

# 彭涛

24 岁 | 男 | 离职-随时到岗

深圳 邮箱 tpeng9240@gmail.com 电话 199 7255 2055 链接 breeze1203.github.io

Breeze1203



作品集

## 个人摘要

2 年后端开发经验，具备扎实的 Java 基础与高并发系统开发经验。熟悉微服务架构，在 JVM 性能调优与复杂慢 SQL 优化方面有丰富实战经验。

曾独立完成多端无状态认证中心的落地与 CI/CD 自动化流水线搭建，并通过全链路调优将核心系统有效吞吐量提升至 96.5%，具有较强的技术攻坚与工程化能力。

## 教育经历

本科 绥化学院 计算机科学与技术（信息工程学院） 2020.9 – 2024.6

## 工作经历

博纬智能识别科技有限公司 后端工程师(研发部) 2025.3 – 2026.2

负责公司主营业务(智能家居系统、AI 数字人平台、企业信息化管理)的后端架构设计与业务迭代。主导核心系统的高并发瓶颈攻坚(JVM、数据库调优)与中间件落地；推动团队工程化规范建设，搭建并维护 Nexus 仓库，提升构建效率 70%。

深圳金雨源科技 后端工程师(研发部) 2023.10 – 2025.3

深度参与“龙岗城投产业大脑”核心业务模块开发，经历从实习到全职的完整研发周期。主导团队 CI/CD 体系建设，引入 Jenkins 将部署周期从 1 小时缩短至 5 分钟；推动单体架构向 Spring Cloud 微服务架构演进。

## 项目经历

企业信息化管理平台(企业级 CRM/ERP 与协同系统) 2025.10 – 2026.2

支撑公司日常运转的核心 B 端中枢系统，涵盖 CRM、ERP、物料申请、内部审核及制作流等核心业务闭环

- 复杂业务流重构：梳理并重构公司核心审批 workflow，集成企业微信通知机制。消除人工干预环节，使平均审批流转时长缩短 40%。
- 海量数据平滑迁移：针对新老系统交替，设计基于双写与实时校验的数据映射策略。在保障业务零中断的前提下，成功合并近 10 万条用户与资产核心数据。
- 效率工具与性能攻坚：针对平台海量物料数据的批量导入痛点，引入 Fork/Join 并发框架重构底层逻辑，使 Excel 批量解析与入库效率提升 23%；基于 Spring Security + OAuth2.0 构建 RBAC 权限体系，保障跨部门数据隔离。

面向 C 端用户的智能生活一体化平台，支撑海量智能硬件设备的状态上报、指令下发及多端实时协同交互

- 面对海量设备并发心跳导致核心系统 Full GC 频繁的痛点，精细化分析 GC 日志，将垃圾回收器迁移至 G1GC 并优化 Region 参数，将系统有效吞吐量从 **88.7% 提升至 96.5%**，彻底消除接口卡顿。
- 缓存与慢 SQL 优化：针对设备状态高频查询场景，引入 Redis 缓存热点数据，并深度优化执行计划(联合重构)。使核心查询耗时从 **231ms 锐减至 124ms**，整体 QPS 提升 **24%**。
- 消息中间件解耦：引入 Kafka 应对高峰期设备数据激增，通过调优 Producer/Consumer 参数(acks/batch size)，使数据传输延迟**降低 20%**，系统吞吐量提升 **20%**。
- 多端统一认证：主导落地基于 JWT 的无状态认证中心，支持小程序/App 等多端用户无缝打通，设计无感 Token 刷新及防重放机制，极大提升用户登录体验。

## 元点 AI(AI 数字人视频生成平台)

2025.3 – 2025.7

结合前沿 AIGC 技术的创新型业务，基于深度定制化开源模型(深度集成 HeyGem)打造的数字人视频自动化生成管线

- AI 模型定制集成：负责后端业务流与底层开源大模型及 HeyGem 核心能力的深度对接，实现从文本/脚本到高质量数字人视频的转化逻辑。
- 异步渲染调度：针对数字人视频生成的高算力消耗和长耗时特性，设计高可靠的异步任务队列与状态机流转机制，确保高并发视频合成任务的不阻塞、防丢失及精准状态回调。

## 专业技能

---

**后端语言与框架:** Java(熟练, JVM 调优、多线程并发、集合框架、并发工具), SpringBoot, SpringSecurity, SpringCloud, ShardingSphere(了解), Go/Python(熟悉)。

**数据库与缓存:** MySQL(熟练, 查询优化、表结构设计、索引优化、性能调优), Redis (熟练, 缓存限流策略、集群应用、持久化、性能优化)。

**中间件与容器:** Kafka (熟练, 消息分发、高吞吐实践、配置优化), Docker (熟练, 容器管理、部署、Compose), Nginx(熟练, 配置、负载均衡、反向代理、性能优化)。

**开发工具与方法:** Linux Shell (熟练), Git (熟练), Maven (熟练), RESTful API 设计与规范, 设计模式, 分布式系统设计, 微服务架构。

**前端技术:** HTML, CSS, JavaScript, Axios, Vue3, React (具备前后端分离协作能力)。

**效率工具:** 熟练运用 AI 辅助编程工具(如 GitHub Copilot、Gemini、Claude Code)进行需求分析、代码生成与重构，提升开发效率。