

배지혁

개발자 · 신입 (2026년 2월 졸업)

qowlgur121@gmail.com / github.com/qowlgur121 / 포트폴리오 qowlgur121.github.io

문제를 찾으면 직접 조사하고 만들어 본 뒤, 아니다 싶으면 빨리 방향을 바꿉니다.
온라인 스토어를 직접 기획하고 주문 처리 서버를 만들어 하루 최대 1,378건을 처리했습니다.
기업 현장실습과 공공기관 협업에서는 기존 업무·데이터를 연결하는 시스템을 개발했습니다.

1,378건
하루 최대 주문 · 혼자 운영

396개
운영 시스템 테스트 함수

1,809건
매일 반영되는 복지사업 데이터

수상 3회
은상 · 캡스톤 우수상 · 기관 표창

강점

서비스 기획부터 운영까지
주문·고객 응대·광고 데이터를 직접 다루고 실제 거래가 발생하는 시스템을 운영했습니다

AI를 백엔드에 연결
RAG 검색과 의도 분류, 에이전트의 업무·실행 권한을 서비스 흐름 안에서 설계했습니다

조직의 문제를 시스템으로
사내 출입 기록과 공공기관 포털을 기존 시스템 변경 없이 연동했습니다

프로젝트

주문 자동화 시스템 · 온라인 스토어 운영

2026.09 ~ 10 · 개인

Python · SQLite WAL · Oracle Cloud ARM VM · systemd · 네이버 커머스 · 검색광고 API · Telegram

- 스토어 시장 조사부터 상품 등록·광고·주문 운영을 직접 맡았습니다. 결제 확인부터 외부 발주·전달·정산 대조까지 자동화해 판매 6일째 하루 최대 1,378건을 처리했습니다.
- 외부 주문 API의 타임아웃처럼 처리 결과가 불명확한 경우에는 상태를 보류하고 거래 이력과 대조하도록 설계했습니다. DB UNIQUE 제약과 조건부 갱신으로 중복 처리를 방지했습니다.
- 광고 ON/OFF 시간대를 비교해 클릭 수 대신 주문당 광고비·마진을 기준으로 판단해, 광고로 1건 파는 비용이 마진의 약 40%인 것을 확인한 뒤 예산을 늘렸습니다. 검색 노출·유입·주문 변화를 함께 기록해 판매 전략을 개선했습니다.
- AI 점검 에이전트로 운영 이상을 감시하고, 결제나 고객 발송처럼 되돌리기 어려운 동작은 검증된 규칙과 사람 승인 아래 분리했습니다.
- 운영 중 발견한 예외 상황을 회귀 테스트로 고정해 11월 3일 기준 테스트 함수 396개를 축적했습니다.

공공 복지 AI 챗봇 고도화 (RAG) · 전북특별자치도사회서비스원

2025.09 ~ 12 · 3인 팀 팀장

FastAPI · LangChain · PostgreSQL + pgvector · bge-m3 · React 19 · Docker · Nginx | 팀장 · 개발 담당

- 기존 파인튜닝 챗봇의 정보 갱신 문제를 해결하려고 게시판의 변경분을 매일 수집·임베딩하는 RAG로 전환했습니다. 복지사업 1,809건과 시설 18,318곳을 검색합니다.
- 인사가 섞인 복지 질문이 검색 없이 끝나는 오류를 실제 질문에서 발견했습니다. LLM이 의도·지역·대상 조건을 함께 추출하고 실패 시 규칙으로 돌아가도록 6개 의도 경로를 설계했습니다.
- 기간·지역·지원 대상을 고려해 실제 신청 가능한 사업이 먼저 나오도록 검색 결과의 우선순위를 조정했습니다.
- 기존 포털의 소스를 수정할 수 없는 조건에서 삽입형 위젯으로 연동했습니다. 운영자 화면·인수인계 문서를 작성해 기관에 결과물을 전달했고 기관 표창과 캡스톤 우수상을 받았습니다.

OpenContext · 사내 문서를 밖으로 보내지 않는 AI 검색 (MCP)

2025.08 ~ 12 · 3인 팀 · 은상

Java 21 · Spring Boot 3.3 · Elasticsearch 8.11 · Ollama bge-m3 · PostgreSQL · React 19 · Docker | github.com/OpenContextAI/open-context

- 현장실습에서 외부 AI 문서 도구를 쓰며 사내 자료의 외부 반출 문제를 느꼈습니다. 기업 내부에 설치해 문서 검색을 통제하는 Self-Hosted 서비스를 3인 팀에 제안했습니다.
- 제가 맡은 검색 API에 BM25와 의미 검색을 결합하고, AI가 결과 목록을 먼저 살핀 뒤 필요한 문서의 원문만 가져가는 2단계 조회 방식을 구현했습니다.
- 초기 DB·API 명세를 먼저 정해 팀 개발의 연결점을 만들었습니다. 관리자 화면·Docker 배포 설정에도 기여해 학부 작품경진대회 은상을 받았습니다.

JH Personal OS · AI에게 필요한 정보만 전달하고 변경을 검증하는 로컬 시스템

2026.09 ~ 현재 · 개인

Python · Git · MCP · launchd

- AI가 오래된 대화 요약을 현재 사실처럼 사용하던 문제를 해결하려고, 사용자 기록·원본 자료·AI 추론을 분리하고 요청에 필요한 근거만 가져오는 Python 기반 커널을 만들었습니다.
- 공부·프로젝트·취업 등 여러 작업 영역을 manifest로 구분하고, 요청마다 필요한 근거만 조회하도록 설계했습니다. 모호한 요청은 임의로 확정하지 않습니다.
- 상태 변경은 조회→계획→검증된 실행으로 분리하고 revision·중복 실행 키를 검사합니다. 병렬 세션의 미저장 변경은 Git 스냅샷으로 보존하며 31개 검증 규칙으로 점검합니다.

경력

(주)텔로스 · 백엔드 개발 현장실습

2025.07 ~ 08 · 백엔드 전담 · 커밋 89개

Java 17 · Spring Boot 3.5 · JPA · Spring Security · JWT · SQL Server · Docker Compose · Nginx

- 기존 출입 DB는 수정하지 않고 30분마다 기록을 읽어 직원별 출근·재실·퇴근을 판정했습니다. 처리한 기록은 표시해 다시 읽지 않게 했고, 업무일은 새벽 5시 기준으로 나눴습니다.
- Swagger로 API를 먼저 정리해 프론트엔드 선임과 공유했고, 공통 실패 형식과 상황별 오류 코드 26개를 만들었습니다. 선임의 응답 인터페이스에 맞춰 성공·실패 응답 구조를 통일하고 상태 코드를 표준화했습니다.
- Access·Refresh 이중 토큰 인증을 만들고 Docker Compose로 회사 서버에 배포했습니다. 배포 후 시간대 차이로 출근 현황이 어긋난 버그를 찾아 한국 시간 기준으로 고쳤습니다.

그 밖의 프로젝트

- **JeonjuEats** (2025.05 ~ 06 · 1인) 고객 · 점주 앱 분리와 주문 상태 흐름 · Spring Boot · Vue 3 · MySQL · AWS EC2 · GitHub Actions · PWA/TWA
- **Calelog** (2025.06 · 4인 팀 · 백엔드 · DevOps) 일정 기록에서 활동일지 Excel 자동 생성 · PRD · ERD · API 설계 주도 · 일정 중복 서버 검증 · CI/CD

학력

전북대학교	컴퓨터공학부 공학사 · 2022.03 ~ 2026.02
성적	전체 평점 3.71/4.5 · 전공 평점 3.83/4.5 3학년 2학기 전 과목 A+ · 학과 1등 (4.50/4.5) 4학년 1학기 4.41 · 2학기 4.17

기술

주로 쓰는 기술	Java · Spring Boot · Python · FastAPI · PostgreSQL · Docker

수상

2025.12	작품경진대회 은상 · 컴퓨터인공지능학부 (OpenContext)
2025.12	캡스톤디자인 우수상 · 전북대학교
2025.11	성과보고회 표창 · 전북특별자치도사회서비스원

자격증

2025.06	정보처리기사 · ADsP
2025.04	SQLD

활동

2023 ~ 2025	JAVICE SW 창업동아리
2022 ~ 2023	ALPS 알고리즘 동아리 스터디 멘토
2019 ~ 2021	공군 병장 만기전역