

ILE3-024 소프트웨어 설계

소프트웨어를 개발하기 위한 설계 방법론

소프트웨어 설계

- "무엇을(What)"으로부터 "어떻게(How)"로 관점이 바뀌며 구현할 소프트웨어의 청사진을 만드는 것
- 개발을 수행할 기능들의 구체적인 구현 방법을 명시하는 단계
- 장치, 프로세스 그리고 시스템을 명확하고 자세하게 정의하여 실질적으로 실현가능한 관련 기술과 방법론을 적용 / 검토하는 과정
- 요구되는 기능과 성능 조건들을 만족하는 기능, 구조 및 동적 행위들을 모델링하여 표현, 분석, 검증하는 과정하고, 해당 과정에서의 산출물을 작성

설계 원칙

- 소프트웨어 설계는 변경이 용이하도록 구조화되어야 함
- 하나의 함수 안에는 특정 기능을 수행하는데 필요한 자료만을 사용하도록 규제할 것
- 독립적이고 기능적인 특성을 지닌 모듈단위로 분할 설계
- 계층적 구조를 가져야 함

모델링

구조 모델링

소프트웨어 내의 컴포넌트들 간의 상호 연결 구조를 표현

Example

인터페이스 정의, 클래스 구조도 작성

행위 모델링

컴포넌트들이 언제, 어떠한 순서로 기능을 수행하는지를 표현

Example

시퀀셜 다이어그램, 플로우차트

모듈화

- 독립적으로 처리할 수 있는 구별단위
- 모듈의 재사용을 통하여 빠르게 개발이 가능
- 장점
 - 복잡도 감소, 수정 용이, 구현 용이
 - 확장성, 융통성, 유지보수성, 재사용성, 경제성

모듈 설계 시 검토 내역

- 모듈화(Modularity)
 - 프로그램을 작고 독립적인 단위로 분할하여 개발하는 것
 - 변경에 의한 영향을 최소화

- 추상화(Abstraction)
 - 구체적인 데이터의 내부 구조를 외부에 알리지 않으면서 데이터를 사용하는 데 필요한 함수만을 알려주는 기법
- 정보 은닉(Information hiding)
 - 각 모듈의 자세한 처리 내용이 시스템의 다른 부분에게 감추어짐
- 단계적 분해(Stepwise refinement)
 - 상위 개념에서 좀더 상세화된 하위개념으로 구체화시키는 과정
- 구조화(Structure)
 - 시스템을 소프트웨어의 구성요소인 모듈의 계층적 구조형태로 표현
- 높은 응집력(cohesion)
 - 모듈 내부의 처리 요소들간의 기능적 연관도가 높음
- 낮은 결합력(coupling)
 - 모듈간의 상호 의존도가 낮음

개발 단계

요구사항 분석

1. 고객 요구사항 분석
 - 고객이 필요하는 기능을 정리/문서화
2. 기능 요구사항 작성
 - 고객 요구사항을 개발자가 구현해야 하는 항목을 작성
3. 비기능 요구사항 작성
 - 개발 이외의 달성 조건, 일반적으로 전체 시스템의 동작의 평가 지표를 작성 (ex. KPI)

설계

1. 모듈 설계
 - 요구사항 단계에서 산출된 요구사항을 만족하기 위한 기능 설계
2. 모듈 관계 및 구조 설계
 - 프로그램 내부에서의 모듈 간의 관계와 구조 설계
3. 전체 프로세스 설계
 - 프로그램 내의 각 모듈에서의 처리 절차나 알고리즘 설계

구현

1. 구현
 - 설계 단계에서 논리적으로 결정한 방법을 프로그래밍 언어를 사용하여 실제 프로그램을 개발하는 단계

테스트

작은 부분부터 큰 부분적으로 점진적으로 테스트
테스트 영역이 넓어 질수록 테스트 비용이 커짐

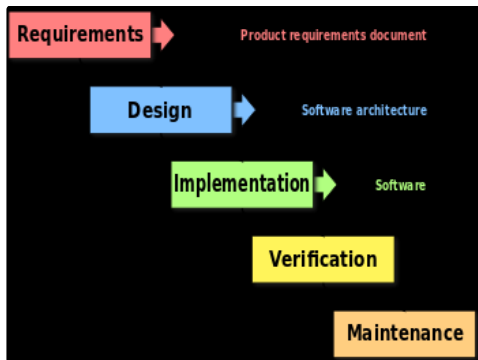
1. 단위 테스트
 - 개발된 모듈의 각 기능에 대한 테스트
2. 모듈 테스트
 - 모듈 단위의 기능 테스트
3. 통합 테스트
 - 전체 개발 시스템을 테스트

배포 [↗](#)

개발이 완료된 프로그램을 고객에게 전달

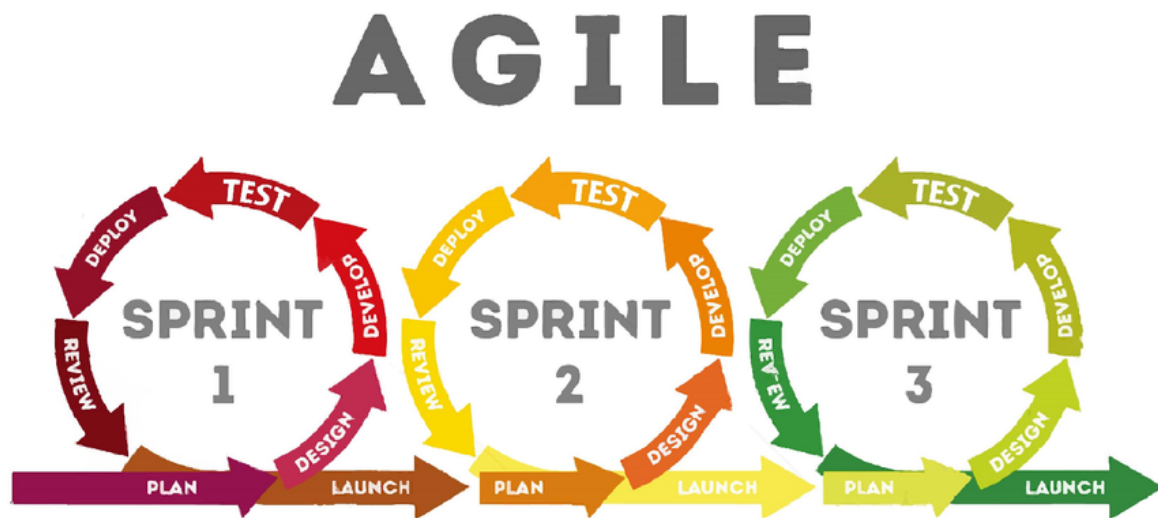
개발 모델 [↗](#)

폭포수 프로세스 [↗](#)



각 단계를 완전하게 완료한 이 후 다음 단계로 진행되는 모델

애자일 프로세스 [↗](#)



일정한 주기를 가지고 끊임없이 프로토타입을 만들어 내며, 매 주기마다 필요한 요구를 더하고 수정하여 진행하는 방식