PAYTALAB · BACKEND APPLICATION

김동현

BACKEND DEVELOPER

이메일 입력 대기
전화번호 입력 대기
github.com/kdh949
kimdonghyun-backend-portfolio.vercel.app

Profile

운영 현장에서 현재 상태만으로 판단할 수 없는 주문·계약 문제를 이력과 규칙으로 복원해 자동화했습니다. 이후 CS와 제품 개발 역량을 확장해 BeanFlow에서는 주문·결제·정산의 거래 생명주기를 설계하고, Orbit에서는 실시간 발표 화면과 Worker OOM을 해결했습니다. 정상 경로뿐 아니라 멱등성, 재시도, 실패 격리와 복구까지 운영 가능한 흐름으로 완성합니다.

Core Skills

- **Domain & Architecture** · 주문·결제·정산·포인트, Aggregate·불변식, 모듈러 모놀리스, REST API, 이벤트 기반 모듈 통신
- **Backend & Data** · Kotlin, Spring Boot, Spring Modulith, JPA, TypeScript, NestJS, Python, PostgreSQL, PostGIS, Redis, Hive SQL
- **Reliability & Performance** · Idempotency-Key, reconciliation, Outbox, 동시성 제어, SQL 실행계획, k6, Testcontainers, 장애 복구·재처리
- **AI-Driven Development** · PRD·ADR·작업 명세, 컨텍스트 설계, 테스트·측정 기반 AI 결과 검증

Experience

쿠팡이츠서비스유한회사

배달파트너 운영팀 EDP Support · 2024.07~2025.10

- 주문·배정·라이더 소속 변경 이력을 결합해 오배달 발생 시점의 **3PL 계약 주체를 복원**하고 일반 라이더·협력사 대표 안내 경로 자동화
- Zendesk Macro·Liquid Markup·JavaScript로 오배달 상담 흐름을 자동화해 실제 처리시간 **72% 단축**, 월 **899만 원** 운영비 절감
- Python·Hive SQL·내부 API·Webhook 기반 안내·이의제기·면책 관리와 E2E QA를 담당해, 프로젝트 전체 월 **2.4억 원** 복합 비용 절감의 실행 기반 구축
- 과적 정산 대상·금액·3PL 여부를 제공하는 Hive Query를 셀프서비스 방식으로 전달해 담당 업무시간을 **4시간에서 1시간으로 단축**
- UTC·KST 장애의 영향 대상을 발송 로그로 복원하고 선택적 재처리한 뒤, 원천별 시간 계약·Dry-run·실패 격리와 수동 복구 절차 도입

Projects

BeanFlow · 다점포 카페 선주문·결제·정산 플랫폼

개인 프로젝트 · 2026 · 8주

Kotlin Spring Boot Spring Modulith JPA PostgreSQL/PostGIS Testcontainers k6

- 가까운 매장→선주문→결제·포인트→픽업·알림→정산·매출 분석을 연결하고, Aggregate 간 ID 참조와 모듈 검증을 적용한 **DDD 모듈러 모놀리스** 구현
- Idempotency-Key·PG reconciliation으로 중복 결제와 승인 결과 유실을 처리하고, 확정 정산은 수정하지 않고 **SettlementAdjustment 원장**으로 환불 보정
- 쿠폰·재고·픽업 슬롯의 낙관·비관·원자 UPDATE를 비교하고, 오류율 **1% 미만**, p95 **300ms 미만**, p99 **700ms 미만**의 거래 API SLO와 초과 발급·oversell 0건 기준 수립
- 매출 Read Model 기반 점주 AI의 수치·기간·근거 평가와, PrintJob 멱등키·offline retry·수동 재출력을 갖춘 POS 프린터 연동 구현

Orbit · AI 프레젠테이션 협업 플랫폼

크래프톤 정글 최종 팀 프로젝트 · 2026.06~2026.07

TypeScript React NestJS Python BullMQ Redis PostgreSQL MinIO/S3

- 고해상도 이미지 요청·decode Promise 공유, current/next preload, ETag/304와 streaming을 적용해 실제 발표 페이지 측정에서 warm transition **100ms 이하**, cold preparation **700ms 이하**, 재방문 payload 0 기록
- Python은 성공했지만 Node Worker가 Base64 JSON parse·decode 중 SIGKILL되는 원인을 확인하고, Python의 Object Storage 직접 업로드와 **Manifest 반환 계약**으로 데이터 경계 재설계
- 음성 인식과 대본 키워드를 결합한 발표 위치 추적·3문장 범위 재동기화, iPad presenter companion과 reconnect backoff 구현
- OOXML Worker 71개, Job Queue 29개, Shared 588개, Storage 4개, Python pytest 835개와 관련 build·정적 검증 통과

Education & Foundations

- **크래프톤 정글 SW-AI Lab 12기** · 2026.03~2026.07 수료
  - C 기반 Malloc·Web Proxy·In-memory SQL API, Pintos Threads·User Program·Virtual Memory 구현
- **충북대학교 정보통신공학 전공**

Awards & Research

- 2025 RISE 문제해결 IDEA 경진대회(한국연구재단 주관) 장려상
- 한국디지털콘텐츠학회 하계 학술대회 발표 논문 채택