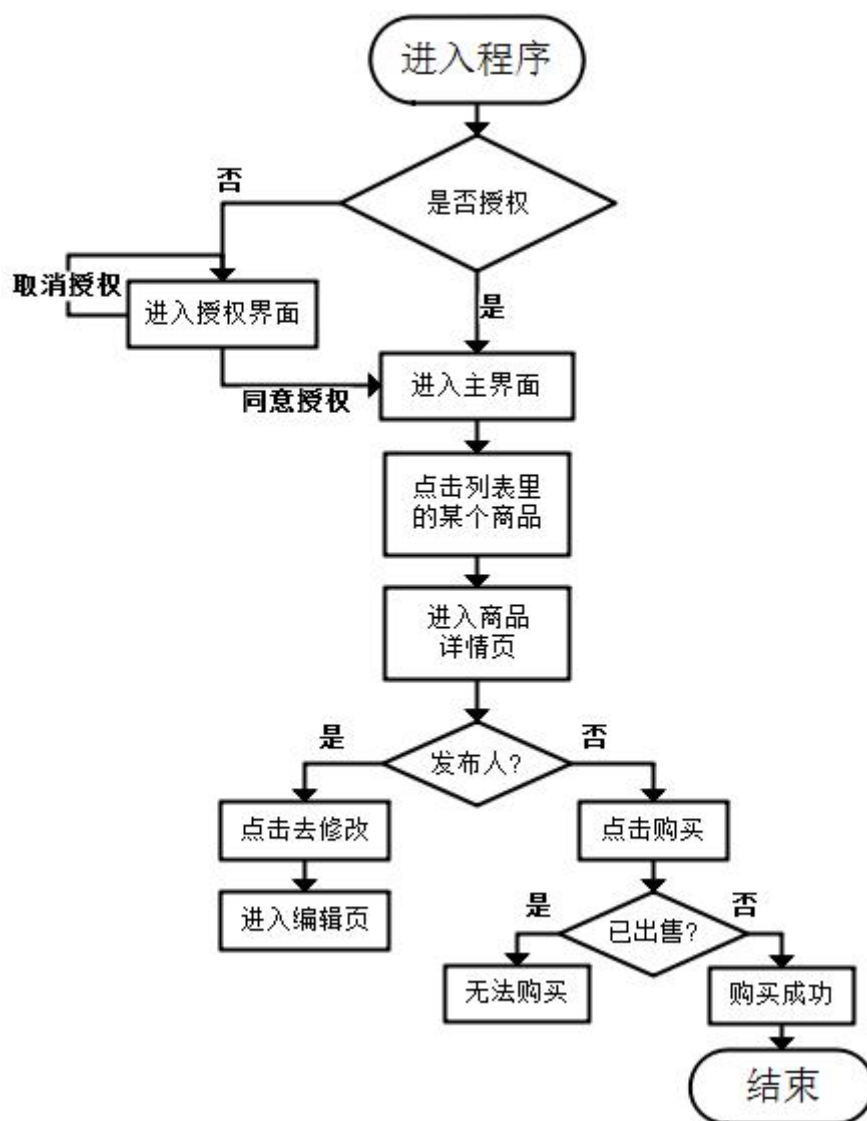


图 3.2 系统主要流程图

3.3 系统的数据库设计

本课题的小程序在 Bmob 后端云网页上创建了同名应用，采用建立云数据库的方式保存数据，既减少了后端搭建的工作量，又方便小程序可以在手机上进行调试预览。依照数据库设计原则，为本程序设计了八张云数据库表，分别用于：用户账号信息的保存、发布商品信息的保存、商品类别数据存储、用户收藏信息保存、用户对商品的评论保存、用户反馈保存、用户收到消息存储、订单详情存储。这些数据表发挥了各自的作用又相互联系，保证了数据的完整性和查询数据的高效性。数据库表汇总如图 3.3 所示。

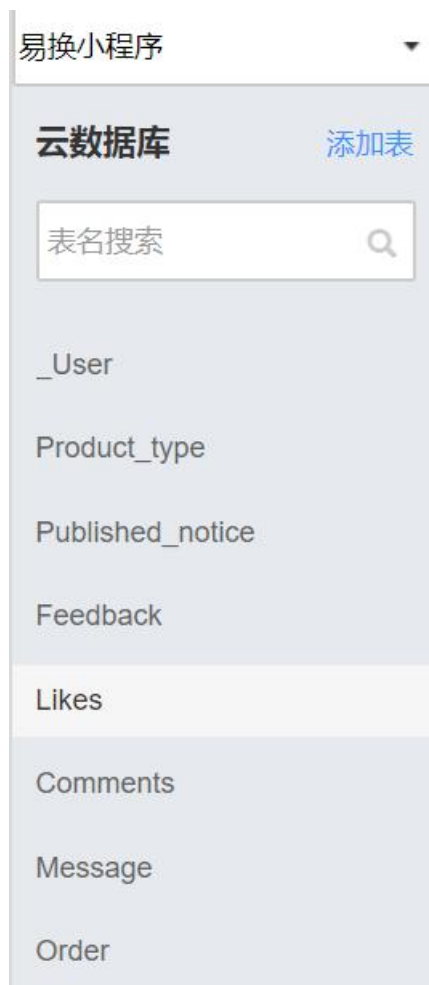


图 3.3 数据库表汇总图

下面将详细说明每个数据库表的设计与作用。

- (1) 用户表（_User），主要存放用户昵称、头像文件、性别、城市、省份、学校等信息。在用户授权微信账号登录时，自动获取用户的微信昵称、头像等信息存入用户表。这些信息可在个人信息里修改。其结构如表 3.1 所示。

表 3.1 用户表

| | | |
|----------|--------|--------|
| nickName | String | 用户昵称 |
| avatar | File | 用户头像文件 |
| gender | Number | 性别 |
| city | String | 城市 |
| province | String | 省份 |
| school | String | 学校 |

(2) 发布商品表 (Published_notice)，其结构如表 3.2 所示。

表 3.2 发布商品表

| | | |
|-------------|----------------|-----------|
| title | String | 发布标题 |
| description | String | 商品描述 |
| typeId | Number | 商品类别 Id |
| price | Number | 商品价格 |
| pic | File | 商品图片 |
| address | String | 地址 |
| longitude | Number | 经度 |
| latitude | Number | 纬度 |
| publisher | Pointer<_User> | 发布人指针 |
| viewCount | Number | 浏览数 |
| likeCount | Number | 收藏数 |
| liker | Array | 收藏人 Id 数组 |
| status | Number | 商品状态 |

商品表里除了包括用户发布商品时填入的信息外，还有发布成功后该商品被浏览的次数和收藏数。收藏人的 Id 数组是方便判断登录用户是否已收藏了该商品。经度和纬度属性是用来存储具体位置，方便用户能够在地图模式里查看精准定位。商品的状态 status 字段表示商品目前是否在售，0 表示在出售，1 表示已转卖。

(3) 商品类别表 (Product_type)，该表包括商品类别 Id 和类别名称字段。

设置该表的目的是方便商品类别数据的更新和获取，提高代码的可维护性。

(4) 收藏表 (Likes)，收藏表里包括收藏人的指针和商品指针，实现对收藏人和被收藏商品的多对多关系记录。通过字段类型 Pointer 可以查询出关联表的内容。

(5) 留言表 (Comments)，其结构如表 3.3 所示。

表 3.3 留言表

| | | |
|---------------|----------------------------|------------|
| content | String | 评论内容 |
| notice | Pointer< Published_notice> | 被评论商品指针 |
| publisher | Pointer< Published_notice> | 发布人指针 |
| olderComment | Poniter<Comments> | 上一条评论指针 |
| OlderUserName | String | 上一条评论用户的昵称 |

在留言表里，主要是对留言信息进行管理，记录评论人的昵称是为了可以显示在界面上。

(6) 反馈表 (Feedback)，其结构如表 3.4 所示。

表 3.4 反馈表

| | | |
|----------|----------------|-----------|
| feedUser | Pointer<_User> | 反馈用户指针 |
| title | String | 反馈标题 |
| content | String | 反馈内容 |
| feedPic | File | 反馈图片 |
| feedInfo | String | 用户信息、设备名等 |

反馈表是用户对于平台意见的记录，用户可以在遇到系统 bug 时做出反馈，提醒开发人员进行修正，或是给出对本程序的改进建议。其中 feedInfo 字段包括用户名、用户手机机型、微信版本号等信息，便于开发人员对与机型、版本相关的程序 bug 进行修复。这些信息是通过调用 wx.getSystemInfo 获得的。feedInfo 具体实例如图 3.4 所示：

```
feedInfo String
**手机型号 : HUAWEI**手机系统 : Android 9 **微信版本号 : 7.0.4
**手机型号 : devtools**手机系统 : iOS 10.0.1 **微信版本号 : 6.6.3
```

图 3.4 反馈表 feedInfo 字段实例图

(7) 消息表 (Message)，消息表是发布的商品受到其他用户操作时要发送给发布人提醒的记录。其结构如表 3.5 所示。

表 3.5 消息表

| | | |
|-----------|----------------|--------|
| fId | String | 发布人 Id |
| wId | String | 商品 Id |
| wTitle | String | 商品名称 |
| user | Pointer<_User> | 操作用户 |
| username | String | 操作用户名 |
| avatarUrl | String | 操作用户头像 |
| msgType | Number | 通知类型 |
| is_read | Boolean | 是否已读 |
| is_delete | Boolean | 是否删除 |

(8) 订单表 (Order)：订单表是用来记录用户间达成的交易订单的。其结构如表 3.6 所示：

表 3.6 订单表

| | | |
|------------|---------------------------|------|
| orderId | String8 | 订单编号 |
| buyer | Pointer<_User> | 买家 |
| seller | Pointer<_User> | 卖家 |
| notice | Pointer<Publisher_notice> | 商品 |
| totalPrice | Number | 订单总价 |

第4章 程序功能详细设计与开发

4.1 用户管理模块详细设计

该模块主要需要实现的功能有：用户一键登录、个人信息修改、个人发布商品管理、订单查看、用户收藏列表管理、用户消息管理。

4.1.1 用户一键登录功能

一键登录使用户避免了手动注册账号的麻烦。由于微信小程序开发不再支持使用 `wx.getUserInfo({})` 方法得到弹出的授权框，需要通过用户主动触发按钮的方式发起授权请求。WXML 文件中的主要界面代码如下：

```
<view wx:if="{{canIUse}}" >
  <view class='content'>
    <view>申请获取以下权限</view>
    <text>获得你的公开信息(昵称，头像等)</text>
    <text>将使用您的公开信息注册账号</text>
  </view>
  <button class='bottom' type='primary' open-type='getUserInfo'
    lang="zh_CN" bindgetuserinfo="bindGetUserInfo">
    授权登录
  </button>
</view>
```

`canIUse` 值取自 `wx.canIUse('button.open-type.getUserInfo')` 的返回值，作用是判断授权按钮在当前的微信版本下是否可用。用户允许授权后就调用 `wx.login` 接口获取登录凭证（`code`），进而换取用户登录的信息，包括用户的唯一标识（`openid`）以及本次登录的会话密钥（`session_key`）等。调用 `Bmob` 用户对象的 `loginWithWeapp`

方法，就会使用当前登录微信的用户身份登录（未注册的话就先注册再登录）；登录成功后，用户的 session 会在设备上持久化保存，之后可调用 Bmob.User.current() 方法获取当前登录的用户信息。数据库表 _Use 中的 objectId 即为用户的 openid。相关代码如下：

```
wx.login({
  success: function(res) {
    var user = new Bmob.User(); //注册 Bmob 用户
    user.loginWithWeapp(res.code).then(function(user) {
      //获取用户信息
      wx.getUserInfo({
        success: function(result) {
          app.globalData.userInfo = result.userInfo;

          var u = Bmob.Object.extend("_User");
          var query = new Bmob.Query(u);
          query.get(user.id, {
            success: function(result) {
              result.set()... // 表中对应属性设置
              result.save();

              wx.switchTab({
                url: '../home/home'
              })
            }
          });
        }
      });
    }
  });
});
```

登录成功后，自动跳转到程序主界面；若用户拒绝授权，则弹出提示框提示

用户先授权才能使用。界面效果如图 4.1 所示。



图 4.1 授权界面效果图

4.1.2 用户模块其它功能

用户的个人信息修改主要包含用户昵称、头像、学校等，用户可照自己的喜好进行设置，如图 4.2 所示。



图 4.2 个人信息页面图

个人发布商品管理、用户收藏列表管理，分别展示收藏用户发布的商品列表，用户收藏的商品列表。点击任意商品，都会跳转到商品详情页。主要涉及根据不同查询条件从对应的数据库表查询出需要的数据。用户收藏列表如图 4.3 中左边所

示。需要从 Likes 数据库表中读取登录用户的所有收藏记录。我的订单管理分为我买到的和我卖出的订单列表，如图 4.3 右边所示。



图 4.3 用户收藏商品及订单列表

4.2 商品管理模块详细设计

商品管理模块主要的功能有：发布商品的列表展示、商品分类搜索功能、商品发布功能、商品修改功能、商品详情查看、地图模式、商品收藏、商品预订等。

4.2.1 商品首页展示

首页是一个程序的门面，需要用心设计，才能给用户带来良好的使用体验。本程序的首页布局如图 4.5 所示。



图 4.5 首页布局图

本程序使用了自定义的标题栏 Navigation Bar。因为每个页面都需要设置标题栏，故将其封装为一个组件，方便多次使用。封装过程中，要设置适配各个机型的高度，通过调用 `wx getMenuButtonBoundingClientRect()` 获取导航栏高度，兼容任意机型。根据不同参数设置，为不同页面配置需要的布局样式。本程序设计的两种导航栏布局如图 4.6 所示。中间文字部分是页面的标题，在每个页面的 js 文件中单独设置。



图 4.6 导航栏布局图

一种是首页带有地图定位的导航栏，在程序首页里使用；另一种是带有返回上一页和返回主页按钮的导航栏，方便用户停留在较深层次的页面里时，快速回到主页。三个按钮事件都封装在了自定义组件中。

首页中间是热门商品或活动的轮播以及最新发布商品列表。为防止一进入首页就加载过多数据，影响用户体验，列表采用了分页加载的形式加载数据，每页加载十条数据。当用户拉动到商品列表底部时，自动载入下一页的数据，直到全部数据加载完。要实现这种功能，需要先获取总的发布数，调用 Bmob SDK 中的 `query.count` 方法，统计发布总数，然后由发布总数除以每页的条数得到总页数。再根据页码数获取商品信息列表。

商品发布的时间显示是根据创建时间，与现在时间进行差值计算，再转换成相应的单位，展示在界面上。需要先将时间数据里的横杠 '-' 转化为斜杠 '/'，再转化为时间戳，做差获得时间间隔。然后根据时间间隔判断适合的单位，最后返回相应的字符串。该方法对时间做了更人性化的处理，主要实现代码如下所示：

```
function pastTime(date) {  
    date = date.replace(/-/g, '/');  
    var timestamp = new Date(date).getTime();  
    var mistiming = Math.round(new Date() / 1000) - timestamp/1000;  
    var arrr = ['年', '个月', '星期', '天', '小时', '分钟', '秒'];  
    var arrn = [31536000, 2592000, 604800, 86400, 3600, 60, 1];  
    for (var i = 6; i >= 0; i--) {  
        if ((i==0 || mistiming < arrn[i-1]) && mistiming >= arrn[i]) {  
            var inm = Math.floor(mistiming / arrn[i]);  
            return inm + arrr[i] + '前'  
        }  
    }  
}
```

4.2.2 商品发布、修改功能

商品的发布界面与修改界面采用同一页面布局。要新增商品时打开编辑界面，各个输入框是空白的，而修改界面则需要根据商品 ID 预先加载好商品的详细信息。

商品发布界面如图 4.7 所示。两个页面的按钮绑定方法分别是向数据库增加一条数据和修改一条数据。

界面中的物品分类数据来自数据库表-商品类别表 (Product_type)，由于物品分类数据较固定，只需在 App.js 中程序刚打开时获取一次并存储，之后的页面和操作里就无需再发请求获取数据。

图中的定位获取功能将在 4.2.4 小节中给出说明。



图 4.7 商品发布界面图

4.2.3 商品搜索

列表在许多页面都有出现，对于不同功能的列表，需要在前端显示不同的数据和布局，具体操作此处不加详细说明。

商品搜索是本程序的重要功能。本程序在搜索页面中引入了一个名为 wxSearch 的组件，实现了查看热门搜索和历史搜索、以及搜索关键词的功能，如图 4.8 所示。更新热门搜索和历史搜索提示的功能代码如下：

```
function updateHotMindKeys(that, hotSearch){  
    var temData = that.data.wxSearchData;
```

```
var samekey = false,samemind = false;
for (var i in temData.keys){
    if (temData.keys[i] == hotSearch){
        samekey = true;
        break;
    }
}
if (samekey == false){
    temData.keys.pop(); //删除热门搜索的最后一个元素
    temData.keys.unshift(hotSearch); //将新的热门搜索添加到数组开头
}
for (var i in __mindKeys) {
    if (__mindKeys[i] == hotSearch){
        samemind = true;
        break;
    }
}
if (samemind == false) {
    __mindKeys.pop(); //删除记录搜索的最后一个元素
    __mindKeys.unshift(hotSearch); //将新记录搜索添加到数组开头
}
that.setData({
    wxSearchData: temData
});
}
```

通过输入的关键词和商品类别的选择组合查找用户需要的商品，帮助用户快速找到自己需要的商品。排序方式的选择有：默认排序，离我最近（距离远近排序），人气最高三种。用户可根据自己的喜好来选择排序方式。

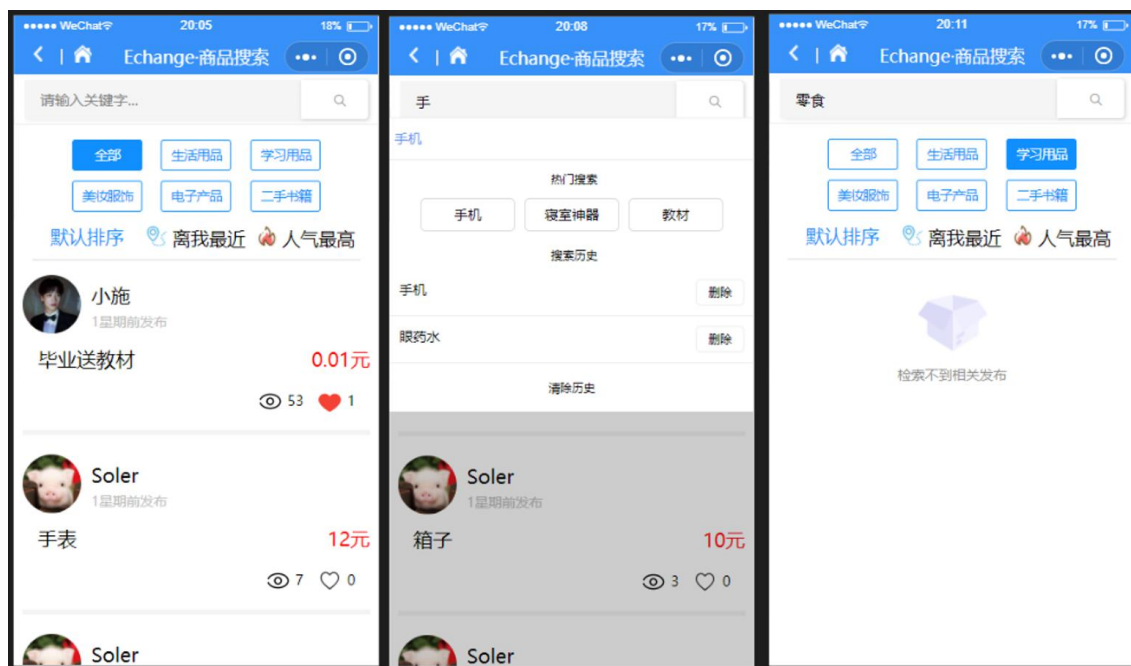


图 4.8 商品搜索界面

4.2.4 商品详情页

商品详情页界面设计如图 4.9 所示。左边是发布者进入详情页时显示的界面，右边是其他用户进入时的界面。实现方式是在跳转进详情页面时，除了带上商品 ID 之外，还要增加是否是发布人的标识；另外需要注意的是，商品的发布者查看自己的商品不会增加浏览数。发布者进入显示的页面，不包括收藏按钮，并显示跳转编辑页的按钮。代码如下：

```

if (notice.publisherId == app.globalData.currentUser.id) {
    wx.navigateTo({
        url: '../postDetail/postDetail?isMyPost=true&data=' +
JSON.stringify(notice)
    })
}
else {
    notice.viewCount++;
    wx.navigateTo({
        url: '../postDetail/postDetail?data=' + JSON.stringify(notice)
    })
}

```

```
this.addViewCount(notice.id); // 更新数据库里的浏览数
}
```

用户输入并发送留言后，会显示在商品详情界面的留言板上，并且系统会自动给商品的发布人发去一条信息，提醒发布者有人对该商品留言。留言前后对比也显示在下图中。

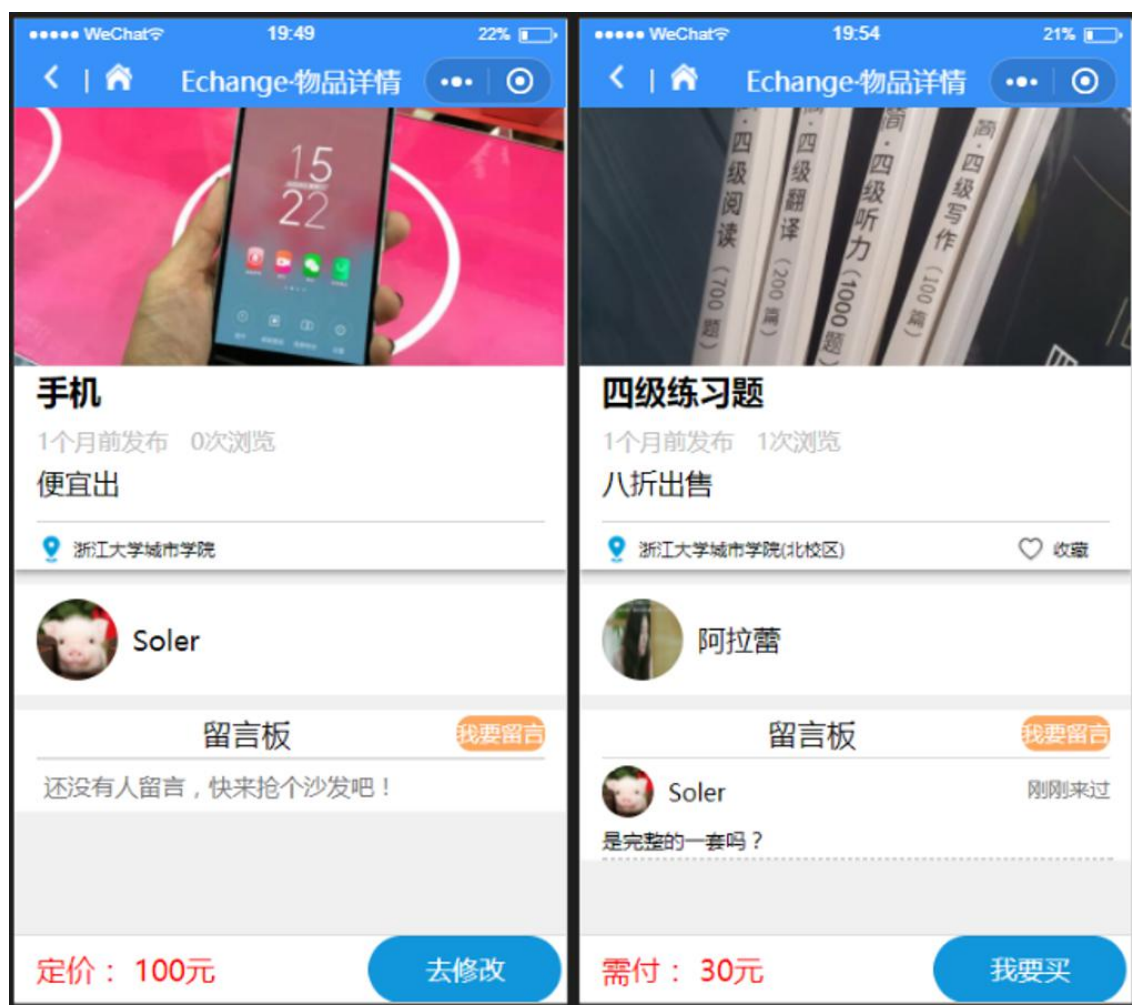


图 4.9 商品详情界面图

4.2.5 地图模式

地图模式分为多种显示情况。一是查看已经选好的设定好的地址。如点击商品详情页中的地址，就进入地图模式并在地图上标记出地址。这里使用了微信内置的接口：`wx.openLocation(Object object)`，需要传入的参数是经纬度和缩放比例等。地图模式如图 4.10 所示。



图 4.10 地图模式图

另一种则是需要用户选择详细地址的模式，如图 4.11 所示。模块上方有搜索框，可以输入地址快速定位到位置。



图 4.11 地图选择模式图

4.2.6 商品收藏

商品收藏功能的实现需要同时对多张表进行操作：发布商品表（Published_notice）、收藏表（Likes）、消息表。对于商品表需要修改收藏数 likeCount 和收藏人数组 liker 两个属性；对于收藏表要做增删操作。关键代码如下：

```
var Notice = Bmob.Object.extend("Published_notice");
var query = new Bmob.Query(Notice);
query.get(that.data.notice.id).then(res => {
    var likerList = res.get('liker') || [];
    if (!isLiked) {
        //增加
        likerList.push(app.globalData.currentUser.id);
        var likecount = res.get('likeCount') + 1;
    } else {
        // 删除
        var index = contains(likerList, app.globalData.currentUser.id);
        likerList.splice(index, 1);
        var likecount = res.get('likeCount') - 1;
    }
    res.set('liker', likerList)
    res.set('likeCount', likecount)
    res.save().then((res) => {
        wx.showToast({
            title: (that.data.isLiked)? '收藏成功': '取消收藏',
        })
    });
})

// 返回用户 Id 在数组里的 index
function contains(arrays, obj) {
    var i = arrays.length;
    while (i--) {
```



```
        if (arrays[i] === obj) {  
            return i;  
        }  
    }  
    return false;  
}
```

同时，要向商品发布用户发送一条消息，提示有其他用户收藏了该商品。消息创建时，要在消息表中查询是否存在该条消息，若不存在，则生成新消息；若已存在，则无需再生成。

4.2.7 商品订购

由于需要经过微信认证的小程序才能申请开通微信支付的接口（而个人类型帐号又不支持微信认证），本程序采用伪支付方式进行商品预定环节。通过弹出一个支付窗口，提醒用户确认交易。

用户确认后，需要对该商品的状态进行更改，同时，要给发布者发送一条消息，提醒发布者准备好交易。用户支付成功后，将会在微信里收到一条服务通知，告知用户已成功付款。这使用了微信小程序提供的模板消息。在微信公众平台网页端配置好相应的模板后，通过脚本控制发送。消息示例如图 4.12 所示。



图 4.12 消息通知示例图

发送的关键代码如下所示:

```
var temp = {
    "touser": wx.getStorageSync("openid"),
    "template_id":
"5VJubolYgG9mC7TzqFyqGCRE3-eHzUY5ieoWYfCieoM",
    "form_id": formId,
    "data": {
        "keyword1": { //订单编号
            "value": that.data.orderId
        },
        "keyword2": { //物品名称
            "value": that.data.notice.title
        },
        "keyword3": { // 金额
            "value": that.data.notice.price
        },
        "keyword4": { // 下单时间
            "value": util.formatTime(new Date())
        },
    },
    "emphasis_keyword": "keyword3.DATA"
}
Bmob.sendMessage(temp).then(function (obj) {
    console.log('发送成功', obj);
}, function (err) {
    console.log('失败', err);
});
```

4.3 消息管理模块详细设计

用户收到的消息主要有: 商品收藏、商品订购、用户留言等。每种消息对应不同的消息类别。

在消息列表里，需要对未读消息做一个特殊标记。用户标记为已读，要修改数据库里消息的读取状态（is_read）属性。点击列表可以展开显示用户对象的头像。左滑单条消息显示删除按钮，实现消息的软删除操作。

消息列表如图 4.13 所示。

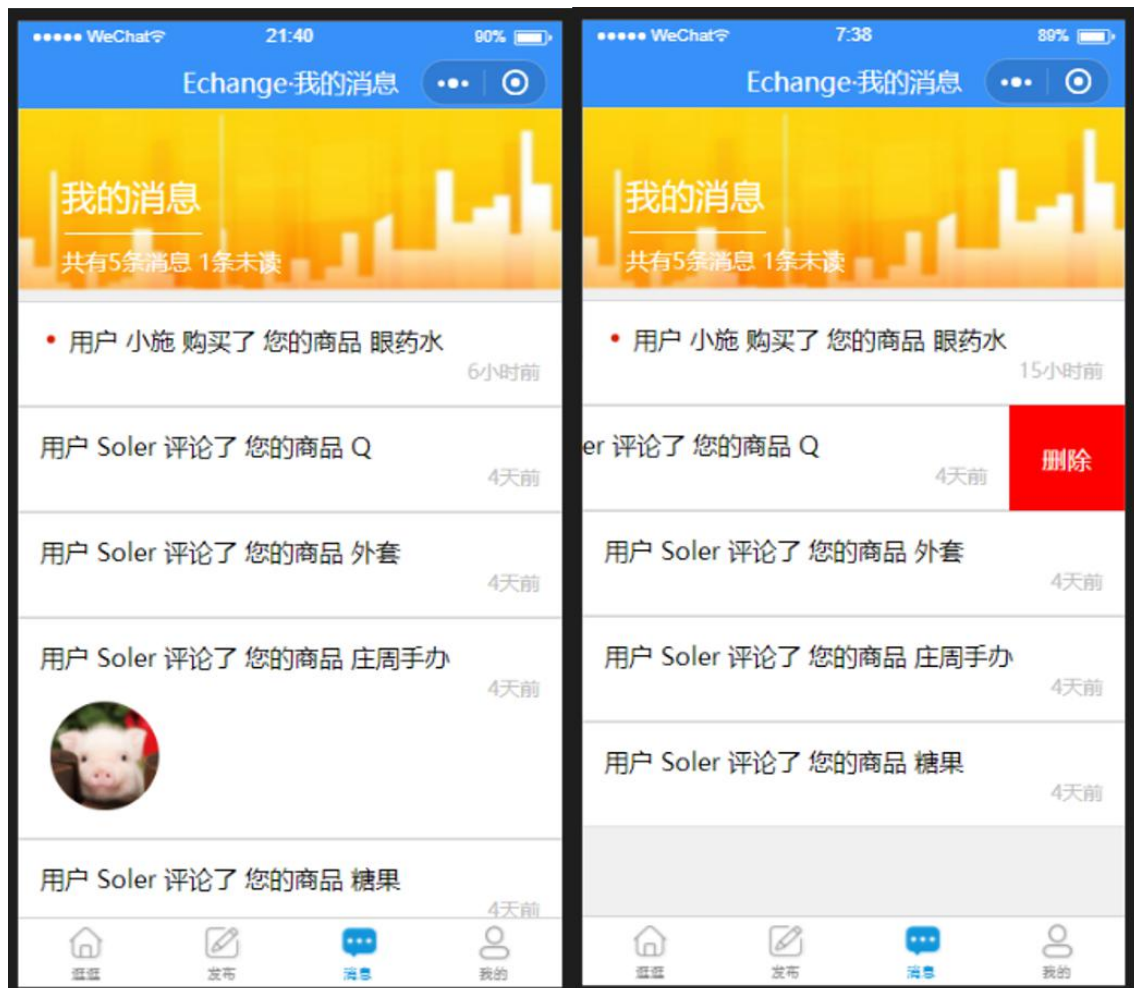


图 4.13 我的消息列表图

第 5 章 系统测试

5.1 系统测试原则

软件测试的基本原则就是站在用户的角度，对产品进行全面的测试，尽早、尽可能多地发现 Bug，并负责跟踪和分析产品中的问题，对不足之处提出质疑和改进意见。软件测试是对即将到来的程序的检查，也就是说测试要从项目一启动就开始，贯穿整个过程开发，而不能等程序开发完后才开始测试。对于软件测试来说，由第三方进行测试会更加客观、有效。但由于此设计的人员限制，本程序的测试将由开发者直接负责。同时会分享给周围的人进行使用和体验，改进不足之处。

制定软件测试计划是做好软件测试工作的前提。要提前设计好测试用例，从而提高测试效率，更多地发现错误，提高程序的可靠性。除了正确的测试用例，还应设计一些不正确的用例，避免忽略意想不到的错误。对修改后的程序进行重新测试时，也要严格依照测试用例执行，防止忽略掉修改错误时引起的新错误，所以一系列回归测试也是必要的。测试无法做到穷举，一定会存在一些问题，因此测试应把重点放到大问题上去而不是一些比较次要的问题[15]。

测试分为黑盒测试（也叫功能测试）和白盒测试。本次测试主要按照黑盒测试的方式进行，测试的重点在于各个界面及功能的规范性和正确性。

5.2 主要的测试用例

本次测试用例的设计将侧重于商品管理模块，就商品的发布、列表、查询、详情、留言、修改等方面进行设计。测试用例的列举和测试结果如表 5.1 所示。

表 5.1 测试用例及测试结果

| 用例编号 | 测试对象 | 测试条件 | 预期结果 | 实际结果 |
|------|---------|-----------------------|------------------------------------|------|
| 1 | 用户信息修改 | 修改用户所有信息（包括头像） | 修改后的信息能正常显示在各个界面 | 通过 |
| 2 | 商品关键字搜索 | 输入一定的关键字 | 搜索到所有含有关键字的商品 | 通过 |
| 3 | 商品类别搜索 | 点击任一分类 | 搜索到正确分类的商品 | 通过 |
| 4 | 商品排序 | 点击距离排序 | 列表优先展示距离近的商品 | 通过 |
| 5 | 商品排序 | 点击人气优先 | 依据人气规则优先展示人气高的商品 | 通过 |
| 6 | 商品收藏 | 收藏未收藏过的商品 | 收藏成功 | 通过 |
| 7 | 商品收藏 | 点击已收藏过的商品 | 取消收藏成功 | 通过 |
| 8 | 商品留言 | 发送留言 | 留言成功，并显示在详情页上 | 通过 |
| 9 | 商品详情 | 点击地址 | 在地图模式上显示正确位置 | 通过 |
| 10 | 商品购买 | 点击订购 | 购买成功，显示商品已出售，我的订单里新增一条数据，微信里收到订购通知 | 通过 |
| 11 | 商品发布 | 正确填写商品信息及地址并发布 | 商品发布成功 | 通过 |
| 12 | 商品详情 | 对于商品而言不同用户进入详情页显示不同界面 | 发布者进入显示修改按钮，普通用户进入显示购买按钮 | 通过 |
| 13 | 商品列表 | 浏览某个商品详情 | 商品的浏览数增加 | 通过 |
| 14 | 用户反馈 | 填写反馈 | 数据库里增加记录 | 通过 |

结论

易换小程序是一个适用于在小范围内使用的校园二手商品交易平台，为需要处理闲置商品的学生们提供一个便利的信息流通平台。学生群体较常进行二手交易的物品主要是教材书籍、学习用品等，具有一定特殊性。相较于市面上流行的二手交易网站或 APP 应用来说，本程序在一定程度上能较好的满足学生们的需求。

在程序的设计和开发过程中，前期的学习、资料收集花了一定的时间。经过科学的梳理和计划，有序地完成了各个功能的编写，实现了基本的功能，最终得到的成果还算令人满意。在界面美观、功能细节上还略有欠缺，有待改进。

由自己一个人进行整个项目的规划设计、实现测试，总是不可避免的存在局限性，以及很多不周全处。界面的美观程度，测试环节中没有覆盖到的测试用例等；功能上，还可以实现一下用户的信用制度建设，例如通过交易中买卖双方信誉评价来打分；还有一些自己暂时没有想到的、对于用户来说更友好的功能点，这些都是本程序可以进一步进行改进的。若是再进一步开展研究，可以在这些方面进行丰富和完善，最终发布正式小程序，推行到校园里使用，让程序发挥其作用。

就目前而言，仍有很大一部分学生的二手商品交易观念较为薄弱，导致二手交易在学校的普及度还不是非常不高。对此，在小程序发布、推广使用是，我认为学校方面还可以适当地对学生进行宣传教育，通过讲座、宣讲会、举办活动等方式，让循环利用的观念深入人心。希望有更多的学生能主动地对二手商品再利用，减少浪费、保护环境。

参考文献

- [1] 穆华. 大学生二手商品交易平台调查分析[J]. 合作经济与科技, 2017(6): 122-123.
- [2] 中国电子商务研究中心. 2018 年(上)中国 B2B 电子商务市场数据监测报告[R]. 浙江杭州, 2018.
- [3] 尼尔森公司. 尼尔森中国电子商务行业发展“杭州指数”白皮书[R]. 浙江杭州, 2015.
- [4] 杨娇. 大学校园二手商品交易市场存在的问题及对策[J]. 科技创业月刊. 2018, 31: 57-60.
- [5] 余箭, 高素勤, 盛亚丹. 高校跳蚤市场的发展现状研究与实践[A]. 经济研究导刊, 2014, 22: 80.
- [6] 李娜. 高校微信二手交易平台构建研究——以齐鲁医药学院为例[J]. 教育现代化. 2017, 32: 348-349.
- [7] 王祎, 曹天歌, 敖丽平. 校园跳蚤市场微信平台案例研究[J]. 中国市场. 2017, 1: 110-111.
- [8] 艾媒咨询. 2018 上半年中国在线二手交易市场监测报告[R]. 广东. 2018.
- [9] 苑亚坤. 校园交易网络平台探究[J]. 科技创业月刊, 2016(07), 25: 27.
- [10] 王昊. 大学校园二手商品交易市场的探讨[J]. 中国市场, 2016(18): 30-31.
- [11] TalkingData. 2018 小程序洞察报告[R]. 北京. 2018.
- [12] 杨平, 陈晓涛. 基于微信公众平台的校园二手交易平台设计[A]. 电子商务, 2018, 10: 95-96.
- [13] 曲宁, 吴渊. 微信二手交易平台在高校中的应用及前景分析[A]. 轻工科技, 2018(03).
- [14] S. Delcev and D. Draskovic. Modern JavaScript frameworks: A Survey Study[C]. 2018 Zooming Innovation in Consumer Technologies Conference (ZINC), 2018, pp: 106-109.
- [15] Hung Viet Nguyen, Hoan Anh Nguyen, Tung Thanh Nguyen, Nguyen. Auto-locating and fix-propagating for HTML validation errors to PHP server-side code[C]. T.N. 2011 26th

IEEE/ACM International Conference on in Automated Software Engineering (ASE). 美国:Hung Viet Nguyen,2011.

致谢

此次毕业设计，是对整个大学学习的一次很好的自我测评，让我从中发现了自己知识储备存在的不足。非常感谢在我进行毕业设计过程中帮助过我的每一个人。

首先要郑重感谢我的毕业设计导师——霍梅梅老师。从确定选题、撰写任务书，一直到毕业设计结束她都给予了我很多的帮助。帮我分析选题内容，解答我在设计过程中遇到的各种问题，以及对各个文档撰写都给出了详细的修改意见；期间更是频频关心我的设计进度，提醒我重要的时间节点及文件的提交流程，非常的耐心负责。

同时也要感谢在我编写程序、撰写论文遇到困难时，热心帮助我的同学和师兄师姐们。在他们的帮助下，我才能解决一个又一个的难题，顺利的完成毕业设计和论文。

最后还要感谢学校给了我一个优渥的学习环境，让我能在四年里快速成长，掌握一门技能，自信地踏入社会，去实现自己的人生价值。