

# 个人简历

## 基本信息

姓名：王崇骁 性别：男  
年龄：23岁 学历：硕士  
当前状态：在读（研二） 求职意向：后端开发  
电话号码：13940241438 邮箱：1316876064@qq.com



## 教育经历

2023.09 ~ 至今（研二） 东北大学（985） 计算机科学与技术  
GPA：3.58/4.00（专业Top 15%）  
荣誉奖项：多次获得校一等奖学金  
2019.09 ~ 2023.06（本科） 东北大学（985） 人工智能  
荣誉奖项：CET-6，多次获得校二等奖学金

## 专业技能

**Java**：熟悉掌握Java语言，具有良好Java编码习惯。熟悉单例、代理等设计模式。熟悉JVM内存结构和JUC并发编程。  
**数据库**：掌握MySQL数据库和库表设计，熟悉其存储引擎、索引设计的优化方法、事务和日志以及锁机制、MVCC等。了解MongoDB非关系型数据库，掌握其文档存储和常见查询操作。  
**Redis**：熟悉Redis基本使用，持久化和淘汰策略，熟悉缓存高并发场景及解决方案，如缓存穿透、缓存击穿、缓存雪崩等。  
**开发框架**：熟练使用Spring各框架开发，如Spring Framework，Spring MVC，Spring Boot，SpringCloud Alibaba，Spring Security等主流框架以及MyBatis-Plus框架。了解Drools规则引擎，轻量级Excel处理库EasyExcel。  
**中间件**：熟悉RabbitMQ消息队列，XXL-JOB分布式任务调度。了解Sentinel流量防护组件，Minio轻量级对象存储服务，Nginx网页服务器，Nacos配置管理服务平台，Elasticsearch搜索引擎等。  
**基础能力**:熟悉计算机网络，操作系统，数据结构，计算机系统等相关知识。

## 项目经历

2024.03 ~ 2024.07 校园智能抢座系统 后端开发  
**技术栈**：SpringBoot，Redis，MySQL，Redisson，Rabbitmq，EasyExcel，XXL-JOB，MongoDB，Caffeine  
**项目介绍**：  
开发了一个校园智能抢座系统，提供教室、图书馆座位预约服务，并集成活动报名、签到互动等功能。系统支持学生快速预约座位，参与校园活动，旨在优化校园资源分配，提升学生校园生活体验。  
**主要工作**：  
●设计API 密钥和令牌校验机制，增强接口安全性，并通过 滑动窗口算法 实现接口调用限流，防止恶意调用；  
●采用 本地消息表 实现分布式事务，结合本地事务和 MQ 消息事务，解决分布式环境下的数据最终一致性问题；  
●利用 RabbitMQ 实现系统内部消息分发，处理超时订单关闭，解耦同步业务场景，如快速抢订通知，实现削峰填谷；  
●使用 WebSocket 技术 实现服务器端向用户端推送抢座消息，提高服务响应速度和用户满意度；  
●使用 EasyExcel 高效解析上传文件的数据，通过 XXL-JOB 定时任务调度结合自定义线程池 同步座位信息至 Redis 缓存预热，结合 布隆过滤器 拦截无效查询，减轻数据库负载，提升响应速度40%；  
●引入 自定义注解，统一处理因网络延迟导致的用户请求重复发送问题，提升系统稳定性和用户体验；  
●采用 Redisson 分布式锁 实现课程库存的非负化管理，有效避免并发环境下的“超卖”问题，确保库存数据的准确性；  
●集成 MongoDB 存储用户预约历史记录、活动数据信息，提升系统数据存储灵活性和查询效率。  
●采用 Caffeine 作为本地缓存解决方案，并与 Redis相结合，构建了双层缓存架构，单机吞吐量提升5倍，接口响应时间从200ms降至40ms。

2024.07 ~ 2024.12 中学生在线教育平台 后端开发  
**技术栈**：SpringCloud，Spring Security，Redis，Rabbitmq，BloomFilter，Seata，MinIO，Elasticsearch，Drools  
**项目介绍**：  
在线教育平台是一个为学习者和教育机构提供服务的在线学习系统，支持课程发布、在线学习、练习、考试以及学习社群互动。平台采用B2B2C模式，通过技术手段优化教育资源的分配和学习体验。  
**主要工作**：  
●扩展 Spring Security 框架，结合Oauth2协议和JWT，实现了包括普通登录、扫码登录的认证和授权流程；  
●使用 Seata 框架，确保支付功能模块在分布式环境中的事务一致性，协调各系统之间的跨服务操作。  
●基于 HTTP 和 MinIO 实现了图片、视频等媒资文件的断点续传功能，有效避免了因网络问题导致的过度重传；  
●采用 BloomFilter 算法实现了请求过滤机制，有效避免了课程查询等公开接口的缓存穿透问题，增强了系统的稳定性；  
●利用 Redisson 实现了分布式锁，确保在缓存击穿情况下，只有一个线程能够对数据库进行查询，保护了数据的一致性；  
●使用 Redis 缓存课程发布信息，分页数据等信息，优化了接口访问速度从 150ms 降低到 38ms；  
●使用 Nacos 进行服务注册与配置管理，结合 OpenFeign、Hystrix、Sentinel、Nginx 实现模块调用、熔断降级、负载均衡及流量控制；  
●利用 RabbitMQ 的死信队列和 TTL 实现订单的延时处理，自动取消超时订单。结合定时任务作为兜底方案，确保在消息队列异常时订单处理的可靠性；  
●使用 Drools规则引擎 解耦课程费用，如打折和优惠逻辑。为了解决缓存击穿，分布式部署多实例并发，借助 XXL-JOB 分布式定时任务来定期更新缓存数据；  
●利用 Elasticsearch 实现课程内容的聚合搜索，将课程品牌、标题、描述等关键信息聚合到单一字段，整合 RabbitMQ 与 Elasticsearch 实现课程上下架。