



안성운

백엔드 개발자 [신입]

✉ asn6878@gmail.com

🌐 github.com/asn6878

📄 asn6878.tistory.com

Introduction

"동료들과 함께 토론으로 성장하고, 본질로 수렴하는 엔지니어"

기획 단계부터 배포, 운영까지 프로젝트의 전체 생명주기를 경험하며 '왜?'라는 질문이 더 안정적인 백엔드를 만든다는 것을 배웠습니다.

동료들과의 자유로운 토론속에서 끊임없이 문제의 본질에 대해 되물고 '정답에 가까운 해법'을 만들어 나가는 방법에 관심이 많으며 궁극적으로 사용자에게 안정적인 환경 속에서 유의미한 가치를 전달하는 백엔드 개발자가 되고자 합니다.

Career



젠스타파트너스 2026/03 ~ 현재

백엔드 엔지니어

국내 최대 규모 상업용 부동산 기업 (주)젠스타메이트의 IT 솔루션 전문 자회사로, 사내 다양한 부서의 업무 전반에 활용되는 백오피스 서비스를 개발 및 운영합니다.

리서치 백오피스 시스템

2026/04 ~ 현재

#Experience

프로젝트 소개 온프레미스 레거시(Windows 네이티브 앱 + Oracle 10g)를 클라우드 웹 서비스로 마이그레이션한 사내 백오피스 시스템

역할 백엔드 설계 및 구현 담당

사용 기술

TypeScript

NestJS

MikroORM

PostgreSQL

Zod

ExcelJS

S3

ECS

● 마감 데이터 불변성과 관리자 보정을 이중 테이블로 관리

문제 확정 마감본은 불변이면서 관리자 보정은 가능해야 하는 요구사항 존재

- **Live/Snapshot 테이블 분리** — 마감 시 정적 팩토리로 현재 상태를 스냅샷에 복사
- 마감 생성을 **단일 트랜잭션**으로 묶어 복사 중 장애 시 원자적 롤백 보장
- 관리자 보정을 **별도 UseCase로 분리**해 운영 흐름과 책임·정책 독립화

● 도메인 정책 레이어로 비즈니스 규칙 산포 방지

문제 역할,시점,필드별 수정 권한이 달라 조건 분기가 UseCase 전반에 산포될 위험.

- **Policy 클래스(Vacancy/Building/Snapshot)** 로 규칙을 모아 단일 파일 수정으로 변경 범위 한정
- **PolicyCtx 타입**을 진입 시 1회 구성·재사용해 반복 DB 조회 제거
- 인프라 의존 없는 **순수 함수**라 DI 없이 단위 테스트 가능

● 스트리밍 기반 대용량 엑셀 Export로 메모리 리스크 차단

문제 수천 건을 전체 적재 방식으로 Export하면 동시 요청 시 메모리 급증 위험.

- **ExcelJS 스트리밍 Writer**로 임시 파일에 즉시 기록 후 HTTP 스트림 응답
- 처리 루프에 **setImmediate** 삽입해 이벤트 루프 블로킹 방지
- 완료·오류·연결 끊김 모두에서 **임시 파일 정리**로 안전한 생명주기 관리

● 레거시 데이터 정제와 스키마 재설계로 DB 용량·조회 성능 동시 개선

문제 신규 서비스에 불필요한 레거시 데이터와 비효율적 스키마(낮은 정규화 수준, 과도한 인덱스, Hot/Cold 데이터 미분리)가 누적되어 DB 용량과 조회 성능이 모두 저하된 상태.

- 신규 서비스 요구사항을 기준으로 **불필요한 레거시 데이터를 폐기**하고 필요한 데이터만 정제·이관, **정규화 수준을 높여** 데이터 일관성 확보
- 실제 조회 패턴을 분석해 **불필요한 인덱스를 최소화**함으로써 쓰기 비용과 저장 공간 절감
- 시계열 데이터를 **Hot/Cold 파티션**으로 분리해 자주 조회되는 최신 데이터 접근 효율을 높이고 과거 데이터는 별도 관리
- 그 결과 레거시 대비 **DB 용량 65% 이상 절감, 주요 조회 쿼리 평균 응답속도 약 70% 이상 개선** 달성

● 시스템 컨텍스트 기반의 의도적 동시성 제어 선택

문제 마감 중 동시 변경 시 불일치 가능하나, 행 단위 락은 관리 복잡도가 큼.

- 사용 패턴(동시 수십 명)을 근거로 **단일 Revision 행**으로 변경 신호를 관리하도록 의도적 선택
- **raw('version + 1')** **아토믹 증분**으로 다중 인스턴스에서도 충돌 감지
- 선택의 한계(고트래픽 락 경합)를 명시하고 **확장 경로**까지 검토 후 채택



플랫폼 통합 개발 블로그 허브

2024/10 ~ 현재

#팀 프로젝트

프로젝트 소개 RSS 기반 개발 기술 블로그 큐레이션 플랫폼. 누구나 자신의 RSS를 등록하고 여러개의 최신 피드 확인 가능.

역할 팀장(스크럼 마스터), 백엔드 엔지니어, 데브옵스 엔지니어

사용 기술

Typescript

NestJS

TypeORM

Redis

MySQL

Docker

NaverCloudPlatform

Terraform

Github <https://github.com/boostcampwm-2024/web05-Denamu>

배포 주소 <https://denamu.dev/>

• RSS 기반 자동 피드 데이터 수집을 위한 크롤링 스케줄러 구현

문제 주기적으로 서비스에 등록된 사용자 블로그 피드를 크롤링하여 데이터로 변환 해야 하는 요구사항 존재

- API 서버와는 별도로, 관심사에 따른 배치성 스케줄링 작업들을 마이크로 서비스로 분리
- 단일 함수에 혼재되어 있던 크롤링 로직을 수집(fetch) → 파싱(XML→DTO) → 필터링(시간 기반) → 저장(DB/Redis) 파이프라인으로 분리하여 단계별 독립적 테스트와 디버깅이 가능한 구조로 개선
- 배치 어플리케이션에 DI Container(TSyringe)를 적용하여 각 모듈 간 결합도를 낮추고 테스트 편의성 향상
- 템플릿 메서드 패턴을 적용으로 새로운 스펙의 XML(RSS 2.0, Atom 1.0 등) 데이터가 들어올 경우에도 해당 파서 클래스를 구현 후 쉽게 추가할 수 있도록 개선

• 모니터링 및 관리 체계 구축

문제 외부 데이터 크롤링 및 파싱의 특성상 예상치 못한 형태의 데이터가 들어오는 경우에 대한 지속적 모니터링 및 빠른 대응이 필요

- 레벨별/일별 로그 파일 분리 후 Loki + Promtail로 수집, Grafana Dashboard를 통한 로그 검색 및 시각화 환경 설계
- NestJS Interceptor 기반 Prometheus 커스텀 메트릭(API 응답시간 히스토그램, 성공/실패율 등) 수집 체계 구축
- Critical 레벨 예외 발생 시 팀 메신저 웹훅 Alert 자동화로 장애 인지 / 대응 사이클 단축

• CI/CD 파이프라인 및 테스트 환경 셋업 자동화

문제 한정된 서버 자원 내 안정적인 배포 환경 및 팀 개발 생산성 향상 필요

- Github Actions 기반 CI/CD 파이프라인 구축 (LeftHook + commitlint로 커밋 컨벤션 강제)
- Self-Hosted Runner와 컨테이너 레지스트리로 빌드/실행 환경 분리해 배포 시 서버 부하 최소화
- TestContainers 기반 자동 통합테스트 환경 셋업 구축 (테스트용 MySQL, Redis 컨테이너 생성)

• 유지보수 및 변경에 유연한 테스트 코드 관리

문제 운영간 비즈니스 로직에 대한 구조개선을 진행하며 기존에 존재하던 테스트 코드의 변경이 함께 필요해지는 문제 발생

- 내부 구현보다는 공개 인터페이스와 비즈니스 로직에 집중한 테스트를 작성하여, 유지보수 간 테스트 코드의 관리 복잡도 감소
- 일부 복잡한 내부 private 메서드의 경우 리플렉션을 사용해 제한적으로 엄격한 검증 수행
- Fixture 패턴을 통한 테스트 데이터 관리로 재사용 가능한 테스트 데이터 생성으로 테스트 코드 유지보수성 향상

• 확장 가능한 OAuth 2.0 기반 소셜 로그인 아키텍처 구축

문제 다양한 소셜 로그인 플랫폼의 상이한 인증 프로토콜을 통합 관리해야 하는 복잡성 존재

- OAuthProvider 인터페이스 설계로 인증 흐름(인가 URL 생성, 토큰 교환, 사용자 정보 조회)을 표준화
- NestJS의 커스텀 Provider를 적용하여 런타임에 동적으로 적절한 OAuth Provider 인스턴스를 사용할 수 있도록 설계
- 플랫폼별로 다른 응답 구조를 통일된 별도 타입으로 정규화 하여 표준화된 토큰 및 사용자 정보 처리 파이프라인 구축



지출 내역 관리 SNS 플랫폼

2024/03 ~ 2024/10

#교내 졸업 프로젝트

프로젝트 소개 소비 습관 개선을 위한 소비 지출 내역 기록, 관리, 커뮤니케이션 플랫폼

역할 백엔드 엔지니어

사용 기술 Java SpringBoot JPA MySQL Redis AWS Docker

Github <https://github.com/CollaBu/pennyway-was>

• 개발이 상당 부분 진행된 프로젝트에 투입 후 빠른 적응

문제 나머지 팀원들에 비해 사용되는 프레임워크 및 아키텍처에 대한 이해도가 상대적으로 낮은 상황

- 적극적으로 코드리뷰에 참여하며 비즈니스 로직을 이해하기 위해 도식화 수행
- 작업 시 기존에 적용되어있는 디렉토리 구조, 의존성, 컨벤션 숙지하여 기능 구현 수행

• 애자일 스크럼 방식을 적용한 팀 프로젝트 참여

문제 기존 팀 업무 처리 방식에 대한 빠른 적응 필요

- 팀이 사용 중인 애자일 스크럼 개발 방법론과 Jira + Confluence 기반의 협업 프로세스 습득
- 스프린트 계획 회의를 통해 목표를 설정하고, 할당된 태스크를 Jira 이슈 및 하위 태스크로 세분화하여 관리
- Jira Smart Commit, Issue Linking을 적극적으로 활용해 커밋/브랜치/PR을 자동으로 이슈와 연동시켜 업무 추적 효율 증대

• 비동기 처리를 활용한 응답속도 개선

문제 문의하기 API에 메일 발송 로직이 동기적으로 엮여있어 응답 속도가 현저히 느려지는 현상 발생

→ 추후 수평확장에 유리한 형태를 위해 @TransactionalEventListener를 활용한 이벤트 발행 방식으로 이메일 발송 로직 구현

→ 이벤트 핸들러만으로는 메일 발송 로직이 개별 스레드에서 실행되지 않고 블로킹 되는 점을 확인

→ @Async 어노테이션을 활용한 비동기 처리를 통한 API 응답속도 약 80% 개선

• 소비 내역 복수 조회 시 발생하는 N+1 문제 개선

문제 소비 내역 복수 조회시, JPA에 의해 해당 소비내역에 연관된 엔티티인 카테고리가 매번 조회 되는 현상을 확인

→ Query DSL의 fetchJoin을 활용해 쿼리 횟수 개선

→ Hibernate가 실제로 발생시키는 쿼리 횟수를 검증하는 테스트를 작성하여 N+1 문제가 발생하지 않음을 보장

Opensource Contribution

TypeORM

Typescript / Javascript 기반 ORM 라이브러리

fix: Migration시 collation 변경을 감지하지 못하는 문제 수정 MERGED

배경 Denamu 프로젝트 진행간 로컬-프로덕션 서버 환경간 특정 컬럼의 Collation이 달라 비교 연산에서 오류가 발생함. 이를 Migration으로 안전하게 수정하려 하였으나, TypeORM의 Migration이 Collation의 변경을 감지하지 못하는 문제 발견

→ 리서치중 실제로 Migration시 Collation은 변경 감지가 되지 않는다는 Issue가 이미 오래전에 보고되었음을 확인

→ Collation을 detection 범위에 추가하고, 이것이 제대로 동작함을 보장하는 테스트 작성

upsert 시 부분 업데이트를 제어하는 updateOnly 옵션 추가 REVIEWING

배경 기존 TypeORM의 EntityManager는 upsert 충돌 처리 시 업데이트 대상을 세밀하게 제한하는 옵션을 제공 하지 않음

→ UpsertOptions 인터페이스에 updateOnly 옵션을 추가해 업데이트 허용 컬럼을 화이트리스트로 지정 가능하도록 설계

→ 업데이트 쿼리 생성 시 옵션을 통한 조건 분기로 updateOnly 기능 구현

→ 특정 드라이버(Postgres 계열)에서 발생 가능한 엷지 케이스에 대한 보정 로직 추가

→ 해당 신규 기능이 정상적으로 동작하는지 보장하는 테스트 작성

Activities

삼성 청년 SW·AI 아카데미 (비영리 교육과정)

삼성 청년 SW·AI 아카데미 [2026/01 ~ 진행중]

Java 전공반 연수생

- Java 기반 객체지향 교육 및 알고리즘 과정 진행

네이버 커넥트 재단 (비영리 교육과정)

부스트캠프 웹 모바일 9기 [2024/06 ~ 2024/12]

웹 풀스택 과정 연수생

- Javascript 기반 문제 해결 역량 향상 및 CS 지식 학습 (HTTP, Socket, Async, EventHandler, FunctionalProgramming ...)
- 다양한 동료들과 코드리뷰, 피어 세션, 페어프로그래밍 등을 통한 협업 및 커뮤니케이션 역량 강화
- 백엔드 개발자에게 필요한 DB 구조 및 활용 능력 학습 (인덱스 설계, 실행 계획 분석, InnoDB 동작 원리 등...)
- 개발자를 위한 블로그 허브 Denamu 프로젝트 팀장 및 백엔드 개발

멋쟁이 사자처럼 대학 (연합 동아리)

영남대학교 11기 ~ 12기 [2023/02 ~ 2024/12]

11기 : 백엔드 트랙 부원 / 12기 : 백엔드 트랙 운영진

- [11기] Python django 기반 백엔드 개발 교육 이수
- [11기] 청각 장애인을 위한 자막 공개 입찰 서비스 HSB 백엔드 개발 및 해커톤 참여
- [12기] 백엔드 세션 Java / Spring Boot 기반 백엔드(Web, DB, JPA, 배포 자동화) 강의 자료 제작 및 오프라인 강의 진행
- [12기] 멘티 프로젝트 진행간 멘토링 수행
- [12기] 동아리 운영을 위한 여러가지 의사결정 진행
- [12기] 최종 해커톤에 멘토로 참여하여 개발간 발생하는 여러가지 상황들 멘토링 수행

영남대학교 - 에스엘 주식회사 (산학협력 활동)

RPA 실무 교육 및 현업 과제 [2023/12 ~ 2024/02]

정보보안팀 / 과제 수행인

- 에스엘 주식회사와 영남대학교 협력 프로젝트인 RPA 프로젝트에 참여
- Microsoft PowerAutomate, Web Automation, Python, Javascript, VBScript 활용 사내 재무팀의 문서 작업 자동화 솔루션 과제를 담당하여 요구사항 이해, 설계, 제작, 운영 및 유지보수 수행
- 사내 SAP, 고객사 ERP 솔루션 기반 조회 재무 데이터 다운로드 자동화 및 엑셀 문서 생성 자동화
- 작성된 작업 문서 자동 메일 전송 프로세스 및 예외 시스템 오류 감지 알림 파이프라인 설계
- 프로젝트 수행 결과, 기관 평가에서 최고 등급 달성 및 최우수 포상금 수여

Award & Certificate & Education

AWARD

대구광역시 공공데이터 활용 경진대회

2023년 / 아이디어 부문 - 최우수상

- 공공데이터(선거 공약 데이터)를 활용하여 선거 공약을 그림으로 보여주는 AI 서비스 기획 및 프로토타입 출품
- 공공데이터를 fetch하고, 데이터 기반 AI 이미지 생성 API 호출하여 결과값을 분류하여 저장하는 배치 프로세스 설계
- 대구광역시 최우수상 수상 및 전국 대회 출전

CERTIFICATE

TOEIC Speaking

영어 회화 능력 시험 / Advanced Low (160)

정보처리기사

2025년 정기 기사 3회

EDUCATION

영남대학교 (졸업 / 23-03 ~ 26-02)

컴퓨터 공학과 / 학사 / 전공 평균 3.7 / 4.5

대구대학교 (중퇴 / 19-03 ~ 23-02)

컴퓨터 소프트웨어 학과 / 학사 / 전공 평균 4.3 / 4.5