

# 성광식

| Full-stack Developer

기록과 ,  
사용자의 UX를 위한 서비스를 생각합니다.



## 스택

백엔드  
Java  
Spring Boot  
JPA / Security  
Next.JS

프론트엔드  
React  
JavaScript  
TypeScript  
Next.JS

데이터베이스  
MySQL  
Supabase

형상관리 / 클라우드  
Git  
Github  
Vercel

## 교육

한국방송통신대학교 컴퓨터과학과

재학 | 2025

한국 ICT 인재개발원

수료 | 2025

Le Cordon Bleu

2011~2012

면목고등학교

2008~2011

## 연락처

010 - 5043 - 2501  
qjeltk52@naver.com

## 웹페이지

깃허브  
포트폴리오

## 대외활동

MVP Builders 1기

개발 동아리 | 2025 ~ 현재

말레이시아 유학

요리 전문 학교 | 2011 ~ 2012

# 포트폴리오

## Note Keeper

깃허브 Repository

2025.12 ~ 현재

#NextJS #React #TypeScript #Tailwind #Supabase #Vercel #Groq/Naver API

크롬 익스텐션 서비스로 협업 또는 같이 학습을 하는 사용자들이 주된 고객입니다. 사용자는 익스텐션을 확장한 후, 특정 페이지에 대한 메모를 작성하며, 주소록에 저장된 유저들과 작성된 메모를 공유하고, 메모에 대한 AI 검색을 통해 정보를 주고 받을 수 있습니다.

### [ 서비스에 대한 신뢰를 위한 보안 정책 ]

초기에는 Supabase의 signInWithOAuth를 사용했으나, 클라이언트에서 토큰을 관리하므로 토큰의 탈취 가능성을 보완으로 Web Crypto API를 사용하여 Nonce를 생성하고, 구글 인증 서버에는 SHA-256으로 '해싱된 Nonce'를 보내 서명하고, Supabase 서버에는 '원본 Nonce'를 보내 검증하는 로직을 사용하였습니다.

### [ 확장성을 위한 컴포넌트 관리 ]

컴포넌트의 비대를 방지하여 원활한 유지보수/협업을 위해서 UI와 로직을 분리하는 전략을 취했습니다.

주된 기능별로 Hook을 useNoteComments(의견 교류의 CRUD), useNoteAi(AI 분석 및 추천, 결과 데이터 관리), useFriendship(주소록 확인 및 관리) 등으로 나누어 재사용성을 확보하였습니다.

### [ Git Flow 전략 활용 ]

소규모 프로젝트에 적합하게 기능 별이 아닌, 팀원 별 Branch에 대한 개발 후 기능 별 main 브랜치로 병합 후, release 브랜치로 버전 관리 체계를 마련했습니다.

# 포트폴리오

---

## English Edu Website

깃허브 Repository

2025.10 ~ 현재

#Java #Spring Boot #React #JavaScript #MySQL #JWT #PostMan

해외에서 교육 받았던 경험을 바탕으로 제작 중인 영어 교육 웹사이트로, 아이들을 위한 교육 프로그램입니다.

특정한 장소 별로 영단어, 영어 문장을 듣고 녹음하고 재생하며 학습하고, 타이핑 또는 클릭을 활용한 흥미를 끄는 게임을 통해서도 학습할 수 있게 합니다.

### [ RESTful API의 설계 ]

단순히 CRUD를 구현한 것을 넘어 적절한 HTTP 메서드를 사용한 API를 설계하고, ResponseEntity를 활용한 HTTP 상태의 명확성과 Postman API테스트를 통해 정상 동작을 세심히 테스트하며 신뢰성을 확보했습니다.

### [ 계층형 아키텍처 적용 및 리팩토링 ]

@Transactional로 데이터의 일관성을 확보하고, try-catch를 통한 예외 처리로 클라이언트에서 오류 상태에 대한 응답을 구현했습니다. 또한 기능 별 DTO를 나누어 Entity의 불필요한 정보 노출에 대한 방지로 보안 강화에 집중할 수 있는 이점을 부각했습니다.

### [ 보안 정보 유출 위험에 대한 강화 ]

Spring Security와 JWT를 결합하여 서버 확장성을 고려한 인증/인가 시스템을 구축했습니다. 관리자와 일반 사용자의 API를 나누어 역할 별 접근을 관리하며, Secret Key를 환경 변수로 분리하여 보안 정보 유출에 대한 위험을 원천 차단하는 방식을 사용하였습니다.

# 자기소개서

---

## [ 개발과의 만남 ]

IT 업계에 종사하는 지인들에게 회사 업무에 대해 종종 듣긴 했지만, 당시에는 전공을 바꾸게 될 거라는 생각은 전혀 없어서 단순히 신기하게만 생각했습니다.

우연히 접한 프로그래밍 강의를 보고 따라해보면서 프로그램을 만드는 과정들과 실생활에 적용할 수 있다는 점이 큰 매력으로 다가왔고, 고민 끝에 어떻게 시작을 해야하는지 모르는 입문자 입장에서 커리큘럼이 잡힌 교육을 받아보자는 생각에 Java 기반의 풀스택 과정을 등록하게 되었습니다.

## [ 사용자의 경험을 로직으로 해결하는 개발자 ]

학원 수료 후 현재까지 동아리 활동을 하며 실생활에 유용하게 사용될 것 같은 아이템에 대한 고민과 공유를 하고, 협업과 리팩토링을 진행하면서 코드에 대한 완성도를 높여나가기 위해 노력하고 있습니다.

사용자 경험을 넘어 협업의 경험을 위해 기존에 중복되는 코드에 대해 재사용성을 고민하고 지속적인 리팩토링과, 해당 기술에 대한 Docs를 참고하여 라이브러리를 적절히 사용하여 어떻게 협업의 유효성 역할을 할 수 있을지 고민하고 있습니다.

또한 서비스의 기본은 신뢰라고 생각하고 있습니다.

그리고 신뢰의 기본은 본인의 정보를 믿고 맡길 수 있는 보안에 있다고 생각합니다.

어떻게 서버에서 필수 정보만 줄 수 있는지 또는 감출 수 있는지,  
클라이언트에서 기능 별로 필요한 정보만을 요청하고 있는지 매순간 방법을 찾고자 노력하고 있습니다.

# 자기소개서

---

## [ 공유와 커뮤니케이션의 이해 ]

혼자 공부하기 보다 문제를 어떻게 해결하는지, 피드백을 주고 받으면 더 좋은 결과물로 이어진 경험을 하면서, 브레인스토밍과 다른 사람의 개발 방법을 이해하는 과정을 좋아합니다. 하지만 여럿이서 협업을 하는 만큼, 먼저 공유할 정보에 대해 이해한 후 공유받는 동료의 혼선을 방지하기 위해 협업툴(Notion, Figma 등)을 사용해 정리를 하고 전달할 수 있도록 노력하고 있습니다.

그런 커뮤니케이션 능력과 개발의 발전을 위해 자격증, 이론에 대한 강의, 서비스 배포를 목적으로 동아리 등 다양한 활동을 이어 나가고 있습니다.

다양한 사람들과의 활동을 통해 자극받고 코드에 대한 개선, 새로운 기술에 대한 정보 공유로 오늘보다 내일 더 나아지는 개발자로 발전하고 있습니다.

클라이언트의 작은 표정 변화를 관찰하며 원인을 분석하던 경험은 개발 방식과 많이 닮아있습니다.

한 제품에 대한 처음과 끝을 함께 하면서, 문제를 추적하고 보완해 온 경험은 더 나은 서비스를 만드는 개발자로 성장하는 기반이 되었습니다.

이를 바탕으로 팀의 신뢰할 수 있는 일원이 되도록 매순간 노력하겠습니다.