

产品文档 · Emerge 电商 Demo

本 Demo 由我司基于开源电商内核构建，用于快速展示我们在“电商业务建模、前后端工程化与一键化交付”方面的能力。整体采用前后端分离、GraphQL API、容器化部署与基础观测的技术路线。

一、产品定位

- 面向对象：需要电商能力的企业客户（B2C/B2B/B2B2C 皆可拓展）
- 目标：以标准可复用内核 + 插件/二开方式，快速搭建可运营、可扩展、可观测的电商系统
- 价值：
 - 上手快：本地/测试环境一键部署
 - 可扩展：GraphQL API、Webhook、App 插件
 - 易运维：容器化、日志、APM、邮件沙箱

1.1 目标用户画像 (Personas)

- 业务方（电商/运营经理）：关注上新、促销、订单履约、报表导出
- 技术方（开发/架构/运维）：关注扩展性、接口规范、发布与监控
- 售前/演示人员：关注快速准备 Demo 数据与一键脚本

1.2 典型使用场景

- 快速搭建样板站点用于评审/投标/路演
- 以 Demo 为蓝本做定制化功能验证 (Poc → MVP)
- 作为私有化项目的本地沙箱环境（联调与培训）

二、功能总览（开箱即用）

- 商品与目录
 - 类目/集合、属性与规格、SKU 变体、库存与仓库、媒体资源
- 搜索与浏览
 - 类目/集合页、搜索、筛选、分页、SEO 元数据
- 价格与营销
 - 多渠道价格、促销/优惠券、礼品卡
- 购物车与结算
 - 单页结算、地址簿、配送方式、税费、支持多种支付网关（可选 Adyen/Stripe，Demo 默认关闭或使用沙箱）
- 订单与售后
 - 订单查看、发票、退款/取消（需支付配置）
- 多渠道与国际化
 - 渠道 (Channel)、多币种、税务与税类、语言（前台可扩展 i18n）
- 用户与权限
 - 账号、地址簿、后台员工角色与权限组
- 运营后台 (Dashboard)
 - 商品、订单、促销、渠道、插件、Apps、系统配置

说明：Demo 默认使用内置示例数据与 Mailpit 邮件沙箱；若需支付与发票等实流转能力，可按需接入沙箱网关（如 Stripe/Adyen）。

2.1 功能矩阵（简表）

- 商品管理：SPU/SKU、属性/规格、媒体、库存、仓库、批量导入/导出
- 价格体系：多渠道定价、税类、含/不含税价展示
- 营销能力：促销（Promotion/Rule）、优惠券（Voucher）、礼品卡
- 订单中心：下单、拆单/部分履约、取消、退款（依赖支付）、发票
- 结算流程：单页 Checkout、配送方式、税费、可插拔支付
- 用户中心：注册/登录、地址簿、订单列表
- 国际化：渠道、币种、税务；前台可扩展 i18n

2.2 Demo 数据与账号

- 默认导入两套渠道与若干类目/商品（含媒体与价格）
- 管理后台账号（可在部署脚本中覆盖）：admin@gmail.com / [Emerg3t3ch!2025](#)
- Mailpit 捕获所有注册/下单邮件：<http://localhost:8025>

2.3 可选/扩展模块（按需启用）

- 支付网关：Stripe/Adyen 沙箱（见 [docs/Payments.md](#)）
- 搜索：可对接 Elasticsearch/Algolia（索引与同义词）
- 内容：接入 Headless CMS（如 Strapi/Contentful）作为图文详情/专题页来源
- 统计：对接 BI 平台（导出订单明细、GMV、转化漏斗）

三、技术架构

- 前端：Next.js 15（App Router、RSC、TypeScript、TailwindCSS）
- 后端：Saleor（Django + GraphQL），Celery（异步任务）
- 中间件：Postgres、Redis、Mailpit（邮件沙箱）、Jaeger（APM）
- 编排：Docker Compose
- 协议：GraphQL（前台/后台与核心系统通信）

```
flowchart LR
    A[Storefront Next.js] -- GraphQL --> B((API GraphQL))
    D[Dashboard] -- GraphQL --> B
    B -- DB --> C[(Postgres)]
    B -- Cache --> E[(Redis)]
    B -- Email --> F[Mailpit]
    B -- Traces --> G[Jaeger]
```

四、部署形态

- 本地开发：一键脚本 [scripts/deploy.sh](#)
- 测试环境：
 - GitLab CI 任务 [deploy_test](#)（SSH 到测试机）
 - 或本机执行 [scripts/remote_deploy.sh](#) 远程推送部署

- 关键端口（默认）：
 - Storefront: 3000, Dashboard: 9001, API: 8001, Mailpit: 8025, Jaeger: 16686

4.1 关键配置清单（上线前）

- 渠道（Channel）/币种
- 运输区域（Shipping Zone）/费用模板
- 税类（Tax Classes）与是否含税展示
- 仓库/库存政策
- 支付网关（沙箱或正式）

4.2 性能开关（前台）

- GraphQL 请求层缓存（RSC/HTTP 缓存）
- 图片优化（Next Image, WebP/AVIF）
- 预渲染策略（SSG/ISR）与按需动态渲染

五、可扩展性

- GraphQL 可扩展 Schema 与 Resolver
- Webhook 订阅与事件总线（订单、支付、库存等）
- App 插件（支付、税务、物流、内容等）
- 前台组件与样式可替换（Tailwind 主题化）

5.1 二开范围建议

- 领域扩展：会员、积分、阶梯价、B2B 合同价、组合商品/套餐
- 运维增强：灰度发布、蓝绿/金丝雀、日志聚合与审计
- 体验优化：多语言、深色模式、搜索联想与埋点

5.2 集成清单（样例）

- 支付：Stripe、Adyen、PayPal
- 税务：Avatax（北美）/自定义税率
- 物流：顺丰/菜鸟/自有 ERP
- 通知：短信/钉钉/飞书/邮件（SMTP）

六、非功能特性

- 安全：基于 JWT 的认证，权限组管理；演示环境禁用危险操作
- 可靠：容器化与幂等脚本；数据初始化脚本
- 可观测：Jaeger APM、容器日志与健康检查

6.1 目标指标（建议）

- 首页 P95 < 1s, 商品详情 P95 < 1.2s（同区域）
- 下单成功率 ≥ 99.5%（演示环境）
- 数据恢复 RPO ≤ 1h（测试/演示）

6.2 可观测指标（参考）

- Web 端：首屏可交互、CLS/LCP、错误率
- API：请求量/错误率/P95、数据库慢查询数、队列堆积
- 业务：下单量、转化率、优惠使用率、退货率

七、演示入口

- 前台：<http://localhost:3000>
- 后台：<http://localhost:9001>（管理员：admin@gmail.com / [Emerg3t3ch!2025](#)）
- GraphQL：<http://localhost:8001/graphql/>

生产化建议：接入真实支付网关、开启 HTTPS、配置对象存储（媒体）、外部缓存、日志/Audit、安全组网与备份策略等。

八、Roadmap（对外展示）

- Q1：多语言前台、可视化内容组件（CMS 对接）
- Q2：会员积分/等级、企业客户（B2B）价目表
- Q3：直播/短视频带货接口适配、搜索引擎（ES/Algolia）可插拔