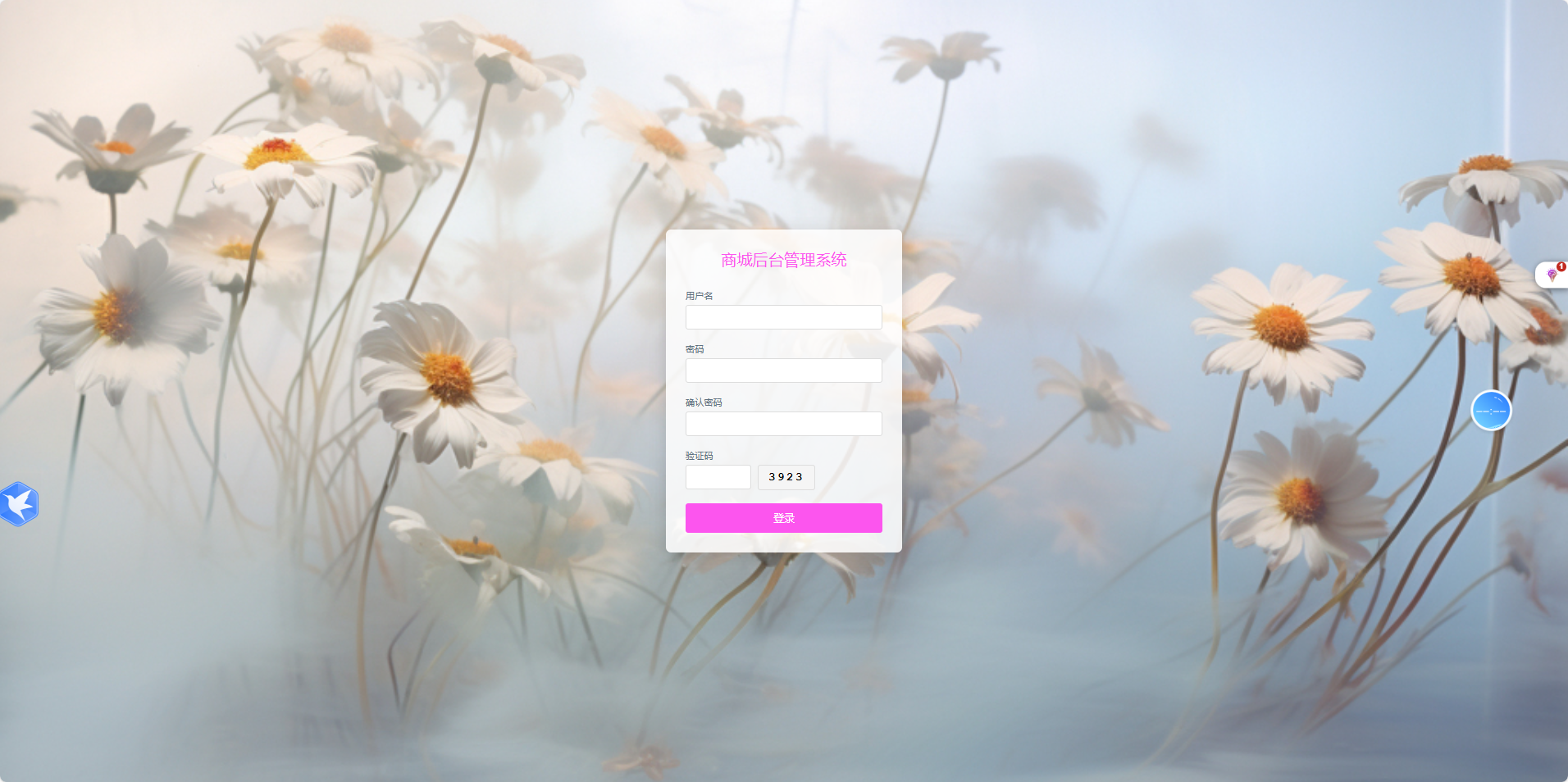
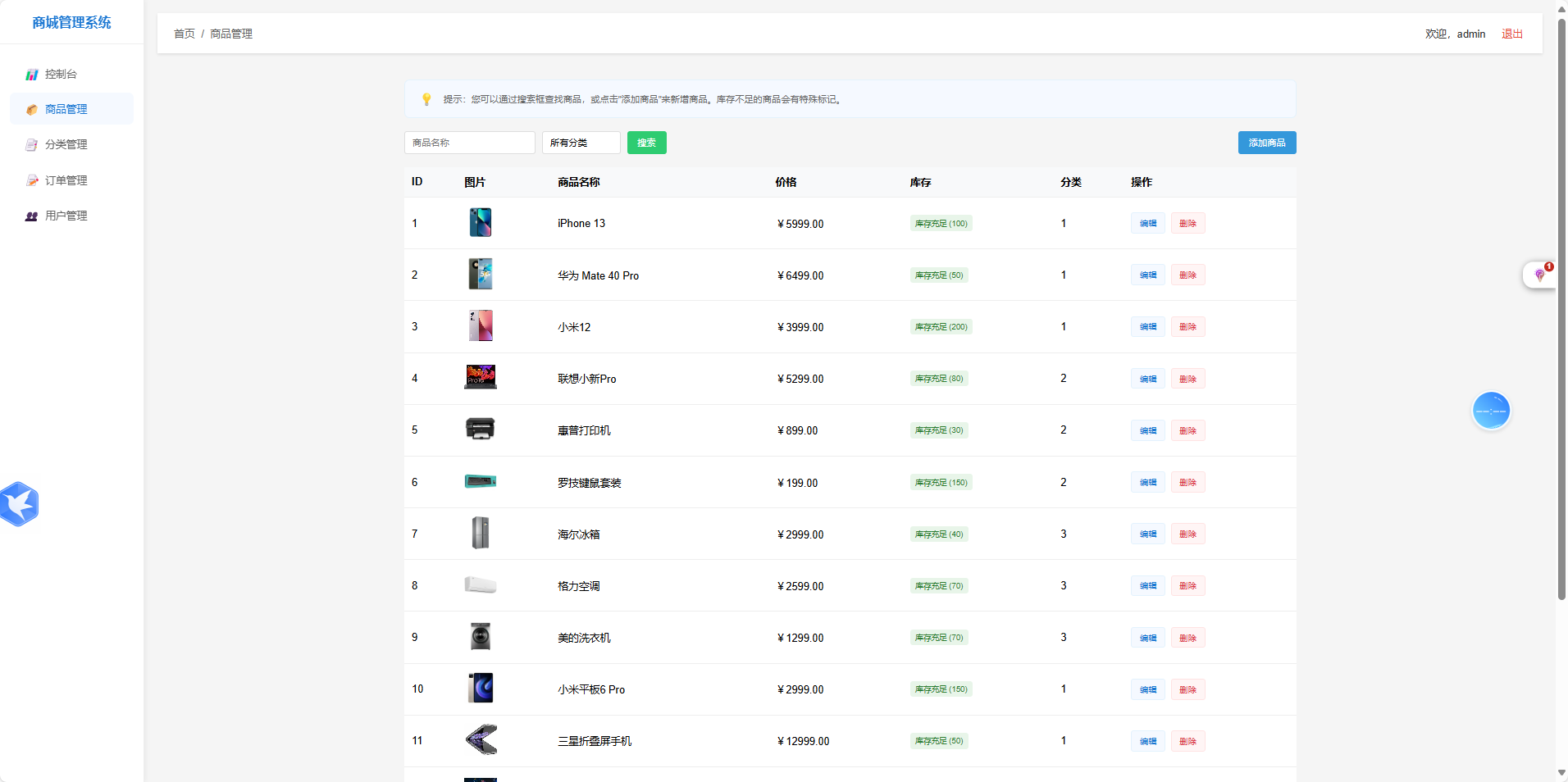
### **商城后台管理系统**

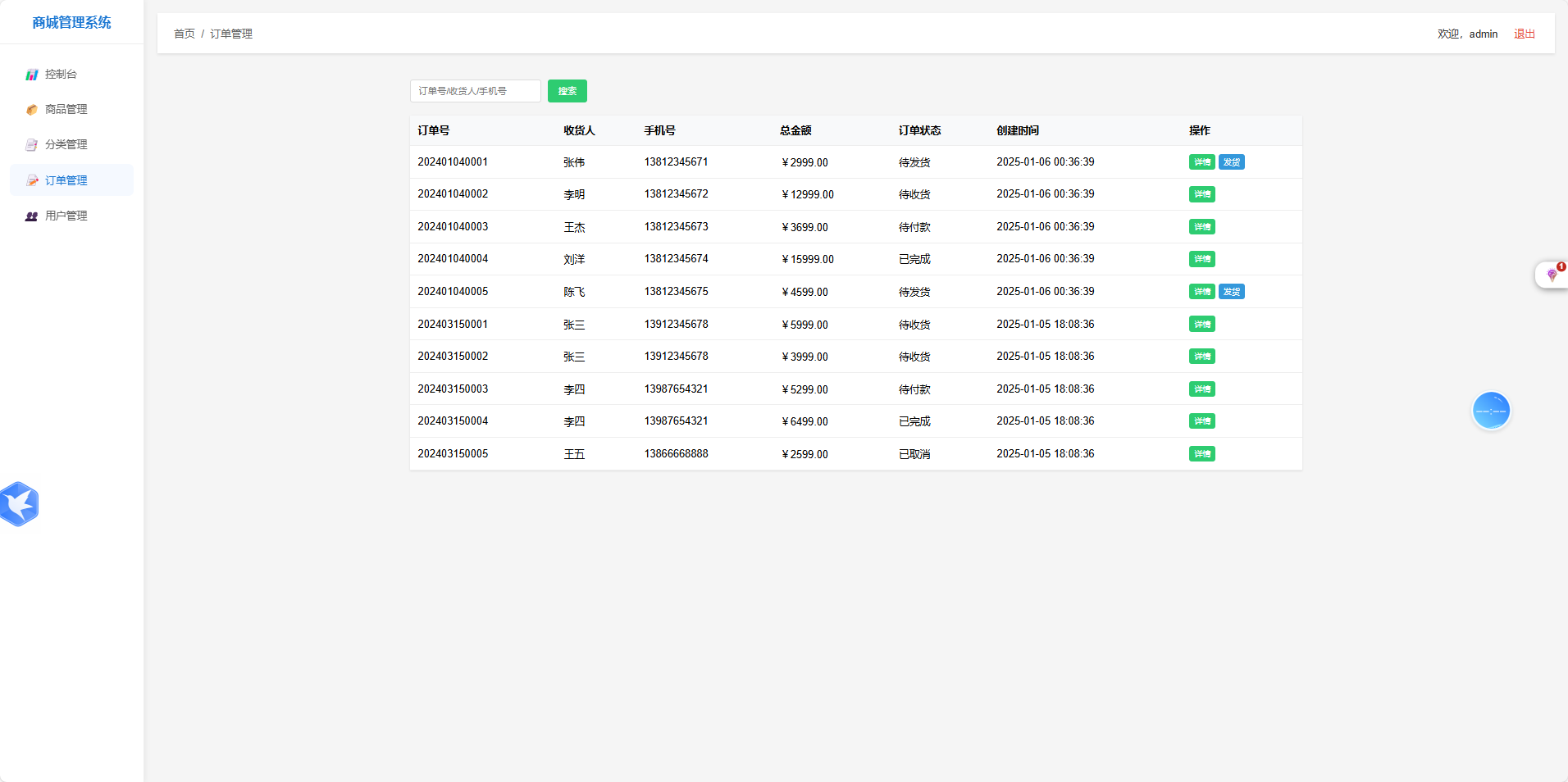
* ****项目概述****：这个商城后台管理系统是用 Java Web 技术做的，主要是为了能更好地管理商城里的各种信息，像商品、订单、用户这些信息的管理都能变得更精细、更方便。通过它，管理员可以轻松实现对商城全方位、精细化的管理，极大地提高工作效率，让商城在激烈的市场竞争中始终保持领先优势，为商家创造更多的商业价值，为消费者带来更优质的购物体验。
* ****主要功能定位****：
  + ****商品分类管理****：能够创建、编辑和删除商品的分类，让商品的分类清楚明白，不管是商家还是消费者，都能很快地找到自己想要的商品，而且还方便统计各类商品的销售情况。
  + ****商品信息管理****：负责商品的各种详细信息的录入、更新、删除和查询，像商品叫什么名字、长什么样、有什么描述、价格是多少、规格是什么等等，这样就能保证商品展示得又准确又及时。
  + ****订单管理****：能全程跟踪订单的情况，从订单生成到完成的每一步都能看到，可以查看订单列表，更新订单的状态，还能查看订单的详细信息，帮助商家及时处理订单，让顾客更满意。
  + ****用户管理****：主要是把用户的信息集中管理起来，比如修改用户的资料，管控用户账户的状态等等，这些数据还能为精准营销和个性化服务提供支持哦。
* ****技术特色****：
  + ****前端****：用 Boostrap相关知识来搭建页面的结构和样式，再加上 JavaScript 实现一些互动的效果，JSP 作为动态页面技术，让前端和后端能更好地进行数据交互，这样界面就会又好看又好用。
  + ****后端****：靠 Java Servlet 来处理网络请求，用 JDBC 来连接数据库，让系统能稳定又高效地运行，快速地对前端的操作做出反应。
  + ****数据库****：选择了 MySQL 8.0，它能很好地存储和管理商城的大量数据，保证数据的安全和查询的高效。
  + ****服务器****：部署在 Tomcat 9.0 服务器上，利用它成熟的特性，给系统提供一个稳定的运行环境。
* ****功能模块****：
  + ****登录模块****：



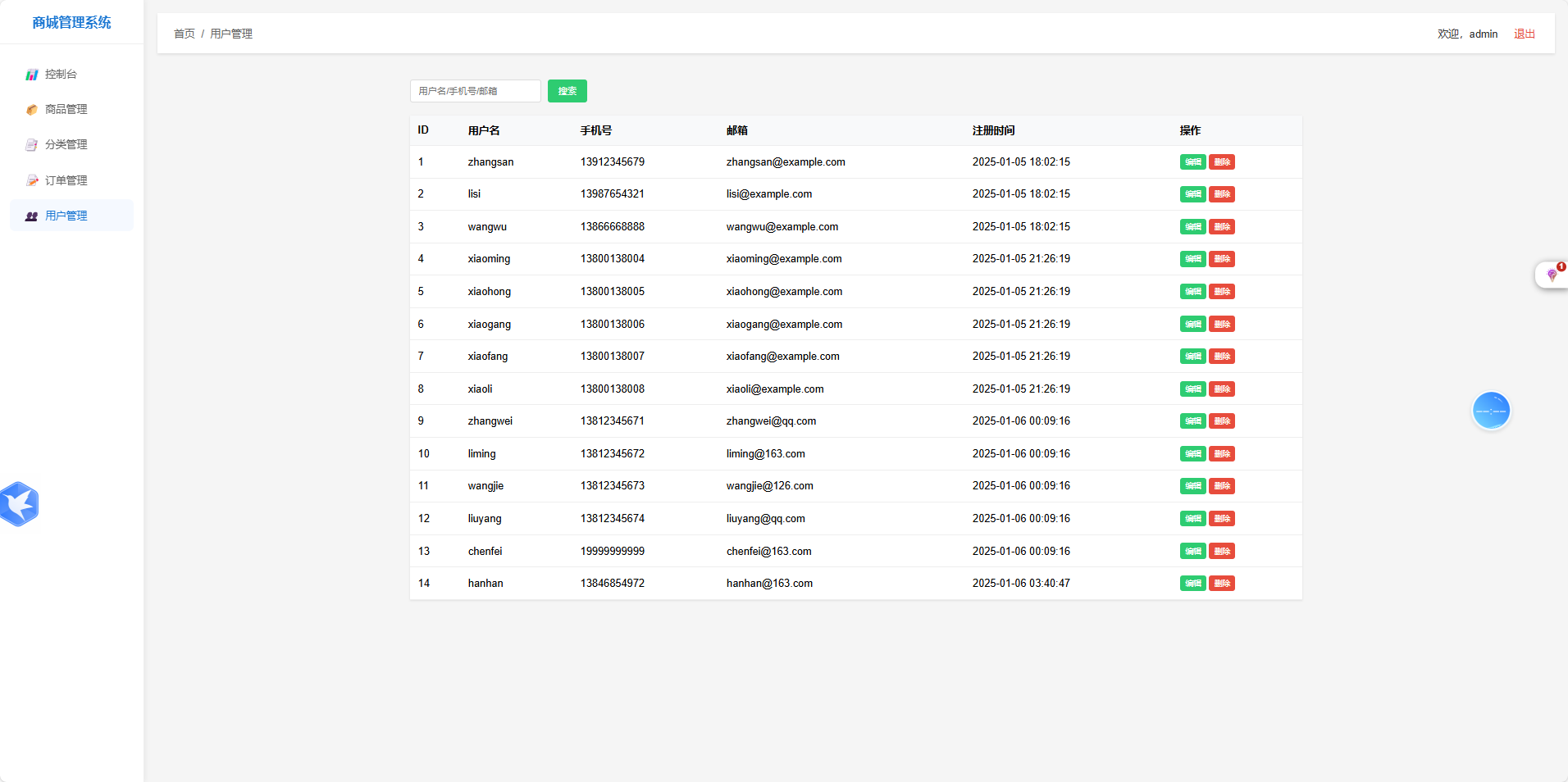
* + - ****登录界面****：设计得很简洁，管理员只要输入用户名、密码和验证码即可登录。
    - ****管理员登录****：后端会仔细检查用户名和密码对不对，和数据库里存的信息比对一下，防止有人非法登录。
    - ****密码确认验证****：密码的第二次输入能有效的保证登录的严谨度。
    - ****验证码功能****：会生成有数字的随机验证码图片，防止有人用机器暴力破解，验证码还有时间限制，过期了就得重新输入。
    - ****Session 会话管理****：登录成功后会创建一个 Session 来记录管理员的登录状态，在操作过程中会一直验证 Session 是不是有效，超时了就会自动退出登录，这样系统就更安全。
  + ****商品管理****：



* + ****商品分类管理****：支持进行分类查询和商品名查询，大大提高商品归类和查找效率。
    - ****商品信息管理****：新增商品时有很多录入的表单，能上传图片，还能用富文本编辑商品描述；修改商品信息方便管理员管理，提高效率；删除商品时会弹出提示，防止误删；查询功能可以根据多个条件筛选，快速找到想要的商品。
    - ****商品库存管理****：能实时监控商品的库存数量，库存不够的时候会自动提醒，还支持库存盘点和调整，也可以设置库存的安全阈值，保证商品能正常销售。
  + ****订单管理****：



* + - ****订单列表查看****：以列表的形式展示所有的订单，默认按订单创建的时间倒着排，能看到订单编号、下单用户、订单金额、订单状态等重要信息，方便快速了解订单的大概情况。
    - ****订单状态管理****：订单状态有很多种，像待付款、已付款、待发货、已发货、已完成、已取消等等，管理员可以手动更新订单的状态。
    - ****订单详情查看****：点击订单列表里的一项，就能展开看到订单的详细信息，包括收货地址、商品信息、订单状态、联系电话等等，这样处理售后问题就更方便。
  + ****用户管理****：



* + - ****用户信息管理****：可以查看和编辑用户的注册信息，比如姓名、联系方式、邮箱等，同时还会记录用户的注册时间。
    - **用户查询：**可以通过查询用户名或者手机号和邮箱等方式查询用户，大大提高管理效率，方便管理员管理。
    - **用户操作：**可以修改用户的信息，同时还可以对不正常的用户进行删除，提高系统的安全性。
* ****数据库设计****：
  + **数据库表结构：**

**admins（管理员表）：**用于存储管理员账号信息，实现管理员登录认证、权限管理等功能，确保系统管理的安全性与规范性。

**categories（商品分类表）：**用于管理商品分类信息，支持创建、编辑、删除商品分类，使商品架构清晰明了，方便商品归类与查找，同时便于统计各类商品的销售数据。

**products**（商品表）：**用于存储商品的详细信息，涵盖商品的基本属性、库存情况以及所属分类等，为商品信息管理（增删改查）、商品库存管理等功能提供数据支持，确保商品展示的准确性与及时性，同时便于商品分类统计与管理。**

****orders （订单表）：**用于管理订单信息，实现对订单从生成到完成的全流程跟踪，包括订单列表查看、订单状态管理等功能，为商家处理订单提供依据。**

****order\_items（订单项表）：**用于存储订单中每个商品的详细信息，建立订单与商品之间的多对多关系，通过关联 orders 表和 products 表，实现订单详情查看功能，方便商家和用户查看订单中具体购买的商品信息、价格、数量等，同时也有助于订单金额的计算和统计。**

****users（用户表）：**用于存储用户信息，实现用户注册、登录、信息管理等功能，为用户管理提供数据支持，包括用户信息管理、用户账户状态管理等，同时也为订单管理提供关联用户信息的依据，有助于精准营销与个性化服务。**

* + ****表之间的关联关系****

**admins 表与其他表无直接关联，但通过管理员账号登录系统后，可对其他表的数据进行管理操作。**

**categories 表与 products 表通过 category\_id 字段建立一对多关联关系，一个商品分类可以包含多个商品，而一个商品只能属于一个分类。**

**products 表与 order\_items 表通过 product\_id 字段建立一对多关联关系，一个商品可以出现在多个订单的订单项中，而一个订单项只对应一个商品。**

**orders 表与 order\_items 表通过 order\_id 字段建立一对多关联关系，一个订单可以包含多个订单项，而一个订单项只属于一个订单。同时，orders 表与 users 表通过 user\_id 字段建立一对多关联关系，一个用户可以下多个订单，而一个订单只属于一个用户。**