백한결

**Backend Engineer** 소개

| Phone | 010-2559-4993 |
| --- | --- |
| Email | hgbaek1128@gmail.com |
| Github | https://github.com/baekhangyeol |
| Blog | https://medium.com/@hgbaek1128 |

스타트업과의 연계를 통해, 해당 기업의 서비스 **MVP** 앱을 제작한 경험이 있습니다.

**Django, Flask, Spring Boot**를 활용하여 서버를 구축할 수 있으며, **Docker**를 이용한 컨테이너 빌드와 **AWS EC2**를 이용한 클라우드 서버를 배포할 수 있습니다.

**End-to-End** 프로젝트의 리더로서 주도적으로 개발한 경험이 있고, 적극적인 커뮤니케이션을 통해 문제를 해결해나갔습니다. 클린코드에 관심이 많으며, 유지보수가 용이하고 확장 가능한 코드를 작성하는 것을 목표로 합니다.

프로젝트

**Fit A Pet -** 반려동물 혈당 데이터 기반 맞춤 사료 추천 서비스 **(23.06 ~ 23.08)**

| **Github : https://github.com/2023-Summer-Bootcamp-Team-K**  **Medium : https://medium.com/@ljh01051826177/fit-a-pet-ba7eb469753c** |
| --- |

기술스택 **: Django, Python, Selenium, AWS EC2, Docker, Nginx, Github Actions, Prometheus, Grafana**

**●** 혈당 데이터의 관리가 용이하도록 **Selenium**과 크롤링을 통해 데이터셋 구성 자동화

○ 반려동물의 혈당 정보가 저장되는 LibreView 사이트에 접속 ➡️ 반려동물 주인의 이름과 생년월일로 검색 ➡️ 해당하는 반려동물의 혈당 데이터 CSV 파일 다운로드 ➡️ 데이터베이스에 삽입

○ 주기적으로 데이터를 업데이트하기 위해, 위 과정이 4시간에 한 번 씩 수행되도록 스케줄러 구현

**● Github Actions**를 이용한 **CI/CD** 파이프라인 구축

● Django 기반 **API** 서버 개발

○ **Django ORM**과 함께 **REST framework**를 활용하여, RESTful 원칙을 준수하는 API 설계 및 구현 ○ **DRF**의 **serializer**와 **validator**를 이용한 데이터 유호성 검사 로직 구현

○ **Redis**를 활용한 API 캐싱으로 응답 시간 단축 및 서버 부하 감소

● **Docker**를 사용한 개발 및 배포 환경 구축 후 **AWS EC2**를 활용한 배포

● **Prometheus**로 metric 및 컨테이너 정보 수집, **Grafana**로 Docker 내부 데이터 및 Node-exporter 데이터 흐름 시각화

**Graphy -** 프로젝트 공유 플랫폼 **(23.02 ~ 23.10)**

| **Github : https://github.com/techeer-sv/graphy**  **Medium: https://medium.com/@hgbaek1128/graphy-poc-%EC%9D%B4%ED%9B%84-2dae835d6c66** |
| --- |

기술스택**: SpringBoot2.7, Java11, QueryDSL, Junit, Python, Jwt, Redis, Python, Selenium**

**●** 토큰 기반 인증인가 기능 구현

○ SpringSecutiry와 JWT를 통합하여, stateless 서비스 인증을 가능하게 함으로써 서버의 확장성 및 유지보수성 향상 ○ RefreshToken의 안정적인 관리를 위해 Redis를 이용하여 TTL을 적용해 자동 만료가 가능한 인증 토큰 시스템 구현 **● SpringBoot** 기반 **API** 서버 개발

**○** GlobalExceptionHandler를 통해 예외 관리를 중앙 집중화하고 통일성 있는 에러 응답체계를 구축 ○ API의 복잡성이 높아짐에따라, 신뢰할 수 있는 코드를 작성하기 위해 **Junit**과 **Mockito**를 사용하여 단위 테스트를 작성 ○ N+1 문제를 해결하기 위해 QueryDSL을 사용하여 쿼리 작성의 유연성을 더하고, TypeSafe한 쿼리 코드를 통해 컴파일 시점에서 오류 탐지 및 수정

● API의 복잡성이 높아짐에따라, 신뢰할 수 있는 코드를 작성하기 위해 **Junit**과 **Mockito**를 사용하여 단위 테스트를 작성 ● Python과 Selenium을 이용한 채용공고 크롤링 서버 구축

○ 스케줄러를 이용해 3일에 한 번씩 채용공고를 크롤링하도록 설정

○ 중복데이터는 DB의 데이터와 비교 후 같은 값이면 저장하지 않도록 구현

**Buyself -** 무인 상품인식 계산대 **(22.12 ~ 23.02)**

| **Github : https://github.com/2023-Winter-Bootcamp-TeamH**  **Medium: https://medium.com/@rabbit\_72/self-checkout-service-with-ai-f127dee34ee3** |
| --- |

기술스택**: Flask, Python, Redis, Ngnix, AWS EC2, Celery, RabbitMQ, Docker**

● AI의 처리 시간 증가로인한 사용자 경험이 방해되지 않도록 **RabbitMQ**와 **Celery**를 통해 AI 결과를 비동기적으로 처리 ● 사용자가 많이 호출하는 API에 의해 서버 과부하가 발생했고, 서버 부하와 응답시간을 줄이기 위해 Redis를 이용하여 **Caching ○** 상품 목록 조회 API를 캐시로 하여 호출 속도를 **2.97s** ➡️ **5ms**로 약 594배 단축

● **Flask** 기반 **API** 서버 개발

○ Flask SQLAlchemy를 통한 ORM 구현으로 코드 재사용성과 데이터 무결성 강화

○ Gunicorn과 Nginx를 조합한 WSGI 서버 배포로 어플리케이션 동시성과 성능 향상

○ Kakao에서 제공하는 API를 이용한 KakaoPay 결제 기능

기술

| ● **Language** : Python, Java11, Dart  ● **DataBase** : MySQL, PostgreSQL  ● **Monitoring** : Prometheus, Grafana  ● **Client** : Flutter | ● **Server** : SpringBoot 2,8, Django, Flask ● **Cloud** : Docker, AWS(EC2, S3, RDS)  ● **etc**: Nginx, Github Actions, Selenium |
| --- | --- |

경험

● 2023.01 - 2023.02 | [티타임즈 x 테커] Silicon Valley 부트캠프 수료

● 2023.02 - 현재 | 실리콘 밸리 개발자 멘토링 그룹 테커 멤버

● 2023.07 - 2023.08 | [티타임즈 x 테커] Silicon Valley 부트캠프 수료

● 2023.08 - 현재 | Start-up 테커 파트너스 3기, 4기 멘토 / 운영진

● 2024.01 - 2024.02 | [티타임즈 x 테커] Silicon Valley 부트캠프 멘토 / 운영진

학력

● 2020.03 - 2026.02 | 한국공학대학교 컴퓨터공학부 소프트웨어전공 (졸업 예정)