



# Software Architecture가 중요한 이유

≡ 태그

Architecture가 필요한 이유

Software Architecture

소프트웨어 구조 또는 소프트웨어 아키텍처는 소프트웨어의 구성요소들 사이에서 유기적 관계를 표현하고 소프트웨어의 설계와 업그레이드를 통제하는 지침과 원칙이다.

출처 : 위키백과

아키텍처란 무엇일까? 본래 건축에서 사용되는 이 단어는 소프트웨어 개발 분야에서도 사용하게 되면서, 그 의미는 건축에서 사용하는 것과는 전혀 다른 것을 나타내게 되었다.

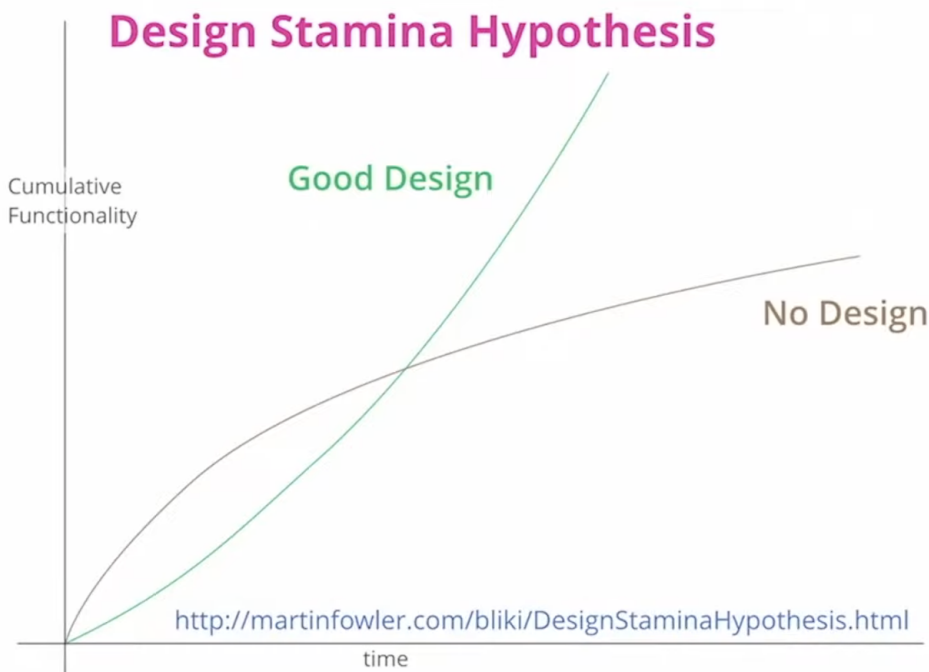
‘위키백과’에서 서술하고 있는 아키텍처의 정의 역시 수 많은 정의들 중 하나이다. 아마 여러분이 직접 구글링을 해본다면, 작성한 사람의 주관적 의견과 추상적 의미가 섞여있는 많은 문장들을 확인할 수 있을 것이다.

그렇다면 아키텍처의 정의에 정답이 있는 것일까? 그렇지 않다. 다만 시대에 따라 소프트웨어 개발에서 중요하게 생각해야 하는 것들의 우선순위는 계속해서 변화되어 왔고, 현재에 이르러서는 **경제학적 관념**과 어우러지기도 한다.

<https://www.youtube.com/watch?v=4E1BHTvhB7Y>

위 영상이 현 시대의 아키텍처의 정의를 가장 잘 설명하는 영상이라고 생각한다.

여기서 백마디 말로 설명하는 것 보다 위 영상을 보는 것이 보다 효율적일 것이다. 그러나 이 마저도 여의치 않을 분들을 위해 아주 간략하게 요약한다.



영상에게 이야기하는 아키텍처의 정의는, 이 한 장의 그림이 모두 담고 있다.

아키텍처는 소비자에게 보이지 않는 영역이다. 만약 동일한 기능의 소프트웨어가 있다면, **소비자는 아키텍처의 완성도와 관계없이 '값'이 싼 소프트웨어를 선택**하게 될 것이다.

하지만 그림을 보면 알 수 있듯이, 좋은 아키텍처는 '기능의 추가' 측면에서 훨씬 유리하다. 시장의 빠른 변화에 대응하기 위해서는 지속적인 업데이트가 필수적이다.

완성도 높은 아키텍처는 소프트웨어의 완성도 뿐만 아니라 **비즈니스**의 성패를 결정하는 중요 요인으로 동작하게 되는 것이다.

현재의 아키텍처의 정의가 **경제학적 관념**을 포괄하는 이유이다.

### **제시한 아키텍처 정의에 동의하는가?**

동의한다면 아키텍처의 설계가 여러분이 수행해야 하는 프로젝트에 얼마나 지대한 영향을 끼치게 될 것인지 예상이 될 것이다.

기획을 완료하고 기능 명세서를 작성한 다음, 프로젝트 구성원들이 중요하게 생각하는 것들을 떠올리며 아키텍처를 설계해보자.

아키텍처를 따라 개발을 수행하다 보면, **개발자로서의 철학**이 담겨있는 포트폴리오가 완성되어 있을 것이다.