



第一阶段评审报告



汪金武 张鑫豪 许森坤 邹耀瑶



目录

CONTENTS

1

项目介绍

2

任务分配

3

项目总体规划

4

基线需求分析

5

成果展示



耳机购物平台

项目简介

任务分配

项目总体规划

基线需求分析

成果展示

项目简介

本在线购物系统是一个基于软件工程原则和现代Web开发技术的简单而可用的电子商务平台。该项目旨在提供一个可靠的购物体验，同时保持代码质量、可维护性和安全性。

项目特点

分层架构：我们采用了分层架构，将应用程序分为前端、后端和数据库层，以提高可维护性和扩展性。

技术堆栈：前端采用现代JavaScript框架Vue.js构建，后端使用Node.js、Springboot，数据库使用MySQL





耳机购物平台

项目简介

任务分配

项目总体规划

基线需求分析

成果展示

项目特点

用户身份验证：用户可以注册、登录、注销，并使用密码加密，以确保数据的安全性。

自动化测试：我们编写了单元测试和集成测试，以确保代码质量和功能的可靠性。

安全性：采用HTTPS、数据加密等安全措施，以保护用户数据和交易安全。

用户友好的界面：用户界面经过设计和用户测试，以确保用户友好的购物体验。

团队协作：开发团队采用版本控制系统（Git）进行协作，遵循敏捷开发和持续集成原则。





项目简介

任务分配

项目总体规划

基线需求分析

成果展示

项目负责人：汪金武

制定项目计划，包括时间表、任务和里程碑。
协调团队成员的分工，与团队成员进行定期会议，
讨论进展和解决问题，确保项目目标的实现。
管理项目进度，跟踪任务的完成情况，识别潜在的
问题和风险。
承担部分开发、测试。

后端开发：张鑫豪

构建和维护后端服务器，处理前端的请求并与数据
库进行交互。
设计和实现数据库模型，包括数据表的创建和关系
的定义。
编写数据库查询和存储过程，以支持应用程序的功
能。
负责用户身份验证和订单处理。

前端开发：许森坤

负责设计和实现用户界面，包括商品列表、购物车、
结账等页面。
处理用户交互和界面响应。
与后端开发者合作设计和使用API，以访问数据库。

测试工程师：邹耀瑶

负责执行数据库性能测试，以识别潜在的性能问题。
协助前端和后端开发团队进行单元测试和集成测试，
以确保数据库相关的功能正常。
提供反馈和建议，帮助团队解决问题。



项目总体规划

项目简介

任务分工

项目总体规划

基线需求分析

成果展示

第3次课	第6次课	第9次课	第12次课	第15次课
制定项目总体规划，完成分工； 基线 需求的需求分析，用例编写； 学习vue、nodejs、等开发技术； 学习git的使用； 部分开发。	完成基线需求及挑战需求	完成升级需求包A； 在远程服务器上完成部署。	完成升级需求包B	整理成果，撰写文档，准备验收。



项目简介

任务分工

项目总体规划

基线需求分析

成果展示

业务功能需求

功能划分

按照用户角色分为两个部分：买家：查看商品、下单商品、填入买家信息；卖家：查看购买意向、账号管理、商品管理。



面向买家

- 查看商品：买家无需注册账号即可查看商品的所有信息。
- 下单商品：买家有购买意向时，可填入个人信息表示自己的购买意向。
- 填入用户信息：在买家点击购买商品后，跳转到此页面，在这里填写买家的姓名，电话，交易地址信息



面向卖家

- 账号管理：账号和密码的验证；修改密码功能。
- 商品管理：卖家可以在系统上发布商品、冻结商品、撤下商品、恢复冻结的商品。
- 查看购买意向：商家可以查看有意向的买家。



项目简介

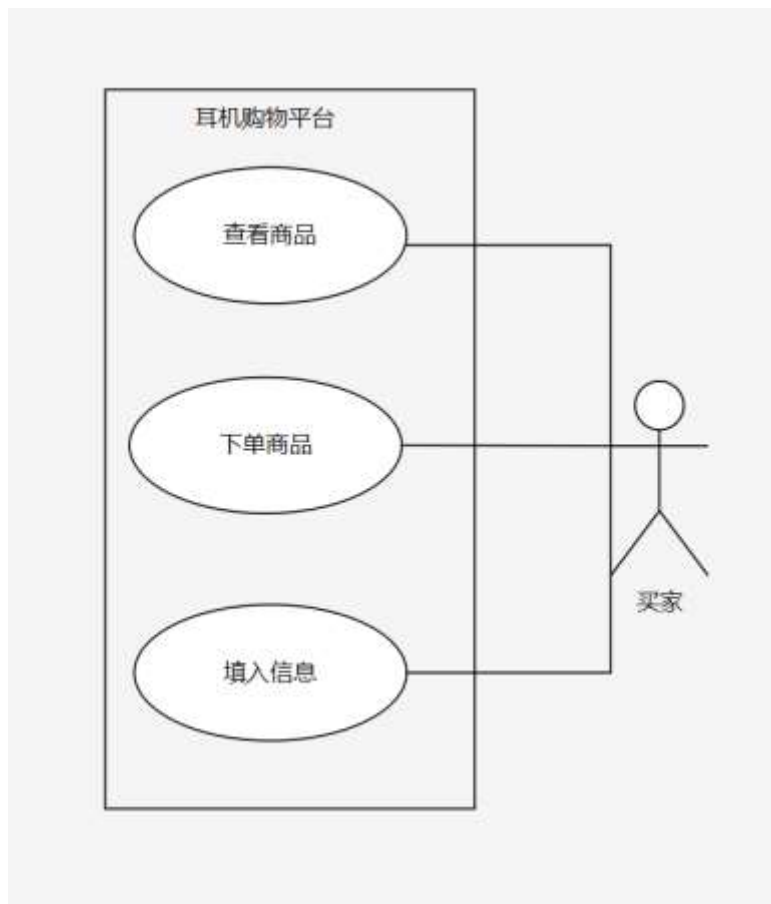
任务分工

项目总体规划

基线需求分析

成果展示

用例图展示





项目简介

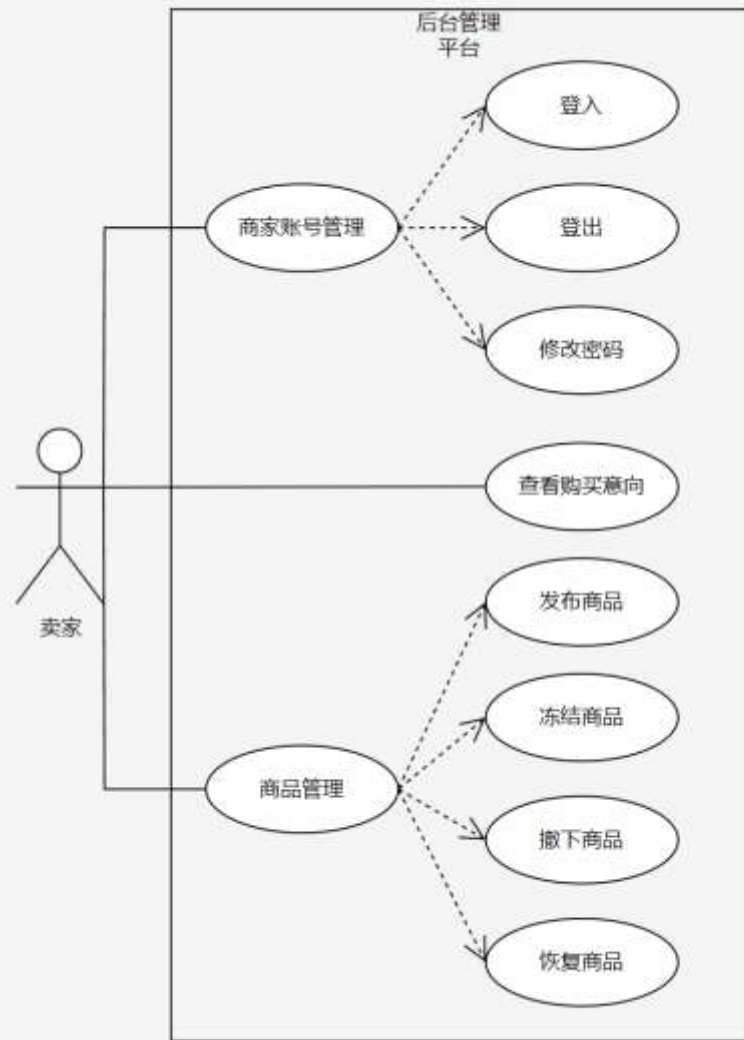
任务分工

项目总体规划

基线需求分析

成果展示

用例图展示





项目简介

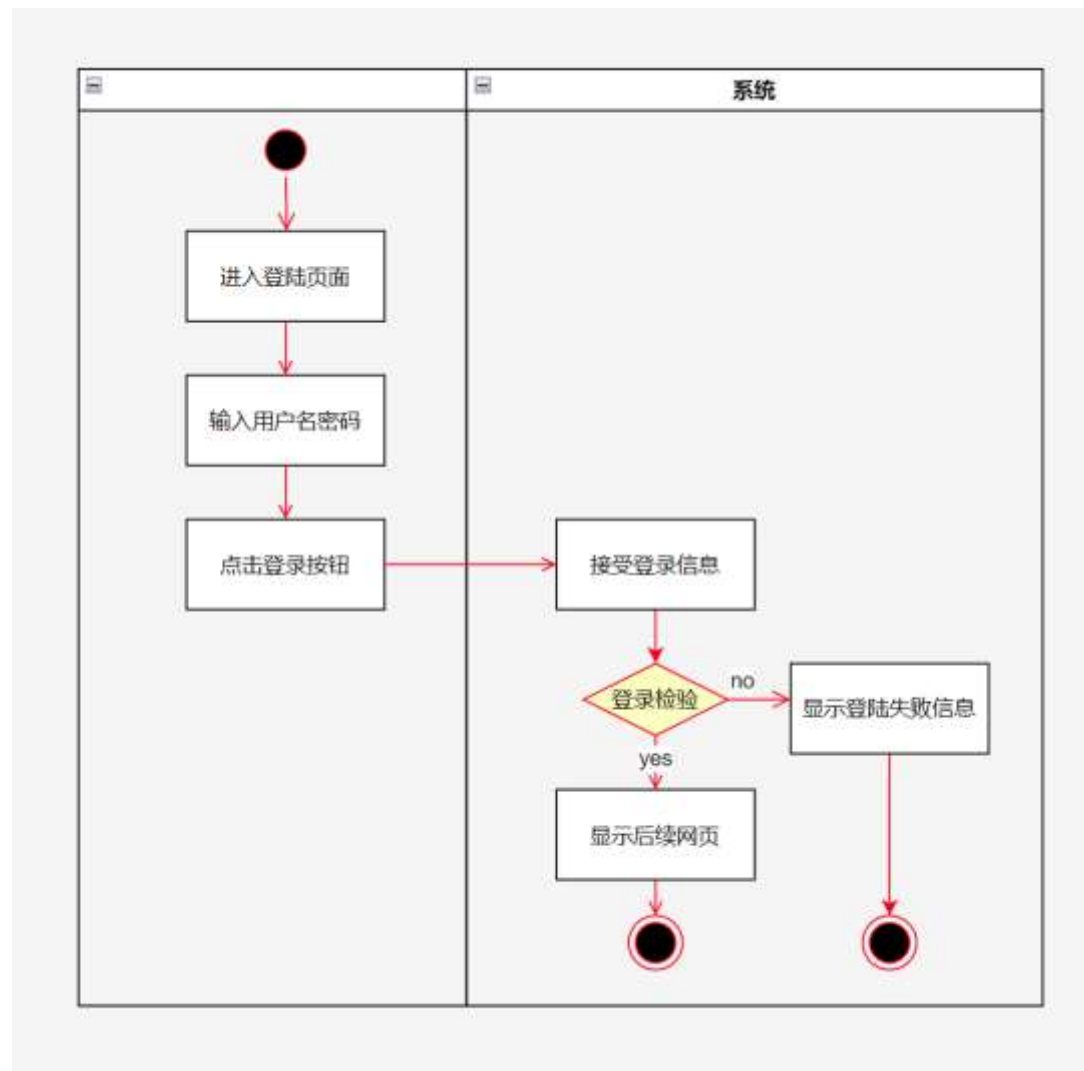
任务分工

项目总体规划

基线需求分析

成果展示

登录功能流程图





项目简介

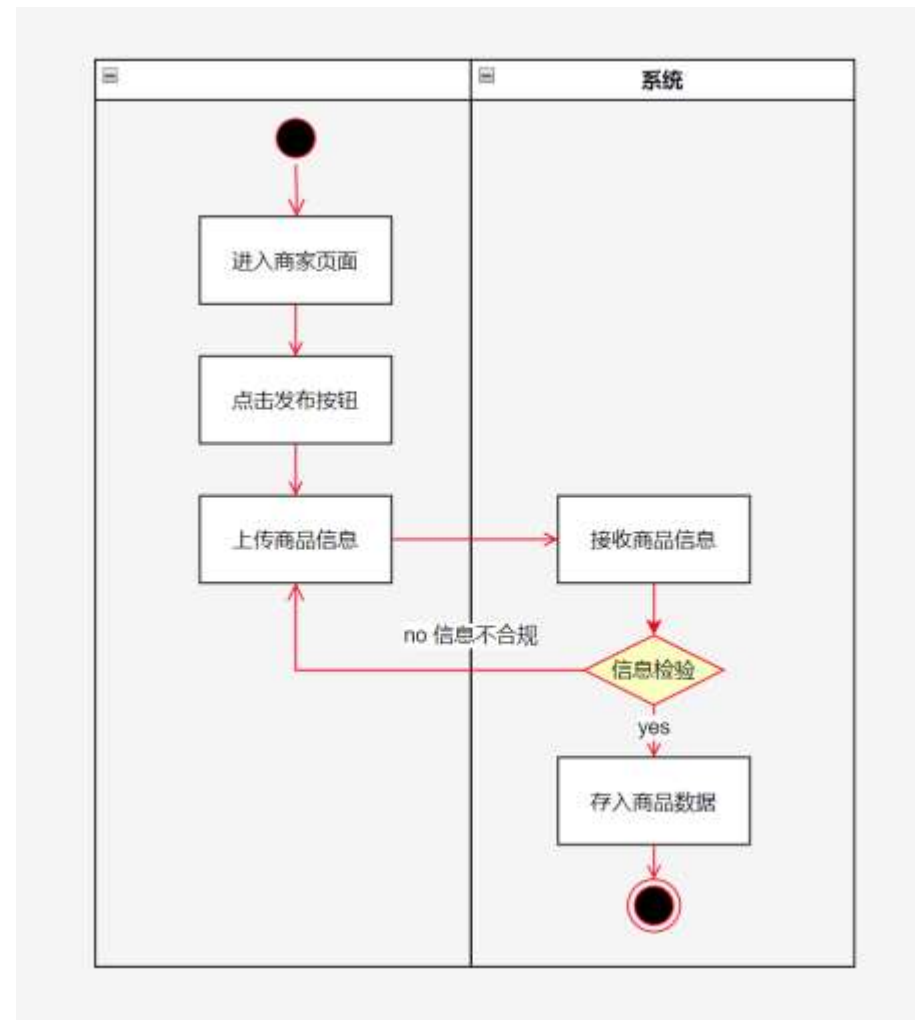
任务分工

项目总体规划

基线需求分析

成果展示

发布商品流程图





项目简介

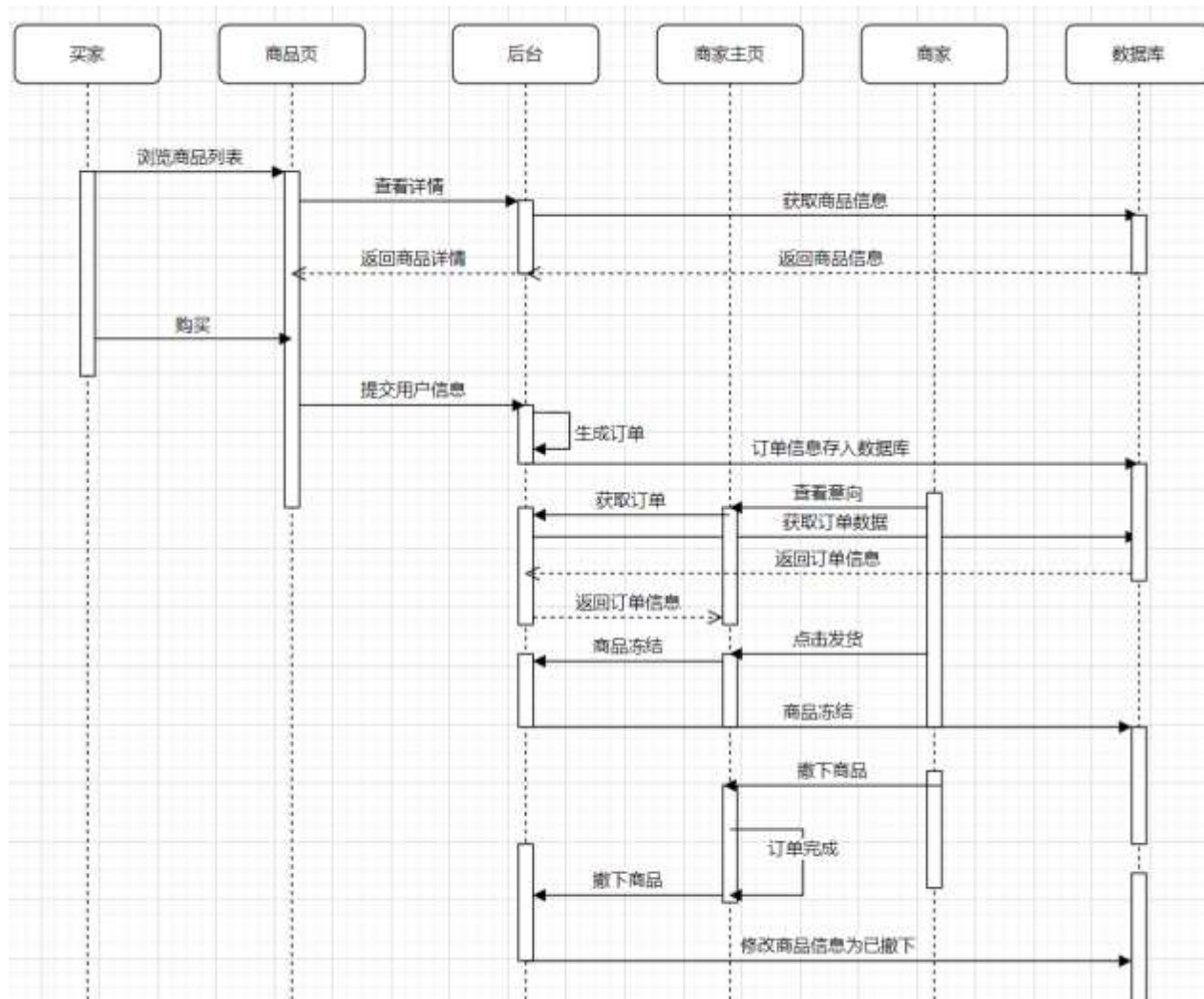
任务分工

项目总体规划

基线需求分析

成果展示

购买商品序列图





项目简介

任务分工

项目总体规划

基线需求分析

成果展示



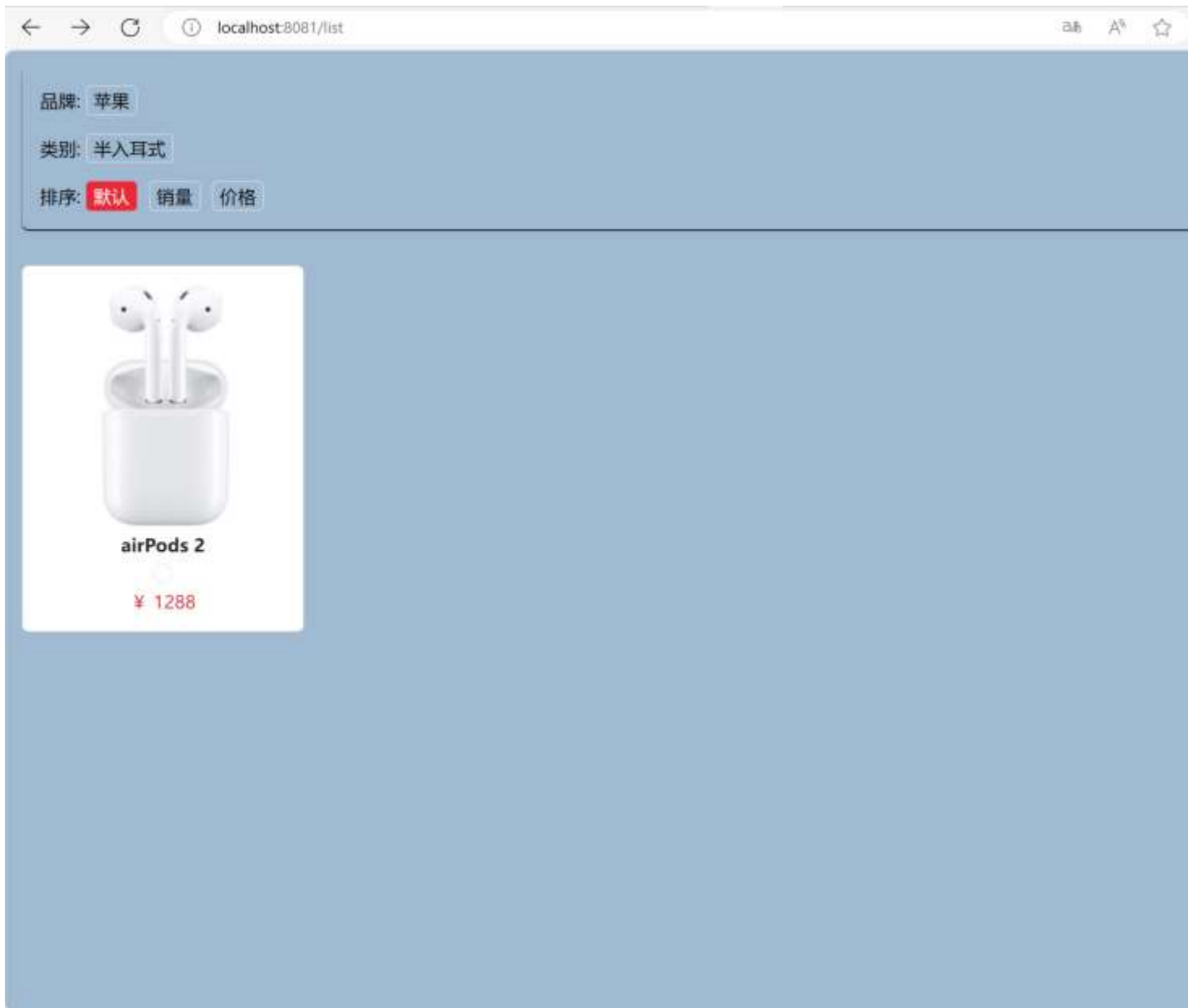
项目简介

任务分工

项目总体规划

基线需求分析

成果展示





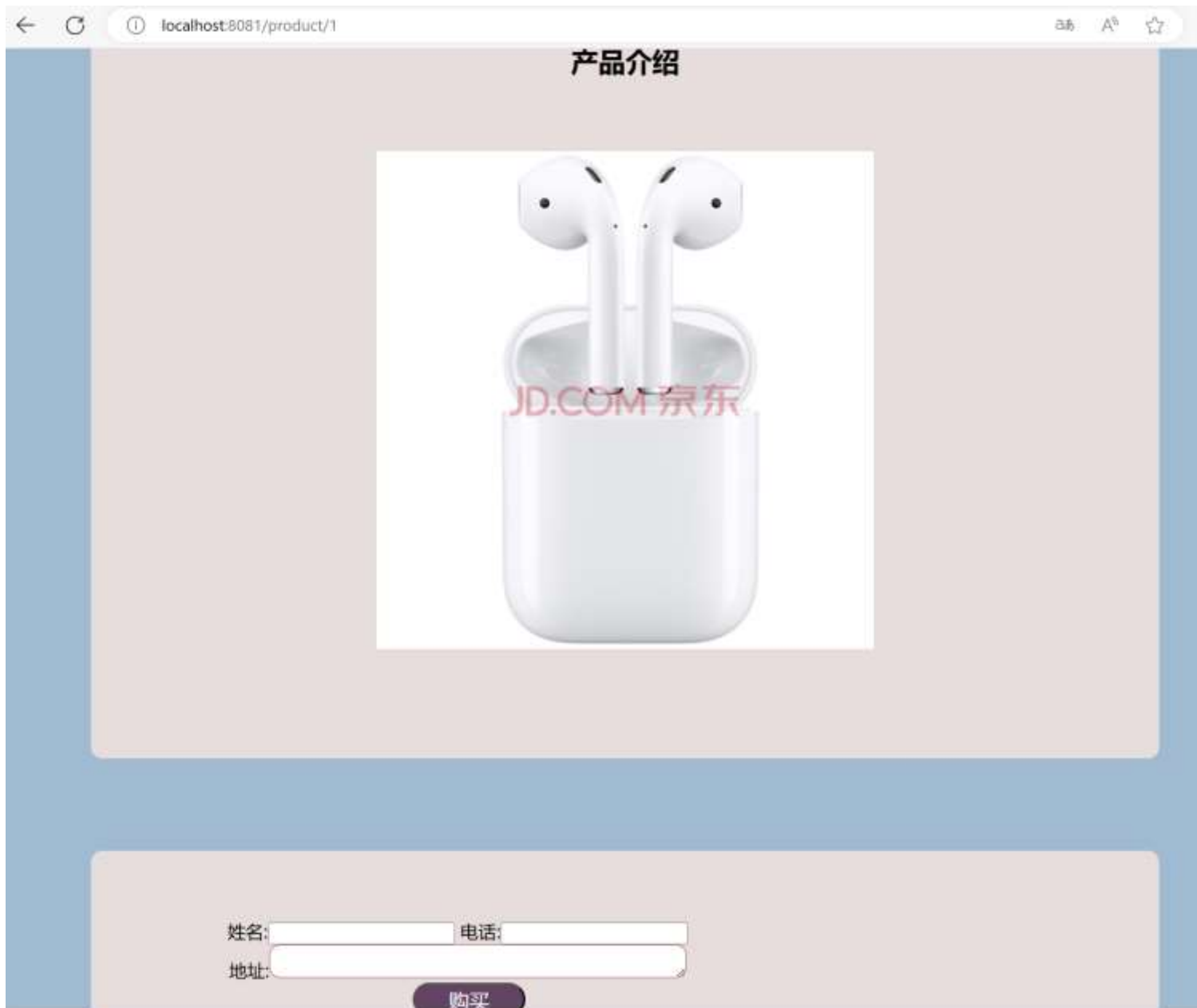
项目简介

任务分工

项目总体规划

基线需求分析

成果展示





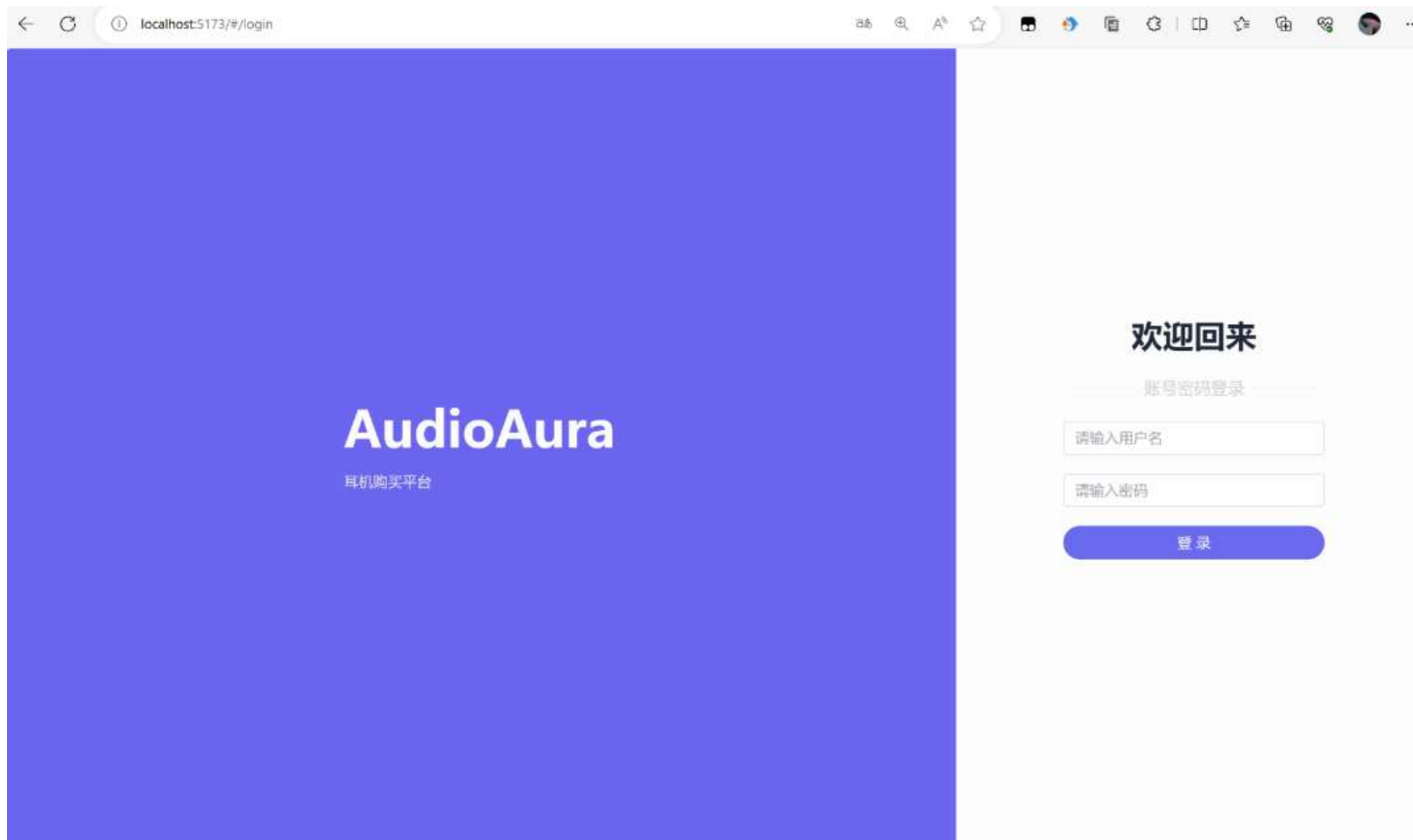
项目简介

任务分工

项目总体规划

基线需求分析

成果展示





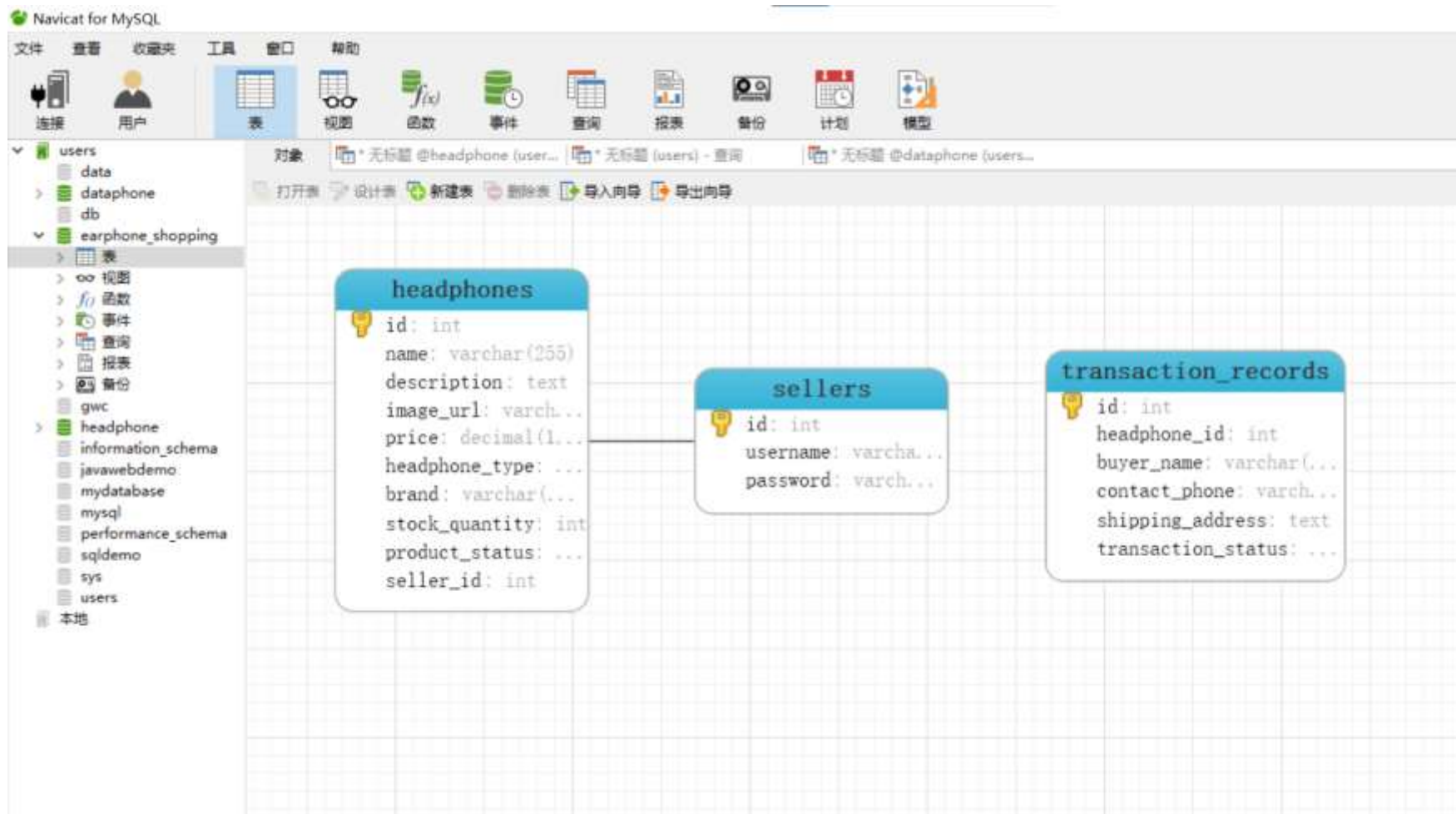
项目简介

任务分工

项目总体规划

基线需求分析

成果展示





项目简介

任务分工

项目总体规划

基线需求分析

成果展示

```
GoodsService.java x GoodsServiceImpl.java x Goods.java x Headphones.java x
1 package com.example.springbootproject01.bean;
2
3 import java.math.BigDecimal;
4
5 public class Headphones {
6     private Integer id;
7     private String name;
8     private String description;
9     private String image_url;
10    private BigDecimal price;
11    private String headphone_type;
12    private String brand;
13    private Integer stock_quantity;
14    private Integer seller_id;
15
16    public Integer GetSeller() {
17        return seller_id;
18    }
19    public Integer GetHeadphonesId() {
20        return id;
21    }
22    public String GetHeadphonesBrand() {
23        return brand;
24    }
25
26    public String GetImage_url() {
27        return image_url;
28    }
29    public String GetDescription() {
30        return description;
31    }
```



项目简介

任务分工

项目总体规划

基线需求分析

成果展示

```
GoodsService.java x GoodsServiceImpl.java x
1 package com.example.springbootproject01.service;
2
3 import com.example.springbootproject01.bean.Address;
4 import com.example.springbootproject01.bean.Goods;
5
6 import java.util.List;
7
8 public interface GoodsService {
9     Boolean update(Goods goods);
10    Integer deleteById(Integer id);
11    Integer insert(Goods goods);
12    Integer updateAddNum(Integer id);
13    Integer updateSubNum(Integer id);
14    List<Goods> getAll();
15 }
16
```

```
GoodsService.java x GoodsServiceImpl.java x
1 package com.example.springbootproject01.service.impl;
2
3 import ...
4
5 @Service
6 public class GoodsServiceImpl implements GoodsService {
7     @Resource
8     private GoodsMapper goodsMapper;
9
10    @Override
11    public Integer insert(Goods goods) { return goodsMapper.insert(goods); }
12
13    @Override
14    public Integer updateAddNum(Integer id) { return goodsMapper.updateAddNum(id); }
15
16    @Override
17    public Integer updateSubNum(Integer id) { return goodsMapper.updateSubNum(id); }
18
19    @Override
20    public Boolean update(Goods goods) { return goodsMapper.update(goods); }
21
22    @Override
23    public Integer deleteById(Integer id) { return goodsMapper.deleteByPrimaryKey(id); }
24
25    @Override
26    public List<Goods> getAll() { return goodsMapper.getAll(); }
27 }
```

THANK YOU

谢谢大家



汪金武 张鑫豪 许森坤 邹耀瑶