

个人简历

基本信息

姓名：杨凯 年龄：23岁
性别：男 工作年限：一年经验
电话：15672647981 邮箱：durant202401@163.com

求职意向

职位：Java开发工程师 随时到岗

工作经历

2024-03~2025-05 西安五达信息科技有限公司 Java开发工程师

技能特长

- Java基础扎实，熟悉JVM原理、多线程、网络原理、设计模式、常用数据结构与算法
- 熟悉SpringBoot、Spring、Mybatis，有丰富实战经验，可以通过需求快速构建项目，深入研究Spring、SpringMVC、SpringBoot核心源码
- 熟悉MySQL索引、锁、隔离级别、MVCC机制及数据库调优，有SQL优化经验
- 熟练掌握Redis核心数据结构的使用，熟悉Redis持久化机制，熟悉多级缓存架构，熟悉各种缓存高并发的使用场景，比如缓存雪崩、缓存穿透、缓存失效等
- 熟练掌握SpringCloud Alibaba（Nacos、sentinel、seata），深入理解
- 分布式服务特点、旧服务改造、服务划分、服务分层与治理，有实战项目经验
- 熟悉常见消息中间件的使用，擅长处理消息丢失、重复消费、消息顺序性、消息积压等问题
- 熟悉Docker、Kubernetes，有部署微服务项目的经验

项目经验

信息化服务平台 Java开发

技术栈：SpringBoot+Activiti+MyBatis-Plus+Nacos+RocketMQ+XXL-JOB

项目描述：信息化服务平台是一款综合性解决方案，涵盖了企业日常运营的多个方面，它实现了先进的用户管理系统，包括员工信息、权限分配和角色管理，以满足企业组织架构的多样需求，平台还集成了智能数据分析模块，提供实时报表和可视化数据，助力企业决策，为了增强工作协同，我们设计并开发了实时通讯功能，支持团队间的及时沟通。信息安全是我们关注的重点，通过加密算法和访问控制，确保敏感数据的保密性。此外，我负责系统注册申请的流程+软件注册申请的流程。这个项目加深了对企业信息化需求深刻的理解，为企业提供了全面的信息化支持。

个人职责：

1.系统注册申请与软件注册申请

- 完成用户添加、查询、删除、修改等基本数据操作
- 基于POI相关技术，将用户在注册申请过程中填写的信息以标准化的表格形式导出。在导出过程根据不同的数据类型和业务需求，对表格的列标题、数据格式、单元格形式等进行设置，确保表格能够清晰地展示注册申请的各项内容，为提高导出效率，采用**缓存策略**，将经常导出的表格模板缓存起来，当再次导出相同类型的表格时，直接从缓存中获取模板并填充数据，减少重复的模板生成操作，加快导出速度。
- 处理其他部门通过接口导入数据的需求，根据格式进行解析，在解析过程中，基于**适配器模式**，将不同格式的数据适配成系统能够理解和处理的统一数据结构，实现了不同部门之间的数据共享与交互，提高了企业内部的数据流通效率与协同工作能力。同时，为确保数据导入的准确性和稳定性，在接口实现中引入数据校验机制，对导入的数据进行完整性、合法性检查。

2.数据扁平化设计与实现：针对工作流程数据的复杂性与冗余性问题，提出并实施了数据扁平化处理方案。深入分析Activiti工作流程产生的25张表的数据结构与业务关联，按照一定的逻辑关系进行整合与存储。并基于组合模式，将不同层次的数据元素组合成一个整体的数据结构，减少前端数据处理的复杂度与工作量，例如，将与注册申请相关的申请人信息、申请内容、审核记录等数据组合成一个扁平的对象，前端只需要通过简单的访问方式就可以获取到所需信息，无需再面对复杂的工作流程表结构，同时，为了提高数据扁平化处理的效率，利用多线程技术，将数据提取和整合的任务分配到多个线程中并行处理，加快处理速度。

3.数据库存储优化：在数据扁平化处理后，将处理后的数据存储到数据库中。在存储过程中，考虑数据库的性能优化与数据一致性维护。除了优化设计数据库表结构，选择合适的数据类型与索引策略外，还利用XXL-JOB定时任务定期对数据库进行优化操作。每月固定时间通过XXL-JOB对数据库表进行碎片整理、索引重建等操作，同时通过事务管理机制，保证数据在存储过程中的原子性、一致性、隔离性与持久性，避免因数据操作失误或并发访问导致的数据错误或不一致问题。基于RocketMQ消息队列，将数据库的操作数据信息发送到队列中，然后由监控服务从队列中消费消息并进行记录和分析，若发现异常操作或数据不一致情况，能够及时采取措施进行处理。

4.循环嵌套优化：在代码优化工作中，针对循环嵌套效率低下的问题进行解决，当遇到需要通过一个列表循环匹配另一个列表相同产品编号并获取某个字段的情况时，摒弃了传统的双层循环遍历方式，而是先将内层的列表转换成Map数据结构。这样，在匹配过程中，只需遍历外层列表，通过产品编号作为Map的键直接获取对应的字段值，避免了内层列表的重复遍历，大大提高了代码的执行效率，在处理大规模数据匹配时，利用**RocketMQ消息队列**将数据匹配任务进行异步处理，将需要匹配的数据发送到RocketMQ队列中，然后由专门的处理服务从队列中消费消息并进行匹配操作，这样可以避免在主线程中进行长时间的匹配操作影响系统响应速度，提高了代码在高并发场景下的处理能力。

5.团队BUG协助解决：积极协助团队解决开发过程中遇到的各种BUG。能够迅速介入并进行问题排查。通过仔细分析错误日志、调试代码、检查数据库数据以及验证业务逻辑等多方面的手段，精准定位BUG的根源，并提出有效的解决方案。基于Nacos配置管理中心，当出现BUG时，能够快速查看相关配置信息是否正确，因为很多时候BUG可能是由于配置不当引起的。

学校资产管理平台

Java开发

技术栈：SpringBoot+Activiti+MyBatis-Plus+Nacos+RocketMQ+MySQL+Redis

项目描述：学校资产管理系统是一个高效、可扩展的平台，旨在优化教育机构资产（包括设备、设施和库存）的管理。该系统集成了资产全生命周期管理模块，支持从采购到报废的跟踪，实时更新状态并自动计算折旧。系统采用Activiti工作流引擎驱动的动态工作流，处理资产分配和维护请求的复杂审批流程。采用Spring Boot单体架构，Redis缓存热点数据，Nacos动态配置，利用RocketMQ实现实时通知和跨部门协作，大幅提升了教育机构的运营效率和透明度。

个人职责：

1.项目申报流程开发

- a. 原项目申报流程缺乏校验机制，其中录入错误、重复申报和流程选择错误导致流程失败，使用**Spring Validation**实现后端表单校验防止录入错误。集成Redis存储用户会话和申报标识，检测并拦截重复提交，申报错误率降至3%，录入错误降低80%，显著提升审批准确性和效率。
- b. 原申报流程中，状态变更部门通知耗时长，状态更新延迟太久，使用**RocketMQ**异步发送状态变更通知，保障跨部门实时同步，优化状态变更事务，集成**Redis缓存状态数据**，加速前端查询和状态刷新
- c. 一般学期开始时候集中申报会导致系统响应缓慢，平均处理耗时过长，引入**Redis分布式锁**控制学期初高峰期并发写入，优化MySQL索引减少50%扫描行数，实现批量插入，提升写入效率，优化后，QPS达4000时，响应时间稳定在200ms

2.实时通知系统：开学时候**RocketMQ**高峰期消息堆积，消费延迟激增，通过增加消费者线程数（从8到16），消费能力提升40%，支持1万条/秒消息处理，消除堆积。优化后，1万条/秒消息时，消费延迟降至600ms，通知送达时间稳定在1秒。

3.缓存优化：高频资产查询场景下，数据库压力大，影响管理员和财务人员操作效率，使用**Redis缓存高频资产数据**，但是Redis缓存失效高峰期，缓存重建导致数据库瞬时QPS激增，响应时间慢，实现随机TTL，分散缓存失效时间，降低80%雪崩风险，数据库QPS降至1000，响应时间稳定在200ms

4.团队协作与BUG协助解决：定期参加项目团队会议，及时汇报工作进展与问题，共同讨论解决方案与技术选型，确保项目开发方向的一致性与团队协作的高效性。在BUG排查与修复方面，使用过多种调试工具与技术手段，如浏览器开发工具、后端日志分析、代码调试器等，对项目中出现的BUG进行精准定位与快速修复。在修复BUG后，及时与测试团队沟通，协助其进行回归测试，确保BUG得到彻底解决且不会引入新的问题，保障项目的稳定运行与高质量交付。

教育背景

2020-09~2024-06

荆楚理工学院

计算机科学与技术（本科）

自我评价

一年多Java开发经验，对企业级项目的后端架构和整体业务有深入的理解，熟悉Java核心技术，熟练掌握MySQL、SSM、SpringBoot等知识，在工作中喜欢解决和挑战技术难题，对IT技术有较浓厚的兴趣，喜欢跟踪与钻研新技术。