

# WEB 프로젝트 양식

layout: post

title: 내가 쓰는 WEB-App project template

category: 개발

## 목차

테이블이나 그림등으로 간단하게 표현하는게 좋다.

## 개발 목표와 의도 또는 요구 사항

### 목표

어떤 가치를 만들고자 하는지.

어떤 문제를 해결하고자 하는지에 대해 작성한다.

\*(짧고! 명확하게! 100자 이상 넘기지 않기.)

### 요구사항 \* 분석

요구 사항	세부 내용	
사용자 관리	회원가입,로그인,비밀번호 재설정,프로필 관리 등의 기능을 구현하고, 사용자의 인증과 인증 권한 관리할 수 있게 해주세요.	
데이터 저장, 검색 기능 구현	저장,필터링,정렬,검색기능을 구현해주세요.	
파일 업로드, 다운로드	파일을 업로드 및 관리, 다운로드 링크를 생성해 유저가 파일을 다운로드 할 게 해주세요.	
외부 API 연동	구글,카카오 등의 외부 서비스와의 데이터 교환으로 [로그인,회원가입,결제시스템 연동,지도] 서비스 등으로 유저에게 확장 된 기능을 제공해주세요.	
실시간 통신	WebSocket, 실시간 채팅, 실시간 알림 등을 구현하여 클라이언트와의 실시간 데이터 교환을 제공 해주세요.	

요구 사항	세부 내용	
보안과 데이터 처리	데이터 암호화, 해시 함수 사용, 개인정보 보호, 데이터 유효성 검사 등의 보안 및 데이터 처리 기능을 구현해 주세요.	
성능 최적화	캐싱, 비동기 처리, 부하 분산, 데이터베이스 인덱싱 등을 활용하여 애플리케이션의 성능을 최적화해주세요.	
모니터링과 로깅	APP의 상태, 로그 및 예외 처리, 성능 통계 등을 모니터링하고 로깅하여 시스템의 안정성과 신뢰성을 유지할 수 있게 해주세요.	

## 개발환경

분류	개발환경
개발 환경	
개발 도구	
개발 언어	
프레임워크	
빌드 툴	
데이터베이스	
버전 관리 시스템	
배포 및 운영	
오픈소스 및 외부 라이브러리	
테스트도구	

- 개발 환경: Windows 11 64bit / Mac OS / Linux
- 개발도구(IDE) : Eclipse, IntelliJ IDEA 등 [해당버전을명시]
- 개발 언어(lang): JAVA(8~20)
- 프레임워크: Spring... 등[해당 버전을 명시]
- 빌드 툴: Maven, Gradle [해당버전을명시]
- 데이터베이스: MySQL, PostgreSQL, Oracle ... 등등!
- 테스트 툴: 앱의 단위 테스트, 통합 테스트 등을 위해 사용되는 테스트 툴. 예) JUnit, Mockito 등
- 형상 관리 시스템: Github, Git [해당버전을명시] 등
- 배포 및 운영: 서버환경 -AWS, Azure 등 / 배포자동화 도구 - Docker, jenkins 등
- 오픈소스 및 외부 라이브러리: Google API, Kakao API 등
- 테스트도구

# 시스템 설계도

고려할 부분이 많다. CI/CD를 인지하며 제작하는것이 좋다.

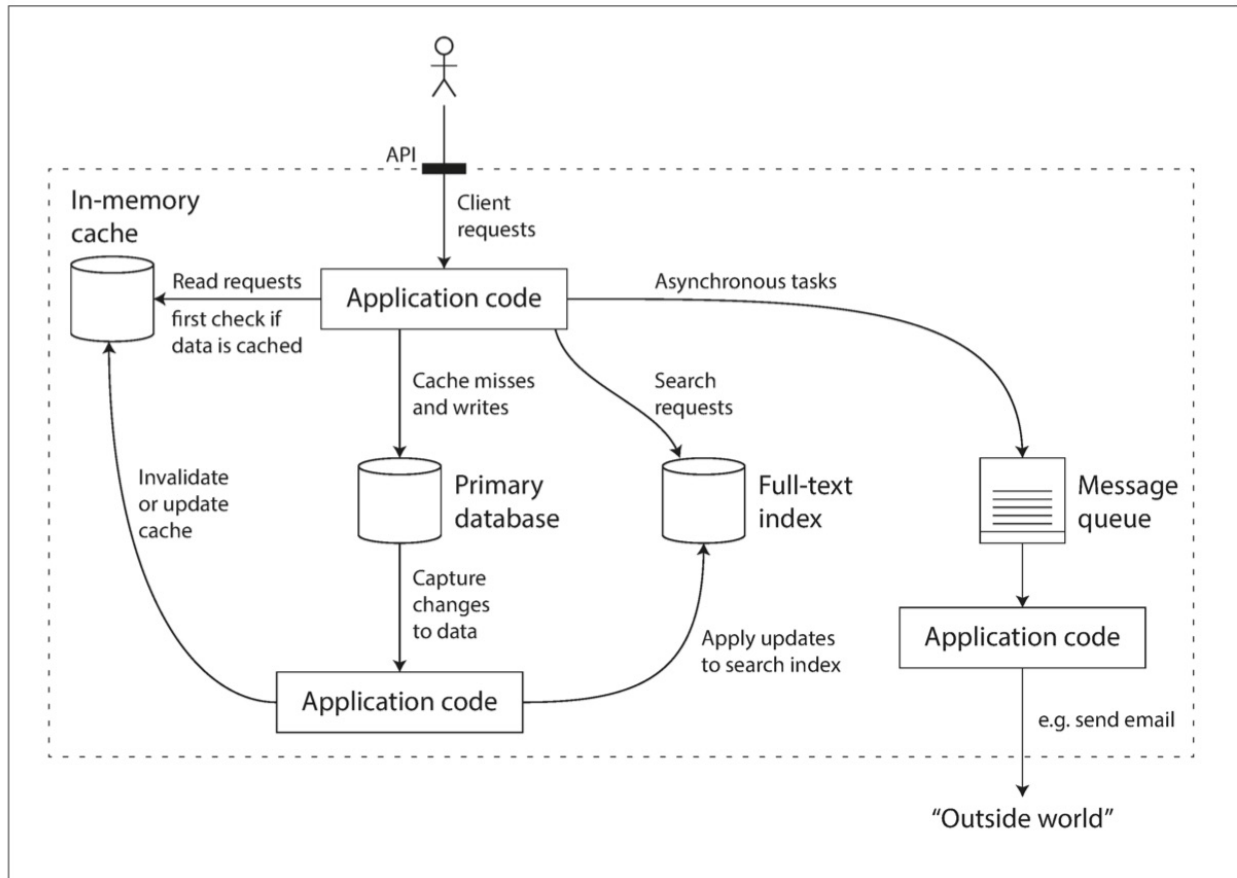
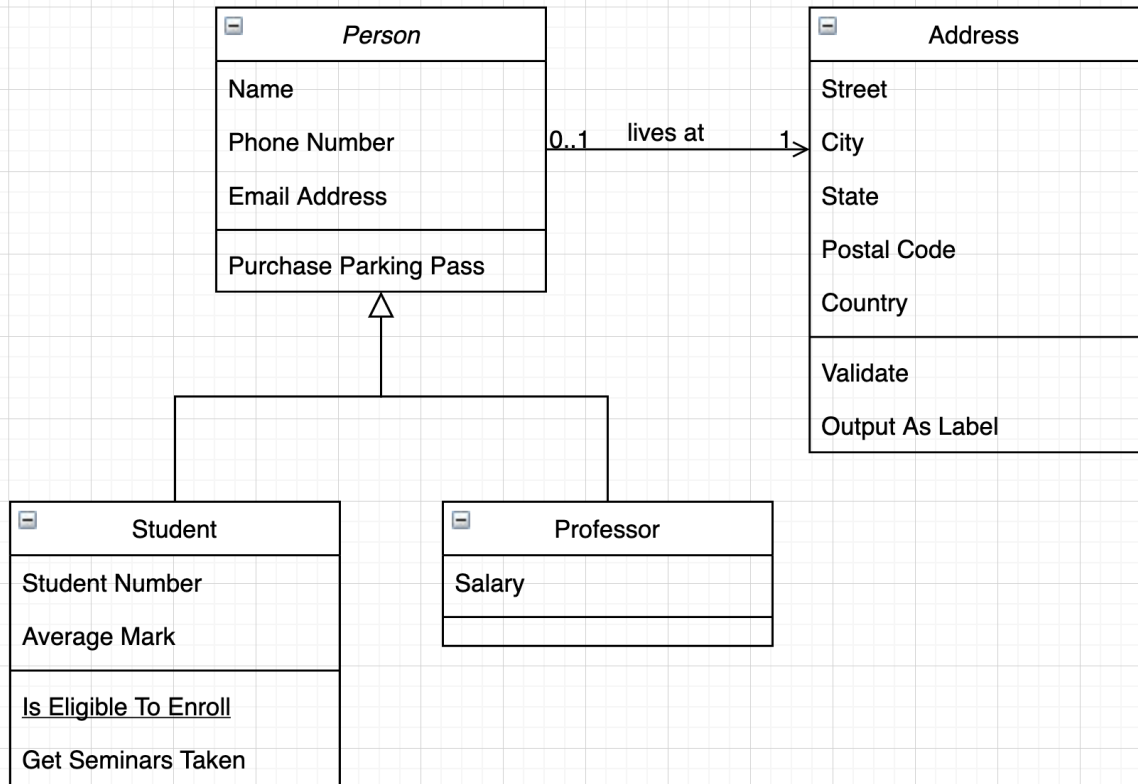


Figure 1-1. One possible architecture for a data system that combines several components.

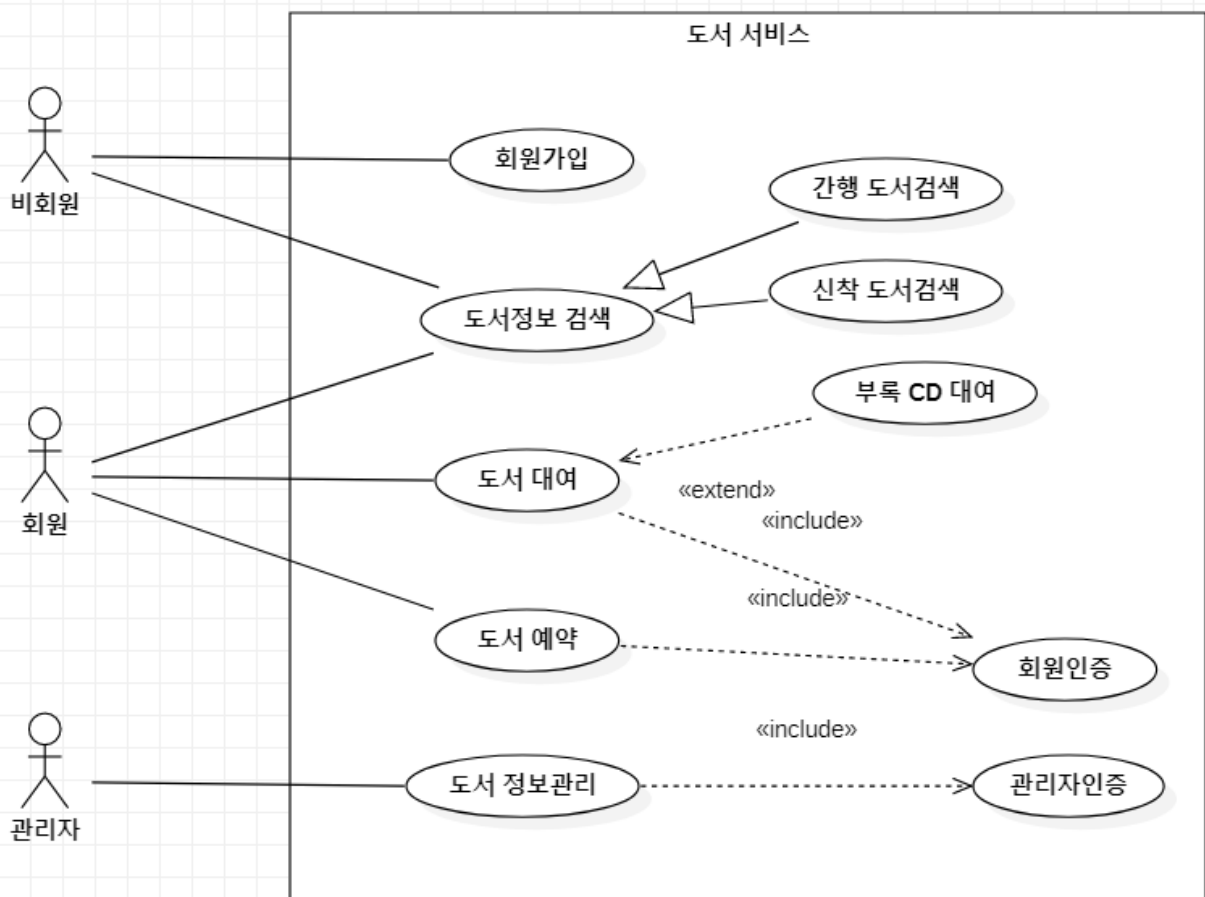
출처: [데이터 중심 애플리케이션 설계](#)

## UML 다이어그램

여러가지 다이어그램 중, 프로젝트의 성격에 어울리는 다이어그램을 활용한다.



(Class 다이어그램 예시)



(UseCase 다이어그램 예시)

# 팀 구성 및 역할

## ex1)

프로젝트매니저 (PM)	프로젝트 총괄. 일정,예산,리소스할당,위험 관리 등을 관리하고 조정함	
시스템아키텍트	아키텍처 설계 와 구축, 시스템의 전반적인 구조부터 데이터의 흐름 등을 고려하고 설계함.	
개발자(FE,BE)	코딩,디버깅,테스트.	
디자이너(UI/UX)	클라이언트의 요구사항과 사용자 경험을 고려하여 인터페이스를 디자인함.	
DB관리자(DBA)	DB의 설계,구축,관리,최적화를 담당함. 보안과 성능을 최우선으로 고려함.	

## ex2)

A(김조장): 게시판, DBA	프로젝트 설계,기획서 작성 / 팀원 역할 분담 게시판 기능 구현,orm 매핑, db설계 및 연동	
B(박팀원): 회원 관리 시스템	회원가입,로그인, 유저 인증 권한 처리,결제시스템,장바구니 등 회원시스템 구축	
C(이팀원): 리소스 관리,설계,매핑	프로젝트 전체 레이아웃 설계와 매핑,유스케이스 알고리즘 설계와 테스트	
D(최팀원): 예외처리,테스트	예외처리,실시간 채팅,별점 기능,게시판 페이징,파일 업로드 기능,검색 기능	

# 위험요소 분석과 대책

### • 기술적 난이도

기술적인 난이도에 대비하여 팀원들의 기술 수준을 확인하고, 필요한 교육과 훈련을 제공합니다. 문제가 발생할 경우 컨설팅이나 외부 전문가의 도움을 받을 수 있는 계획을 수립합니다.

### • 외부 의존성

외부 의존성을 식별하고, 대체 가능한 옵션을 탐색하여 여러 개의 백업 계획을 수립합니다. 외부 의존성이 변경되거나 중단될 경우에 대비하여 알림 시스템이나 대안적인 솔루션을 찾는 계획을 세웁니다.

### • 인력 부족

팀원의 역량과 역할을 고려하여 프로젝트에 필요한 인력을 모집하고 훈련시키는 계획을 수립합니다. 필요한 경우 외부 리소스를 고려하거나 작업 일정을 조정하여 인력 부족 문제를 해결

합니다.

- 예산 초과

예산을 철저히 계획하고 추적하여 비용이 예산을 벗어나지 않도록 합니다. 비용 통제를 위한 감독 체계를 수립하고, 비용이 예상보다 초과될 경우 조치 방안을 마련합니다.

- 일정 지연

가능한 모든 위험 요소를 사전에 식별하고, 일정에 여유를 가미하여 위험에 대비합니다. 문제가 발생할 경우 조기 경보 시스템을 도입하고, 우선순위를 재조정하여 핵심 기능의 우선 개발을 진행합니다.

---