



同濟大學

TONGJI UNIVERSITY

软件工程课程设计

基于 3D 模型展示的云开发点餐小程序

软件设计规范

夏哲辉 1853996

董震宇 1852143

陈永顺 1854002

2021 年 1 月 1 日

目录

1	引言	7
1.1	设计依据	7
1.2	编写目的	7
1.3	参考资料	8
1.4	假定和约束	8
2	系统总体架构设计	9
2.1	系统体系结构总概	9
2.2	视图层	9
2.3	逻辑层	10
2.4	系统层	10
3	模块设计	11
3.1	用户信息管理模块	11
3.1.1	模块描述	11
3.1.2	模块功能	11
3.1.3	模块详细设计	12
3.2	商品信息管理模块	14
3.2.1	模块描述	14
3.2.2	模块功能	14

3.2.3 模块详细设计	14
3.3 支付订单管理模块	23
3.3.1 模块描述	23
3.3.2 模块功能	23
3.3.3 模块详细设计	23
4 类设计	27
4.1 用户信息管理模块	27
4.1.1 类图总览	27
4.1.2 User 类	27
4.2 商品信息管理模块	28
4.2.1 类图总览	28
4.2.2 addGoods 类	28
4.2.3 goodsController 类	30
4.2.4 changeDetail 类	31
4.2.5 goodsDetail 类	31
4.3 支付订单管理模块	32
4.3.1 类图总览	33
4.3.2 addCart 类	33
4.3.3 Pay 类	35

4.3.4 Order 类	36
5 接口设计	38
5.1 用户接口	38
5.2 内部接口	38
5.3 外部接口	38
6 界面设计	39
6.1 首页	39
6.2 点餐页面	41
6.3 “我的”界面—顾客界面	42
6.4 “我的”界面—商家界面	43
6.5 查看我的订单	44
6.6 发布新商品	45
6.6.1 初始界面	45
6.6.2 发布成功	46
6.7 修改商品信息	47
6.7.1 初始界面	47
6.7.2 左滑后界面	48
6.7.3 修改界面	49
6.7.4 修改信息成功	50

6.7.5 删除商品界面	51
6.7.6 删除成功界面	52
6.8 请求更新模型界面	53
6.9 商家查看全部订单	54
6.10 支付成功界面	55
6.11 查看购物车界面	56
7 数据库设计	57
7.1 E-R 图	57
7.2 逻辑设计	57
7.3 表的设计	58
7.3.1 administrator 表	58
7.3.2 merchant 表	58
7.3.3 Customer 表	59
7.3.4 goods 表	59
7.3.5 order 表	60
7.3.6 sell 表	60
7.3.7 goods_order 表	61
7.3.8 consume 表	61
7.3.9 cart 表	61

8 系统出错设计	63
8.1 出错信息与补救措施	63
8.1.1 购物车为空时点击结算	63
8.1.2 商家发布商品时商品名称为空	64
8.1.3 商家发布商品时商品价格为空	64
8.1.4 商家发布商品时商品配料为空	65
8.1.5 商家发布商品时商品简介为空	66
8.1.5 商家发布商品时商品图片为空	67
8.1.6 商家发布商品时商品价格输入非法	68
8.1.7 用户登录小程序失败	69
8.2 系统维护设计	69

1 引言

1.1 设计依据

本文档旨在将软件系统功能进一步细化,说明系统总体设计的技术方案,从程序系统的设计考虑,包括系统的基本处理流程、模块划分、功能分配、接口设计、运行设计、数据结构设计和出错处理设计等内容,以向整个设计期提供关于程序系统的逻辑和数据功能实现方式的总体描述,从而作为程序详细设计或编码的基础。设计阶段将以本文档为核心文档。

本文档将原本复杂的功能拆分成不同的模块,并逐步开发健壮的系统结构,使软件系统需求能够很好的转化为未来设计,并进一步提高软件各方面的性能。本文档依据高内聚低耦合的原则,将结构分解为不同的模块,各模块分别负责自己的功能,且模块与模块之间,模块与环境之间的用关系尽可能简单。同时,在进行概要设计分析的过程中,发现并解决了系统中各个模块之间功能重合、冗余和功能短缺的部分问题,为详细设计和代码编写提供了保障。

本文档的目标读者为本软件的开发人员、测试人员和其他相关人员。

本文档根据需求分析规约中的功能建模部分整理编写。

参考《计算机软件产品开发文件编制指南(GB8567-88)》。

1.2 编写目的

《详细设计书》是项目设计的第二个阶段,这个阶段的主要任务是在《概要设计书》基础上,对概要设计中产生的功能模块进行过程描述,设计功能模块的内部细节,包括类的详细逻辑和接口及类之间的调用关系,为编写源代码提供必要的说明。

从该阶段开发正式进入软件的实际开发阶段本阶段完成系统的大致设计并明确系统的数据结构与软件结构。在软件设计阶段主要是把一个软件需求转化为软件表示的过程,这种表示只是描绘出软件的总的概貌。

本文档的目的旨在推动软件工程的规范化,使设计人员遵循统一的详细设计书写规范,节省制作文档的时间,降低系统实现的风险,做到系统设计资料的规范性与全面性,以利于系统的实现、测试、维护、版本升级等。也就是进一步细化软件设计阶段得出的软件总体概貌把它加工成在程序细节上非常接近于源程序的软件表示。

概要设计解决了软件系统总体结构设计的问题,包括整个软件系统的结构、模块划分、模块功能和模块间的联系等。详细设计则要解决如何实现各个模块的内部功能,即模块设计。具

体的说，模块设计就是要详细描述各模块中包含的类的处理逻辑及其提供接口的参数格式、请求方式。但这并不等同于系统实现阶段用具体的语言编码，它只是对实现细节作精确的描述，这样编码阶段就可以将详细设计中对功能实现的描述，直接翻译、转化为用某种程序设计语言书写的程序。详细设计的详细程度，应达到可以编写程序的水平。

1.3 参考资料

《软件工程：实践者的研究方法》 Roger S. Pressman, Bruce R. Maxim 机械工业出版社

《实用软件工程》 殷人昆 清华大学出版社

1.4 假定和约束

发布时间：2020 年 12 月 29 日

系统开发要求：Windows / Mac OS

开发条件：开发成员自备开发设备，通过微信小程序开发者工具进行项目开发

开发工具：微信小程序开发工具

数据库：云数据库

经费支持：无经费支持

设计成本：开发成员共同承担设计成本

开发效率：采用迭代开发模型，开发期间团队成员结伴开发，选择设备齐全的教室。团队相互协作，每位成员都熟悉开发过程中设计到的各项技术，保证项目的顺利推进以及每个迭代都朝着预定的目标推进并达到最高的质量。

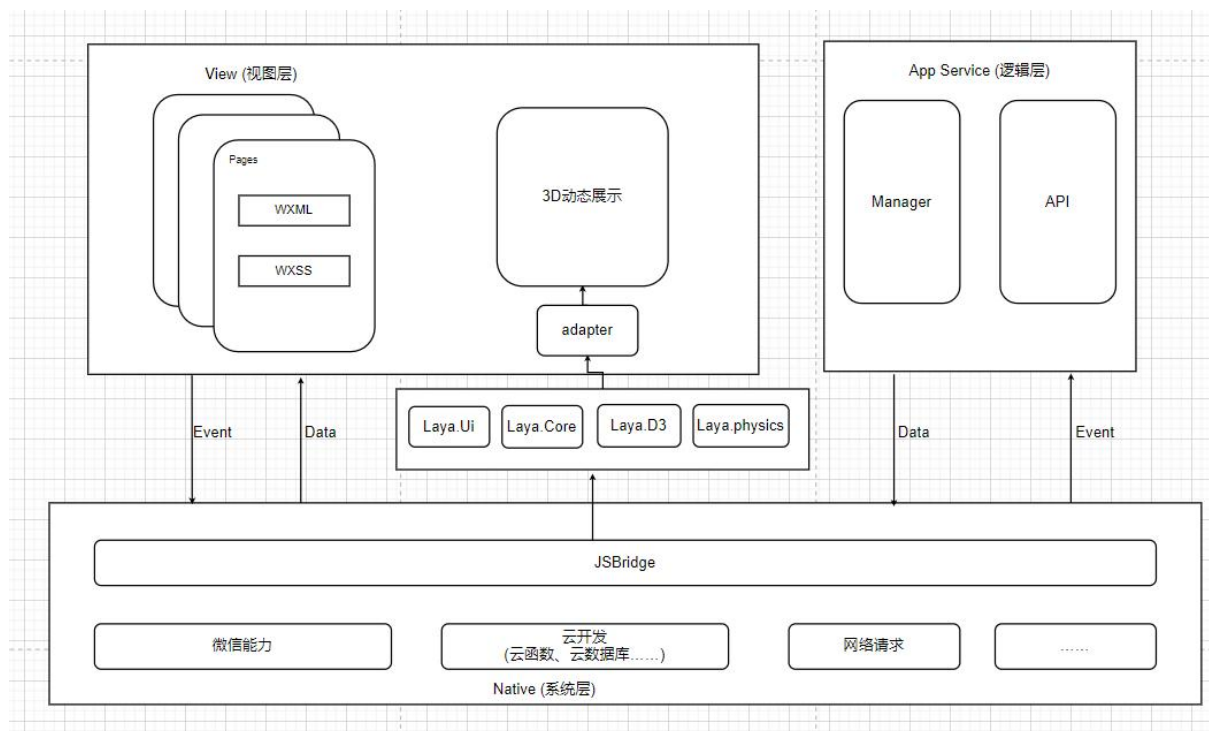
小组合作：小组合作采用线上线下结合的方式，通过线上腾讯会议以及群内讨论，线下图书馆以及教师讨论，完成项目的任务分工、实际开发、进度汇报和联合调试等，最大化的利用成员们的时间并保证开发的效率。

代码托管：为了充分践行软件工程的思想，流程化、平台化地管理我们的整个项目，我们使用 GitHub 托管我们代码，每个成员各自进行开发，开发完成部分后提交合并，产生新版本的代码。每次更新之后，最新的版本都将部署在服务器上。

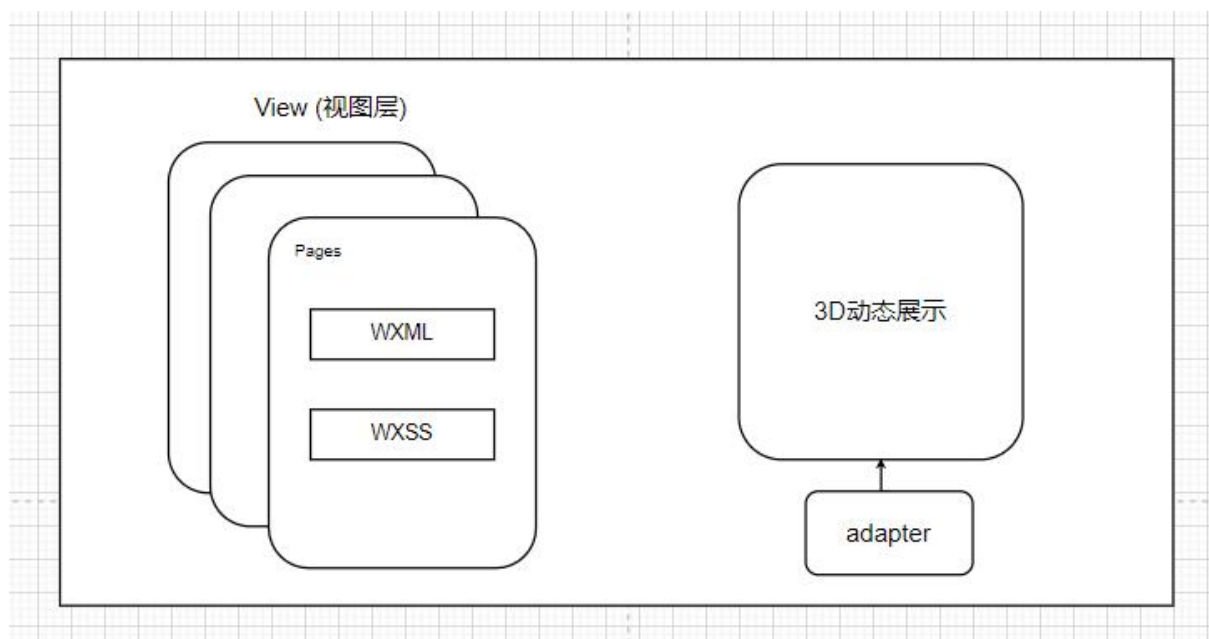
开发时限：60 天

2 系统总体架构设计

2.1 系统体系结构总概

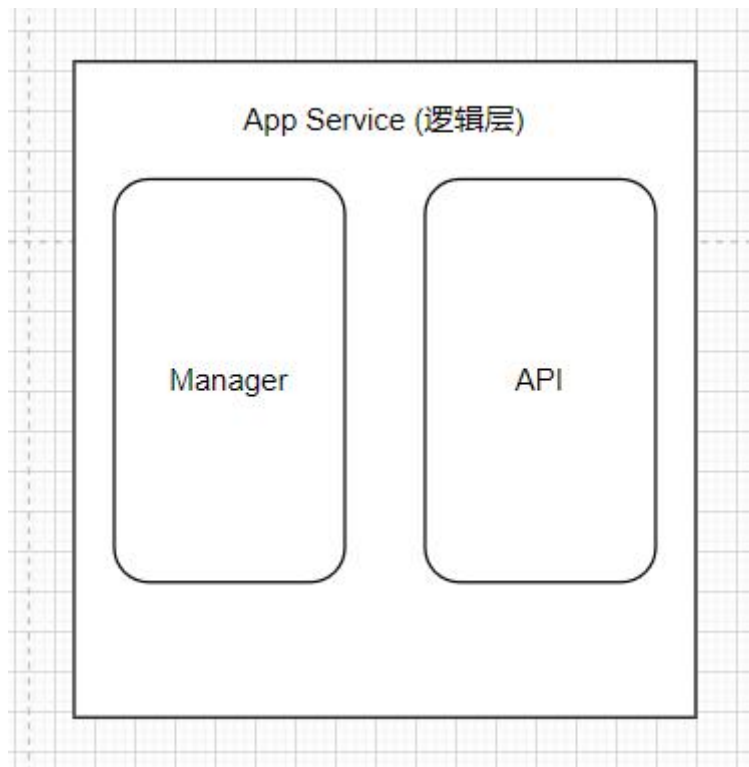


2.2 视图层



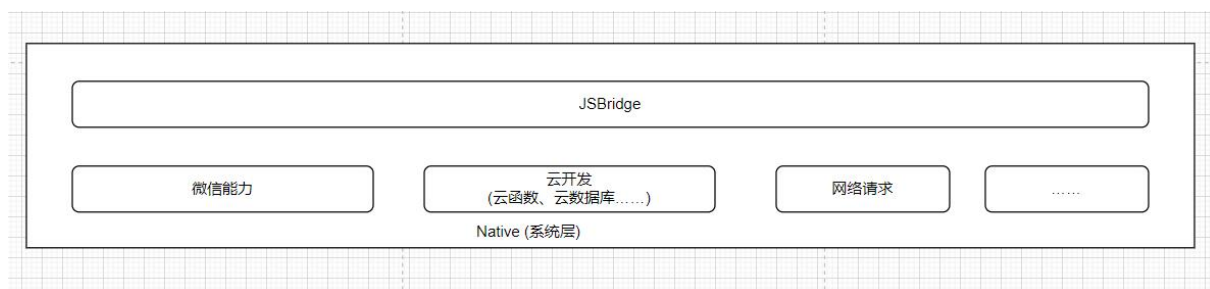
视图层负责渲染页面结构，逻辑层负责逻辑处理、数据请求、接口调用等。视图层和逻辑层分离，通过数据驱动，事件交互。

2.3 逻辑层



视图层与逻辑层通过数据和事件进行通信，逻辑层提供数据给视图层，视图层通过绑定/捕获事件发起交互让逻辑层处理。

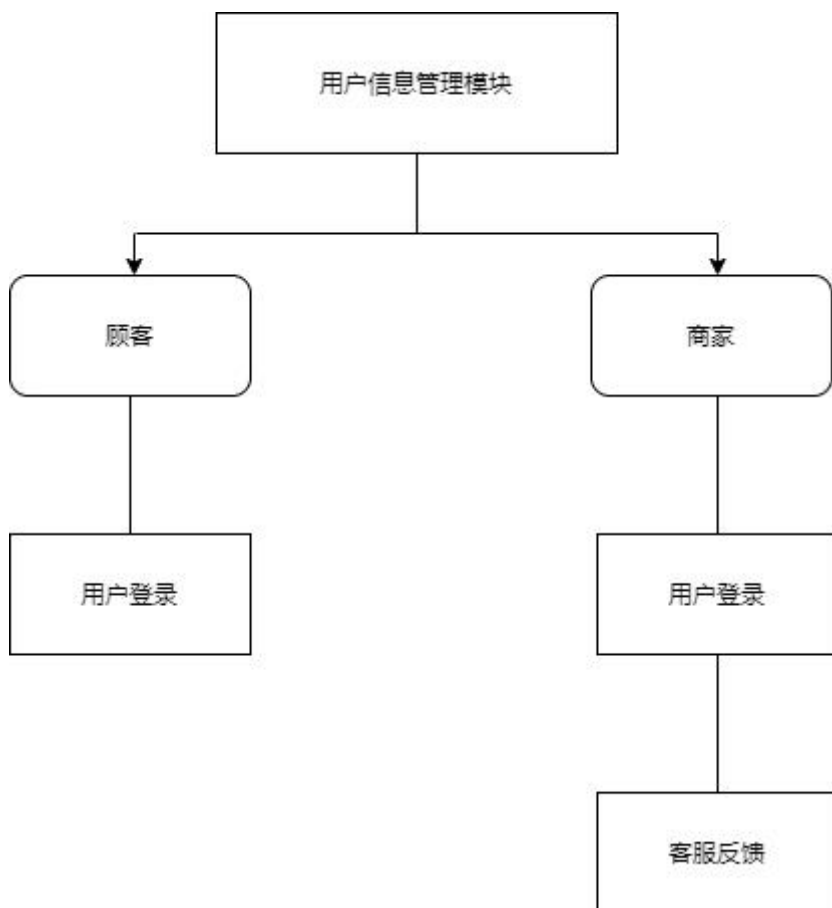
2.4 系统层



JSBridge 下架起上层开发与 Native（系统层）的桥梁，使得小程序可通过 API 使用原生的功能，部分组件为原生组件实现，从而有良好体验。实现了对底层 API 接口的调用，所以在小程序里面开发，开发者不用太多去考虑 OS 实现差异的问题，安心在上层的视图层和逻辑层进行开发即可。

3 模块设计

3.1 用户信息管理模块



3.1.1 模块描述

本模块为小程序的用户信息管理模块，进行用户的登录以及商家的认证。同时，商家也可以向客服反馈，以说明自己的需求或者请求商品模型更新。

3.1.2 模块功能

1. 用户登录
2. 客服反馈（商家）

3.1.3 模块详细设计

3.1.3.1 用户登录

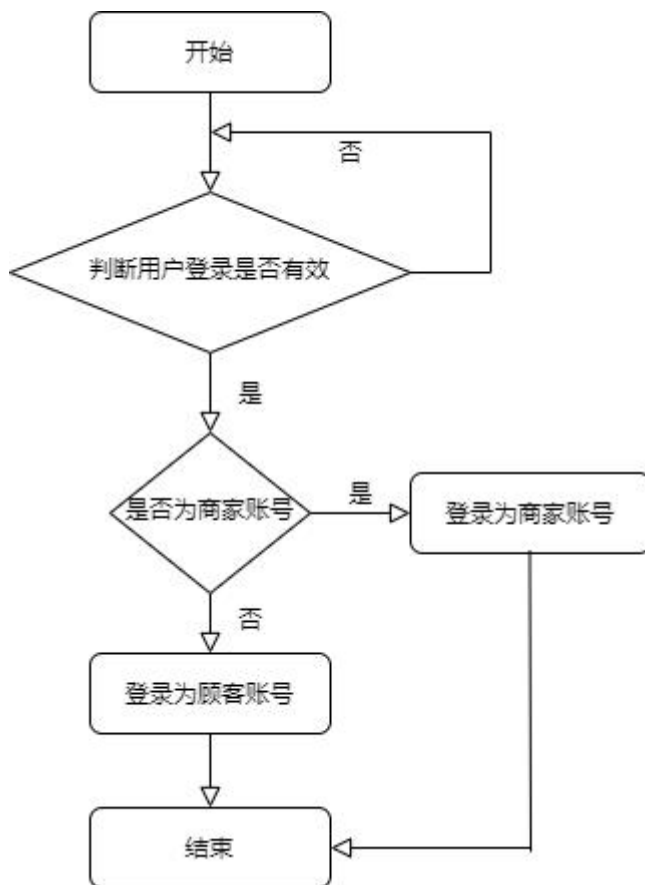
3.1.3.1.1 输入项

1. 从微信接口获取_openid

3.1.3.1.2 输出项

1. 用户昵称
2. 用户头像
3. 用户身份

3.1.3.1.3 逻辑流程



3.1.3.1.4 限制条件及出错处理

无

3.1.3.2 客服反馈

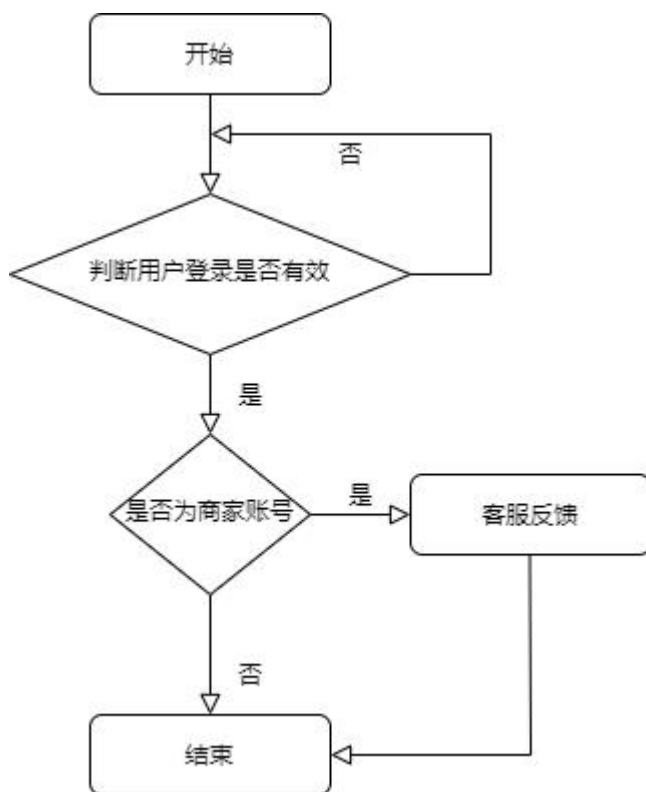
3.1.3.2.1 输入项

1. 商家反馈请求

3.1.3.2.2 输出项

1. 客服反馈消息

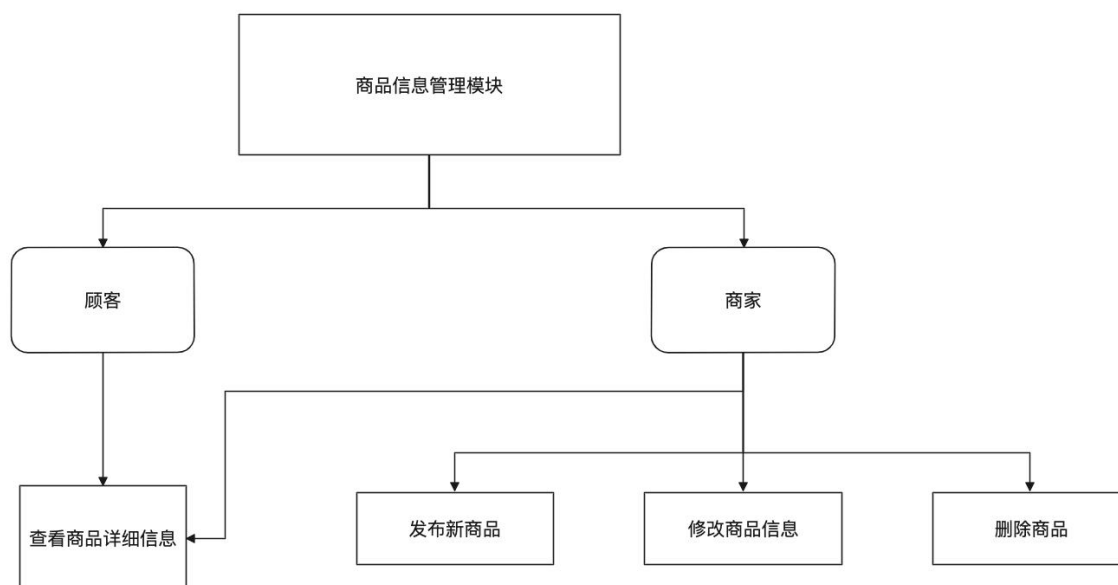
3.1.3.2.3 逻辑流程



3.1.3.2.4 限制条件及出错处理

1. 用户已登录
2. 用户为商家账户

3.2 商品信息管理模块



3.2.1 模块描述

本模块为小程序的商品信息管理模块，商家和顾客都可以查看任意商品的详细信息，商家可以对本店内已上架商品进行修改或删除，也可以发布新的商品。

3.2.2 模块功能

1. 查看商品详细信息
2. 发布新产品（商家）
3. 修改商品信息（商家）
4. 删除已上架商品（商家）

3.2.3 模块详细设计

3.2.3.1 查看商品详细信息

3.2.3.1.1 输入项

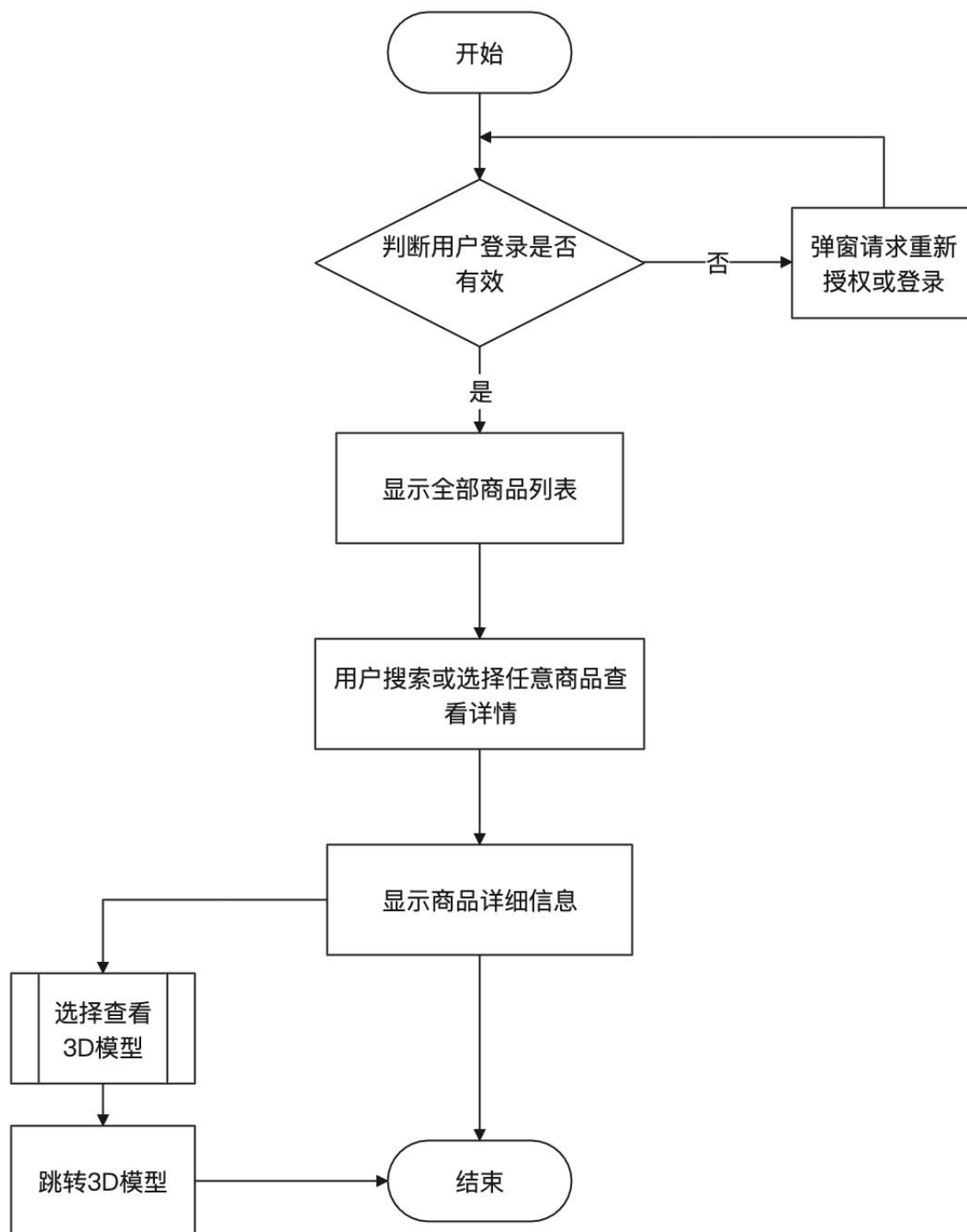
无

3.2.3.1.2 输出项

1. 商品名称

2. 商品价格
3. 商品配料成分
4. 商品简介
5. 商品图片
6. 商品 3D 模型

3.2.3.1.3 逻辑流程



3.2.3.1.4 限制条件及出错处理

1. 用户已登录

3.2.3.2 发布新商品

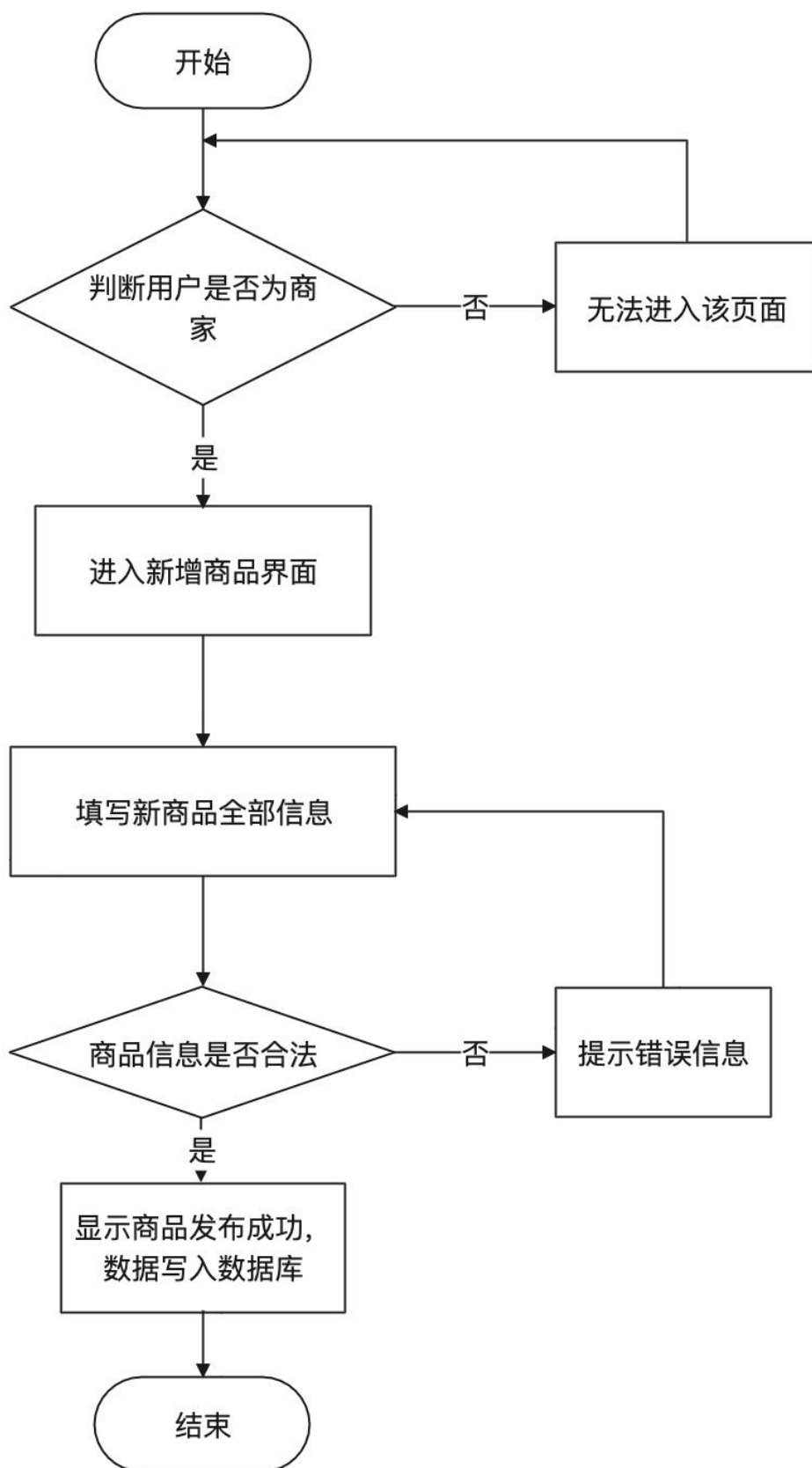
3.2.3.2.1 输入项

1. 商品名称
2. 商品价格
3. 商品配料成分
4. 商品简介
5. 商品图片

3.2.3.2.2 输出项

1. 是否发布成功
2. 如果出错的错误信息

3.2.3.2.3 逻辑流程



3.2.3.2.4 限制条件及出错处理

1. 用户已登陆且类型为[商家](#)用户，若非[商家](#)用户[无法进入此页面](#)。
2. 输入的各项信息的字符长度符合要求, 且必选项都要填入有效的信息, 若信息无效或者漏填, 弹窗提示用户并且无法提交。

3.2.3.3 修改商品信息

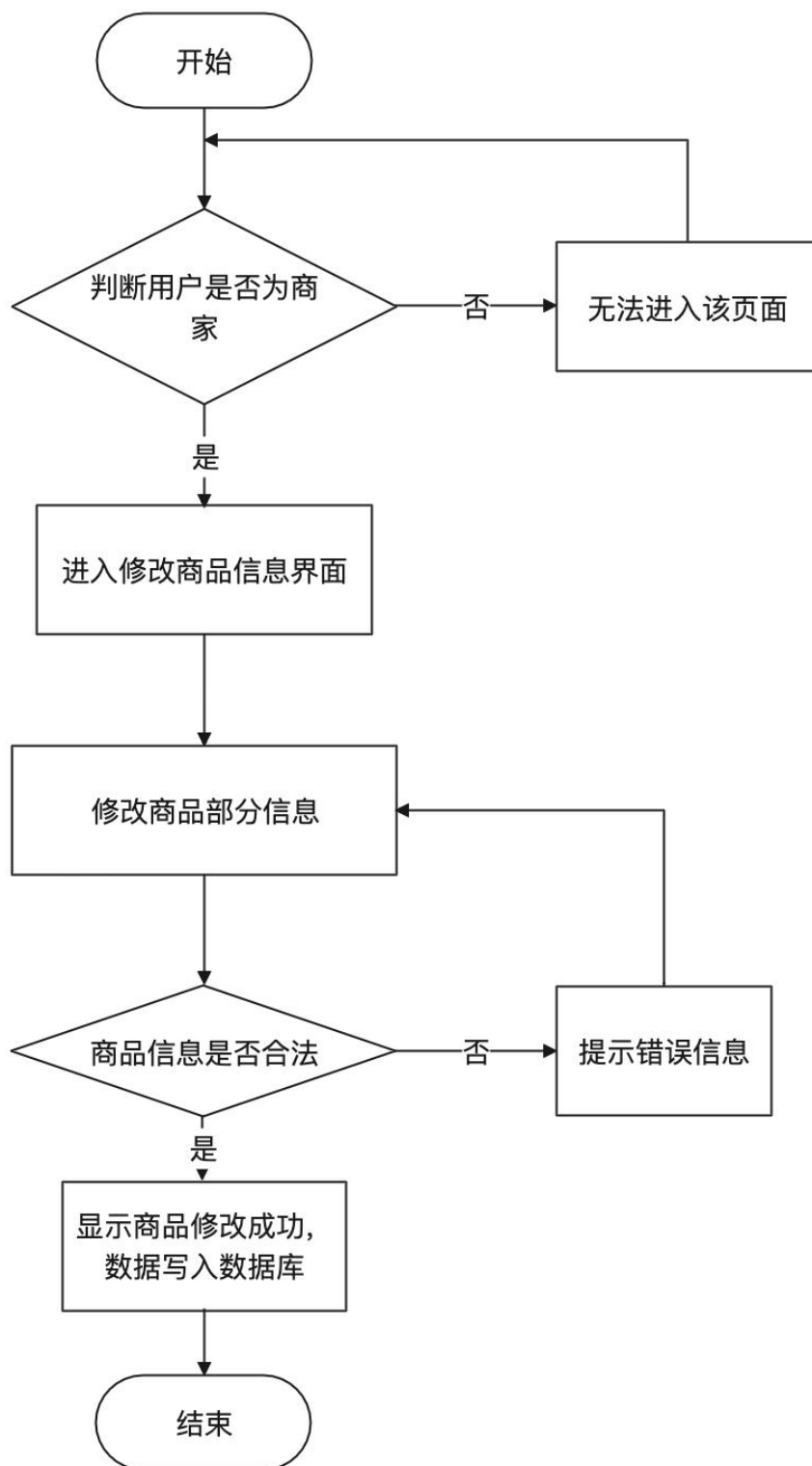
3.2.3.3.1 输入项

1. 商品名称
2. 商品价格
3. 商品配料成分
4. 商品简介
5. 商品图片

3.2.3.3.2 输出项

1. 是否修改成功
2. 如果出错的错误信息

3.2.3.3.3 逻辑流程



3.2.3.3.4 限制条件及出错处理

1. 用户已登陆且类型为商家用户，若非商家用户无法进入此页面。
2. 修改的各项信息的字符长度符合要求，且必选项都要填入有效的信息，若信息无效或者漏填，

弹窗提示用户并且无法提交。

3.2.3.4 删除商品

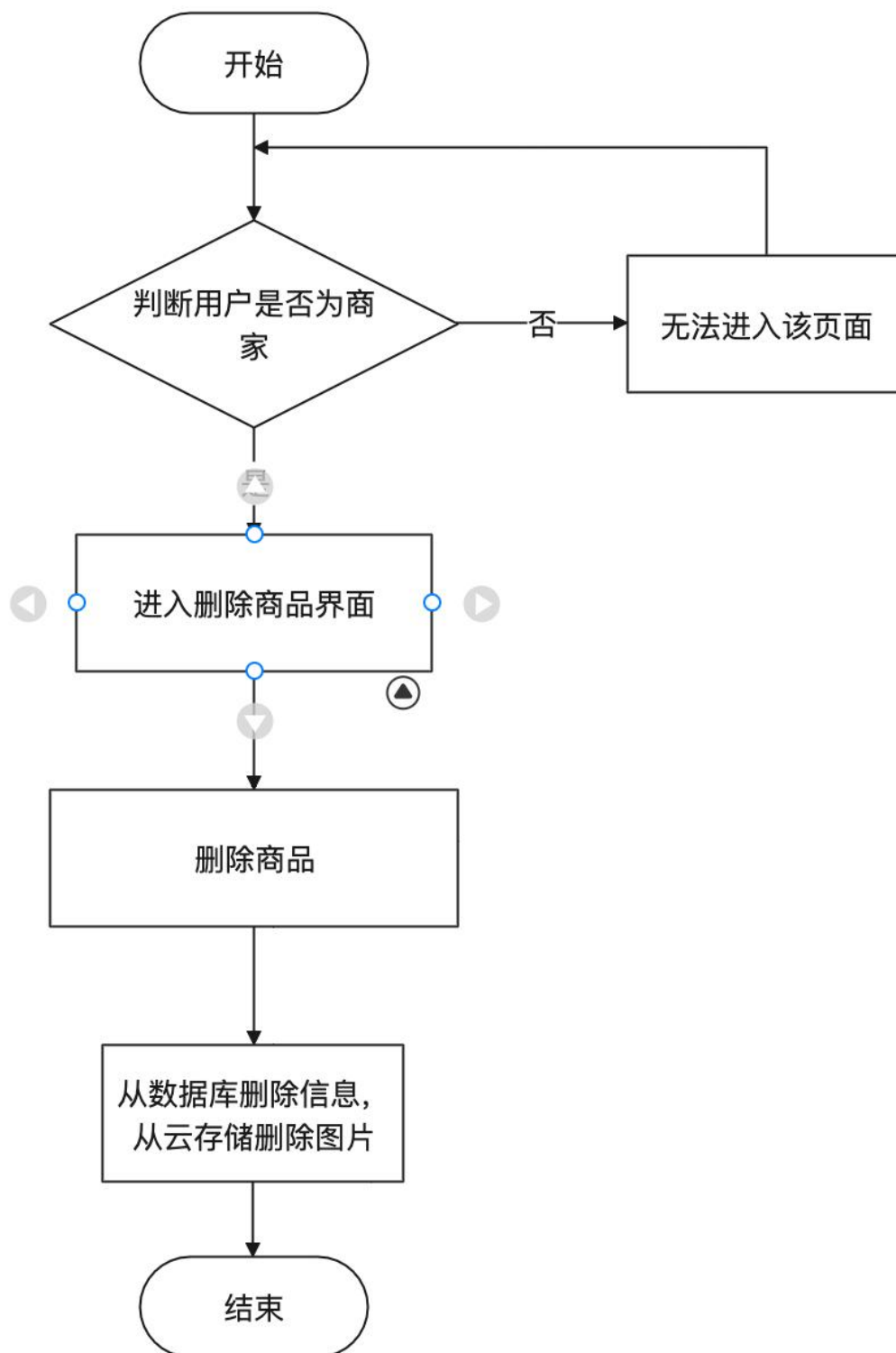
3.2.3.4.1 输入项

无

3.2.3.4.2 输出项

商品是否删除成功

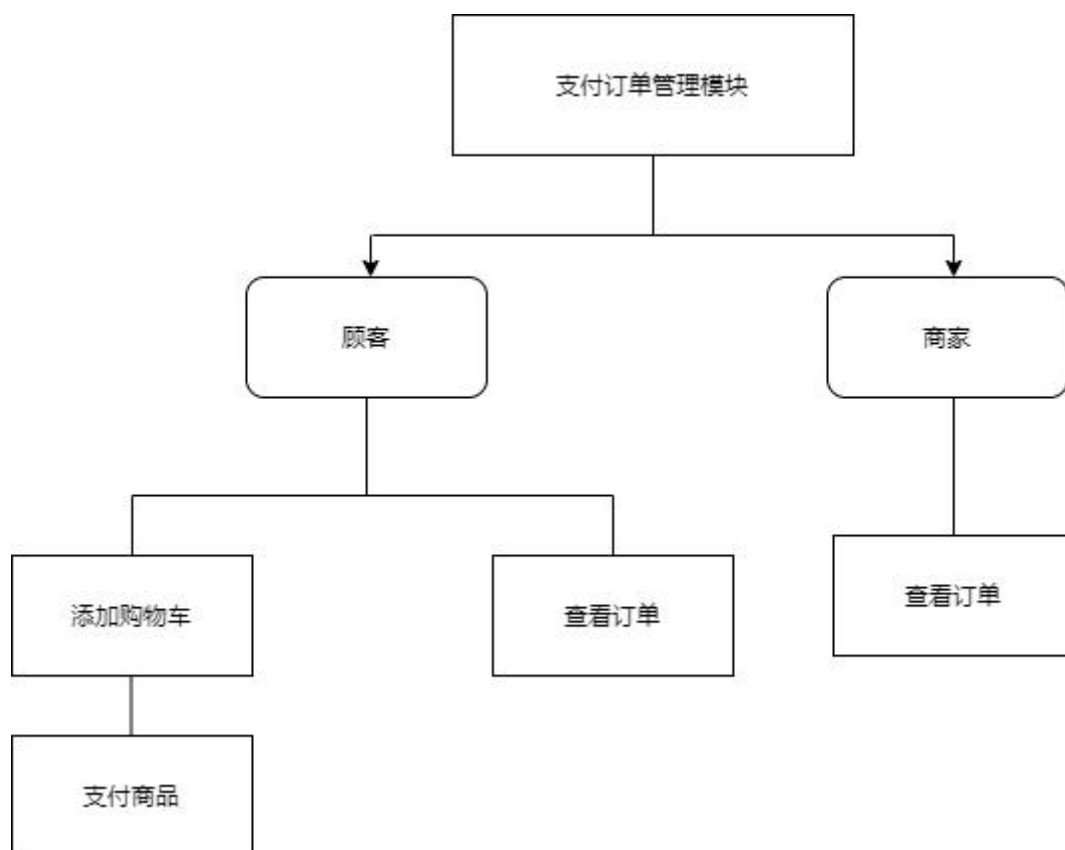
3.2.3.4.3 逻辑流程



3.2.3.4.4 限制条件及出错处理

1. 用户已登陆且类型为商家用户，若非商家用户无法进入此页面。

3.3 支付订单管理模块



3.3.1 模块描述

本模块为小程序的支付订单管理模块，进行用户购物车的添加、商品的支付以及订单的查看确认。在顾客提交订单后，商家可以查看所有顾客下的订单，以进行菜品准备。

3.3.2 模块功能

1. 添加购物车
2. 支付商品
3. 订单查看

3.3.3 模块详细设计

3.3.3.1 添加购物车

3.3.3.1.1 输入项

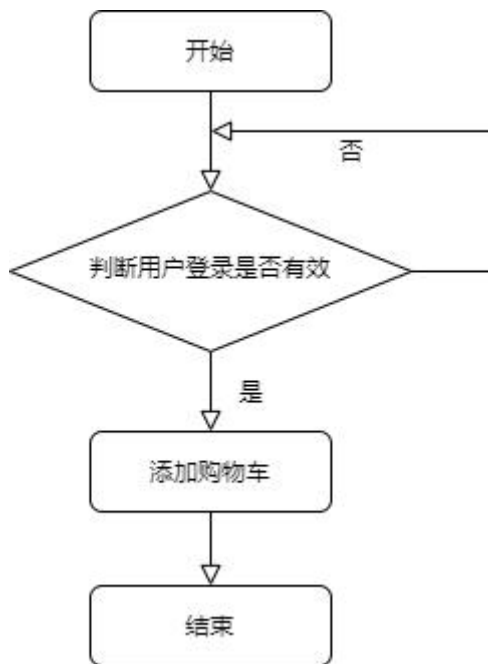
1. 商品名

2. 商品数量
3. 商品单价

3.3.3.1.2 输出项

1. 商品总价
2. 购物车列表

3.3.3.1.3 逻辑流程



3.3.3.1.4 限制条件及出错处理

1. 用户已登录

3.3.3.2 支付商品

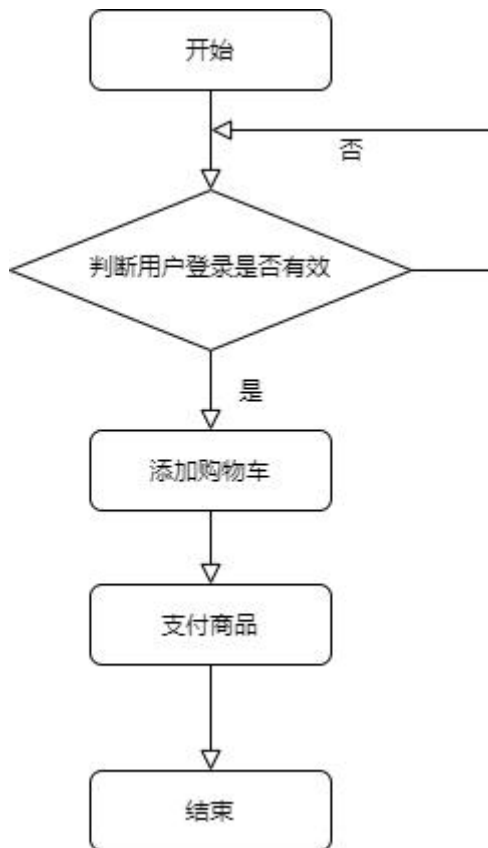
3.3.3.2.1 输入项

1. 添加到购物车的商品

3.3.3.2.2 输出项

1. 支付成功消息
2. 支付订单

3.3.3.2.3 逻辑流程



3.3.3.2.4 限制条件及出错处理

1. 用户已登录

3.3.3.3 订单查看

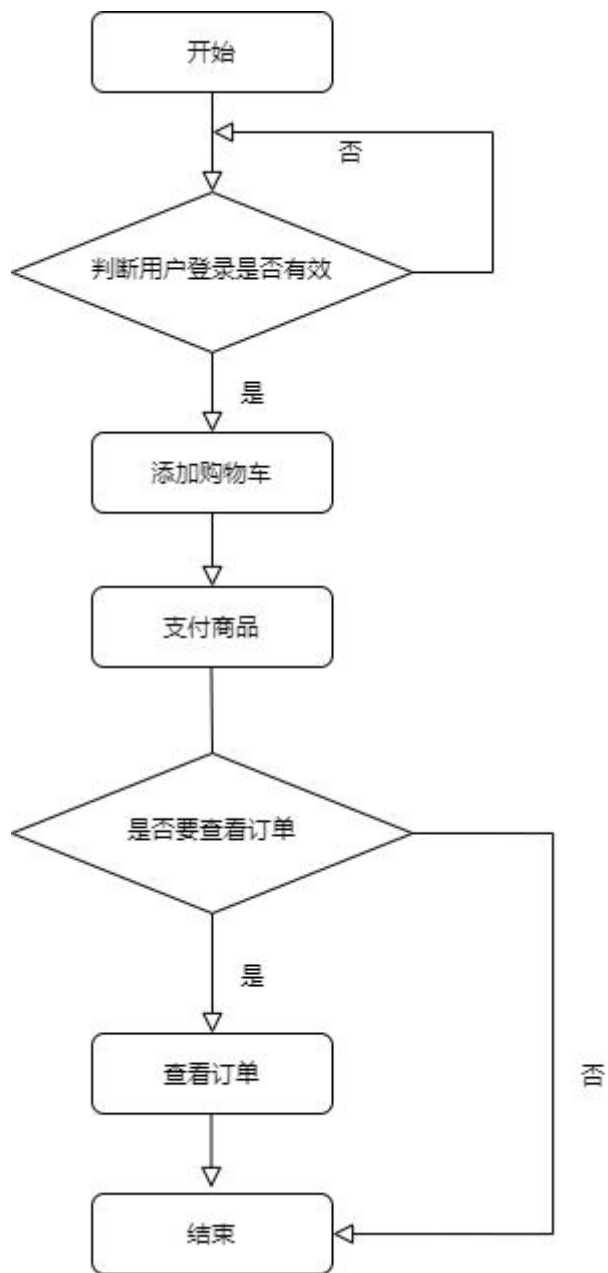
3.3.3.3.1 输入项

无

3.3.3.3.2 输出项

1. 用户昵称
2. 支付订单
3. 订单商品
4. 商品数量
5. 商品价格

3.3.3.3 逻辑流程



3.3.3.3.4 限制条件及出错处理

1. 用户已登录

4 类设计

4.1 用户信息管理模块

4.1.1 类图总览

<i>User</i>
- _openid
+ login():void

4.1.2 User 类

4.1.2.1 属性详细描述

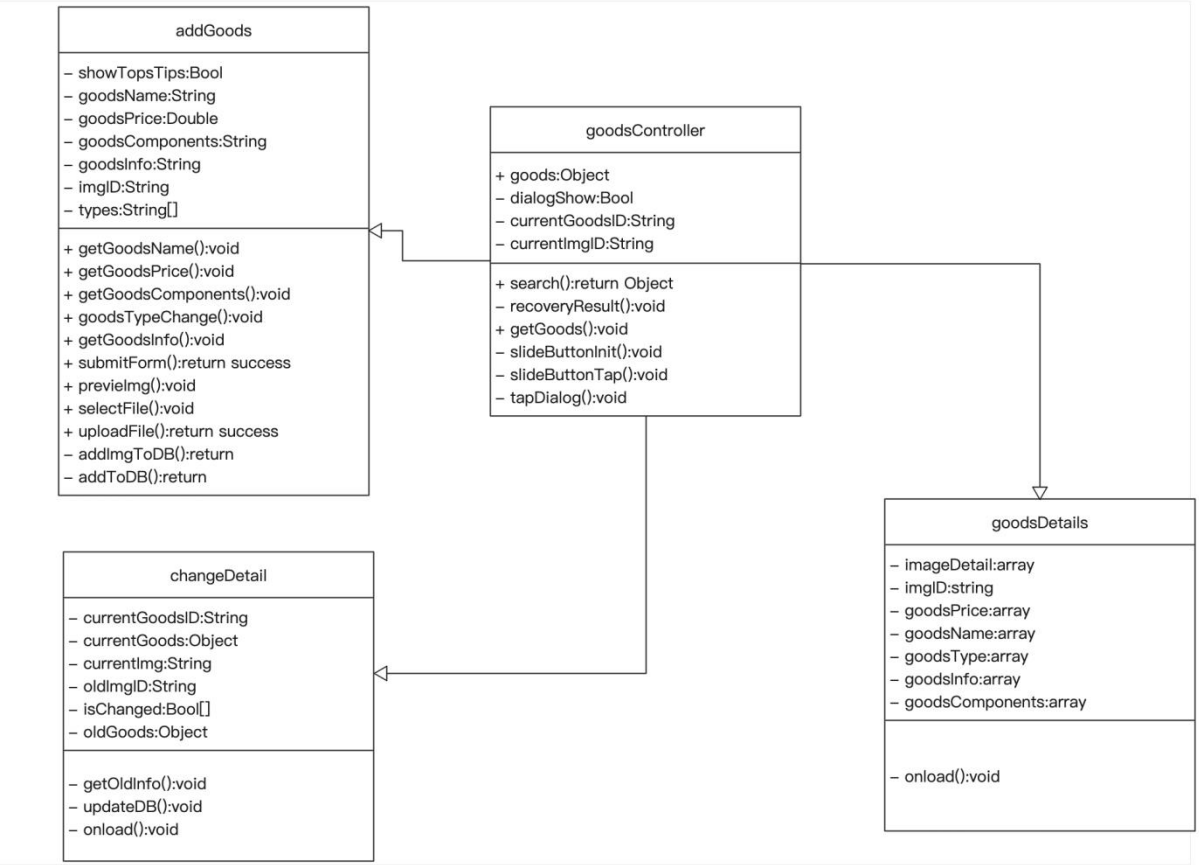
属性名称	类型	说明
_openid	string	用户唯一标识的 openid

4.1.2.2 方法描述

方法名称	说明
login()	进行用户登录

4.2 商品信息管理模块

4.2.1 类图总览



4.2.2 addGoods 类

4.2.2.1 属性详细描述

属性名称	类型	说明
showTopTips	bool	是否显示错误提示
goodsName	string	商品名称
goodsPrice	double	商品价格
goodsComponents	string	商品配料

goodsInfo	string	商品简介
imgID	string	图片 id
types	string[]	类型选择

4.2.2.2 方法描述

方法名称	说明
getGoodsName()	获取商品名称
getGoodsPrice()	获取商品价格
getGoodsComponents()	获取商品配料成分
goodsTypeChange()	获取商品类型
getGoodsInfo()	获取商品简介
submitForm()	提交表单
previewImg()	预览图片
selectFile()	选择文件
uploadFile()	上传文件
addImgToDB()	将图片上传数据库
addToDB()	将商品信息上传数据库

4.2.3 goodsController 类

4.2.3.1 属性详细描述

属性名称	类型	说明
goods	object	存储商品信息
dialogShow	bool	是否有错误提示
currentGoodsID	string	当前选择商品的 ID
currentImgID	string	当前选择商品图片的 ID

4.3.3.2 方法描述

方法名称	说明
search()	按商品名称查询商品
recoveryResult()	恢复商品列表
getGoods()	获取全部商品信息
slideButtonInit()	左滑显示
slideButtonTap()	实现左滑功能
tapDialog()	删除商品功能

4.2.4 changeDetail 类

4.2.4.1 属性详细描述

属性名称	类型	说明
currentGoodsID	string	当前商品 ID
currentGoods	object	当前商品全部信息
currentImg	string	当前商品图片
oldImgID	string	未修改的图片 ID
isChanged	bool[]	商品某一信息是否发生改变
oldGoods	object	未修改的商品全部信息

4.3.4.2 方法描述

方法名称	说明
getOldInfo()	获取修改之前的商品信息
updateDB()	将改动更新至数据库
onload()	在加载时获取传入的信息

4.2.5 goodsDetail 类

4.2.5.1 属性详细描述

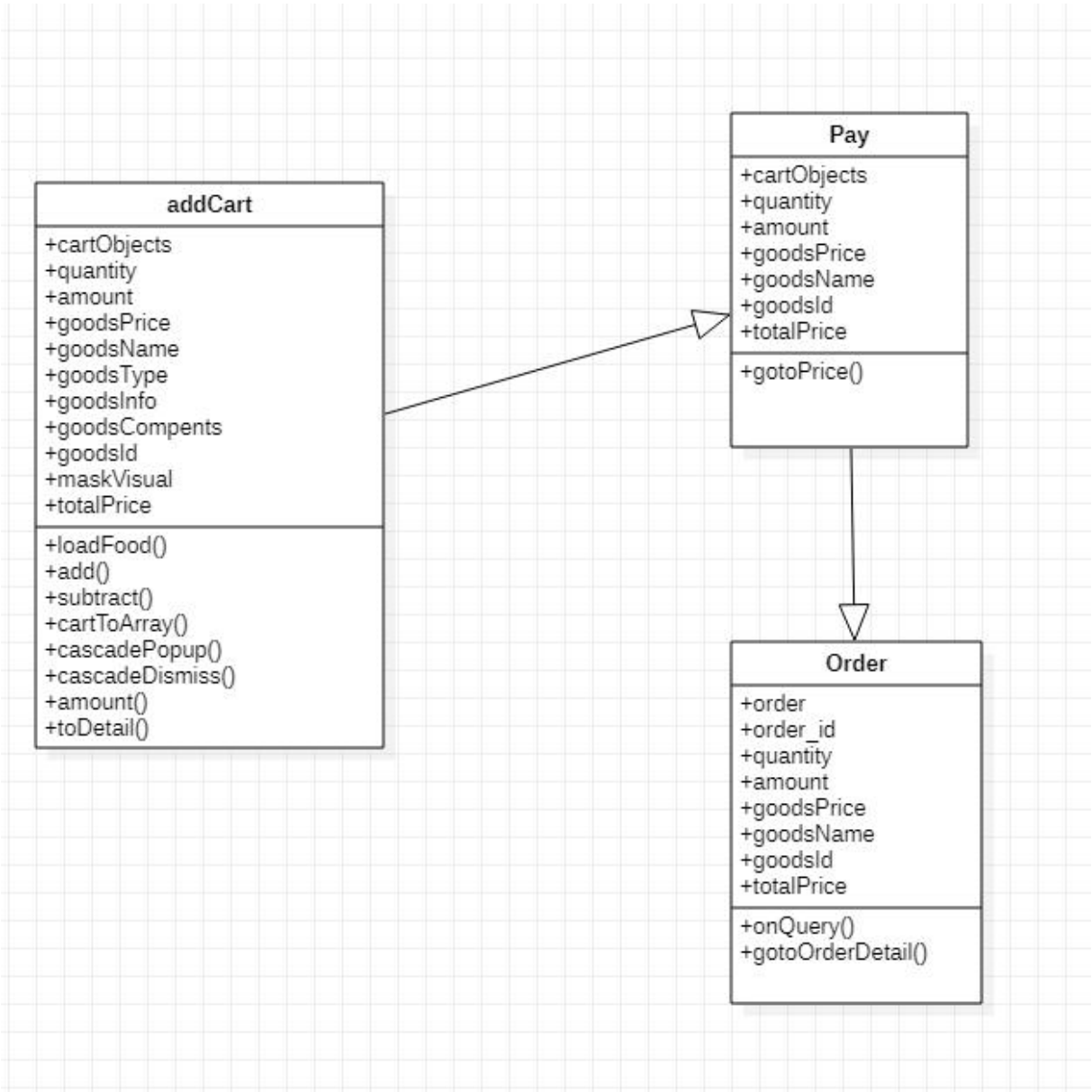
属性名称	类型	说明
imgDetail	array	图片信息
imgID	string	图片 ID
goodsPrice	array	商品价格
goodsName	array	商品名称
goodsType	array	商品类型
goodsInfo	array	商品简介
goodsComponents	array	商品配料成分

4.3.5.2 方法描述

方法名称	说明
onload()	在加载时获取全部数据并存储

4.3 支付订单管理模块

4.3.1 类图总览



4.3.2 addCart 类

4.3.2.1 属性详细描述

属性名称	类型	说明
cartObjects	object	购物车内容
quantity	array	商品价格

amount	array	商品数量
goodsPrice	array	商品价格
goodsName	array	商品名称
goodsType	array	商品类型
goodsInfo	array	商品简介
goodsComponents	array	商品配料成分
goodsId	array	商品 id
maskVisual	string	购物车开关状态
totalPrice	array	商品总价格

4.3.2.2 方法描述

方法名称	说明
loadFood	从数据库加载食品信息
add	商品加入购物车
subtract	购物车食品减一
cartToArray	转换购物车数据为数组
cascadeToggle	切换购物车开与关

cascadePopup	购物车打开动画
cascadeDismiss	购物车关闭动画
amount	返回购物车内食品数目价格
toDetail	跳转到食品详情页

4.3.3 Pay 类

4.3.3.1 属性详细描述

属性名称	类型	说明
cartObjects	object	购物车内容
quantity	array	商品价格
amount	array	商品数量
goodsPrice	array	商品价格
goodsName	array	商品名称
goodsId	array	商品 id
totalPrice	array	商品总价格

4.3.3.2 方法描述

方法名称	说明
gotoPay	点击去支付生成订单

4.3.4 Order 类

4.3.4.1 属性详细描述

属性名称	类型	说明
order	object	订单列表
order._id	array	订单号
quantity	array	商品价格
amount	array	商品数量
goodsPrice	array	商品价格
goodsName	array	商品名称
goodsId	array	商品 id
totalPrice	array	商品总价格

4.3.4.2 方法描述

方法名称	说明
onQuery	从云数据库获取订单列表
gotoOrderDetail	进入商品详情页

5 接口设计

5.1 用户接口

- 查看商品详细信息接口：用户通过点击进入商品详情页查看商品的详细信息。
- 发布新商品接口：商家可以在商品发布页发布新的商品。
- 修改商品信息接口：商家可以在商品信息修改页修改商品信息，修改完成之后调用接口更新商品信息数据。
- 删除已上架商品接口：商家可以在商品信息修改页删除已上传的商品，删除完成之后调用接口删除对应商品信息数据。
- 添加购物车接口：顾客在商品列表页添加商品至购物车。
- 支付商品接口：顾客点击支付按钮进行商品支付，支付完成之后调用接口生成订单。
- 订单查看接口：顾客可以在订单页查看自己的历史订单，商家可以在订单管理页查看所有顾客支付的订单。

5.2 内部接口

- 上传商品信息接口
- 上传商品图片接口
- 用户权限管理

5.3 外部接口

- 微信登录接口
- 微信客服反馈接口

6 界面设计

6.1 首页

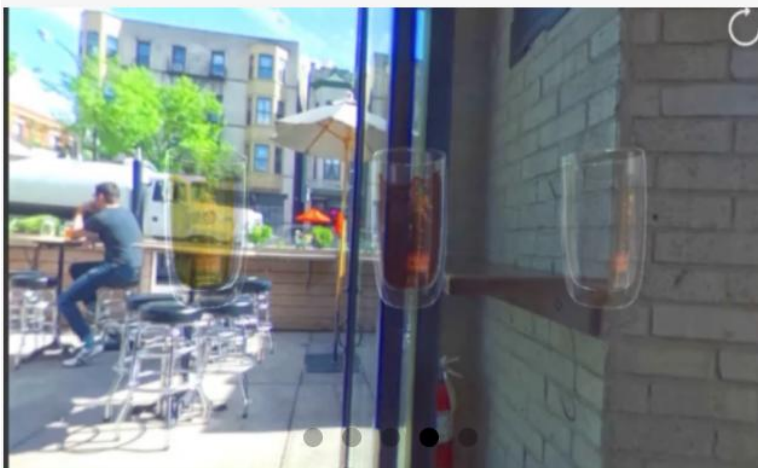


Virtual Drinks



动态展示你的美食

晃动手机仔细观赏吧！



首页



点餐



我的

6.2 点餐页面



6.3 “我的”界面—顾客界面



6.4 “我的”界面—商家界面



6.5 查看我的订单

<div><div><</div><div>我的订单</div><div><div>...</div><div>🕒</div></div></div>		
可乐350ml	等2件商品	¥ 8.00
可乐350ml	等2件商品	¥ 8.00
可乐350ml	等3件商品	¥ 26.00

6.6 发布新商品

6.6.1 初始界面

发布商品

商品信息

商品名称

请输入商品名称

商品价格

请输入商品价格

配料成分

请输入商品的配料成分

商品类型

饮品

图片上传

图片上传提示

联系管理员增加模型

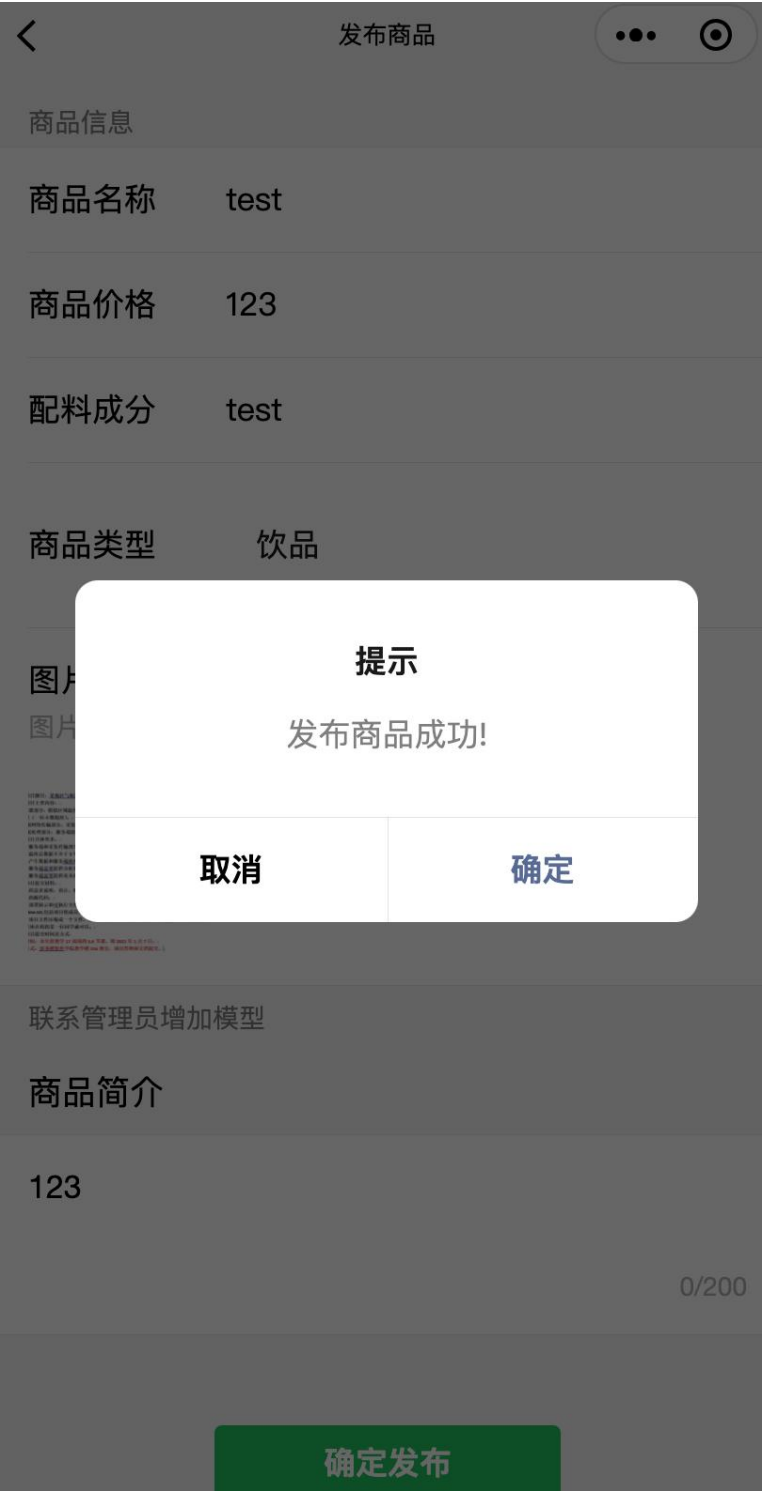
商品简介

请输入商品简介

0/200

确定发布

6.6.2 发布成功



6.7 修改商品信息

6.7.1 初始界面

<

修改商品信息

...

🎯

🔍 搜索

左滑对相应商品进行修改



樱桃可乐

¥5



可乐350ml

¥3



雪碧350ml

¥4



test

¥123

6.7.2 左滑后界面

修改商品信息

...

Q 搜索

左滑对相应商品进行修改



樱桃可乐

¥5



可乐350ml

¥3



雪碧350ml

¥4

¥123

查看

修改

删除

6.7.3 修改界面

[illegible]

6.7.4 修改信息成功



6.7.5 删除商品界面



6.7.6 删除成功界面



6.8 请求更新模型界面



6.9 商家查看全部订单

<

我的订单

...

📷

Athenahrl

可乐350ml

等2件商品

414px × 808px
¥ 8.00

Athenahrl

可乐350ml

等2件商品

¥ 8.00

Athenahrl

可乐350ml

等3件商品

¥ 26.00

6.10 支付成功界面

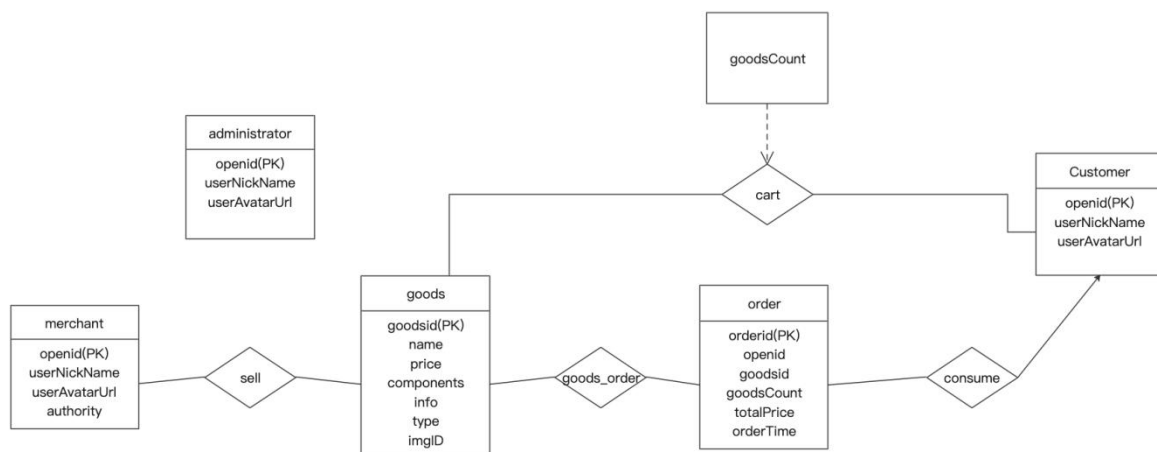


6.11 查看购物车界面



7 数据库设计

7.1 E-R 图



7.2 逻辑设计

E-R 图向关系模型的转换

关系模型的逻辑结构是一组关系模式的集合。E-R 图则是由实体，实体的属性和实体间的联系三个要素组成。所以将 E-R 图转换为关系模型实际上就是要将实体，实体的属性和实体间的联系转换为关系模式。转换原则如下。

1. 实体类型的转换:一个实体型转换成一个关系模式。实体的属性就是关系的属性，实体的码就是关系的码。
2. 联系类型的转换，根据不同的情况做不同的处理。
 - a. 一个 1:1 的联系可以转换为一个独立的关系模式，也可以与任意一端对应的关系模式合并。如果转换为一个独立的关系模式，则与该联系相连的各实体的码以及联系本身的属性均转换为关系的属性，每个实体的码均是该关系的候选码。如果与某一端实体对应的关系模式合并，则需要在该关系模式的属性中加入另一个关系模式的码和联系本身的属性。
 - b. 一个 1:N 的联系可以转换为一个独立的关系模式，也可以与 N 端对应的关系模式合并。如果转换为一个独立的关系模式，则与该联系相连的各实体的码以及联系本身的属性均转换为关系的属性，而关系的码为 N 端实体的码。

- c. 一个 M：N 联系转换为一个关系模式。与该联系相连的各实体的码为各实体码的组合。
- d. 三个或三个以上实体间的一个多元联系可以转换成一个关系模式。与该多元联系相连的各实体的码以及联系本身的属性均转换为关系的属性，而关系的码为各实体码的组合。
- e. 具有相同码的关系模式可合并。

7.3 表的设计

7.3.1 administrator 表

字段名	数据类型	长度	说明	备注
openid	String	50	用户唯一标识符	PK UN
userNickName	String	20	用户昵称	UN
userAvatarUrl	String	100	用户头像图片路径	UN

7.3.2 merchant 表

字段名	数据类型	长度	说明	备注
openid	String	50	用户唯一标识符	PK UN
userNickName	String	20	用户昵称	UN
userAvatarUrl	String	100	用户头像图片路径	UN
authority	Bool		是否有高级权限	

7.3.3 Customer 表

字段名	数据类型	长度	说明	备注
openid	String	50	用户唯一标识符	PK UN
userNickName	String	20	用户昵称	UN
userAvatarUrl	String	100	用户头像图片路径	UN

7.3.4 goods 表

字段名	数据类型	长度	说明	备注
goodsid	String	50	商品唯一标识符	PK UN
name	String	20	商品名称	UN
price	Double		商品价格	UN
components	String	80	商品配料成分	UN
info	String	100	商品简介	
type	String	10	商品类型	UN
imgID	String	100	商品图片路径	UN

7.3.5 order 表

字段名	数据类型	长度	说明	备注
orderid	String	50	订单唯一标识符	PK UN
openid	String	50	下单的顾客 id	UN
goodsid	String	50	商品 id	UN
goodsCount	Int		商品数量	UN
totalPrice	Double		订单总金额	UN
orderTime	Date		订单时间	UN

7.3.6 sell 表

字段名	数据类型	长度	说明	备注
openid	String	50	商家 id	PK UN FK 参照 merchant 表属性
goodsid	String	50	商品 id	PK UN FK 参照 goods 表属 性

7.3.7 goods_order 表

字段名	数据类型	长度	说明	备注
orderid	String	50	订单 id	PK UN FK 参照 order 表属性
goodsid	String	50	商品 id	PK UN FK 参照 goods 表属性

7.3.8 consume 表

字段名	数据类型	长度	说明	备注
orderid	String	50	订单 id	PK UN FK 参照 order 表属性
openid	String	50	顾客 id	PK UN FK 参照 customer 表属性

7.3.9 cart 表

字段名	数据类型	长度	说明	备注
-----	------	----	----	----

openid	String	50	顾客 id	PK UN FK 参照 customer 表属性
goodsid	String	50	商品 id	PK UN FK 参照 goods 表属 性

8 系统出错设计

8.1 出错信息与补救措施

8.1.1 购物车为空时点击结算

出错信息：购物车为空，用户点击“结算”。

补救措施：顶部红色提示框进行提示，取消结算操作。



8.1.2 商家发布商品时商品名称为空

出错信息：商品名称为空，商家点击“确定发布”。

补救措施：顶部红色提示框进行提示，取消发布操作。

<

发布商品

...

📷

商品类型

商品

请输入商品名字!

图片上传

图片上传提示

+

联系管理员增加模型

商品简介

请输入商品简介

0/200

确定发布

8.1.3 商家发布商品时商品价格为空

出错信息：商品价格为空，商家点击“确定发布”。

补救措施：顶部红色提示框进行提示，取消发布操作。

<

发布商品

...

📷

请输入商品单价!

图片上传

图片上传提示

+

联系管理员增加模型

商品简介

请输入商品简介

0/200

确定发布

8.1.4 商家发布商品时商品配料为空

出错信息：商品配料成分为空，商家点击“确定发布”。

补救措施：顶部红色提示框进行提示，取消发布操作。

发布商品

请输入配料成分!

图片上传

图片上传提示

联系管理员增加模型

商品简介

请输入商品简介

0/200

确定发布

8.1.5 商家发布商品时商品简介为空

出错信息：商品简介为空，商家点击“确定发布”。

补救措施：顶部红色提示框进行提示，取消发布操作。

发布商品

请输入配料成分!

图片上传

图片上传提示

联系管理员增加模型

商品简介

请输入商品简介

0/200

确定发布

8.1.5 商家发布商品时商品图片为空

出错信息：商品图片为空或图片未完全上传，商家点击“确定发布”。

补救措施：顶部红色提示框进行提示，取消发布操作。

补救措施：顶部红色提示框进行提示，取消发布操作。

员独立地在不同的分支上进行开发，有效避免多人开发导致程序版本控制的混乱，最后再 merge 到 master 分支上。

2. 通过云开发控制台监控用户访问，避免非法访问与攻击。
3. 定期对云数据库的数据进行增量备份，即使云数据库发生意外事件导致数据丢失，也可以从其他地区的数据中心将数据恢复。