



김하린

Backend Engineer

Email harin.bird@gmail.com

Mobile 010-9079-5023

Github <https://github.com/kimhalin>

Blog <https://velog.io/@kimhalin>

Summary

인턴 근무를 하며 PG사 연동 결제 시스템과 약 50,000건의 동시 처리 목표를 가진 푸시 알림 시스템 개발 경험이 있습니다.

Spring Boot, NestJS, FastAPI를 이용해 서버를 구축할 수 있으며, **클라우드에 서비스를 배포**할 수 있습니다.

프로젝트 기획부터 개발, 배포까지의 과정을 겪는 **End-to-End 프로젝트 경험**이 있습니다. 이를 통해 개발의 전 과정을 이해했으며, 또한 프로젝트 리드 경험을 통해 좋은 커뮤니케이션의 중요성을 알고 노력하는 개발자입니다.

Skill

Language	Framework	Database	Queue	Etc
- Java - Python - TypeScript	- Spring Boot - NestJS - FastAPI - React	- MySQL - PostgreSQL - Redis	- RabbitMQ - BullMQ	- AWS EC2 - AWS S3 - Github Actions - Prometheus - Grafana

Career

- Entropy Paradox 인턴 | 2023.05 ~ 2023.08
 - 백엔드 개발자로 2개 프로젝트 참여

Projects

Entropy Paradox(인턴) | 서로의 일정을 거래 및 구독할 수 있는 서비스, TimesCare
2023.06 ~ 2023.08

기술 스택	NestJS, TypeORM, MySQL, Redis, AWS EC2, S3, RDS, CloudFront, Github Actions
개발 내용	<p>[NestJS와 TypeORM을 활용한 API 서버 구축]</p> <ul style="list-style-type: none">PG사 연동을 통한 인증 결제를 구현해 다양한 결제 니즈를 충족시키는 결제 시스템 구축직접 데이터를 삽입해 진행하던 복잡한 로직 테스트에 소모되는 비용을 줄이고자 Jest, Mockito를 활용한 테스트 코드를 작성해 기능 구현에만 집중할 수 있는 개발 환경 조성 <p>[결제 내역 자동 구매 확정을 위한 Batch 작업 구현]</p>

	<ul style="list-style-type: none"> • 별도 인스턴스를 통한 Batch 서버 배포로, API 서버와 분리함으로써 추후 Scale-out이 용이하고 가용성이 높은 인프라 구축 • 'where - In' 을 이용한 bulk update 방식을 사용해 기존의 update 방식으로 발생하던 레코드 수에 비례한 DB I/O 수를 최적화하여 약 80%의 성능 향상 달성 <p>[Redis Queue(BullMQ)를 이용한 푸시 알림 발송]</p> <ul style="list-style-type: none"> • 알림 발송 과정에서 대규모 알림 요청의 무한정 대기 상황을 예방하기 위해 Redis Queue(BullMQ)를 사용한 메시지 기반 푸시 알림 비동기 발송으로 서버 부담 최소화 • NHN Cloud 기반 SMS/이메일/알림톡 및 OneSignal 기반 푸시 알림 구현
--	---

개발한 프로젝트 기록과 공유 및 팀 모집 지원 서비스, Graphy 2023.03 ~

기술 스택	Java, Spring Boot, JPA, MySQL, Redis, Nginx, AWS EC2, Github Actions
개발 내용	<p>[토큰 방식을 이용한 로그인 및 권한 기능]</p> <ul style="list-style-type: none"> • 확장이 가능한 Stateless 서버를 구축하기 위해 Access Token과 Refresh Token 방식 로그인 및 권한 기능 구현 • Refresh Token 탈취를 방지하기 위해 Redis에 Refresh Token을 저장해 기본적으로 제공하는 TTL과 빠른 액세스 속도를 이용할 수 있는 <p>[STOMP와 RabbitMQ를 이용한 채팅 기능 구현]</p> <ul style="list-style-type: none"> • 일회성 채팅을 제공했을 때의 이전 기록을 확인할 수 없는 불편함을 개선하기 위해 Pub URL로 수신하는 채팅 데이터는 DB에 저장하여 이전 채팅을 확인할 수 있도록 구현 • STOMP 인메모리 브로커 사용 시, 인스턴스 개수가 한 개로 한정되는 문제를 해결하기 위해 외부 브로커 RabbitMQ를 사용하여 Scalable한 구조로 개선 <p>[Spring Boot 기반 API 서버 개발]</p> <ul style="list-style-type: none"> • JPA만을 이용한 쿼리 작성의 복잡함과 N+1 문제를 해소하고자 QueryDSL을 사용해 높아진 가독성과 컴파일 시 발견 가능한 오류를 활용한 코드 생산성 확보 • 요구사항에 따른 올바른 코드임에 대한 판단을 위해 Junit과 Mockito를 이용해 단위 및 통합 테스트를 작성 • 실제 프로덕션 환경과 유사한 환경 안에서, 역등성 있는 테스트 환경을 유지하기 위해 TestContainers를 이용한 통합 테스트 및 DB 테스트 환경 구축 <p>[Github Actions를 이용한 CI 파이프라인 구축]</p> <ul style="list-style-type: none"> • PR 생성 시에 Jacoco로 측정한 Test Coverage와 테스트 코드 실행 결과를 검증하는 CI를 구축해 안정적인 코드 상태 유지
링크	https://github.com/techeer-sv/graphy

사용자가 그린 그림을 AI가 예측하는 서비스, Doodle Doodle

2022.06 ~ 2022.08

기술 스택	Flask, React, TailwindCSS, Nginx, RabbitMQ, AWS Lightsail, Prometheus, Grafana, AWS S3
개발 내용	<p>[Flask와 SQLAlchemy를 이용한 API 및 AI 서버 구축]</p> <ul style="list-style-type: none">● 사용자가 그린 그림의 AI 예측 결과를 반환하는데 오랜 시간이 소요되는 것을 발견해, AI로 인한 메인 서버의 과부하를 방지하기 위해 AI 서버를 분리하여 구축● AI 분석 요청 API의 긴 응답 시간으로, 사용자의 불편을 예측해 RabbitMQ와 Celery를 활용해 많은 요청을 비동기적으로 분산 처리하여 문제 해결
링크	https://github.com/ALLBACK-2022/DoodleDoodle

Activity

- Techeer [테커] - 실리콘밸리 개발자의 SW 개발자 커리어 그룹 | 2022.09 ~ 현재
- Wanted 프리온보딩 인턴십 | 2023.08 ~ 2023.09
 - 백엔드 아키텍처 설계, 자료 구조 및 알고리즘 학습
- Techeer 실리콘밸리 SW 부트캠프 세션단 | 2022.11 ~ 2023.07
 - DB 및 REST API를 포함한 백엔드 기초 세션 진행
 - 서버 부하 테스트 세션 진행
- Techeer Partners 운영진 - Techeer 지원 스터디 그룹 | 2022.11 ~ 2023.07

Education

- 한국공학대학교 컴퓨터공학부 소프트웨어전공 | 2020.03 ~ 2024.02 (졸업예정)