

#### CONTACT

phone: 010-3626-0750 | mail: ugs628@naver.com

#### **DEVELOPER**

Yoo JI SUK (2000.06.28) | https://github.com/MonaLisaSy

YOO-JISUK



이력사항

스킬 및 역량

프로젝트 소개

회고록

# 이력사항



**유지석** 2000.06.28 개발자

**\** 010-3626-0750

☑ ugs628@naver.com

### 학력사항

2016.03~2019.02

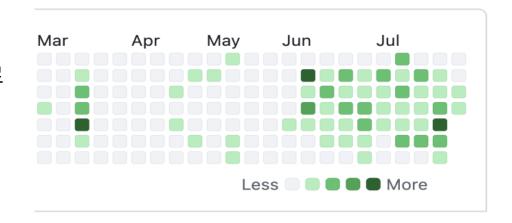
광주전자공업고등학교 전자통신과 졸업

2019.03~2025.02

조선대학교 컴퓨터공학과 졸업

### 교육사항

2025.02~2025.07(5개월) NHN JAVA BackEnd 10기 수료



### 자격사항

2017.07 정보처리기능사

2018.04 컴퓨터활용능력 2급

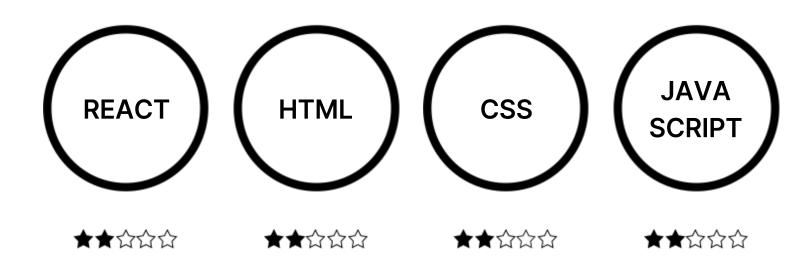
2024.06 SQL 개발자(SQLD)

# 스킬 및 역량

### 백엔드& 협업 툴



### 프론트엔드



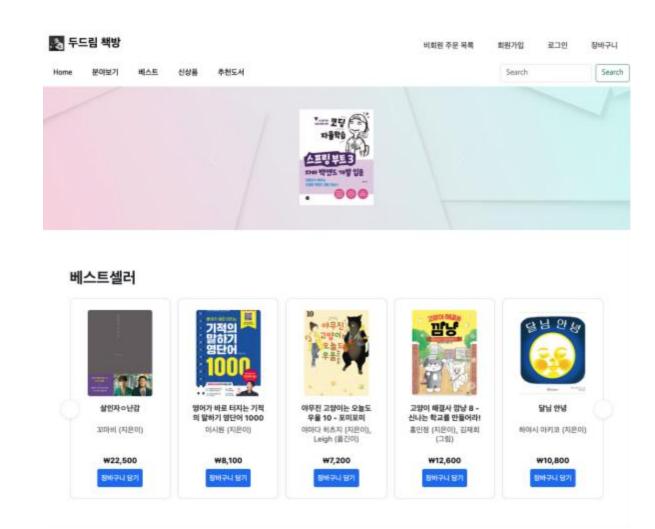
# 경력 기술

## 온라인 서점 사이트(Dodream)

https://dodream.shop

도서를 구매하고 주문이 가능한 온라인 서점 사이트

2025.06~2025.07(8주)



#### 프로젝트 관련 링크

깃허브 주소 : <a href="https://github.com/nhnacademy-be10-DoDream">https://github.com/nhnacademy-be10-DoDream</a>

소개영상 : <u>https://youtu.be/8\_q4XyZloBM</u>

#### 인원

총 7명 (도서 개발자 2명, 주문 개발자 1명, 장바구니 개발자 1명, CI/CD & 쿠폰 개발자 1명, 인증/인가 1명, 유저 개발자 1명)

#### 담당 업무

카테고리 & 태그 개발 : 100%

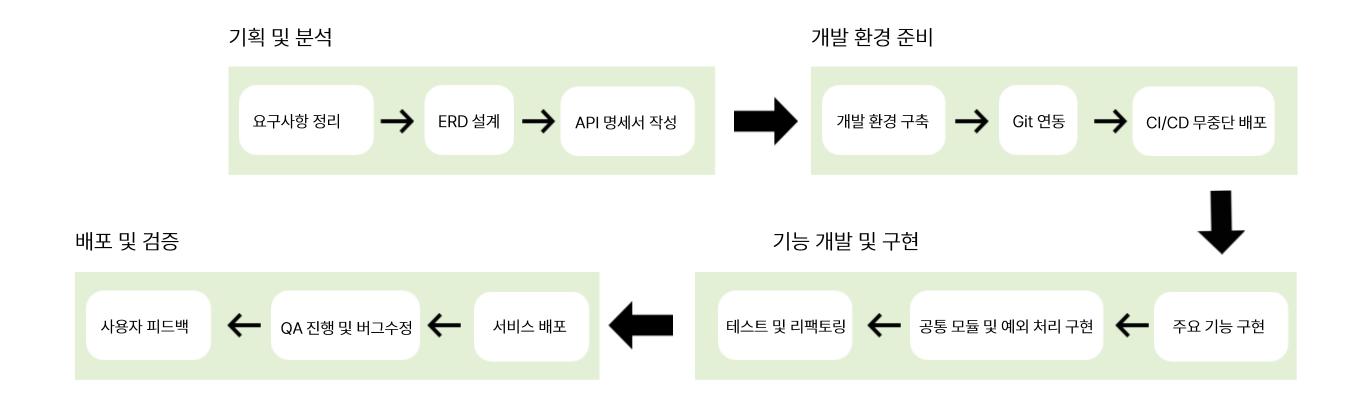
검색 개발 : 90%

관리자 페이지 : 15%

# 프로젝트 진행 순서

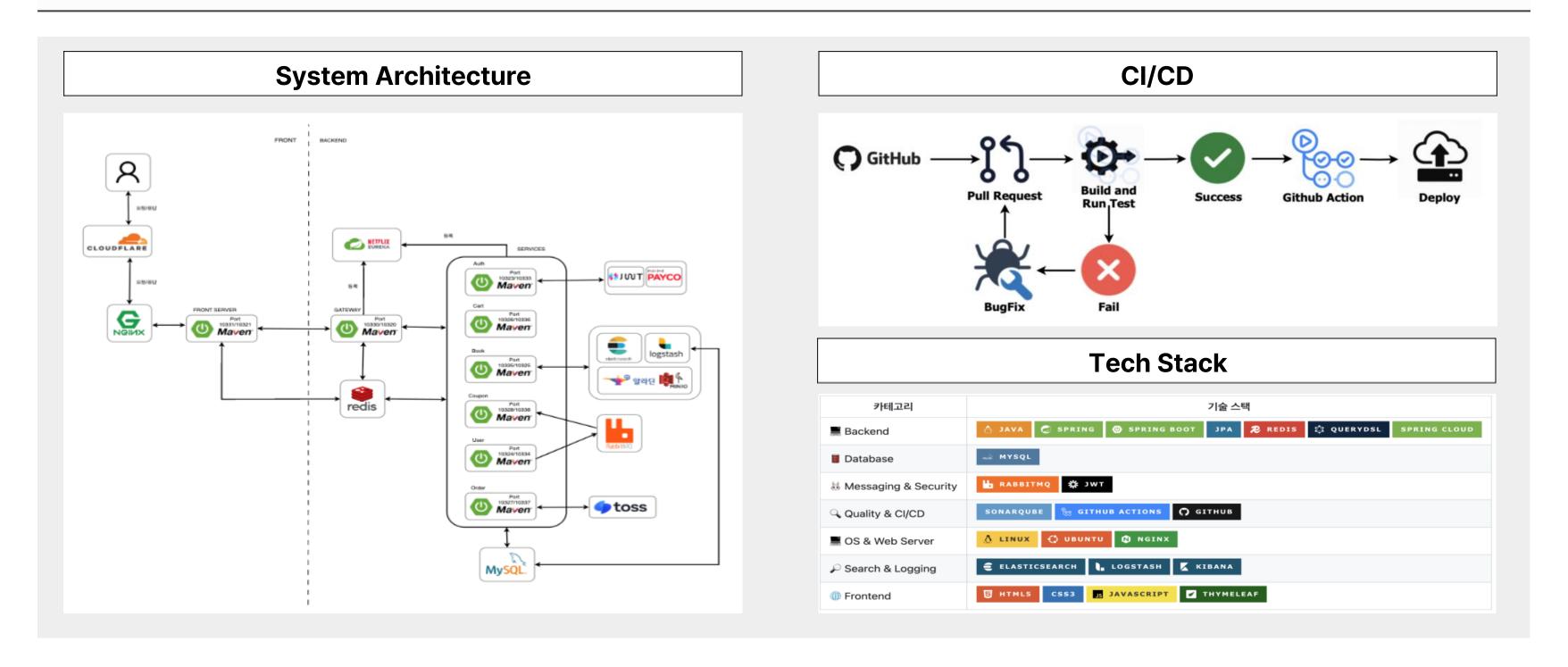
# 온라인 서점 사이트(Dodream)

## 프로젝트 진행 순서

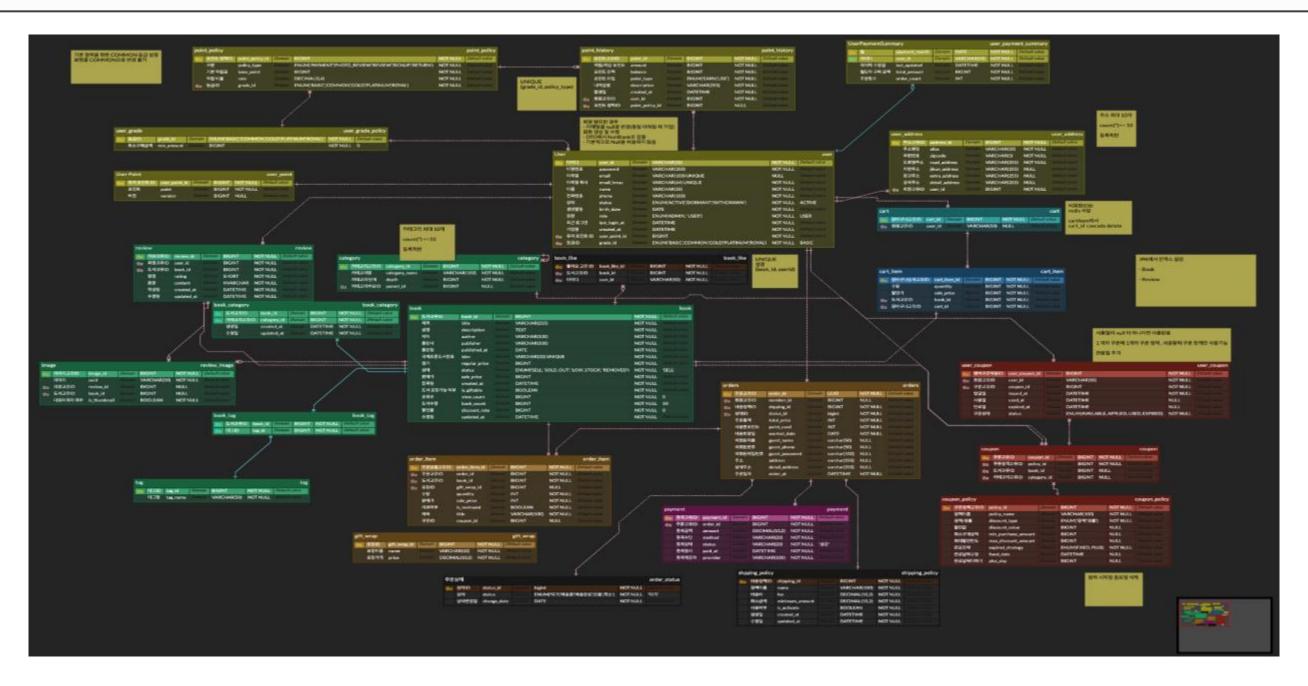


# 아키텍처

## 온라인 서점 사이트(Dodream)



# 온라인 서점 사이트(Dodream)



https://www.erdcloud.com/d/nHdEc6k4QQA34zRwr

## 온라인 서점 사이트(Dodream)

### 카테고리 도서 정렬

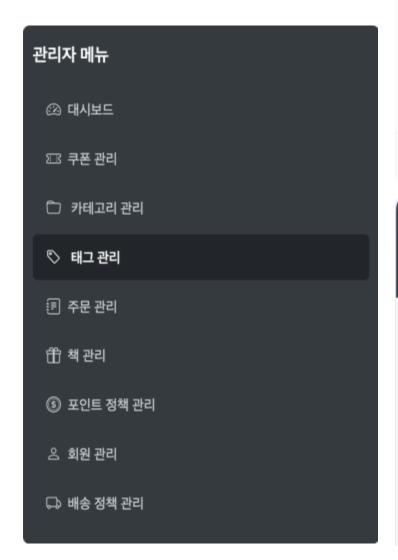
사용자가 관심있는 분야의 카테고리를 누르면 해당 카테고리의 하위카테고리까지 포함해서 도서가 정렬됩니다.

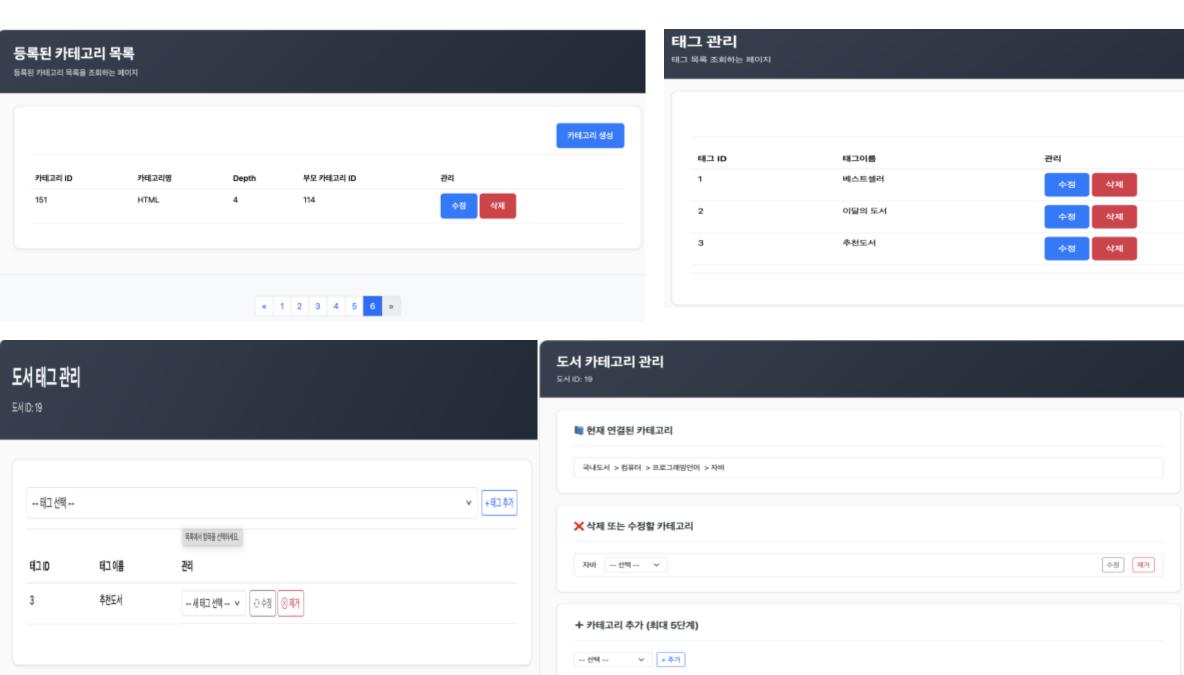


# 온라인 서점 사이트(Dodream)

## 관리자 페이지

카테고리와 태그를 생성가능하고 도서에 생성한 카테고리와 태그를 붙일 수 있습니다.





### 온라인 서점 사이트(Dodream)

### 검색 정렬

사용자가 도서의 제목, 저자, 설명에 관해 검색하면 관련있는 도서가 나오고 카테고리별, 가격대별, 리뷰순, 평점순, 최신순으로 정렬이 가능합니다.

Modern

Java 👀

모던 자바 인 액션 - 랍다, 스트립,

함수형, 리액티브 프로그래밍으로

장바구니 당기

in Actio

오면 자바 인 역선





자바 백엔드 개발 입문

코딩 자율학습 스프링 부트 3 자바

백엔드 개발 입문 - 만들면서 배우

\$9 (XIP0)

W29.700

정하구나 당기

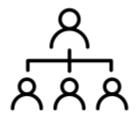




```
SearchBookResponse response = bookClient.searchBooks(keyword, sort, pageable, categoryIds, minPrice, maxPrice);
  Map<Long, Integer> categoryCountMap = response.getCategoryCountMap();
  PageResponse<BookItemResponse> books = response.getBooks();
  if (categoryTree == null) {
      List<CategoryResponse> depth1Categories = bookClient.getCategoriesByDepth(1L);
      categoryTree = new ArrayList<>();
      for (CategoryResponse parent : depth1Categories) {
          List<CategoryTreeResponse> children = bookClient.getCategoriesChildren(parent.getCategoryId());
          categoryTree.addAll(children);
  if (categoryId != null) {
      List<CategoryResponse> breadcrumb = bookClient.getCategoriesPath(categoryId);
      model.addAttribute( attributeName: "breadcrumb", breadcrumb);
  model.addAttribute( attributeName: "selectedCategoryTree", categoryTree);
  model.addAttribute( attributeName: "books", books.getContent());
  model.addAttribute( attributeName: "keyword", keyword);
  model.addAttribute( attributeName: "totalPages", books.getTotalPages());
  model.addAttribute( attributeName: "currentPage", page);
  model.addAttribute( attributeName: "sort", sort);
  model.addAttribute( attributeName: "minPrice", minPrice);
  model.addAttribute( attributeName: "maxPrice", maxPrice);
  model.addAttribute( attributeName: "categoryId", categoryId);
  model.addAttribute( attributeName: "categoryCountMap", categoryCountMap);
  return "book/bookSearchList";
rivate void collectCategoryIds(List<CategoryTreeResponse> nodes, Set<Long> idSet) { 2개 사용위치 ± MonaLisaSy
  if (nodes == null) return;
  for (CategoryTreeResponse node : nodes) {
      idSet.add(node.getCategoryId());
      collectCategoryIds(node.getChildren(), idSet);
```

# 프로젝트 회고

## 쉽지 않아도, 끝까지 하면 결국 해낼 수 있다는 걸 경험으로 배웠습니다.



### 역할

도서 웹사이트의 백엔드 개발을 담당했습니다. 주로 계층형 카테고리 구현과 도서에 태그를 매핑하는 기능을 개발했고, Elasticsearch를 활용한 도서 검색 기능도 맡아 도서를 카테고리, 가격별, 최신순, 리뷰순 등으로 정렬이 가능하게 만들었습니다.



### 소감

단순한 데이터 저장이 아니라, 사용자 편의성과 검색 효율성을 고려한 구조를 설계하는 경험이 인상 깊었습니다. 특히 계층형 카테고리를 구현하면서 재귀적인 트리 구조를 어떻게 응답할지 고민했고, ELASTICSEARCH를 처음 도입해보며 문서 기반 검색 시스템의 특징과 활용법을 익힐 수 있었습니다.



### 어려웠던 점

카테고리를 트리 형태로 구성하면서 각 DEPTH 별 계층 관계를 성능적으로 최적화시키고, 이를 클라이언트에 재귀적으로 전달하는 구조를 설계하는 과정이 쉽지 않았습니다.

도서 검색 시에는 카테고리 정보를 프론트에서 합산·렌더링하는 방식으로 접근했지만, 처리 과정이 복잡해지고 성능에도 부담이 되어 백엔드에서 통합 처리하는 방향으로 구조를 변경했습니다. 다만, 현재 구현에서는 다국어 처리 같은 확장성을 고려한 설계가 부족해 보완이 필요하다고 느꼈습니다.

#### #카테고리 #태그 #검색

#### #재귀 #Elasticsearch

#### #최적화 #확장성

YOO JI SUK PORTFOLIO

# 검토해 주셔서 감사합니다.



CONTACT

phone: 010 3626 0750 | mail: ugs628@naver.com

**DEVELOP** 

Yoo Ji SUK (2000.06.28) | https://github.com/MonaLisaSy