

蔡志锋

求职意向: Java | 深圳

26岁 | 男 | 共青团员 | 3年经验 | 广西城市职业大学 (本科)

13377254691 | 481266251@qq.com

技能特长

- Java 核心:** 深入理解 JVM 内存管理、多线程与并发编程 (JUC), 掌握常用设计模式, 具备系统性能调优能力。
- 框架与架构:** 精通 Spring 全家桶 (Spring Boot/Cloud/MVC) 及 MyBatis、Netty 等主流框架, 具备微服务架构设计及高可用系统构建经验。
- 数据与缓存:** 熟练掌握 MySQL、PostgreSQL 及事务与索引优化, 熟悉 Redis、MongoDB 等 NoSQL, 可设计高效数据存储与缓存方案。
- 物联网领域:** 具备物联网平台、云原生边缘网关设计与开发, 熟悉 MQTT、CoAP、Modbus 及 TCP/HTTP 等通信协议, 设备接入、物模型设计及复杂场景交互能力。
- 消息队列:** 熟悉 RabbitMQ、RocketMQ、Kafka 等消息中间件, 支持数据采集与流量削峰; 具备边缘网关设计与优化经验, 降低延迟与带宽消耗。
- 运维与协作:** 熟悉 Docker、Nginx 及 CI/CD 流程 (Jenkins/k8s), 具备容器化部署与自动化运维能力; 掌握基础前端技术, 可高效协同开发。

工作经验

保游网 - Java工程师

2025-02 ~ 至今

工作职责: 负责软件系统核心代码的实现, 编写高质量代码与文档。依据需求完成模块的完整开发, 包括编码、调试、测试与维护。参与系统功能定义、架构设计与详细方案设计。

项目管理: 负责与产品沟通, 协调并明确需求, 推动项目开发进程; 进行日常任务分配与跟踪, 确保团队目标达成。

接口文档: 参与制定统一的接口规范, 并编写相关技术文档, 确保设计一致性。

技术沉淀: 撰写核心代码文档与技术方案, 定位并解决开发中的关键技术难题, 保障系统稳定与代码质量。

运维部署: 通过规范的编码、测试与维护流程, 支撑系统的高质量部署与稳定运行。

深圳简道智能有限公司 - Java工程师

2023-03 ~ 2024-12

工作职责: 参与物联网平台核心模块从0到1的重构与设计, 构建高可用、可扩展、稳定的系统, 通过模块化与服务化提升平台能力。

接口文档: 制定统一接口规范与版本管理机制, 标准化API设计, 降低沟通成本, 保障系统兼容性与扩展性。

技术沉淀: 编写技术文档与系统设计方案, 推动知识共享与团队成长, 定期组织技术分享, 持续更新文档。

运维部署: 运用Docker容器化部署与CI/CD流程, 确保多环境一致性、系统稳定与高效交付。

项目经验

支付平台 - 负责人

2025-03 ~ 至今

概述

作为项目负责人, 主导设计并开发了一个高可用、可扩展的统一支付平台。该平台通过抽象和封装, 统一接入了微信、支付宝、拉卡拉等多个主流支付服务商, 为公司内部各业务线 (如小程序、H5、PC网站等) 提供标准化、安全、稳定的支付、退款、数据统计等核心能力, 极大提升了业务方的支付接入效率和整个系统的可维护性。

背景

随着公司业务快速发展, 各产品线对支付功能的需求日益增多。为整合支付资源、降低接入成本, 公司决定构建统一的支付服务平台。本项目旨在聚合微信、支付宝、拉卡拉等支付渠道, 为各业务系统提供标准化、高可用、易扩展的支付能力支持, 实现支付业务的统一管理和高效运营。

技术栈

后端: SpringCloud Alibaba、MyBatis-Plus、Redis、RabbitMQ、Jenkins、Docker

前端: Vue3、Element UI

数据库: MySQL

项目架构设计

架构愿景：

- 主导设计并落地了一套高可用、高扩展、安全合规的统一支付中台体系，旨在通过标准化服务赋能全公司业务，提升研发效率与系统稳定性。

核心架构：

- 分层与模块化：采用清晰的分层架构（API网关/业务逻辑/数据持久化），并通过模块化将支付核心、商户管理、开放接口等解耦，实现高内聚、低耦合。
- 设计模式应用：核心运用策略模式与抽象工厂模式，将众多支付渠道（微信、支付宝、拉卡拉等）的差异封装在底层，向上提供唯一、统一的服务接口，实现了架构上的高度抽象。

关键设计决策：

- 动态切换支付通道：设计了 Appld 与多支付通道的绑定与动态切换机制，能在单一渠道故障时实现自动无缝切换，保障业务连续性。

安全体系构建：

- 构建了以API网关为核心的安全防线，实现基于 Appld + Secret 的认证鉴权。对所有敏感信息进行加密存储，并对关键操作记录不可篡改的审计日志，满足金融级安全与合规要求。

核心功能

多支付渠道统一接入：

- 成果：成功接入微信（直连/合作伙伴）、支付宝（直连/合作伙伴）、拉卡拉支付平台。

多场景支付能力覆盖：

- 成果：完整支持 JSAPI、小程序、H5、电脑网站等全端支付场景，无缝支撑公司所有业务线。

支付开放平台：

- 核心接口：设计并实现了统一下单与统一退款相关核心接口，提供给业务平台快速接入。
- 难点攻克：成功处理了不同支付渠道在订单状态、错误码体系、退款规则（如部分退款、货币单位）上的巨大差异，实现了完美的统一，极大降低了接入方的认知负担。

支付平台核心模块体系化建设：

- 认证鉴权体系：构建了类似微信开放平台的 Appld 管理体系，实现了支付平台与业务平台之间的安全认证与权限控制。
- 商户配置中心：打造了集中、安全的第三方商户配置管理模块，实现了敏感配置的加密存储与生命周期管理。

可靠订单与通知系统：

- 设计了全局唯一的平台订单号，并维护与多渠道订单号的映射关系，为全局查询与对账奠定基础。
- 实现了具备指数退避策略的可靠通知机制，确保业务方在极端情况下也能最终接收到支付结果。

财务风控流程：

- 为财务部门定制了「申请-审核-执行」的线上化退款流程，将技术系统与公司财务制度深度融合，有效管控资金风险。

数据运营支撑：

构建了数据统计模块，聚合各渠道流水，为管理层提供交易额、成功率、渠道分析等可视化报表，驱动业务决策。

C-AIoT 智慧物联协作平台 - Java后端开发

2023-03 ~ 2024-12

项目背景

C-AIoT智慧物联协作平台基于云边端协同架构，集成多协议通信与物模型管理，实现企业级物联网设备的统一接入、控制、数据采集与可视化管理，助力企业提升运营智能化水平。

技术栈

- 后端：Spring Boot/Cloud Alibaba、MyBatis-Plus、Redis、Elasticsearch、Netty、MongoDB
- 前端：Vue.js、Ant Design、wujie
- 数据库：MySQL
- 消息队列：RocketMQ
- 物联网协议：MQTT、CoAP、Modbus、TCP/UDP、WebSocket
- DevOps：Jenkins、Docker、Kubernetes

责任模块

- 授权中心、系统管理、设备管理、通信管理、智能场景、物模型、告警中心、任务调度中心等。

核心模块

- 授权中心：采用RBAC模型实现用户、设备及系统的统一身份认证与权限管理，基于OAuth2/JWT构建认证体系并支持单点登录

(SSO), 保障设备安全接入与系统安全。

- 系统管理: 提供用户、配置及日志管理等基础服务, 实现配置的动态管理与统一监控, 建立完整的日志追踪与健康检查体系以保障系统高可用性。
- 设备管理: 负责设备从注册、状态监控到配置同步的全生命周期管理, 实现设备数据的采集、存储、实时查询及云边端双向同步。
- 设备通信: 支持MQTT、CoAP、TCP/UDP等多协议通信, 通过消息队列确保设备与平台间消息的可靠传输与双向实时通信。
- 智能场景: 基于设备状态、时间或外部条件触发智能场景, 支持用户自定义自动化规则, 并协调多设备按预定规则执行联动操作。
- 物模型: 统一定义设备的属性、服务与事件, 通过版本控制确保设备接入兼容性, 并实现设备与物模型间的动态映射与灵活配置。

C-AIoT 数据中心 - Java后端开发

2024-03 ~ 2024-12

项目背景

C-AIoT数据中心项目通过构建统一的设备物模型管理平台, 整合IoT平台、边缘网关与前端设备组件, 实现设备模型的高效管理与展示, 提升平台兼容性、扩展性及用户体验。

技术栈

- 后端: Java、Spring Boot、MyBatis、Redis、RocketMQ
- 前端: Vue.js、Ant Design
- 数据库: MySQL
- DevOps: Jenkins、Docker、Kubernetes

责任模块

- 物模型管理、版本控制、在线编程、组件模块等

核心模块

- 物模型管理: 统一定义并管理设备属性、服务与事件, 确保设备数据准确性及功能灵活性, 支持物模型创建、存储与跨平台兼容。
- 版本控制: 管理物模型版本号与历史变更, 确保版本间兼容性, 提供迁移工具支持设备平滑升级。
- 在线编程: 支持物模型在线修改与实时生效, 无需设备重启; 提供在线调试、脚本编写功能, 动态调整设备行为, 提升开发运维效率。
- 组件模板: 基于物模型动态生成前端UI组件, 实现个性化展示; 优化设备交互界面, 提升用户操作体验与系统响应速度。

云原生边缘网关 - Java后端开发

2023-06 ~ 2024-12

项目背景

基于云原生架构设计并实现边缘网关, 专注于IoT设备的高效接入与管理, 支撑设备与云端平台的实时通信。项目致力于构建高可靠、易扩展的智能边缘计算平台, 实现多协议设备统一接入与管理。

技术栈

- 后端: Java、Spring Boot、MyBatis、Docker、MQTT
- 数据库: MySQL
- 通信协议: HTTP/HTTPS、WebSocket、TCP/UDP、MQTT、Modbus、CoAP等

责任模块

- 设备管理、通信管理、状态管理、协议转换、设备控制、智能场景、系统监控

核心模块

- 设备管理: 实现设备注册认证与物模型管理, 确保设备接入安全; 提供设备运行日志与操作记录管理。
- 通信管理: 支持多协议解析与高并发通信, 维护设备长连接与断线重连; 优化数据传输效率, 降低带宽占用。
- 状态管理: 实时监控设备在线状态与连接质量, 支持异常检测与上下线事件通知。
- 协议转换: 完成多协议解析与数据转换, 支持协议扩展; 统一封装JSON格式数据, 实现标准化平台交互。
- 设备控制: 提供远程指令下发与执行反馈, 支持定时与批量控制任务管理。
- 智能场景: 定义基于状态、数据、时间等触发条件的场景规则, 支持多设备联动与执行监控。
- 系统监控: 监控系统资源使用情况, 实现健康状态可视化预警; 提供日志查询与告警通知功能。

自我评价

资深Java开发工程师, 拥有扎实的技术功底与丰富的项目实战经验。对技术保持高度热情, 能快速追踪并应用行业新技术。具备良好的团队协作精神与抗压能力, 能在快节奏环境中保证代码质量与项目交付。