

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | B/S体系软件设计 |
| 实验名称： | 商品比价网站 |
| 姓 名： | 周玥儿 |
| 学 院： | 计算机学院 |
| 系： | 计算机科学与技术 |
| 专 业： | 计算机科学与技术 |
| 学 号： | 3220102179 |
| 指导教师： | 胡晓军 |

2024年 12月 23 日

**目录**

[项目启动 2](#_Toc185843903)

[使用说明 3](#_Toc185843904)

[docker使用方法 10](#_Toc185843905)

### 项目启动

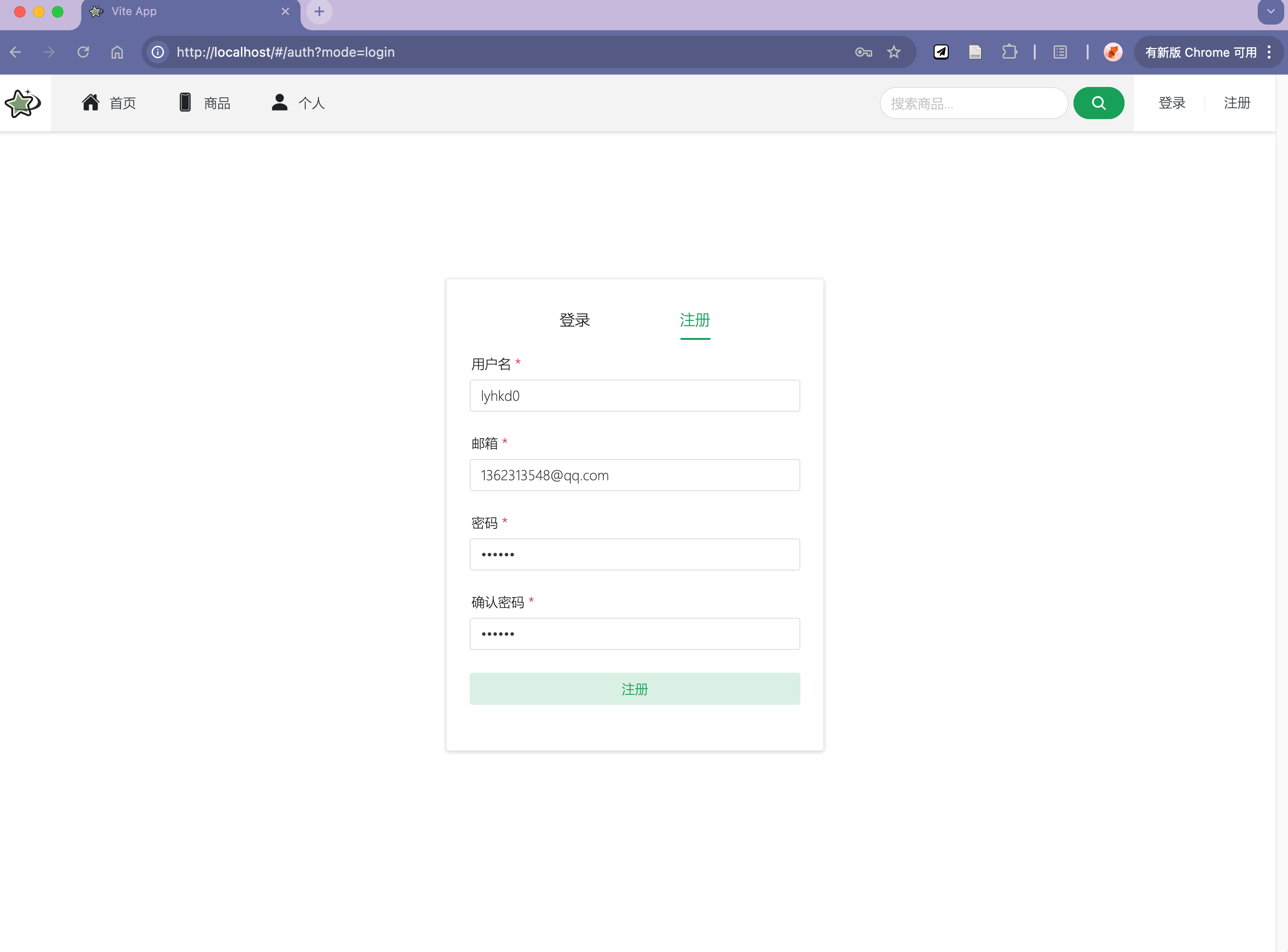
front-end/ 文件夹下存放了前端的源代码，back-end/ 文件夹下存放了后端的源代码，sql/ 文件夹下存放了数据库数据，crawler/文件夹下存放了爬虫脚本。

1. 在根目录下（包含docker-compose.yml的目录）进行Docker构建 ，在终端输入docker-compose up --build进行构建。如果提示加载错误可以使用Docker Hub内已经上传的镜像。
2. mysql数据库文件存放在sql目录下，在构建mysql容器使会使用里面的内容进行初始化。已经存在的测试用户是lyhkd0,密码是123456。
3. cookie获取，使用浏览器的无痕模式打开<https://m.jd.com/>，登陆后选择任意一个请求复制cookie存入back-end/data/JDcookie.txt文件内。打开<https://www.amazon.com/>，选择好地址（国家），价格单位（RMB）后存储cookie入back-end/data/Amazoncookie.txt内。打开<https://www.gwdang.com/>，登陆后存储cookie入back-end/data/GWcookie.txt内。**配置好cookie后，才可以正常使用实时搜索功能**，否则只能从数据库中获取商品信息。如果依然不成功，可以从crawler文件夹中使用各个平台单独的爬虫脚本，测试是否正常。
4. 短信发送提醒功能需要在<https://www.smsbao.com/>网站充值使用，目前免费额度已经测试用完（😢）

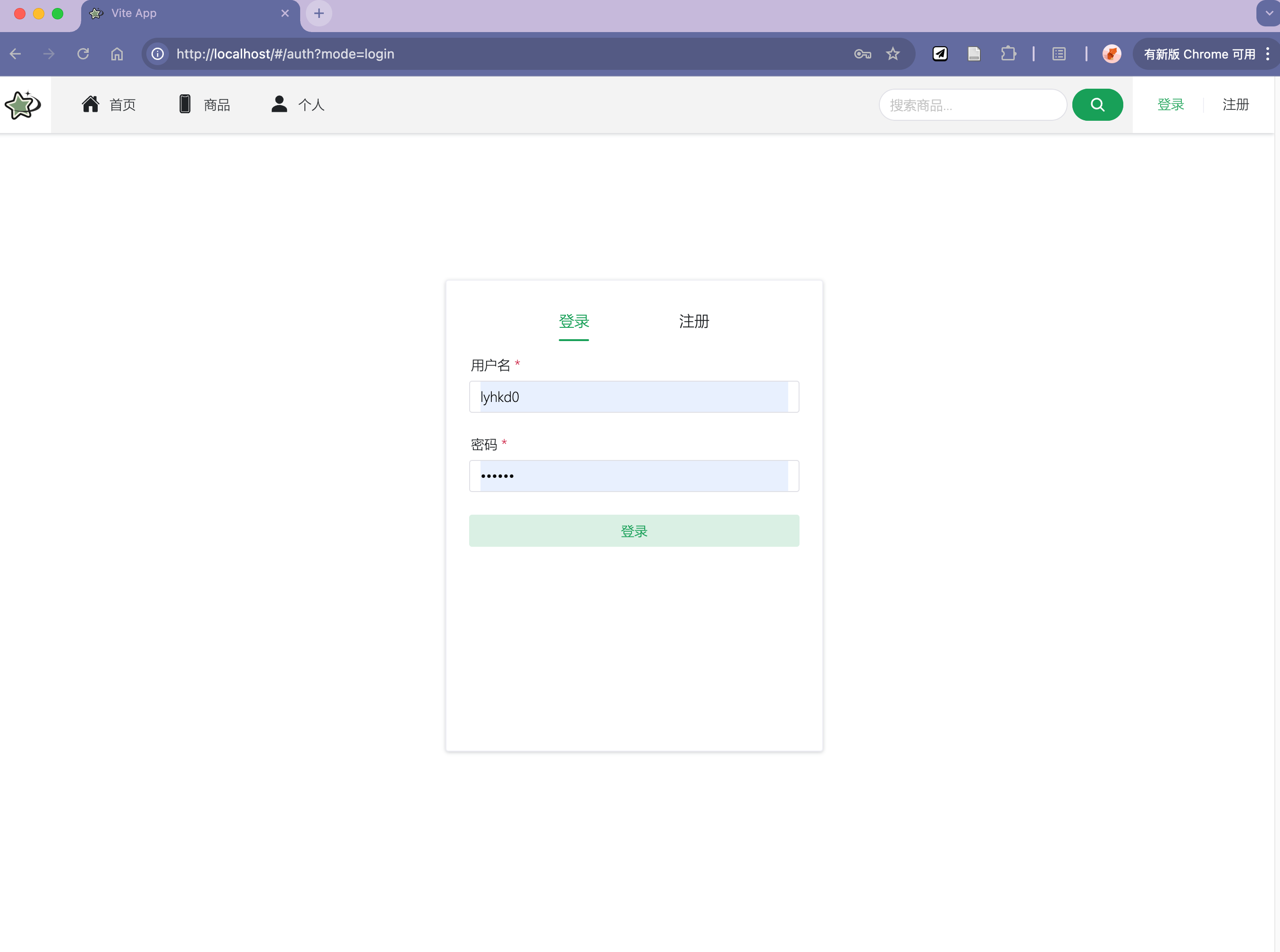
### 使用说明

#### 登录注册

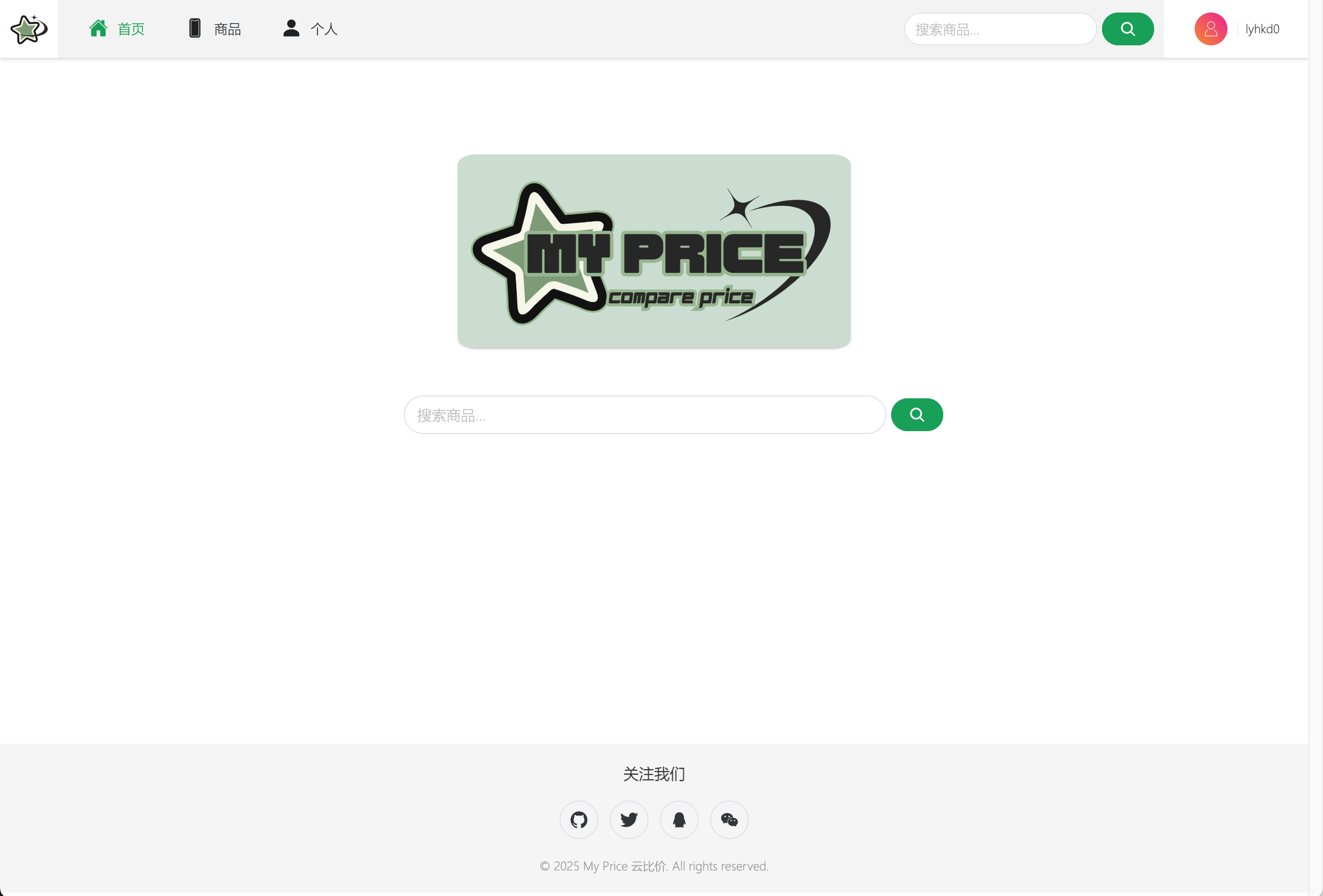
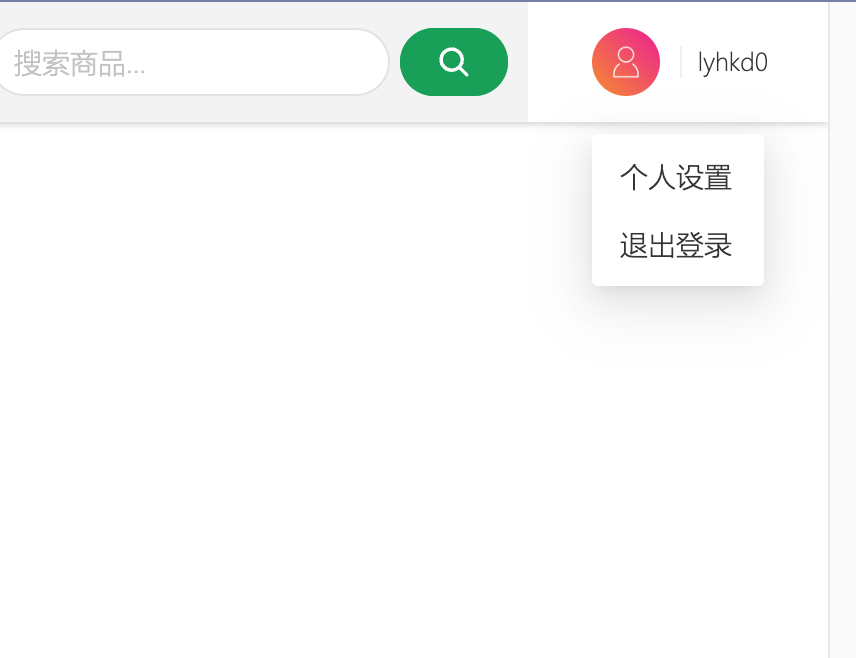
在注册页输入合法的用户名、邮箱、密码可以实现注册

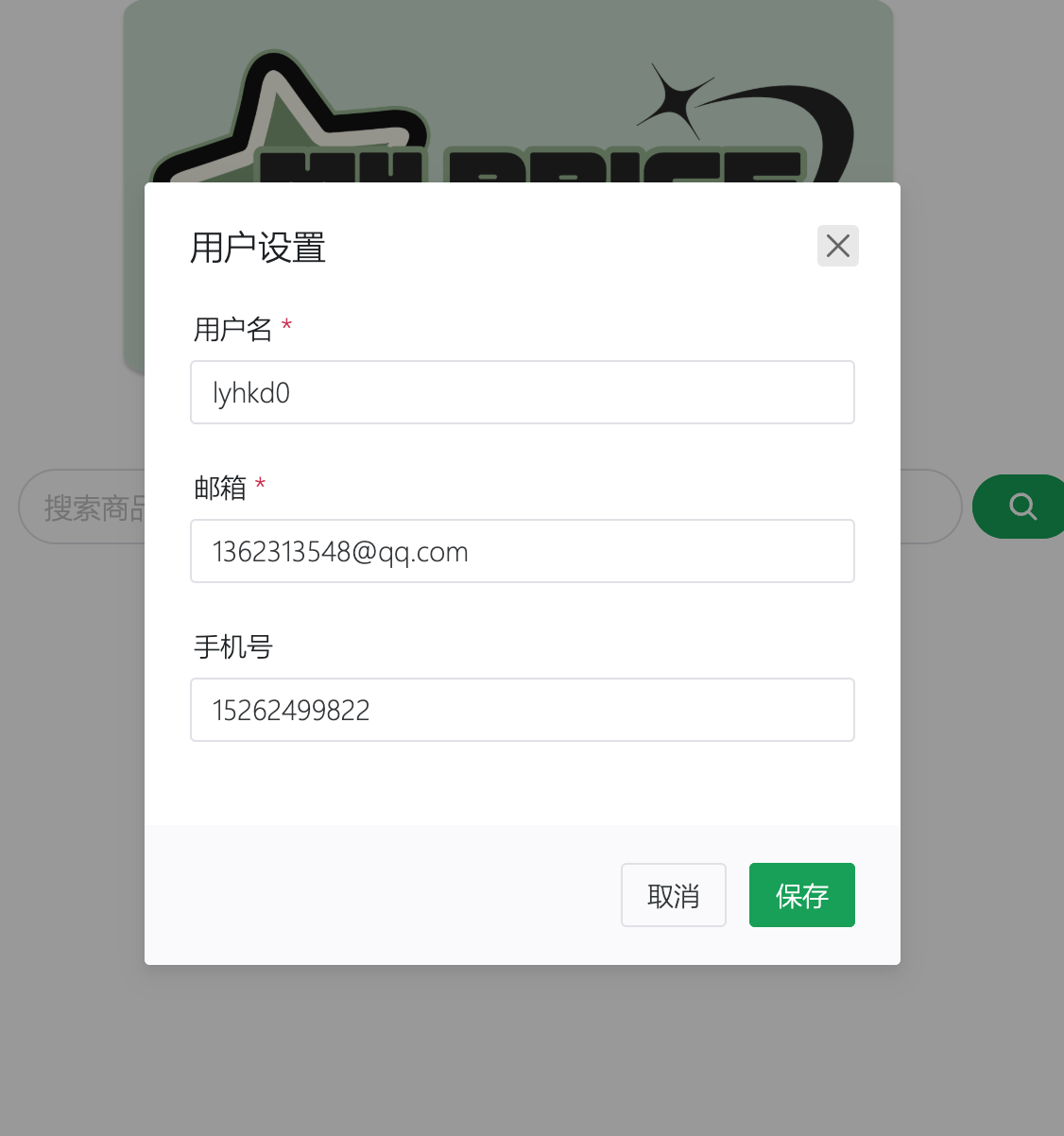


在用户登录页面输入注册的账户信息即可登录。这里可以使用已经注册过的测试账户进行测试（账号：lyhkd0，密码：123456）



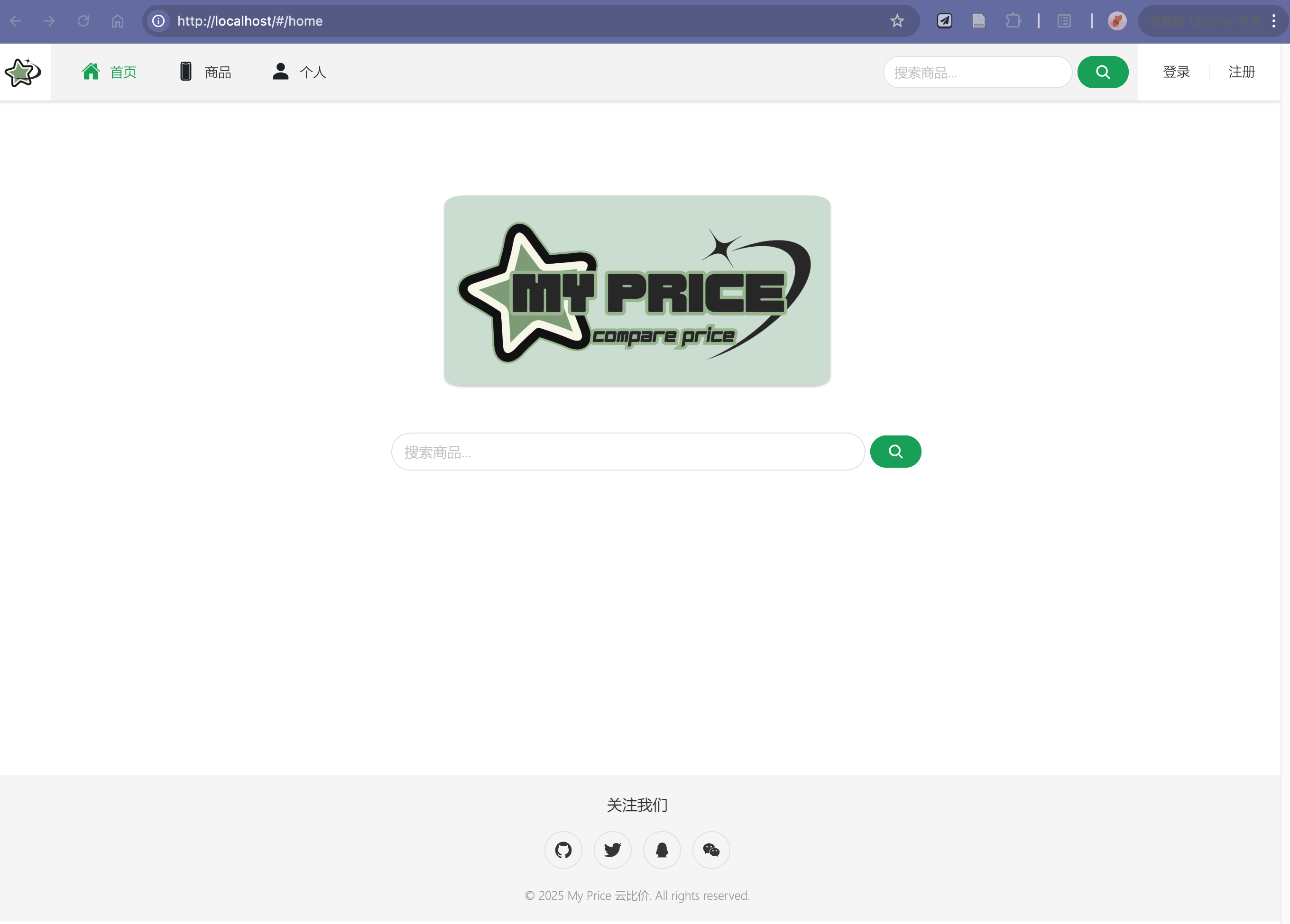
登录后个人信息会显示在右上角，并且可以收藏商品。可以通过个人设置修改账户内容。





#### 商品搜索

在网站的home页可以直接进行商品搜索



或者通过header访问商品页面。

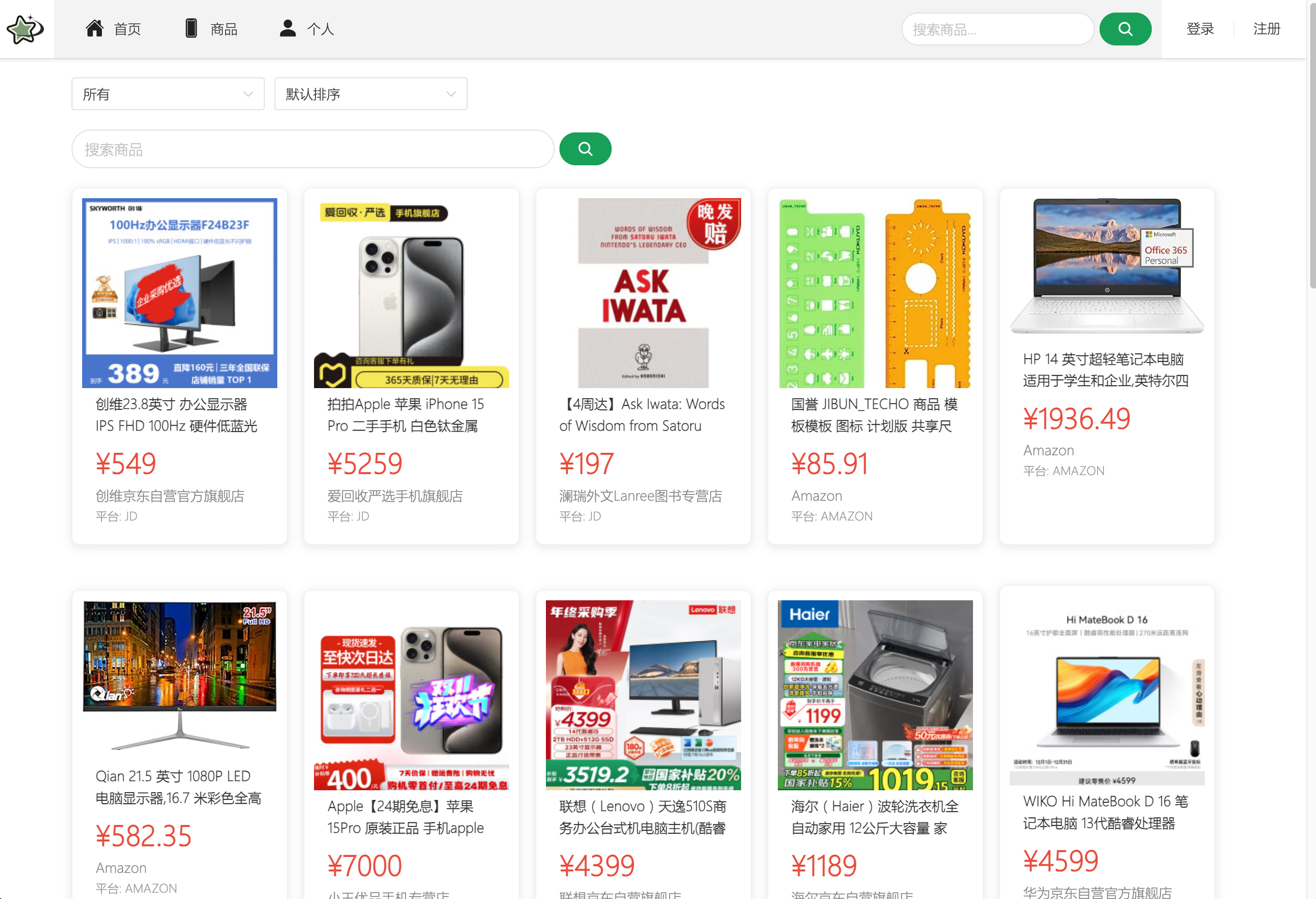
如果没有搜索关键词，平台会随机展示一些商品。

在搜索结果页可以选择平台、价格进行筛选排序。

如果搜索结果在数据库内没找到相关信息也会随机展示商品。

如果发现搜索结果和预期不符，可以等待一会再次搜索以检查后台运行的爬虫是否更新数据库。

不可在短时间内进行多次搜索！！！否则爬虫可能失效！！！！



#### 浏览商品

在商品展示页会显示商品的详情、价格趋势图等。

（第一次加载商品时如果详情为空，**可以等待一段时间后刷新**。查看从网站爬取的价格和商品详情）。

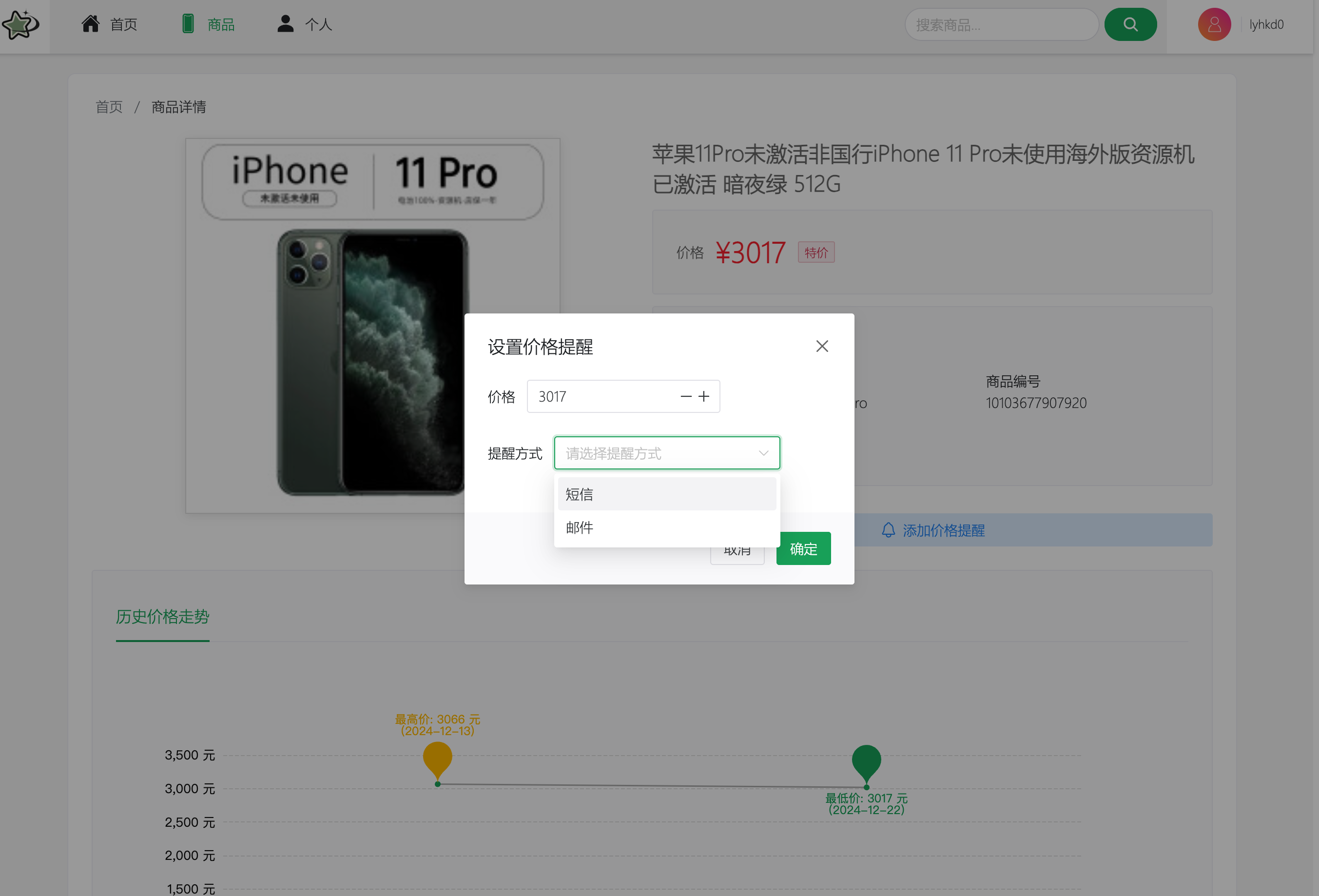
在每次搜索或者进入商品详情页后都会更新历史价格数据。





点击添加价格提醒即可加入收藏夹（如果未登录会跳转到登陆界面）

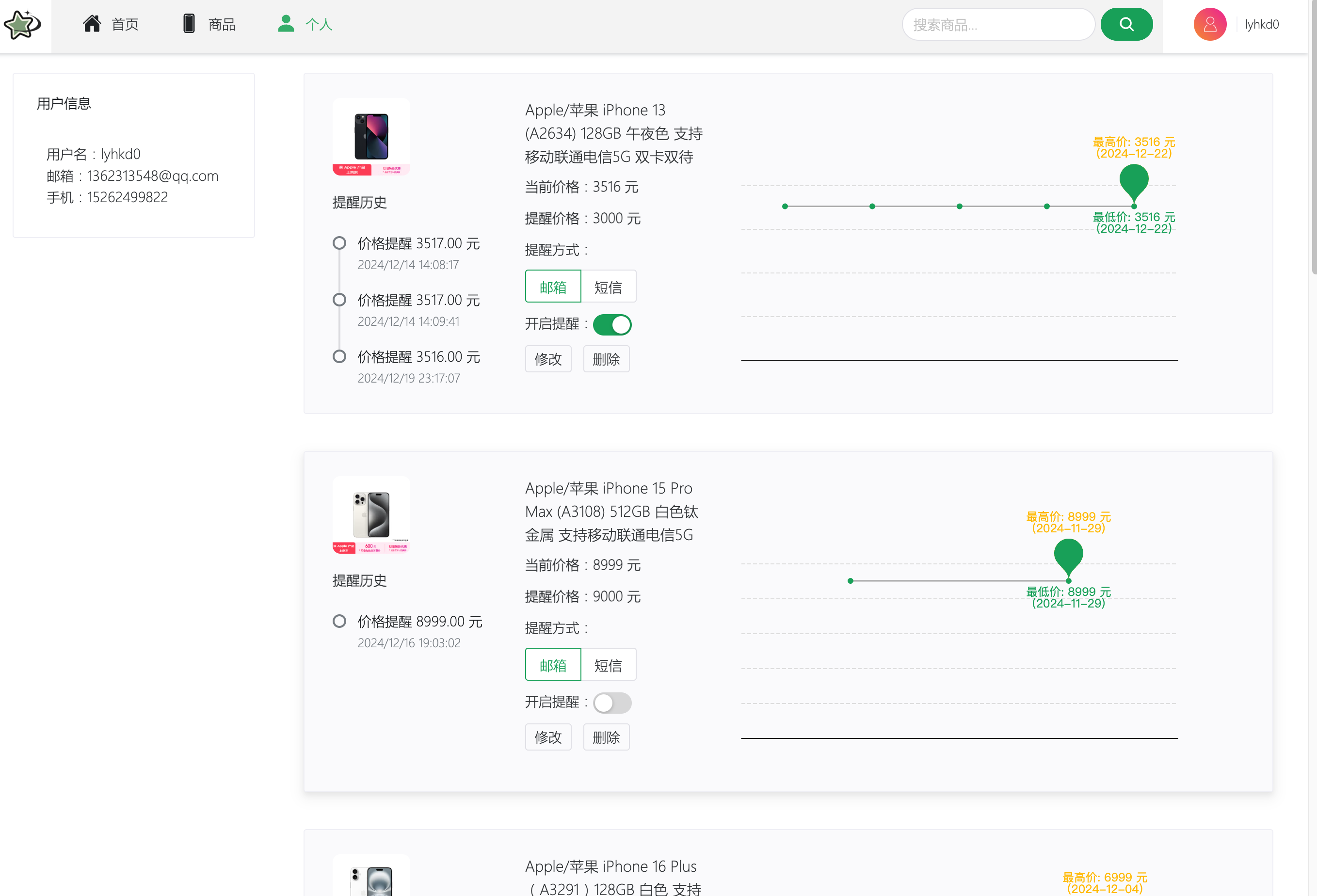
在设置价格提醒中可以选择价格的具体提醒点以及提醒方式。

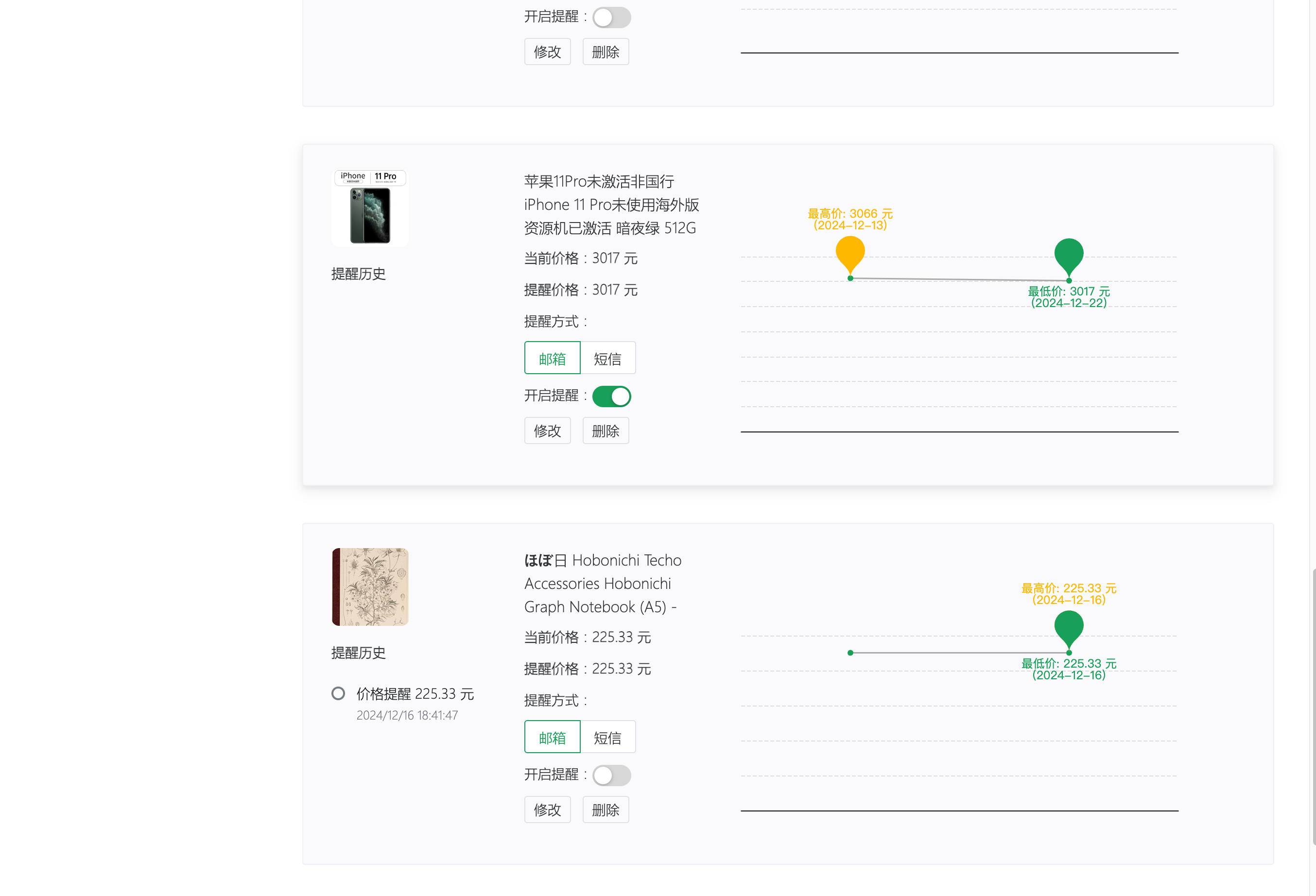


#### 查看收藏商品

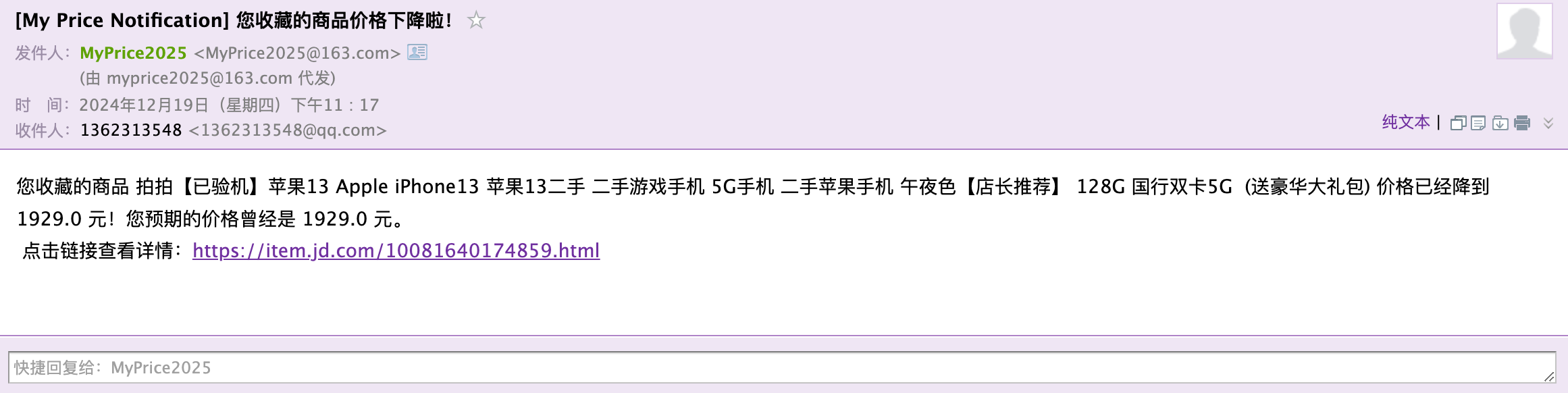
添加成功后会在个人页面展示所有添加的商品并且允许修改提醒方式/价格/是否启动提醒等。

对于已经提醒过的商品会显示提醒的历史记录。





价格波动后会在邮箱或短信收到提醒。





### docker使用方法

构建镜像：docker-compose build  
启动容器：docker-compose up  
后台运行容器：docker-compose up -d  
停止容器：docker-compose down  
查看服务状态：docker-compose ps