

이건

서버/백엔드 개발자 | 01031804727 | doolchong@gmail.com

B 블로그

저는 호기심과 열정으로 개발하는 백엔드 개발자입니다. Java와 Spring 프레임워크를 주로 사용하며, 기술적 문제를 깊이 분석하고 해결하는 것을 즐깁니다. 코드의 성능과 가독성, 유지보수성을 항상 고려하며, 팀원들과의 협업을 통해 더 나은 솔루션을 만들어내는 것을 중요하게 생각합니다. 새로운 기술 습득에 열정적이며, 지속적인 성장을 추구합니다.

경력 사항

더선한

인턴

2025.02~ 2025.03 (1개월)

기업 정보 크롤링 자동화

- 대규모 기업 정보 수집 자동화 시스템 구축 (누적 43만 건)
 - Levenshtein 거리 알고리즘 기반 중복 데이터 통합
 - Spring Batch 기반 주기적 업데이트(주 단위)

2. 안정성 강화 설계

[에러 처리]

- 재시도: 일시적 네트워크 장애
- 건너뛰기: 비정형 데이터 예외
- 중단: 시스템 장애 등 치명적 오류

[복구 기능]

- 중단 시점부터 작업 재개 구현

프로젝트

FitNus

2024.10~ 2024.11 (1개월)



팀프로젝트(BE 5명)

프로젝트 소개

FitNus는 기존에 월 단위로 운동을 결제해야 했던 시스템을 벗어나 일 단위로 원하는 날짜에 원하는 운동을 일정으로 등록하고 예약하는 서비스입니다.

적용 기술

- 프레임워크: Spring Boot, Spring Batch
- 데이터베이스: MySQL, Redis
- 메시지 브로커: SSE, Kafka
- 검색 엔진: Elasticsearch
- 인증/보안: Spring Security, JWT
- 배포 환경: AWS (EC2, RDS, S3, ECR, VPC, ALB, AutoScaling), Docker, GitHub Actions, Terraform
- 외부 API: 카카오페이, 카카오 로그인, 카카오 지도
- 고가용성: Redis Cluster

역할

1. 모임, 멤버, 일정 기능 CRUD
2. 일정 등록 기능 동시성 제어 및 성능 개선
3. 정산 기능 - 대용량 데이터 처리를 위해 스프링 배치 사용
 - 일 단위 센터 이용내역 기록
 - 월 단위 센터별 정산
4. 모놀리식 구조에서 멀티 모듈로 구조를 변경
5. Slack API를 이용한 개발자 알림

트러블슈팅 및 성능개선

1. 모듈별 의존 관계를 재정비

기존 구조에서 'notification' 모듈에서 처리해야 할 Kafka 파티션을 'service' 모듈이 가져가는 문제가 발생

- 'notification'과 'service' 모듈 간의 불필요한 의존성을 제거
- 모든 모듈이 'common' 모듈만을 의존하도록 수정
- 모듈 간의 의존 관계가 단순화되었으며, 유지 보수성과 확장성이 향상됨

2. 일정 등록(예약) 동시성 제어 및 성능 개선

일정 등록(예약) 기능 부하 테스트 진행 중 동시성 문제 발생

- 동시성 제어를 위해 Redis 분산락을 이용했으나, TPS가 108.6/sec로 성능이 떨어짐
- Redis atomic 연산을 활용하여 개선 후 TPS가 522.3/sec로 개선됨
- Redis Lua Script를 활용하여 개선 후 TPS가 550/sec로 개선됨

3. 일 단위 센터 이용내역 기록 성능 개선(10만건 기준)

대용량 데이터 처리를 위해 Spring Batch를 활용하여 기능 구현 → 처리 속도 9m45.6s(기준)

- 멀티스레드 적용 → 처리 속도가 4m40.99s로 기준 대비 52.01% 개선
 - 파티셔닝 적용 → 처리 속도가 4m37.8s로 기준 대비 52.56% 개선
 - 쿼리메소드 saveAll 적용 → 처리 속도가 2m19.2s로 기준 대비 76.22% 개선
-

SAL(업무현황일지)

2024.10~ 2024.10



팀프로젝트(BE 4명)

프로젝트 소개

업무현황일지는 부대 운영 진행 상황과 계획을 실시간으로 공유 및 기록할 수 있는 칸반 보드 형식의 애플리케이션 백엔드 시스템입니다.

적용 기술

- 프레임워크: Spring Boot
- 데이터베이스: MySQL, Redis
- 배포 환경: Docker, AWS(EC2, ECR, S3), GitHub Actions

역할

1. 회원가입/로그인
2. 효율적인 쿼리 작성과 인덱스 설계를 통한 카드 검색 최적화
3. Slack API를 활용한 알림기능
4. 배포와 CI / CD

CI / CD 경험

- 코드 변경 시, 자동으로 빌드 및 테스트를 수행하는 CI 파이프라인을 구성
- 테스트가 성공적으로 완료되면 자동 배포되도록 CD 파이프라인을 설정
 - CI / CD 관련 이슈를 해결하기 위해 다음을 고려함
 1. 배포 전 코드 안정성을 높이기 위해 테스트 커버리지를 강화
 2. 배포 과정에서 민감한 정보나 설정이 노출되지 않도록 깃허브 secrets를 이용하여 환경 변수를 관리

트러블슈팅 및 성능개선

1. 카드 검색 성능 개선
 - 카드 검색 속도를 향상시키기 위해 적절한 인덱스를 설계
 - 카드 검색 조건이 다양하지만 최적의 성능을 내는 2가지 조건에만 인덱스를 설정하여 성능 개선
2. EC2 메모리 부족 문제 해결
 - Swap Memory 설정
 - Docker Container가 사용할 수 있는 최대 메모리 양 제한

대외활동 및 수상

스파르타 내일배움캠프 Spring 6기

2024.07 ~ 2024.11 (4개월)
2024.09.02 ~ 2024.09.06 뉴스피드 프로젝트
2024.09.19 ~ 2024.09.25 아웃소싱 프로젝트
2024.10.14 ~ 2024.10.18 심화 프로젝트
2024.10.21 ~ 2024.11.22 최종 프로젝트
내일배움캠프 Spring 6기 모범상

학력 사항

세종대학교

컴퓨터공학과 학사
2017.03 ~ 2024.02(졸업)

2017년 교내 제4회 SW경시대회(세종코딩챌린지워크) 장려상
2023년 제16회 창의설계경진대회 장려상