

e_commerce_product_list

项目开发文档

武汉大学计算机学院 陶大文
taodawen_0718@qq.com

项目概述

e_commerce_product_list 是一个基于 React + Redux 的电商商品列表项目，包含商品展示、筛选、搜索、购物车、用户登录等核心功能。

技术栈

前端框架: React 18

状态管理: Redux Toolkit

路由: React Router

UI 组件库: Ant Design

构建工具: Create React App

目录结构说明

/src 根目录文件

文件	功能描述
App.js	应用主组件，配置路由、主题和全局布局
App.css	应用全局样式，包含布局、响应式设计和自定义滚动条
App.test.js	React 默认测试文件
index.js	应用入口文件，渲染根组件
index.css	基础全局样式
reportWebVital.js	性能监测工具
setupTests.js	测试环境配置

/src/components 组件目录

组件文件	功能描述
------	------

组件文件	功能描述
ErrorBoundary.js	错误边界组件，捕获并处理子组件错误
FilterSidebar/index.js	商品筛选侧边栏，支持价格、分类、品牌、库存筛选
Footer/index.js	页面底部组件，展示网站信息和链接
Header/index.js	页面头部组件，包含 Logo、搜索框、用户操作
LoginModal/index.js	登录/注册模态框，支持用户认证
ProductCard/index.js	商品卡片组件，展示商品信息和操作按钮
SortHeader/index.js	排序和分页控制头部组件
LazyImage.js	图片懒加载组件（待完善）
VirtualProductList.js	虚拟滚动商品列表组件（待完善）

/src/hooks 自定义 Hooks

Hook 文件	功能描述
useDebounce.js	防抖 Hook，优化频繁的状态更新
useProducts.js	商品数据管理 Hook，封装 Redux 操作

/src/pages 页面组件

页面文件	功能描述
ProductList/index.js	商品列表页面，集成筛选、排序、分页功能
CartPage/index.js	购物车页面，管理购物车商品和结算

/src/store Redux 状态管理

Store 文件	功能描述
index.js	Store 配置和创建
productSlice.js	商品状态管理，包含筛选、排序、分页逻辑
cartSlice.js	购物车状态管理，支持增删改查操作
userSlice.js	用户状态管理，处理登录、注册、退出

/src/styles 样式文件

样式文件	功能描述
global.css	全局样式重置和工具类

/src/utils 工具函数

工具文件	功能描述
api.js	API 接口封装，支持真实 API 和模拟数据降级
debounce.js	防抖工具函数，包含多种防抖实现
mockData.js	模拟数据生成，用于开发和测试

核心功能模块

1. 商品管理模块

- 商品展示：网格布局展示商品卡片
- 筛选功能：价格范围、分类、品牌、库存状态
- 排序功能：价格、销量、评分排序
- 搜索功能：关键词搜索商品
- 分页功能：支持分页浏览

2. 购物车模块

- 添加商品：将商品加入购物车
- 数量管理：修改商品数量
- 删除商品：从购物车移除商品
- 清空购物车：一键清空所有商品
- 价格计算：自动计算总价和总数量

3. 用户认证模块

- 登录/注册：模态框形式的用户认证
- 状态管理：维护用户登录状态
- 模拟数据：内置测试账户用于演示

4. 状态管理架构

- Redux Toolkit：现代化 Redux 使用方式
- 切片模式：模块化状态管理
- 异步处理：使用 createAsyncThunk 处理 API 请求

数据流设计

1. **商品数据流**: UI → Redux Action → API → Redux Reducer → UI 更新
2. **筛选数据流**: 筛选操作 → 防抖处理 → Redux 状态更新 → 重新请求数据
3. **购物车数据流**: 用户操作 → Redux Action → 本地状态更新 → UI 反馈

特色功能

1. **响应式设计**: 支持移动端和桌面端
2. **错误边界**: 完善的错误捕获和用户提示
3. **性能优化**: 防抖处理、图片懒加载（待完善）
4. **降级策略**: API 失败时自动使用模拟数据
5. **用户体验**: 加载状态、空状态、操作反馈