



**QG工作室项目报告**

学 院 自动化学院

专 业 物联网工程

班 级 1班

组 别 后台组

姓 名 黎颖武

学 号 3122003153

2023年 月 日

广东工业大学计算机学院制

目录

[1. 项目简介 1](#_Toc30766)

[2. 设计思路 1](#_Toc24252)

[3. 详细功能设计 1](#_Toc20914)

[4. 程序测试 1](#_Toc7265)

[5. 项目亮点 1](#_Toc19276)

[6. 心得体会 1](#_Toc25747)

# 项目简介

本项目意为搭建一个购物网站，具有游客，用户，店家，管理员多种角色，游客可以查看商品，阅读商品评论等等，登录了的用户则还可以进行购物，查看自己所购商品，店家能够上架商品和处理订单，管理员则进行管理管理网站，处理投诉信息，维护网站的正常运行等等。

# 设计思路

1. 归总主要功能模块

首先，从需要实现哪些业务开始设计，纵观日常使用的购物平台，如x东、x宝等等，从它们上归类出购物网站必须有的功能，如用户登录，搜索商品，查看商品，上架商品，购买商品，查看购买的商品，评论商品，投诉功能，保存个人信息，还有管理网站的功能等等。

1. 将功能划分模块  
   如此众多的功能，必须需要归类，以利于管理和实现，将功能按不同的模块进行划分，如搜索商品和查看商品等功能就划分到商品模块
2. 设计数据库表  
   整理完全要实现的功能之后，就开始进行数据库表设计，设计表的字段，索引等等，如商品表goods等等
3. 商品id自增主键
4. 商品文字介绍
5. 商品价格
6. 图片id
7. 商品所属店铺id
8. 月销量（成功收货之后 月销量++）
9. 库存
10. 启用状态（0或1）

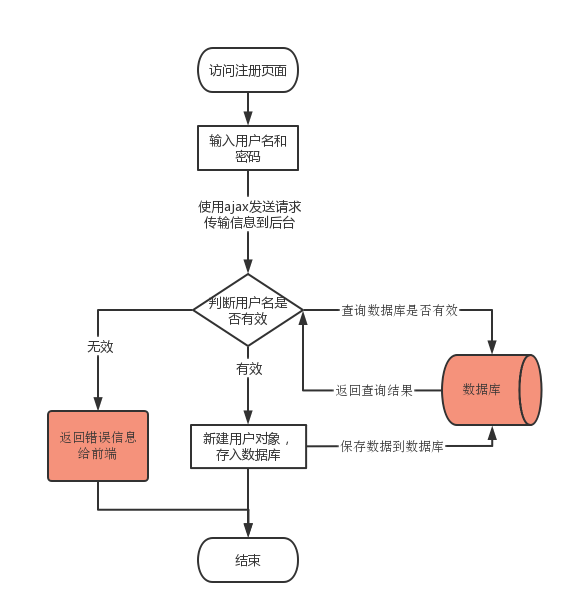
表的设计会与业务功能的实现有着很大关系，数据库表需认真花时间设计好

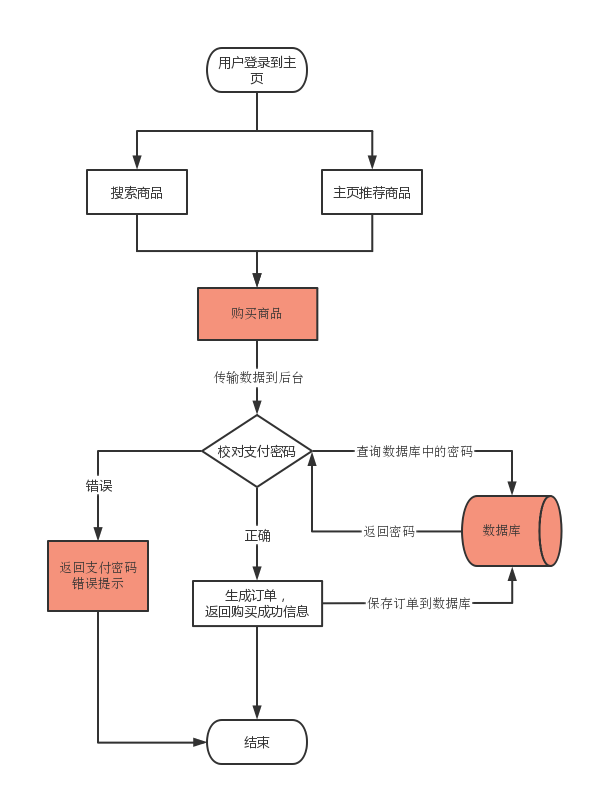
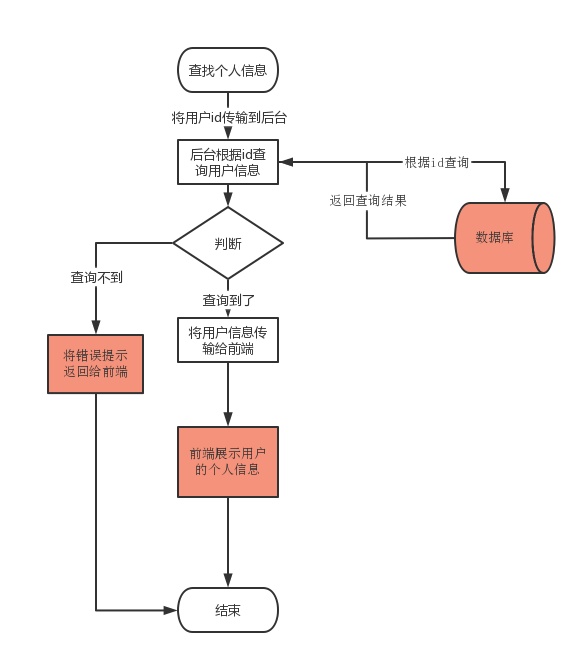
# 详细功能设计

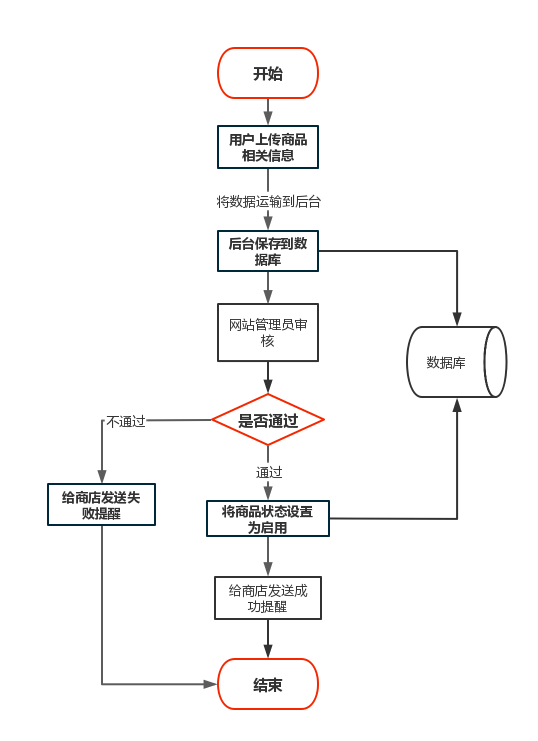
# 列出部分功能流程图:

# 用户登录 wps

# 用户注册



1. 购买商品  
   
2. 查找用户信息  
   
3. 商家上架商品



# 程序测试

1.登录功能

输入：lisi，1234

预期结果：登录失败

实际结果：登录失败

输入：lisi，123

预期结果：登录成功

实际结果：登录成功

2.注册功能

输入：lisi，1234

预期结果：注册失败

实际结果：注册失败

输入：wangwu，123

预期结果：注册成功

实际结果：注册成功

3.购买商品

输入：dawdafa(错误密码)

预期结果：支付密码错误

实际结果：支付密码错误

输入：123（正确密码）

预期结果：购买成功

实际结果：购买成功

4.修改支付密码功能  
输入原支付密码，新密码，确认新密码:wadsd,234,234

预期结果：输入原密码错误

实际结果：输入原密码错误  
输入原支付密码，新密码，确认新密码:wadsd,234,123

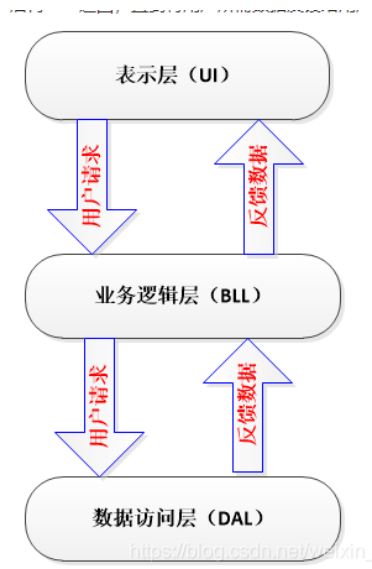
预期结果：确认新密码不一致

实际结果：确认新密码不一致  
输入原支付密码，新密码，确认新密码:123,234,234

预期结果：修改成功

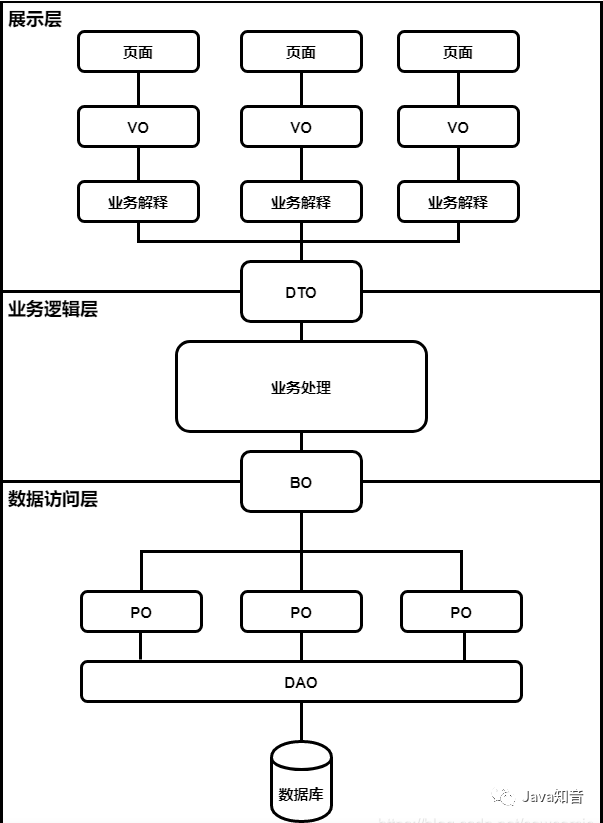
实际结果：修改成功

# 项目亮点

* 1. 使用MVC与三层架构模型  
     将代码分成UI(表现层)、BLL（业务处理层）和DAL（数据访问层）。区分层次的目的即为了“高内聚低耦合”的思想。在软件体系架构设计中，分层式结构是最常见，也是最重要的一种结构，三层架构将代码逻辑分层，降低代码的耦合度，利于维护和开发。  
     具有以下优点
* 结构清晰、耦合度低
* 可维护性高，可扩展性高
* 利于开发任务同步进行, 容易适应需求变化  
  
  1. 运用反射

编写了工具类方法beanToMap，该方法意在将对象的属性名和属性值取出来，存放在map中，然后该方法返回该map集合，  
如果有1个属性，就调用一次get方法，如果有100个属性，难道就调用100个方法吗？显然不可能，所以我运用反射，来获取类中所有属性字段和方法，用for循环，自动获取属性名称，然后调用实体类的get方法得到属性值，将属性名称和属性值分别作为键和值，存储到map集合中，方法最后返回该map集合

使用反射来获取，实现高效开发

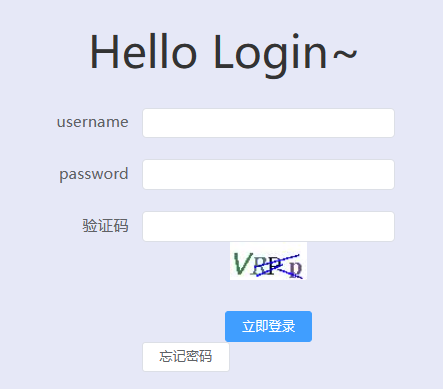
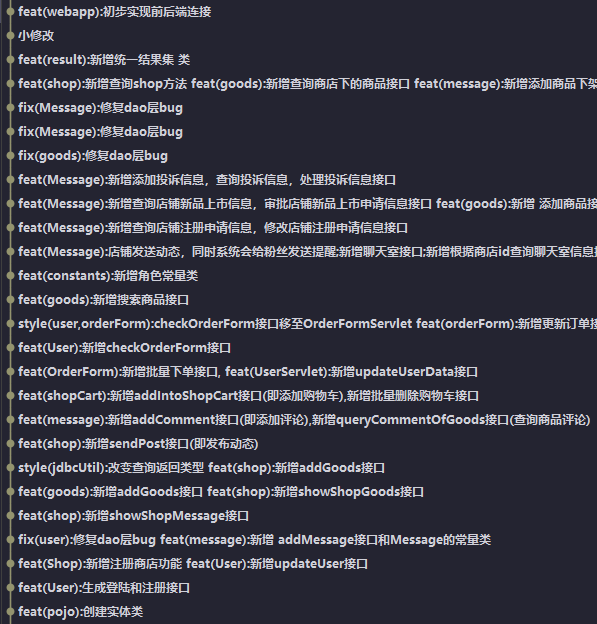
* 1. 使用postman记录接口文档  
     编写后台代码过程，每实现一个功能，我就会及时用postman测试接口，记录测试接口，使代码利于维护和管理，方便进一步的开发。
  2. 实现手写数据库连接池  
      数据库连接是一种关键的有限的昂贵的资源，这一点在多用户的网页应用程序中体现得尤为突出，每次操作都打开一个连接，使用完都关闭连接，这样的系统性能低下，资源也大幅消耗，而数据库连接池则完美的解决了这样的问题。  
      数据库连接池的基本原理是在内部对象池中维护一定数量的数据库连接，并对外暴露数据库连接获取方法，我们需要连接数据库时，就从池子中取出连接，用完就将连接放回池子里，实现资源重用，同时具有更快的响应速度，大大提高程序运行效率，在统一管理连接的情况下，也能避免数据库连接泄露。
  3. 编写JDBC工具类，CRUD操作工具类  
     每次使用JDBC的时候都要书写几长的代码段，不符合复用的理念，于是要单独写一个类，将通用的JDBC操作写到一个类中，便于重复使用和精简代码，大大提高开发效率和提升代码美观性
  4. 后台封装统一返回结果集  
     在开发中，返回到前端的数据内容格式趋于一致，我们在开发过程中最好能够将返回数据对象的格式进行约定，将返回结果定义成统一格式，此项目中就封装了Result实现统一返回结果集，能够降低开发人员沟通成本，提高开发对接效率。
  5. 使用常量类和配置文件  
     在实现统一返回结果集中，配置了常量类以方便返回错误信息时调用该常量类，而不是直接将字符串以魔法值的形式写在代码中，解决硬编码问题；在实现手写数据库连接池中，对于连接数据库的配置信息，专门配置了配置文件，而不是以硬编码的形式直接写在代码中，将本应该（可以）写到配置文件中的信息直接在程序代码中写死，这样既优化了代码，也便于日后修改和维护代码。
  6. 使用jwt进行鉴权认证  
     编写jwt工具类，实现得到token令牌方法，解析token令牌方法，还实现了从token令牌中获取特定内容的方法，将用户的角色id、用户名和用户id存放在token令牌的载荷中，前端每次访问后台时，需携带token，以便结合多角色多权限模型，来判断用户角色是否足够权限使用该功能。
  7. 使用泛型提高效率  
     实现jwt鉴权中，实现一个方法从token中获取载荷中的特定内容时，使用了泛型作为形参，调用该方法时则传入想要获得的值的数据类型，返回同样的数据类型，这样不但能够做到类型安全，还能消除代码中出现的强制转换，同时也能带来潜在的性能优化。
  8. 使用正则表达式校验输入内容  
     每一门编程语言都包含正则表达式这个内容，并且正则表达式还不受其编程语言的影响  
     项目中校验输入手机号码时，前后端均使用了正则表达式校验手机号码的格式，相比手动处理手机号码字符串，使用正则表达式大大减少了代码量，做到速度更快，效率更高，准确性更高。
  9. 采用VO、PO等模型  
       
      VO对应于页面上需要显示的数据 (表单)，PO就是数据库中的记录，一个PO的数据结构对应着库中表的结构。  
     在项目开发中，在后台DAO(持久层）与Service（服务层）与Controller（控制器）之间使用PO模型传输数据，但如果传输数据到前端显示也用PO模型类，会将不需要的字段也会传递到前端页面，不仅使数据的传输量增大，还可能有安全性问题，比如容易造成password等数据泄露  
      故采用了VO模型，VO模型中只包含展示所需要的字段，不会将多余的数据传输到前端页面，防止数据泄露等。
  10. 封装BaseServlet类  
      在我们刚接触servlet 的时候，我们开发是这个样子的：  
      为了写一个用户登录功能，我们新建一个 Loginservlet 。

为了写一个用户注册功能，我们新建一个 signupServlet 。

为了写一个用户更新功能，我们新建一个 UserUpdateServlet 。

为了写一个用户删除功能，我们新建一个 UserDeleteServlet 。

为了......  
但是正常的项目上两位数的实体类是很正常的，难道我们对每个实体类的操作都要像上面那样建立 N个 Servlet才行吗?显然不可能，所以就有了封装BaseServlet类  
 BaseServlet将重复的东西归到一起，然后让其他的servlet继承该类,其他servlet只需要根据实际情况进行方法的改变，比如UserServlet就整合了Loginservlet、signupServlet和UserUpdateServlet等等，解决了控制层类和代码冗余问题，各功能区分到不同的 Servlet 中方便维护。  
 在此基础上还结合了反射技术，通过反射自行寻找对应的方法并调用与获取结果，相比手动使用switch语句进行判断，实现逻辑简单、使用简单，能够自动判断并执行方法、减少人为出错。

* 1. 实现用户验证码登录，忘记密码等功能  
     验证码最大的功能就是保护网站安全，一般网站都会通过验证码来防止机器大规模注册，机器暴力破解数据密码等危害。  
     本项目也在登录和注册页面配置了验证码功能，保护网站的安全运行。  
     除此之外，还实现忘记密码登录功能，提高用户体验，吸引用户，便利用户，留住用户。  
     
  2. 用户敏感信息加密传输，数据库数据加密存储  
     作为一个为用户提供购物服务的网站，用户放心使用的同时，我们也必然保护用户的数据信息，包括但不限于登录密码和支付密码等等，在系统传输过程中，都在先进的MD5加密技术的基础上进行了进一步的加密处理，加密存储在数据库中，保护用户数据安全，让用户安心。
  3. 使用Git管理源代码  
      在项目一步步实现过程中，利用Git的强大的分支功能，每实现一部分功能，就会在代码正确的前提下提交到自己创建的dev分支，相当于给代码备份，能够随时进行代码回退，同时也做到了小粒度提交代码，既能降低与别人冲突的可能性，避免代码冲突合并的痛苦，也能让所有的开发都能看到最新的代码，相互了解各个模块之间的进度，最重要的就是能够避免辛辛苦苦好几天写的代码就因为硬盘损坏、突然断电或电脑丢失而回到解放前的情况。

# 心得体会

寒假就开始跟着路线学习Java知识，一路上经历了不少的磨难和艰险，无论是克服那提不起学习的态度，还是战胜一个个代码里的爆红警告，一切我都克服了。

在学习的过程中，不止一次遇到各种困难，历经1小时、2小时、1天......才解决的困难，我当然有想过放弃，因为我觉得这条路好难走，路上只能靠自己，也没有人教，遇到bug也不知道怎么解决，有些技术也不知道该怎么实现，一次次的麻烦同学也会觉得不好意思不再继续打扰，但当我不断克服一个个代码爆红，解决一个个技术问题时身上充满的自信，是除了学习之外绝无仅有的，我一点点充满信心，充满动力，每次遇到困难都会积极安慰自己“我难别人也难”“最难不过坚持”，一路上一点点坚持着，终于坚持到了最后。  
 如果问我回到一开始，我还会继续学习java，卷工作室吗？答案是肯定的，如果只空有绩点，还不算一个很强的大学生；如果整日只会待在宿舍打游戏，那我认为这将毫无意义（我认为玩游戏只有忙里抽闲的玩才是最好玩的）。我不喜欢浪费时间，我更喜欢效率高的生活，无论是打游戏，还是做功课，还是安排时间，大学生有极多的空闲时间，有人将这些空闲时间用在玩游戏，用在出去玩，空闲时间放松自己无可厚非，但如果只将这些空闲时间用于放松自己将会“不划算”，我更愿意将这些时间的大部分用于提升自己，成就更好的自己。

在这学习的一路上乘风破浪，克服不少困难，经历不少挫折，但是一切都过去了，此时恰好能用李白《早发白帝城》中的“回头看，轻舟已过万重山”来形容走过的这一段充满艰险的漫长历程。不论结果如何，无论是这三个月的学习还是这一个月多的训练营，还是这两个星期的做项目，都让我受益颇多。我相信，无论是这几个月学习到的有用知识，还是学习过程中养成的坚持不懈、孜孜不倦的优良学习态度，都定会在我大学前进的道路上助我一臂之力。