

임민석

email : ddalgu2@gmail.com

github : <https://github.com/minseoklim>

blog : <https://mslim8803.tistory.com>

자기 소개

오랜 방황 끝에 천직을 찾은 늦깎이 개발자입니다. 주로 Spring + Java / Kotlin 을 이용한 개발을 해왔습니다. 항상 타인의 입장에서 먼저 생각하고 행동하는 개발자이며, 꾸준하고 지속 가능한 개발자로서의 삶을 살아가기 위해 노력하고 있습니다.

보유 기술

Backend	Java, Kotlin, Spring Boot, Spring MVC, JPA, Hibernate, Kafka, Spring Cloud, Spring Batch
Database	MySQL, MariaDB, Oracle, Redis
Devops	AWS (Elastic Beanstalk, ECS, RDS, ElastiCache, Lambda, S3, Cloud Watch 등), Sentry, CentOS, Ubuntu, Nginx, Apache
Frontend	Javascript, HTML, CSS
Communication & VCS	Jira, Confluence, Slack, Git

경력 사항

야놀자

2022.05.16 ~ 2024.01.19 (1년 8개월)

- 기술 스택 : Kotlin, Java, Spring Boot, JOOQ, JPA, Kafka, MySQL, Redis, AWS
- 주 업무 : 회원 프로젝트 백엔드 개발
 - 회원 프로젝트 유지 보수
 - 숙박세일 페스타, BTS 프로모션 등의 대용량 트래픽이 유입되는 이벤트에 대한 대응
 - 선불전자지급수단 프로젝트
 - 선불망 내 별도의 회원 서버 구축
 - PLCC(Private Label Credit Card) 프로젝트
 - 카드 발급/해지 및 실적 달성 여부 등을 회원 정보와 연동
 - Kafka Consumer 구현을 통해 관련 정보를 마케팅 톨과 연동

- 서비스 : 맵스플렛(수학 문제 은행)
- 기술 스택 : Kotlin, Spring Boot, JPA, MySQL, Redis, AWS
- 주 업무 : 백엔드 개발
 - 맵스플렛 2.0 프로젝트
 - S3 Life Cycle 설정을 위한 다운로드 로직 리팩토링([소스 코드 참고](#))
 - 문제 상황 : 오래된 문제지를 학원에서 다운로드 받아 사용할 확률은 낮음. 하지만 모든 파일을 보관하고 있어 높은 S3 비용 발생
 - 솔루션 : S3 Life Cycle을 지정하여 일정 기간이 지난 파일 자동 삭제. 대신 삭제된 파일들에 대한 다운로드 시도 시 파일 재생성
 - 결과 : AWS 비용 절감 기대
 - 1.0 to 2.0 데이터 마이그레이션
 - 문제 상황 : 테이블 구조가 완전히 바뀜. 기존 데이터가 많음(가장 많은 테이블의 경우 2억건 이상). 실시간 단방향 동기화 필요.
 - 솔루션
 - 1) 테이블명, action의 종류(INSERT, UPDATE, DELETE), target_id, created_time 을 컬럼으로 가지는 trigger_table을 생성
 - 2) 1.0의 모든 테이블에 INSERT, UPDATE, DELETE 액션에 대한 트리거를 생성하여 trigger_table에 적절한 데이터를 인서트
 - 테이블, target_id 별 레코드들 중 가장 마지막에 일어난 이벤트만 적용해 주면 됨
 - 3) 1.0과 2.0이 서로 다른 스키마를 가지고, 같은 DB 인스턴스를 공유. 쿼리를 통해 전체 이관
 - 4) 전체 이관 시간 동안 들어진 데이터를 trigger_table에 쌓인 데이터를 기반으로 동기화
 - 5) Spring batch를 이용해 매일 새벽 trigger_table에 쌓인 데이터를 기반으로 동기화
 - 결과 : 일 단위 동기화에 성공
 - 1.0 to 2.0 데이터 마이그레이션 - 전체 이관
 - 문제 상황 : 전체 이관 시뮬레이션 결과 11일이 소요. 같은 성격의 컬럼이 서로 다른 타입(int, varchar)을 가짐
 - 솔루션 : 레거시 DB 에서 해당 컬럼의 타입이 varchar로 되어있는 또 하나의 테이블을 발견. 이 테이블과 조인하도록 쿼리 변경
 - 결과 : 인덱스를 타게 되면서 소요 시간 이틀로 줄어듦
 - 레거시 시스템(맵스플렛 1.0) 유지 보수
 - 장애 모니터링을 위한 Health Checker 구현
 - 문제 상황 : 장애가 발생하였으나 1시간이 넘도록 아무도 알아차리지 못함.
 - 솔루션 : 모든 레거시 시스템에 대한 health check를 주기적으로 하는 시스템 구현
 - 결과 : 장애 발생 시, 웹 혹은 통해 슬랙 채널로 장애 발생 메시지가 전달
 - PHP 서버 유지 보수 및 일부 기능 추가

가온아이

2018.04.12 ~ 2020.10.15 (2년 6개월)

- 서비스 : 기업용 그룹웨어 솔루션
- 기술 스택 : Java, JSP, Spring MVC, MariaDB/Oracle, Javascript
- 주 업무 : 풀스택 개발
 - 표준 솔루션(마스터 브랜치) 기능 추가, 개선 및 버그 수정
 - 고객사별 요구사항에 따른 커스터마이징 개발
 - 개발 PL
 - 한국경제신문 그룹웨어 구축 (2019.07 ~ 2019.12, PM 1명, 개발 1명 + α(본사 지원))
 - 투썸플레이스 그룹웨어 구축 (2019.12 ~ 2020.05, PM 1명, 개발 3명 + α(본사 지원))
 - PM과 함께 고객의 요구사항을 분석하고 구현 방식을 설계
 - 메인 커스터마이징 기능 구현 (예: 전자결재 연동, 인사시스템 연동)
 - 고객사 상주 SM
 - 투썸플레이스 그룹웨어 유지 보수 (2020.05 ~ 2020.10, 사용자 약 1000명)
 - docker를 통한 다운 타임 최소화
 - 문제 상황 : 기존 DRM의 Key 값 변경과 버전 업이 필요 → 많은 추가적인 설정이 필요
→ 다운 타임 발생 가능성
 - 솔루션 : 모든 설정이 완료되고, 정상 동작이 확인된 docker 이미지를 생성
 - 결과 : 다운 타임을 0에 가깝게 만듦
 - 메세지 프로퍼티 테스트 코드 추가([소스 코드 참고](#))
 - 문제 상황 : 다국어(한국어, 영어, 중국어)를 지원. 일부 언어들의 메세지 프로퍼티들이 누락되어 버그가 지속적으로 발생
 - 솔루션 : 투포인터 알고리즘을 활용한 테스트 코드를 작성하여, 프로퍼티 누락 시 아예 컴파일이 안되도록 설정
 - 결과 : 메세지 프로퍼티 누락으로 인한 버그 발생 방지. 고객의 서비스 질 향상. 불필요한 공수 낭비 방지

[비개발] LG전자(구 V-ENS)

2012.07.01 ~ 2014.03.14 (1년 9개월)

- LG CNS의 자회사였던 V-ENS가 2013년 7월, LG전자로 합병되었습니다. 저는 여기서 생산기술 업무를 담당하였습니다. 생산기술이란 각 공정을 설계하고 그 공정에 들어가는 설비들을 검토하고 도입하고 관리하는 업무를 말합니다.

학력

연세대학교 천문우주학과 졸업

2004.03 ~ 2012.08

용문고등학교 졸업

2001.03 ~ 2004.02

교육

우아한테크캠프 Pro 3기

2021.11.01 ~ 2021.12.31

- Java 백엔드 경력 개발자(만 3년 이상)들을 대상으로 하는 강의입니다. 유지 보수하기 좋은 클린 코드, JPA, TDD, ATDD 기반으로 애플리케이션을 개발하고, 레거시 코드를 안정적으로 리팩터링하는 방법을 배웠습니다. 또한 AWS 인프라를 직접 구축하고, 서비스의 문제를 진단하여 최적화하는 방법에 대해서도 학습하였습니다.

웹개발 교육 (국비 과정)

2017.10.30 ~ 2018.03.30

- Java, JSP, Spring Framework, MySQL, Oracle, Javascript, HTML 등 웹개발에 필요한 기본 지식들을 학습하였습니다.

자격증

정보처리기사

2017.11.24

OCJP

2017.09.25