

线上商城系统验收报告

软工8组

开发人员：顾田、邢益鸣、戴威扬、孙龙灵、赵晚甜

汇报人员：戴威扬

1 领域背景和存在问题

电子商务的迅猛发展

01

全球电子商务概述

电子商务在全球范围内迅猛发展，特别是在互联网普及率高的地区。电子商务平台提供便捷的购物方式，涵盖广泛的商品种类。

02

知名平台的影响

亚马逊、阿里巴巴和京东等平台不仅改变了购物方式，还推动了物流和支付等相关行业的发展。它们为消费者提供了更广泛的选择。

03

线上商城的优势

线上商城系统适用于各行各业，为商家提供了便捷的销售渠道。消费者可以通过互联网轻松浏览和购买商品。



当前线上商城系统的问题

推荐页面冗长繁琐

许多线上商城首页堆砌大量推荐商品，页面显得冗长且杂乱无章。用户需要不断滚动页面，浏览效率低。

核心功能位置不合理

一些商城将核心功能放在推荐商品清单之后，用户必须滚动到页面底部才能看到。关键功能被隐藏增加了操作负担。

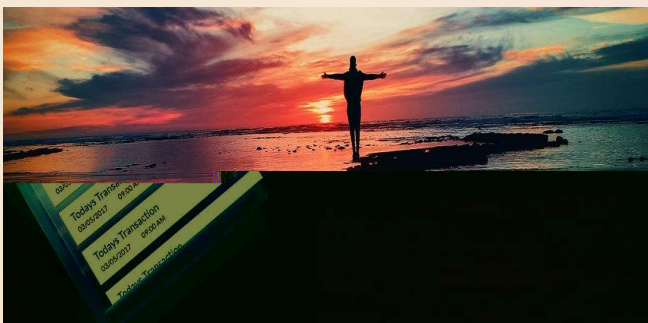
信息呈现不清晰

商品信息描述复杂冗长，用户需要仔细阅读才能获取关键信息。信息不清晰导致用户无法迅速了解商品特点。

2 用户需求及系统目标

用户需求分析

界面设计与用户体验



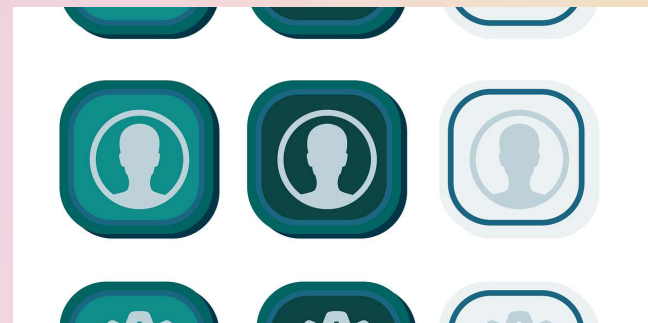
简洁清晰的界面

用户希望商城界面简洁明了，信息呈现清晰，能够快速找到所需商品和功能。



高效搜索导航

用户希望能够快速搜索到所需商品，并通过简便的导航找到感兴趣的商品类别。



便捷购物流程

用户希望购物流程简单流畅，包括商品选择、加入购物车、订单确认和支付等环节。

个性化与订单管理



个性化推荐

用户希望根据自己的浏览和购买历史，获得个性化的商品推荐，提升购物体验。



订单管理功能

消费者希望看到自己过去一段时间的购买记录和浏览记录，方便管理和查询。

PayA

系统目标

简洁明了的用户界面



设计简洁界面

设计简洁、直观的用户界面，使用户能够轻松浏览和操作，提高使用满意度。



优化布局内容

通过优化布局、精简页面内容和突出关键信息来实现简洁明了的界面设计。

高效搜索和流畅购物



精准搜索算法

开发精准的搜索算法和直观的导航菜单，帮助用户快速找到所需商品，提升搜索效率。



优化购物流程

简化购物车、订单确认和支付流程，减少用户操作步骤，提供无缝的购物体验。

订单管理与可扩展性



完善订单管理

提供订单管理功能，方便用户掌握订单信息，提升购物体验和用户满意度。



系统可扩展性

通过模块化设计提高系统的可扩展性，确保系统能够灵活应对未来的变化和扩展。

3 系统开发过程中的软件工程方法



计划阶段的应用

01

需求分析联合工作小组

通过成立需求分析联合工作小组确定项目总体方向。包括产品经理、开发人员、测试人员和用户代表的多方合作。

02

甘特图和任务网络图

使用甘特图展示项目时间计划和任务进度，使用任务网络图展示任务依赖关系。工具帮助跟踪项目整体进度和优化时间安排。

分析阶段的应用

01

用例图描述系统功能

用例图用于描述系统的功能需求和用户交互。帮助明确用户的具体需求和系统应提供的功能。

02

ER图和数据流图

ER图定义系统中的主要实体及其关系，数据流图描述系统中数据的流动和处理过程。确保数据库设计的合理性和数据处理逻辑。

。

03

状态转化图

状态转化图描述系统中各个对象在不同状态下的行为和状态转移。帮助理解系统的动态行为和状态管理。

设计和实现阶段的应用



类图和顺序图

类图描述系统中的类及其属性和方法，顺序图描述对象间的交互顺序和消息传递。确保系统的静态结构和动态交互。



风险管理计划

通过识别、分析和应对项目中的潜在风险，制定应对措施。确保项目的顺利进行和高质量交付。



持续集成和代码重构

持续集成确保每次代码更新都不会破坏系统功能，代码重构提升代码可读性和维护性。核心实践确保代码质量和系统稳定性。



灵活的需求响应

通过极限编程方法快速响应用户需求变化，持续交付高质量软件。频繁用户反馈和改进优化系统性能和用户界面。

4 面临的挑战和解决方案

面临的挑战和解决方案



系统的可扩展性

系统需要具备良好的可扩展性，以应对用户数量和业务需求的增长。通过模块化设计提高系统的可扩展性。



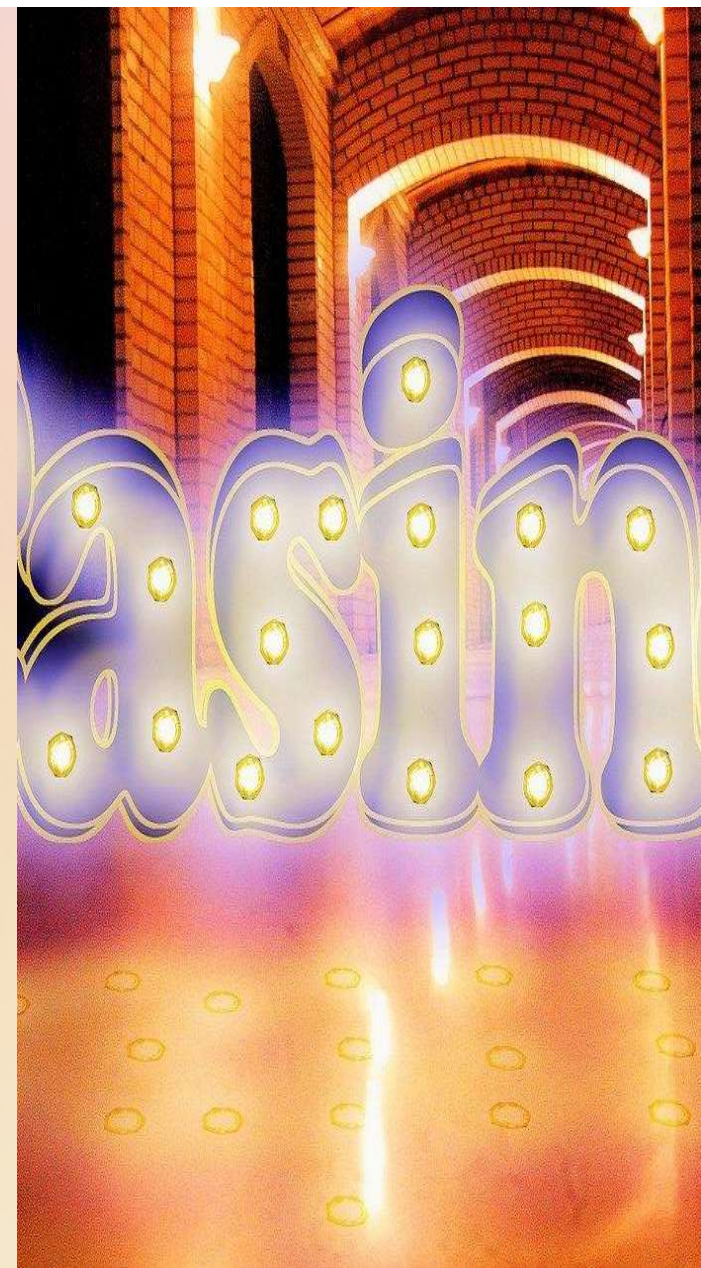
技术困难的解决

部分成员不熟悉前后端知识，导致开发速度较慢。通过学习和经验分享，及时解决技术问题，提升开发效率。



项目管理与团队协作

实践课程较多，成员时间协调困难。通过讨论和合理分工，确保项目按时完成，提高协作效率。



成员不熟悉前后端开发知识，临近期末任务重。

解决方法：

- 调整开发计划，开发阶段先开发首页、下单功能、购物车功能等核心功能。
- 边开发边学习，先尝试实现功能，遇到问题再查看教程和文档。
- 前后端开发同时进行。

成员不熟悉前后端开发知识，临近期末任务重。

- 在Github仓库中添加相关教程方便查看。



两种编码风格：Vue2和Vue3

```
1 usage  yks1+2
33 export default {
34   components: { Detail },
35   data() {
36     return {
37       searchQuery: '',
38       searchResults: [],
39       selectedProduct: null
40     };
41   },
42   methods: {
43     showDetail(product) {
44       this.selectedProduct = product;
45     }
46   }
47 }
```

OPTION API

```
<script setup>
> import ... // 引入 AddressModal 组件
> import ...

const store = useStore();
const username = computed( getter: () => store.getters.

const totalPrice = ref( value: 24.00);
const defaultAddress = ref( value: {}); // 存储默认地址

const addresses = ref( value: []); // 存储所有地址
const showModal = ref( value: false); // 控制弹窗显示
```

COMPOSITION API

前端开发过程中，不同开发成员选择了不同版本Vue的代码样式。虽然能正确运行，但是在最后整合页面时不易理解代码。

两种编码风格：Vue2和Vue3

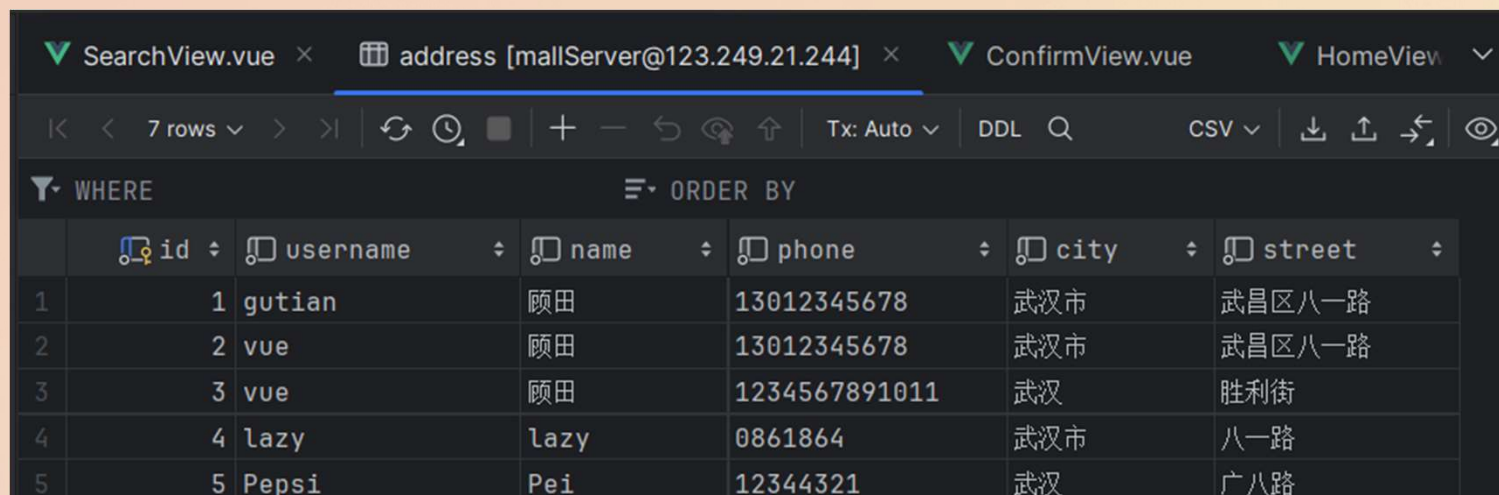
解决方法：

- 有必要时，小组成员自行查看另一种API风格的相关文档或博客理解代码，必要时借助ChatGPT。
- 在之后的开发中应提前约定使用的代码风格。

数据库配置问题

使用本地数据库开发时，多名成员在数据库连接上遇到问题。后端项目运行后不能正确访问数据库。

解决方法：在华为云购买一个数据库服务器，开发时共用一个数据库。



The screenshot shows a database management interface with a table containing 5 rows of data. The table has columns for id, username, name, phone, city, and street. The data is as follows:

	id	username	name	phone	city	street
1	1	gutian	顾田	13012345678	武汉市	武昌区八一路
2	2	vue	顾田	13012345678	武汉市	武昌区八一路
3	3	vue	顾田	1234567891011	武汉	胜利街
4	4	lazy	lazy	0861864	武汉市	八一路
5	5	Pepsi	Pei	12344321	武汉	广八路

后端数据处理逻辑复杂

后端在订单的处理上遇到问题。提交订单需要更新商品库存并计算商品总价，逻辑较为复杂。

解决方法：JavaScript的异步编程。

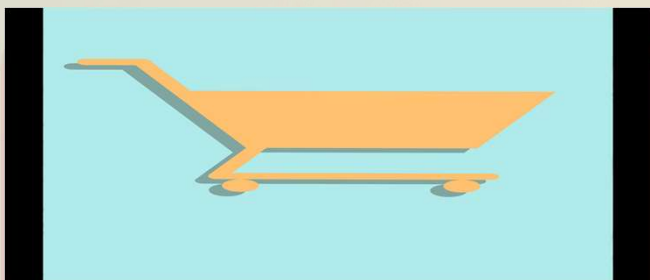
```
new Promise( executor: (resolve, reject) :void => {  
    // 创建订单, 返回订单id  
    pool.query(sql.createOrder, [username, address, orderTime], function (err : QueryError | null , result : T ) :void {...})  
}).then(function (orderId) : Promise<...> {  
    return new Promise( executor: function (resolve, reject){...}).then((Maps) : Promise<void> => {  
        return new Promise( executor: function (resolve, reject) :void {...}).then(() : void => {  
            // 返回订单id和总价  
            res.status( code: 200 ).json({msg: '创建订单成功'...})  
        }).catch(err => {...})  
    }).catch(err => {...})  
}).catch(err => {...})
```


5 系统满足用户需求的程度

系统功能演示

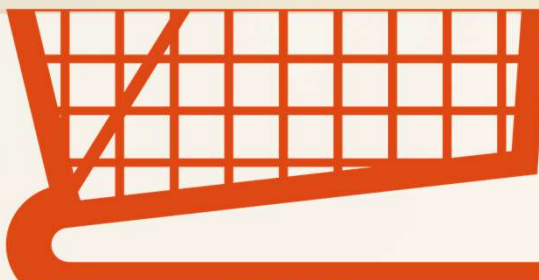
<http://34.81.194.255/>

系统满足用户需求的程度



商品浏览和搜索功能

用户可以方便地浏览所有可用商品，并通过搜索功能快速定位目标商品。界面设计简洁直观，提高用户体验。



购物车和支付功能

用户可以轻松管理购物车中的商品，完成订单确认和支付。购物流程简单流畅，用户体验良好。



用户注册和登录功能

系统提供用户注册和登录功能，确保用户数据的安全性和个性化体验。用户可以访问个人订单和地址信息。

6 开发总结

开发总结

01

开发过程概述

采用了极限编程 (Extreme Programming, XP) 模型, 确保高效的开发流程和灵活的需求响应能力。定期会议和结对编程提高代码质量和效率。

使用Git进行版本控制, 使用QQ群进行日常沟通。文档管理和共享确保所有成员方便访问和更新项目文档。

02

界面改进

目前界面较为简单, 后续可以持续完善美化界面从而提高用户使用体验感。

此外, 需考虑用户高频需求, 避免关键功能被隐藏, 提高用户体验。

03

功能改进

目前功能只实现了消费者方面的商城系统功能, 后续可以持续添加商家、客服以及管理员的功能。

此外, 目前的支付系统较为简陋, 后续可以考虑如何添加更多的支付方式, 为用户提供更多选择。

Thank you