김가을 

Backend Engineer

**Summary**

**Phone** | 010-7196-1788

**Email** | gaeul.kim812@gmail.com **Velog** | https://velog.io/@falling **Github** | https://github.com/fallkim

● **RabbitMQ**를 활용한 비동기 및 병렬처리를 통해 안정적이고 효율적인 서비스 설계 경험이 있습니다. ● 기획부터 배포까지 **End-To-End** 경험을 통해 개발의 전 과정을 이해하고 있습니다. **Spring Boot, Nest.js**, **Django** 로 서버를 구축하고, **Docker** 및 **AWS EC2**를 사용하여 클라우드 서버를 배포할 수 있습니다.

● **'**도전**'**과 **'**성장**'**을 추구하는 사람으로 새로운 기술을 습득하고 실제 결과물을 도출해내는 과정에서 가장 큰 보람을 느낍니다. 또한 커뮤니케이션과 협업의 중요성을 알고 팀원과의 적극적인 소통, 문서화 작업 등 업무 환경 개선을 위해 노력합니다.

**Projects**

**Talk-Taka | AI** 기반 대화형 기록 서비스 **:** AI를 활용하여 캐릭터와 음성으로 대화를 나누며 기록해주는 서비스 2023.12 ~ 2024.02 | Github

기술 스택 | Django REST framework, Django Channels, Celery, RabbitMQ, MySQL, Docker, Nginx, AWS(EC2, S3)

● AI 분석 요청 지연으로 인해 사용자 대기 시간 증가

○ **Celery**와 **RabbitMQ**를 사용하여 AI 분석 요청 API의 처리를 비동기적으로 분산 처리

○ API 응답 시간을 약 **33%** 단축하여 사용자 대기 시간 감소

● 서버 부하 증가로 인해 서비스의 안정성이 저하

○ **Nginx**를 사용하여 웹 서버 앞단에서 로드 밸런싱을 구현

○ 사용자 요청을 여러 서버에 균등하게 분산시켜 서버의 부하를 최소화하고 서비스의 안정성을 높임 ● 사용자와 캐릭터 간의 대화가 실시간으로 이루어지지 않아 대화의 몰입도 저하

○ **Django Channels**와 **WebSocket**을 사용하여 사용자 간의 음성채팅 기능을 구현

○ 사용자가 캐릭터와 실시간으로 음성 대화가 가능하게 하였으며, 캐릭터와 대화 경험을 더욱 자연스럽게 개선 ● 6주 만에 백엔드 서버 구축 완료

○ 짧은 기간 내에 안정적인 API 서버를 구축하기 위해 Django 기반의 API 구현

○ Django의 다양한 내장기능을 활용하여 **Django ORM**, **REST framework**를 적용해 **RESTful** 원칙을 준수하는 **API**를 효율적으로 설계 및 구현

● **AWS EC2** 인스턴스에서 **Docker** 컨테이너를 사용하여 배포

● AI가 질문을 생성하고 사용자가 답변을 받을 때까지의 응답 시간이 불규칙하여 성능 모니터링이 필요 ○ Prometheus와 Grafana를 적용하여 **Docker** 컨테이너 및 노드의 메트릭을 수집 및 시각화 ○ **Cadvisor**로 컨테이너 메트릭을 수집하여 시스템의 안정성을 검증

**Techeer.log |** 프로젝트 결과물 공유 서비스 **:** 테커 내에서 진행한 프로젝트 결과물을 공유할 수 있는 서비스 2023.09 ~ 현재 | Github | 배포사이트

기술 스택 | Spring Boot, MySQL, Nginx, AWS(EC2, S3), Redis

● 실제 배포 및 운영 중인 서비스

○ 사용자들이 테커 내의 프로젝트와 부트캠프 정보를 공유할 수 있도록 개발

○ 외부인들에게도 사이트를 홍보하는 효과를 얻었으며 일일 활성 사용자 수 **200**명을 기록

● 댓글 관련된 데이터를 조회할 때 각 댓글에 대해 연관된 사용자 정보를 가져오는 과정에서 **n+1** 문제로 성능 저하 발생 ○ JPA의 **fetch join** 기능을 활용하여 관련 엔티티를 한 번의 쿼리로 불러오는 방식으로 최적화하여 데이터베이스 접근 횟수를 줄여 성능을 향상

● 에러 처리의 효율성 및 일관성이 필요했기 때문에 에러를 **GlobalExceptionHandler**로 한 번에 처리 ○ 각 예외 상황에 맞는 적절한 에러를 중앙에서 관리할 수 있도록 개선

○ HTTP 상태 코드와 에러 메시지를 반환하도록 설계하여 클라이언트에게 명확하고 일관된 에러 응답을 제공 ● 코드 복잡성과 관리의 어려움에서 벗어나기 위해 **Mapstruct**를 사용하여 DTO를 객체로 변환할 때 발생할 수 있는 오류 방지

○ 객체 간 매핑 작업을 자동화하여 유지보수를 개선하였고, 이를 통해 필드 매핑 시 누락 및 타입 불일치 및 잘못된 데이터 매핑과 같은 오류를 줄임

○ 복잡한 객체 변환 로직의 경우 Mapping 어노테이션을 통해 명시적으로 설정하여 객체 변환 제어함 ● **Mockito**를 사용한 비즈니스 로직 검증 **Unit Test**를 진행

○ 특정 상황에서 서비스가 정상적으로 동작하는 지 확인하기 위해 Mockito를 사용하여 테스트 코드 작성 ○ API 호출 시 예상치 못한 예외 상황이 발생하여 Jacoco를 통해 테스트 커버리지 확인

○ 낮은 커버리지를 가지는 서비스 로직에서 문제가 발생하는 것을 확인하여 테스트 코드 추가 작성 ● **Slf4j**와 **Log4j2**를 활용한 로그 구조화

○ 기존의 비구조화된 로그는 검색 및 분석이 어려워 시간이 많이 소모되고 일관성이 부족한 문제가 존재 ○ 여러 Logging 라이브러리 중 비동기적으로 동작하면서 시스템 성능에 미치는 영향이 적은 **Log4j2**를 통해 로깅을 적용

○ 이를 통해 어떠한 프로젝트가 가장 많이 조회되고, 어떤 API가 가장 많이 호출되는지에 대한 로그 수집 및 분석 수행

**Skill**

**Language** Python, Java, JavaScript, Typescript

**Framework & Library** Django, Django REST Framework, Spring Boot, Node.js, Nest.js

**Database** MySQL, Mongo, PostegreSQL

**DevOps** AWS, Docker, Nginx, Celery, Redis, RabbitMQ, Promethus, CAdvisor

**Experience**

| **2023.08 ~** 현재 | [Techeer] 실리콘밸리 개발자의 SW 개발자 커리어 그룹 |
| --- | --- |
| **2023.06 ~ 2023.08** | [디캠프 x 테커] 2023 하계 실리콘벨리 SW 부트캠프 |
| **2023.12 ~ 2024.02** | [디캠프 x 테커] 2023 동계 실리콘벨리 SW 부트캠프 |
| **2024.03 ~ 2024.06** | [Techeer Partners] 전국 대학생 코딩동아리 4기 백엔드 멘토 - API 기초 세션 진행 |

**Education**

| 2023.03 ~ 2025.02 | 성결대학교 컴퓨터공학과 |
| --- | --- |