**용길한(Jaden)**

P.H.: (+82) 010-2900-4874

EMail: [rlfgks97@gmail.com](mailto:rlfgks97@gmail.com)

Github: https://github.com/Yong-ga-ri

# **요약**문서의 섹션을 나누는 길고 얇은 직사각형

간소한 창업 프로젝트와 부트캠프 교육프로그램에 참여하여 End to End로 개발 프로세스에 대한 전반적인 구조적인 이해를 도모했습니다. 특히 Python을 기반으로 한 Django와 Flask 프레임워크를 이용하여 AWS와 Docker를 활용한 웹 배포 경험이 다수 있으며, 다양한 SQL/NoSQL의 DB와, Message Queue, Monitoring Tools을 활용하여 프로젝트의 완성도를 높이고자 노력했습니다.

평소 새로운 언어를 배우는 데에 관심이 많고, 토이프로젝트를 통한 개발 프로세스를 이해하고자 노력하며, 유용한 신기술을 접하는 데에 흥미가 있어 프로젝트에 어떻게 적용하는가에 대한 고민을 하곤 합니다.

# **기술**문서의 섹션을 나누는 길고 얇은 직사각형

Main Languages: Python, Java

Sub Languages: Java, JavaScript, SQL etc

Development: Django, Flask, Docker, JWT, RabbitMQ, Elasticsearch, Redis, Prometheus

* ORM, MySQL, PostgreSQL, SQL Server, MongoDB, Firebase
* AWS(S3, EC2, RDS, LightSail)
* Grafana, node-exporter, cadvisor

# **경력**문서의 섹션을 나누는 길고 얇은 직사각형

# 2022. 03 - 2022.08

## UniRobotics, langur *Position: DevOps & Backend*

요약: 자체 제작 IoT 데이터 수집 및 모니터링 서비스

Git-Lab

사용 기술: Spring-Boot, JPA, BigQuery, MariaDB, GCP, Swagger, React, Docker

* 기존 Git 전략에 따른 협업 문제를 Wip, Develop, Master 브랜치를 전략을 도입하여 해결
* 스크럼 도입 및 Jira를 활용한 협업 체계 구축
* DB 데이터 휘발성을 고려하여 GCP VM을 구축하여 DB를 클라우드로 관리
* Spring Boot를 활용한 DDD 설계로 Layer를 나누고 도메인을 구분하여, 응집도를 높이고 결합도를 낮춰 서비스 수정과 확정에 용이한 설계를 구축
* MVC 패턴을 통한 Spring Boot 개발 환경 구축
* 관련된 객체에 대한 쿼리를 실행했을 때, 발생하는 N+1 문제 해결
* Spring Security를 도입하여 Token에 포함된 Authorization을 통한 접근 URI 계층화
* OpenAPI를 도입하여 생성된 API에 대한 문서화
* Error를 Global Handling으로 통합적으로 유지보수가 가능하게 함

# 2021. 07 - 2021.11

## LoopUS *Position: DevOps & Backend*

요약: 창업프로젝트로 학생과 기업의 커뮤니케이션을 통해 취직/채용을 쉽게 하는 커뮤니티 어플리케이션

Github: https://github.com/LoopUsProject/LoopUs\_Backend

사용 기술: Django, JWT, MySQL, S3 bucket, Nginx, Docker, Lightsail

* JWT token, 쿠키, 해쉬함수를 바탕으로 사용자의 보안을 위한 로그인 API 구현
* Django Admin을 활용한 관리자 페이지 구현
* Docker container를 통해 서버를 분리하여 서비스 중단 및 장애 대처를 용이하게 하고 개발 환경을 동일시 함.

# **프로젝트**문서의 섹션을 나누는 길고 얇은 직사각형

# 2022. 01 - 2022.02

## Silicon Valley Winter pre-Internship *Position: DevOps & Backend*

요약: 이미지와 키워드를 기반으로 한 이미지 및 텍스트 검색 엔진

Github: https://github.com/SeongNam-Preinternship-TeamF/Flower

사용 기술: Flask, MongoDB, S3 bucket, Swagger, Elasticsearch, Prometheus, Grafana, redis, Nginx, Docker, Lightsail

* Python의 Flask 프레임워크를 활용하여 특정 단어 및 꽃의 이미지를 입력하면 특정 꽃의 정보를 반환하는 API를 restful하게 설계
* 이미지를 입력받아 AWS S3 버킷과 MongoDB에 저장하여 사용자가 정보를 획득할 수 있는 검색 API 구현
* MongoDB에 저장된 데이터를 Bulk로 Elasticsearch에 인덱스 형식으로 저장
* Elasticsearch를 이용하여 인덱스 조회를 통해 검색 효율을 늘렸으며, nori분석기를 통해 한글의 형태소 분석 구현
* API 서버와 AI 서버를 분리하여 micro service 단위로 각 작업을 독립적으로 운영하여 유지/보수가 용이하고 추후 대규모 서비스를 위한 Kubernetes로의 확장성이 용이해짐
* Docker container를 통해 서버를 분리하여 서비스 중단 및 장애 대처를 용이하게 하고 개발 환경을 동일시 함
* Prometheus로 각 API를 매핑하여 사용자가 각 API를 얼마나 이용하는지 counting하고 Grafana를 통해 시각화 함
* 로드 밸런싱과 reverse proxy를 통한 IP 보호를 위해 Nginx 사용

# 2021. 06 - 2021.07

## Silicon Valley Summer pre-Internship *Position: DevOps & Backend*

요약: 특정 영상에 대하여 사용자에게 키워드와 등장 인물을 입력받아 인물이 언제 나오는지, 특정 키워드를 언제 언급하는지 알려주는 검색엔진

Github: https://github.com/CrayonTeamJ/docker

사용 기술: Flask, JWT, PostgreSQL, MongoDB, S3 bucket, RabbitMQ, Celery, Elasticsearch, Gunicorn, Nginx, Docker, Lightsail

* Python의 Flask 프레임워크를 활용하여 특정 단어 및 인물의 이미지의 등장 시간을 반환하는 API를 restful하게 설계
* JWT token, 쿠키, 해쉬함수를 바탕으로 사용자의 보안을 위한 로그인 API 구현
* 영상을 초 단위로 잘라 이미지와 시간을 AWS S3 버킷과 MongoDB에 저장하여 사용자가 정보를 획득할 수 있는 검색 API 구현
* AI 조회 API의 성능을 올리기 위한 비동기 처리로 RabbitMQ와 Celery를 이용한 Queuing System 구현
* 인덱스 조회를 통해 검색 효율을 늘리기 위한 Elasticsearch 사용
* gunicorn의 multi-processing을 통해 다중 접근에 대한 부하 감소와 비동기 작업의 처리 가능
* API 서버와 AI 서버를 분리하여 micro service 단위로 각 작업을 독립적으로 운영하여 유지/보수가 용이하고 추후 대규모 서비스를 위한 Kubernetes로의 확장성에서 유리하게 함
* Docker container를 통해 서버를 분리하여 서비스 중단 및 장애 대처를 용이하게 하고 개발 환경을 동일시

# **학력**문서의 섹션을 나누는 길고 얇은 직사각형

## 인천대학교 송도캠퍼스

2017년 03월 - 2023 08월

인천대학교 산업인공지능 연구실 2021.02 - 2021.12

# **수상 내역**문서의 섹션을 나누는 길고 얇은 직사각형

## 스마트팩토리 머신스케쥴링 - 2nd

요약: 지속적인 demands 공급을 병렬 머신에 할당하는 데에 최소한의 cost를 만드는 해커톤 대회

Site: <http://ideaboom.net/page/competition_main.php?seq=115>

* 앙상블 기법과 post processing을 통한 문제 해결