**文本

描述已自动生成**

**卡通人物

中度可信度描述已自动生成**

**项目总体设计报告**

项目名称：校园二手交易平台

小组成员：胡奕晨 仇鑫宇 黄哲瀚

编写日期：2024年11月9日

目录

[一． 项目总体概述 - 2 -](#_Toc182249316)

[1.1项目背景 - 2 -](#_Toc182249317)

[1.2项目目的 - 3 -](#_Toc182249318)

[二．方案选择 - 3 -](#_Toc182249319)

[2.1方案一 - 3 -](#_Toc182249320)

[2.2 方案二 - 6 -](#_Toc182249321)

[2.3 方案选择 - 8 -](#_Toc182249322)

[四．项目功能分解 - 8 -](#_Toc182249323)

[4.1 系统功能概述 - 8 -](#_Toc182249324)

[4.2 功能模块详细分解 - 9 -](#_Toc182249325)

[五．软件架构与设计 - 13 -](#_Toc182249326)

[5.1 软件层次结构设计 - 13 -](#_Toc182249327)

[六．设计数据库 - 18 -](#_Toc182249328)

[6.1 数据库需求分析 - 18 -](#_Toc182249329)

[6.2 数据库概念模型设计 - 21 -](#_Toc182249330)

[八．书写文档计划 - 23 -](#_Toc182249331)

[8.1 系统说明 - 23 -](#_Toc182249332)

[8.2 用户手册 - 23 -](#_Toc182249333)

[8.3 测试计划 - 23 -](#_Toc182249334)

[8.3 详细实现计划 - 23 -](#_Toc182249335)

[8.4 数据库设计结果 - 23 -](#_Toc182249336)

[九．总结 - 23 -](#_Toc182249337)

[十．参考资料 - 23 -](#_Toc182249338)

[十一．小组组员评价 - 23 -](#_Toc182249339)

# 项目总体概述

## 1.1项目背景

我们小组发现每年新生入学和毕业生离校期间，大量的学习资料、生活用品等物品被丢弃或低价出售，造成了极大的资源浪费。以及在校大学生，有些闲置物品大量积累。这些闲置物品占用空间，且具有一定的价值，很多学生希望通过二手交易将其变现，从而节省空间并为自己带来一定的收益。但是由于缺乏一个专门的平台，他们往往只能在论坛，社交媒体上进行交易，这样往往不安全也不方便，校园二手交易平台出现是非常有必要的。

## 1.2项目目的

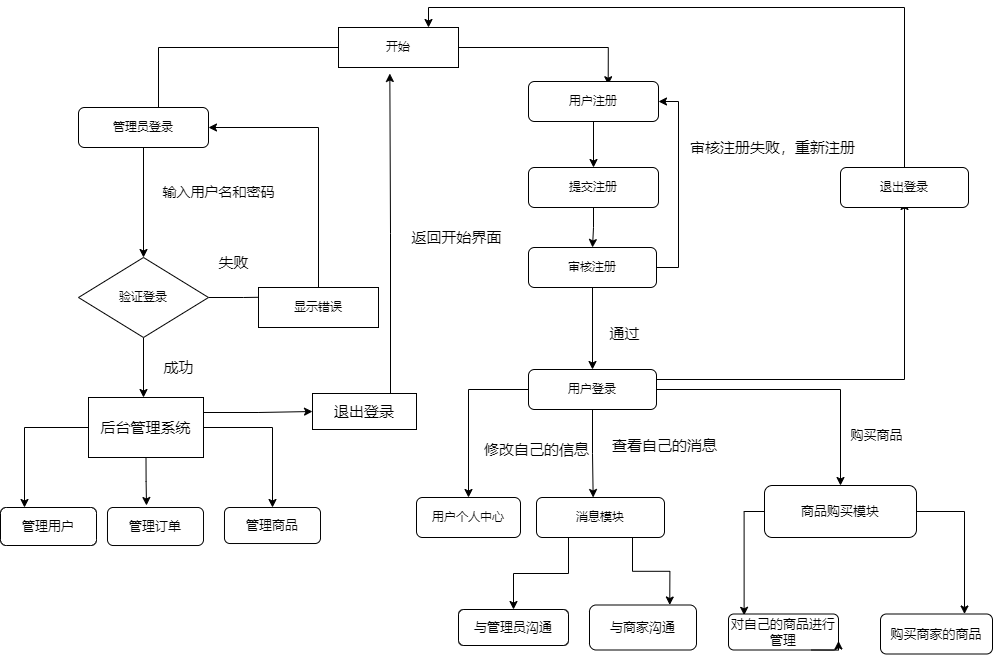
我们打算为在校大学生打造，创建一个安全可靠、操作简便的线上校园二手交易微信小程序，提高校园内闲置物品的再利用率，降低资源浪费。

# 二．方案选择

## 2.1方案一

我们前端技术选择前端采用 WXML 和 WXSS 技术，后端服务运用微信小程序云开发技术，数据库采用云数据库技术。

**1.系统流程图**



2.系统物理元素清单：

方案一在硬件设备方面，依托云服务器如阿里云 ECS 实例，为小程序后端服务提供稳定环境，同时学生通过智能手机等设备作为终端访问小程序。软件模块中，前端采用 WXML 和 WXSS 技术。WXML 作为微信小程序的标记语言，类似 HTML 但有特定的简洁语法和标签，能高效构建页面结构且支持数据绑定和条件渲染等功能。WXSS 则类似 CSS，可定义页面样式，为不同页面打造美观舒适的用户界面，提高操作体验。后端服务运用微信小程序云开发技术，包括云函数、云数据库和云存储。云函数用 Node.js 编写处理业务逻辑和数据库交互，云数据库是 NoSQL 数据库方便存储查询数据，云存储用于存储图片等资源。通过互联网连接和微信平台网络基础设施确保小程序正常访问和数据传输。由于主要依靠云服务和学生自有设备，无特定其他物理元素，具有成本低、灵活性高和易于维护的优势，能满足校园二手交易平台的基本需求。

3.成本效益分析：

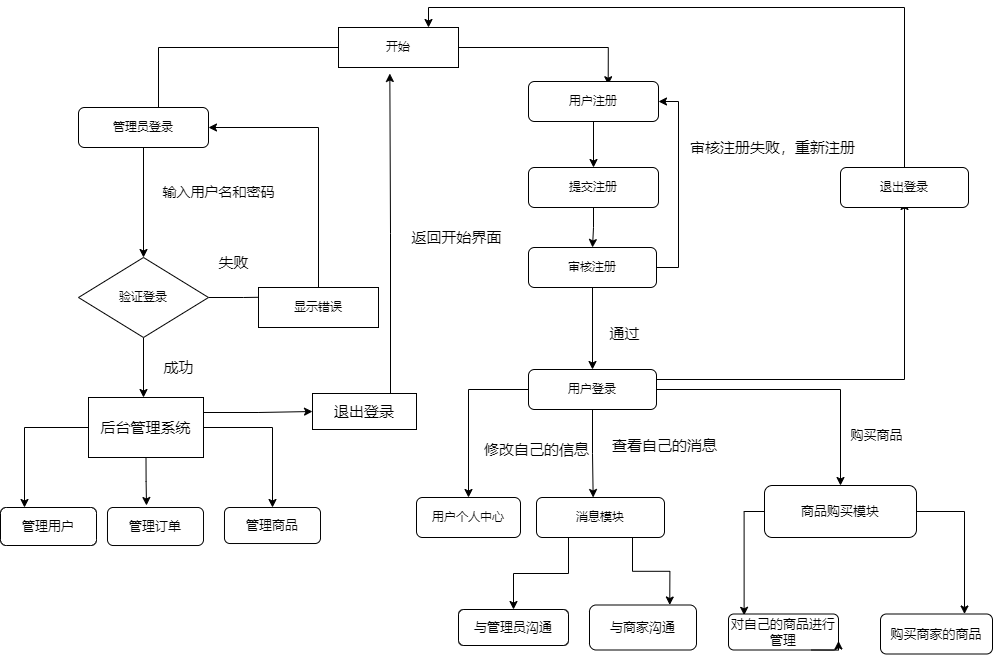
方案一中，成本方面具有明显优势。硬件成本主要是云服务器的租赁费用，这一成本相对可控且可根据实际需求进行灵活调整。对于学校来说，初期无需投入大量资金购买和维护实体服务器，极大地降低了硬件成本。开发成本方面，前端采用 WXML 和 WXSS 技术，这些技术对于有网页开发经验的开发者而言易于上手，减少了学习成本和开发时间，从而降低了整体开发成本。后端运用微信小程序云开发技术，无需搭建和维护传统服务器，进一步节省了成本。

从效益角度来看，清单一的灵活性极高。完全依托云服务意味着可以根据用户数量的变化迅速扩展或缩减服务器资源，不会受到硬件设备的限制。开发效率也较高，前端技术简单易懂，后端云开发简化了开发流程，能够加快小程序的上线速度，满足校园二手交易市场快速启动的需求。同时，云服务提供商负责服务器的维护和更新，大大减轻了运营方的维护工作量。此外，由于只要有网络连接学生就可以在任何地方使用小程序，不受校园地理位置的限制，实现了广泛的用户覆盖，为校园二手交易提供了便捷的平台，有助于提高交易效率和活跃度。

## 2.2 方案二

我们前端技术采用React.js + Ant Design技术，我们后端技术采用Django + Django REST framework技术，我们云数据库可以采用PostgreSQL技术。

**1.系统流程图**



2.系统物理元素清单：

方案二的硬件设备包括学校内部强大配置的校园服务器以及学生的多种移动设备等等。软件模块的前端采用 React.js 和 Ant Design 技术组合。React.js 的组件化开发模式可将复杂界面拆分为独立可复用组件，提高开发效率。Ant Design 提供丰富的 UI 组件，涵盖各种界面元素且设计美观、功能精心，为用户操作提供清晰指引。后端服务采用 Django 和 Django REST framework 技术，Django 的 MTV 架构模式明确分工，模型处理数据存储管理，模板负责页面展示渲染，视图连接两者处理用户请求。数据库管理系统选用 PostgreSQL，这是一个功能强大的开源关系型数据库，具有高可扩展性、稳定性和安全性，支持标准 SQL 语言及丰富扩展功能。通过校园网络和网络防火墙保障安全性，外部存储设备和打印机等其他物理元素满足数据备份和订单打印等需求，在性能、安全性和校园特色方面表现突出。

3.成本效益分析：

方案二在成本方面面临一定挑战。硬件成本较高，学校内部服务器的采购和维护需要投入大量资金，并且还需配置外部存储设备等，这使得初期投入较大。开发成本也不低，前端采用 React.js 和 Ant Design 技术组合，要求开发者熟悉这些较为复杂的技术栈，可能需要较高的技术水平和开发成本。后端使用 Django 和 Django REST framework 以及 PostgreSQL 数据库，同样需要专业的开发和维护人员，增加了人力成本。

方案二在效益方面表现出色。在性能和安全性方面，部分服务部署在校园内网的服务器上，能够显著提高访问速度和数据安全性，尤其对于涉及敏感信息的业务，如用户管理和交易记录等，提供了更可靠的保障。丰富的功能和良好的用户体验也是其优势之一，React.js 和 Ant Design 提供了丰富多样的 UI 组件，带来良好的交互体验，使小程序界面更加美观和易用。Django 和 PostgreSQL 具有强大的功能和扩展性，可以满足校园二手交易平台复杂的业务需求，如商品分类、搜索筛选、订单管理等。此外，校园服务器上配置的外部存储设备可以更好地控制数据备份和存储，确保数据的安全性和可靠性，为长期稳定运行提供了坚实基础。

## 2.3 方案选择

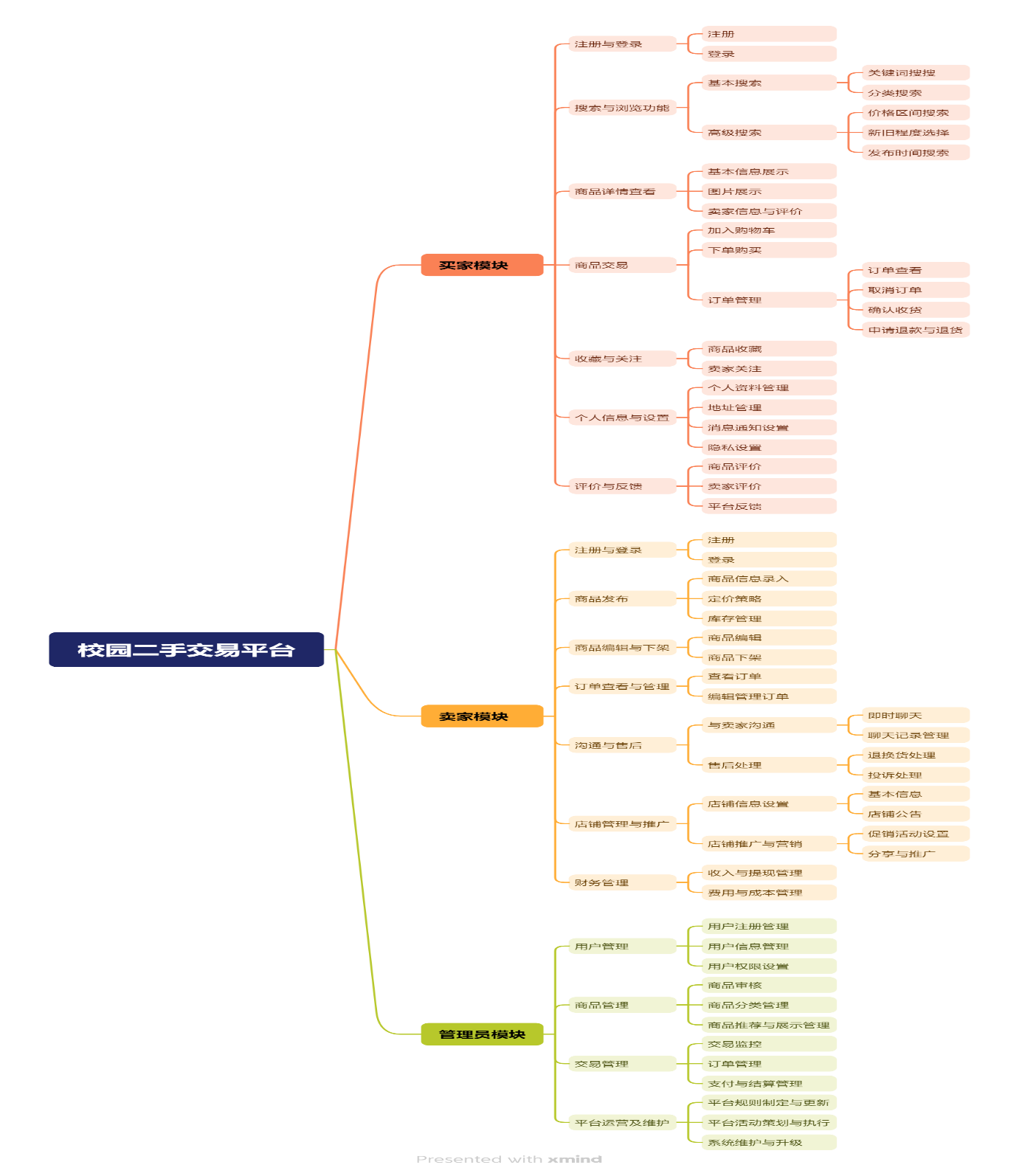
经过以上分析，两种方案都能满足我们的需求，可我们三个人之前都没有开发经验，制作时间也简短，考虑学习新技术的时间，由于方案二成本也比较高，方案一快洁高效，也能满足我们的的需求，所以我们选择方案一。

# 四．项目功能分解

## 4.1 系统功能概述

这系统涵盖买家购买、卖家出售和管理员管理三大模块。在买家购买模块，用户可通过手机号、邮箱注册登录或借助微信、QQ 等第三方平台登录。具备强大的搜索与浏览功能，包括关键字和分类搜索、高级筛选及分类展示。商品详情查看细致，支持多图展示和放大。能将心仪商品加入购物车进行集中管理，下单支付方式多样且可查看订单状态。收货后可关注商品及卖家并留下评价，还能管理个人信息和查看评价反馈以辅助购买决策。卖家出售模块中，与买家共用账号系统但注册需详细身份验证信息。可发布商品并选择分类，管理商品状态，进行编辑和下架操作，包括批量处理。能查看订单、更新状态并接收通知消息。处理售后问题和通过客服入口回复买家。可设置店铺、进行推广并查看统计数据，还能管理财务，包括收入结算、查看账单明细和提现。管理员管理模块则负责用户管理，审核新注册卖家和异常用户，可查看、编辑、禁用及封禁和解封用户账户。对商品进行审核、管理分类、处理违规商品。监控订单、处理投诉、统计交易数据。还负责平台设置及维护，包括广告位管理、公告发布以及系统升级与扩展。

## 4.2 功能模块详细分解



**买家购买模块**

**注册与登录**

用户可以通过手机号或邮箱注册并登录平台。

支持第三方平台登录，如微信或QQ。

**搜索与浏览功能**

**搜索框**：允许用户通过关键字或分类（如电子产品、书籍、生活用品等）搜索商品。

**高级筛选**：支持价格范围、成色筛选，方便用户快速找到心仪商品。

**分类浏览**：商品按类别、价格等进行分类展示。

**商品详情查看**

显示商品图片、标题、价格、成色、描述、卖家信息等。

支持商品图片的多图展示和放大查看。

**加入购物车**

用户可以将心仪商品加入购物车，便于集中管理。

支持对购物车内商品进行数量修改或删除。

**下单与支付**

用户在购物车中选择商品并下单。

支持多种支付方式（微信支付、支付宝等）。

支持查看订单状态（待支付、已支付、已发货、已收货等）。

**收货与关注**

用户可选择关注商品及卖家，便于获取动态信息。

支持确认收货并留下收货评价。

**个人信息管理**

支持修改个人资料，如昵称、联系方式、头像等。

支持查看自己的交易记录及评价。

**评价与反馈**

用户可以对商品和卖家进行评分和评价。

支持查看其他用户的评价，有助于决策购买。

**卖家出售模块**

**注册与登录**

卖家与买家共用一个账号系统，但注册时需填写详细身份验证信息，如学号、联系方式等。

**商品发布**

发布信息：卖家可以填写商品名称、描述、价格、成色、图片等信息。

商品分类：商品发布时需选择分类，便于买家搜索。

商品状态管理：卖家可设置商品状态（在售、已售出、下架等）。

**商品编辑与下架**

商品编辑：卖家可以对发布的商品进行编辑，修改信息。

商品下架：卖家可以将商品设置为下架状态，无法被买家查看。

批量操作：支持批量编辑、批量下架等功能。

**订单管理**

查看订单：卖家可以查看买家的订单信息，包括交易状态。

订单状态更新：卖家可将订单标记为已发货、已完成等状态。

消息通知：订单状态变更时，系统会发送通知消息给买家。

**问题反馈**

售后问题：卖家可以处理买家在售后过程中提出的问题，如退货、退款等。

客服系统：系统提供客服入口，卖家可在线回复买家的问题。

**店铺管理与推广**

店铺设置：卖家可以设置店铺名称、简介、联系信息等。

店铺推广：平台提供推广功能，帮助卖家提升店铺曝光度。

店铺数据统计：提供浏览量、成交量、商品收藏量等统计数据。

**财务管理**

收入结算：卖家可以查看交易收入，按周期进行结算。

账单明细：系统生成账单，包含每笔交易的金额、手续费等信息。

提现功能：卖家可以申请提现到绑定的银行卡或电子钱包。

**管理员管理模块**

**用户管理**

用户审核：平台管理员可对新注册的卖家或有异常的用户进行审核。

用户信息管理：支持用户信息的查看、编辑、禁用等操作。

封禁和解封：平台管理员可封禁违规用户账户，并根据情况解封。

**商品管理**

商品审核：管理员可对平台上的商品进行审核，确保内容符合规范。

商品分类管理：管理员可以编辑商品分类，确保分类信息准确。

违规商品处理：管理员可下架或删除违规商品，并通知相关用户。

**交易管理**

订单监控：管理员可查看平台所有订单的状态，发现异常交易时可进行处理。

投诉处理：管理员可处理买卖双方的投诉，调解纠纷。

交易数据统计：提供订单数量、交易金额等数据，供平台运营分析。

**平台设置及维护**

广告位管理：平台提供广告位，管理员可对广告内容进行更新。

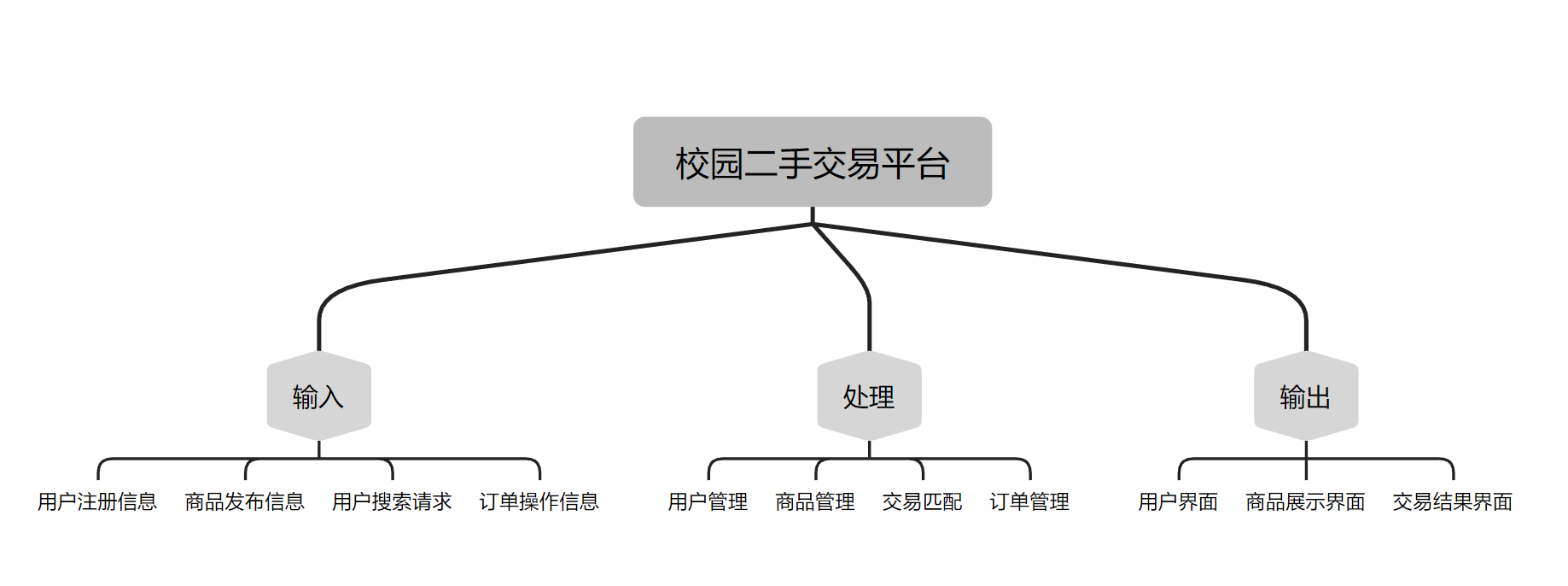
平台公告管理：发布重要公告、系统维护通知等信息。

系统升级与扩展：根据用户反馈和需求，定期进行系统优化和功能扩展。

# 五．软件架构与设计

## 5.1 软件层次结构设计

顶层HIPO图



**输入**：

用户注册信息（姓名、学号、联系方式、邮箱等）

商品发布信息（商品名称、描述、价格、图片、类别等）

用户搜索请求（关键词、类别、价格范围等）

订单操作信息（下单、支付、确认收货等）

**处理**：

用户管理（注册、登录、验证、权限设置）

商品管理（发布、审核、上架、下架、修改）

交易匹配（根据搜索请求匹配商品）

订单处理（生成订单、处理支付、物流跟踪、订单完成）

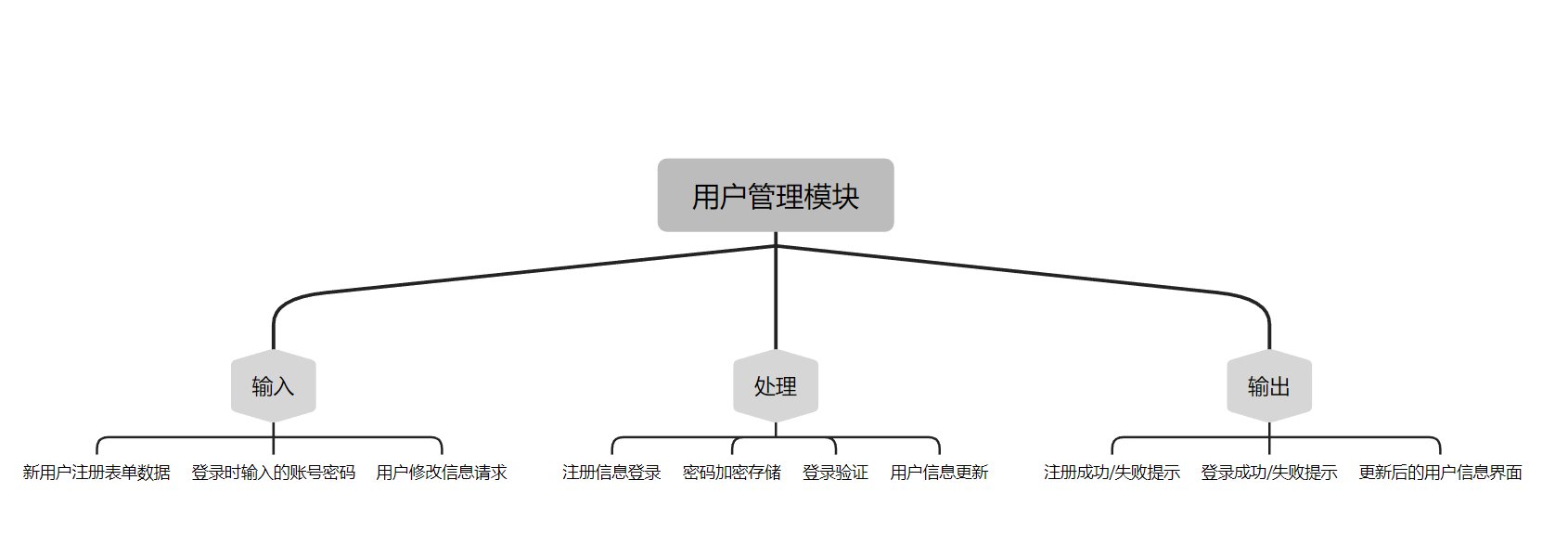
**输出**：

用户界面（注册 / 登录成功提示、个人信息显示、操作反馈等）

商品展示界面（商品列表、商品详情页）

交易结果界面（订单状态更新、支付成功提示等）

二、一级分解 - 用户管理模块



**输入**

新用户注册表单数据

登录时输入的账号密码

用户修改信息请求

**处理**：

注册信息验证（格式、唯一性检查）

密码加密存储

登录验证（与数据库存储信息比对）

用户信息更新操作

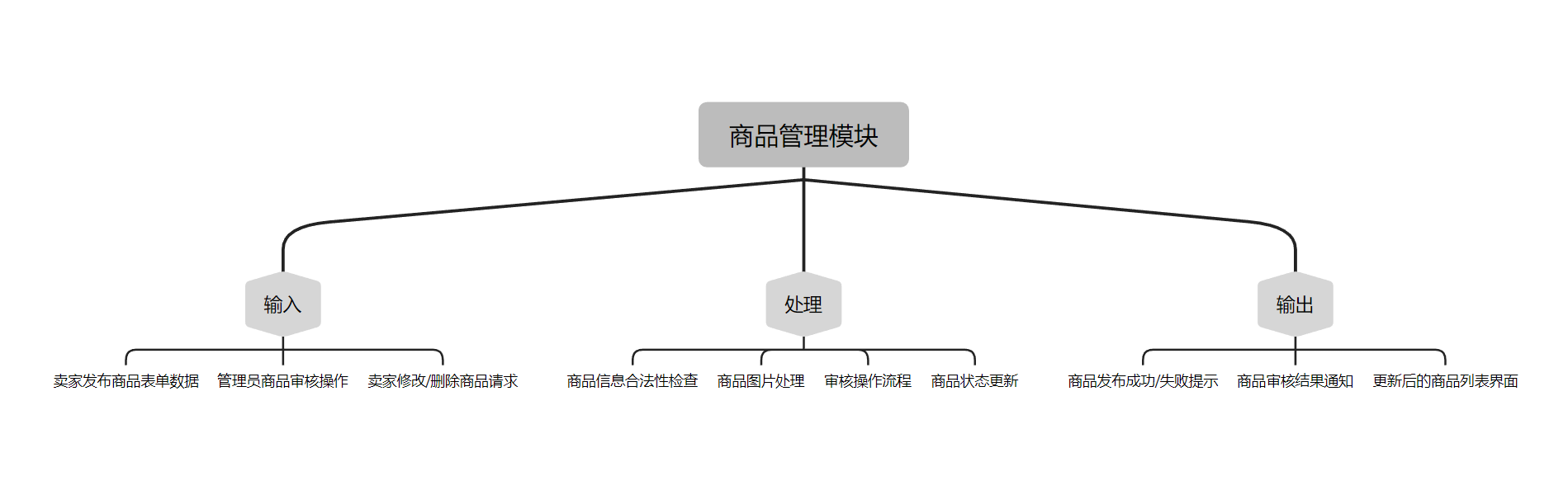
**输出**：

注册成功 / 失败提示

登录成功 / 失败提示

更新后的用户信息界面

三、一级分解 – 商品管理模块



**输入**：

卖家发布商品表单数据

管理员商品审核操作

卖家修改 / 删除商品请求

**处理**：

商品信息合法性检查（必填项、价格格式等）

商品图片处理（存储、压缩、显示优化）

审核流程（根据规则判断商品是否合规）

商品状态更新（上架、下架等）

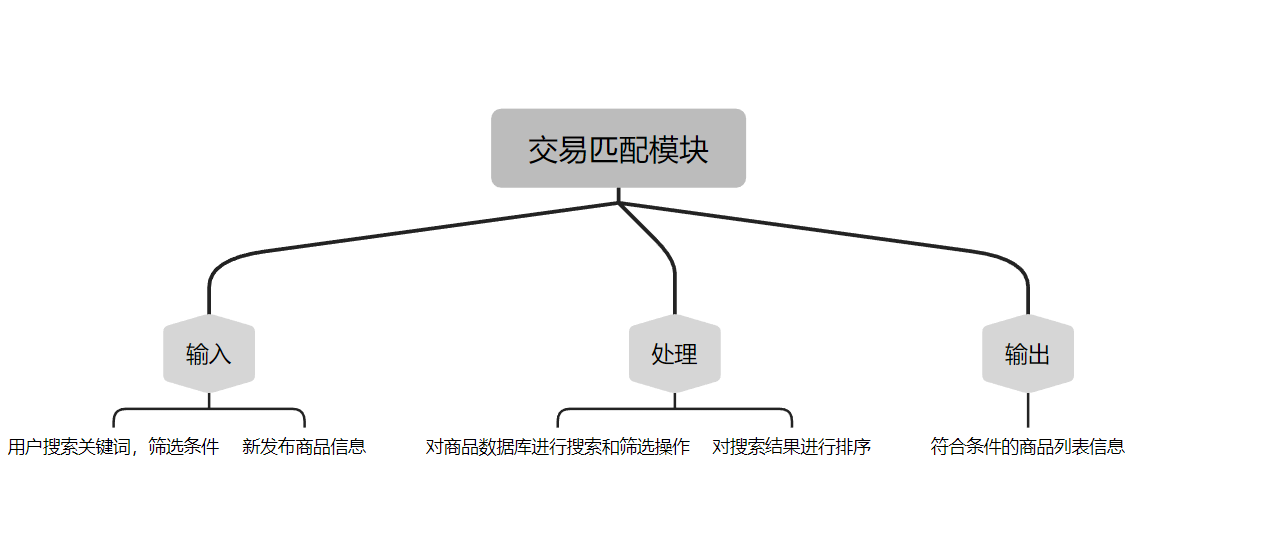
**输出**：

商品发布成功 / 失败提示

商品审核结果通知（卖家）

更新后的商品列表界面

四．一级分解 – 交易匹配模块



输入：

用户搜索关键词、筛选条件

新发布商品信息

处理：

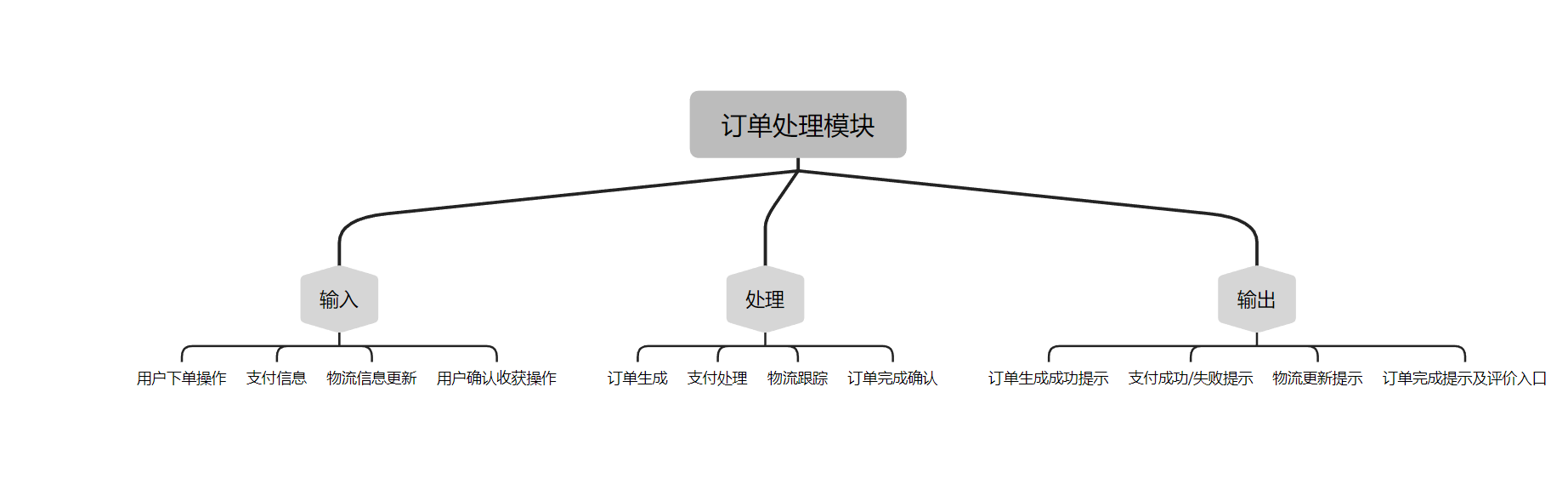
对商品数据库进行搜索和筛选操作

对搜索结果进行排序（根据相关性、价格等因素）

输出：

符合条件的商品列表界面

五．一级分解 – 订单处理模块



输入：

用户下单操作（选择商品、数量、收货地址等）

支付信息（支付平台返回结果）

物流信息更新（物流公司数据接口）

用户确认收货操作

处理：

订单生成（计算总价、生成订单编号）

支付处理（与支付网关交互、验证支付结果）

物流跟踪（更新订单物流状态）

订单完成确认（更新商品库存、评价开启等）

输出：

订单生成成功提示

支付成功 / 失败提示

物流状态更新提示

订单完成提示及评价入口

# 六．设计数据库

## 6.1 数据库需求分析

一、业务功能需求分析

**（一）用户管理**

1. **用户注册与登录**  
   用户需要通过注册成为平台成员，注册信息包括学号、姓名、联系方式、密码等。登录时通过学号和密码验证身份。用户分为普通用户和商家用户。
2. **用户信息修改**  
   用户可修改除学号外的个人信息，如联系方式、密码等。

**（二）管理员功能**

1. **用户管理**  
   管理员可查看、审核用户信息，对违规用户进行处理，如冻结账号。
2. **商家管理**  
   审核商家入驻申请，监督商家经营行为，处理商家相关问题。
3. **平台维护**  
   对商品分类、交易规则等平台设置进行维护和更新。

**（三）商家功能**

1. **商品发布与管理**  
   商家可发布商品信息，包括商品名称、描述、价格、库存、成色、图片等。同时可对商品信息进行修改、下架等操作。
2. **订单处理**  
   商家处理来自用户的订单，包括确认订单、发货、处理售后等。

**（四）商品浏览与交易**

1. **商品浏览与搜索**  
   用户可浏览所有商品，按分类、关键词、价格范围等条件搜索商品。查看商品详细信息和图片。
2. **下单购买**  
   用户选择商品后可下单，填写收货地址、联系人、联系电话等信息。
3. **订单跟踪与评价**  
   用户可跟踪订单状态，在交易完成后对商品和商家进行评价。

**二、数据实体与关系分析**

**（一）用户信息表**

1. **属性**：学号（主键）、姓名、联系方式、密码、用户类型（普通用户 / 商家）、注册时间、最后登录时间等。
2. **关系**：一个用户可以有多个订单（与订单信息表一对多关系）。商家用户与商家信息表存在一对一关系。

**（二）管理员信息表**

1. **属性**：管理员编号（主键）、管理员姓名、登录账号、登录密码、权限级别等。
2. **关系**：管理员可管理多个用户和商家（一对多关系）。

**（三）商家信息表**

1. **属性**：商家编号（主键）、商家名称、店铺名称、联系电话、经营范围、商家简介、入驻时间、商家账号（外键关联用户信息表中商家用户的学号）等。
2. **关系**：一个商家可发布多个商品（与商品信息表一对多关系），处理多个订单（与订单信息表一对多关系）。

**（四）商品信息表**

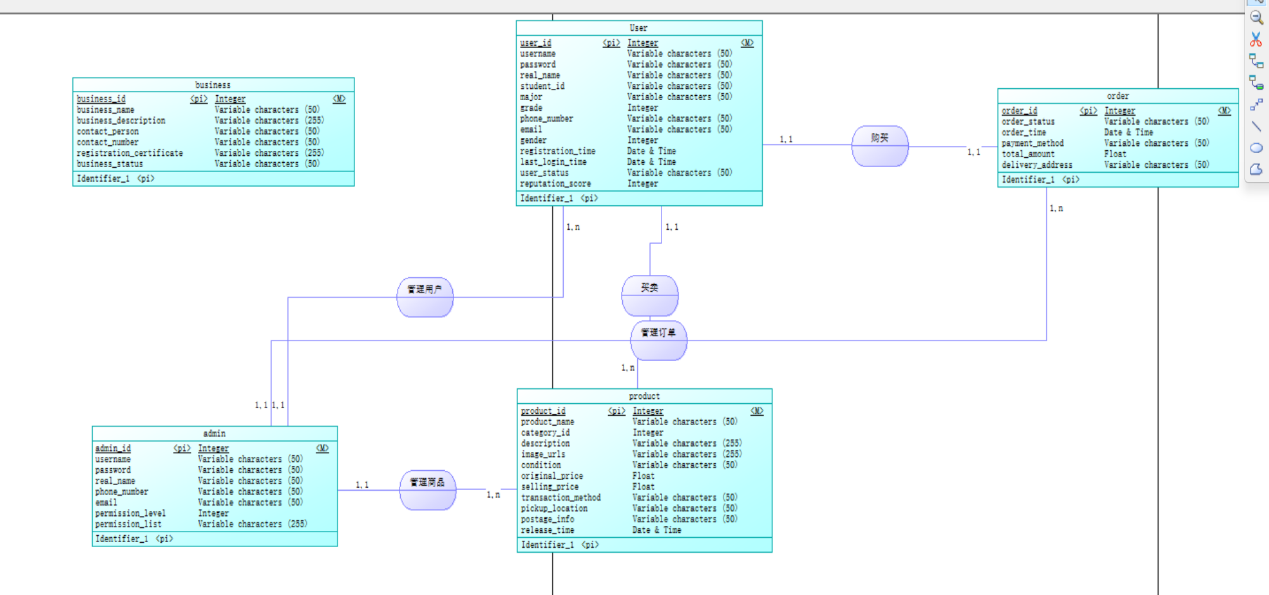
1. **属性**：商品编号（主键）、商品名称、商品描述、价格、库存、成色、图片路径、商品分类、发布时间、商家编号（外键关联商家信息表）等。
2. **关系**：一件商品可在多个订单中出现（与订单信息表多对多关系，通过中间表体现）。

**（五）订单信息表**

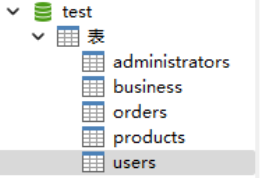
1. **属性**：订单编号（主键）、下单时间、用户学号（外键关联用户信息表）、商品编号（外键关联商品信息表）、订单状态（待付款、已付款、待发货、已发货、已完成、已取消等）、收货地址、联系人、联系电话、支付方式、金额等。
2. **关系**：一个订单对应一个用户和一个或多个商品。

## 6.2 数据库概念模型设计

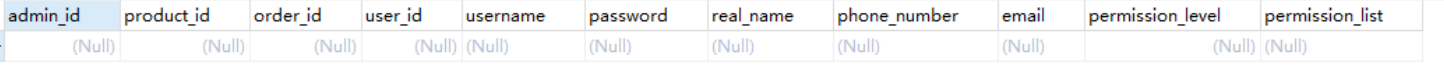
**ER图设计**



**导入到数据库生成数据表**

**ii**

**adminisstrators**

****

**business**

****

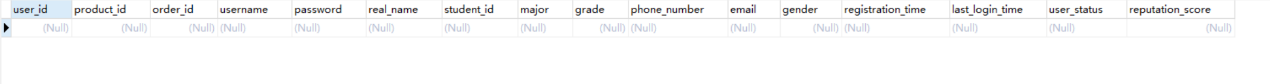
**orders**

****

**products**

****

**users**

****

# 八．书写文档计划

## 8.1 系统说明

## 8.2 用户手册

## 8.3 测试计划

## 8.3 详细实现计划

## 8.4 数据库设计结果

# 九．总结

通过该总体设计报告，我们成功完成了校园二手平台从需求分析、方案选择、功能设计、软件架构搭建到数据库设计的全过程。各个环节紧密相连，为平台的开发和实现奠定了坚实的基础。在项目执行过程中，团队成员充分发挥各自的专业技能，共同克服了遇到的各种问题。未来，该平台将为校园二手交易提供便利，推动校园资源的可持续利用。

# 十．参考资料

项目总体设计报告模板：

https://zhuanlan.zhihu.com/p/623270926?utm\_campaign=shareopn&utm\_medium=social&utm\_psn=1828814499123183616&utm\_source=wechat\_session

# 十一．小组组员评价

|  |  |  |  |
| --- | --- | --- | --- |
| 小组成员 | 胡奕晨 | 黄哲瀚 | 仇鑫宇 |
| 负责内容 | 项目总体设计报告制作以及测试计划制作 | 系统说明以及用户手册文档制作 | 详细实现计划以及数据库设计结果制作 |
| 评分 | 8 | 7.5 | 7.5 |