**电商ERP多平台系统需求规格说明书**  
（西班牙本地化私有部署版）  
版本号：V1.0

**1. 技术栈与设计原则**

后端框架：Flask（应用工厂模式，Blueprints，Jinja2模板）

数据库：MySQL（FlaskSQLAlchemy ORM，FlaskMigrate迁移

前端：Bootstrap 5（响应式UI）、Jinja2模板、jQuery（表单/交互）、Chart.js（可视化）

认证：FlaskLogin角色的权限

自动化：APScheduler（定时任务）、自动同步、自动预警

AI集成：OpenAI API/本地LLM（知识库问答，FAQ，智能助手）

多平台扩展：支持TikTok Shop、SHEIN、TEMU、MIRAVIA、ALIEXPRESS

开发周期：1~3个月完成主要功能，优先核心业务，后续易扩展

用户体验：自适应布局，极简交互，数据可视化，批量操作，自动化提示

**2. 项目结构优化**

s7star\_erp/

├── app/ 主应用目录，所有业务功能模块均在此下分目录实现

│ ├── \_\_init\_\_.py Flask应用工厂，注册蓝图和扩展，统一入口

│ ├── config.py 配置文件（数据库、平台参数等）

│ ├── models.py 全局ORM模型（可拆分到各模块）

│ ├── extensions.py 第三方扩展注册（SQLAlchemy、Login等）

│ ├── scheduler.py 定时任务调度（自动同步、预警等）

│ ├── ai.py AI助手/知识库集成（AI相关业务逻辑）

│ ├── utils.py 全局工具函数

│ ├── platform/ 多平台适配（TikTok、SHEIN等API对接，平台抽象层）

│ ├── auth/ 用户-登录-权限-证相关（views.py、forms.py、service.py等）

│ ├── roles/ 角色管理（角色增删改查、分配权限，views.py等）

│ ├── permissions/ 权限管理（菜单、操作权限定义，views.py等）

│ ├── dashboard/ 仪表盘（首页、业务指标、趋势、地图等，views.py等）

│ ├── store/ 店铺管理（授权、配置、Webhook等，views.py等）

│ ├── product/ 商品管理（类目、品牌、商品、变体、图片等，views.py等）

│ ├── order/ 订单管理（订单、明细、标签、导出等，views.py等）

│ ├── inventory/ 库存管理（库存、日志、调整、预警等，views.py等）

│ ├── finance/ 财务管理（结算、交易、报表、税务等，views.py等）

│ ├── promotion/ 促销管理（活动、日历、效果分析等，views.py等）

│ ├── report/ 报表与分析（销售、客户、库存、绩效等，views.py等）

│ ├── webhook/ Webhook与API集成（事件订阅、回调等，views.py等）

│ ├── settings/ 系统设置、参数、备份、通知、监控等（views.py等）

│ ├── after\_sales/ 售后/退货/退款管理（工单、沟通、入库等，views.py等）

│ ├── logistics/ 物流/包裹/轨迹/物流商管理（views.py等）

│ ├── comment/ 评论与评价管理（统计、列表、分析等，views.py等）

│ ├── aiassistant/ AI助手/知识库/FAQ/智能推荐（views.py等）

│ ├── static/ 静态资源（css/js/img/map/chart/各功能资源，按功能细分）

│ │ ├── css/ 全局样式

│ │ ├── js/ 全局脚本

│ │ ├── img/ 全局图片

│ │ ├── map/ 地图相关静态资源

│ │ ├── chart/ 图表相关静态资源

│ │ ├── product/ 商品模块静态资源（如list.js、edit.css等）

│ │ ├── order/ 订单模块静态资源

│ │ ├── ... 其他功能静态资源

│ └── templates/ Jinja2模板（每功能一个子文件夹，页面分离，页面与views.py一一对应）

│ ├── base.html 基础布局模板

│ ├── layout.html 主布局模板

│ ├── dashboard/ 仪表盘相关页面（如index.html，对应dashboard/views.py）

│ ├── auth/ 用户、登录、权限相关页面（如login.html、user\_list.html，对应auth/views.py）

│ ├── roles/ 角色管理页面（如role\_list.html，对应roles/views.py）

│ ├── store/ 店铺管理页面（如list.html、edit.html，对应store/views.py）

│ ├── product/ 商品管理页面（如list.html、edit.html、import.html，对应product/views.py）

│ ├── order/ 订单管理页面（如list.html、detail.html，对应order/views.py）

│ ├── inventory/ 库存管理页面（如list.html、log.html，对应inventory/views.py）

│ ├── finance/ 财务管理页面（如settlement.html、transaction.html，对应finance/views.py）

│ ├── promotion/ 促销管理页面（如list.html、calendar.html，对应promotion/views.py）

│ ├── report/ 报表与分析页面（如sales.html、kpi.html，对应report/views.py）

│ ├── webhook/ Webhook配置页面（如config.html，对应webhook/views.py）

│ ├── settings/ 系统设置页面（如base.html、notify.html、backup.html，对应settings/views.py）

│ ├── after\_sales/ 售后管理页面（如return\_list.html、refund\_list.html，对应after\_sales/views.py）

│ ├── logistics/ 物流管理页面（如provider.html、package.html，对应logistics/views.py）

│ ├── comment/ 评论管理页面（如list.html，对应comment/views.py）

│ ├── aiassistant/ AI助手页面（如faq.html、chat.html，对应aiassistant/views.py）

│ └── ... 其他功能页面

├── run.py 启动入口

├── requirements.txt 依赖

├── init\_db.py 初始化数据库脚本

备注（页面说明）：

登录页：极简风格，支持记住密码，自动跳转

数据看板：今日订单、销售额、库存预警、销售趋势、热销排行、平台分布等卡片和图表

店铺管理：多平台店铺授权、状态、切换、Token管理

商品管理：商品列表、筛选、批量上下架、编辑、同步

订单管理：订单列表、状态筛选、详情、批量处理

库存管理：库存总览、预警、调整、日志

财务管理：结算、交易、报表

促销管理：活动列表、折扣设置、日历视图、效果分析

履约管理：批量发货、包裹管理、物流追踪、标签打印

退货/退款：退货申请处理、退款管理、售后工单跟进

物流管理：物流商配置、服务选择、运费设置、轨迹查询

报表分析：销售报表、客户分析、库存周转、绩效统计

用户权限：用户管理、角色配置、权限分配、操作日志

系统设置：基础参数、备份恢复、通知配置、监控告警

Webhook配置：事件订阅、回调管理、日志查看

AI助手/知识库：FAQ、智能问答、业务流程指引

自适应布局：顶部导航+侧边栏+主内容区，移动端自动折叠

**3. 数据库设计补充与细化**

**3.1 用户与权限相关**

users（补充字段：email, phone, last\_login, avatar）

roles（id, name, description, created\_at）

permissions（id, code, name, description, created\_at）

role\_permissions（role\_id, permission\_id）

user\_roles（user\_id, role\_id）

user\_logs（id, user\_id, action, detail, ip, created\_at）

**3.2 店铺与平台相关**

platforms（补充字段：auth\_url, callback\_url, app\_key, app\_secret）

shops（补充字段：region, logo, webhook\_url, last\_sync\_time）

webhooks（id, shop\_id, event, url, status, created\_at）

**3.3 商品相关**

categories（id, platform\_id, name, parent\_id, level, is\_leaf, created\_at）

brands（id, platform\_id, name, logo, created\_at）

product\_variants（id, product\_id, sku, name, price, stock, image, created\_at）

product\_images（id, product\_id, url, position, created\_at）

product\_audit\_logs（id, product\_id, status, remark, operator\_id, created\_at）

**3.4 订单相关**

orders（补充字段：buyer\_id, buyer\_contact, payment\_status, payment\_time, shipping\_status, shipping\_provider, tracking\_number, fulfillment\_status, platform\_order\_status, is\_cod, total\_product\_price, shipping\_fee, order\_total, original\_json, remark）

order\_items（id, order\_id, product\_id, variant\_id, name, sku, quantity, price, original\_price, platform\_sku\_id, product\_image, is\_gift, created\_at）

order\_tags（id, order\_id, tag, created\_at）

order\_notes（id, order\_id, user\_id, content, created\_at）

order\_timeline（id, order\_id, status, remark, operator\_id, created\_at）

order\_buyers（id, order\_id, buyer\_name, buyer\_phone, buyer\_email, buyer\_message, platform\_buyer\_id, created\_at）

shipping\_addresses（id, order\_id, recipient\_name, phone, country, province, city, district, address\_line1, address\_line2, postal\_code, is\_default, created\_at）

buyer\_history（id, buyer\_id, platform\_id, total\_orders, total\_amount, first\_order\_time, last\_order\_time, created\_at, updated\_at）

**3.5 履约与物流相关**

packages（id, order\_id, package\_code, status, shipping\_provider, tracking\_number, platform\_package\_id, split\_status, combine\_status, shipping\_documents, pickup\_config, last\_mile\_tracking, expected\_delivery\_date, created\_at）

package\_items（id, package\_id, order\_item\_id, quantity, shipping\_provider\_id, created\_at）

shipping\_tracking（id, package\_id, update\_time, description, location, created\_at）

logistics\_providers（id, name, code, contact, fee\_rate, service\_types, active\_status, created\_at）

shipping\_templates（id, name, provider\_id, default\_settings, created\_at）

shipping\_labels（id, package\_id, label\_url, created\_at, expires\_at）

**3.6 售后与退货相关**

returns（id, order\_id, status, reason, amount, return\_reason\_category, platform\_return\_id, shipping\_provider, tracking\_number, images, reject\_reason\_id, created\_at）

refunds（id, order\_id, amount, status, reason, refund\_type, platform\_refund\_id, created\_at）

after\_sales\_logs（id, order\_id, user\_id, content, type, created\_at）

reject\_reasons（id, platform\_id, reason\_code, reason\_text, category, created\_at）

return\_inspections（id, return\_id, inspector\_id, quality\_status, remark, created\_at）

**3.7 库存相关**

inventory\_logs（补充字段：reference\_type, reference\_id, original\_quantity, new\_quantity, change\_reason）

inventory\_adjustments（id, product\_id, type, quantity, operator\_id, reason, created\_at）

inventory\_warnings（id, product\_id, warning\_type, threshold, status, created\_at）

inventory\_forecasts（id, product\_id, forecast\_quantity, forecast\_date, created\_at）

inventory\_locations（id, product\_id, warehouse\_id, location\_code, quantity, created\_at）

**3.8 财务相关**

settlements（补充字段：currency, remark, settlement\_type, payment\_method, platform\_settlement\_id, items\_count, fee\_details, settlement\_period\_start, settlement\_period\_end）

transactions（id, shop\_id, order\_id, type, amount, status, transaction\_type, platform\_transaction\_id, reference\_id, fee\_details, tax\_details, created\_at）

tax\_reports（id, shop\_id, period, amount, status, tax\_details, created\_at）

financial\_accounts（id, name, account\_type, currency, balance, created\_at）

payment\_methods（id, name, type, fee\_rate, is\_active, created\_at）

**3.9 促销相关**

promotions（id, shop\_id, name, type, start\_time, end\_time, status, promotion\_type, platform\_promotion\_id, discount\_type, discount\_value, min\_order\_value, budget, remaining\_budget, rules, created\_at）

promotion\_items（id, promotion\_id, product\_id, discount, discount\_type, discount\_value, status, applied\_count, created\_at）

promotion\_effects（id, promotion\_id, date, views, orders, revenue, conversion\_rate, created\_at）

promotion\_calendars（id, shop\_id, start\_date, end\_date, events, created\_at）

**3.10 报表与分析**

report\_tasks（id, type, params, status, file\_url, created\_at）

dashboard\_cache（id, shop\_id, data\_type, data\_json, updated\_at）

sales\_statistics（id, shop\_id, date, orders\_count, sales\_amount, refund\_amount, created\_at）

product\_statistics（id, product\_id, views, sales, revenue, conversion\_rate, period, created\_at）

customer\_segments（id, shop\_id, segment\_name, segment\_criteria, customers\_count, created\_at）

**3.14 数据看板统计表**

daily\_shop\_metrics（id, shop\_id, date, orders\_count, gmv, sales\_amount, refund\_amount, new\_customers, shipped\_orders, avg\_order\_value, created\_at）

product\_performance（id, product\_id, shop\_id, date, views, clicks, add\_to\_carts, purchases, view\_to\_purchase\_rate, revenue, profit\_margin, created\_at）

inventory\_metrics（id, shop\_id, date, total\_sku\_count, in\_stock\_count, low\_stock\_count, out\_of\_stock\_count, turnover\_rate, avg\_days\_in\_stock, created\_at）

financial\_metrics（id, shop\_id, date, total\_revenue, total\_cost, gross\_profit, platform\_fees, shipping\_cost, marketing\_cost, net\_profit, profit\_margin, created\_at）

logistics\_performance（id, shop\_id, date, shipped\_orders, delivered\_orders, avg\_processing\_time, avg\_shipping\_time, on\_time\_delivery\_rate, late\_delivery\_count, created\_at）

customer\_metrics（id, shop\_id, date, total\_customers, new\_customers, returning\_customers, churn\_rate, avg\_customer\_value, top\_region, created\_at）

promotion\_performance（id, promotion\_id, shop\_id, date, views, participants, orders, revenue, cost, roi, conversion\_rate, created\_at）

regional\_sales（id, shop\_id, date, country, province, city, orders\_count, sales\_amount, customer\_count, created\_at）

hourly\_sales\_stats（id, shop\_id, date, hour, orders\_count, sales\_amount, visitors\_count, created\_at）

product\_category\_stats（id, shop\_id, category\_id, date, products\_count, sales\_count, sales\_amount, profit, created\_at）

**3.11 系统与配置**

system\_settings（id, key, value, description, updated\_at）

backup\_logs（id, type, file\_url, status, created\_at）

notifications（id, user\_id, title, content, type, is\_read, created\_at）

api\_logs（id, shop\_id, endpoint, method, request\_body, response\_body, status\_code, error\_message, duration\_ms, ip\_address, created\_at）

scheduled\_tasks（id, task\_name, task\_type, cron\_expression, last\_run\_time, next\_run\_time, status, created\_at）

system\_performance（id, cpu\_usage, memory\_usage, disk\_usage, response\_time, recorded\_at）

**3.12 AI与知识库**

ai\_knowledge（补充字段：category, tags, source, question, answer, keywords, relevance\_score）

ai\_logs（id, user\_id, question, answer, source, feedback, created\_at）

ai\_feedback（id, user\_id, log\_id, rating, comment, created\_at）

ai\_training\_data（id, content\_type, content, metadata, created\_at）

ai\_suggestions（id, user\_id, context, suggestion, applied, created\_at）

3.13 Webhook与API集成

webhooks（id, shop\_id, event, url, status, created\_at, auth\_token, last\_triggered\_at）

webhook\_logs（id, webhook\_id, request\_body, response\_body, status\_code, triggered\_at）

api\_keys（id, user\_id, name, key, permissions, expires\_at, created\_at）

integration\_configs（id, integration\_type, config\_json, is\_active, created\_at）

**4. 功能模块详细设计与业务流程**

**4.1 用户权限管理**

**4.1.1 用户管理**

- 设计思路：

- 用户表含用户名、密码、邮箱、手机号、角色、状态、创建时间、最后登录等字段。

- 支持用户增删改查、重置密码、禁用/启用、批量导入导出。

- 用户操作日志自动记录，支持按用户/时间/操作类型筛选。

- 业务流程：

1. 管理员进入用户管理页面，查看/搜索/筛选用户。

2. 新增/编辑/删除用户，分配角色，支持批量操作。

3. 用户可修改个人信息、密码。

4. 所有操作自动记录日志，支持导出。

**4.1.2 角色权限管理**

- 设计思路：

- 角色表定义不同角色（管理员、运营、财务、客服等），权限表定义所有菜单和操作权限。

- 角色与权限多对多关联，支持菜单树、按钮级权限。

- 支持角色的增删改查、权限分配、批量授权。

- 业务流程：

1. 管理员进入角色管理页面，查看/新增/编辑/删除角色。

2. 配置角色对应的菜单、页面、操作权限。

3. 用户登录后根据角色动态加载菜单和功能。

**4.1.3 登录认证**

- 设计思路：

- 使用Flask-Login实现会话管理，支持验证码、记住我、登录日志。

- 支持JWT接口认证，便于API/第三方集成。

- 业务流程：

1. 用户访问登录页，输入账号密码（可选验证码）。

2. 登录成功后跳转首页，失败提示原因。

3. 登录日志自动记录，支持异常登录告警。

**4.2 店铺管理**

**4.2.1 店铺授权**

- 设计思路：

- 对接TikTok Shop，支持输入APP\_KEY、APP\_SECRET、DEFAULT\_REGION、SHOP\_ID、授权地址、回调地址，其他平台后期开发。

- "平台配置"页面集中管理所有平台的授权参数，便于后续扩展。

- 授权流程配置化，支持多店铺切换、Token自动刷新、Webhook自动注册。

- 业务流程：

1. 管理员进入"平台配置"页，维护TikTok Shop参数。

2. 在店铺管理页，选择平台发起授权，系统生成授权链接。

3. 用户完成OAuth授权，系统保存Token。

4. Token到期自动刷新，异常自动预警。

**4.2.2 店铺设置**

- 设计思路：

- 店铺表含平台、名称、店铺ID、Token、状态、仓库、物流、Webhook等字段。

- 支持店铺信息编辑、仓库/物流配置、Webhook管理。

- 业务流程：

1. 管理员进入店铺设置页，编辑店铺基本信息、仓库、物流、Webhook。

2. 支持批量导入导出店铺信息。

**4.2.3 数据看板**

- 设计思路：

- 展示销售额、订单量、客户数、绩效等核心指标，支持多平台/多店铺切换。

- 图表卡片自适应，支持时间筛选、导出。

- 业务流程：

1. 用户进入看板页，查看各项业务指标。

2. 支持按平台/店铺/时间筛选，导出图表数据。

**4.3 商品管理**

**4.3.1 类目管理**

- 设计思路：

- 同步平台类目，支持树形展示、属性管理。

- 支持类目批量导入导出、属性编辑。

- 业务流程：

1. 用户进入类目管理页，查看/同步/编辑类目。

2. 支持批量导入/导出类目，编辑类目属性。

**4.3.2 品牌管理**

- 设计思路：

- 支持品牌创建、编辑、导入、同步。

- 品牌表含平台、品牌ID、名称、Logo等。

- 业务流程：

1. 用户进入品牌管理页，查看/新增/编辑/同步品牌。

2. 支持批量导入/导出品牌。

**4.3.3 商品发布/编辑**

- 设计思路：

- 商品表兼容多平台字段，支持多规格、变体、图片、价格、库存。

- 支持多步骤表单、批量导入导出、商品复制。

- 业务流程：

1. 用户进入商品管理页，新增/编辑商品，填写基本信息、规格、图片、价格、库存。

2. 支持批量导入/导出商品，商品复制。

3. 商品审核、发布、同步、上下架、删除、恢复。

**4.3.4 商品审核/导入导出**

- 设计思路：

- 支持商品发布前预检、审核流程、问题反馈。

- 支持Excel/CSV导入导出，导入校验、错误提示。

- 业务流程：

1. 用户批量导入商品，系统校验并提示错误。

2. 商品发布前自动预检，审核通过后发布。

3. 支持商品问题反馈、审核日志。

**4.4 订单管理**

**4.4.1 订单概览/列表**

- 设计思路：

- 订单表含平台、店铺、订单ID、状态、买家、明细、支付、物流等。

- 支持多条件搜索、批量操作、状态筛选、导出。

- 业务流程：

1. 系统定时同步平台订单，用户可按状态/时间/关键词筛选。

2. 支持批量确认、发货、取消、导出订单。

**4.4.2 订单详情/标记**

- 设计思路：

- 订单详情页展示买家、商品、支付、物流、操作历史。

- 支持自定义标签、优先级、异常标记。

- 业务流程：

1. 用户进入订单详情页，查看全流程信息。

2. 支持添加标签、优先级、异常标记。

**4.4.3 订单导出**

- 设计思路：

- 支持自定义导出字段、格式（Excel/CSV/PDF）。

- 业务流程：

1. 用户选择订单，点击导出，选择字段和格式。

2. 系统生成文件并下载。

**4.4.4 买家信息与地址管理**

- 设计思路：

- 买家信息表存储买家姓名、电话、邮箱、平台买家ID等基础信息。

- 订单收货地址表存储收件人、电话、国家、省市区、详细地址等。

- 买家历史记录表统计买家的订单次数、总金额、首次/最后下单时间等。

- 支持买家信息查询、地址管理、历史订单查看。

- 业务流程：

1. 同步订单时自动保存买家信息和收货地址。

2. 订单详情页展示完整买家信息和收货地址。

3. 支持查看买家历史订单和消费记录。

4. 发货处理时自动填充最新收货地址。

5. 提供买家画像分析，识别高价值客户。

**4.5 履约管理**

**4.5.1 发货处理**

- 设计思路：

- 支持批量发货、手动发货、发货单打印。

- 发货流程与平台API联动，自动推送物流信息。

- 业务流程：

1. 用户在订单页选择订单，批量发货或手动发货。

2. 系统自动推送物流信息到平台。

3. 支持发货单打印。

**4.5.2 物流追踪/包裹管理**

- 设计思路：

- 支持物流状态自动同步、异常处理、轨迹查询。

- 包裹可拆分/合并，支持包裹标签生成、状态更新。

- 业务流程：

1. 用户查看物流状态，支持轨迹查询、异常提醒。

2. 包裹拆分/合并、标签生成、状态更新。

**4.5.3 物流商管理**

- 设计思路：

- 支持物流商配置、费率设置、服务选择。

- 业务流程：

1. 管理员配置物流商、费率、服务。

2. 用户发货时选择物流商。

**4.6 退货退款管理**

**4.6.1 退货申请/审核**

- 设计思路：

- 支持退货申请、审核、确认、拒绝，自动同步平台售后状态。

- 业务流程：

1. 用户在订单详情页发起退货申请。

2. 管理员审核，确认/拒绝，系统同步平台。

**4.6.2 退款流程/售后跟进**

- 设计思路：

- 支持退款审核、记录、统计，售后工单、客户沟通、问题解决。

- 业务流程：

1. 用户发起退款，系统自动同步平台。

2. 售后工单、客户沟通、问题解决全流程记录。

**4.6.3 退货商品处理**

- 设计思路：

- 退货商品入库、质检、库存自动更新。

- 业务流程：

1. 退货商品入库，质检合格自动更新库存。

**4.7 库存管理**

**4.7.1 库存总览/操作**

- 设计思路：

- 库存总览、预警、入库、出库、盘点、调拨、日志一体化。

- 业务流程：

1. 用户进入库存页，查看总览、预警。

2. 入库、出库、盘点、调拨，自动记录日志。

**4.7.2 库存同步/预警/报表**

- 设计思路：

- 库存变动自动同步平台，支持多渠道库存、预警、补货建议、报表。

- 业务流程：

1. 库存变动自动同步平台，预警自动推送。

2. 库存报表、周转率、滞销分析一键生成。

**4.8 财务管理**

**4.8.1 结算管理/交易记录**

- 设计思路：

- 结算、交易、报表、税务一体化，自动同步平台财务数据。

- 业务流程：

1. 系统定时同步平台结算、交易数据。

2. 用户查看结算周期、明细、核对异常。

**4.8.2 财务报表/税务管理**

- 设计思路：

- 支持销售、退款、手续费等多类型交易统计，自动生成财务/税务报表。

- 业务流程：

1. 用户查看财务/税务报表，支持导出、对账、异常预警。

**4.9 促销管理**

**4.9.1 促销活动/效果分析**

- 设计思路：

- 支持促销活动创建、折扣设置、活动排期、效果分析、日历、冲突检测。

- 业务流程：

1. 用户新建促销活动，设置折扣、时间、商品。

2. 系统同步到平台，自动统计效果，日历展示。

**4.10 数据分析与报表**

**4.10.1 销售/客户/库存/绩效分析**

- 设计思路：

- 多维度分析，支持趋势、排行、区域、目标、客户画像、库存周转、滞销、缺货、绩效等。

- 业务流程：

1. 用户进入数据看板，查看各类分析报表。

2. 支持自定义筛选、导出、定时推送。

**4.10.2 数据看板详细设计**

- 设计思路：

- 多层次看板设计：首页总览看板、业务模块看板、深度分析看板

- 数据缓存机制：定时计算聚合数据并缓存，减少实时查询压力

- 可自定义看板：用户可拖拽配置自己关注的指标和图表

- 预警指标突出显示：异常数据自动高亮，辅助决策

- 业务流程：

1. 系统定时（每小时/每天）计算统计数据并写入专用统计表

2. 用户访问数据看板时优先读取缓存数据，提高加载速度

3. 支持下钻分析，从宏观指标点击进入详细数据视图

4. 数据支持多种导出格式和定时推送配置

**4.10.3 看板类型与实现**

1. 销售看板

- 实现方式：

- 使用daily\_shop\_metrics表聚合销售数据

- 整合regional\_sales表实现地图分布

- 结合hourly\_sales\_stats实现销售时段分析

- 核心指标：GMV、销售额、订单量、客单价、转化率、热销商品、地区分布

2. 库存看板

- 实现方式：

- 基于inventory\_metrics表实现库存健康度分析

- 整合inventory\_forecasts表实现库存预测

- 核心指标：库存周转、滞销商品、缺货预警、库存价值、平均库龄、补货建议

3. 财务看板

- 实现方式：

- 基于financial\_metrics表构建财务图表

- 整合transactions表实现交易明细

- 核心指标：毛利润、净利润、费用构成、利润趋势、平台费用、税务统计

4. 客户看板

- 实现方式：

- 基于customer\_metrics和buyer\_history表构建客户分析

- 整合regional\_sales表实现客户地域分布

- 核心指标：客户增长、客户分层、客户价值、回购率、流失率、客户来源

5. 物流看板

- 实现方式：

- 基于logistics\_performance表构建物流分析

- 整合shipping\_tracking表实现物流轨迹统计

- 核心指标：发货时效、物流成本、物流异常、配送时间、准时率、物流商绩效

**4.10.4 数据可视化实现**

- 技术选型：

- 前端图表库：Chart.js（轻量）或ECharts（功能丰富）

- 地图可视化：基于GeoJSON的交互式地图

- 数据表格：结合排序、筛选功能的动态表格

- 交互设计：

- 时间范围选择器：支持今日/本周/本月/自定义日期

- 下拉筛选：按商品类别、物流商、店铺等维度筛选

- 视图切换：提供表格/图表/卡片多种展示方式

- 数据钻取：支持从汇总数据点击下钻到详细数据

**4.11 系统集成**

**4.11.1 API/Webhook/第三方集成/数据同步**

- 设计思路：

- API密钥、Webhook、第三方系统、数据同步统一管理，支持多平台API适配、定时/实时/增量/全量同步、调用日志、失败重试、异常告警。

- 业务流程：

1. 管理员配置API密钥、Webhook、第三方集成。

2. 系统自动同步平台数据，所有API调用、Webhook事件自动记录日志。

**4.12 系统设置**

**4.12.1 基础/通知/监控/备份/参数**

- 设计思路：

- 企业、时区、语言、通知、监控、备份、参数等基础设置集中管理，支持自动备份、恢复、系统性能监控、操作日志、权限/菜单/参数动态配置。

- 业务流程：

1. 管理员进入系统设置，配置企业信息、时区、语言、通知、备份、参数等。

2. 系统自动备份、支持手动恢复、灾备演练。

**4.13 自动化与AI集成**

**4.13.1 定时任务/AI助手/FAQ/智能推荐**

- 设计思路：

- APScheduler定时任务，自动同步、预警、批量处理。AI助手集成OpenAI/本地LLM，支持FAQ、业务流程问答、智能推荐。

- 业务流程：

1. 系统定时自动同步多平台商品、订单、库存、财务等数据。

2. 自动检测库存低、Token过期、订单异常等，推送预警。

3. 用户可在任意页面通过AI助手提问，获取知识库/业务流程解答。

4. 支持FAQ维护、智能推荐、业务流程指引。

5. 自动化任务、AI助手均有可视化提示和日志。

**6. 优化后的开发策略与阶段划分**

第一阶段：UI框架与页面搭建

- [√] 项目结构搭建与配置（Flask应用工厂、数据库初始化）

- [√] 全局UI框架实现（导航、侧边栏、响应式设计）

- [√] 基础页面模板开发（列表、表单、详情、统计页通用结构）

- [ ] 所有功能模块的UI页面骨架实现（静态页面，暂不对接API）

- [√] 登录/注册页面

- [ ] 数据看板页面

- [ ] 商品管理相关页面（列表、详情、编辑）

- [√] 订单管理相关页面（列表、详情、操作）

- [ ] 库存管理相关页面

- [ ] 财务管理相关页面

- [ ] 设置/配置页面

第二阶段：核心功能模块实现

- [√] 用户认证与权限管理功能实现

- [√] 店铺管理与授权流程实现

- [ ] 商品管理核心功能实现

- [ ] 商品列表数据展示与筛选

- [ ] 商品详情查看

- [ ] 商品创建与编辑

- [ ] 商品状态管理（上架/下架）

- [ ] 订单管理核心功能实现

- [ ] 订单列表与筛选

- [ ] 订单详情查看

- [ ] 订单状态更新

- [ ] 简单发货处理

第三阶段：业务功能完善与API集成

- [ ] 平台API集成与数据同步

- [ ] TikTok Shop API客户端封装

- [ ] 自动同步机制实现

- [ ] Webhook事件处理

- [ ] 库存管理功能完善

- [ ] 履约与物流管理功能实现

- [ ] 财务管理功能实现

- [ ] 数据看板功能实现

**6. 功能参考**

领星：<https://erp.lingxing.com/>

店小秘：<https://www.dianxiaomi.com/>

Tiktok shop API文档：<https://partner.tiktokshop.com/doc/page/63fd743c715d622a338c4e54>

**7. 补充**

**数据看板模块**

**模块 1：全平台销量排行榜**

* **展示信息：**
  + **五个平台综合销量 TOP20 的 SKU（按周 / 月销量排序）**
  + **各平台独立的销量 TOP5 SKU（区分亚马逊、TikTok Shop 等平台）**
  + **近 7 天 / 30 天/自定义时间的销量环比增长率**
* **展示形式：**
  + **电脑端：横向柱状图展示综合排行（柱长代表销量，标注平台来源）；5 个并列小柱状图展示各平台独立排行**
  + **手机端：可左右滑动的卡片列表，每张卡片显示 SKU 图片、销量、增长率**
* **解决问题：**
  + **明确回国采购优先级，优先选择综合销量高且增长快的 SKU**
  + **支持平台差异化选品（如亚马逊侧重高毛利款，TikTok Shop 侧重性价比爆款）**
  + **通过环比增长筛选潜力款，避免遗漏销量上升的新晋爆款**

**模块 2：区域需求分布看板**

* **展示信息：**
  + **西班牙前 5 大自治区（如马德里、加泰罗尼亚）的销量占比**
  + **各区域的 TOP3 热销品类（如马德里的 3C 品类、巴伦西亚的家居品类）**
  + **区域销量环比变化（对比上周 / 上月）**
* **展示形式：**
  + **电脑端：西班牙地图热力图（颜色深浅代表销量）；右侧表格展示区域、品类、销量三维数据**
  + **手机端：简化地图（仅显示前 3 大区域）；点击区域通过下拉列表显示热销品类**
* **解决问题：**
  + **支持本地化选品决策，根据区域偏好调整采购品类（如巴塞罗那增加户外用品采购）**
  + **辅助物流前置规划，在高销量区域提前租赁海外仓（如马德里销量占比 30%）**
  + **规避低需求区域的无效采购投入**

**模块 3：平台表现对比看板**

* **展示信息：**
  + **五大平台的销售额占比（如亚马逊、TikTok Shop 等平台的贡献比例）**
  + **各平台客单价对比（如亚马逊客单价 50 欧元，TikTok Shop 客单价 30 欧元）**
  + **各平台特有的爆款 SKU（如 SHEIN 的快时尚单品）**
* **展示形式：**
  + **电脑端：饼图展示平台销售额占比；折线图对比各平台客单价趋势**
  + **手机端：横向条形图对比平台客单价；标签式切换查看各平台爆款**
* **解决问题：**
  + **按平台销售额占比分配采购预算（如亚马逊占比高则多备货）**
  + **针对平台特性调整采购策略（如 TikTok Shop 低客单价，侧重引流款）**
  + **捕捉新兴平台红利（如 SHEIN 的美妆工具品类潜力）**

**模块 4：商品复购率看板**

* **展示信息：**
  + **全平台复购率 TOP10 的 SKU（复购用户数 / 总购买用户数）**
  + **各平台复购率对比（如亚马逊复购率 vs TikTok Shop 复购率）**
  + **复购用户的地域分布（如瓦伦西亚地区复购率较高）**
* **展示形式：**
  + **电脑端：表格展示 SKU、复购率、区域分布；环形图展示复购率区间（＞30%/10-30%/＜10%）**
  + **手机端：卡片列表标绿高复购率 SKU；简短视频动态展示复购率趋势**
* **解决问题：**
  + **核心用户维系，对高复购率 SKU 增加安全库存（如数据线复购率 25%，补货量 ×1.5）**
  + **区域深耕，在复购率高的区域增加相关品类采购（如瓦伦西亚加购配件）**
  + **通过复购用户评价优化产品（如复购用户关注 “耐用性”）**

**模块 5：退货率分析看板**

* **展示信息：**
  + **退货率 TOP10 的 SKU（退货订单数 / 总订单数）**
  + **退货原因分类统计（如质量问题、尺寸不符等）**
  + **各平台退货率对比（如亚马逊退货率 vs SHEIN 退货率）**
* **展示形式：**
  + **电脑端：红色预警表格展示 SKU、退货率、原因；柱状图分析退货原因占比**
  + **手机端：退货率＞15% 的 SKU 标红标签；高退货率 SKU 触发语音播报警示**
* **解决问题：**
  + **供应商优化，对高退货率 SKU 更换供应商（如耳机退货率 20%，寻找新工厂）**
  + **产品改进，根据退货原因调整采购标准（如 “尺寸不符” 多则增加尺码选项）**
  + **风险控制，暂停采购高退货率 SKU 以避免库存积压**

**交互需求：单看板整合与操作**

* **模块隐藏 / 显示：**
  + **电脑端：支持模块折叠（点击标题栏收起内容）和隐藏（右上角按钮一键隐藏，通过模块管理恢复显示）**
  + **手机端：底部 Tab 导航切换模块（点击 Tab 显示对应内容，长按 Tab 可隐藏模块，通过设置恢复）**
* **全局筛选：**
  + **按平台筛选（如仅查看亚马逊数据）、按时间筛选（近 7 天 / 30 天 / 自定义时间范围），所有模块数据同步更新**
* **导出功能：**
  + **电脑端支持 Excel 导出采购清单，包含 SKU、建议补货量、供应商等信息**