

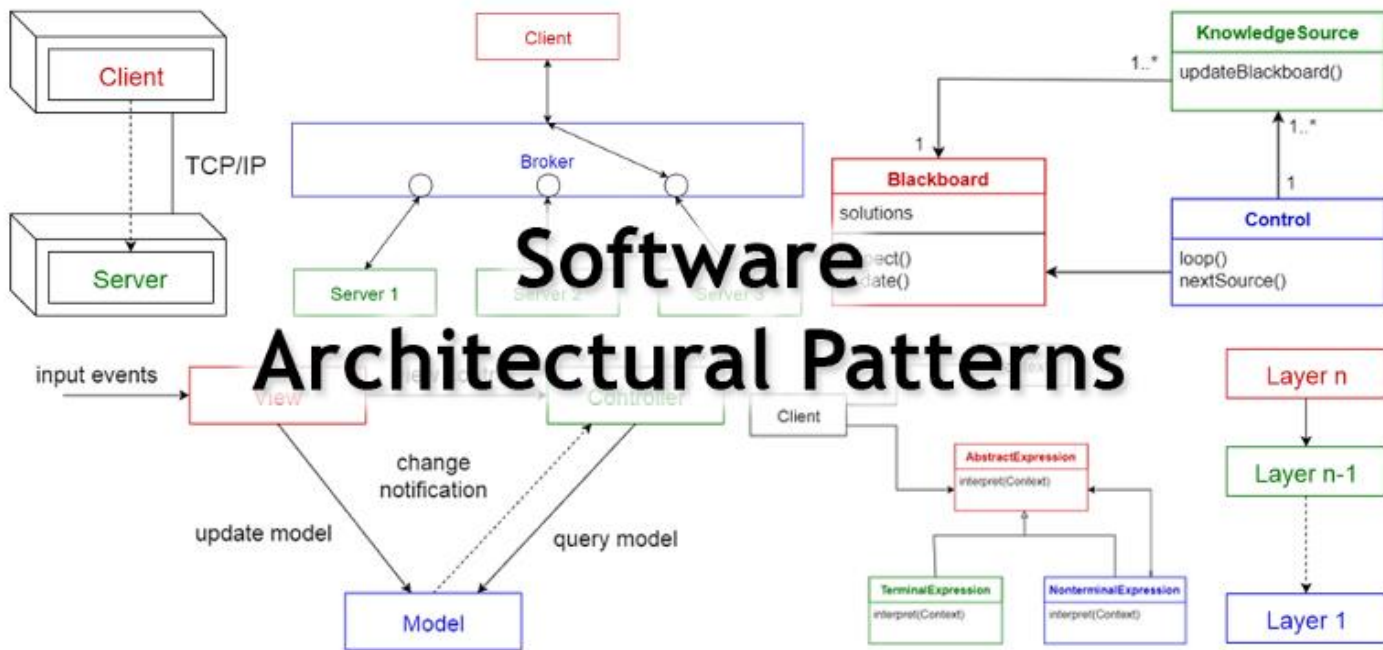
WEB DEVELOPMENT STUDY

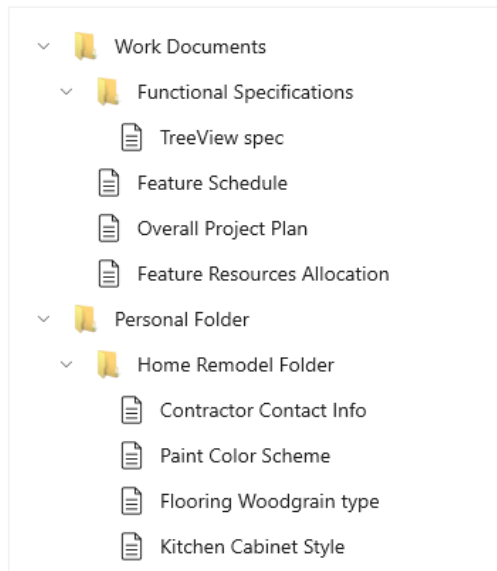
백엔드 스터디 4회차

신동민

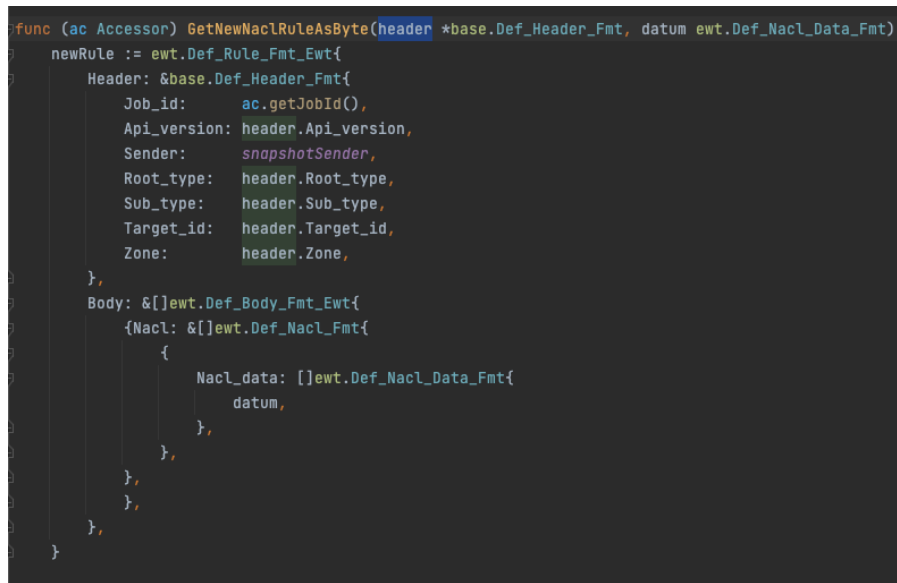
College of Art & Technology
Chung-Ang University



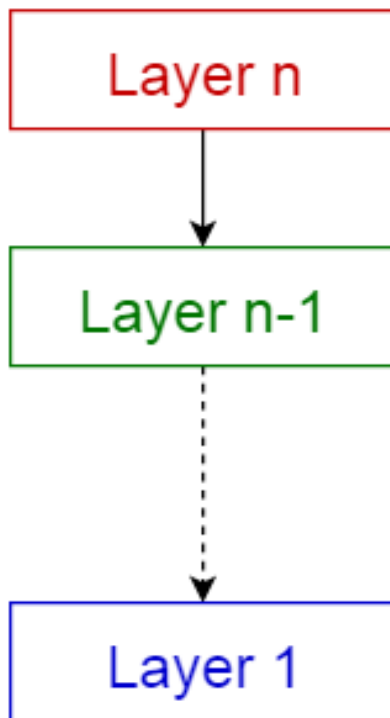


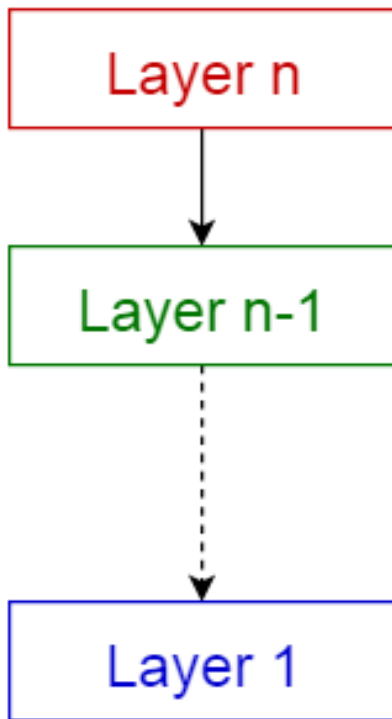


아키텍처 패턴

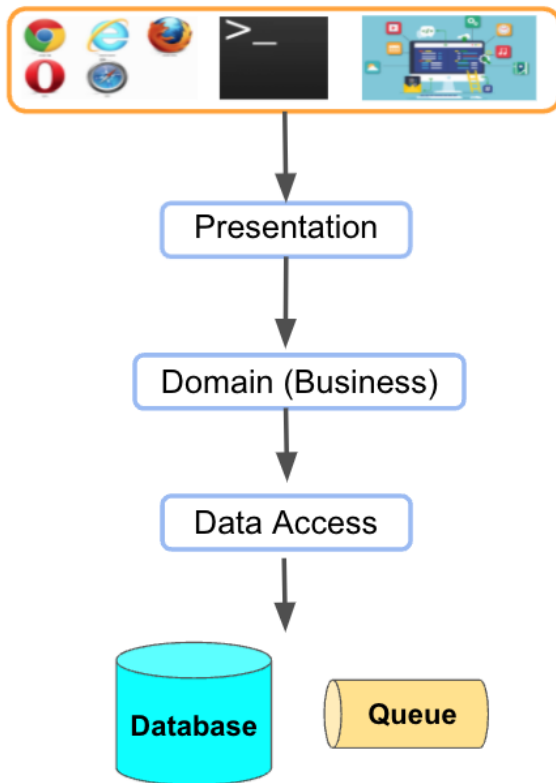


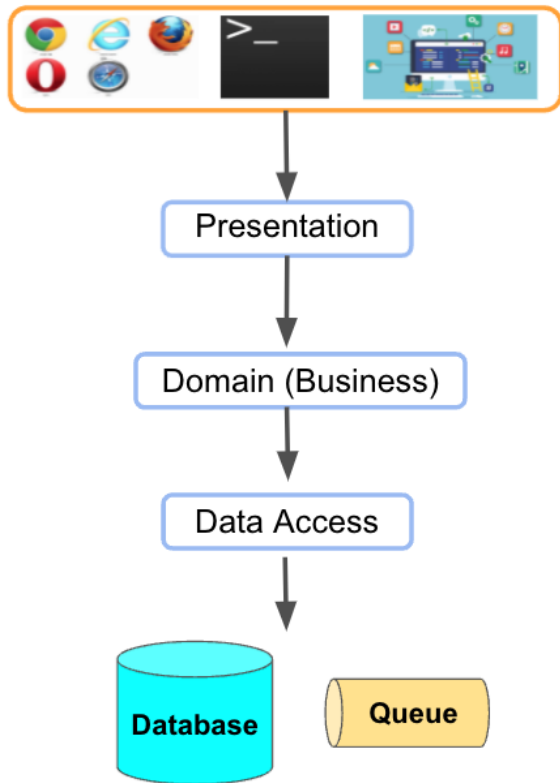
디자인 패턴





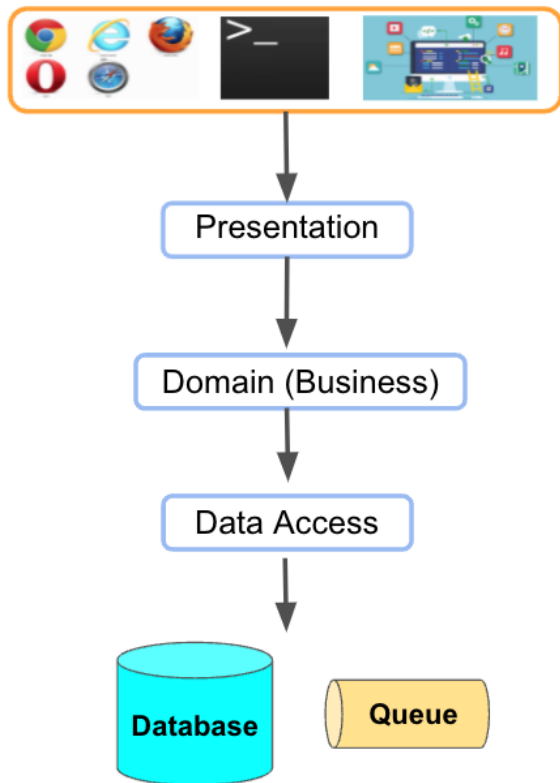
- 하위 모듈들의 그룹으로 나눌 수 있는 구조화된 프로그램에서 사용할 수 있다.
- 각 하위 모듈들은 특정한 수준의 추상화를 제공한다.
- 각 계층은 다음 상위 계층에 서비스를 제공한다.





Presentation 계층

- 사용자와 상호 작용 처리 계층
- CLI, HTTP 요청, HTML 처리 등(=API)을 담당한다.
- HTTP 요청 처리 및 HTML 렌더링에 대해 알고 있는 웹 계층



Domain(Business or Service) 계층

- 서비스/시스템의 **핵심 로직**
- **유효성 검사** 및 **계산**을 포함하는 Business 논리 계층
- 애플리케이션이 수행해야하는 **도메인**과 관련된 작업들을 담당한다.
- 입력/저장된 데이터를 기반으로 계산
- Presentation 계층에서 받은 데이터의 유효성 (Validation) 검사
- 어떤 Data Access 를 선택할지 결정

인터넷 주소

비즈니스 또는 서비스의 영역

도메인 객체 및 모델

비즈니스 또는 서비스의 영역

정의 :

개발 대상이 되는 **특정 문제 영역**이나 **업무 분야**를 의미한다.
즉, "무엇을 위한 소프트웨어인가?"에 대한 답이다.

예시 :

- 인터넷 쇼핑몰 : 상품 관리, 결제, 주문, 배송 등
- SNS : 친구 관계, 피드, 채팅 등
- 병원 시스템 : 환자 정보 관리, 예약, 진료 기록 등

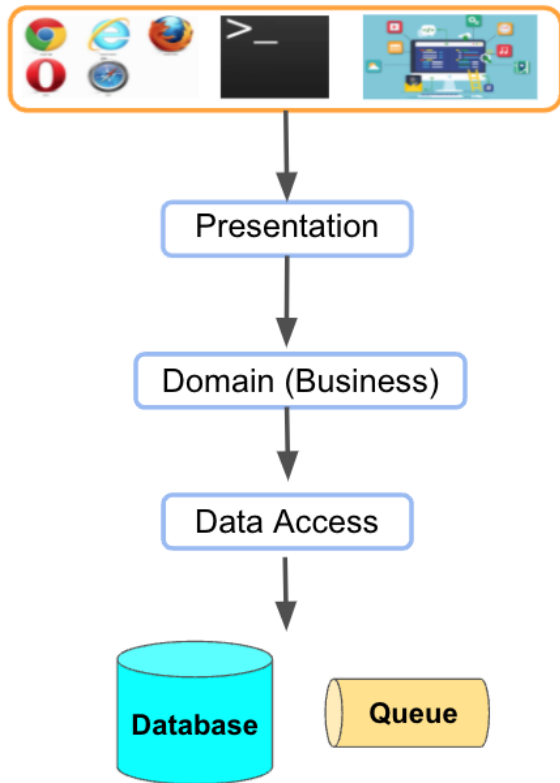
도메인 객체 및 모델

정의 :

소프트웨어 내에서 **비즈니스 규칙**과 **상태**를 표현하는 **핵심 객체**나 **모델**을 의미함.

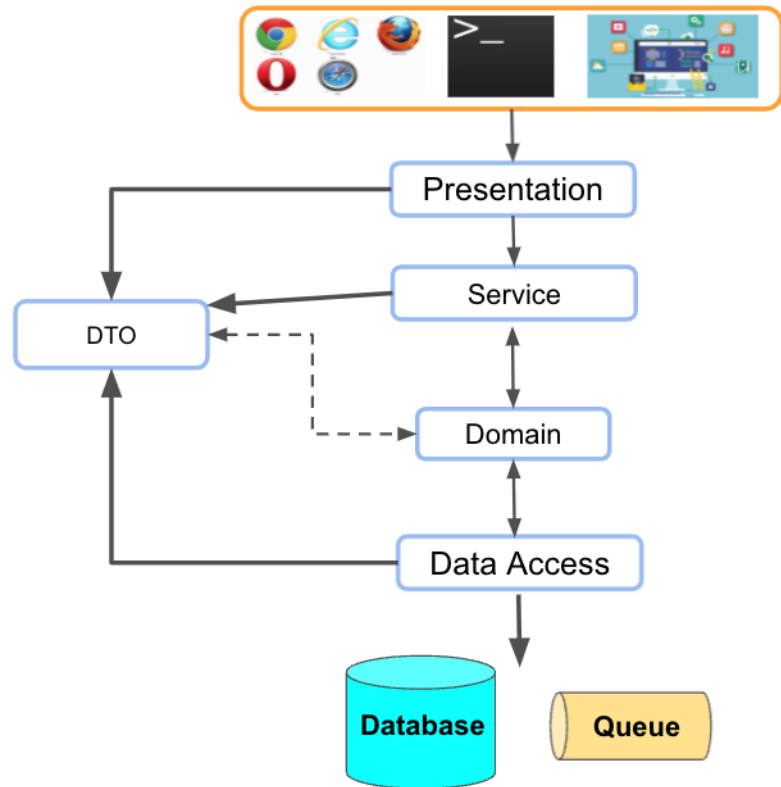
예시 :

- 인터넷 쇼핑몰의 "주문(Order)", "장바구니(Cart)"
- SNS의 "사용자(User)", "게시글(Post)"



Data Access(Persistence) 계층

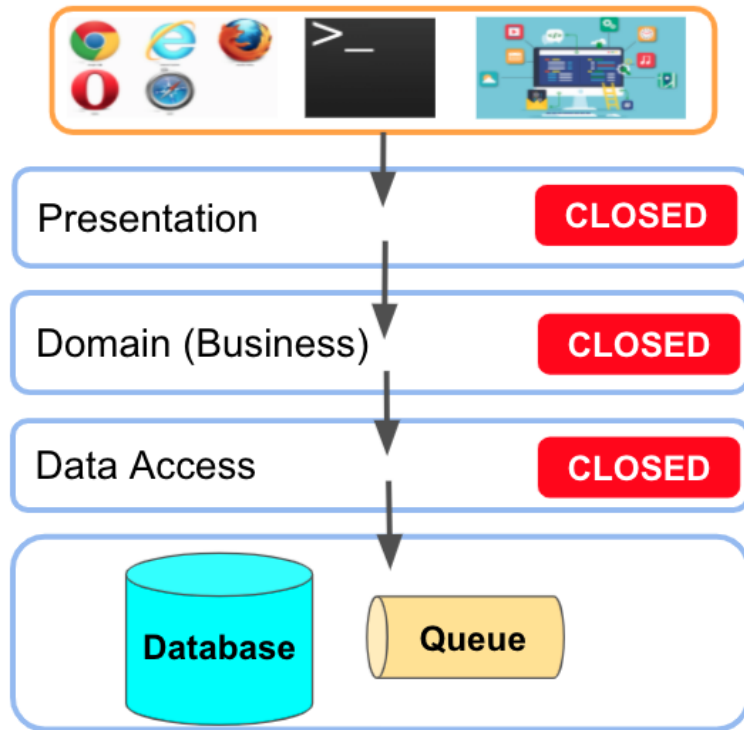
- DAO 계층
- **Database** / Message Queue / 외부 API와의 통신 등 처리
- 데이터베이스 또는 원격 서비스에서 영구 데이터를 관리하는 방법을 분류하는 **데이터 접근 계층**

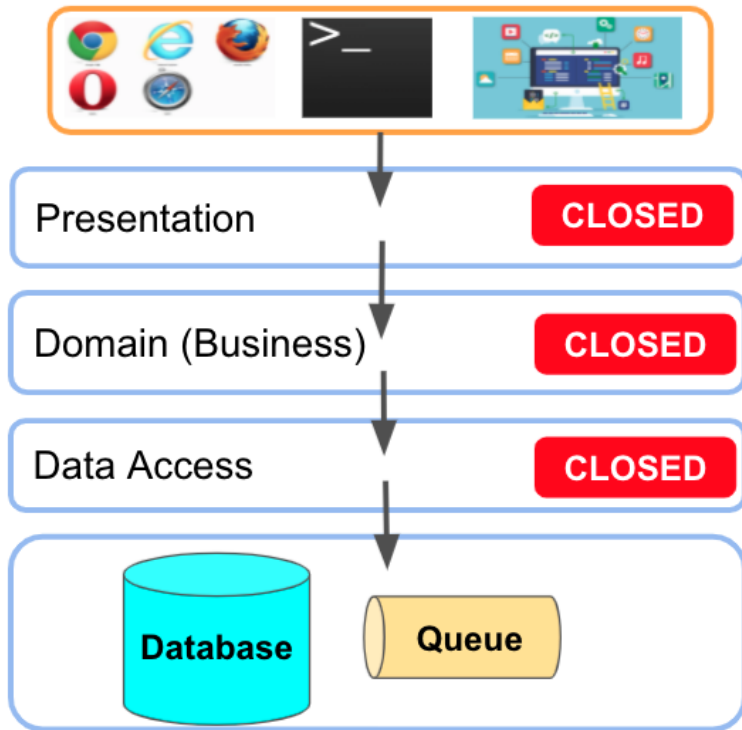


DTO (Data Transfer Object)

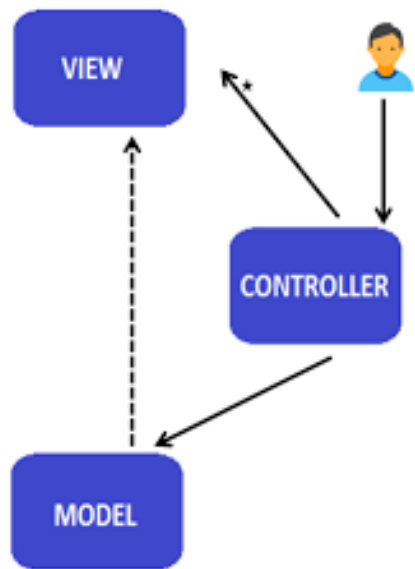
- 도메인 객체가 도메인 계층을 벗어나지 못하도록 지원
- **Presentation** 영역에서 필요한 로직과 **도메인 로직**을 분리할 수 있다.
- 도메인 객체는 단순히 값 조작과 연산이 들어가는게 아니라 주요한 도메인 로직을 담고 있다.
- 화면 노출 혹은 외부 API 제공에 도메인 객체가 사용된다면 public, private 등의 scope 오염부터 시작해서 굉장히 많은 부분에서 **도메인 오염**된다.
- 여러 테이블의 컬럼값을 조합해서 사용하기도 하는데, 적절한 도메인 객체를 선택하기 보다는 DTO를 통해 단순히 값만 담은 형태로 제공한다.

계층화의 핵심은 각 계층은 응집도가 높으면서,
다른 계층과는 낮은 결합도를 가지는데 있다.

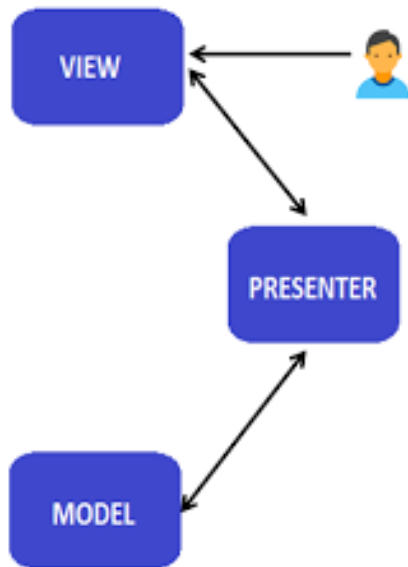




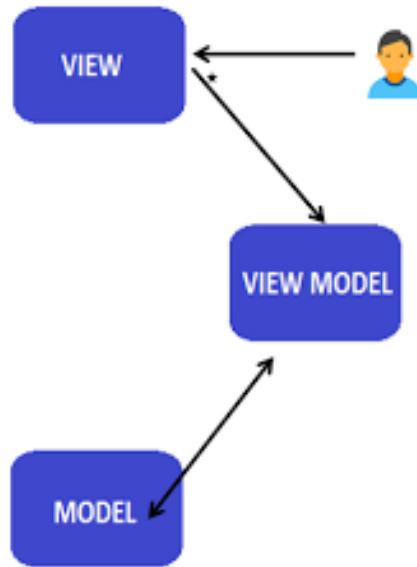
- (1) 관심 범위 축소 (관심사 분리)
- (2) 모듈 교체의 용이성
- (3) 좀 더 용이한 테스트



MVC



MVP



MVVM