

同济大学计算机系

《软件开发方法》课程期末项目报告



项目名称 基于 Web 的电子商城系统分析与设计

学 号 2251745

姓 名 张宇

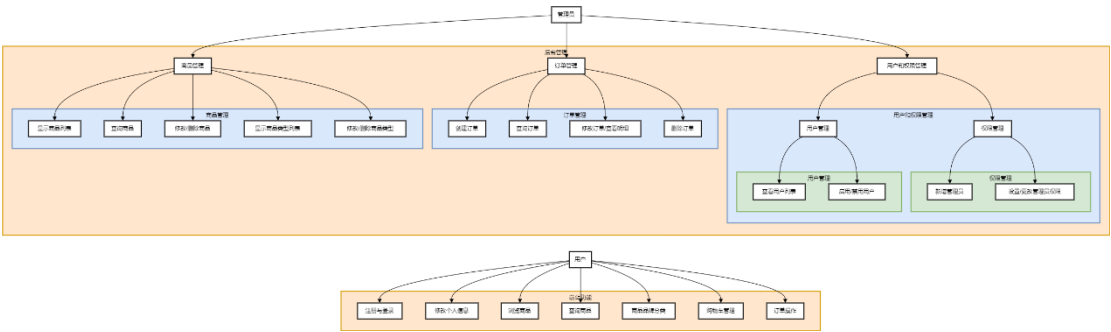
专 业 计算机科学与技术

任课老师 王小平老师

完成日期 2024 年 5 月 30 日

一、UML 需求分析

1、用例图



2、用例的必要说明

- (1) 注册与登录： 用户可以在商城注册账号，并使用用户名和密码登录。
- (2) 修改个人信息： 用户可以修改个人信息，包括地址、联系方式等。
- (3) 浏览商品： 用户可以浏览商城中的商品列表，查看商品的基本信息。
- (4) 查询商品： 用户可以通过关键词搜索商品，快速找到想要的商品。
- (5) 商品品牌分类： 用户可以按品牌查看商品分类。
- (6) 购物车管理： 用户可以将商品加入购物车，暂时存放选购的商品，并在购物车中管理这些商品。
- (7) 生成订单： 用户可以将购物车中的商品生成订单，进行购买。
- (8) 查看订单： 用户可以查看自己的订单信息，包括订单状态、订单详情等。
- (9) 显示商品列表： 管理员可以查看商城中的所有商品列表。
- (10) 查询商品： 管理员可以根据条件查询商品。
- (11) 修改/删除商品： 管理员可以修改或删除商品信息。
- (12) 显示商品类型列表： 管理员可以查看商品类型列表。
- (13) 修改/删除商品类型： 管理员可以修改或删除商品类型。
- (14) 创建订单： 管理员可以创建新的订单
- (15) 查询订单： 管理员可以根据条件查询订单。
- (16) 修改订单/查看明细： 管理员可以修改订单信息或查看订单详细信息。
- (17) 删除订单： 管理员可以删除订单。
- (18) 查看用户列表： 管理员可以查看商城中的用户列表。
- (19) 启用/禁用用户： 管理员可以启用或禁用用户账号。
- (20) 新增管理员： 管理员可以新增管理员账号。

(21) 设置/更改管理员权限：管理员可以设置或更改其他管理员的权限。

二、数据库和数据表的设计

1、数据模型

- (1) 用户信息表(user_info)
- (2) 管理员信息表(admin_info)
- (3) 类型表(type)
- (4) 商品信息表(product_info)
- (5) 订单信息表(order_info)
- (6) 订单明细表(order_detail)
- (7) 系统功能表(functions)
- (8) 权限表(powers)

2、数据库表设计

- (1) 用户信息表(user_info)

```
mysql> CREATE TABLE user_info (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     userName VARCHAR(16) NOT NULL,  
->     password VARCHAR(16) NOT NULL,  
->     realName VARCHAR(8),  
->     sex VARCHAR(4),  
->     address VARCHAR(255),  
->     question VARCHAR(50),  
->     answer VARCHAR(50),  
->     email VARCHAR(50),  
->     favorite VARCHAR(50),  
->     score INT,  
->     regDate DATE  
-> );  
Query OK, 0 rows affected (0.04 sec)
```

- (2) 管理员信息表(admin_info)

```
mysql> CREATE TABLE admin_info (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     name VARCHAR(16) NOT NULL,  
->     pwd VARCHAR(50) NOT NULL,  
->     role VARCHAR(8)  
-> );  
Query OK, 0 rows affected (0.02 sec)
```

(3) 类型表(type)

```
mysql> CREATE TABLE type (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     name VARCHAR(20)  
-> );  
Query OK, 0 rows affected (0.03 sec)
```

(4) 商品信息表(product_info)

```
mysql> CREATE TABLE product_info (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     code VARCHAR(16) NOT NULL,  
->     name VARCHAR(255),  
->     tid INT,  
->     brand VARCHAR(20),  
->     pic VARCHAR(255),  
->     num INT,  
->     price DECIMAL(10, 0),  
->     intro LONGTEXT,  
->     status INT,  
->     bigpic VARCHAR(255),  
->     FOREIGN KEY (tid) REFERENCES type(id)  
-> );  
Query OK, 0 rows affected (0.06 sec)
```

(4) 商品信息表(product_info)

```
mysql> CREATE TABLE order_info (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     uid INT,  
->     status VARCHAR(16),  
->     ordertime DATETIME,  
->     orderprice DECIMAL(8, 2),  
->     FOREIGN KEY (uid) REFERENCES user_info(id)  
-> );  
Query OK, 0 rows affected (0.06 sec)
```

(6) 订单明细表(order_detail)

```
mysql> CREATE TABLE order_detail (  
->     id INT AUTO_INCREMENT PRIMARY KEY,  
->     oid INT,  
->     pid INT,  
->     num INT,  
->     FOREIGN KEY (oid) REFERENCES order_info(id),  
->     FOREIGN KEY (pid) REFERENCES product_info(id)  
-> );  
Query OK, 0 rows affected (0.05 sec)
```

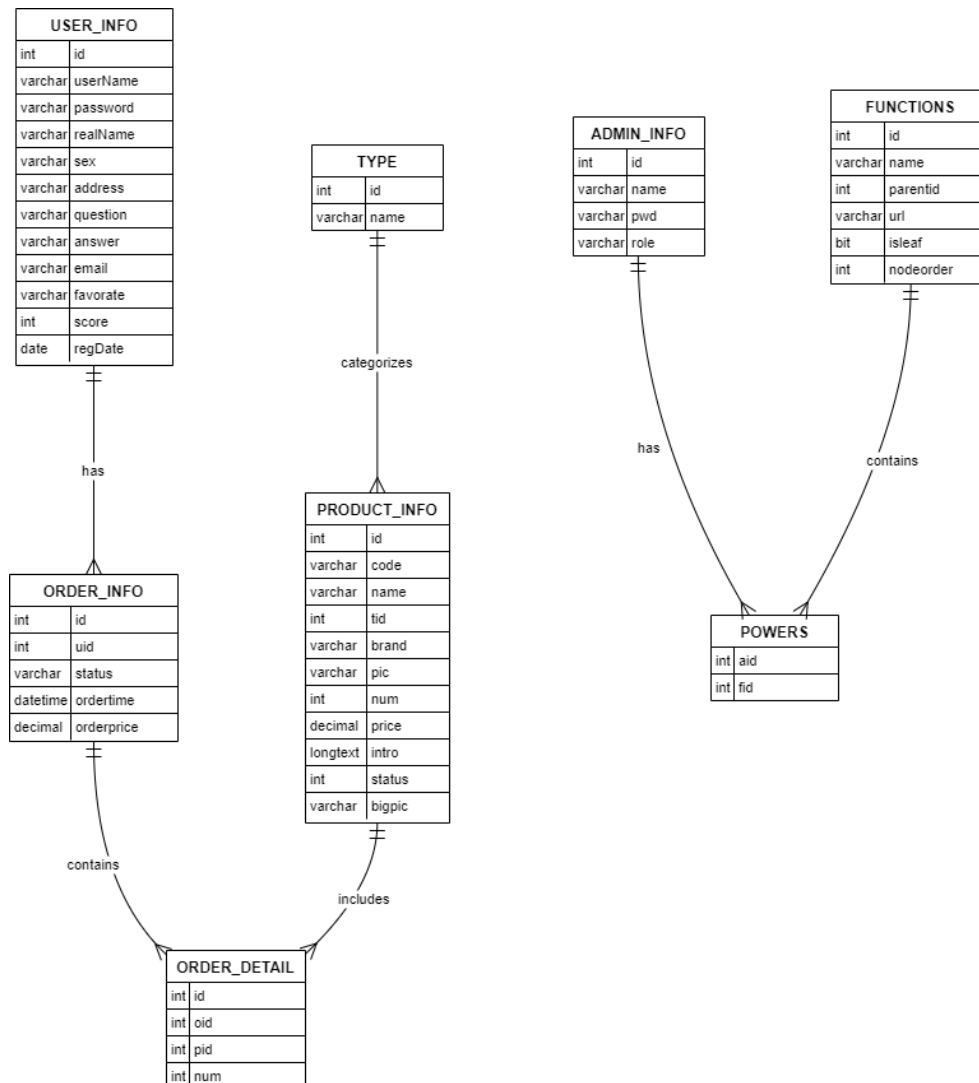
(7) 系统功能表(functions)

```
mysql> CREATE TABLE functions (
  ->   id INT AUTO_INCREMENT PRIMARY KEY,
  ->   name VARCHAR(20),
  ->   parentid INT,
  ->   url VARCHAR(50),
  ->   isleaf BIT(1),
  ->   nodeorder INT
  -> );
Query OK, 0 rows affected (0.02 sec)
```

(8) 权限表(powers)

```
mysql> CREATE TABLE powers (
  ->   aid INT,
  ->   fid INT,
  ->   PRIMARY KEY (aid, fid),
  ->   FOREIGN KEY (aid) REFERENCES admin_info(id),
  ->   FOREIGN KEY (fid) REFERENCES functions(id)
  -> );
Query OK, 0 rows affected (0.08 sec)
```

3、E-R 图



三、类、类图、包图设计

1、类定义

(1) User (用户类)

- 属性:
 - id: int
 - userName: String
 - password: String
 - realName: String
 - sex: String
 - address: String
 - question: String
 - answer: String
 - email: String
 - favorate: String
 - score: int
 - regDate: Date
- 方法:
 - register(): void
 - login(): void
 - viewProfile(): void
 - editProfile(): void
 - addToCart(): void
 - placeOrder(): void
 - viewOrder(): void

(2) Admin (管理员类)

- 属性:
 - id: int
 - name: String
 - pwd: String
 - role: String
- 方法:
 - manageProducts(): void
 - manageOrders(): void
 - manageUsers(): void
 - setPermissions(): void

(3) Product (商品类)

- 属性:
 - id: int
 - code: String
 - name: String
 - tid: int
 - brand: String
 - pic: String

- num: int
- price: BigDecimal
- intro: String
- status: int
- bigpic: String
- 方法:
 - addProduct(): void
 - editProduct(): void
 - deleteProduct(): void
 - viewProduct(): void

(4) Order (订单类)

- 属性:
 - id: int
 - uid: int
 - status: String
 - ordertime: Date
 - orderprice: BigDecimal
- 方法:
 - createOrder(): void
 - editOrder(): void
 - deleteOrder(): void
 - viewOrder(): void

(5) OrderDetail (订单明细类)

- 属性:
 - id: int
 - oid: int
 - pid: int
 - num: int
- 方法:
 - addOrderDetail(): void
 - editOrderDetail(): void
 - deleteOrderDetail(): void
 - viewOrderDetail(): void

(6) Type (类型类)

- 属性:
 - id: int
 - name: String
- 方法:
 - addType(): void
 - editType(): void
 - deleteType(): void
 - viewType(): void

(7) Function (功能类)

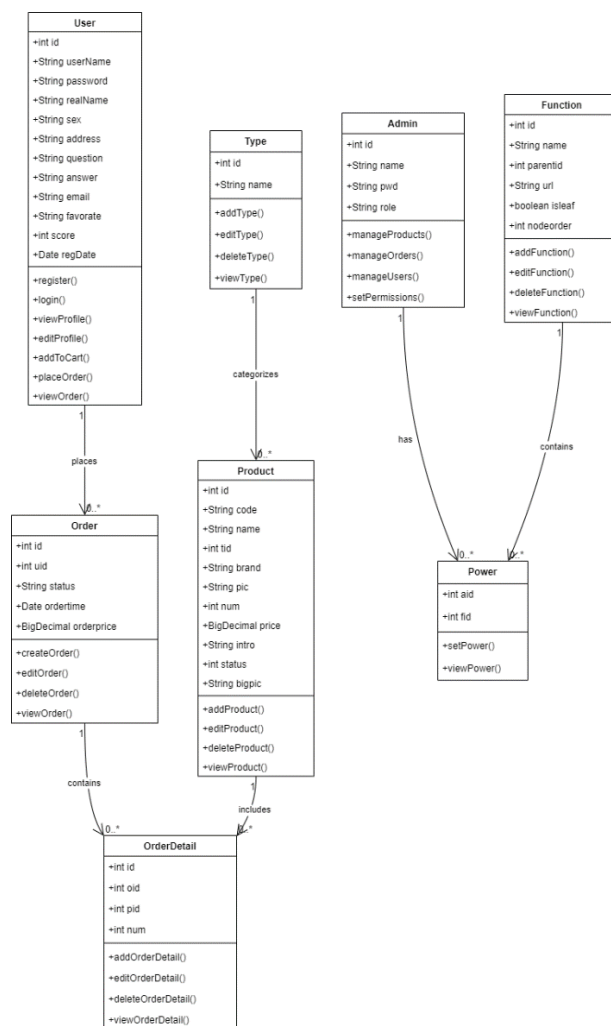
- 属性:

- id: int
- name: String
- parentid: int
- url: String
- isleaf: boolean
- nodeorder: int
- 方法:
 - addFunction(): void
 - editFunction(): void
 - deleteFunction(): void
 - viewFunction(): void

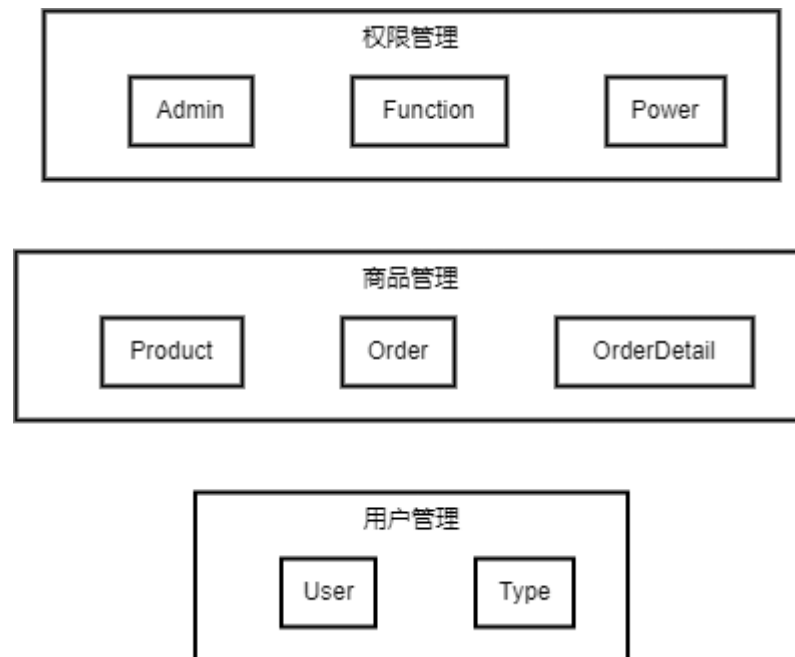
(8) Power (权限类)

- 属性:
 - aid: int
 - fid: int
- 方法:
 - setPower(): void
 - viewPower(): void

3、类图



3、包图



4、类和类之间的关系说明

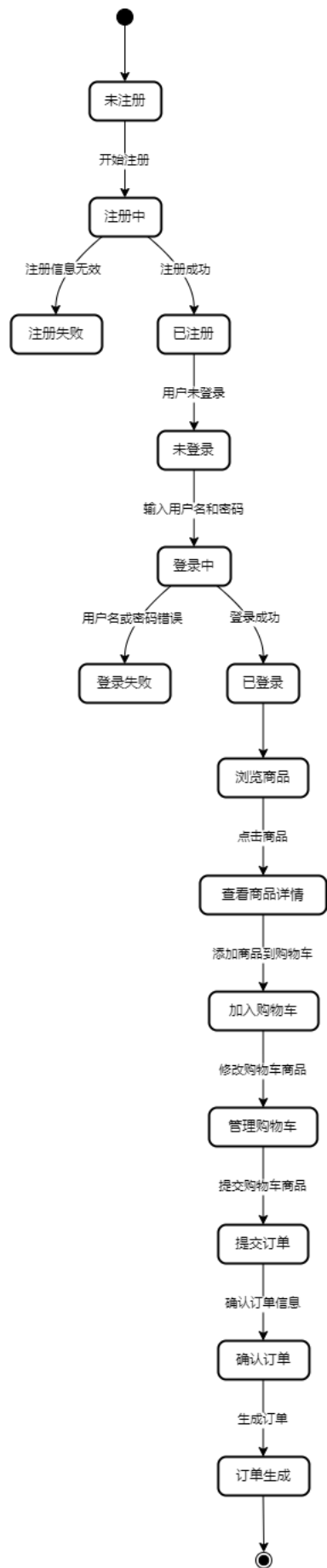
- 用户类 (User):
 - 管理用户信息、用户注册、登录、查看和编辑个人信息、管理购物车、下订单等功能。
- 管理员类 (Admin):
 - 管理系统中的商品、订单、用户和权限设置。
- 商品类 (Product):
 - 管理商品信息，包括添加、编辑、删除和查看商品。
- 订单类 (Order):
 - 管理订单信息，包括创建、编辑、删除和查看订单。
- 订单明细类 (OrderDetail):
 - 管理订单的具体商品信息。
- 类型类 (Type):
 - 管理商品的分类。
- 功能类 (Function):
 - 管理系统功能的信息。
- 权限类 (Power):
 - 管理管理员的权限设置。

四、系统动态模型设计

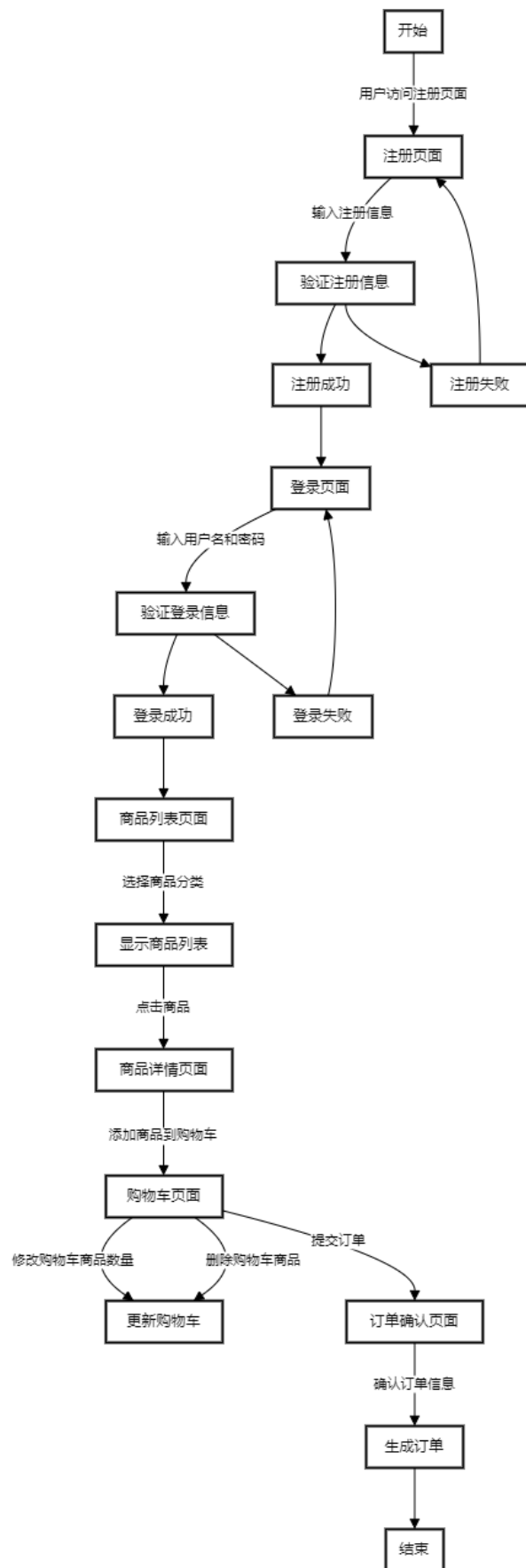
(一) 前台功能

- 1、 用户注册和登录、商品浏览和详细信息查看、购物车管理和订单处理

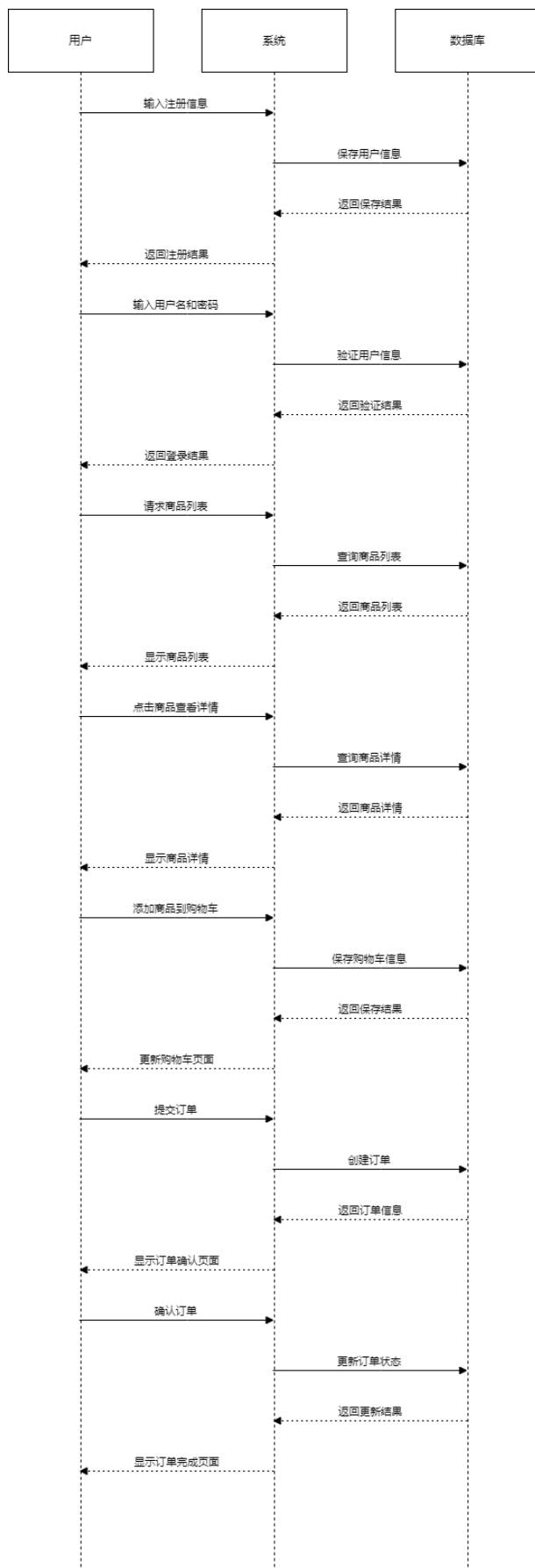
(1) 状态图



(2) 活动图

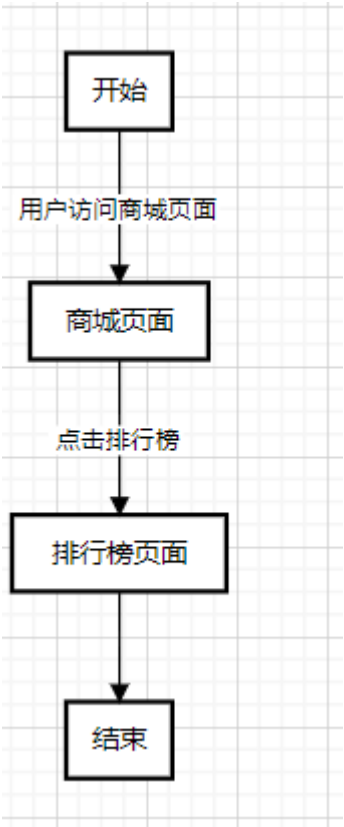


(3) 顺序图

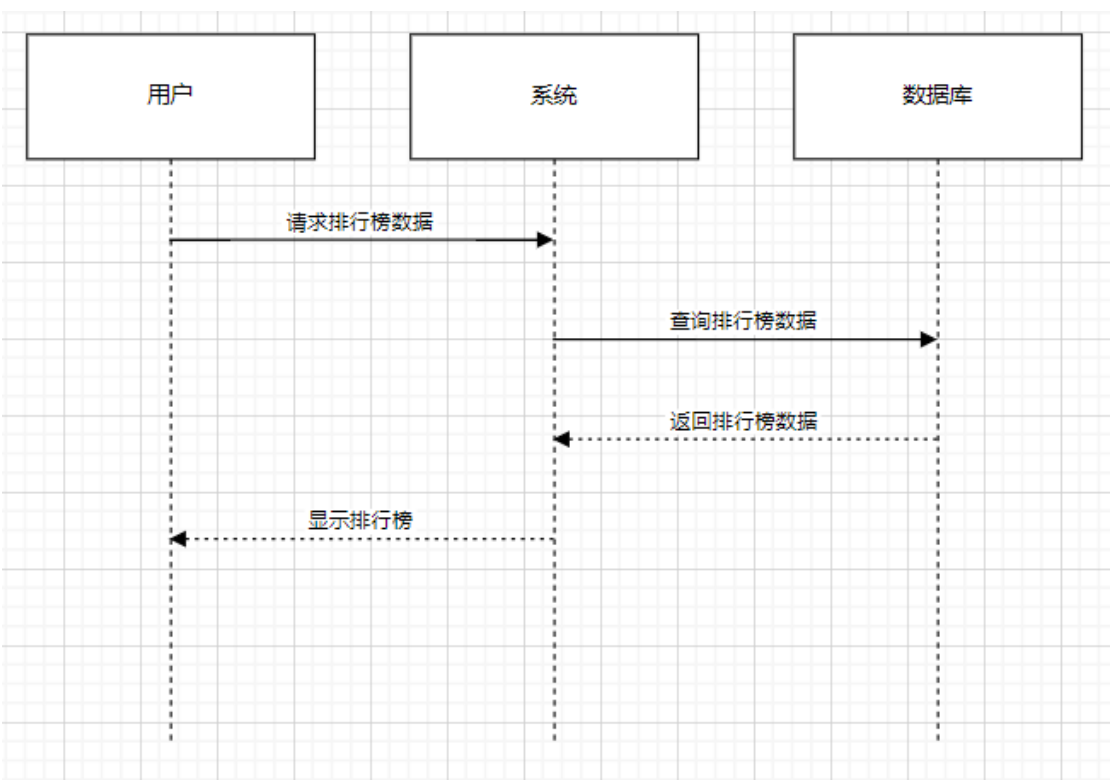


2、浏览排行榜

(1) 活动图

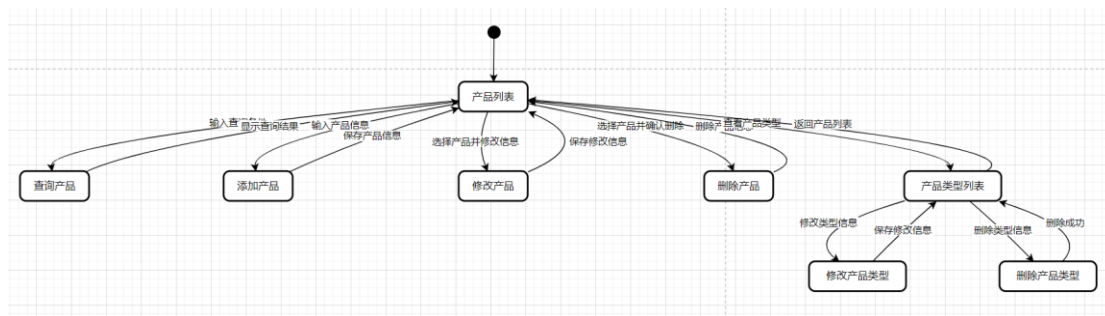


(2) 顺序图

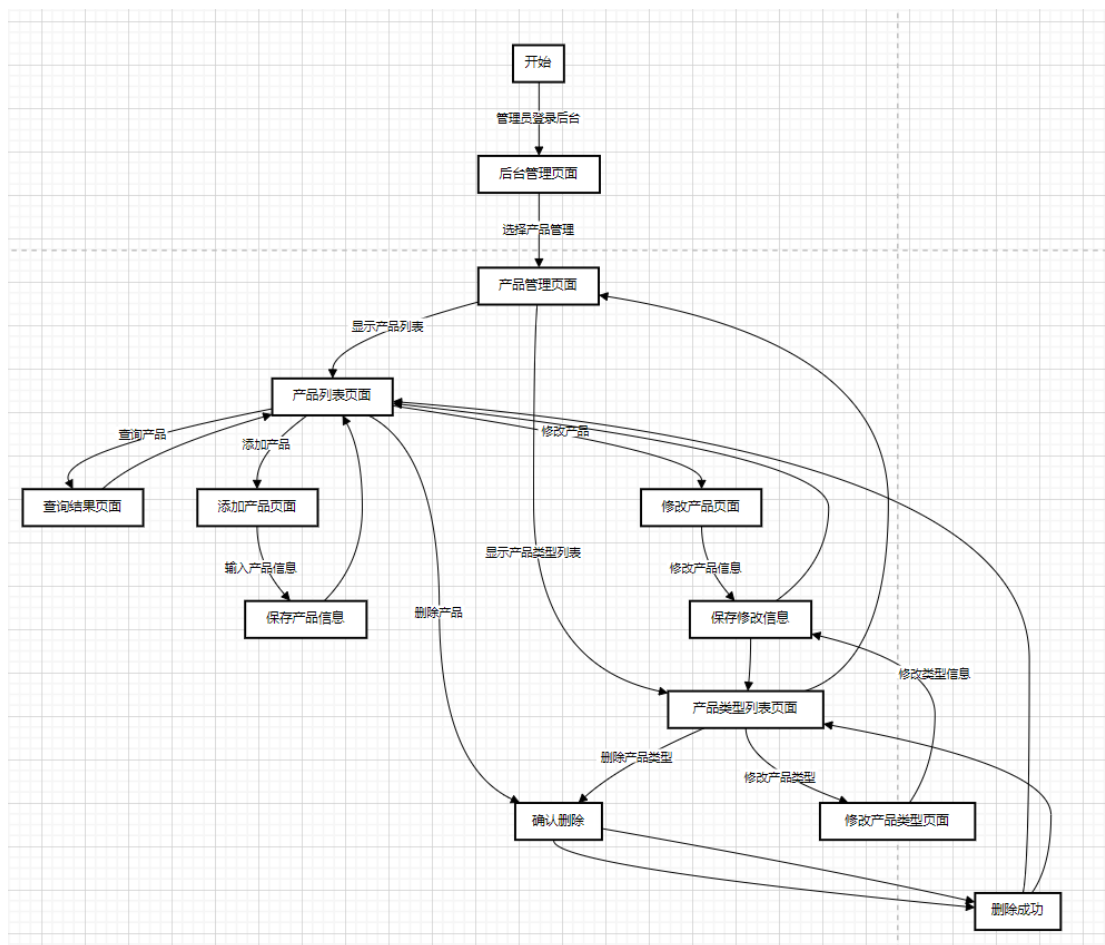


(二) 后台功能一：产品管理

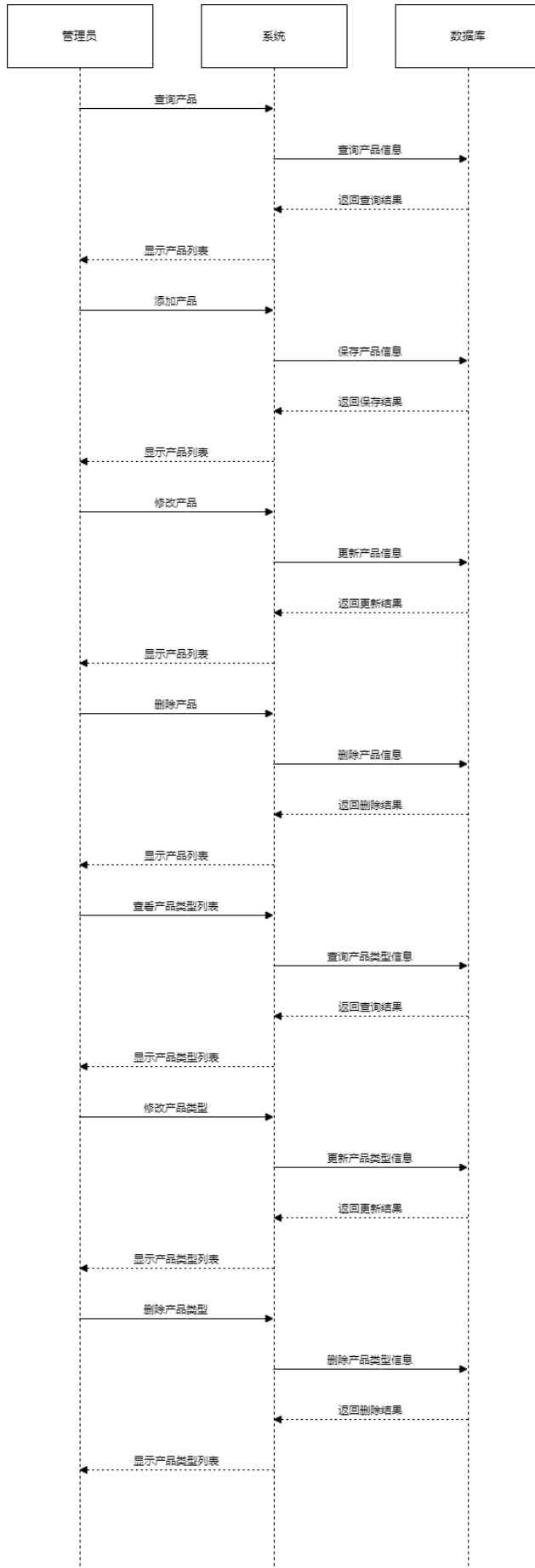
1、状态图



2、活动图

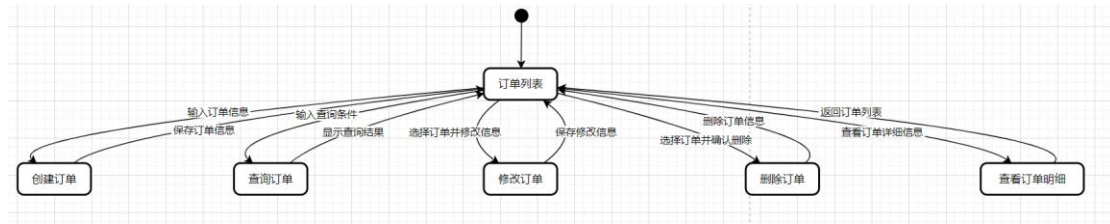


3、顺序图

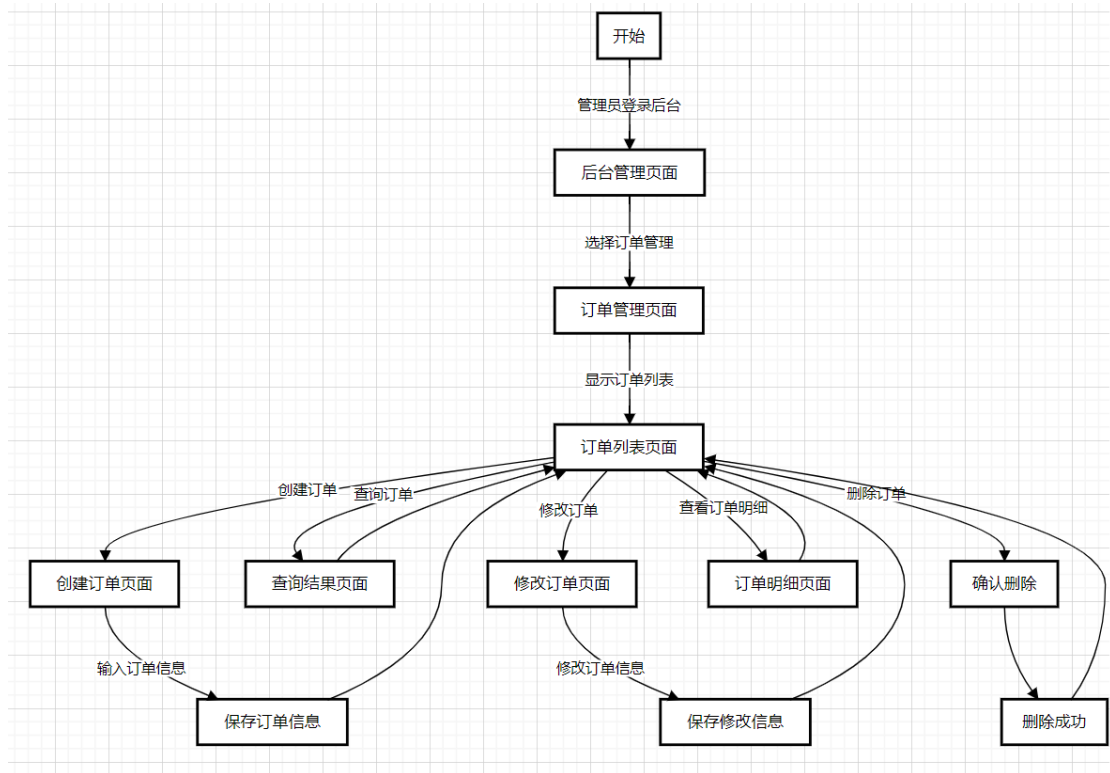


(三) 后台功能二：订单管理

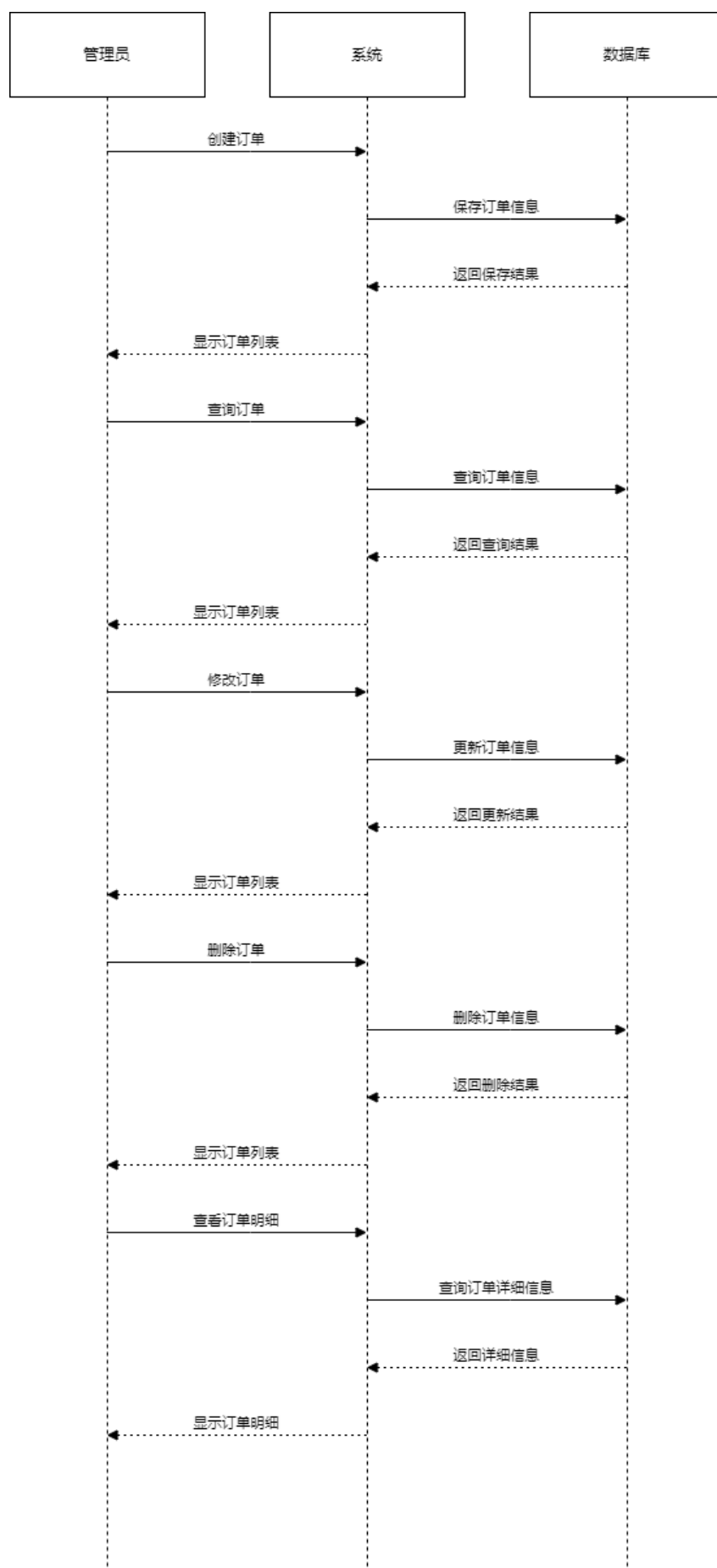
1、 状态图



2、 活动图

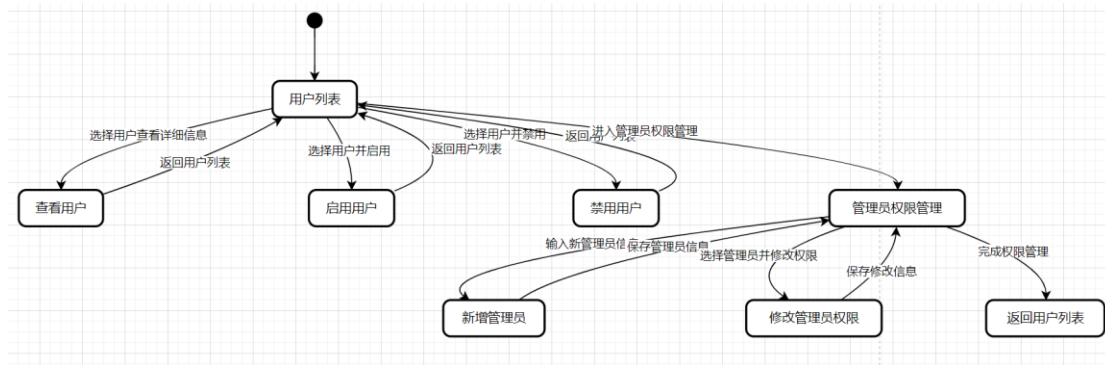


3、 顺序图

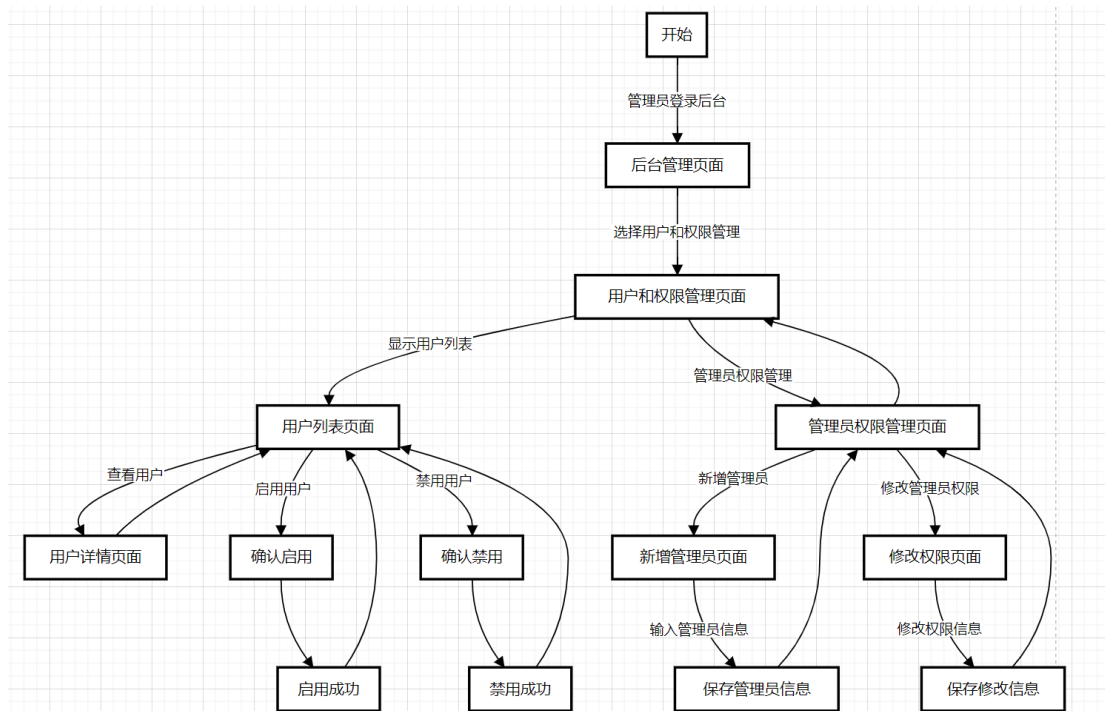


(四) 后台功能三：用户和权限管理

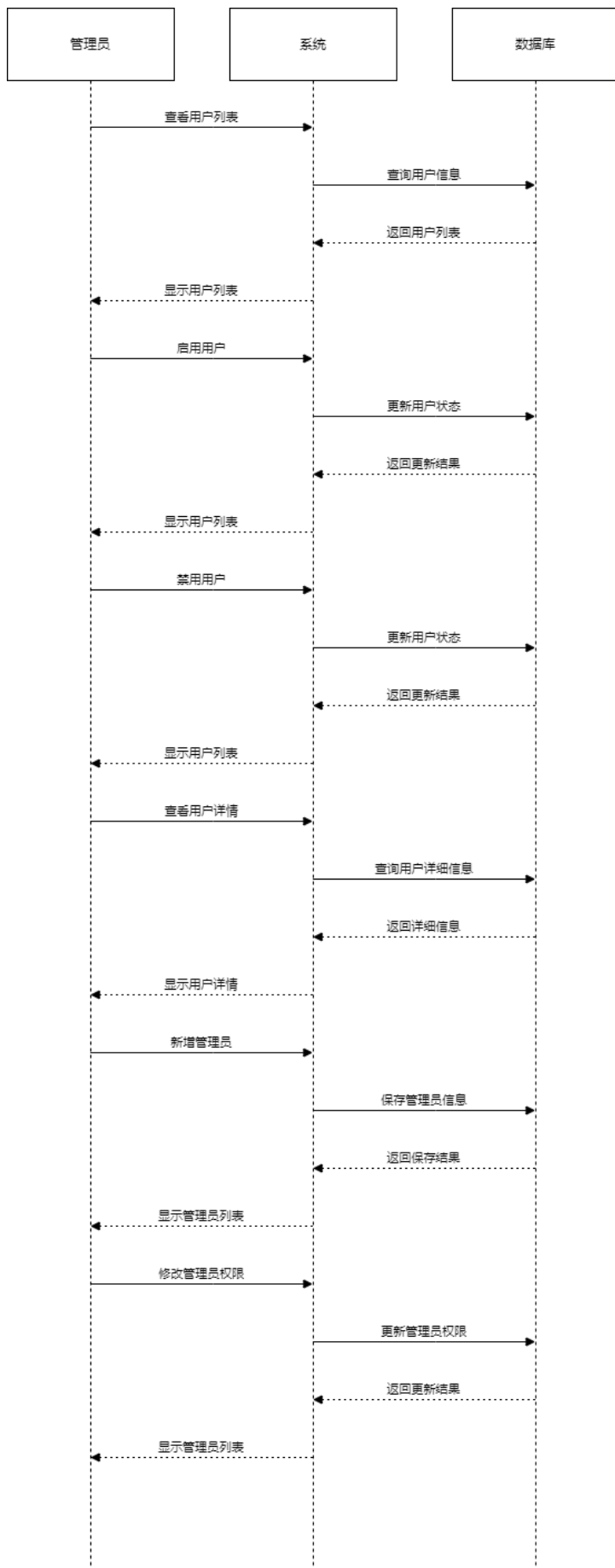
1、 状态图



2、 活动图



3、 顺序图

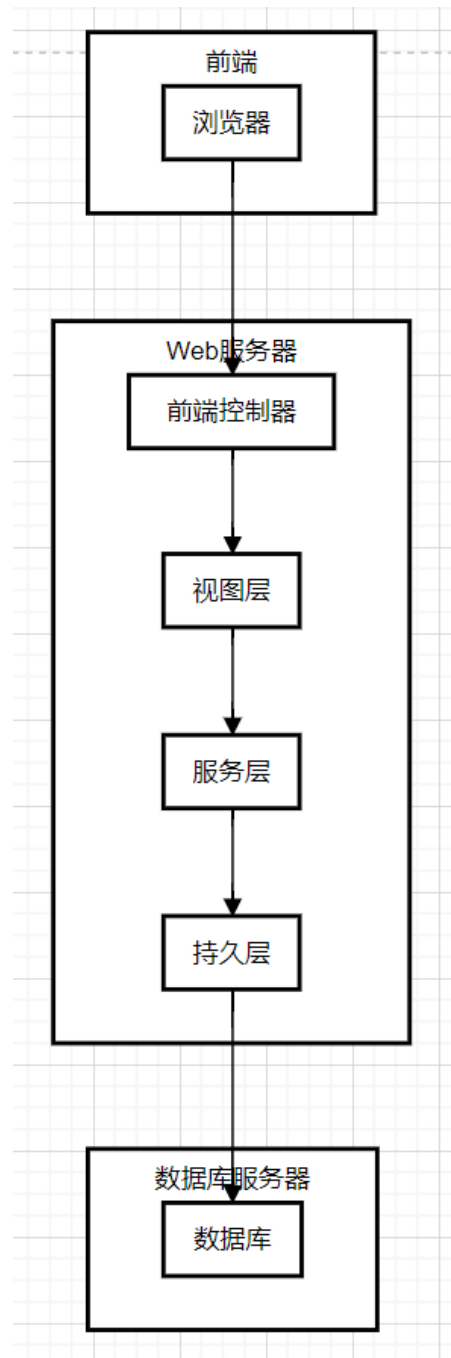


五、系统实现模型设计

1、构件图

构件图展示了系统的物理架构，描述了系统中各个构件之间的关系。下面展示电子商城系统中主要构件之间的关系。

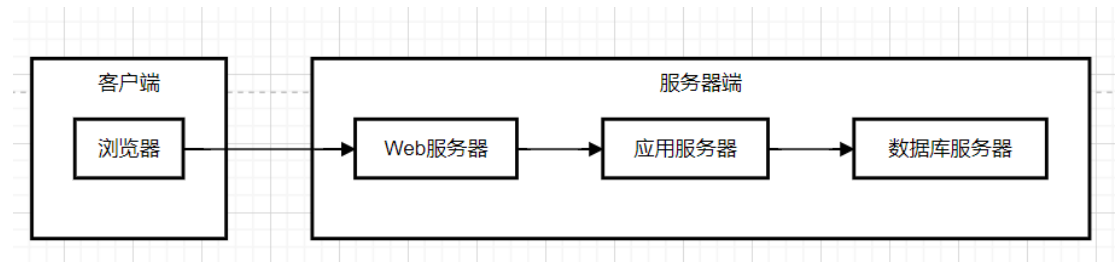
- 浏览器：负责用户交互，发送请求到 Web 服务器并接收响应。
- 前端控制器：负责处理所有 HTTP 请求，调用相应的服务层进行业务处理。
- 视图层：负责渲染页面，将数据展示给用户。
- 服务层：包含业务逻辑，处理具体的业务操作。
- 持久层：负责数据持久化操作，与数据库交互。
- 数据库：存储系统中的所有数据。



2、配置图

配置图展示了系统的硬件和软件的部署情况，描述了各个物理节点之间的关系。下面展示电子商城系统的部署情况。

- 客户端：用户通过浏览器访问系统。
- Web 服务器：接收和处理客户端请求，调用应用服务器进行业务处理。
- 应用服务器：处理业务逻辑，将数据存取操作委托给数据库服务器。
- 数据库服务器：存储和管理系统数据。



3、详细说明

- (1) 浏览器：用户通过浏览器访问电子商城系统。
- (2) Web 服务器：处理用户请求，进行页面渲染和前端控制。包含前端控制器和视图层。
- (3) 应用服务器：处理业务逻辑和数据访问。包含服务层和持久层。
- (4) 数据库服务器：存储系统数据。包括用户信息、商品信息、订单信息等。

六、系统实现模型设计基于本系统需求，开发一个基于Web 的电子商城用户注册和登录管理功能

1、这是我开发网站的登录页面



2、这是我开发网站的注册页面



3、这是我开发网站的登录成功页面

