

손호영

Backend Developer · 인하대 컴퓨터공학과 재학

010-8586-4134 | tls1234568@naver.com | github.com/pyrimidine02 | portfolio.noranekeo.cc

안녕하세요. 인하대학교 컴퓨터공학과에 재학 중이며, 3년째 운영 중인 지하철 추적 서비스 — 지금도 매달 **5,200여 명**이 사용하는 — 를 만든 백엔드 지망 개발자 손호영입니다.

알고리즘 문제풀이에서 길러온 “어떤 입력이 들어와도 깨지지 않는 로직”에 대한 고민이 서비스 아키텍처와 데이터 모델링으로 자연스럽게 이어졌습니다. 아직 배우는 단계지만, 실제 사용자가 쓰는 서비스를 직접 운영하며 부딪힌 문제들이 저를 성장시켰다고 생각합니다.

RailNetwork — 실 트래픽에서 배운 것

서울 지하철 실시간 운행 정보를 다루는 **RailNetwork**는 제가 처음으로 실서비스 코드를 오래 운영해 본 경험입니다. 공식 API가 없어, 지하철 앱이 서버와 주고받는 트래픽을 분석해 요청 형식을 파악하고 문서 없이 응답 스키마를 직접 정의했습니다. 열차 상태가 숫자 코드로만 내려오는 환경에서 이를 “역 도착/출발/지연/운행 종료”라는 도메인 상태로 재해석하고, 시간표와 결합해 지연을 계산하는 로직을 만들었습니다.

공공 API는 자주 불안정했습니다. 그래서 서울 열린데이터 API를 주 소스로 두고 지하철공사 SMSS API를 보완 소스로 병합하되, 두 소스를 같은 포맷으로 정규화해 상위 로직이 데이터 출처를 몰라도 되도록 만들었습니다. 덕분에 안정성 관측 결과에 따라 주 소스를 바꿀 때도 상위 코드를 고치지 않고 반영할 수 있었습니다. 이 과정에서 얻은 “외부가 바뀌어도 도메인 모델은 안정적으로 유지되도록” 설계하는 감각이, 이후 아키텍처와 캐싱을 고민할 때 기준점이 되었습니다.

또 하나 배운 것은 “모든 문제를 코드로만 풀지 않는다”는 점입니다. 공공 API가 주지 않는 열차 편성 정보는 사용자 제보와 투표 검증이라는 클라우드소싱 구조로 사용자와 함께 만들었습니다.

oshi@log — 배운 것을 설계에 적용하다

기획부터 설계·구현까지 맡은 **oshi@log** (서버 운영 중, 앱 내부 테스트)에서는 RailNetwork에서 배운 것을 처음부터 설계에 반영해 보려 했습니다. 데이터가 계속 쌓이는 구조라, Kotlin + Spring Boot 3 + Spring Modulith 기반 이벤트 드리븐 아키텍처로 9개 도메인 모듈을 느슨한 결합으로 분리했습니다. 1인 개발에서

마이크로서비스는 과하다고 판단해, “나중에 필요하면 떼어낼 수 있는 모듈러 모놀리스”를 택하고 모듈 경계 위반을 CI의 아키텍처 테스트로 잡도록 했습니다.

위치 기반 서비스라 “클라이언트가 보내는 위치는 신뢰할 수 없다”는 전제에서 출발했습니다. PostGIS로 거리를 계산하되, 물리적으로 불가능한 이동인지 판정하는 것은 도메인 로직의 몫으로 나누고, 위치 인증 토큰은 기기별 키로 서명해 위조를 어렵게 했습니다. 캐시는 “무엇을”이 아니라 “어디에” 돌지를 데이터 성격으로 갈라, 잠깐의 불일치가 허용되는 참조 데이터는 로컬(Caffeine), 정합성이 곧 보안인 데이터는 Redis에 뒀습니다. 코드 품질은 Detekt/Ktlint와 Kover 커버리지 게이트(서비스 계층 **80%** — 미달 시 빌드 실패)를 CI에 강제해 관리했습니다.

■ 지금 하고 있는 것

현재는 산업기능요원으로 게임 회사 슈퍼플래닛에서 백엔드 개발과 AWS 인프라 운영을 맡고 있습니다. 결제·보상·지표 같은 게임 운영 서버를 개발·유지보수하고 회사 대부분의 서버 인프라를 관리하며 장애에 대응하는데, 개인 프로젝트에서 설계로만 고민하던 문제들을 실무 트래픽 위에서 직접 확인하고 있습니다. 개인적으로는 Go와 배포 자동화로 다룰 수 있는 범위를 넓히는 중입니다. 아직 깊이가 부족한 영역이라는 것을 알기에, 만들어 본 것을 스스로 설명할 수 있을 때까지 되짚어 보는 것을 원칙으로 삼고 있습니다.

■ 맺으며

RailNetwork에서 실제 사용자의 트래픽을 마주하며 문제를 발견하고 고치는 힘을 길렀고, oshi@log에서는 그 경험을 바탕으로 아키텍처와 데이터 모델을 처음부터 설계해 보았습니다. 아직 배울 것이 많은 학부생이지만, 실서비스를 오래 운영해 본 경험과 부족한 부분을 스스로 메우려는 태도를 무기로, 귀사에서 좋은 백엔드 개발자로 성장하고 싶습니다. 감사합니다.

손호영 드림