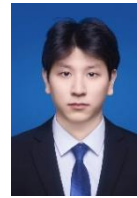


李道杭

年龄：22 岁 | 电话：13225999295 | 邮箱：13225999295@163.com
教育经历：厦门大学嘉庚学院-软件工程 本科 2020.09-2024.07



专业技能

后端：熟悉 javaSE、集合、Mysql、Redis、SpringBoot、MybatisPlus、RocketMQ、Nacos、Gateway、OpenFeign、Sentinel，了解并发、jvm。
前端：熟悉 HTML5、CSS、JavaScript、Vue3、Element Plus、Tailwind CSS，了解 Pinia、Vuex、Vite。
开发工具：熟练使用 Git 进行版本控制、apipost 基础测试工具、linux 基础操作。

工作经历

厦门思而立科技有限公司 全栈工程师 2024.08-2024.09

项目简介：汇智中台系统（海外 saas 电商平台）。

技术栈：SpringCloud、Mysql、JDBC、Vue3、Element Plus。

主要负责：

- 副屏设置模块和快捷商品模块：
 - 前端：实现副屏设置和快捷商品的列表、详情、修改页面组件化，界面语言 i18 国际化。
 - 后端：实现副屏设置和快捷商品的增删改查（CRUD）功能

项目成果：

- 通过 CodeReview，代码质量符合团队标准。
- 在测试环境中成功部署，验证了功能的稳定性和用户体验。

实习经历

福州夸度网络科技有限公司 java 后端 2024.06-2024.08

项目简介：抖音账号引流脚本系统。

技术栈：SpringBoot、Mysql、Element Plus。

主要负责：

- 权限控制模块：
 - 使用 AuthenticationInterceptor 检查请求的接口是否含有@UserLoginToken。存在则解析 JWT 进行身份验证。
- 账号模块：
 - 二维码收号与验证码业务：二维码登录流程，员工通过脚本生成的二维码扫码登录抖音账号。使用 DeferredResult 长轮询以及异步线程处理二维码生成和状态查询，以及验证码验证的上传和处理。
- IP 管理模块、窗口模块：
 - 将每个工作单元（网页）抽象成窗口。实现 IP、窗口增删改查（CRUD）功能。
- 通道模块：
 - 该模块抽象了数据传输通道（Channel），通过 Channel 统一配置和管理 WindowsGroup 与 ProxyGroup。

项目成果：

- 简化业务逻辑：
 - 通过业务分析发现并不需要 Redis，使用 ConcurrentHashMap 本地通信对象池替代 Redis 实现业务通信。
- 支持并发处理：
 - 使用 DeferredResult 和异步线程优化二维码收号与验证码业务，减少前端等待时间。

项目经历

Murmur62Link 短链接系统 2024.02-2024.04

项目简介：该系统是短链接系统的 SaaS 软件。该系统涵盖短链接展示、创建、短链接访问数据的统计与展示等功能模块。前端开发选用 Vue 作为框架，Element-Plus 充当组件库，借助 Axios 实现数据交互，运用 Vite 完成项目构建，并采用 echarts 进行数据可视化呈现，同时结合 Vuex 进行状态管理

技术栈：Vue、Element-Plus、Axios、Vite、echarts、Vuex

主要负责：

- 短链接展示模块：
 - 设计并实现短链接的列表展示页面，包括短链接的基本信息、访问次数等。
 - 确保页面数据的实时更新和准确展示，优化列表加载性能。
 - 负责页面布局和样式的优化，提升用户视觉体验。
- 短链接创建模块：
 - 构建用户友好的短链接创建页面，包括输入原链接、自定义短链接等功能。
 - 实现表单数据的前端验证，确保用户输入的有效性和完整性。
- 短链接访问数据统计模块：
 - 对各种图表组件进行模块化处理。
 - 设计直观的数据图表，通过 echarts 进行展示，方便用户理解和分析访问趋势。
- 数据展示模块：
 - 实现数据的分页、排序和筛选功能，提升用户对数据的操作便利性。

项目成果：

- 数据图表可扩展性增强：
 - 采用了模块化的设计架构。通过一个组件去集成不同图表，使得后续新增图表类型时，只需简单扩展模块。

CFXs Token 交易脚本 2023.12-2024.01

项目简介：该脚本实现 Conflux 网络中 CFXs Token 的交易的半自动化，包括批量合并，上架、下架。

技术栈：Node.js、JavaScript。

主要负责：

- 业务需求分析与设计：
 - 分析自身对 CFXs token 交易的细节与核心需求。
- 代码开发与优化：
 - 实现交易的半自动化，批量合并，上架、下架功能，确保交易操作的高效执行。
- 异常处理机制构建：
 - 设计并实现完善的异常处理机制，捕获网络波动、请求超时和服务器错误等异常情况。
 - 实现自动重试逻辑，确保在异常情况下交易请求的高成功率。

项目成果：

- 性能提升：
 - 通过 Promise.all 和 async/await 对多个 fetch 请求进行并行处理，交互速度提升 **500%**。
- 成功率提升：
 - 捕获失败交易，通过自动重试，和降低请求速度，将交易请求的错误率从最初的 **9%**降低至 **1%**。
- 产品落地：
 - 成功实现产品，在自己微信群内 **20+**名用户使用，累计交易流水超过 **10 万元**。

自我评价

知道如何正确提问，在提问前会明确目标、整理思路，对内容进行初步思考并收集资料，提问后落实结果。