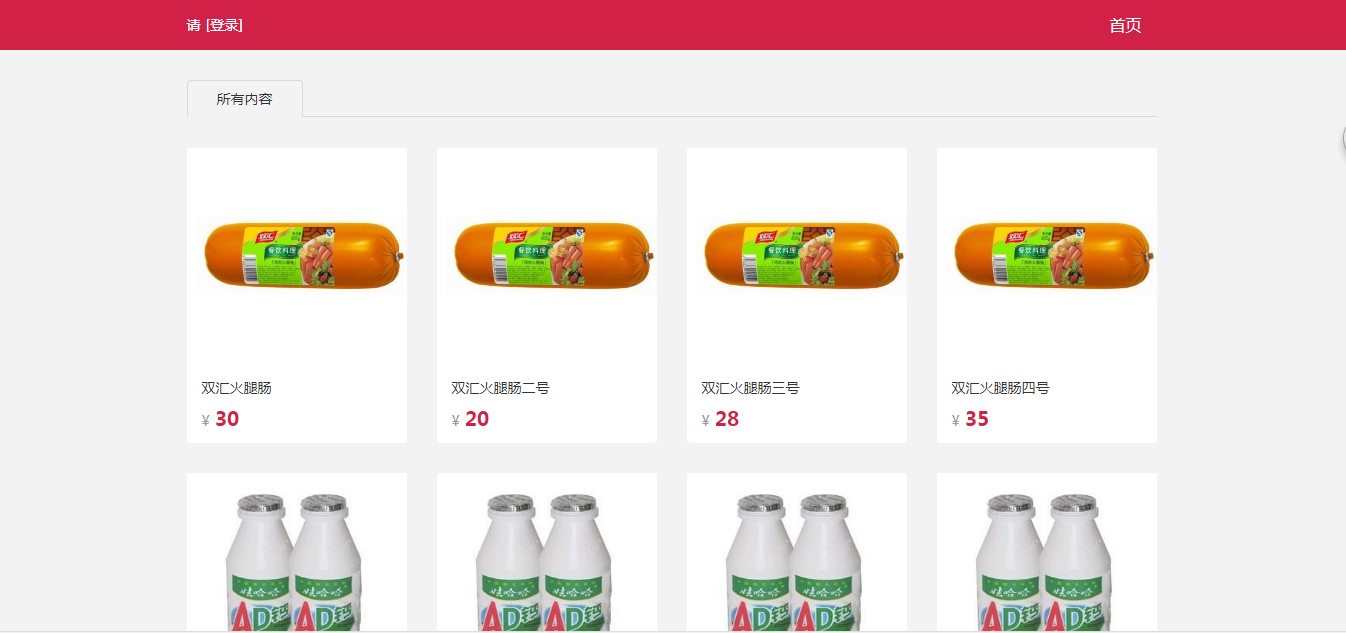
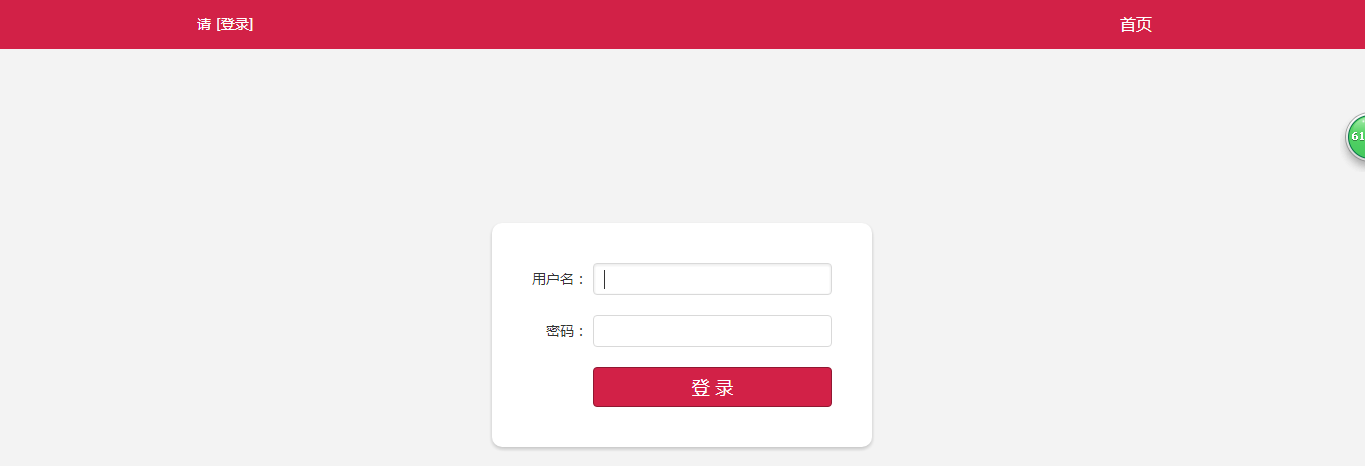
# 相关功能概述及截图

1. 首页截图：首次访问所显示的页面，主要内容是商品列表，点击商品后只能查看商品的相关信息，不能进行任何操作。截图如下所示：

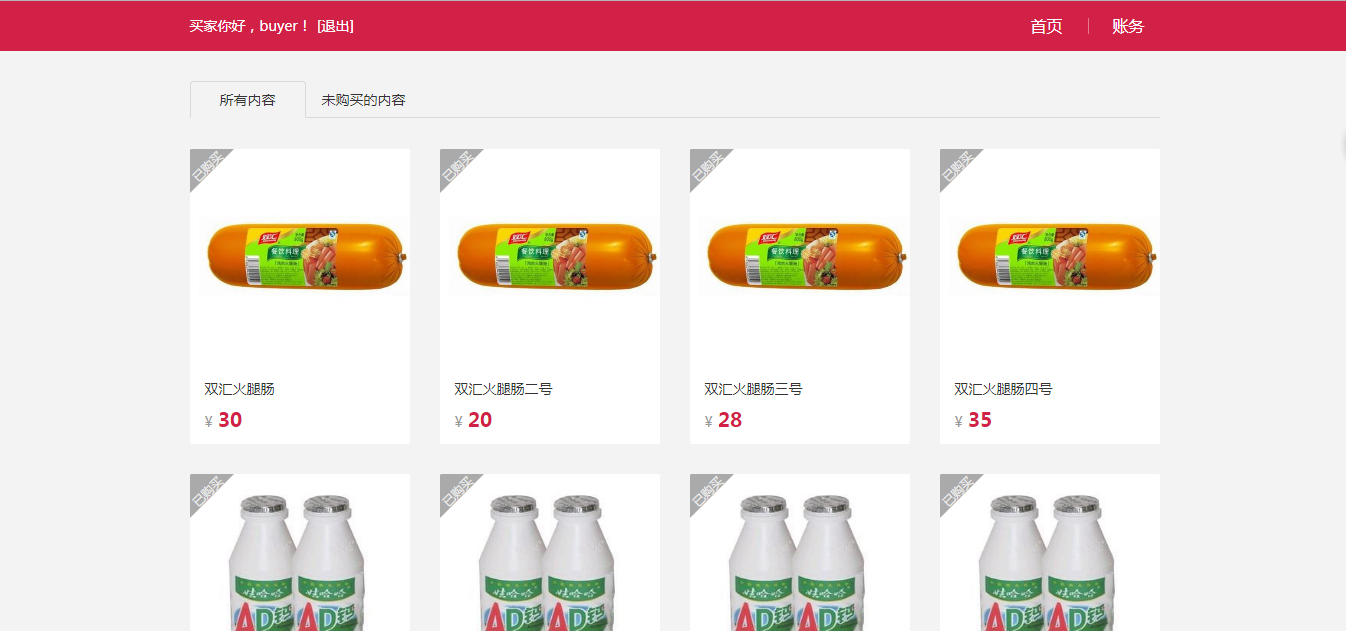


1. 登录模块截图：点击登录按钮，跳转至登录页面。需要输入用户名密码进行登录，登陆后会根据用户的类型进行分别显示。截图如下所示：

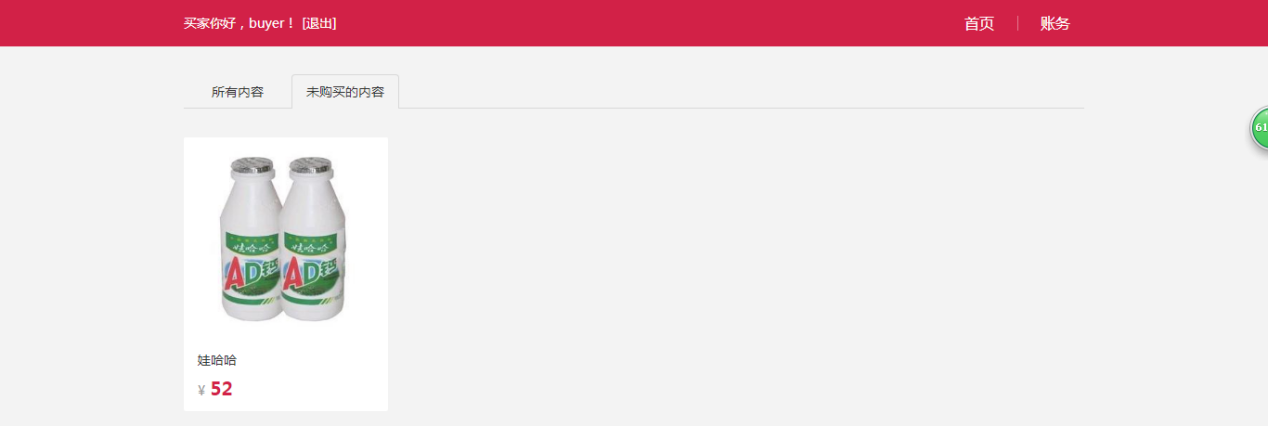


1. 买家模块：买家登陆后，首先显示出的商品清单页，已购买商品会在列表项里会进行标注，点击未购买商品，会展示没有购买的商品清单，点击商品图标，即可跳转购买页，点击购买，购买成功后，在买家账务中会有列表显示。

商品清单页截图：



未购买商品清单截图：



商品购买界面截图：

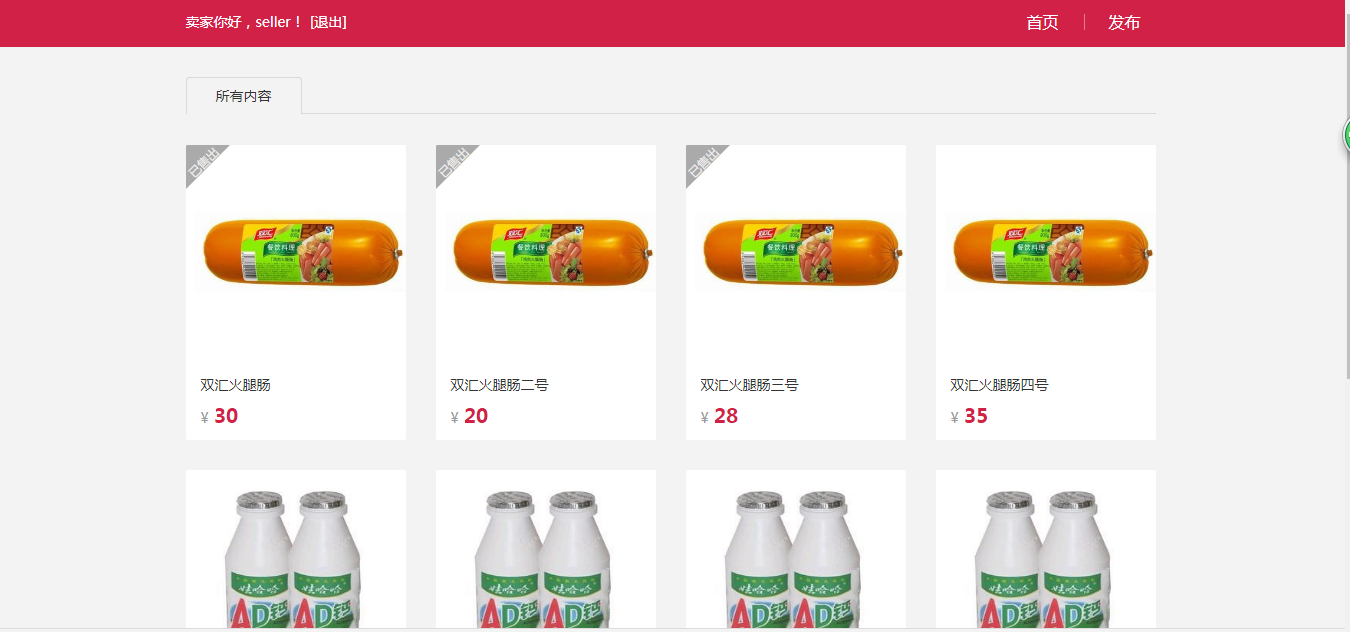


账务清单截图：



1. 卖家模块：卖家登陆后，会显示商品清单列表。其中，已经被售出的商品会进行标示。针对未售出的商品，卖家可以进行下架删除操作，也可以进行编辑信息操作。此外，卖家还可以进行商品发布，填写相关信息，即可发布商品。

商品清单页截图：



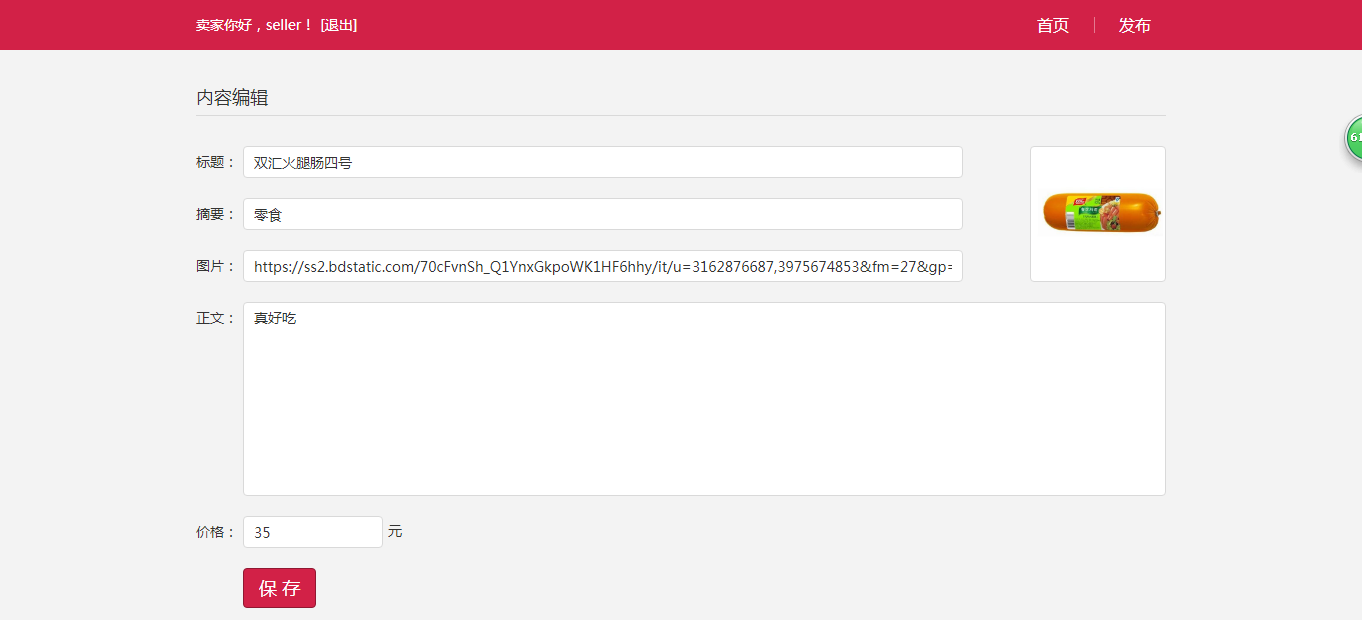
商品删除操作截图：



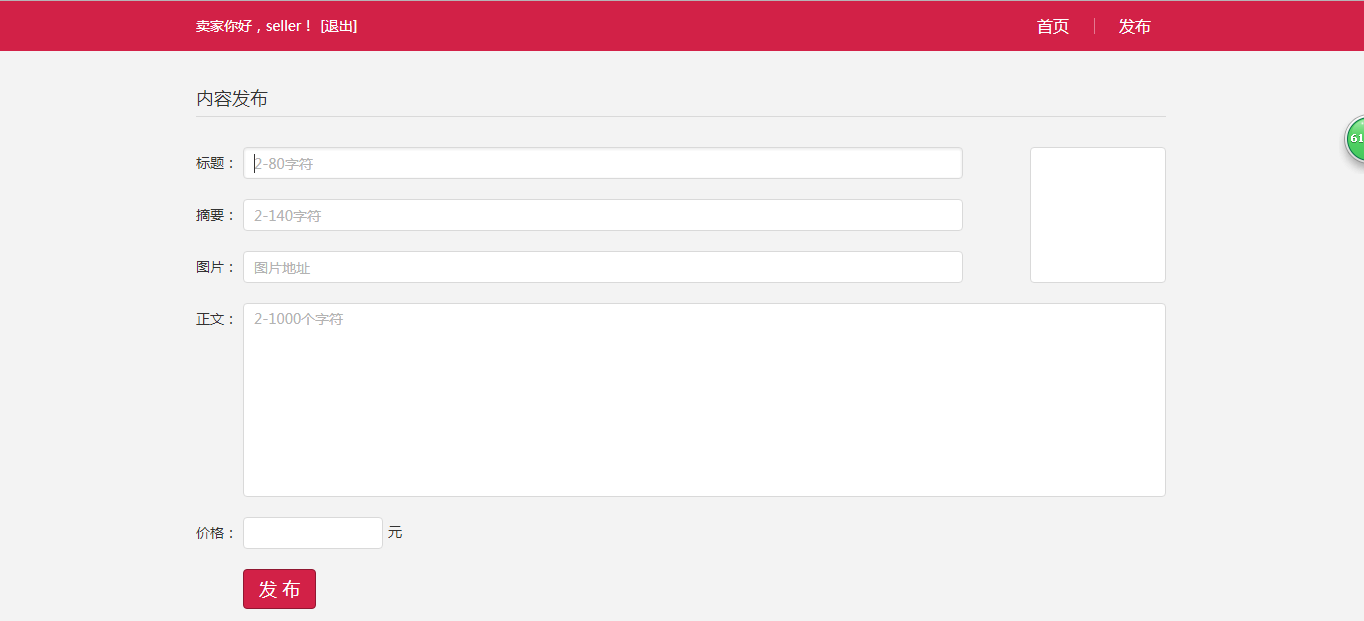
商品编辑页展示截图：



商品编辑操作截图：



商品发布操作截图：



# 实现步骤

1. 准备数据库及其数据表

根据给定的文档中的数据库文件，建立数据库及其数据表，并添加相关记录。

1. 工程目录设置：

Java目录下建立：

Controller层：存放数据交互的相关controller

Dao层：存放数据库访问接口，采用Mybatis和Spring整合后进行数据库访问。

Service层：业务逻辑处理，承上启下，Controller层请求Service层，Service层调用Dao层。采用接口实现类的方法进行编写。

Pojo层：简单的实体类，对应数据库中的数据表。

1. 整合Spring，Mybatis框架，配置文件归纳

a .建立db.properties 文件，存储数据源的配置。

b .建立spring-mybatis.xml 文件，进行数据源配置和mybatis的相关配置。

c .spring-mvc.xml 文件，freemarker相关配置和返回json数据的配置

d . web.xml文件，前端控制器，配置文件的相关声明。

1. 根据数据交互文档构建Controller

控制层将异步数据请求接口和常规请求分开编写。

1. 部署运行

部署到tomcat服务器上调试运行。