



# 이세창 · Java Backend Developer

웹 서비스 성능 개선과 안정적인 운영에 강한 백엔드 개발자

경력 7년차

peersc@naver.com · Portfolio 2026

## 프로필 & 요약

Java/Spring 기반 **백엔드 경력 7년차** 개발자로서, 웹 서비스의 성능·결제·보안을 중심으로 안정적인 서비스 운영과 아키텍처 고도화를 담당해 왔습니다.

2021년 1월부터 2025년 6월까지 **블라이스 웹서비스의 운영 및 통합 기능 개발**을 주도하며, SQL 튜닝과 인덱스 설계를 통해 주요 쿼리·배치 처리 속도를 **최대 5~6배까지 개선**하고, 정기결제·인앱 결제 연동 및 Kafka 기반 알림 시스템을 설계·구현해 서비스 안정성과 확장성을 높였습니다.

최근에는 **한국그린전력 홈페이지 리뉴얼 프로젝트에서 1인**으로 사용자 백엔드, 관리자 화면·백엔드, 배치, AWS 인프라 구성, 보안 설계(Rate Limit, Turnstile, JWT/TOTP, AES-GCM)를 엔드투엔드로 설계·구현하며 전체 흐름을 책임졌습니다.

### Core Competencies

SQL 튜닝·인덱스 설계 (5~6배 성능 개선)  
결제/인앱 결제 설계 및 정기결제·영수증 검증  
Kafka 기반 비동기 이벤트·알림 아키텍처 설계  
Spring Batch 배치 설계·운영 최적화

RESTful API 및 관리자 시스템 설계·개발  
보안 설계 (요청 제한·2단계 인증·데이터 암호화)  
AWS 기반 백엔드 배포 및 인프라 구성

### Technical Skills

#### Backend

Java Kotlin Spring Boot  
Spring Security Spring Batch  
Spring Data JPA MyBatis

#### Database

MariaDB Oracle Redis  
SQL 튜닝 인덱스 설계

#### Infra/DevOps

Docker Nginx AWS Lightsail  
AWS S3/CDN Jenkins CI/CD Git  
Kafka Apache NiFi

#### Security

JWT (Token Rotation) TOTP  
Bcrypt Cloudflare Turnstile  
Rate Limit HMAC-SHA256

#### Frontend

JSP JavaScript jQuery Vue 3  
TypeScript Tailwind CSS Vite

#### Monitoring/연계

Grafana Loki/Promtail LUPIN  
Google In-App Billing PASS

### Career Brief

2025.08 - 2026.01

**한국그린전력 홈페이지 리뉴얼**  
백엔드·관리자 시스템 단독 설계·개발

**메인 프로젝트** 2021.01 - 2025.06

**Bllice 웹소설·웹툰 서비스**  
서비스 백엔드 주도 개발·운영

2020.11 - 2022.08

**Bllice Choice 운영·개발**  
웹소설·웹툰 초기 플랫폼 운영·개발

2019.08 - 2020.07

**개방형 검증지원 시스템**  
TTA | 개방형 검증지원 포털 1·2차 구축 참여

# Project 01: 블라이스(Blice)

Main Project

2021.01 - 2025.06 (4년 6개월)

## 블라이스(Blice) 웹소셜·웹툰 서비스

KT 고객사 웹소셜/웹툰 콘텐츠 플랫폼의 운영 및 통합 기능 개발

Role

백엔드 리드 개발, 성능 개선, 결제/이벤트/쿠폰/웹툰 기능 개발, 운영/유지보수 전반

핵심 성과

- 메인 페이지 2s → 300~400ms (5~6배 단축)
- 정기결제/인앱 결제 신규 구축
- 이벤트 캐시·쿠폰·웹툰 확장

### Tech Highlights

Backend

Java 8 · Spring Framework · MyBatis

Infra/DB

MariaDB 10.1.32 · Redis · Kafka · NiFi · Batch

Frontend

JSP, jQuery

CI/CD

GitLab · Jenkins

### 시스템 아키텍처

백엔드

- Java 8
- Spring Framework
- MyBatis

메시징/배치

- Apache Kafka (3대)
- Apache NiFi
- Spring Batch

데이터베이스

- MariaDB 10.1.32
- 메인/통계 DB 분리
- Redis (3대)

프론트엔드

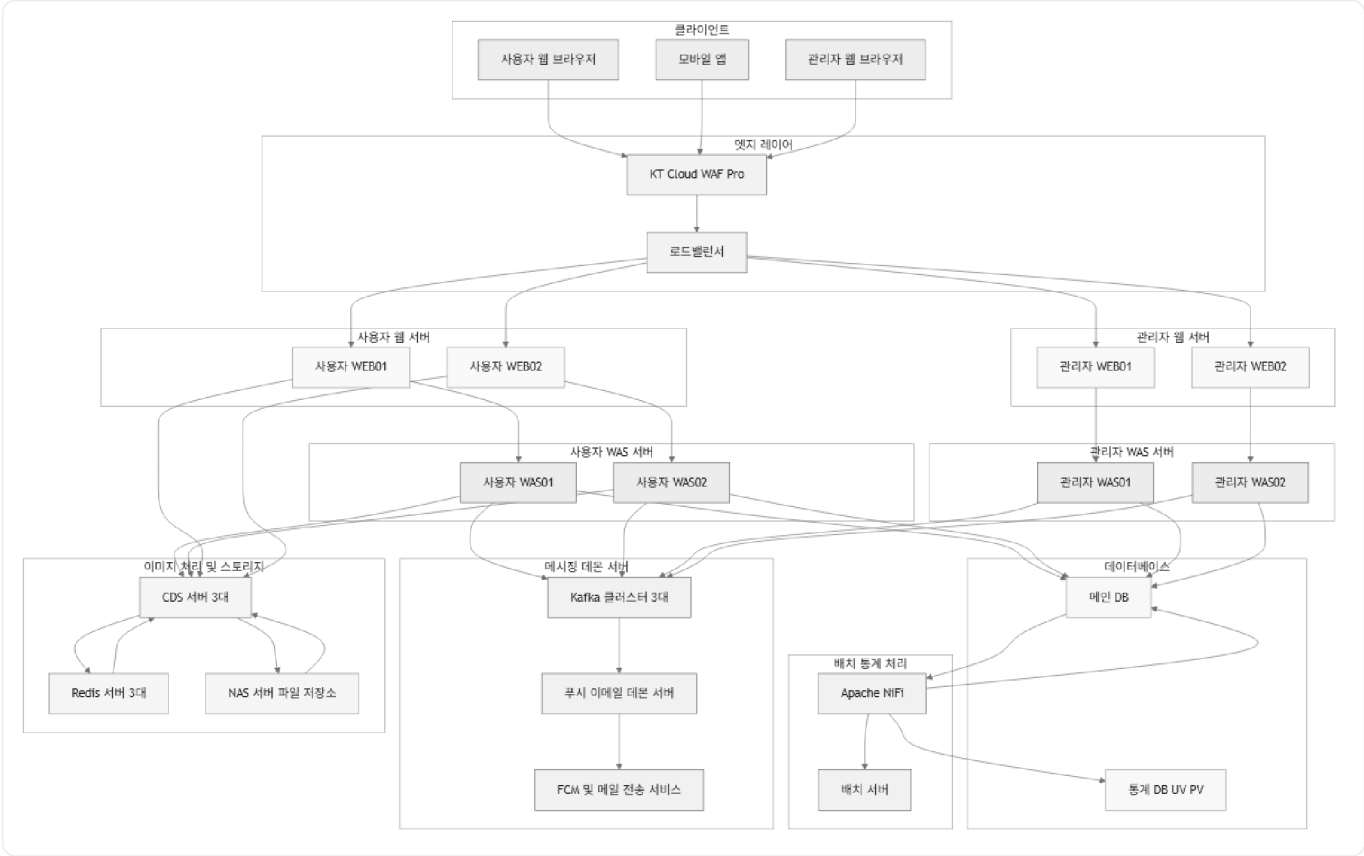
- JSP
- jQuery

인프라/보안

- KT Cloud WAF Pro
- 로드밸런서
- WEB/WAS 이중화

CI/CD

- GitLab
- Jenkins



성능 개선: 메인 페이지 2초 → 300-400ms

<b>문제</b> <ul style="list-style-type: none"><li>메인 페이지 응답 지연, 조회 성능 저하</li><li>NiFi 기반 10분 주기 배치 중 특정 쿼리 1분 이상 소요</li><li>동시간대 트래픽 증가 시 메인 조회 지연 체감 확대</li></ul>	<b>분석</b> <ul style="list-style-type: none"><li>조건 컬럼 인덱스만 사용되어 PK lookup과 정렬 비용이 누적</li><li>보조 인덱스 탐색 → PK lookup → 삭제 반복으로 랜덤 I/O 비용 증가</li><li>실행 계획에서 Using temporary / Using filesort 발생</li></ul>
<b>해결</b> <p>복합 인덱스로 필터+정렬+PK lookup을 한 번에 커버하도록 설계</p> <div><ul style="list-style-type: none"><li>PK: test_seq · 기존 인덱스: sub_seq</li><li>추가 인덱스: (sub_seq, test_seq) · DELETE WHERE sub_seq=1 커버</li></ul></div>	<b>5-6배 성능 향상</b> 삭제 배치 처리 시간 안정화, 피크 시간대 지연 민감도 완화

정기결제 & 인앱 결제 연동

<b>문제</b> <ul style="list-style-type: none"><li>신규 정책(셀렉트 이용권/캐시) 도입으로 정기결제 필요</li><li>모바일 인앱 결제 연동 요구</li><li>느린결제/실패 재시도, 증액·전환 시 사용자 동의 절차 등 복잡한 시나리오 대응 필요</li></ul>	<b>LUPIN + BillKey 정기결제</b> <ul style="list-style-type: none"><li>오전 1회/오후 1회 2회 결제 처리</li><li>오전 실패 고객에게 알림 발송 후 오후 재시도</li><li>오후 재시도까지 실패 시 정기결제 해지 처리</li></ul>	<b>Google In-App Billing</b> <ul style="list-style-type: none"><li>playToken + packageName + productId 기반 영수증 검증</li><li>검증 실패 시 결제 프로세스 중단 &amp; 실패 이력 저장</li><li>VOC 대응 근거 확보</li></ul>
결제 승인·전환 알림을 Kafka 이벤트로 비동기 전파하고, 사용자 동의 흐름과 연동		
<b>성과</b> <ul style="list-style-type: none"><li>오전/오후 2회 결제 처리 + 실패 알림 후 재시도로 정기결제 성공률 개선, 실패 케이스 운영 표준화</li><li>영수증 검증 실패 이력 저장으로 오탐·중복 청구 차단 및 VOC 근거 확보</li></ul>		

이벤트 캐시 & 웹툰 확장

<b>이벤트 캐시 자동 지급 &amp; 정산 제외</b> <ul style="list-style-type: none"><li>구분값 추가로 이벤트 캐시와 구매 캐시 명확히 분리</li><li>즉시 발송: 관리자 서버가 즉시 처리</li><li>예약 발송: 배치가 예약 시각 도달 시 처리</li><li>정산 단계에서 이벤트 캐시 제외 규칙 고정 적용</li></ul> <p>이벤트성 지급 자동화로 운영 비용/오류를 감소, 정산 시 구매 캐시만 반영</p>	<b>웹툰 시스템 신규 구축</b> <ul style="list-style-type: none"><li>기존 이용권 타입이 최대 2글자로 설계됨</li><li>단일 DB 환경 + 온라인 DDL 불가 + 무중단 운영 요구</li><li>2글자 타입 유지 결정, 통계·정산은 이용권 타입 기준으로 웹소설/웹툰 구분</li></ul> <p>스키마 대규모 변경 없이 웹툰 도입/운영 안정화</p>
--	---

운영 고도화 및 품질 관리

<b>게시판 통합 &amp; 마이그레이션</b> <ul style="list-style-type: none"><li>신규 테이블 먼저 생성</li><li>1차 마이그레이션 후 배포</li><li>증분 데이터 추가 마이그레이션</li></ul>	<b>웹툰 표지 이미지 리사이징</b> <ul style="list-style-type: none"><li>단일 이미지 확대 표시 퀄리티 저하</li><li>리사이징 방식 개선으로 체감 화질 개선</li></ul>	<b>기다무 &amp; 쿠폰/모바일 UI</b> <ul style="list-style-type: none"><li>기다리면무료 실행 딜레이 보완</li><li>분산된 쿠폰 등록 화면 통합</li></ul>
<b>작품 예약 공개 배치 안정화</b> <ul style="list-style-type: none"><li>후속 삭제 단계에서 데드락 발생 → 롤백</li><li>예약 공개는 chunk 기반, 후속 삭제는 분리</li></ul>	<b>엑셀 다운로드/업로드 최적화</b> <ul style="list-style-type: none"><li>다운로드: SXSSFWorkbook(스트리밍)</li><li>업로드: SAX 방식으로 메모리 최소화</li></ul>	<b>릴리즈 &amp; 품질 관리</b> <ul style="list-style-type: none"><li>GitLab → Jenkins CI/CD</li><li>롤백 시 직전 브랜치 재빌드</li></ul>

협업 & 회고

<b>협업 방식</b> <ul style="list-style-type: none"><li>고객사 용어가 내부 정의와 다른 경우가 잦아, 반복 확인을 통해 의미를 확정</li><li>Redmine 일감/댓글에 히스토리를 상세 기록</li><li>온/오프라인 회의를 통해 일감 단위로 신속 합의</li></ul>	<b>회고</b> <ul style="list-style-type: none"><li>분석/설계의 우선순위: 예외/장애 시나리오를 먼저 설계</li><li>수익 직결 기능(결제 등)은 꼼꼼한 설계와 지속 모니터링 필수</li><li>성능 개선은 전/후 지표로 증명</li><li>작은 의문도 즉시 확인하고 용어/정책을 명확히</li></ul>
---	---

## Project 02: 한국그린전력 홈페이지 리뉴얼

Project 02 2025.08 - 2026.01 (6개월)

### 한국그린전력 홈페이지 리뉴얼

CMS/사용자 API 구축 - PHP 기반 레거시 시스템을 Kotlin/Spring Boot + Vue 3로 재구축하고, ERP 연동 및 보안 아키텍처 설계

#### Role

백엔드 리드 개발, 인프라/보안 설계

#### 프로젝트 목표

- PHP 기반 홈페이지를 Kotlin/Spring Boot + Vue 3 기반으로 재구축
- ERP 시스템과 연동해 수주/시공 실적·시공사례를 자동 동기화
- 보안·안정성·운영 편의성을 동시에 만족하는 아키텍처 구축

#### Tech Highlights

Backend Kotlin 2.2.x · Spring Boot 3.5.x · JPA · Security

Frontend Vue 3.5.x · TypeScript · Vite · Tailwind

Monitoring Grafana · Loki

Infra/DB MariaDB 11.x · Redis 7.x · Docker · Lightsail

CI/CD GitHub Actions (Docker 빌드·배포)

### 시스템 아키텍처 설계

#### 백엔드

- Kotlin 2.2.x
- Spring Boot 3.5.x
- Spring Data JPA 3.5.x
- Spring Security 6.5.x
- Gradle 9.0.x

#### 프론트엔드

- Vue 3.5.x
- TypeScript 5.7.x
- Vite 7.x
- Tailwind CSS 4.x
- TailAdmin 2.x

#### 배포·인프라

- Docker / Compose
- AWS Lightsail
- Nginx
- S3 + CloudFront

#### 데이터베이스

- MariaDB 11.8.3
- Redis 7.x
- ERP 스냅샷 동기화

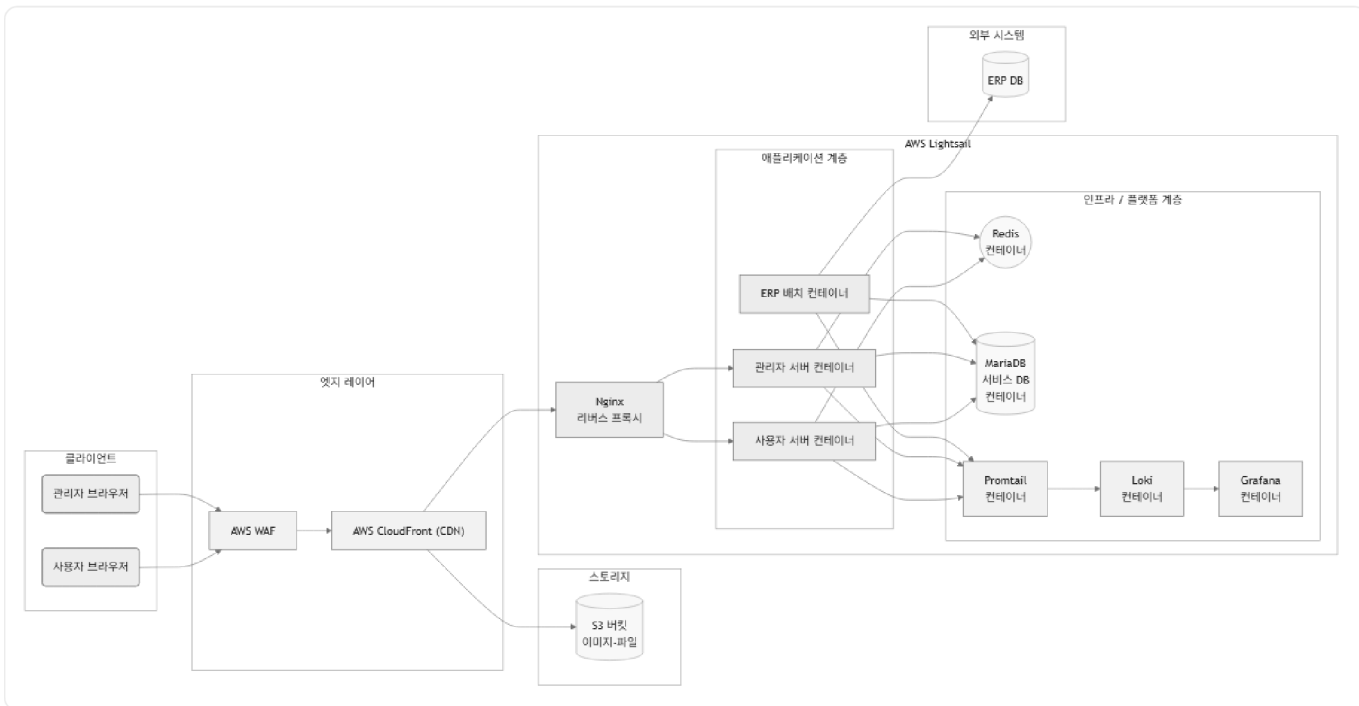
#### 모니터링

- Grafana
- Loki / Promtail

#### CI/CD

- GitHub Actions
- Docker 빌드·배포

**AWS Lightsail 컨테이너 서비스:** 모든 애플리케이션(사용자 API, 관리자 API, ERP 배치, Redis, MariaDB, 모니터링 스택)이 단일 컨테이너 서버에서 Docker로 운영됩니다. 외부 트래픽은 AWS WAF → CloudFront → Nginx를 거쳐 각 서비스로 라우팅됩니다.



보안 및 데이터 보호 설계

<p><b>Rate Limit &amp; IP 차단</b></p> <ul style="list-style-type: none"><li>엔드포인트별 위험 등급(저·중·고) 정의 후 한도 차등 적용</li><li>반복 초과/필수 헤더 누락 패턴이 임계치를 넘으면 IP 자동 차단</li></ul>	<p><b>Cloudflare Turnstile</b></p> <ul style="list-style-type: none"><li>이상 패턴 감지 시 Turnstile 챌린지 요구</li><li>성공 시 단일 사용·짧은 만료시간의 내부 챌린지 토큰 발급</li><li>Siteverify API로 응답 토큰 검증</li></ul>	<p><b>문의 상세 접근 토큰</b></p> <ul style="list-style-type: none"><li>비밀번호 일치 시에만 문의 단위 접근 토큰 발급</li><li>유효기간 동안 비밀번호 재입력 없이 조회/수정/삭제 가능</li><li>UX 개선 + 불필요한 비밀번호 전송 감소</li></ul>
<p><b>액션 토큰 + 위변조 방지</b></p> <ul style="list-style-type: none"><li>X-Action-Token, X-Request-Id, X-Payload-Hash 필수</li><li>Redis 저장값 비교로 변조/재전송 여부 검증</li><li>중간자 공격 방어 레이어 확보</li></ul>	<p><b>관리자 인증</b></p> <ul style="list-style-type: none"><li>JWT Access + Refresh Token Rotation</li><li>Refresh Token: HttpOnly-Secure Cookie 저장</li><li>TOTP 기반 2단계 인증</li><li>비밀번호 검증: Bcrypt</li></ul>	<p><b>데이터 암호화 &amp; 캐시</b></p> <ul style="list-style-type: none"><li>개인정보 AES-GCM 암호화 + 검색용 해시 분리 저장</li><li>정적 텍스트 Redis 캐싱</li></ul>

보안 플로우 다이어그램



## 기타 프로젝트

주요 경력 외 참여했던 프로젝트들입니다.

<div><div><div>Blice Choice 운영 및 유지보수</div><div>2020.11 ~ 2022.08</div></div><div><div>관리자 통계 설계·구현. UI·기능 개선. 웹소셜·웹툰 초기 플랫폼 운영·개발.</div><div><div>Java</div><div>Spring</div><div>MyBatis</div><div>MariaDB</div></div></div></div>	<div><div><div>행복택시 인증/회원 관리</div><div>2023.11 ~ 2023.12</div></div><div><div>PASS 인증 연동, 회원 가입 및 로그인 기능 개발. 본인인증 프로세스 구현.</div><div><div>Java</div><div>Spring Boot</div><div>PASS API</div></div></div></div>
<div><div><div>한국컴패션 비전트립</div><div>2022.08 ~ 2022.11</div></div><div><div>Salesforce 연동, 기능 마무리 작업 및 외부 API 개선. CRM 시스템 통합 연동 작업 수행.</div><div><div>Java</div><div>Salesforce API</div><div>REST API</div></div></div></div>	<div><div><div>Smart Fish Farm (IoT)</div><div>2020.04 ~ 2020.05</div></div><div><div>TTN 기반 IoT 센서 데이터 수집/저장/모니터링. 양식장 환경 데이터 실시간 관제 시스템 개발.</div><div><div>Java</div><div>TTN</div><div>IoT</div><div>REST API</div></div></div></div>
<div><div><div>개방형 검증지원 시스템 1·2차</div><div>2019.08 ~ 2020.07</div></div><div><div>검증지원 포털 일반·관리 기능 개발 및 UI 개선(전자정부프레임워크)</div><div><div>전자정부프레임워크</div><div>Java</div><div>Oracle</div></div></div></div>	