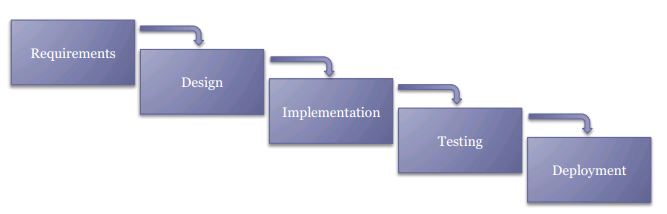
**소프트웨어 생명 주기 Software life cycle**

개발 -> 사용 -> 변경 -> 사용 -> 변경... (반복)

일단 SW(소프트웨어, 이하 SW) 가 개발되면 사용과 유지보수의 주기에 돌입하는데, 이는 그 sw의 생명이 남아있는 동안 계속 진행되는 주기이다.

이러한 패턴은 제조업체의 제품들에서도 일반적으로 나타나지만, 차이점은 일반 제품의 경우 유지보수 단계의 주요 작업이 수리인데 비해, SW의 경우 유지보수 단계의 주요 작업이 오류 수정이나 갱신이라는 점이다. 실제로 SW에 오류가 발견되거나, SW의 응용 분야의 변경으로 인해 SW에 변경이 필요하거나, 이전의 변경으로 인해 SW의 다른 부분에 문제가 생겼거나 등으로 인해 유지 보수 단계로 이행한다.

**소프트웨어 생명 주기의 개발 단계**



요구사항 명세 Requirements Specification  
  
그 목표는 제안 시스템에서 제공할 서비스들을 규정하는 것, 이러한 서비스들에 부과되는 시간 제약이나 보안 등등의 조건을 식별하는 것, 외부 세계와 시스템 사이의 상호작용을 정의하는 것.추후의 사용자나 기타 법률적, 재정적 이해 등의 다른 유대 관계를 가진 사람 등의 제안 시스템의 이해관계자들의 요구사항을 조사하는 일을 포함한다.

소프트웨어 개발의 방향이 될 확고한 목표를 정의해야 한다. "WHAT"

설계 Design

제안 시스템의 구축을 위한 계획을 만드는 일. 이 단계에서 소프트웨어 시스템의 구조가 정해진다.

요구사항 명세에서 해결해야 할 문제를 파악했다면, 설계는 문제의 해답을 구하는 것이라고 할 수 있다. "HOW"

설계 단계의 결과는 프로그램으로 변환될 수 있는 SW 시스템 구조에 대한 세부 기술이다.

소프트웨어 시스템이 아니라 업무용 빌딩을 짓는 프로젝트라면, 설계는 지정된 요구 사항들을 충족시키기 위한 세부 도면을 개발하는 단계이다. 예를 들면, 그러한 도면은 제안되는 빌딩을 ㅁ뵤사하는 여러 단계의 세부도면들의 청사진을 포함할 것이며, 때로는 제안된 구조의 3차원 축소 모형이 포함될 수도 있다. 실제 빌딩의 건축에 이용되는 것은 이러한 문서들이다. 이러한 도면을 개발하기 위한 기법들이 발전해 왔고, 이에는 표준화된 표기 체계와 수많은 다이어그램 작성 방법들도 포함된다.

마찬가지로, 소프트웨어 설계에서도 다이어그램과 모델링 작업이 중요한 역할을 한다. 그러나 SW엔지니어들이 사용하는 방법론이나 표기 체계는 건축 분야처럼 안정되어 있지는 않다. 공학상의 설계는 변화가 많으며 학자들은 여전히 나은 방법들을 모색하고 있다.

구현 implementation

프로그램을 실제로 작성하고, 데이터 파일을 만들고, 데이터베이스를 개발하는 작업들을 포함한다.

소프트웨어 분석가(시스템분석가) 와 프로그래머가 구별되는 것은 바로 이 구현 단계에서이다. 전자는 전체 개발 과정에 관련된 사람이며, 주로 요구사항 명세와 설계 단계를 담당한다. 프로그래머는 주로 구현 단계를 수행하는 사람을 말한다.

테스트 testing

과거의 전통적 개발 단계에서의 테스트는 기본적으로 프로그램의 오류를 찾아 고치고 최종 소프트웨어 제품이 SW의 요구사항 명세 문서에 합치되는지를 확인하는 과정이었다. 그러나 오늘날 테스트는 실제로 전체 개발 과정의 각 중간 단계 결과물에 대해 정확한지 '테스트' 해야한다. 즉, 품질 보증을 위한 전반적 노력의 한 부분으로 인식되어야 한다.

결과적으로 SW 개발 과정은 '요구사항 명세와 *확인*' 설계와 *검증*' '구현과 *테스트*' 등 세 개의 단계로 이루어져야 한다고 주장한다.

Retrieved from 컴퓨터과학 총론 edition 10 (홍릉과학출판사)