

编号:

2020 - 2021 学年第 二 学期

浙大城市学院

ZHEJIANG UNIVERSITY CITY COLLEGE

实 验 报 告



实验课程名称 软件架构与应用开发

专 业 班 级 计算机 1803

学 号 31801141、31801142、31801140

学 生 姓 名 谢健、劳帅权、徐洋

实验指导教师 罗荣良

一、项目综述

现在的社会资源已经非常充裕，普通人基本上已经达到衣食无忧；但资源配置空洞还很多，共享思维的兴起，引发了产业的转型与思考。如摩拜单车就是典型例子，还有更大市场等着我们摸索。共享经济的洪流下，“共享租赁”不失为一条探索的主干道。

“安租”平台致力于为租赁商家和消费者提供一个进行自由共享租赁交易的平台，租赁分为“先租后买”、“以租代售”、“共享租赁”三种模式，具体商品的具体模式由租赁商家自主决定。消费者自由选择商品，下单、使用、退还全流程自动化管理，操作流程借鉴传统电商平台，极大降低消费者的学习成本。

平台满足不同租赁模式的商品配置及订单下单、支付、续租、退租全流程管理，同时针对每笔完成订单提供评价功能。此外，针对超时的订单，平台实时进行判断，并扣除相应超时时间押金。取消交易或完成交易后，平台会自动将剩余押金退还消费者，并将收益存入商家账户中。

二、功能简介

作为一个B2C模式的平台系统，相应功能主要分为三个端：

1.平台管理端

- ①对商家申请上架商品进行审核
- ②对已上架商品强制下架操作
- ③商品详情编辑

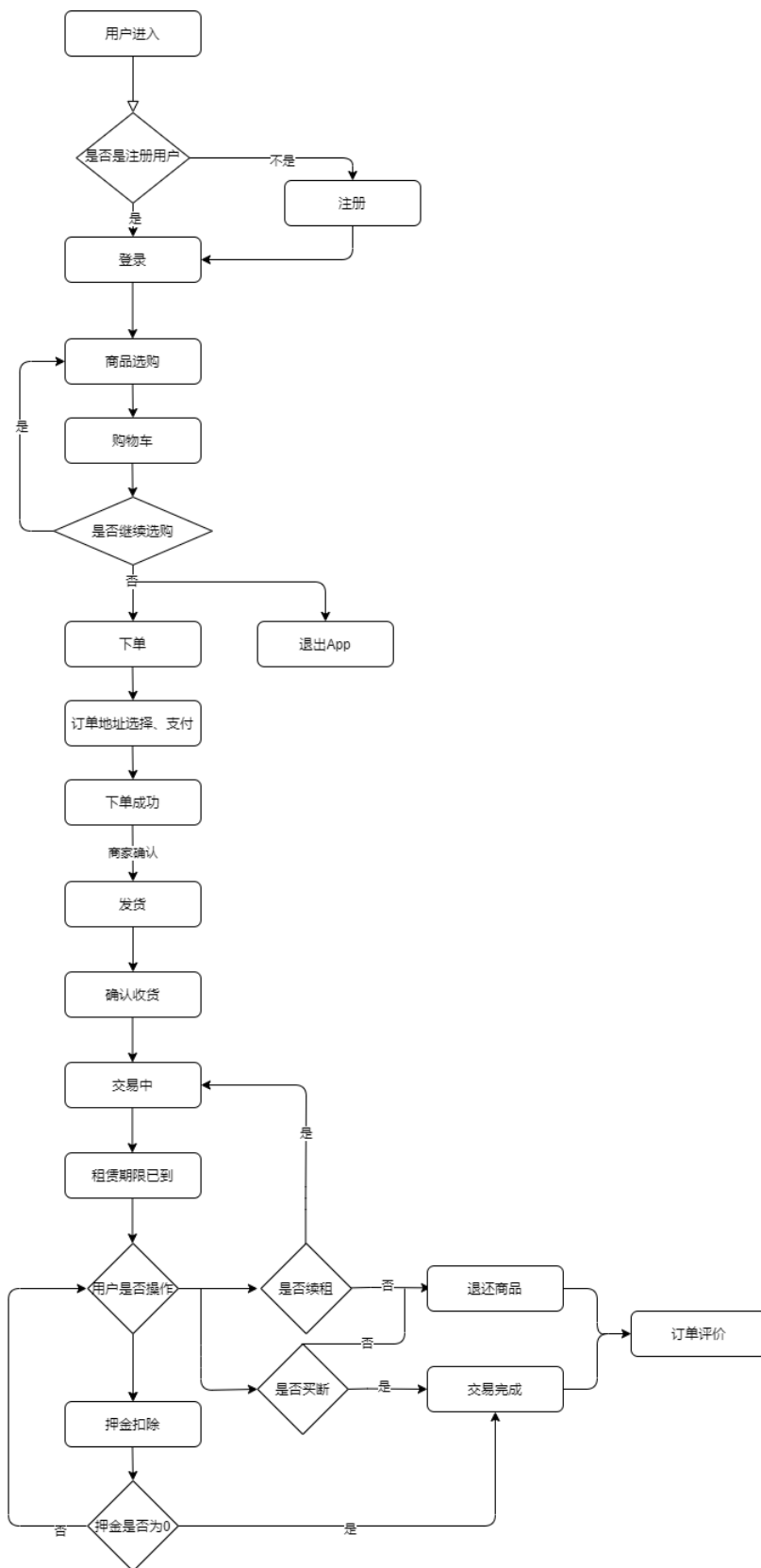
2.商家端

- ①商品编辑
- ②订单编辑
- ③地址编辑
- ④注册与登录

3.用户端

- ① 用户注册安租账号及登录
- ② 用户在首页查看及分类查看商品信息
- ③ 用户对收货地址进行查看、新增、修改、删除
- ④ 用户查看商品详情
- ⑤ 用户将商品加入到购物车中
- ⑥ 用户在购物车查看商品，选择相应商品进行下单
- ⑦ 用户选择地址，订单结算
- ⑧ 用户查看各类订单，退款、退货、确认到货、寄出到期商品
- ⑨ 用户对完成的订单进行评价，并可在商品详情页查看

三、项目流程图



四、项目设计

接口设计：

tbl-aftersales-controller Tbl Aftersales Controller		▼
POST	/aftersale/add	addAftersales
GET	/aftersale/deletebyid	deleteByid
GET	/aftersale/findByMerchantId	findByMerchantId
GET	/aftersale/findByOrderId	findByOrderId
GET	/aftersale/findByUserId	findByUserId
POST	/aftersale/modifyAftersales	modifyAftersales
tbl-cart-controller Tbl Cart Controller		▼
POST	/cart/add	addCart
GET	/cart/deletebyid	deleteCartByid
GET	/cart/findAllByUserId	findByUserId
GET	/cart/findById	findById
POST	/cart/modify	modifyCart
tbl-category-controller Tbl Category Controller		▼
POST	/category/add	addCategory
GET	/category/deletebyid	deleteCategoryByid
GET	/category/findAll	findAll
GET	/category/findById	findById
POST	/category/modify	modifyCategory
tbl-comment-statistic-controller Tbl Comment Statistic Controller		▼
POST	/comment_statistic/add	addComment
GET	/comment_statistic/findByGoodsId	findByGoodsId

tbl-goods-controller Tbl Goods Controller



POST /goods/add addGoods

GET /goods/deletebyid deleteGoodsById

GET /goods/findAllByCategoryId findByCategoryId

GET /goods/findAllByMerchantId findByMerchantId

GET /goods/findAllByMode findByMode

GET /goods/findAllForAdmin findAllForAdmin

GET /goods/findAllShuffle findAllShuffle

GET /goods/findById findById

POST /goods/modify modifyGoods

tbl-goods-order-controller Tbl Goods Order Controller



POST /goods_order/add addGoodsOrder

GET /goods_order/findAllByGoodsId findByGoodsId

GET /goods_order/findAllByMerchantId findByMerchantId

GET /goods_order/findAllByUserId findByUserId

GET /goods_order/findById findById

POST /goods_order/modifyMerchantAddress modifyGoodsOrderMerchantAddress

POST /goods_order/modifyOrderStatus modifyGoodsOrderStatus

POST /goods_order/modifyReceivingTime modifyGoodsOrderReceivingTime

tbl-goods-statistic-controller Tbl Goods Statistic Controller



GET /goods_statistic/findByGoodsId findByGoodsId

tbl-merchant-address-controller Tbl Merchant Address Controller



POST /merchant_address/add addMerchantAddress

GET /merchant_address/deletebyid deleteMerchantAddressById

GET /merchant_address/findAllByMerchantId findByMerchantId

GET /merchant_address/findById findById

POST /merchant_address/modify modifyMerchantAddress

POST /merchant_address/modifyDefault modifyMerchantDefaultAddress

tbl-user-address-controller Tbl User Address Controller



POST /user_address/add addUserAddress

GET /user_address/deletebyid deleteUserAddressById

GET /user_address/findAllByUserId findByUserId

GET /user_address/findById findById

POST /user_address/modify modifyUserAddress

POST /user_address/modifyDefault modifyUserDefaultAddress

tbl-user-controller Tbl User Controller



POST /user/add addUser

GET /user/deletebyaccount deleteUserByAccount

GET /user/deletebyid deleteUserById

GET /user/findByAccount findByAccount

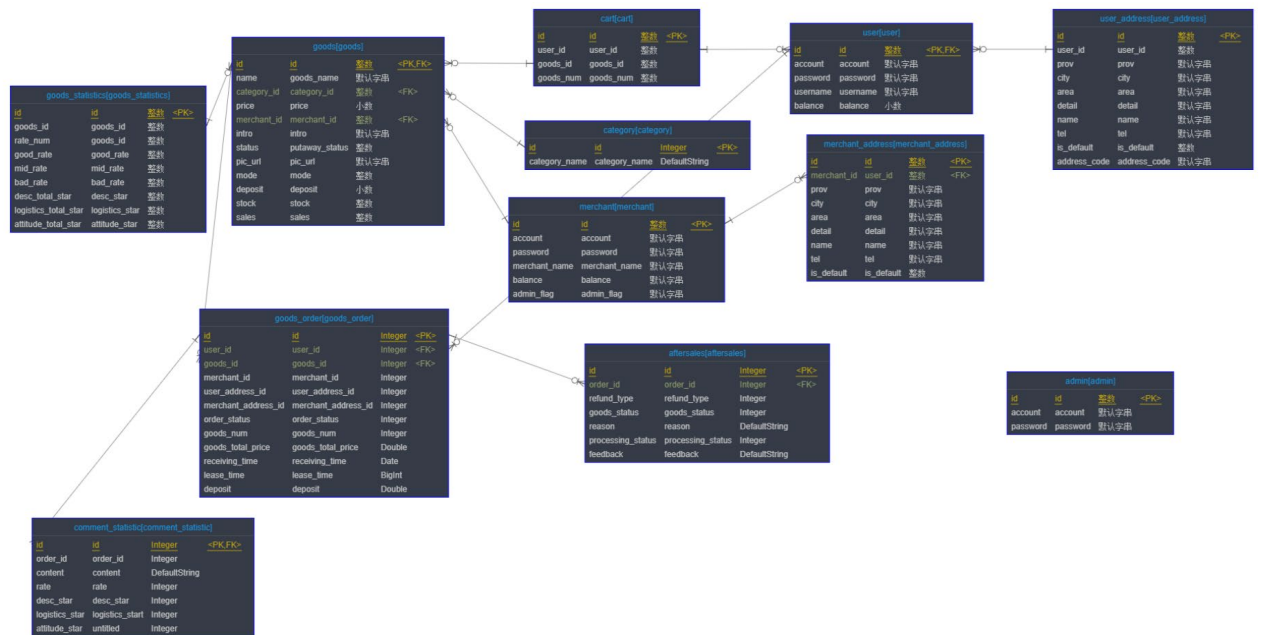
GET /user/login login

POST /user/modifyBalance modifyUserBalance

POST /user/modifyInfo modifyUserInfo

POST /user/modifyPassword modifyUserPassword

数据库设计



五、开发工具与技术

开发工具：

IntelliJ IDEA、VSCode

开发技术：

前端：JavaScript、HTML5、CSS3、Vue全家桶

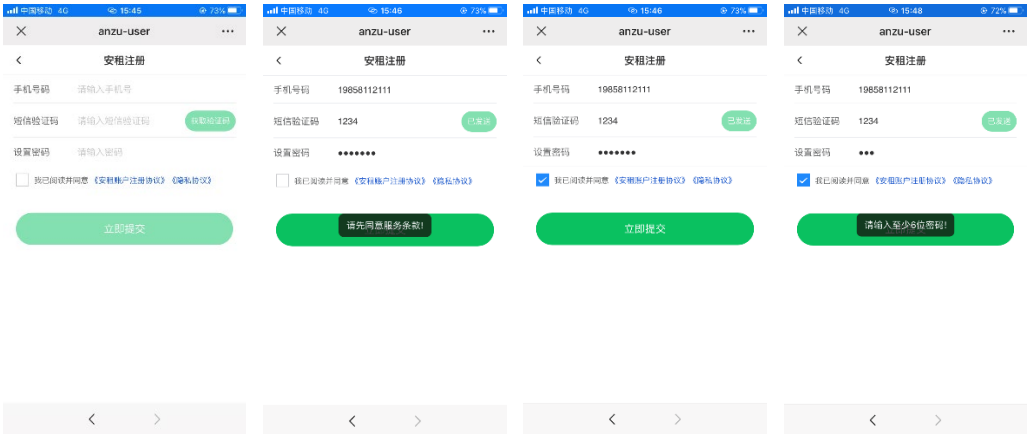
后端：SpringBoot2.0、SpringData JPA（使用JAVA）

数据库：MySQL

六、项目展示

用户端

① 注册与登录

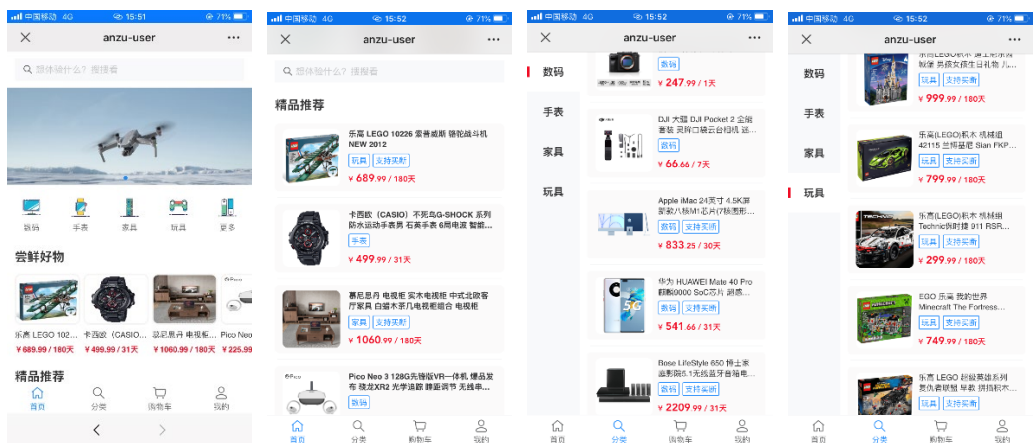


用户可以通过输入手机号码，获取短信验证码（实际无法获取），输入密码，并勾选同意用户协议等来完成一个账号的注册。用户所输入的信息必须完整有效，比如，密码至少为 6 位及以上。



用户可以使用注册完成的账号，通过两种方式进行登录。第一，手机快捷登录，通过获取短信验证码来完成登录；第二，账号密码登录，用户通过输入账号（手机号）和密码来完成登录。

② 首页与分类



用户可以在首页查看到系统提供的商品信息，从上到下为：轮播图、商品分类目录、新鲜好物栏、精品推荐栏。其中，通过点击商品分类可以直接跳转至相应的分类页面，新鲜好物和精品推荐会展示商品相应的图片、名称、价格等信息，用户通过点击对应商品，可以跳转至相应的商品详情页。

用户可以在分类页（数码、手表、家具、玩具四类）分别查看各类相应的商品及其信息，同样可以通过点击来跳转至相应的商品详情页。

③ 收货地址



用户可以在我的页中跳转到收货地址页，查看当前已添加的收货地址以及默认收货地址。用户可以自由切换默认收货地址。点击底部新增按钮，用户可以新增一个地址，需输入名字、电话号码、省市市区、详细地址等完整信息。用户亦可通过修改可删除按钮来对当前的地址信息进行操作。

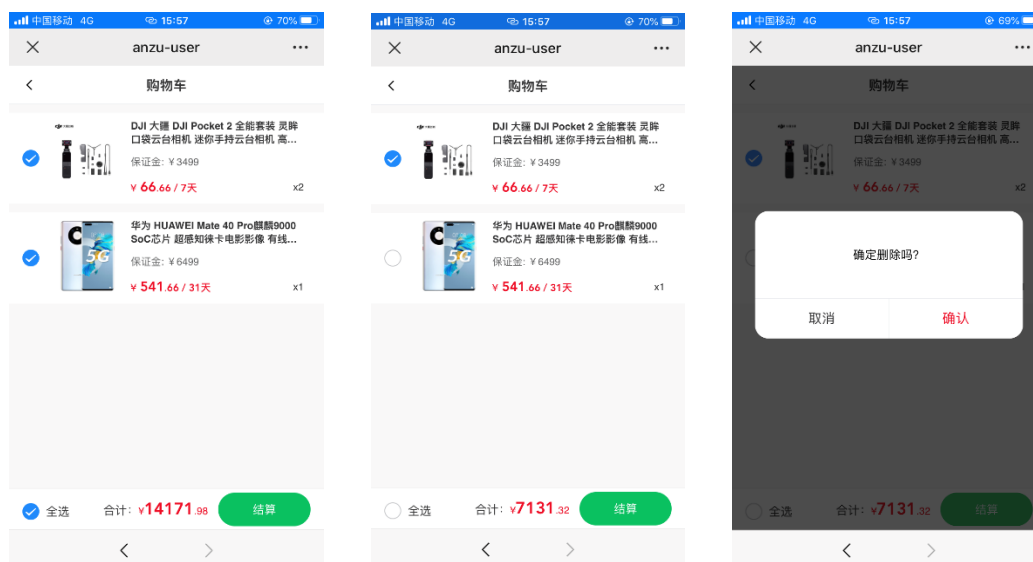
④ 商品详情



用户在商品详情页中查看到该商品的图片、名称、简介、租金/时间、是否支持买断、押金、销量等信息，下面信息包括，商家评价（商品描述、卖家服务、物流服务）和商品评价（来自用户的评价，好评/中评/差评及评价内容）。这里的商品信息及评价信息都会随交易的发生而产生相应的变化。

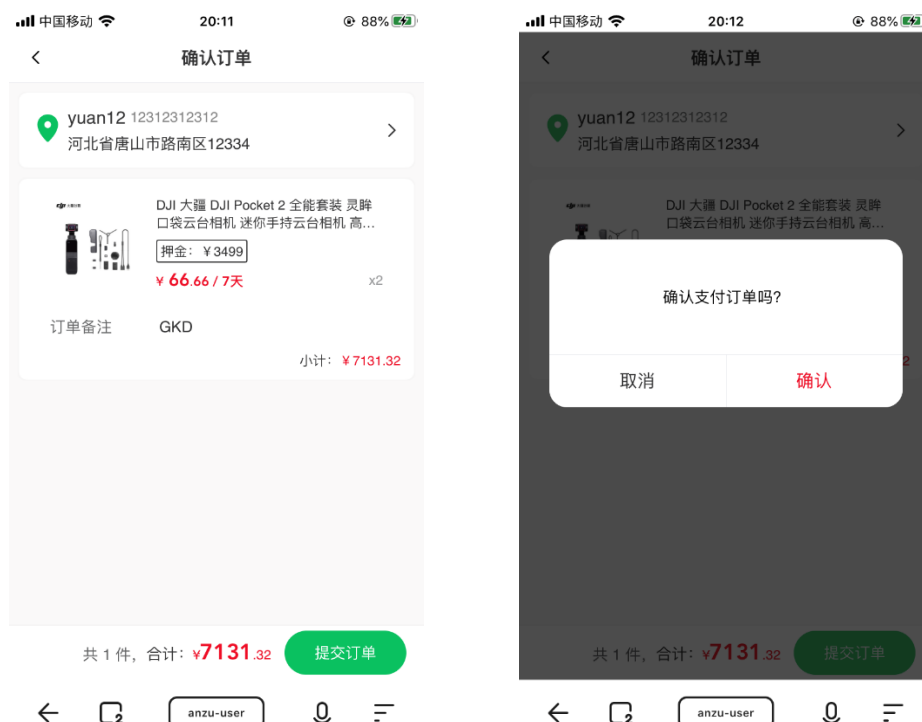
商品详情页底部导航栏包括左侧的首页及购物车按钮（直接跳转），右边的加入购物车按钮。点击加入购物车后，弹出底部提示框，用户可查看到当前商品的价格、押金、库存等信息，以及选择租赁商品的数量。点击确认后，相应的商品就加入到了购物车中。

⑤ 购物车



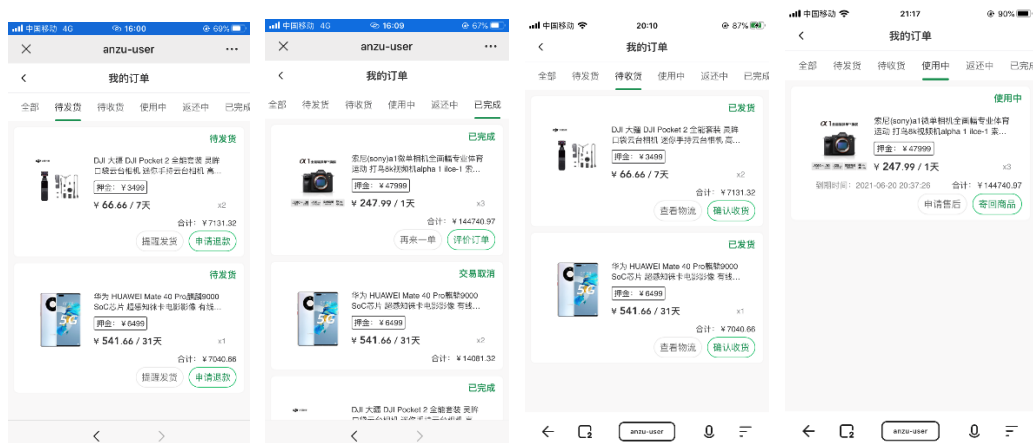
用户所有加入购物车的商品都会出现这里，其中有每一项的商品基本信息、价格、数量，用户可以选择其中的一项或多项（可以方便地全选或取消全选），或者删除购物车中的商品来进行操作。勾选需要下单的商品后，可以在下方查看到当前选择商品的合计需支付的金额（包括押金及租金），然后点击结算按钮进入结算。

⑥ 结算支付



用户选择相应的商品进入结算页面后，用户可以查看到，最上面是显示的默认地址，如果没有会显示需添加地址。用户自己自由在自己的地址列表里选择地址。地址下面是用户的订单信息，分商品展示，信息包括商品图片、名字、押金、价格、租期，订单小计等。底部栏显示当前有多少件订单，以及需支付的金额总计。用户可以点击结算按钮进行支付，若用户没有选择任何地址，则会提示用户进行选择。支付完成后，选择的商品将会从购物车中删除。

⑦ 订单



用户可以从我的页进入我的订单页。

订单状态分为以下几种：待发货、已发货、已寄出、已完成、使用中、超时、退款中、交易取消、评价完成等 9 种。

用户下单后，订单进入“待发货”状态，此时用户可以申请退款、提醒发货；申请退款后，会进入“退款中”状态，等待商家接收；商家发货后，显示“已发货”状态，此时用户可以选择查看物流或确认收货，确认后进入“使用中”状态，此时用户可以选择买断商品（如果可以），或申请售后，或寄出商品；买断后，订单直接进入“已完成”状态，寄出后，订单进入“已寄出”状态；商家收货后订单进入“已完成”状态，此时，用户可以对订单进行评价或再来一单。到此，一笔订单完成。

⑧ 评价

中国移动 4G 16:10 66%

anzu-user

发表评价

商品评价 ☒ 好评 ☐ 中评 ☐ 差评

索尼大法好!!

7/140

☒ 匿名评价

给本次购物打个星吧~

描述相符 ★★★★★ 好

物流服务 ★★★★★ 非常好

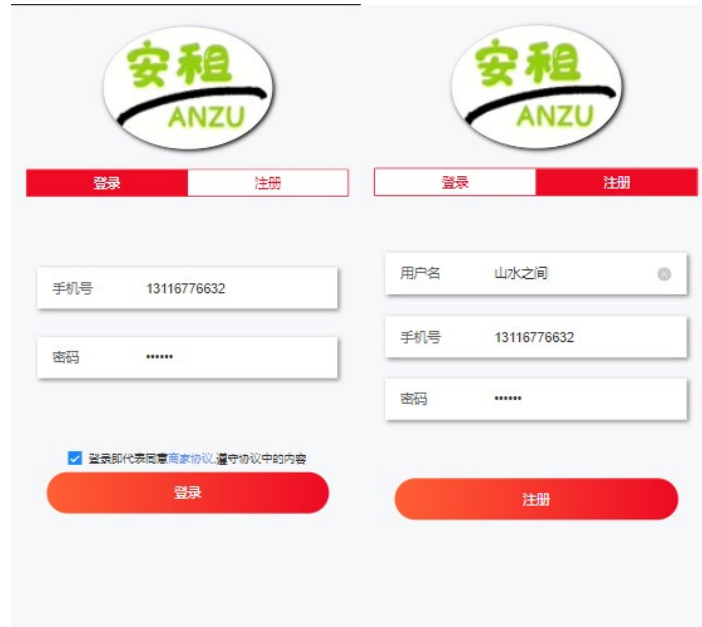
服务态度 ★★★☆☆ 一般

提交

用户可以对已完成的订单进行评价。首先是对商品评价，可以选择好评/中评/差评，然后输入文字评价。其次，可以对本次购物分别从商品描述、物流服务、商家服务进行评分（1-5 分）。评价完成后，用户的评价将会相应出现在对于的商品详情页中。

商家端

① 登录



The screenshot displays the ANZU merchant interface, which is split into two main sections: '登录' (Login) on the left and '注册' (Registration) on the right. Both sections feature the ANZU logo at the top. The '登录' section has a red bar with '登录' and '注册' tabs, and a '手机号' (Mobile Number) input field with the value '13116776632'. The '注册' section has a red bar with '登录' and '注册' tabs, and a '用户名' (Username) input field with the value '山水之间'. Both sections have a '密码' (Password) input field with the value '*****'. At the bottom of each section is a red button labeled '登录' or '注册'. A checkbox labeled '登录即代表同意商家协议, 遵守协议中的内容' is located below the login button in the '登录' section.

在登录界面可以实现用户的登录、注册以及管理员的登录

注册：商家输入用户名、手机号、密码之后点击注册按钮，通过手机号查询数据库中是否存在该用户，如果数据库中已经存在，则提示错误信息“用户已存在”，否则提示“注册成功，请前往登录”，登陆界面的手机号与密码输入栏会保留注册界面的信息，避免重新输入的麻烦。

登录：商家输入手机号、密码之后点击登录按钮，系统检查手机号和密码是否正确，如果不正确提示登录失败，如果正确检查该商家是否有管理员标识，如果有则显示管理员端界面，否则显示商家界面。

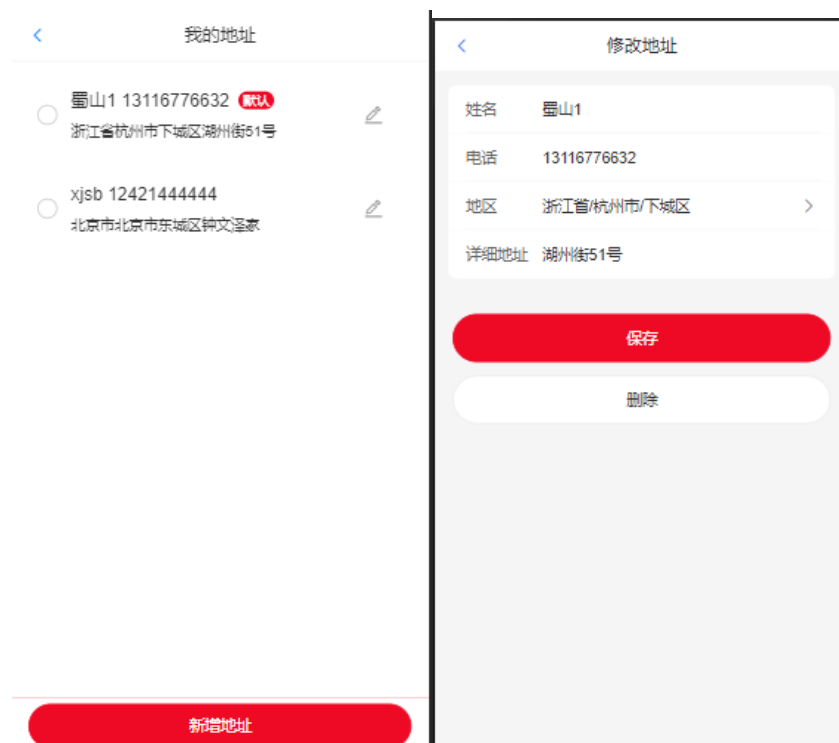
② 个人管理



在个人页显示商家手机号，商家名称以及余额等信息

在个人界面初始化时，根据商家手机号查询商家信息，获取用户名，手机号，余额等信息并渲染，如果强行进入该界面，会检查用户是否已登录，若未登录则强制跳转登录界面

③ 收货地址管理



在收获地址页面，商家可以实现选择地址，新增地址，以及对原有地址的修改和删除

在个人页面中点击我的地址按钮后跳转该页面，初始化时根据商家 id 查询该商家的所有地址信息，渲染出地址列表；点击新增地址后输入地址信息，将地址信息存入数据库，点击编辑按钮，进行修改信息后，对数据库中指定地址 id 的地址信息进行更新。

④ 订单管理



在订单页，商家可以查看自己商品的交易信息，有待发货、待确认、退款中、已完成 4 个状态，并且可以根据不同的状态进行不同的操作。

在个人页面中点击我的订单按钮跳转至该页面，初始化时根据商家 id 获取所有订单信息，在前端将其分为以上四类，并根据不同的状态提供给商家不同的操作，点击对应的操作改变订单状态，如在待确认页面点击完成即可完成订单，同时商家和用户的余额也会发生变化，商家获得本次交易金额，用户余额多出预先交付的押金。

⑤ 商品管理



在商品页中，商家可以管理自己的商品，商品的信息都会显示在卡片当中，商品分为四个状态：已上架、审核中、已下架、需调整，其中对已上架状态的商品可以进行下架商品的操作，审核中状态的商品可以进行极速操作，无需管理员审核即可商家，已下架商品可以重新上架，而需调整即被管理员驳回的商品需要修改信息后重新提交审核。

点击商品跳转商品页，在初始化时根据商家 id 获取所有商品信息，经过处理后分为以上四个状态并进行展示，根据不同状态提供不同操作，如点击修改跳转修改商品界面，点击下架可以改变数据库中该商品的状态，调整为下架状态并重新渲染。

上架与修改商品

<

添加商品

商品名


商品名

商品描述

商品描述(200字以内)

0/200

商品图片



商品分类

请选择

租赁模式

请选择

价格(元)

请输入价格

押金(元)

请输入押金

租赁时间(天)

请输入租赁时间

库存

请输入库存

提交商品

<

添加商品

商品名


DJI 大疆 无人机 悟Inspire 2 四轴专业超

商品描述

【DJI FPV新品上市】大疆全新品类，第一人称视角沉浸式飞行体验，4K/60fps，超广角拍摄，全新运动模式。

55/200

商品图片



商品分类

数码类

租赁模式

以租代售

价格

55.99

押金

19999

租赁时间

1

库存

53

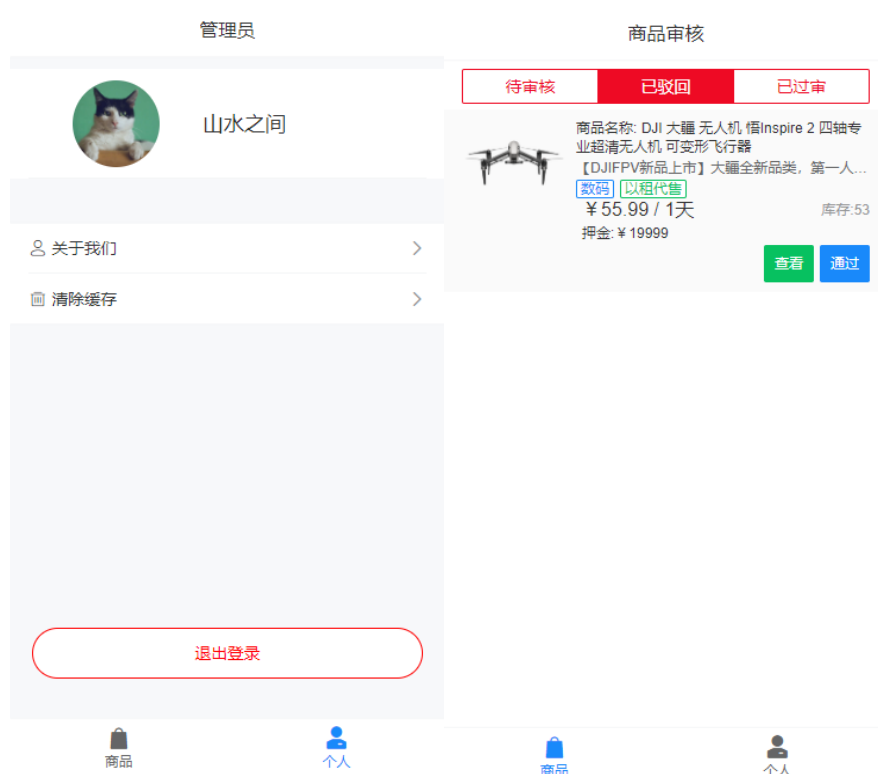
修改商品

删除商品

商家可以对商品进行添加、修改删除操作，点击会跳转到表单页面，在此用户填写不同表单内容后可以进行提交商品，使之进入审核状态，也可以修改和删除已存在的商品，更好的进行管理。

点击添加或修改跳转到商品编辑页面，在这里商家可以指定商品的信息，其中商品图片时商家选择本地图片进行上传，这里上传到七牛云中，并获取图片url渲染本地的图片。若未填写完所有信息就点提交或修改按钮则会提示错误，在填写完所有信息和点击提交会将以上信息传入商品数据库中，点击修改会更新数据库中对应商品的信息

管理员端



在登陆界面输入管理员的账号和密码后即可登录，管理员可以对商品进行审核，对违规商品进行驳回操作，同时也可以对已驳回的商品重新审核，并决定是否通过。

由于是管理员，在获取商品信息时可以获取所有商家的商品信息，点击查看按钮可以查看商品更详细的信息，但不能对其进行编辑，只能对商品状态进行更改，如点击通过可以把审核中商品的状态改为已上架，以便用户端可以查看并购买对应的商品。其也可以通过将已上架的商品强制驳回，将其改为待调整状态，需商家修改信息并重新提交审核后才能上架。

七、核心功能实现逻辑简述

1、账户密码 md5 加密,保证用户安全:

```
1. String md5Password = DigestUtils.md5DigestAsHex(password.getBytes())
;
2. if (userEntity == null)
3. {
4.     result.put("port","500"); //无该用户
5.     result.put("msg","无该用户");
6.     return result;
7. }
8. else if(!userEntity.getPassword().equals(md5Password)){
9.     result.put("port","400"); //密码错误
10.    result.put("msg","密码错误");
11.    return result;
12. }
13. else{
14.     result.put("port","200");
15.     result.put("data",userEntity);
16.     return result; }
```

数据库呈现:

id	account	password
3	123	5152aff82519f6047cee52c01c287a94
5	12343	65ded5353c5ee48d0b7d48c591b8f430

2、下单或取消订单后，自动更新库存和销量:

```
if (goodsEntity.getStock() < goods_num){
    result.put("port","400");
    result.put("msg","库存不足");
}
else{
    tblGoodsOrderService.addGoodsOrder(user_id,goods_id,merchant_id,user_address, merchant_address: -1, order_status: 1,
    goods_num,goods_total_price, receiving_time: null,goodsEntity.getLeaseTime(),deposit);
    tblGoodsService.modifyGoods(goodsEntity.getGoodsName(),goodsEntity.getPrice(),goodsEntity.getCategoryId(),
    goodsEntity.getMerchantId(),goodsEntity.getIntro(),goodsEntity.getStatus(), goodsEntity.getPicUrl(), goodsEntity.getMode()
    ,goodsEntity.getDeposit(), stock: goodsEntity.getStock()-goods_num, sales: goodsEntity.getSales()+goods_num,
    goodsEntity.getLeaseTime(),goods_id);
    result.put("port","200");
}
}

else if (order_status == -4)
{
    double user_money = userEntity.getBalance() + goodsOrderEntity.getDeposit() + goodsOrderEntity.getGoodsTotalPrice();
    tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),user_money);
    tblGoodsService.modifyGoods(goodsEntity.getGoodsName(),goodsEntity.getPrice(),goodsEntity.getCategoryId(),goodsEntity.getMerchantId(),goodsEntity.getIntro(),goodsEntity.
```

3、对多条数据修改出现异常时，数据库实现回退：

```
public JSONObject modifyGoodsOrderStatus(@RequestParam(value = "id")int id,
                                         @RequestParam(value = "order_status")int order_status)
{
    JSONObject result=new JSONObject();
    GoodsOrderEntity goodsOrderEntity = tblGoodsOrderService.findById(id);
    GoodsEntity goodsEntity = tblGoodsService.findById(goodsOrderEntity.getGoodsId());
    UserEntity userEntity = tblUserService.findById(goodsOrderEntity.getUserId());
    MerchantEntity merchantEntity = tblMerchantService.findById(goodsOrderEntity.getMerchantId());
    try {
        tblGoodsOrderService.modifyGoodsOrder(goodsOrderEntity.getUserId(),goodsOrderEntity.getGoodsId(),goodsOrderEntity.getMerchantId(),goodsOrderEntity.getUser
        if(order_status == 4)
        {
            double merchant_money = merchantEntity.getBalance() + goodsEntity.getDeposit() * goodsOrderEntity.getGoodsNum() - goodsOrderEntity.getDeposit() + good
            double user_money = userEntity.getBalance()+goodsOrderEntity.getDeposit();
            tblGoodsService.modifyGoods(goodsEntity.getGoodsName(),goodsEntity.getPrice(),goodsEntity.getCategoryId(),goodsEntity.getMerchantId(),goodsEntity.getI
            tblMerchantService.modifyMerchant(merchantEntity.getId(),merchantEntity.getAccount(),merchantEntity.getPassword(),merchantEntity.getMerchantName(),mer
            tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),user_money);
        }
        else if (order_status == -4)
        {
            double user_money = userEntity.getBalance() + goodsOrderEntity.getDeposit() + goodsOrderEntity.getGoodsTotalPrice();
            tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),user_money);
        }
        result.put("pont","200");
    }
    catch (Exception e){
        tblGoodsOrderService.modifyGoodsOrder(goodsOrderEntity.getUserId(),goodsOrderEntity.getGoodsId(),goodsOrderEntity.getMerchantId(),goodsOrderEntity.getUser
        tblGoodsService.modifyGoods(goodsEntity.getGoodsName(),goodsEntity.getPrice(),goodsEntity.getCategoryId(),goodsEntity.getMerchantId(),goodsEntity.getIntr
        tblMerchantService.modifyMerchant(merchantEntity.getId(),merchantEntity.getAccount(),merchantEntity.getPassword(),merchantEntity.getMerchantName(),merchan
        tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),userEntity.getBalance());
        result.put("pont","500");
        result.put("msg","修改异常");
    }
    return result;
}
```

4、自动检测订单是否超时,修改状态并扣除押金。

实现逻辑：每 15s 对数据库订单状态为交易中或超时的进行检索，若交易中订单已超过租赁期限则更新为超时，并从押金中按照租金*1.5/天的超时费率进行扣除。若订单押金已被扣至为 0，则订单状态自动更新为“已完成”。

```
1. @Scheduled(cron = "* * /5 * * * ?")
2. public void Schedule(){
3.     List<GoodsOrderEntity> goodsOrderEntityList = tblGoodsOrderService.findAll();
4.     for (int i=0;i<goodsOrderEntityList.size();i++){
5.         GoodsOrderEntity goodsOrderEntity = goodsOrderEntityList.get(i);
6.         GoodsEntity goodsEntity = tblGoodsService.findById(goodsOrderEntity.getGoods
            Id());
7.         long dueTime = goodsOrderEntity.getReceivingTime().getTime() + goodsOrderEn
            tity.getLeaseTime()*1000;
8.         long nowTime = System.currentTimeMillis();
9.         double perPrice = goodsEntity.getPrice() / (goodsEntity.getLeaseTime()*1.0/60/60
            /24) * 1.5;
10.        if ( dueTime < nowTime)
11.        {
12.            if (goodsOrderEntity.getOrderStatus() == 5){
13.                goodsOrderEntity.setOrderStatus(6);
14.            }
15.            if (goodsOrderEntity.getDeposit() > goodsEntity.getDeposit()-((int)((nowTime-d
                ueTime)*1.0/1000/60/60/24)+1)*perPrice){
```

```

16.         goodsOrderEntity.setDeposit(goodsEntity.getDeposit()-(int)((nowTime-due
Time)*1.0/1000/60/60/24)+1)*perPrice);
17.     }
18.     if (goodsOrderEntity.getDeposit() < 0){
19.         goodsOrderEntity.setDeposit(0);
20.         goodsOrderEntity.setOrderStatus(4);
21.     }
22.     tblGoodsOrderService.modifyGoodsOrder(goodsOrderEntity.getUserId(),goods
OrderEntity.getGoodsId(),goodsOrderEntity.getMerchantId(),goodsOrderEntity.getUserA
ddressId(),goodsOrderEntity.getMerchantAddressId(),goodsOrderEntity.getOrderStatus()
,goodsOrderEntity.getGoodsNum(),goodsOrderEntity.getGoodsTotalPrice(),goodsOrderE
ntity.getReceivingTime(),goodsOrderEntity.getLeaseTime(),goodsOrderEntity.getDeposi
t(),goodsOrderEntity.getId());
23.     }
24. }
25. }
26. }

```

5、交易完成或交易取消后，自动返还剩余押金以及将收益存入商家账户当中

```

if(order_status == 4)
{
    double merchant_money = merchantEntity.getBalance() + goodsEntity.getDeposit() * goodsOrderEntity.getGoodsNum() - goodsOrderEntity.getDeposit() + goodsOrderEntity.getGoodsTotalPrice();
    double user_money = userEntity.getBalance()+goodsOrderEntity.getDeposit();
    tblMerchantService.modifyMerchant(merchantEntity.getId(),merchantEntity.getAccount(),merchantEntity.getPassword(),merchantEntity.getMerchantName(),merchant_money,merchant_money);
    tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),user_money);
}
else if (order_status == -4)
{
    double user_money = userEntity.getBalance() + goodsOrderEntity.getDeposit() + goodsOrderEntity.getGoodsTotalPrice();
    tblUserService.modifyUser(userEntity.getId(),userEntity.getAccount(),userEntity.getPassword(),userEntity.getUsername(),user_money);
    tblGoodsService.modifyGoods(goodsEntity.getGoodsName(),goodsEntity.getPrice(),goodsEntity.getCategoryId(),goodsEntity.getMerchantId(),goodsEntity.getIntro(),goodsEntity.getGoodsNum());
}

```