

1

## 소프트웨어 개발 프로세스

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 소프트웨어 개발의 목표

- 정해진 기한 내에, 주어진 예산을 이용해 사용자가 원하는 좋은 품질로 개발하는 것

### 계속되는 소프트웨어 프로젝트 실패의 원인

- 소프트웨어의 요구사항이 복잡해지고 규모가 점점 커짐
- 정해진 기간 내에 고품질의 소프트웨어를 개발하는 것이 점점 더 어려워짐

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 소프트웨어 개발 프로세스의 중요성


- 소프트웨어 제품의 품질은 그 제품을 만들기 위해 사용된 프로세스의 품질에 의해 결정됨  
[Watts S. Humphrey]

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

예시)

웹사이트 제작 프로세스

 관계 인력

- 고객 : 웹사이트 제작 의뢰
- PM(Project Manager) :  
전체 웹사이트 제작 관리 책임
- 웹기획팀 : 전체 개발 절차 및 일정 관리
- 웹디자인팀 : 전체 웹사이트 디자인 담당
- 웹개발팀 : 코딩을 통한 실제 웹 개발(구현)

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

 단계 1) 웹사이트 제작 의뢰서 작성

- 고객은 웹사이트 제작 의뢰서를 통해 요구사항을 작성함

번호	요구항목	주요기능	비고
1	사이트 소개	연혁 관리, 업무 관리, 인사말 관리	
2	교육 소개 관리	교육과정 소개 관리, 교육신청 절차 소개 관리, 교육신청서 다운받기, 문의 메일 보내기	

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

🔍 단계 1) 웹사이트 제작 의뢰서 작성

- 고객은 웹사이트 제작 의뢰서를 통해 요구사항을 작성함

번호	요구항목	주요기능	비고
3	컨퍼런스 관리	초대글 관리, 프로그램 소개 관리, 강연자 소개 관리, 컨퍼런스 등록절 차 안내관리, 온라인 등록, 사전등록, 행사장 약도 관리, 스폰서 소개 관리	

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

🔍 단계 1) 웹사이트 제작 의뢰서 작성

- 고객은 웹사이트 제작 의뢰서를 통해 요구사항을 작성함

번호	요구항목	주요기능	비고
4	공지사항 관리	교육 관련 공지사항 관리, 컨퍼런스 관련 공지사항 관리	게시판
5	회의문의 관리	회원 문의 응답, 삭제 관리	게시판

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

 단계 1) 웹사이트 제작 의뢰서 작성

- 고객은 웹사이트 제작 의뢰서를 통해 요구사항을 작성함

번호	요구항목	주요기능	비고
6	회원 관리	회원 가입/탈퇴, 회원 로그인, 회원 문의 글쓰기	



# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

#### 단계 2) 요구사항 분석서 작성

- 웹 기획팀, 웹 디자인 팀, 그리고 웹 개발팀은 고객으로부터 의뢰 받은 요구사항을 분석하여 고객의 의도를 파악

#### 단계 3) 인터뷰 실시

- 고객과 PM, 웹 기획팀은 요구사항에 대한 논의 및 협의를 위한 인터뷰 실시

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

#### 단계 4) 요구사항 정의서 작성

- 웹 기획팀은 고객의 요구사항과 인터뷰 결과를 기반으로 요구사항 정의서 작성

#### 단계 5) 요구사항 정의서 검토

- 고객과 PM은 작성이 완료된 요구사항 정의서를 함께 검토

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 웹사이트 제작 프로세스

#### 단계 6) 승인

- 고객은 요구사항 정의서를 바탕으로 수정사항이 있으면 PM과 웹기획팀과 함께 다시 논의하여 수정
- 만약 수정사항이 없으면 고객은 요구사항 정의서를 승인

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 요구사항 정의서 예제

요구사항명	상세요구사항	우선순위	비고
관리자 모드	사이트 수정 및 업데이트, 게시판 관리	1	관리자 사전 교육 필요
회원가입	회원정보 : 이름, 전화번호, 관심분야, 직업, 회사, 전자메일, 메일 수신 여부	1	중복회원가 입 불가

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 요구사항 정의서 예제

요구사항명	상세요구사항	우선순위	비고
회원관리	회원 가입/탈퇴 관리, 로그인 권한, 회원 문의 글쓰기 권한	1	
메뉴	- 회사 소개 - 교육 소개 - 컨퍼런스 - 공지사항 - 회원문의 게시판	1	메뉴별 서브 메뉴 필요

# 01 소프트웨어 개발 프로세스

## 1 소프트웨어 프로세스의 중요성

### 예) 요구사항 정의서 예제

요구사항명	상세요구사항	우선순위	비고
최신 공지사항	메인화면에는 공지사항에 가장 최신 글을 표시해야 함	2	
교육 및 컨퍼런스 공지	새로운 교육과 컨퍼런스 개최 시 메인 화면에 공지	1	

## 2 소프트웨어 개발 생명주기

## 02 소프트웨어 개발 생명주기



### 1 소프트웨어 개발 생명주기란?

- 🔍 소프트웨어를 어떻게 개발할 것인가에 대한 전체적인 큰 흐름을 나타내는 추상적 표현으로, 순차적 또는 방법적인 단계로 구성됨
- 🔍 생명주기는 개발 생명주기의 각 단계에는 관련된 여러 활동들이 정의되어 있고, 이 활동들을 통해 다음 단계에 활용될 수 있는 산출물이 만들어 짐



## 02 소프트웨어 개발 생명주기

### 1 소프트웨어 개발 생명주기란?

-  전체 프로젝트의 비용 산정과 개발 계획을 수립할 수 있는 기본 골격을 제시하고 참여자들 간에 의사소통의 기준과 용어의 표준화를 가능하게 함
-  각 단계별로 그 단계의 완성을 나타내는 산출물을 정의함으로써 문서화가 충실한 프로젝트 관리를 가능하게 함

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 주먹구구식 개발 모델(Build-Fix Model)

##### 개요

- 요구사항 분석, 설계 단계 없이 일단 개발에 들어간 후 만족할 때 까지 수정작업 수행

##### 적용 가능한 경우

- 크기가 매우 작은 규모의 소프트웨어 개발

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 주먹구구식 개발 모델(Build-Fix Model)

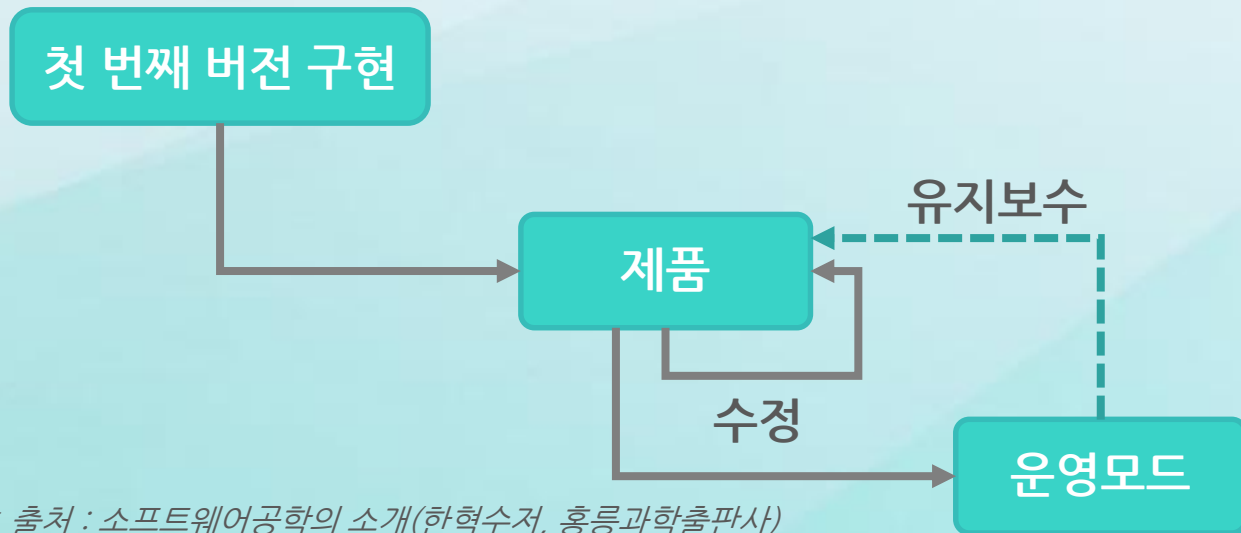
##### 단점

- 정해진 개발 순서가 없기 때문에 계획이 정확하지 않음
- 관리자는 프로젝트 진행 상황 파악에 어려움
- 개발 문서가 없기 때문에 개발 및 유지보수에 어려움

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 주먹구구식 개발 모델(Build-Fix Model)



※ 출처 : 소프트웨어공학의 소개(한혁수저, 홍릉과학출판사)

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

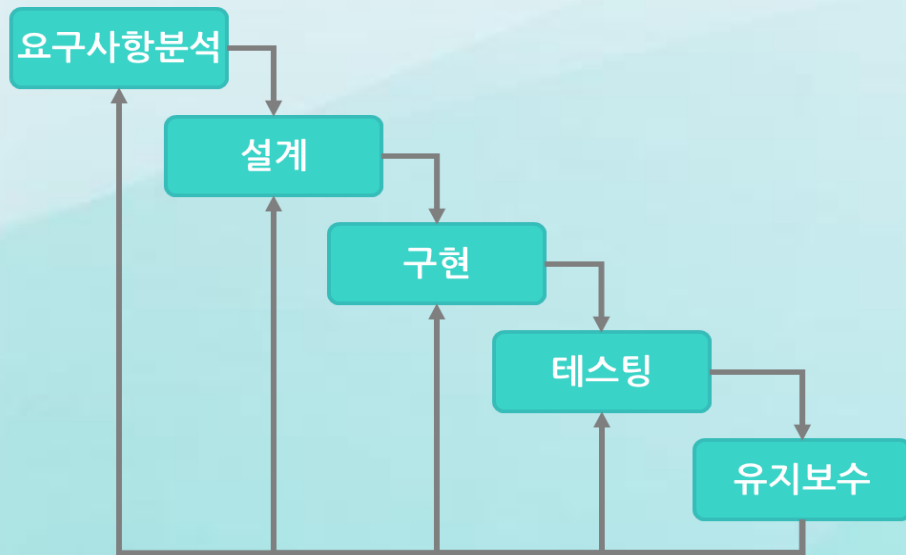
##### 개요

- 순차적으로 소프트웨어를 개발하는 전형적인 개발 모델
- 대부분의 소프트웨어 개발 프로젝트의 기본적 모델이며 가장 많이 사용되는 모델
- 소프트웨어 개발의 전 과정을 나누어 체계적이고 순차적으로 접근하는 방법
- 개발 과정  
: 요구사항 분석, 설계, 구현, 테스트, 유지보수

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)




※ 출처 : 소프트웨어공학  
의 소개(한혁수저, 홍릉과  
학출판사)

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

 단계 - 1단계 : 요구사항 분석

##### ① What의 단계

- 사용자 요구나 주어진 문제를 정확히 분석 이해하는 과정
- 구현 시스템의 기능, 목표, 제약사항 등을 정확히 파악
- 발주자와 개발자가 요구분석 사항에 동의하여야 다음 단계(설계)로 진입

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

🔍 단계 - 1단계 : 요구사항 분석

##### ② 요구사항 명세서(Requirement Specification)


- 요구 분석 단계의 결과물
- 사용자와 개발자의 의사소통 수단으로 사용



## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

 단계 - 2단계 : 설계


##### ① How의 단계

- 분석된 결과를 어떻게 프로그램으로 구성할 것인가
- 솔루션에 집중

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

 단계 - 2단계 : 설계

##### ② 설계 내용

시스템  
구조 설계

- 시스템을 구성하는 모듈의  
관계와 구조

프로그램 설계

- 각 모듈 내에서의 알고리즘을  
설계

사용자  
인터페이스 설계

- 사용자에게 보이는 부분을 설계

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

🔍 단계 - 2단계 : 설계

##### ③ 설계 명세서

- 설계 단계의 결과물
- 소프트웨어 구조를 기술한 것으로 모듈의 기능과 모듈 사이의 관계를 기술

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)


🔍 단계 - 3단계 : 구현

- 'Do it' 단계
- 이전 단계의 모듈 설계를 기준으로 프로그래밍을 함
- 결과물 : 소스 코드

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)


 단계 - 4단계 : 테스트

- 프로그램이 입력에 따라 요구되는 결과대로 작동하는지, 내부적 이상 여부 및 오류 발견을 위해 수행
- 테스트 계획을 세운 후 문서화

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

 단계 - 5단계 : 유지보수

- 개발된 소프트웨어의 변경사항을 수정하는 것
- 수정 유지보수, 적응 유지보수, 기능 추가 유지보수 등이 있음

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)



#### 장점

- 프로세스가 단순하여 초보자가 쉽게 적용 가능
- 중간 산출물이 명확, 관리하기 좋음
- 코드 생성 전 충분한 연구와 분석 단계

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)



##### 단점

- 앞 단계가 완료될 때까지 다음 단계들은 대기 상태여야 함
- 실제 작동되는 시스템을 개발 후반부에 확인 가능하기 때문에 고객이 요구사항 확인하는데 많은 시간이 걸림



## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 폭포수 모델(Waterfall Model)

##### 적용

- 단순하거나 이미 잘 알고 있는 문제에 적합
- 변화가 적은 프로젝트에 적합

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 개요

- 폭포수 모델의 단점을 보완한 모델
- 점진적으로 시스템을 개발해 나가는 접근 방법
- 원형(Prototype)을 만들어 고객과 사용자가 함께 평가한 후 개발 될 소프트웨어의 요구사항을 정제하여 보다 완전한 요구사항 명세서를 완성함

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 목적

- 소프트웨어 개발 초기에 고객의 요구사항을 완전히 파악하기 어려울 때 원형을 가능한 빨리 개발하여 고객과 검증하는 것

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

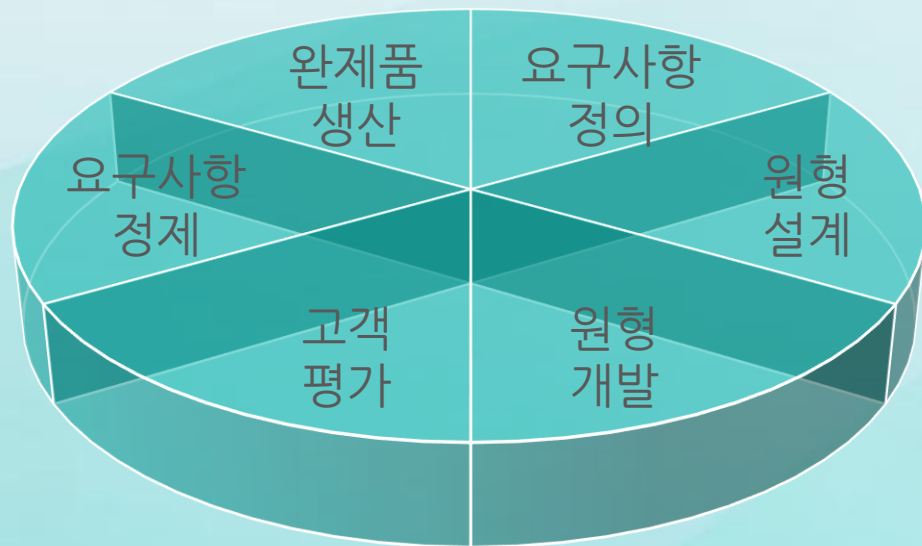
##### 방법

- 고객으로부터 피드백을 받은 후 원형을 폐기
- 시스템 기능 중 중요한 부분만 구현하여 피드백을 얻은 후 지속적으로 발전시켜 완제품을 제작

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)



## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단계

##### ① 1단계 : 요구사항 정의(Requirement Gathering and Refinement)

- 고객의 일부 요구사항 또는 불완전한 요구 사항으로부터 제품의 윤곽을 잡음

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단계

##### ② 2단계 : 원형 설계(Quick Design)

- 주어진 요구사항을 기반으로 빠른 설계를 함
- 주로 제품의 사용자 인터페이스에 초점을 맞춤

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단계

#### ③ 3단계 : 원형 개발(Building Prototype)

- 설계된 원형을 RAD(Rapid Application Development) 도구 등을 사용하여 빠르게 구현함
- 고객이 요구하는 기능을 구현하고 필요한 요소를 파악하는데 중점을 둠
- 프로그램의 신뢰도나 품질이 아니라 가능한 빨리 원형을 구현하는 것이 목적



## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단계

#### ④ 4단계 : 고객 평가(Customer Evaluation of Prototype)

- 고객과 개발자가 함께하는 가장 중요한 단계
- 고객 요구사항을 정확하게 규명하기 위해 원형에 대한 사용 및 평가시간을 충분히 제공
- 개발될 소프트웨어의 요구사항 정제에 중요한 정보로 활용

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단계

##### ⑤ 5단계 : 원형 정제(Refining Prototype)

- 원형이 어떻게 수정되어야 할지를 결정함
- 원형 개발과 검증, 요구사항 정제의 순환을 반복하여 추가적인 정보를 통해 요구사항을 완성해 나감

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 장점

- 사용자의 의견 반영이 잘 됨
- 사용자가 더 관심을 가지고 참여할 수 있고 개발자는 요구를 더 정확히 도출할 수 있음

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 단점

- 프로토타입이 최종 결과라고 믿고 소프트웨어 개발이 완성되리라는 사용자의 오해, 기대심리 유발
- 개발자 관점 :  
관리가 어려움(중간 산출물 정의가 난해)

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 원형 모델(Prototyping Model)

##### 적용

- 개발 착수 시점에 사용자 요구가 불투명할 때
- 실험적으로 실현 가능성을 타진해 보고 싶을 때
- 혁신적인 기술을 사용해 보고 싶을 때

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 개요

- 폭포수 모형과 원형 모형의 장점을 수용하고 위험 분석(Risk Analysis)을 추가한 점증적 개발 모델
- 프로젝트 수행 시 발생하는 위험을 관리하고 최소화 하려는 것이 목적

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

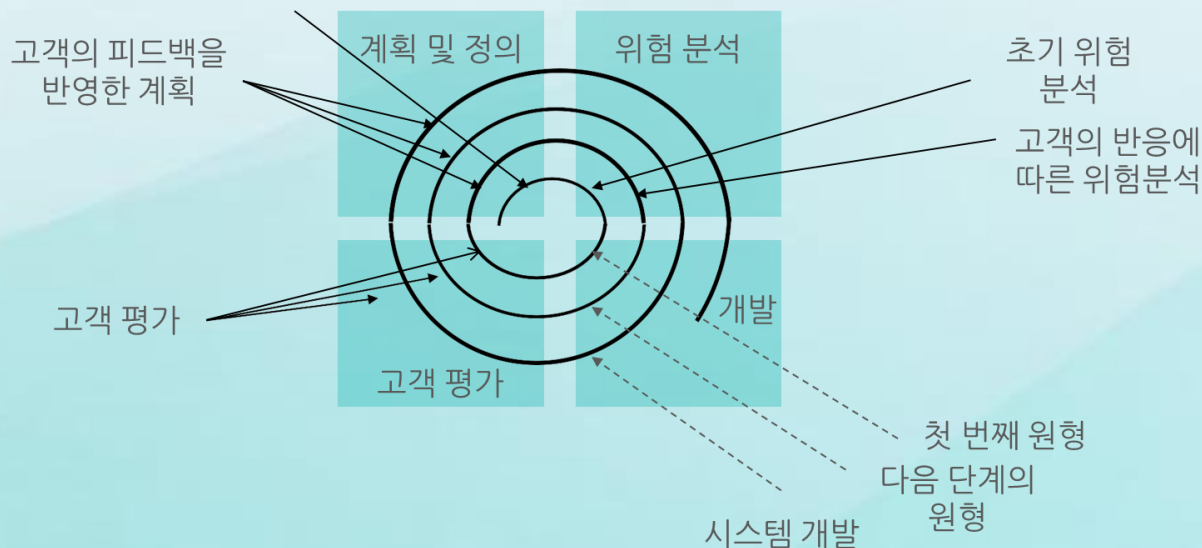
##### 특징

- 여러 개의 작업 영역으로 구분
- 나선상의 각 원은 소프트웨어 개발의 점증적 주기 표현
- 가장 안쪽 타원부터 개념적 개발 프로젝트, 실제 제품 개발 프로젝트, 제품 향상 프로젝트, 유지보 수 프로젝트
- 단계가 명확히 구분되지 않고, 엔지니어가 프로젝트 성격이나 진행상황에 따라 단계 구분

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)



※ 출처 : 소프트웨어공학의 소개(한석수저, 홍릉과학출판사)



## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 단계

##### ① 1단계 : 계획 및 정의 단계

- 개발자는 고객으로부터 요구사항을 수집
- 개발자는 시스템의 성능, 기능을 비롯한 시스템의 목표를 규명하고 제약 조건을 파악
- 목표와 제약 조건에 대한 여러 대안들을 고려하고 평가함으로써 프로젝트 위험의 원인을 규명 가능

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 단계

##### ② 2단계 : 위험 분석 단계

- 초기의 요구 사항을 토대로 위험 규명
- 위험에 대한 평가가 이루어지면 프로젝트를 계속 진행할 것인지 아니면 중단할 것인지를 결정

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 단계


##### ② 2단계 : 위험 분석 단계

- 초기의 요구 사항을 토대로 위험 규명
- 위험에 대한 평가가 이루어지면 프로젝트를 계속 진행할 것인지 아니면 중단할 것인지를 결정

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

 단계

#### ③ 3단계: 개발 단계

- 시스템에 대한 생명주기 모델을 선택하거나 원형 또는 최종적인 제품을 만드는 단계

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 단계


#### ④ 4단계: 고객 평가 단계

- 구현된 소프트웨어(시뮬레이션 모형, 원형 또는 실제 시스템)를 고객이나 사용자가 평가함
- 고객의 피드백을 얻는데 필요한 작업이 포함
- 다음 단계에서 고객의 평가를 반영할 수 있는 자료 획득 가능

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

 적용 가능한 경우

- 고비용의 시스템 개발
- 시간이 많이 소요되는 큰 시스템 구축 시 유용

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)



#### 장점

- 프로젝트의 모든 단계에서 기술적인 위험을 직접 고려할 수 있어 사전에 위험 감소 가능
- 테스트 비용이나 제품 개발 지연 등의 문제 해결 가능

## 02 소프트웨어 개발 생명주기

### 2 소프트웨어 개발 생명주기 모델의 종류

#### 나선형 모델(Spiral Model)

##### 단점

- 개발자가 정확하지 않은 위험 분석을 했을 경우 심각한 문제 발생
- 폭포수, 원형 모델에 비해 상대적으로 복잡하여 프로젝트 관리 자체가 어려울 수 있음