

# 孙磊 – Platform Engineer / Developer Productivity Engineer

全栈工程底座 | PayPal 金融科技 | 开发者平台建设 | 环境自动化 / 多技术栈 / AI 工具链

邮箱: [erishen@qq.com](mailto:erishen@qq.com) | 手机: 18621796513 | 上海

个人站点: <https://erishen.cn> | GitHub: <https://github.com/erishen>

## 个人优势

专注平台工程和开发者工具建设, 有全栈工程底座支撑。在 PayPal 从零搭建覆盖 6 种语言栈 13 个电商平台的标准化 Docker 测试环境体系, 三层镜像缓存架构将冷启动从分钟级降至 10 秒内, 30+ 双架构 (arm64/amd64) 镜像推送 JFrog, 封装自助启动工具和 Makefile 统一入口, 让 AI Agent 和工程师都能无障碍自助启动环境; 同步建设 AI 辅助代码分析工具链, 注重根因分析和 pattern 文档沉淀, 形成团队知识库而非个人经验。技术栈覆盖 Docker、Makefile、Shell、JFrog、Python、FastAPI、React、Next.js、Node.js、TypeScript、Redis、gRPC。

### 核心竞争力

- 平台化思维 — 将高频、重复、易出错的工程流程抽象为可复用基础设施, 屏蔽底层复杂性, 让上层开发者或 Agent 专注业务
- 多技术栈环境治理经验 — 独立建设覆盖 PHP / Java / Ruby / Python / .NET / Node.js 共 13 个电商平台的标准化测试环境体系
- 多架构兼容深度实践 — 系统解决 arm64 (本地 Mac) 与 amd64 (CI / 云) 原生扩展不兼容问题, 三层镜像缓存架构可复用
- AI 工具链工程化 — 构建 AI 辅助代码分析工具, 将 <1 秒代码库采样与多模型分析整合为开发者日常工具
- 文档沉淀习惯 — 每个环境问题均有根因分析和可复用 pattern 文档, 形成团队知识库而非个人经验

## 工作经历

### PayPal 贝宝支付

#### 全栈工程师 | Platform & Tooling | 2024.07 – 至今

2026 年主要承担 AI 支付迁移验证基础设施的从零搭建, 为 AI Agent 和工程师提供标准化多技术栈电商测试环境、自动化验证流水线和 AI 辅助工具链。

#### 工作内容

- 设计并实现覆盖 6 种语言栈 (PHP、Java、Ruby、Python、.NET、Node.js) 共 13 个电商平台的标准化 Docker 测试环境体系
- 建立三层镜像缓存架构 (依赖缓存层 + 平台编译层 + 应用镜像层), 统一解决各平台原生扩展多架构兼容问题
- 封装自助启动工具, 实现环境变量注入、架构自动检测、volume 预填充、数据库初始化一体化, 屏蔽底层复杂性
- 使用 Makefile 统一各平台操作命令入口 (start / stop / reset / test / build), 形成标准化工程规范
- 构建 AI 辅助代码分析工具, 集成 MCP + 多模型 (Claude / GPT-4 / DeepSeek / Gemini), 支持大型代码库快速采样分析和 Docker 配置自动生成
- 建立 docker-compose-patterns.md 和 issues-found.md 知识库, 按学习价值排序沉淀环境工程经验

#### 工作业绩

- 建设 13 平台电商支付测试体系，推送 30+ JFrog 双架构 (arm64/amd64) 镜像，支持离线完整冷启动
- 新项目环境接入时间：从此前需要半天排查，缩短至 1-2 小时
- Ruby gem 冷启动：从 2 分钟缩短至 10 秒；Java Maven 构建：从 5 分钟缩短至 10 秒
- AI Agent 可在测试环境中独立完成环境启动 + 支付流程验证，无需人工介入处理环境问题
- 建设 60+ Playwright E2E 测试用例，覆盖购买、授权、捕获、取消、退款等核心支付链路
- 识别并修复 OrbStack BuildKit 独立网络命名空间、VirtioFS nested bind mount 静默失效等复杂平台问题，产出可复用解决方案文档
- 参与 Next.js 12→14、Node.js 18→20 技术栈升级；负责客户身份验证平台从 0 到 1 建设并上线

**技术栈：** Docker、Makefile、Python、Shell、JFrog、Playwright、Node.js、Ruby、Java、PHP、.NET

---

## 上海全特信息科技

**全栈工程师 | 2024.01 – 2024.06**

负责迪士尼度假区 AI 数字人客服系统前端集成，1 个月完成初步调通。

### 工作内容

- 使用 React、Three.js、react-three-fiber 实现 Web 端 3D 数字人渲染、动作和表情控制
- 集成 Azure ASR / TTS 语音联动，解决 iOS / Android 移动端 WebGL、音频播放兼容问题

### 工作业绩

- 1 个月内完成 AI 数字人系统首版集成调通，支撑项目进入演示阶段

**技术栈：** React、Next.js、Three.js、react-three-fiber、Azure AI、Socket.io

---

## 携程旅游网络技术

**高级前端工程师 → 资深前端工程师 | 2017.04 – 2024.01**

负责 Trip.com 国际站 Blog / Destinations 内容频道开发和维护，推动团队 React / Node.js 技术转型。

### 工作内容

- 构建部门首个 Node.js SSR 内容平台，推动团队从 .NET 技术栈迁移到 React / Node.js
- 长期支撑 Trip.com 国际站多语言内容频道迭代，持续优化 Core Web Vitals 和 SEO 表现

### 工作业绩

- 推动团队技术栈转型，建立 SSR / SEO / Hybrid 多端交付体系
- 日常带领 2-4 人前端小组，建立开发规范、Code Review 和技术分享机制

**技术栈：** React、Next.js、Node.js、Express、Redis、MySQL、React Native

---

## 上海久振网络科技

**前端工程师 → 前端开发经理 | 2015.03 – 2017.04**

### 工作内容

- 负责社交 APP 的 React Native 跨平台开发，支持 Android / iOS 双平台核心功能
- 管理 4-7 人前端团队，搭建 Jenkins 多分支自动化编译流程

**技术栈：** React、React Native、Node.js、PHP、MySQL、Jenkins

---

## 迈智微电子（上海）

**软件工程师 | 2009.10 – 2015.03**

## 工作内容

- 负责全家拍 Hybrid 移动 APP 开发，实现照片管理与分享功能
- 负责路由器嵌入式 Boa WebServer 及管理界面开发，参与中美团队协作

**技术栈：** Cordova、HTML5、JavaScript、Java、Linux / C / Shell

---

## 中兴通讯股份有限公司

软件工程师 | 2008.10 – 2009.10

### 工作内容

- 负责 3G 网管平台告警模块开发，实现告警任务处理和桌面客户端展示

**技术栈：** Java、Swing、EJB、JBoss、Oracle

---

## 策堤特信息科技（上海）

软件工程师 → 技术经理 | 2007.04 – 2008.10

### 工作内容

- 负责租房网站、医生预约等 Web 项目开发，管理 3 人技术团队

**技术栈：** Java、Struts、Spring、PHP、JavaScript、jQuery

---

## 江苏省金思维信息技术

软件工程师 | 2005.04 – 2007.04

### 工作内容

- 负责企业 CRM 系统定制开发和项目实施

**技术栈：** Java、JavaScript、Struts、Tomcat、MS SQL Server

---

## 项目经历

---

### PayPal – 多技术栈电商测试环境平台

2026 | Docker、JFrog、Shell、Makefile、Playwright、多语言技术栈

为 AI 支付集成验证团队设计并实现统一测试环境基础设施，目标是让 AI Agent 和工程师拿到项目后，不了解底层细节也能在 5 分钟内跑起来，且 arm64 / amd64 行为一致。

- 覆盖 PHP×6 + Python×1 + Java×2 + .NET×2 + Node.js×1 + Ruby×1 共 13 个电商平台，20+ 独立 Docker 环境
  - 三层镜像缓存架构：以 Ruby 为例，拆分为 gem-cache（平台无关 gem 源码）→ bundle-cache（各架构原生扩展编译产物）→ 应用镜像，解决 arm64/amd64 原生扩展不兼容问题
  - 离线冷启动：npm prebuild-cache（@next/swc、better-sqlite3 等）和 Maven 依赖镜像预打包进 JFrog，容器按架构自动选取，无需外网
  - 自助启动接口：对外只暴露几个环境变量，全部初始化细节（镜像拉取、volume 填充、凭据注入、数据库初始化）由工具内部处理
  - Makefile 统一入口：各平台统一 start / stop / reset / test / build 命令，AI Agent 可直接调用
- 

## AI 辅助代码分析工具

2026 | Python、FastAPI、MCP、Claude / GPT-4 / DeepSeek / Gemini

面向大型代码库快速分析场景，<1 秒采样分析 4000+ 文件项目，多模型协作生成代码结构洞察和 Docker 配置。

- 集成 MCP 工具链，实现代码库结构采样、技术栈识别、依赖分析和配置文件生成
- 支持 Claude / GPT-4 / DeepSeek / Gemini 多模型接入，可按场景和成本路由
- 自动生成 Dockerfile 和 docker-compose.yml 草稿，开发者只需调整环境变量即可启动

**个人同类方向早期实践：** [ai-analyze](#) — 基于 MCP + Serena 的代码库采样分析工具

---

## PayPal – 大规模商户站点扫描分析

2026 | Python、Playwright、DNS、HTTP、数据分析

对数十万商户 URL 进行分层扫描，识别旧版支付集成站点，支撑业务团队迁移决策。

- DNS 预检 + HTTP 静态扫描 + Playwright 深度探索三层架构，逐层过滤无效目标
  - 识别数千个 WPS 旧版支付集成站点，分析 auto-submit / encrypted / hosted / cart 等集成模式分布
  - 整合三源威胁情报 (URLhaus / Phishing.Army / HaGeZi)，自动过滤恶意站点
  - 产出多分类结构化站点数据集，直接支撑业务团队迁移优先级决策
- 

## PayPal – 客户身份验证平台

2025 | Next.js、Node.js、Express、Redis、gRPC、K8s

由本人负责从 0 到 1 的设计开发、测试验证、安全审计和生产发布。

- 搭建项目基础结构、页面路由、BFF API、Redis 缓存和 K8s 生产部署配置
  - 实现隐私政策管理和用户授权追踪，满足 PIPL 合规要求
- 

## 全特科技 – 迪士尼 AI 数字人客服

2024 | React、Next.js、Three.js、react-three-fiber、Azure AI

- 接入 3D 数字人模型，实现 Web 端渲染、Azure ASR / TTS 语音联动、口型同步；1 个月完成首版
- 

## 携程 / Trip.com – 国际站内容平台

2020 – 2024 | React、Next.js、Node.js、Express、SSR、Redis、MySQL

**项目链接：** <https://www.trip.com/blog>

- 构建部门首个 Node.js SSR 内容平台，推动团队从 .NET 技术栈转型
  - 长期负责 Trip.com Blog / Destinations 多语言内容频道，优化 Core Web Vitals 和 SEO
- 

## 个人开源实践

---

围绕开发者工具、CLI 工程和平台标准化持续做个人实验，均为个人公开项目。

**开发者工具 / CLI：** [lobster](#)、[nsgm](#) — OpenAI Function Calling 兼容工具注册表 + REST API (面向 AI Agent 标准化工具调用)；Next.js + GraphQL + MySQL 全栈代码生成脚手架 (94+ 次提交持续维护)

**组件注册表 / 文档平台：** [shadcn-registry](#) — 自建 shadcn/ui 组件注册表 + Storybook 9 文档站，体现内部组件分发与标准化的平台工程思路

---

## 教育经历

---

南京信息工程大学 | 信息与计算科学 (本科) | 2001 - 2005

---

## 求职期望

---

**期望职位:** Platform Engineer / Developer Productivity Engineer / DevOps / Tooling Engineer

**工作地点:** 上海优先, 可考虑远程

**到岗时间:** 随时

**理想方向:** 开发者平台、内部工具、AI 工具链、金融科技基础设施、DevX 工程

---

我的优势不是运维某套现成 CI/CD 流程, 而是能从零识别工程师的痛点, 设计可复用的环境抽象和工具链, 让团队和 AI Agent 都能自助、可靠地完成工作。