길현준

Node.js Developer

kboxstar@gmail.com GitHub ↗ LinkedIn ↗ 웹 포트폴리오 ↗

콘서트 예약 서비스

2024.10 — 2024.11 · 개인

GitHub ↗ 상세 ↗ JPA 비관적 락과 낙관적 락 및 재시도 ↗ Spring Boot 콘서트 예약 시나리오 동시성 문제 분석 ↗ Spring Boot Redis를 활용한 분산 락 구현 ↗

캐시(Cache)와 캐싱 전략(Caching Strategy) ↗

1만 동시 요청 환경에서 좌석 선점·결제·포인트 충전의 동시성 문제를 시나리오별 하이브리드 락 전략으로 해결한 콘서트 좌석 예약 서비스 — 낙관적 락 + Redisson 분산 락, Redis 캐싱 TPS 15배 향상, Kafka 이벤트 드리븐 아키텍처, K6 부하 테스트 기반 병목 개선

Java Spring Boot JPA Redis MySQL Kafka Docker K6

주요 내용

- 낙관적 락으로 좌석 예약 응답 시간 50% 개선 (1,678ms → 835ms)
- 시나리오별 하이브리드 락 전략 설계 (낙관적 락 + Redisson 분산 락)
- Redis 캐싱으로 TPS 15배 향상 (100 → 1,500), DB 쿼리 94% 감소
- Kafka 이벤트 드리븐 아키텍처로 도메인 간 결합도 제거
- K6 부하 테스트 기반 성능 병목 식별 및 개선

담당 기능

- 좌석 예약 — 낙관적 락 기반 좌석 선점, 임시 배정 후 5분 내 결제 미완료 시 자동 해제
- 포인트 충전·결제 — Redisson 분산 락으로 동시 충전/결제 시 잔액 정합성 보장
- 대기열 시스템 — Redis Sorted Set 기반 대기열로 입장 순서 관리, 폴링 기반 순번 조회
- Redis 캐싱 — 콘서트·좌석 조회에 캐시 적용, TPS 100→1,500 (15배 향상), DB 쿼리 94% 감소
- Kafka 이벤트 드리븐 — 예약 완료 이벤트를 Kafka로 발행하여 결제·알림 도메인 간 결합도 제거
- K6 부하 테스트 — 1만 동시 요청 시나리오별 성능 측정, 병목 식별 및 락 전략 비교 검증

깨달은 점

- 비관적/낙관적/분산 락의 트레이드오프를 비교하며 동시성 제어의 본질적 차이를 이해
- 시나리오별 하이브리드 락 전략 설계로 단일 솔루션이 아닌 맥락 기반 선택의 중요성 학습
- Redis 캐싱과 DB 인덱스 최적화를 병행하여 읽기 성능을 극대화하는 계층적 캐싱 전략 경험
- Kafka 기반 이벤트 드리븐 아키텍처로 도메인 간 결합도를 제거하는 비동기 통신 설계 학습

Ledgerly

2025.08 — 진행 중 · 개인

상세 ↗

가족·조직 단위 공유 가계부 앱 — Next.js 16 풀스택, Supabase PostgreSQL, 3단계 역할 기반 접근 제어(OWNER/ADMIN/MEMBER), NestJS Cron 정기거래 자동 처리, 99.45% 라인 커버리지 단위 테스트 + 300건 이상 E2E 통합 테스트

Next.js React TypeScript Supabase PostgreSQL Prisma NestJS Tailwind CSS TanStack Query Zod Vitest Playwright Docker Vercel

주요 내용

- Vitest 397건 단위 테스트, 라인 커버리지 99.45% 달성
- Playwright 300건+ E2E 테스트, 역할별 시나리오 분리
- OWNER/ADMIN/MEMBER 3단계 RBAC + Supabase RLS 적용
- NestJS Cron 정기거래 자동 처리 (Docker 배포)
- 풀스택 통합 테스트 환경 구축 (FE+BE+Supabase)

담당 기능

- 거래 관리 — 수입·지출 CRUD, 카테고리별 분류, 날짜·메모·금액 기반 거래 기록
- 예산 관리 — 카테고리별·기간별 예산 설정 및 실시간 사용율 추적
- 조직 관리 — 다중 조직 생성·전환, 조직별 독립 데이터 관리
- 3단계 역할 기반 접근 제어 — OWNER(전체 권한), ADMIN(멤버·설정 관리), MEMBER(본인 거래 관리)
- 정기거래 자동 처리 — NestJS Cron 서비스로 매일 자정(KST) 반복 거래 자동 생성
- 재무 분석 대시보드 — Recharts 기반 수입·지출 추이, 카테고리별 비율 시각화
- 다국어 지원 — next-intl 기반 한국어·영어 UI 전환

깨달은 점

- Next.js 16 App Router와 Supabase Auth를 조합하여 SSR 환경에서의 인증 상태 관리와 Row Level Security 기반 데이터 접근 제어를 구현하며, 서버·클라이언트 컴포넌트 간 인증 컨텍스트 전파 패턴을 학습
- OWNER/ADMIN/MEMBER 3단계 역할 체계를 설계하며 조직 단위 멀티테넌시에서의 권한 분리와 데이터 격리 전략을 경험
- Vercel Cron 제한을 우회하기 위해 NestJS 기반 별도 백엔드를 Docker로 분리 운영하며, 마이크로서비스 분리의 실무적 판단 기준과 공유 DB 스키마 관리 전략을 학습

- Vitest 단위 테스트 397건(99.45% 커버리지)과 Playwright E2E 300 이상을 역할별 시나리오로 구성하고, 실제 백엔드와 로컬 Supabase를 연동한 풀스택 통합 테스트 환경을 구축하며 테스트 피라미드 전략을 체득

대신물류 배차현황 챗봇

2026.01 — 진행 중 · 개인

GitHub ↗ 상세 ↗

대신물류 배차현황 데이터를 Cheerio로 크롤링하고 카카오톡 챗봇 스킬서버와 Next.js 모바일웹으로 조회할 수 있는 서비스 — Clean Architecture + TSyringe DI, Express 5, Prisma SQLite, Traefik Blue-Green 무중단 배포

TypeScript Express Next.js React Prisma SQLite Cheerio TSyringe TanStack Query Recharts Docker Traefik Tailwind CSS Vitest

주요 내용

- Clean Architecture + TSyringe DI로 계층 분리 설계
- Cheerio 크롤링 + node-cron 자동 동기화 (월~토 매시)
- 카카오톡 챗봇 스킬서버 연동 (노선·차량·도착지 검색)
- Traefik Blue-Green 무중단 배포 + 자동 배포 스크립트
- Next.js 모바일웹 + Recharts 통계 대시보드

담당 기능

- 자동 크롤링 — Cheerio + Axios로 배차현황 데이터 수집, node-cron으로 월~토 06:00~20:00 매시 정각 자동 동기화
- 카카오톡 챗봇 스킬서버 — 카카오 i 오픈빌더 연동, 노선코드·차량번호·도착지 검색, 일별 전체 현황 및 통계 조회
- REST API — 노선·차량·날짜별 검색 엔드포인트, 통계 조회 API, 수동 동기화 트리거
- 모바일 웹 — Next.js 기반 반응형 모바일 UI, TanStack Query 서버 상태 관리, Recharts 통계 시각화
- Blue-Green 무중단 배포 — Traefik 리버스 프록시, 자동 배포 스크립트(빌드→헬스체크→트래픽 전환→롤백)
- Clean Architecture — 도메인·애플리케이션·인프라 계층 분리, TSyringe DI 컨테이너, Value Object(LineCode, SearchDate) 패턴

깨달은 점

- Clean Architecture의 도메인·애플리케이션·인프라 계층 분리와 TSyringe DI 컨테이너를 적용하며, 비즈니스 로직과 외부 의존성(크롤러, DB, 카카오 API)의 결합도를 제거하는 설계를 학습
- 카카오 i 오픈빌더 스킬서버 프로토콜을 직접 구현하며 챗봇 플랫폼의 요청·응답 규격과 시나리오 블록 연동 방식을 이해
- Traefik 리버스 프록시 기반 Blue-Green 배포 구조를 설계하고 자동 배포 스크립트를 구축하며, 무중단 배포의 트래픽 전환·헬스체크·롤백 전략을 경험
- Value Object 패턴(LineCode, SearchDate)으로 도메인 규칙을 타입 수준에서 강제하며, 원시 타입 남용을 방지하는 도메인 모델링 기법을 학습

기타 프로젝트

냠냠위듀 — 구내식당 식단 알림 봇 (개인) — TypeScript, Playwright, Slack Bot API, Express 상세 ↗

병원 채용공고 알림 프로젝트 (개인) — NestJS, TypeScript, PostgreSQL, TypeORM 상세 ↗

스타트업풀 (팀) — NestJS, TypeScript, PostgreSQL, JavaScript 상세 ↗