



**QG工作室项目报告**

学 院 信息工程学院

专 业 电子信息类

班 级 10班

组 别 工业软件-后台组

姓 名 古永智

学 号 3122002448

2023年 4月 30日

广东工业大学计算机学院制

目录

**[广东工业大学计算机学院制 1](#_Toc25416)**

**[1. 项目简介 3](#_Toc17040)**

[1.1. 购物网项目功能介绍 3](#_Toc25466)

[1.2. 购物网项目技术介绍 4](#_Toc26922)

**[2. 设计思路 5](#_Toc28871)**

[2.1. 整体思路 5](#_Toc23064)

[2.2. 前端设计思路 5](#_Toc32535)

[2.3. 后端设计思路 5](#_Toc2289)

**[3. 详细功能设计 5](#_Toc28112)**

[3.1. 数据库设计 6](#_Toc12573)

[3.2. 后端代码框架设计 6](#_Toc26089)

[3.3. 项目功能分类设计 9](#_Toc3617)

**[4. 程序测试 19](#_Toc14736)**

[4.1. 页面展示 19](#_Toc9004)

[4.2. 表单展示 24](#_Toc23057)

**[批量购买表单 25](#_Toc29666)**

**[5. 项目亮点 26](#_Toc24033)**

**[6. 心得体会 27](#_Toc24770)**

# 项目简介

* 1. 购物网项目功能介绍

项目有用户的基本功能：用户登录、用户注册功能，登录的用户还可以退出登录。

项目分为四个权限：游客权限、普通用户权限、店铺管理员权限、网站管理员权限。

①游客：只能进行商品的浏览和店铺的浏览，可以进入店铺查看对应店铺的商品；

还可以进行店铺、商品搜索。

②普通用户：除游客权限的功能外，普通用户可以对商品进行立即购买、加入购物车、查看评论操作；在“购物车”标签中，用户可以进行批量购买、批量移出购物车；用户购买商品后，可以在“我的订单”标签中查看自己订单的状态，也有对订单进行确认售后、评论、售后处理的权限；此外，用户还可以在“个人信息”标签中修改个人信息，如果用户想要自己开店，也可以在“个人信息”标签中填写“我想开店”的表单，提交申请后可以在“我的申请”标签中看到自己提交过的申请，也可以看到申请的当前状态；若申请审核通过，即可成为一名店铺管理员。

③店铺管理员：成为店铺管理员，当然也能像普通用户一样正常购物。除此之外，店铺管理员可以在“个人信息”标签中看到自己店铺的信息，亦可修改自己店铺信息。店铺管理员新增了“我的店铺”和“店铺订单”的标签，在“我的店铺”标签中，店铺管理员可以申请上架本店商品，申请审核通过后，商品会自动上架到“商品”标签中，还可以在“我的店铺”标签中管理自己的商品，对商品的编辑和对商品进行下架操作；在“店铺订单”标签中，店铺管理员可以看到本店铺接到的所有订单，查看订单的基本信息，对订单进行发货和处理售后操作。

④网站管理员：用户无法通过自己提交申请而成为网站管理员，网站管理员需要程序员在后台自己设置而成为，除了可以查看和修改个人信息的“个人信息”标签之外，网站管理员有“商品”和“申请审核”两个标签，在“商品”标签中，网站管理员可以查看商品评论，并有对恶意评论删除的功能，还可以对商品进行强制下架，删除评论和强制下架商品之后都会通知用户或店铺管理员；在“申请审核”标签中，店铺管理员可以处理用户提交的开店申请或店铺管理员提交的上架商品申请，审核通过或不通过都会通知到用户，若开店申请通过，用户成为店铺管理员，上架商品申请通过的话，商品可以被所有人看到。

* 1. 购物网项目技术介绍

总体：Javaweb、javase

①前端：Vue、JavaScript、html、JQuery、axios

②后端：小组作业中的JDBCUtils、连接池、BasicServlet、三层架构【表现层（Servlet）、业务管理层（Service）、数据访问层（DAO）】

③环境：tomcat8.5、数据库MySql5.7、运行环境jdk1.8、IDEA2022.3.2、不是maven项目

# 设计思路

* 1. 整体思路

先简单地搭好前端页面的框架，使功能可视化，这样写起来更加清晰，将前端分为六个页面：登录页面、注册页面、游客权限页面、普通用户页面、店铺管理员页面和网站管理员页面，按照先易后难的逻辑来写代码；搭好了前端框架，再分析每个页面都有哪些功能，根据功能创建所需要的类，根据功能来创建数据库，做好这些准备工作。

* 1. 前端设计思路

前端视图框架使用的是Element-UI的组件，使用到了分类标签、表单、表格、按钮等。

数据方面使用了Vue对象，通过axios发送HTTP请求，使用POST方法，用axios简洁的API方便地处理请求和相应数据，在Vue对象中写前端需要用到的方法和数据，前端数据都来自于后端调取数据库相应给前端。

* 1. 后端设计思路

后端整体采用三层框架，即表现层（Servlet）、业务管理层（Service）、数据访问层（DAO）。表现层负责与用户交互，展示数据和接收用户输入，业务逻辑层负责处理业务逻辑，数据访问层负责与数据库交互，执行SQL语句来查询、插入、更新和删除数据。

# 详细功能设计

详细功能可以分为数据库的设计，后端代码框架设计，项目功能分类设计。

* 1. 数据库设计

创建了一个数据库“qg\_case”，共有8张表，分别为：

①用户表“tb\_user”：其中有用户id、用户名、用户密码、用户头像、用户地址、用户电话、用户身份/权限和用户店铺id共八个列。

②商品表“tb\_goods”：其中有商品id、商品名、商品库存、商品状态（上架/未上架）、商品月销量、商品图片、商品所在店铺id、商品介绍和商品价格九个列。

③店铺表“tb\_store”：其中有店铺id、店铺名、店铺logo、店铺发货地址和该店铺管理员id共五个列。

④购物车表“tb\_cart”：其中有用户id、商品id共两个列。

⑤订单表“tb\_order”：其中有订单id、订单状态（未发货/待收货/待评价/售后/已完成）、收货地址（买家地址）、发货地址（店铺地址）、买家id、购买时间、商品id、商品图片、商品名和商品对应的店铺id共十个列。

⑥申请表“tb\_application”：其中有申请id、申请图片（店铺logo/商品图片）、申请介绍（店铺介绍/商品介绍）、申请状态（审核中/审核通过/审核未通过）、用户id、申请类型（开店铺/上架商品）、用户名、申请名字（店铺名/商品名）、申请日期、申请地址（发货地址）、和商品id（上架商品申请时用）共十一个列。

⑦评价表“tb\_comment”：其中有商品id、评价评分、评价内容、评价id和用户id共五个列。

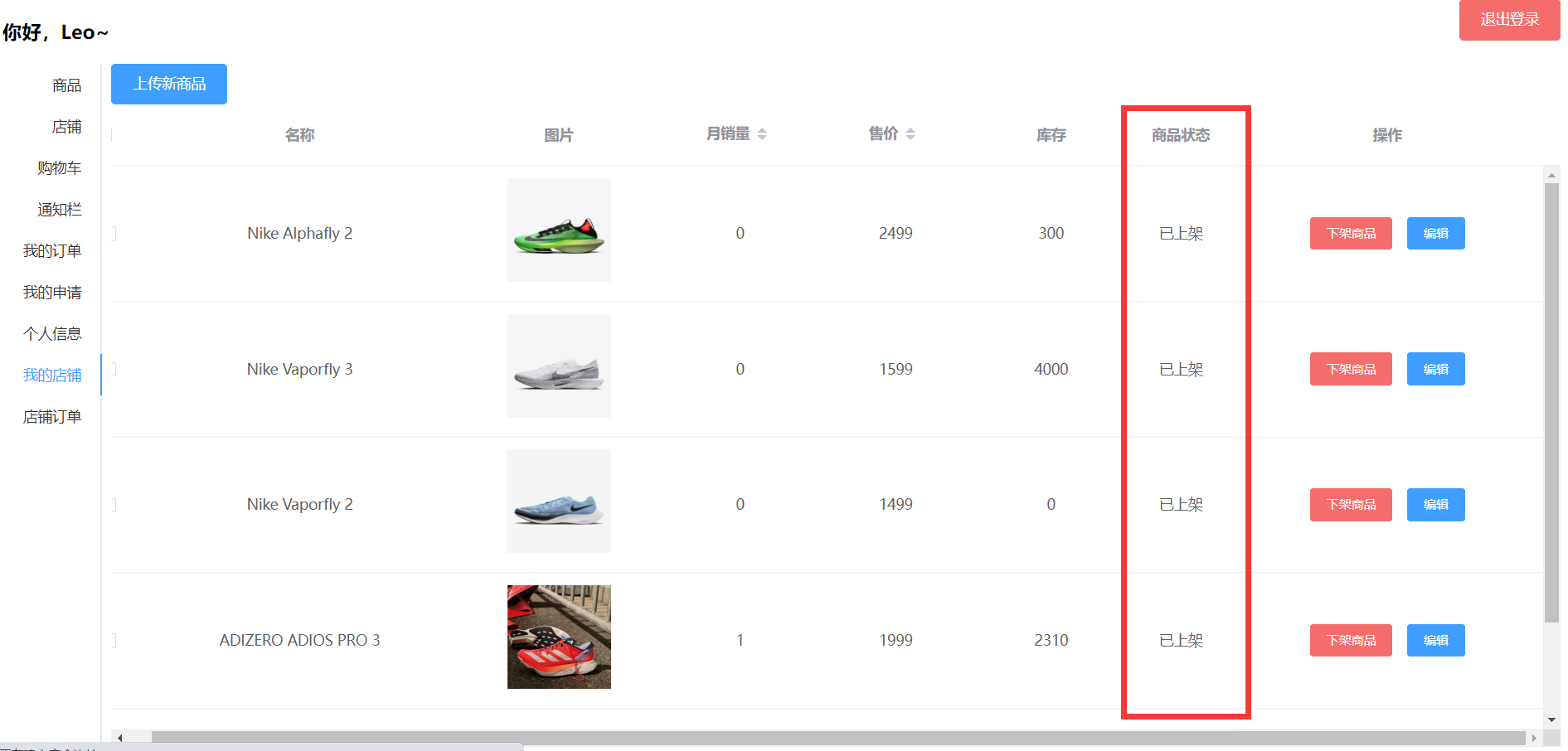
⑧通知表“tb\_notice”：其中有通知时间、通知内容和接收通知的用户id共三个列。

* 1. 后端代码框架设计

整体采用三层框架来设计后端代码，大的文件夹分成java包、resources包和webapp包。

Java包中又可以分为dao包、pojo包、service包、util包、和web包。

①dao包中写访问数据的对象，有一个通用的BasicDao，可用于对各表的查询、修改操作，在BasicDao的基础之上，实现一到两张表对应一个DAO，更好地完成功能。

②pojo包中写java类，对应数据库中表的属性，用来接收数据库表的一条记录，实现数据库与后端的数据传输和接收，每个类都有构造函数、getter/setter方法，重写toString方法，有的类还写了获取逻辑视图要用到的字符串，例如：Goods类中，有一个getGoodsStateStr的方法，它将goodsState属性转化为对应的字符串，在JSON.toJSONString(goods)方法中可以获取到goodsStateStr属性的值，如goodsState为1，goodsStateStr就为“上架”，若goodsState为2，goodsStateStr就为“未上架”，goodsStateStr可以传输到前端页面中，在店铺管理员“我的店铺”标签中看到商品的状态不是数字1或2，而是可以直观地看到是否上架。如下图：

③service包有不同java类对应的service接口，之所以要使用到接口，是因为[它允许不同的类实现相同的接口，从而可以在运行时动态地替换实现。这样，我们就可以在不改变代码的情况下更换实现，使代码更加灵活和可维护](https://www.zhihu.com/question/20111251" \t "https://www.bing.com/_blank)。[Service层的作用是将业务逻辑与数据访问分离，使代码更加清晰和可维护](https://www.zhihu.com/question/431911268" \t "https://www.bing.com/_blank)，它可以组织Sql语句，并调用对应的XxxDao来完成综合需求。

④util包主要是用来放后端代码需要用到的工具类，大部分为之前训练营小组考核的作业，例如JDBCUtils、连接池和CRUDUtils，这里的CRUDUtils我改成了BasicDao，相关代码基本一致。JDBCUtils工具类主要用于获取连接和释放资源（关闭资源），主要包括静态代码块初始化，getConnection获取连接的方法和close关闭资源的方法，获取连接的方法使用到了连接池；[数据库连接池的目的是实现连接的高效、安全的复用。它在内部维护一定数量的数据库连接，并对外提供获取和返回连接的接口。当应用程序需要访问数据库时，它可以从连接池中获取一个空闲的连接，使用完毕后再将其返回给连接池](https://zhuanlan.zhihu.com/p/435416052" \t "https://www.bing.com/_blank)，它主要的特点是允许应用程序重复使用一个现有的数据库连接，它允许应用程序重复使用一个现有的数据库连接，而不是重新建立一个。在getConnection()方法中，获取连接逻辑如下：首先使用synchronized关键字对lock对象进行同步，以确保多线程环境下的安全性。然后，检查连接池中是否有空闲的连接。如果连接池为空，则使用DriverManager.getConnection()方法创建一个新的数据库连接并返回。否则，从连接池中取出一个空闲的连接并返回。在releaseConnection方法中，就是把连接放回连接池中。utils包下还有一个用于随机生成验证码图片的工具类，这个不是自己写的，是用现成的，它可以从所输入的字母和数字中随机搭配选择4位生成验证码图片。

⑤web包主要是写三次框架web层的代码，负责处理HTTP请求和响应，web层将Web应用程序与底层的业务逻辑分离，使代码更加清晰和可维护。通常会调用Service层来处理业务逻辑，并根据Service层返回的结果生成HTTP响应。web包下写不同的servlet类，用于接收客户端发送的HTTP请求，调用业务逻辑处理请求，并生成HTTP响应返回给客户端。一个Java类基本上都对应一个特定的servlet类，这些servlet类都继承训练营小组作业完成的BaseServlet类，可以让业务相关的servlet都来继承BaseServlet，而让BaseServlet来继承HttpServlet，并重写里面的service方法。于是，每次业务相关的servlet被执行时，都会执行BaseServlet里面的代码，在BaseServlet里面实现业务servlet相关的方法。

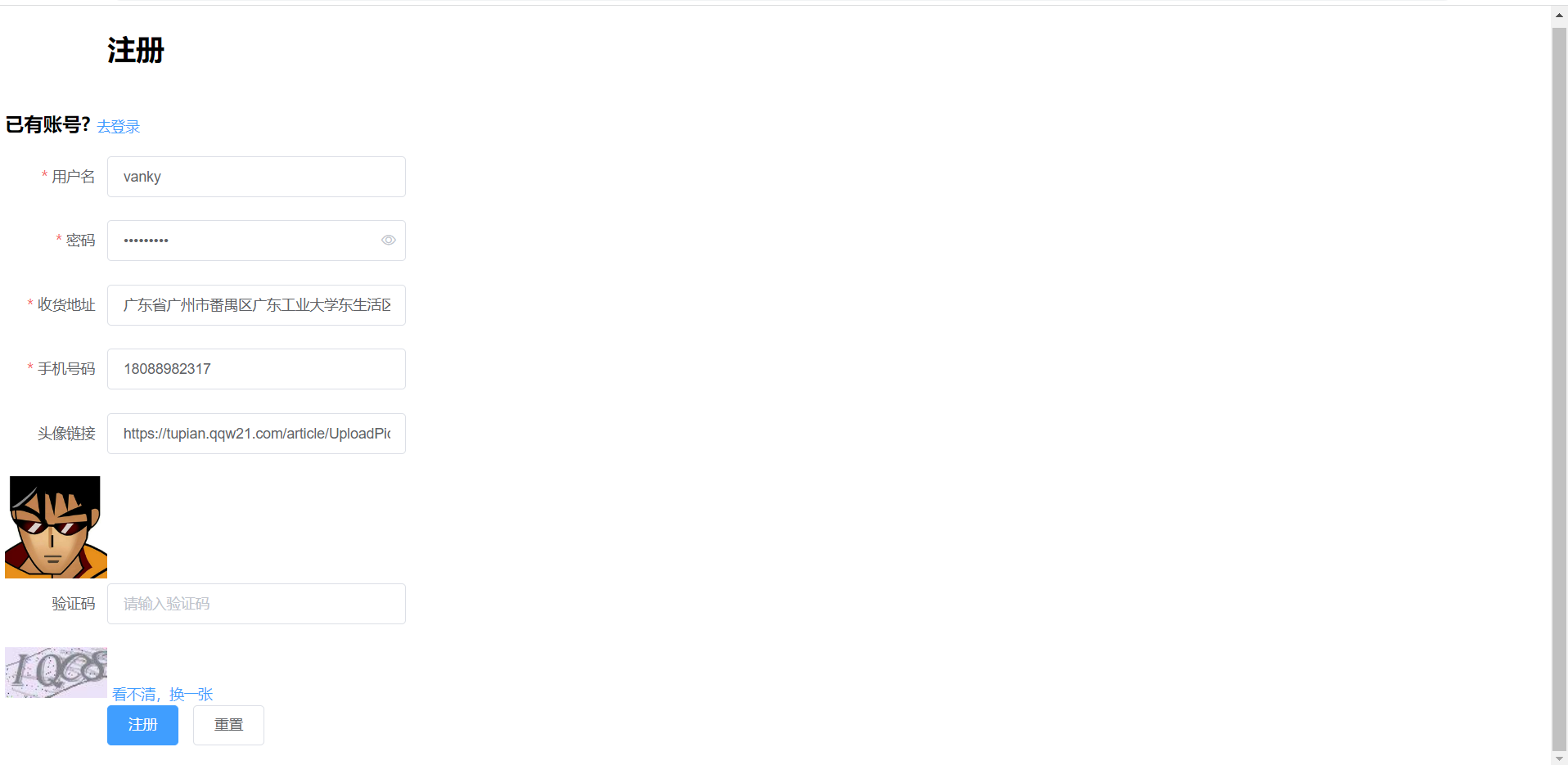
resources包用于存储数据库连接的信息，里面包含一个mysql.properties文件，存储数据库user、password、url、driver，user和password信息用于登录数据库；url用于指定数据库连接信息的字符串，包含了数据库服务器的地址、端口号、数据库名称以及其他连接参数；driver信息用于指定JDBC驱动程序，程序就可以使用指定的驱动程序连接到MySQL数据库。这个文件中的数据在JDBCUtils静态代码块进行初始化的时候用到。

webapp包中主要存储了和前端相关的文件，在本次项目中，使用到了Element-UI的组件，需要引入Element-UI的包，还有axios和Vue的插件包，还有WEN-INF包下的web.xml配置文件以及前端html页面文件。

①web.xml文件是Java Web应用程序的部署描述符，描述了应用程序的类、资源和配置，以及Web服务器如何使用它们来处理Web请求。

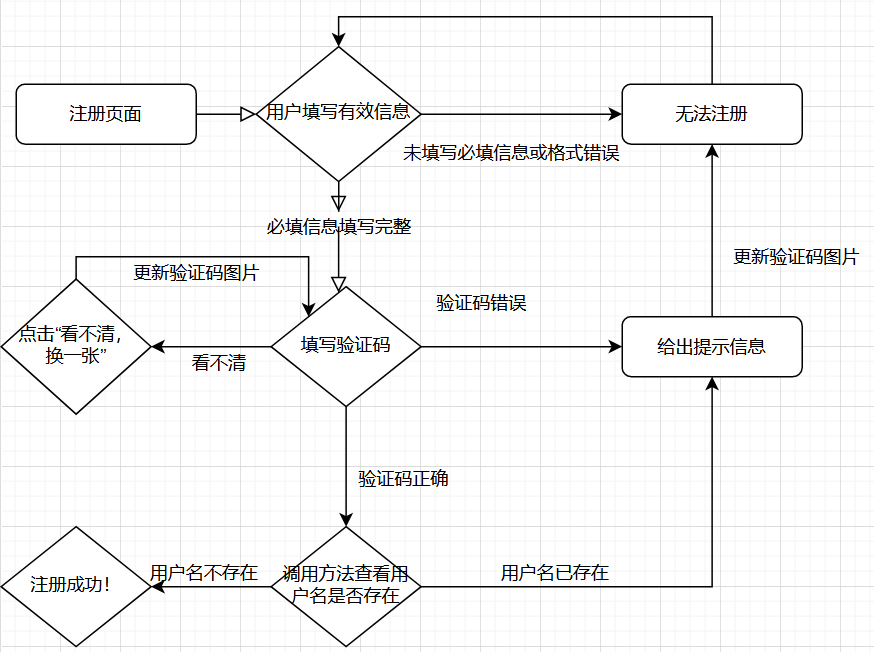
②前端html文件显示使用到了Element-UI的组件；axios异步框架，对原生的AJAX进行封装；使用了Vue前端框架，免除原生JavaScript中的DOM操作，简化书写。

* 1. 项目功能分类设计
     1. 注册功能



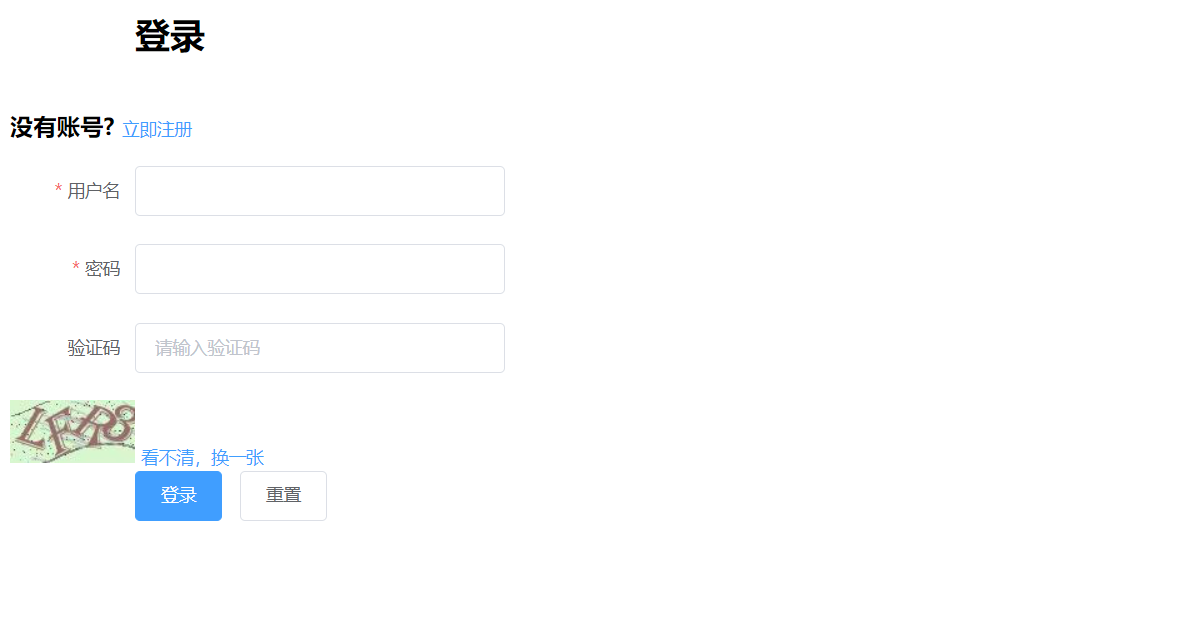
注册页面图

若用户没有账号，可以在登录页面点击“立即注册”前往注册页面，首先用户需要填写个人有效信息，其中用户名、密码、收货地址、电话号码为必填项，且都有字数的限制，前端代码会自动检测填写信息是否有效，若必填项未填写正确则无法进行注册，上传头像时，只能支持图片链接上传图片，上传链接时可以看到链接是否有效。填好信息后需要根据验证码图片输入正确的验证码，若验证码图片看不清楚，可以点击“看不清，换一张”更新验证码图片，点击注册时，前端会向后端发送输入的验证码信息，让接收到的验证码与后端生成的验证码作比较，若填写验证码与图片不匹配，则会给前端发送错误信息，前端向用户给出提示信息“验证码错误”，无法注册，且后端会更新验证码图片让用户重新填写；若验证码正确，后端先调用查找方法，查找数据库中是否存在与前端发送过来的用户名一样的数据，若存在，则向前端发送错误信息，前端给用户提示信息“用户名已存在”，无法注册；若不存在，后端执行添加用户数据到数据库的操作，给前端发送成功的信息，前端给用户提示“注册成功”。

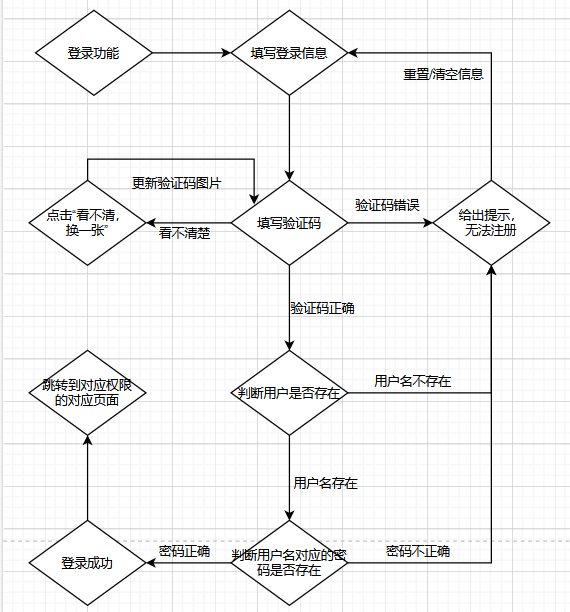


用户注册流程图

* + 1. 登录功能

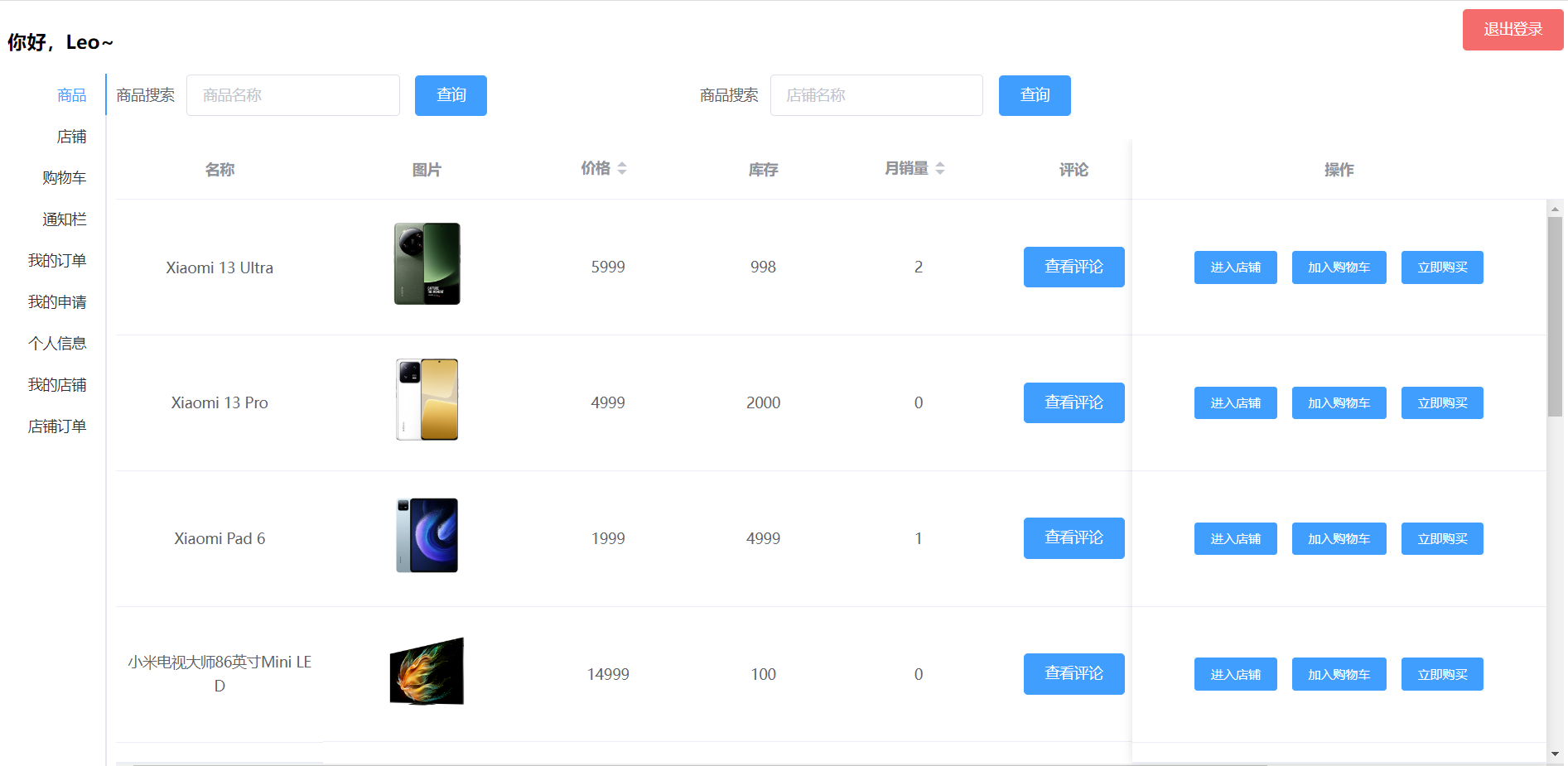


用户可以从游客页面点击“去登陆”跳转到登录页面，用户需要填写自己的用户名和密码，还有验证码。点击登录时，前端给后端发送用户填写的信息，后端首先判断填写的验证码是否与后端生成的验证码图片中的验证码匹配，如果验证码错误，后端向前端发送错误信息，前端给用户提示信息“验证码错误”，无法进行登录；若验证码正确，后端会先调用方法查看用户输入的用户名是否存在，若不存在，后端向前端发送错误信息，前端给用户提示信息“用户名存在”，无法进行登录；若用户名也存在，则调用方法查看用户名对应的密码是否与数据库中的匹配，若匹配，则后端向前端发送成功信息，用户登录成功，然后跳转到用户对应权限的页面中。



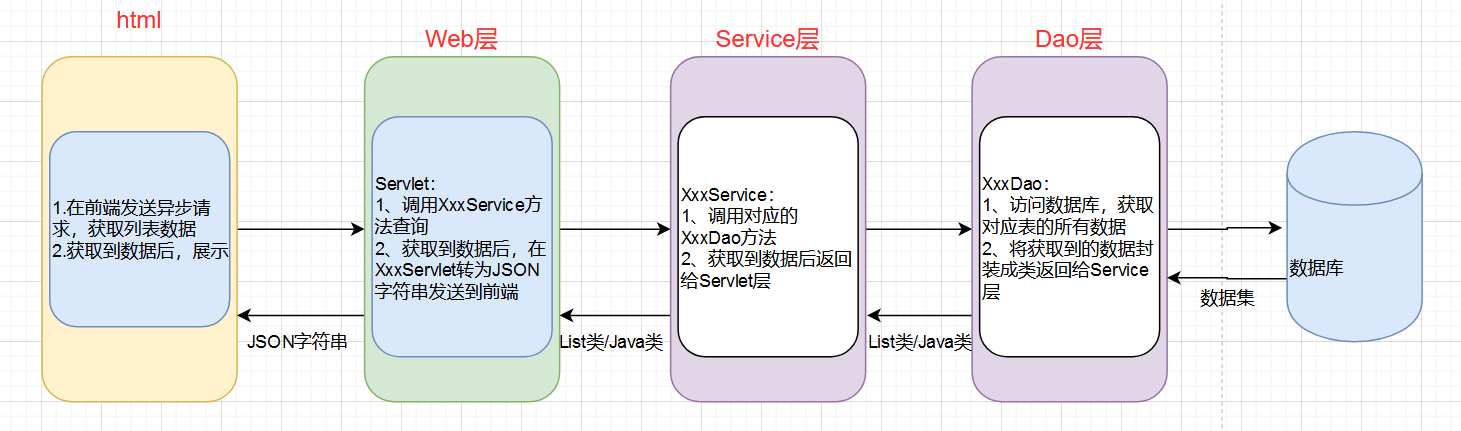
登录页面流程图

* + 1. 无条件查询的功能



商品展示图

无条件查询即在页面加载完成后，用户无序进行任何操作，自动加载数据的功能，例如各用户权限中“商品”标签中的商品展示，及一些权限中的查看所有店铺的功能。流程如下：html首先向后端发送异步请求，后端直接调用XxxService中的方法，XxxService调用XxxDao方法，XxxDao中编写sql语句，对数据进行查询，对返回的结果集进行封装，封装成List集合或者对应的Java类，并将结果先返回给XxxService，然后XxxService返回给XxxServlet，在XxxServlet将结果封装成JSON字符集，发送到前端，html获取到数据后，存到对应的数组中，并在页面上展示给用户。



无条件查询流程图

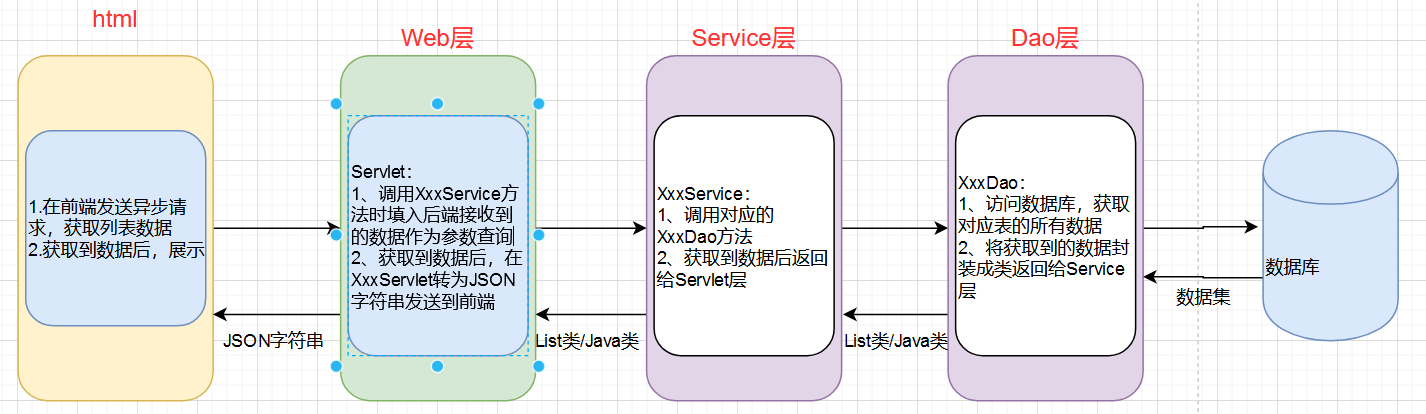
* + 1. 有条件查询功能

除了无条件查询功能，剩下的查询功能就是有条件查询，必须根据用户特定的需求或特定的信息，给出特定的查询结果，并展示到页面中。例如：展示用户的购物车信息，我的订单信息，店铺订单信息，个人信息，申请信息，通知栏信息，还有许多详细的功能。

①用商品名查找商品时，前端给后端发送用户填写的表单信息，后端先在Servlet将接收到的字符串前后加上“%”号，发送到Service层，再发送到Dao层，再Dao层对数据库进行模糊查询，将结果封装返回。

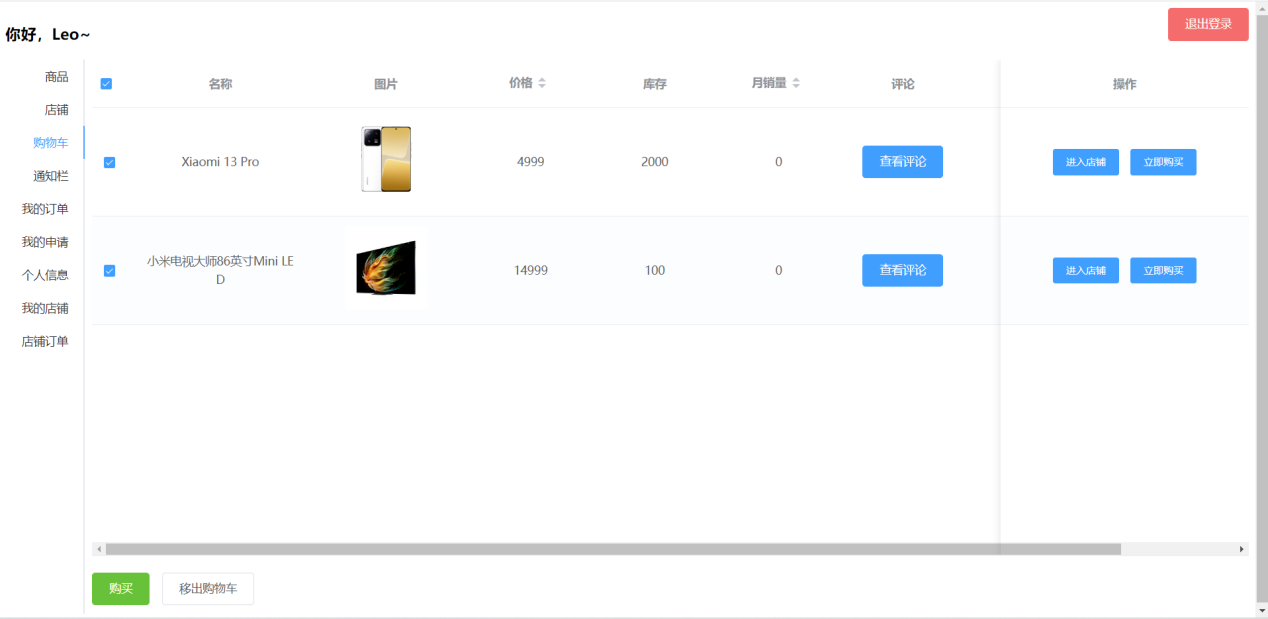
②“进入店铺”的操作时，前端向后端发送商店的id，进行查询。

③“查看评论”的操作时，前端向后端发送的是该商品的id，查看该商品的评论。



有条件查询功能流程图

* + 1. 立即购买、加入购物车、批量购买、批量移除购物车、确认收货、申请售后、下架商品、发货、处理售后、审核申请、删除评论



购物车中的操作实例图

这类方法为无需填写表单对数据库数据进行增删改的操作。

①立即购买：前端向后端发送用户的id和对应商品的id，调用OrderServlet的加入订单的方法，发送数据之前，前端需要判断对应商品的库存是否大于零，若不大于零，则给用户提示“商品售罄”，无法购买，若库存大于零，则可以进行加入订单的操作。后端通过获取的用户id调用UserDao查找到用户的收货地址，和收货人名字，通过获取的商品id获取到商品所在店铺的发货地址，商品id、商品图片、商品名和商品所在店铺的id，将这些信息作为一条记录，添加到数据库中订单表中。

②加入购物车：前端向后端发送用户的id和对应商品的id，将数据添加到数据库中的购物车表中。

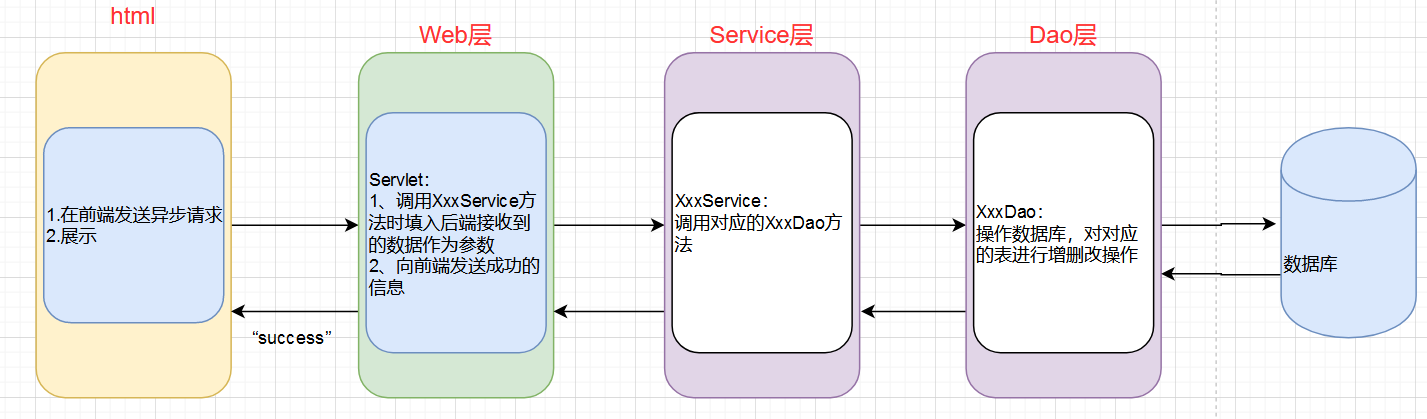
③批量移出购物车：方法与批量购买类似，发送的是商品的id数组，对数据库中的购物车表进行删除的操作。

④确认收货、申请售后、发货、处理售后：无需向后端发送数据，提交异步请求，在数据库中的订单表中修改订单的状态即可。

⑤下架商品：也无需向后端发送数据，提交异步请求，在数据库中的商品表中修改商品的状态即可。

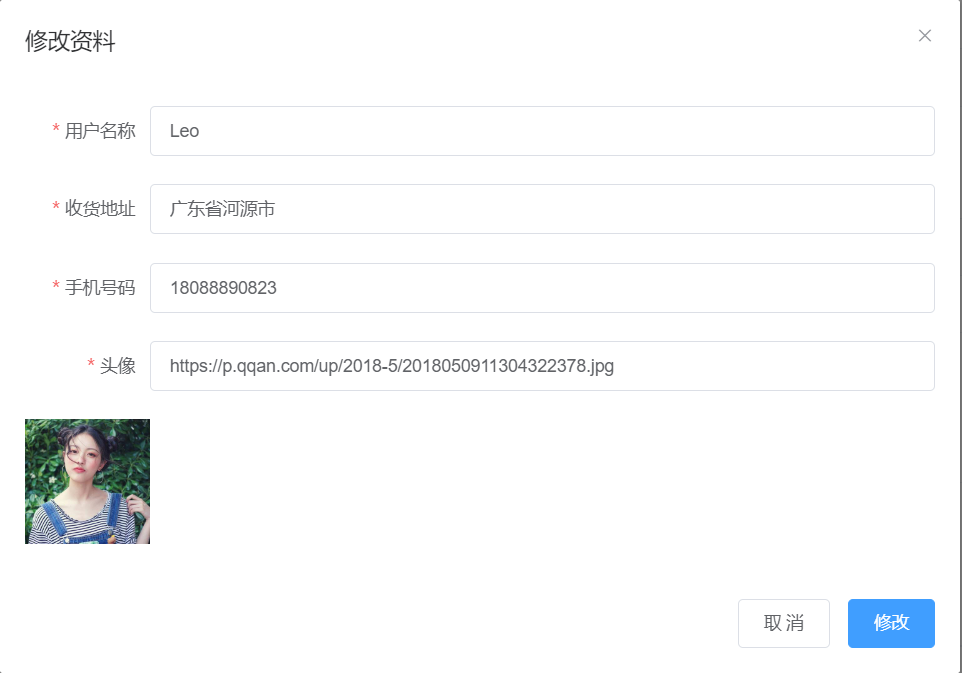
⑥审核申请：无需向后端发送数据，提交异步请求，在数据库中的申请表中修改申请的状态即可，若审核通过，还会进行其他方法，如开店申请通过后，该用户的身份将由普通用户转变为店铺管理员有了自己的店铺；若申请上架商品审核通过后，将会修改商品的状态。

⑦删除评论：该功能只有网站管理员能实现，可以对不良的评论进行删除，前端给后端发送该评论的id和发该评论的用户的id，对数据库的评论表进行删除记录的操作，删除后，还会对用户进行通知。



流程图

* + 1. 对商品评论、申请开店、修改个人资料、修改店铺信息、申请上传新商品、编辑自己店铺商品



修改个人资料示例图

这些功能可以归类为需要填写表单向后端发送数据的功能。

①对商品评论：若想对商品进行评论，用户的商品必须先确认收货，确认收货后商品为待评价状态即可对商品进行评论，填写评论的表单，将表单数据和用户的id发送到后端，后端接收到数据后，对数据库评论表进行添加记录的操作。添加成功后，向前端发送成功信息，提示用户。

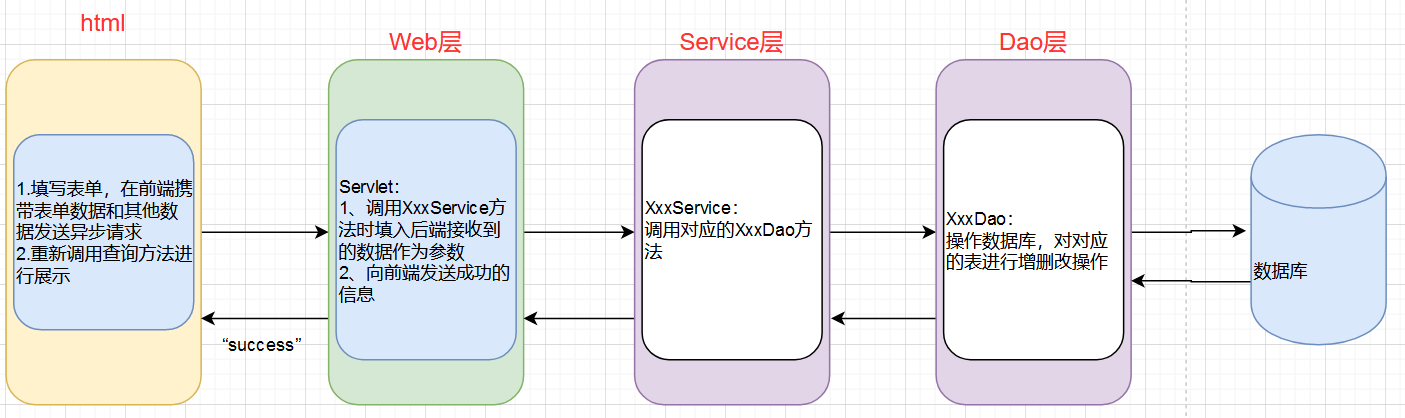
②申请开店：这个功能只有普通用户可以实现，点击申请开店后需要填写店铺信息的表单，将表单数据和用户的id发送到后端，后端接收到数据后，对数据库申请表进行添加记录的操作。添加成功后，向前端发送成功信息，提示用户。

若审核成功，后端会对数据库申请表进行查询操作，将申请表中的数据转为店铺信息，再对店铺表进行添加操作，还有对用户表进行修改用户身份的操作。

③修改个人信息、修改商店信息：每个登录的用户都能实现修改个人信息的操作，但只有店铺管理员才能实现修改店铺信息的操作。点击修改按钮时，会自动同步个人信息到表单上，需要填写需要修改的信息，点击修改后将表单数据和用户的id或店铺id发送到后端，后端接收到数据后，对数据库用户表和店铺表进行修改记录的操作。修改成功后，向前端发送成功信息，提示用户。

④申请上传新商品：店铺管理员可以申请上传新商品，需要填写商品信息的表单，向后端发送用户id、店铺id和商品信息，后端将接收到数据，将数据添加到数据库的商品表中，但商品的状态为未上架状态，若审核通过，会将商品改为上架状态，所有客户都能看到该商品。

⑤编辑自己店铺商品：和修改个人信息、修改商店信息的逻辑基本相同。



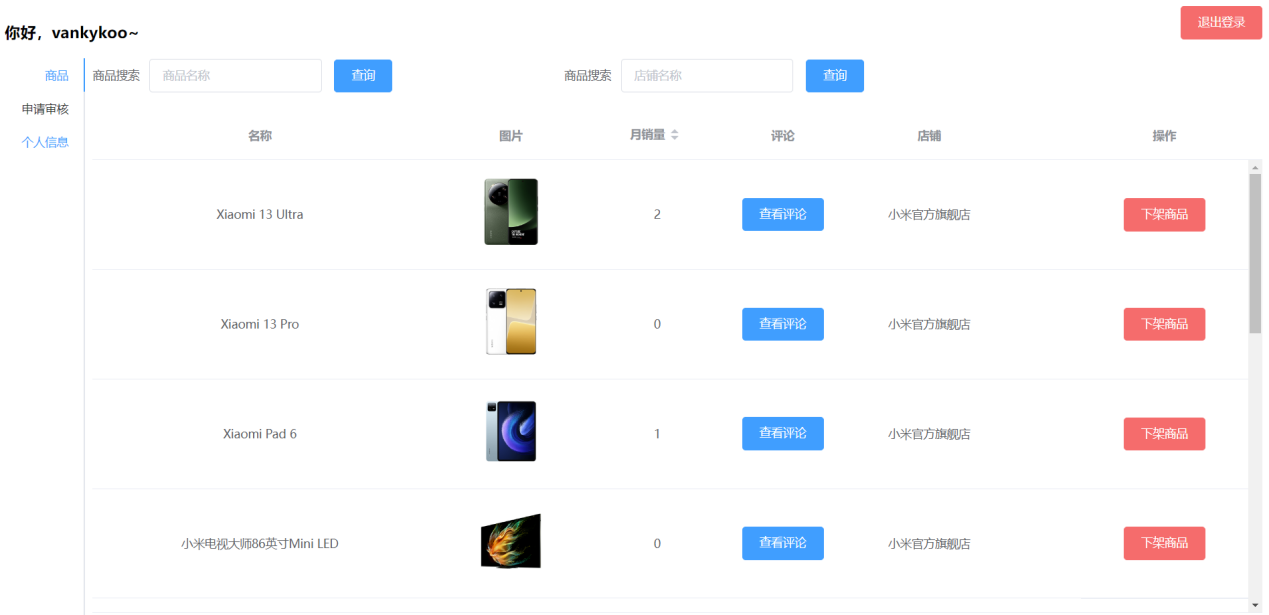
流程图

* + 1. 退出登录功能

退出登录后，将跳转到游客的页面。

# 程序测试

* 1. 页面展示



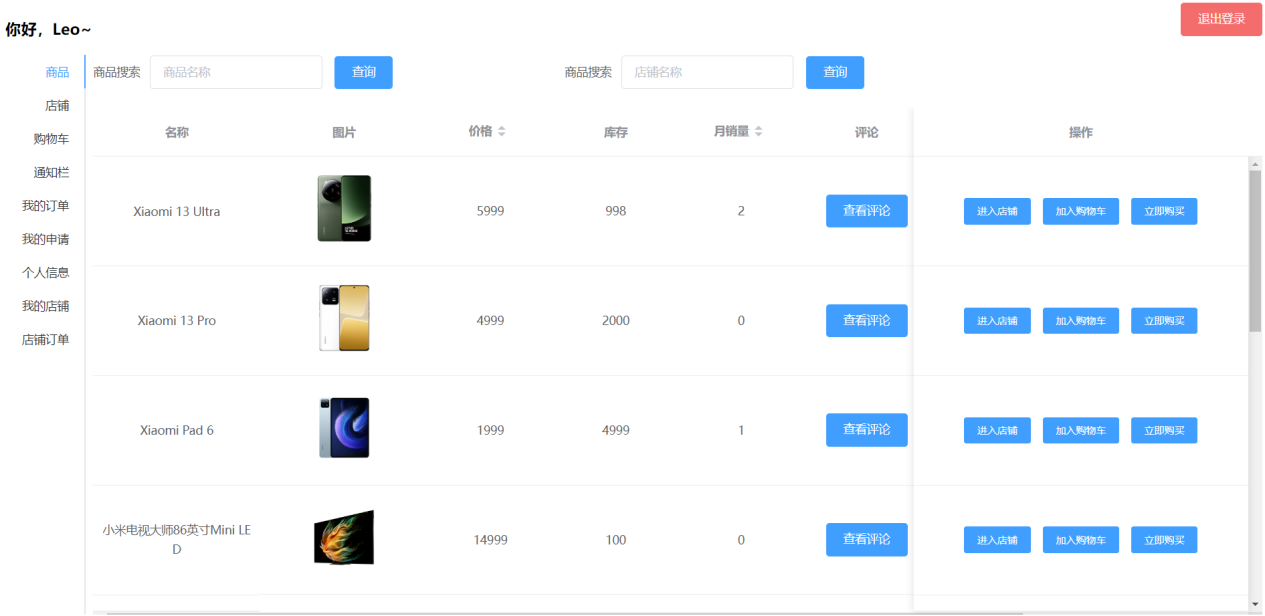
网站管理员页面1



网站管理员页面 2



网站管理员页面3



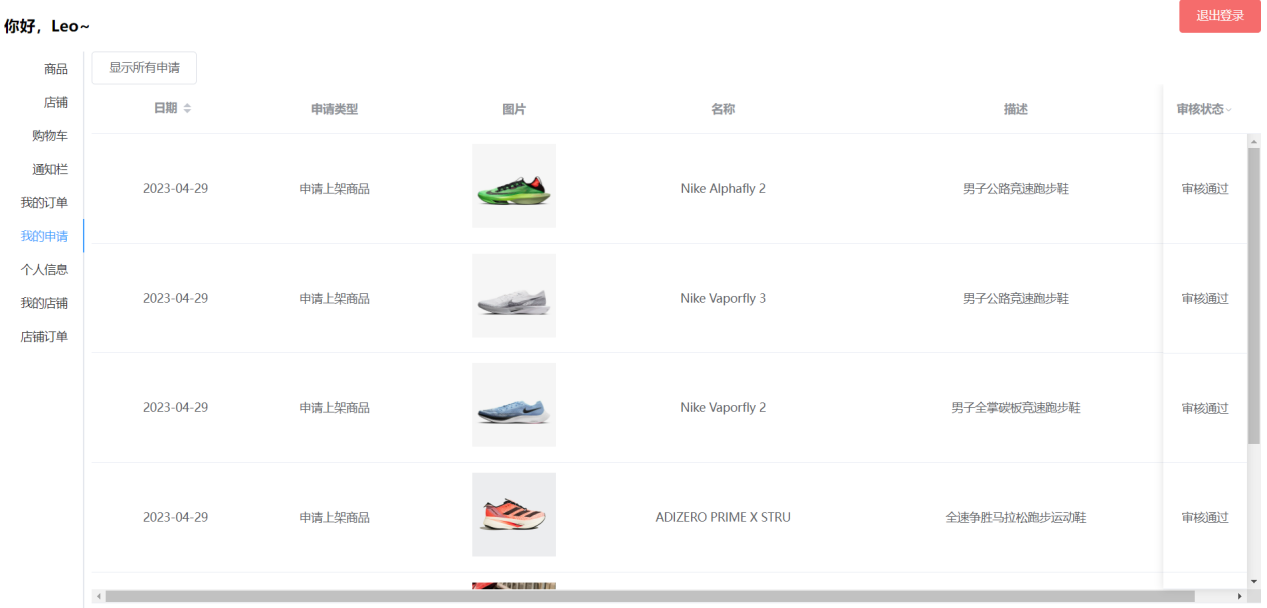
店铺管理员和普通用户页面1



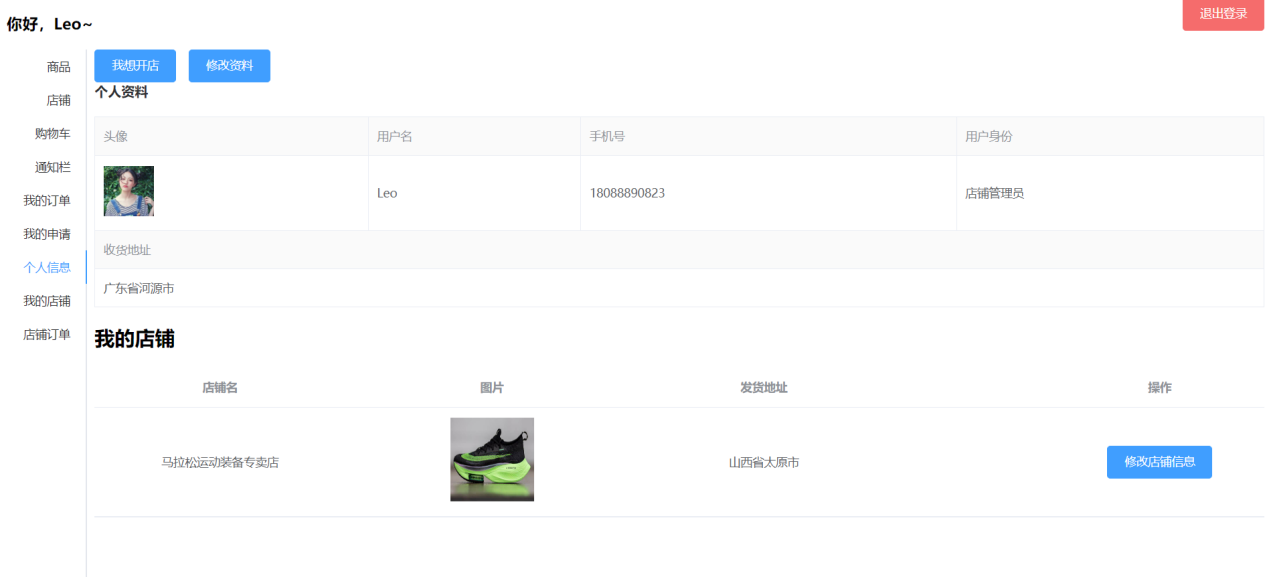
店铺管理员和普通用户页面2



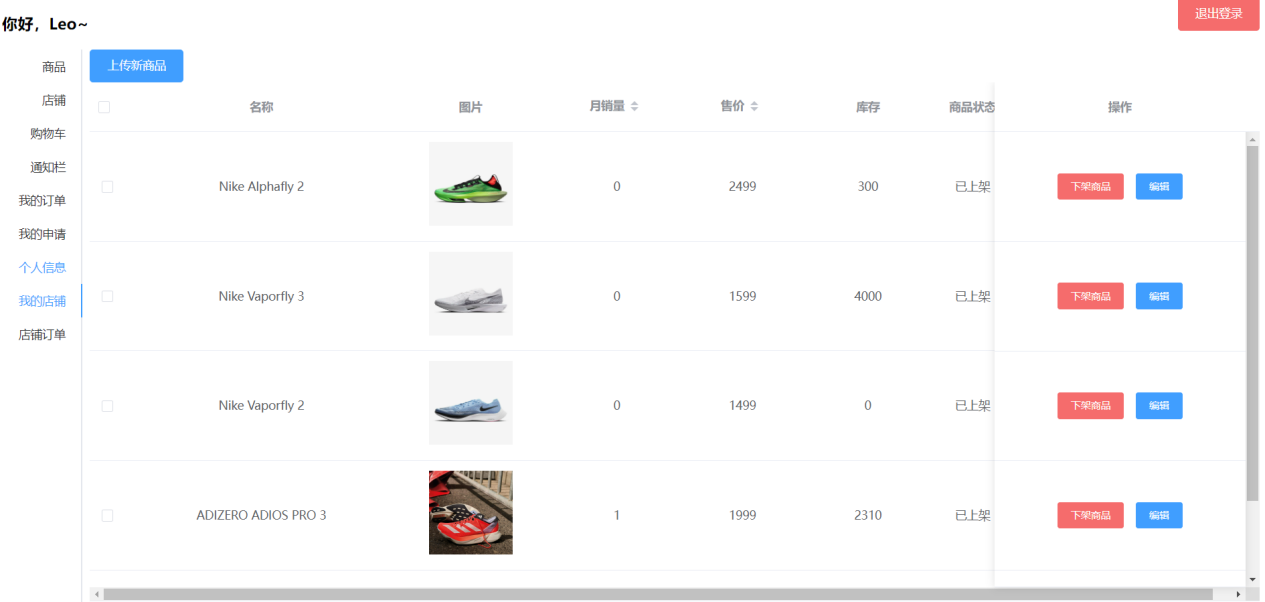
店铺管理员和普通用户页面3



店铺管理员和普通用户页面4

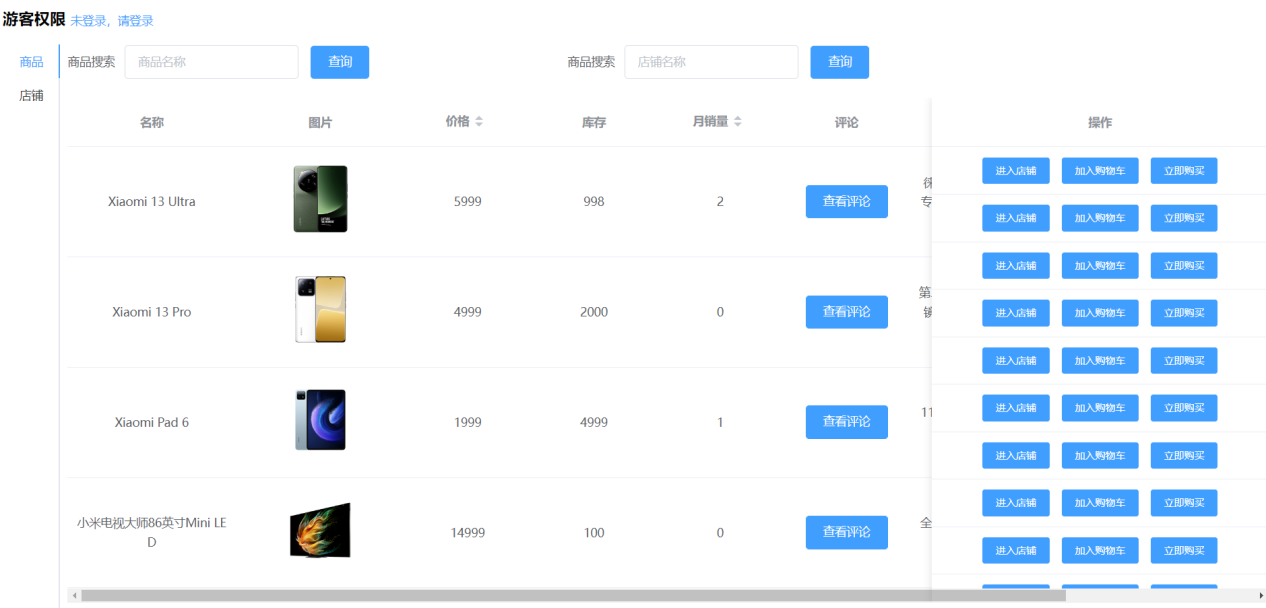


店铺管理员页面1

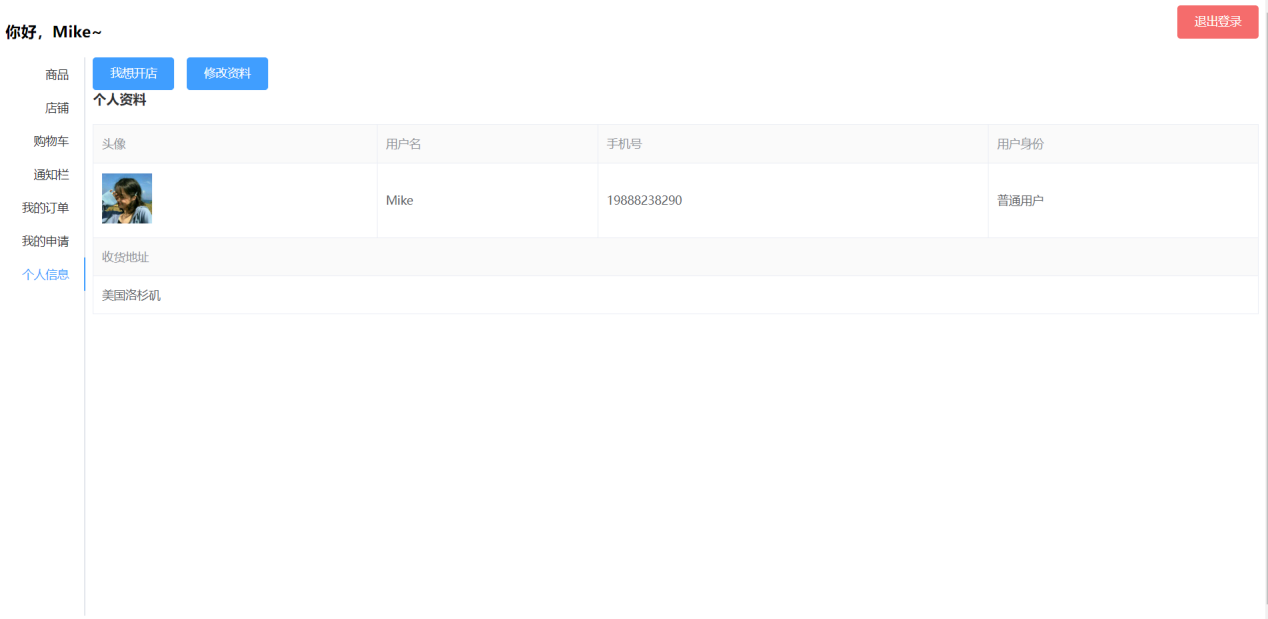
店铺管理员页面2



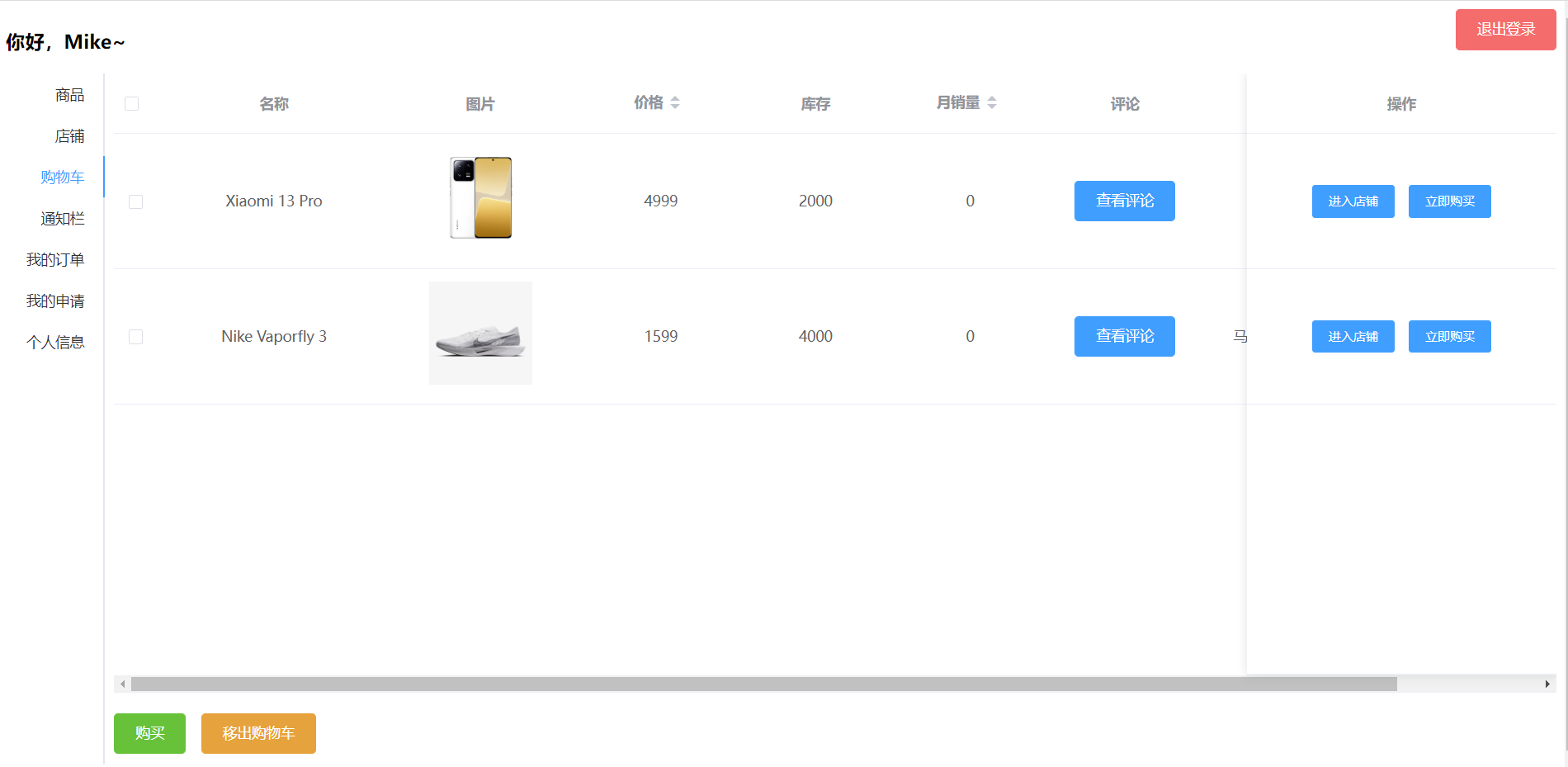
店铺管理员页面3



游客页面1

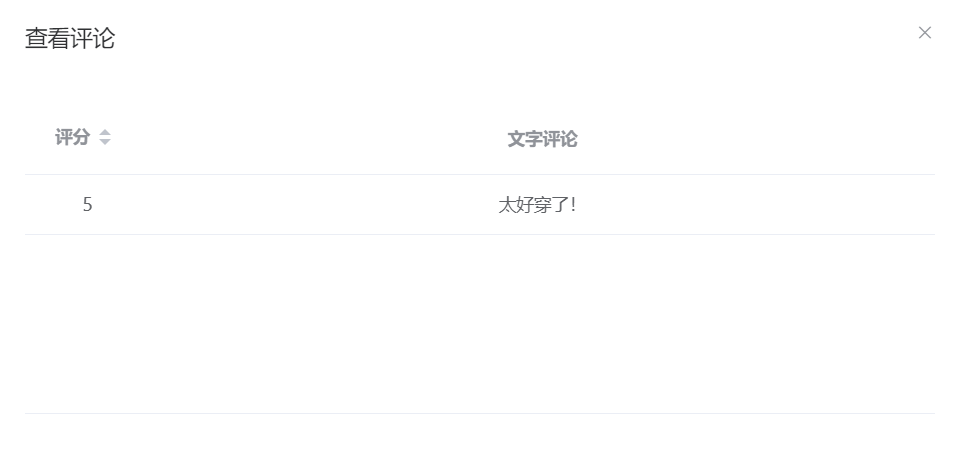


普通用户页面1



普通用户页面2

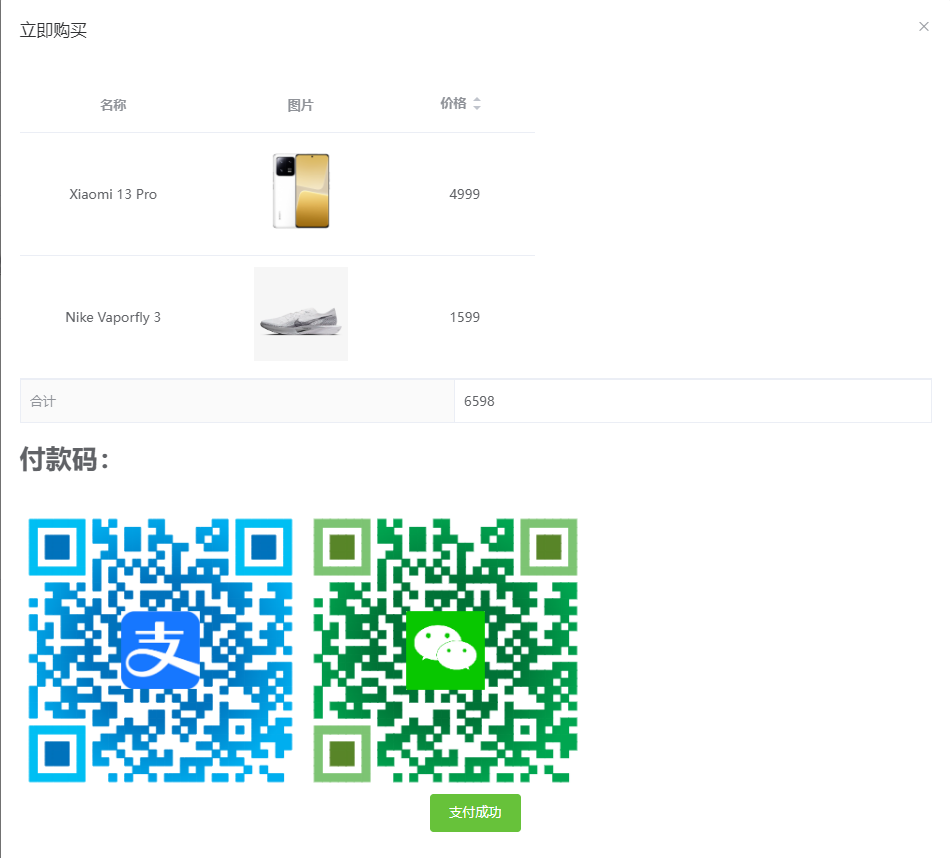
* 1. 表单展示



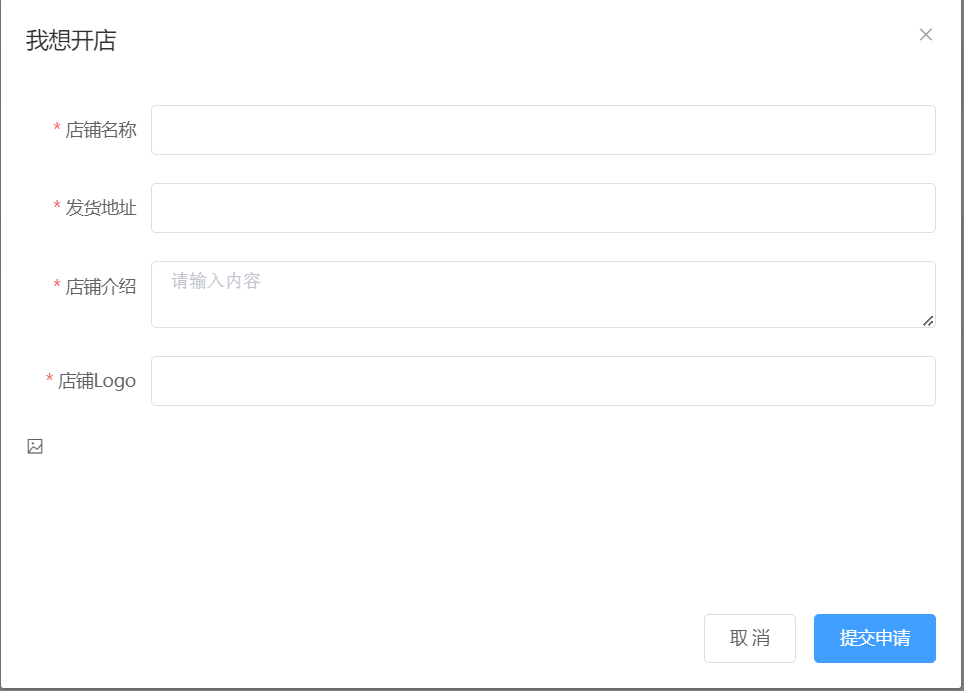
评论表单



立即购买表单



批量购买表单



申请开店表单

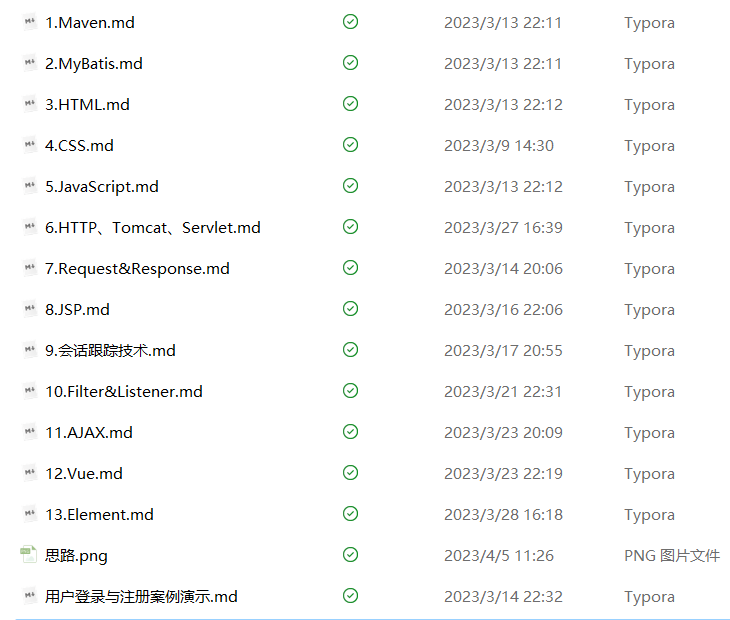
# 项目亮点

1. 使用三层框架写代码，后端文件分好包，需要查找一个方法时更加方便。
2. 数据库的设计较为合理，使数据库中每个表的每个列都有其用处，合理利用内存空间。
3. 使用到了训练营小组作业中的JDBCUtils、数据库连接池、BaseServlet等知识，学以致用。
4. 用户进行登录和注册时，需要验证码验证，防止机器人暴力注册，提高安全性。
5. 在连接池获取连接时，给获取连接的方法和释放连接的方法上锁，不允许被多个线程同时访问，保证数据在任何时刻最多有一个线程访问，以保证数据的完整性。
6. 在网站管理员对申请进行审核后，将审核结果对用户进行通知，还有删除不良评论时也对用户进行通知，使用户使用体验更好。

# 心得体会

通过写这次QG最终考核的项目，我不仅学习到了许多东西，对以前学习的知识有了更加深刻的理解，还知道自己需要学习的东西还有很多很多。

自训练营开始，我学习了Javaweb的内容，初步了解的完成一个网站需要前端和后端程序员相互配合，也知道了后台程序员若想和前端程序员合作需要学习的东西。



我也对之前学习的java基础知识有了更加深刻的理解，例如数据库的连接池、JDBC等等，仅仅学习了知识，如果不去运用，是很难理解深刻的。

同时我也了解到了自己还有许多需要学习的知识。通过这次最终考核，我才知道一个看似小小的功能，需要考虑这么多的问题，在程序的安全性方面需要考虑的问题尤其重要，而我对这方面的了解微乎其微，使我对以后的学习有了更加明确的方向。

