

정원희

✉ dnjsgml0578@gmail.com
☎ 010-2268-1950
🌐 <https://github.com/daiseek>
📖 <https://velog.io/@daiseek/posts>



Summary

- Github Actions로 CI/CD를 구현하고, Docker와 GCP를 이용하여 서버를 배포한 경험이 있습니다.
- RabbitMQ와 Celery를 사용하여 비동기 작업 속도를 67%까지 단축시켰습니다.
- End-to-End 프로젝트를 여러 번 경험하여 Frontend, Backend, DevOps 분야와 개발 전반의 프로세스를 이해하고 있습니다.

Projects

EpiLog

2025.06 - 2025.08 | Leader, Backend, DevOps | Github : [Link](#) | Medium : [Link](#)

사용자가 선택한 등장인물을 주인공으로 하여 브이로그를 제작하는 서비스

기술 스택 | Django REST Framework, MySQL, Redis, RabbitMQ, Celery, Github Actions, Grafana, Prometheus, Loki, Promtail, Docker

응답 시간이 오래 걸리는 API를 비동기적으로 구현

문제 상황)

- 사용자가 책 PDF 입력시 Gemini API를 이용하여 책 내용을 요약하는 작업이 약 20s 가량 걸렸습니다.
- 기존에 생성했던 캐릭터 정보를 바탕으로 영상을 위한 대본과 프롬프트 생성시 약 30s 가량 걸렸습니다.
- 해당 기능에서 병목현상이 발생하여 긴 API 응답 시간으로 인해 사용자 경험이 저하되었습니다.

해결)

- RabbitMQ와 Celery를 이용하여 API를 비동기 작업으로 구현하였습니다.
- Redis pub/sub을 구현하여 비동기 작업 상태를 SSE 알림으로 클라이언트에게 전달하였습니다.

결과)

- API 응답 속도가 **약 2s**로 감소하여 사용자 경험을 개선하였습니다.

비동기 작업 속도 개선(약 67% 감소)

문제 상황)

- 600 페이지 이상 PDF를 토대로 등장 인물과 장면 데이터를 생성하는 작업이 약 300s 가량 걸렸습니다.
- 600 페이지 이상의 PDF의 컨텍스트가 많아 Gemini API 호출시 500 에러가 발생하였습니다.

해결)

- 대용량 PDF 파일을 100페이지 단위로 청킹하였습니다.
- Gemini API에 캐릭터 관련 데이터를 생성하는 요청을 동시에 4번 넣어 병렬적으로 수행하였습니다.

결과)

- 등장 인물과 장면 데이터를 생성하는 비동기 작업이 300s에서 약 84s로 감소하였습니다. (**약 67% 감소**) ([Blog](#))
- Gemini API 호출시 500 에러가 발생하지 않아 서비스의 안정성을 향상시켰습니다.

Github Actions로 CI/CD 파이프라인 구축

- Github Actions를 이용하여 CI/CD 파이프라인을 구축하여 안정적으로 서버가 배포되도록 하였습니다.

GCP 환경에서 서버 배포

- Docker로 백엔드 개발 환경을 통일하여 관리하였습니다.
- 프로젝트 이미지를 Docker Hub에서 관리하고, GCP로 서버를 배포하였습니다.

서비스 전반의 CRUD 기능 구현

- Django REST Framework를 PDF 입력시 저장, 캐릭터와 장면 데이터 생성, 대본과 영상 프롬프트 생성 기능을 구현하였습니다.

DevSketch

2024.12 - 2025.02 | Frontend | Github : [Link](#) | Medium : [Link](#)

개발 전 사용자의 아이디어를 PoC하고 초기 세팅과 API, ERD설계까지 도와주는 서비스

기술 스택 | JavaScript, React, Axios, TailwindCSS, Prettier, ESLint, Vercel, Docker

문제)

- 네 페이지에 걸쳐서 사용자가 선택한 값을 전역적으로 관리하기 위해 useContext 혹은 도입했습니다.
- 이로인해 각 페이지에서 선택한 값이 바뀔때마다 props drilling과 불필요한 컴포넌트의 리렌더링이 발생했습니다.
- 중간에 새로고침 시 이전에 선택했던 값이 소멸하였습니다.

해결 및 결과)

- Zustand를 도입하여 useStore() 함수를 통해 필요한 값만 선택적으로 관리하였습니다.
- persist 미들웨어를 적용하여 사용자가 선택한 값을 local storage에서 관리해 새로고침 해도 유지되도록 수정하였습니다.

공통 컴포넌트 제작

- React와 TailwindCSS를 이용하여 페이지 전반에서 사용되는 컴포넌트를 제작하였습니다.

Axios를 이용해 API 연동

- Axios를 이용하여 API를 연동하였습니다.

Truvel

2025.05 - 진행중 | Backend | Github : [Link](#)

여행 일정을 관리하고, 여행지 최적 경로와 위험지역을 보여주는 서비스

기술 스택 | Java, SpringBoot, MySQL, Redis, Docker, Github Actions

Spring boot를 이용하여 CRUD 구현

- 여행 계획 생성 및 조회 기능을 구현하였습니다.
- 여행 국가와 도시를 검색할 수 있는 기능을 개발하였습니다.
- 여행 계획의 편집자를 초대하는 기능과 역할과 초대 수락 여부에 따라 여행 계획에서 관리할 수 있는 기능을 개발했습니다.

Global Exception Handler와 커스텀 예외 제작

- Runtime Exception을 상속하여 커스텀 예외를 제작하고, 예외 발생시 로그를 통해 추적할 수 있도록 하였습니다.

JavaMailSender를 이용하여 이메일 인증 기능 개발

- 사용자가 회원가입시 입력한 이메일 주소로 인증 코드를 발송하고, 사용자가 해당 코드를 입력하면 확인하는 기능을 개발했습니다.
- 코드의 인증 여부를 저장하고 만료 시간을 10분으로 설정해 인증 코드가 재사용 되는 것을 방지하였습니다.

Activities

Techeer SW Boot camp 2024.12 - 2025.02 / 2025.06 - 2025.08

- Techeer Silicon Valley SW 부트캠프

Techeer[테크어] 9기 2025.02 - 현재

- 실리콘밸리 기반 소프트웨어 전문가 성장 프로그램
- 실무 기반 세션 수강 및 네트워킹 행사 참여

Skills

- Framework : React, Spring Boot, Django REST framework
- Database : PostgreSQL, MySQL, Redis, RabbitMQ
- DevOps : Github Actions, Docker, Grafana, Prometheus, Loki, GCP

Education

강릉원주대학교 수학과 재학

2020.03 ~ 2022.02

인천대학교 수학과 학사

2023.03 ~ 2025.02