**최세연** 

**Backend Engineer**

**Summary**

| Email | choi20014830@gmail.com  Github | https://github.com/barabobBOB Phone | 010-4830-6738  Blog | https://velog.io/@chltpdus48 |
| --- |

**Spring Boot, BigQuery, MySQL**를 활용해 IoT 장비에서 데이터를 실시간으로 수집 및 모니터링을 제공하는 서비스 개발한 경험이 있습니다. **GCP, Docker, Gitlab CI/CD**를 활용해 **배포 파이프라인**을 구축 할 수 있습니다. 쿼리 성능 개선에 관심이 많으며, **MySQL, PostgresSQL**에서 Explain를 이용해 쿼리를 분석하고 **쿼리튜닝**을 진행해 성능 개선한 경험이 다수 있습니다.

**Django**, **Flask**를 활용한 **Python 기반** End-to-End 프로젝트 경험과 **Celery**와 **RabbitMQ**를 활용한 **AI 모델 서빙** 경험이 있습니다.

**Experience**

**유니로보틱스 | Server Engineer, 인턴** (2022.3 ~ 2022.7)

IoT 센서 디바이스에서 다양한 종류의 데이터를 수집하고 실시간 장비 제어 및 모니터링을 제공하는 서비스 프로젝트를 진행하였습니다.

사용 기술: Spring Boot, Spring Data JPA, QueryDSL, JUnit5, Mockito, MySQL, GCP [Compute Engine, BigQuery], Gitlab CI/CD, Jacoco, Docker, Prometheus, Grafana

문제해결

● IoT 모니터링이 **딜레이**되는 문제점 발견하였습니다.

○ 권한, 조건에 따라 기기 번호를 조회하는 API 로직 중에 **MySQL의 잦은 Join 연산**과 **N+1 이슈로 인한 Slow Query 문제**가 발생함을 알게 되었습니다.

■ **비정규화**를 진행하여 JOIN 연산 횟수를 줄이도록 하였습니다. 그 후, **QueryDSL**을 활용하여 조건별 동적 쿼리를 구현함으로써 **N+1 쿼리 문제를 해결**했습니다. 그 결과, 해당 API의 **Latency 2s** ➞ **600ms** 개선하였습니다.

○ 기존 BigQuery의 IoT 센서 데이터 테이블 **파티션 정책이 월간**인 것을 알게 되었습니다.

■ **주간 파티션**으로 변경하면 조회할 전체 row의 수가 줄어들어 쿼리 성능이 향상될 것이라고 판단했고 적용 결과, **600ms** ➞ **200ms**으로 개선하였습니다.

기여

● **IoT 센서 데이터 시각화 API** 개발 및 성능 개선을 하였습니다.

● BigQuery 기반 **IoT 센서 디바이스 데이터 수집 파이프라인**을 개발하였습니다.

● **GitLab CI/CD**와 **Docker**을 활용해 브랜치의 변경 사항이 해당 서버에 반영되는 **CI/CD 파이프라인**을 구축하였습니다.

● **JUnit5, Mockito, Jacoco**를 이용한 **Unit Test**를 작성하였습니다.

**Project**

**TripDoodle | 개인 프로젝트 Github** (2023.7 ~ 현재) 요약: 위치 기반 낙서(게시글) 및 채팅 서비스로, 관광지에 방문 중일 때 이용할 수 있는 온라인 컨텐츠입니다. 사용 기술: Go[Fiber, Websocket, Goroutine, Channel], PostgreSQL[Postgis], Docker, Redis, TypeScript

● 현재 위치를 기준으로 **반경 내 관광지 안내 REST API** 개발 및 성능 개선을 하였습니다. | 관련 블로그

○ **PostGIS**를 사용해 반경 내 관광지 조회 쿼리를 작성했지만, **Slow Query 문제**가 발생했습니다. 주로 읽기 연산이므로 **인덱스** 적용으로 문제를 해결할 수 있을 것으로 예상했습니다.

○ GiST 방식의 **공간 인덱스** 활용해 **쿼리튜닝**을 진행하였고 그 결과, **1.8s** ➞ **123ms**로 조회 성능을 개선하였습니다.

● **WebSocket**, **Goroutine**, **Channel**, **Redis Pub/Sub** 이용해 실시간 다중 채팅과 메시지 브로드캐스팅 구현하였습니다.

● **Testify**를 활용해 Post, Sight 도메인 **UnitTest** 작성하였습니다.

**Lego2me | Backend Engineer, 팀 리더 Github** (21년 12월 - 22년 1월) 요약: **2022 Silicon Valley Internship Program**에서 진행한 프로젝트로, 전신 사진의 옷 색을 분석하여 이를 바탕으로 LEGO 캐릭터를 제공하는 서비스를 **End-to-End**로 개발하였습니다.

사용 기술: Django, RabbitMQ, Celery, GCP[BigQuery, Google Cloud Storage], Nginx, Docker ● **Python**, **RabbitMQ**, **Celery**를 활용해 **이미지 분석 파이프라인** 구현하였습니다.

○ 운영 서버와 이미지 분석 서버를 별도로 분리하여 **시스템의 가용성**과 **확장성**을 높였습니다. ● Django, BigQuery와 Google Cloud Storage를 활용해 **사용자 피드백 REST API** 구현하였습니다. ● **GCP Compute Engine**, **Docker**을 이용한 서버 배포 및 **Nginx**를 이용한 **포트포워딩**하였습니다.

**Skills**

****

| **Language** : Go, Java, Python, TypeScript **Server** : Spring Boot, Django, Fiber, Flask |
| --- |
| **Database** : MySQL, BigQuery, PostgreSQL, MongoDB **MQ** : RabbitMQ, Redis |
| **Scheduling :** Airflow, Celery **Monitoring** : Prometheus, Grafana, K6 |
| **Client** : React, HTML, CSS **ETC** : WebSocket |
| **Testing** : JUnit5, Mockito, Testify(Go Testing) **Infra**: Nginx, Github Actions, Gitlab CI/CD, AWS, GCP |

**Activity**

**성남청년 프리인턴십 / 멘토** (2022.9 ~ 2022.10)

● Git, 데이터베이스 세션 진행 및 팀별 발생한 트러블 슈팅 해결책 제시, 면담 등 다양한 피드백 제공 **Techeer / 1기** (2021.9 ~ 현재)

● 실리콘밸리 엔지니어의 SW 개발자 커리어 그룹

● 고성과자로써 미국 실리콘밸리 한달 살기 프로그램 참여

● 스터디 활동

○ 오브젝트 (객체지향) | 인턴 근무 중, 객체지향의 이해도가 떨어진다 판단하여 참여하게 되었습니다. ○ Real MySQL | 인턴 경험 후, DB 성능의 중요성을 깨닫고 스터디를 열게 되었습니다.

**Education**

****성결대학교 - 컴퓨터공학전공 (2020.03 ~ 재학 중(2024.2 졸업예정))