

소프트웨어공학 활용



소프트웨어 비용 산정



한국기술교육대학교
온라인평생교육원

학습내용

- 비용 구성요소
- 비용 산정기법

학습목표

- 프로젝트의 설계 비용 산정 시 고려 사항을 설명할 수 있다.
- 비용 산정기법을 이용하여 프로젝트 비용을 산정할 수 있다.

비용 구성요소



1 소프트웨어 비용 산정 개요

1 소프트웨어 비용 산정

1 개발 비용 산정의 정의



개발 비용 산정이란?

- 개발 소프트웨어 범위를 측정할 수 있는 소요공수, 투입인력, 개발 기간 등을 파악하여 **실행 가능한 계획을 수립**하는 것
- 소프트웨어의 개발 시에 정보 시스템의 구축에 필요한 기능과 규모를 근거로 하여 **직접 투입이 필요한 비용을 예측하는데 필요한 과학적, 합리적, 경험적인 활동**
- 발주자가 프로젝트 발주 시 예정 가격의 산정과 개발 업자가 수주한 개발 용역에 대하여 **적정한 대가를 산정할 수 있는 기준**
- 단위작업공수(비용)를 통한 총 공수(총비용) 산정

2 규모 산정



규모 산정이란?

- 소프트웨어 개발 프로젝트 계획 수립 시 소요 비용 및 기간의 산출
- 아웃소싱 계약 시 계약 근거

비용 구성요소



1 소프트웨어 비용 산정 개요

2 소프트웨어 비용 산정 필요성

- 1 소프트웨어 제품의 비용을 산정하는 과정에서 정확한 비용과 일정 계산에 오류가 많음
- 2 소프트웨어의 전체적인 규모파악을 통해서 사전에 비용을 분석하고 대응하기 위한 방안
- 3 발주자와 수주자 간 합리적이고 신뢰할 수 있는 수준의 비용을 계약하는데 필요한 방법
- 4 소프트웨어 개발의 생산성 측정 방안과 개발비용에 대한 근거제시로 저가 입찰 및 소프트웨어 품질 저하 지양

3 소프트웨어 비용 산정 관점

발주기관

- 소프트웨어 비용 산정방법 모호, 정보화 예산 편성의 어려움
- 합리적인 비용 산정기준을 정립, 프로젝트 저가 수주 근절 필요

수주기관

- 소프트웨어 비용 산정기준 표준 부재, 적정 비용 산정에 난항, 비현실적인 비용 산정 피해
- 발주자와 수주자 간 합리적인 비용 산정을 위한 표준 기준 필요

비용 구성요소



2 고려 요소

1 프로젝트 요소



어떤 소프트웨어를 개발할 것인가에 따라
비용이 달라짐

제품의 복잡도

프로젝트의 난이도,
유형

시스템 크기

입출력 양식의 수

시스템 신뢰도

정확성, 견고성,
완전성, 일관성

2 자원 요소

인적 자원

관리자, 개발자,
지원체계

하드웨어 자원

개발 장비,
운영 장비

소프트웨어 자원

개발 지원 도구

3 생산성 요소

개발자 능력

경험, 전문지식
습득 정도

개발 방법론

최신기법,
관리 방법론

비용 구성요소

3 결정 요소

1 프로그래머 자질

1

초대형 프로젝트에서 개별적인 능력의 차이는 평준화되는 경향이 있음

2

5명 이하의 프로그래머를 활용해야 하는 프로젝트인 경우에는 개인의 능력 차이는 매우 중요

3

초급 프로그래머와 고급 프로그래머의 생산성 차이가 큼

2 소프트웨어 복잡도

응용 프로그램

유틸리티 프로그램

시스템 프로그램

- 개발 난이도 비율 : 1
- 복잡도 등급 : 단순(Organic)
- 프로그래머 비용 공식(PM) : $PM = 2.4 \times (KDSI)^{1.05}$
- $TDEV = 2.5(PM)^{0.38}$

- * KDSI : Kilo Delivered Source Instruction
- * PM : Programmer – Month
- * LOC : Lines of code : 1 KDSI = 1,000 LOC
- * TDEV : Total Development Time

비용 구성요소

3 결정 요소

2 소프트웨어 복잡도

응용 프로그램

유틸리티 프로그램

시스템 프로그램

- 개발 난이도 비율 : 3
- 복잡도 등급 : 중간형(Semidetached)
- 프로그래머 비용 공식(PM) : $PM = 3.0 \times (KDSI)1.12$
- $TDEV = 2.5(PM)0.35$

- * KDSI : Kilo Delivered Source Instruction
- * PM : Programmer – Month
- * LOC : Lines of code : 1 KDSI = 1,000 LOC
- * TDEV : Total Development Time

응용 프로그램

유틸리티 프로그램

시스템 프로그램

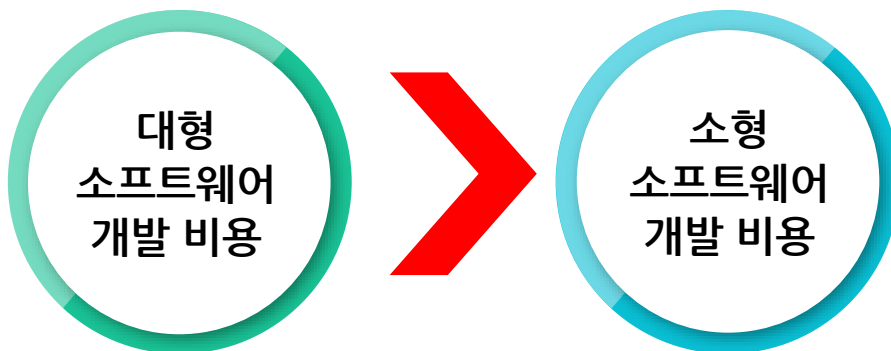
- 개발 난이도 비율 : 9
- 복잡도 등급 : 임베디드(Embedded)
- 프로그래머 비용 공식(PM) : $PM = 3.6 \times (KDSI)1.20$
- $TDEV = 2.5(PM)0.32$

- * KDSI : Kilo Delivered Source Instruction
- * PM : Programmer – Month
- * LOC : Lines of code : 1 KDSI = 1,000 LOC
- * TDEV : Total Development Time

비용 구성요소

3 결정 요소

3 소프트웨어의 크기



4 가용 시간

개발 기간을 단축하기 위해 인력과 자원을 늘려도
개발 기간을 무한정 단축할 수 없음

소프트웨어 프로젝트는 개발 기간이 최적의 기간보다 단축되거나
늘어난다면 더욱 많은 노력이 필요



보험

정상적인 계획에서 최대 75%가
줄일 수 있는 한계

비용 구성요소

3 결정 요소

5 요구되는 신뢰도의 수준



소프트웨어
신뢰도

프로그램이 일정한 기간 내에 주어진
조건으로 필요한 기능을 실행하는 확률



신뢰도는 원시코드의 정확성, 강인성, 완전성,
일관성에 관하여 표시할 수 있음

6 기술 수준



어셈블리
언어 사용

프로그래머의
생산성 :
5배 이상 상승



고급언어 사용

비용 산정기법



1 상향식 산정

1 상향식 산정 방법의 정의



상향식 산정 방법이란?

- 경험과 전문 지식이 많은 개발자가 인력, 시스템 크기, 필요 예산 등을 합하여 결정
- 프로젝트 세부 작업 단위별로 비용을 산정한 후 전체 비용을 합산하여 산정
- 측정하기 쉽고 이해하기 쉬움, 기존의 많은 프로젝트 측정모델들을 주요 입력으로 LOC(Lines Of Code)를 이용

2 원시코드 라인 수(LOC : Lines Of Code) 기법

1 정의 및 개념



원시코드 라인 수의 기법이란?

원시 코드 라인 수의 비관치, 낙관치, 중간치, 기대치를 측정한 후 예측치를 구해 비용 산정



라인 수 혹은 개발별 M/M(Man/Month, 인월)를
예측하여 이를 비용으로 변환

비용 산정기법



1 상향식 산정

2 원시코드 라인 수(LOC : Lines Of Code) 기법

2 산정 방법

비관치

가장 많이 측정된 코드 라인의 수(가중치 1부여)

낙관치

가장 적게 측정된 코드 라인 수(가중치 1부여)

기대치(중간치)

측정된 모든 코드 라인 수의 평균(가중치 4부여)

예측치

$\{(\text{낙관치}) + 4 \times (\text{기대치}) + (\text{비관치})\} / 6$

3 산정 공식

인월(m/m)

$= \text{개발 기간} \times \text{투입 인원} = \text{LOC} / \text{1인당 월평균 생산코드 라인 수}$

$\text{개발 비용} = \text{인월(m/m)} \times \text{단위 비용(1인당 월평균 인건비)}$

$\text{개발 기간} = \text{인월(m/m)} / \text{투입 인원}$

$\text{생산성} = \text{LOC} / \text{인월(m/m)}$

비용 산정기법



1 상향식 산정

2 원시코드 라인 수(LOC : Lines Of Code) 기법

4 산정 예

예시 1

개발 기간은 2년(24개월)이며, 10명의 개발자가 24개월, 14명의 개발자가 10개월 동안 참여한다. 소프트웨어의 인월(m/m)은?

풀이

인월 : $(10\text{명} \times 24\text{개월}) + (14\text{명} \times 10\text{개월})$
 $= 240\text{m/m} + 140\text{m/m} = 380\text{m/m}$

예시 2

LOC 기법에 의해 예측한 총 라인이 30,000라인이며, 개발자가 6명 참여하고 개발자들이 월 평균 30 라인을 코딩한다. 개발 기간은?

풀이

인월 : $\text{LOC}(\text{원시코드 라인 수}) / 1\text{인당 월평균 생산 코드 라인 수}$
 $= 30,000\text{라인} / 30\text{라인} = 100\text{m/m}$
 개발 기간 : $\text{인월(m/m)} / \text{참여 인원}$
 $= 100(\text{m/m}) / 10\text{명} = 10\text{개월}$

비용 산정기법



1 상향식 산정

3 개발 단계별 인월수(Effort Per Task) 기법



개발 단계별
인월수
기법이란?

LOC 기법을 보완하기 위하여, 각 기능 구현시키는데 필요한 노력(인월, m/m)을 생명 주기에 단계별 산정



LOC 기법보다 **정확**

비용 산정기법



2 하향식 산정

1 하향식 산정 방법



하향식 산정 방법이란?

- 업무 분류 구조로 정의
- 각 구성요소에 대한 산정을 독립적으로 실시한 후 이를 집계하여 산정
- 프로젝트 전체 비용을 산정한 후 각 작업별로 비용을 세분화함

2 전문가 감정기법



전문가 감정기법이란?

조직 내에 경험이 많은 두 명 이상의 전문가에게 비용 산정을 의뢰하는 기법

1

가장 편리하고 신속하게 비용 산정 가능, 의뢰자로부터 믿음을 얻을 수 있음

2

새로운 프로젝트에는 과거의 프로젝트와 다른 요소들이 있다는 것을 간과할 수 있음

3

새로운 프로젝트와 유사한 프로젝트에 대한 경험이 없을 수 있음

4


개인적이고 주관적일 수 있음

비용 산정기법

2 하향식 산정

3 델파이(Delphi) 기법

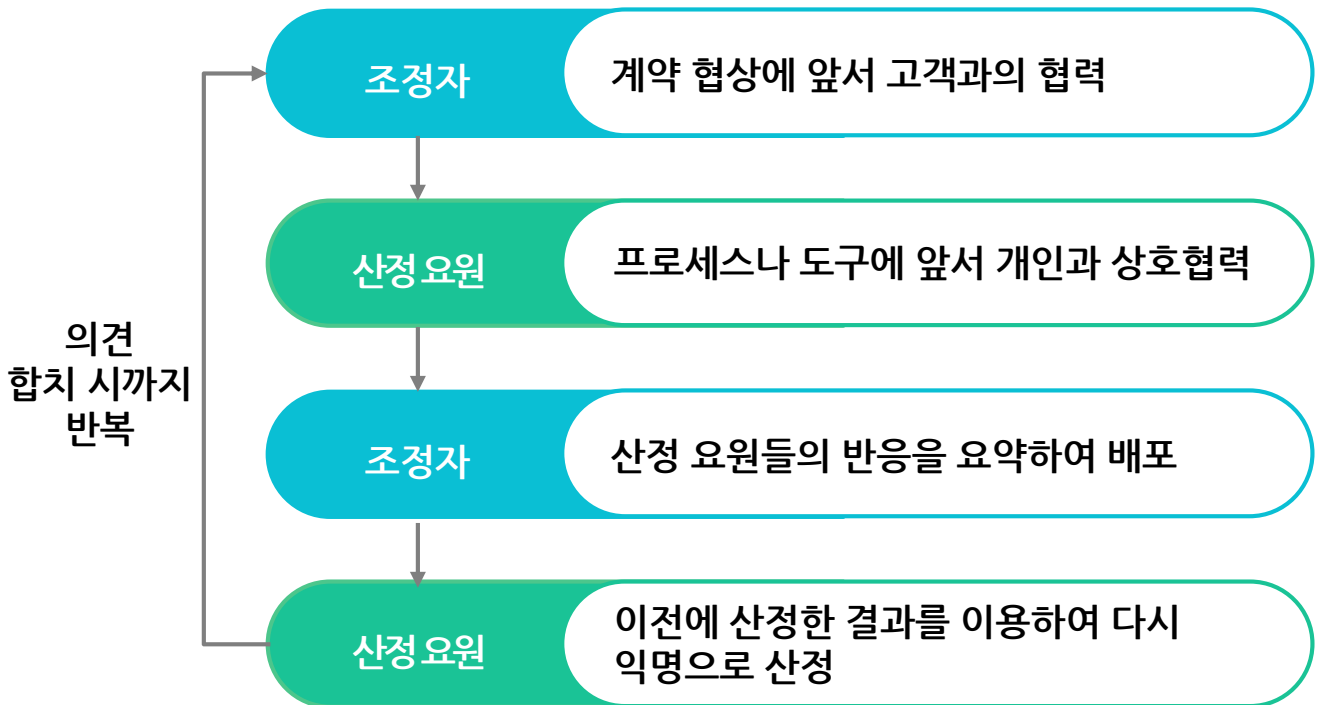
1 정의 및 개념



델파이(Delphi) 기법이란?

- 전문가 감정기법의 주관적인 편견을 보완하기 위해 많은 **전문가의 의견을 종합하여 산정**하는 기법
- 전문가 합의법

2 비용 산정 순서 및 방법

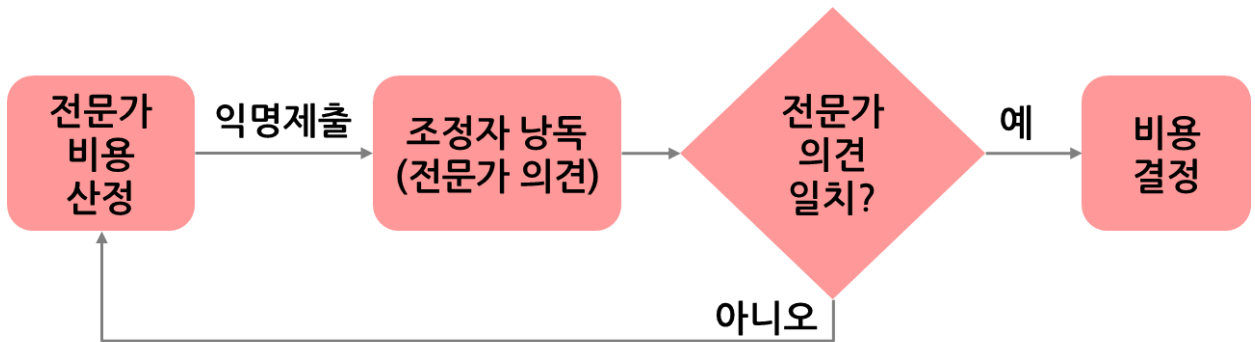


비용 산정기법

2 하향식 산정

3 델파이(Delphi) 기법

2 비용 산정 순서 및 방법



비용 산정기법

3 수학적 산정

1 수학적 산정기법



수학적 산정기법이란?

- 상향식 비용 산정기법 중 한 가지
- 경험적 추정기법 또는 실험적 추정기법이라고 함



목표 : 개발 비용 산정의 자동화

2 COCOMO(CONstructive COst MOdel) 방법

1 정의 및 특징

1

보헤미가 제안한 것으로 원시 프로그램의 규모인 **LOC에 의한 비용 산정기법**

2

소프트웨어 규모(LOC) 예측한 후 소프트웨어 종류에 따라 각 비용 산정 공식에 대입하여 비용 산정

3

비용 건적의 강도 분석 및 비용 건적의 유연성이 높아 **소프트웨어 개발비 건적에 널리 통용되고 있음**

4

같은 규모의 프로그램이라도 **그 성격에 따라 비용이 다르게 산정됨**

5

비용 산정 결과는 **프로젝트를 완성하는데 필요한 인월(m/m)**로 나타남

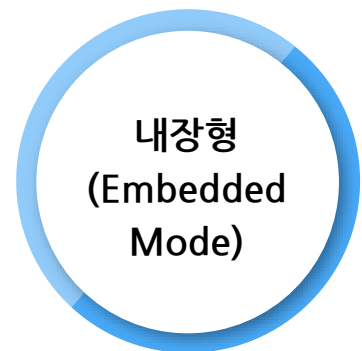
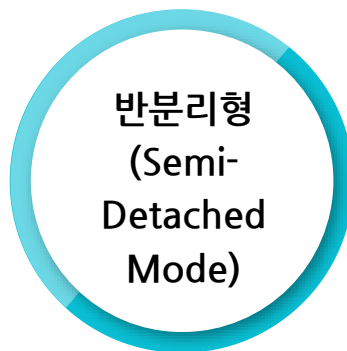
비용 산정기법

3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

2 소프트웨어 개발 유형

소프트웨어 복잡도 혹은 원시 프로그램 규모에 따라 분류



* KDSI(Kilo Delivered Source Instruction) :
전체 라인 수를 1,000라인 단위로 묶은 것으로 KLOC(Kilo LOC)와 같은 의미

조직형
(Organic Mode)

반분리형
(Semi-Detached Mode)

내장형
(Embedded Mode)

- 기관 내부에서 개발된 중소규모의 소프트웨어
- 일괄 자료 처리, 과학 기술 계산용, 비즈니스 자료 처리용
- 50KLOC(=5KDSI) 이하의 소프트웨어
- 사무 철용, 업무용, 과학용 응용 소프트웨어 개발에 적합
- 비용 산정 공식
- 인월(m/m) = $2.4 \times (\text{KDSI})^{1.05}$
- 개발 기간(TDEV) = $2.5 \times (\text{m/m})^{0.38}$

비용 산정기법

3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

2 소프트웨어 개발 유형

조직형
(Organic Mode)

반분리형
(Semi-Detached Mode)

내장형
(Embedded Mode)

- 트랜잭션 처리 시스템, 운영체제, 데이터베이스 관리 시스템 등
- 300KLOC 이하의 소프트웨어
- 컴파일러, 인터프리터와 같은 유틸리티 개발에 적합
- 비용 산정 공식
 - 인월(m/m) = $3.0 \times (KDSI)^{1.12}$
 - 개발 기간(TDEV) = $2.5 \times (m/m)^{0.35}$

조직형
(Organic Mode)

반분리형
(Semi-Detached Mode)

내장형
(Embedded Mode)

- 최대형 규모의 트랜잭션 처리 시스템, 운영체제
- 300KLOC 이상의 소프트웨어
- 신호기 제어 시스템, 미사일 유도 시스템, 실시간 처리 시스템 등의 시스템 프로그램 개발에 적합
- 비용 산정 공식
 - 인월(m/m) = $3.6 \times (KDSI)^{1.20}$
 - 개발 기간(TDEV) = $2.5 \times (m/m)^{0.32}$

비용 산정기법

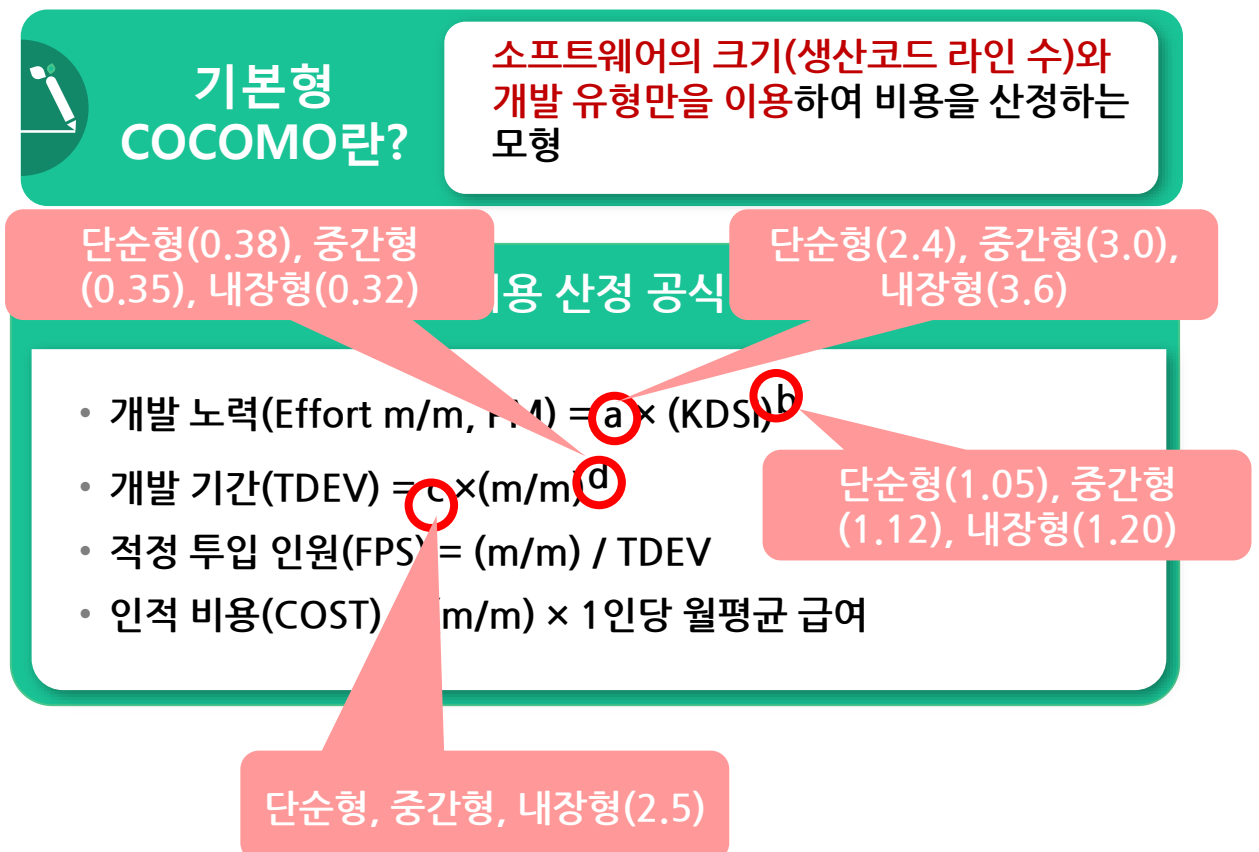
3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

3 COCOMO 모형의 종류



■ 기본형(Basic) COCOMO



비용 산정기법




3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

3 COCOMO 모형의 종류

▪ 중간형(Intermediate) COCOMO



중간형
COCOMO란?

기본형 COCOMO의 공식을 토대로
사용하나 **4가지 특성**의 **15가지 요인**에
의해 비용을 산정하는 모형

제품의 특성

- 요구되는 신뢰도
(0.75~1040)
- 데이터베이스 크기
(0.94~1016)
- 제품의 복잡도(0.7~1065)

컴퓨터의 특성

- 수행시간의
제한(1.00~1.66)
- 기업장소의
제한(1.00~1.56)
- 가상 기계의 안정성
(0.87~1.30)
- Turn Around Time
(0.87~1.15)

비용 산정기법

3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

3 COCOMO 모형의 종류

▪ 중간형(Intermediate) COCOMO

개발 요원의 특성

- 분석가의 능력(1.46~0.71)
- 개발 분야의 경험 (1.29~0.82)
- 가상기계의 경험 (1.21~0.90)
- 프로그래머의 능력 (1.42~0.70)
- 프로그래밍 언어의 경험 (1.14~0.95)

프로젝트 특성

- 소프트웨어 도구의 이용 (1.24~0.83)
- 프로젝트 개발 일정 (1.23~1.10)
- 최신 프로그래밍 기법 이용 (1.24~0.82)

비용 산정 공식

- 개발 노력(m/m) = (기본COCOMO의 m/m) × 요인별 노력 승수
- 개발 기간($TDEV$) = $C \times (m/m)^d$
- 적정 투입 인원(FPS) = $(m/m) / TDEV$
- 인적 비용($COST$) = $m/m \times 1\text{인당 월평균 급여}$

단순형(0.38), 중간형(0.35), 내장형(0.32)

단순형, 중간형, 내장형(2.5)


비용 산정기법

3 수학적 산정

2 COCOMO(CONstructive COSt MOdel) 방법

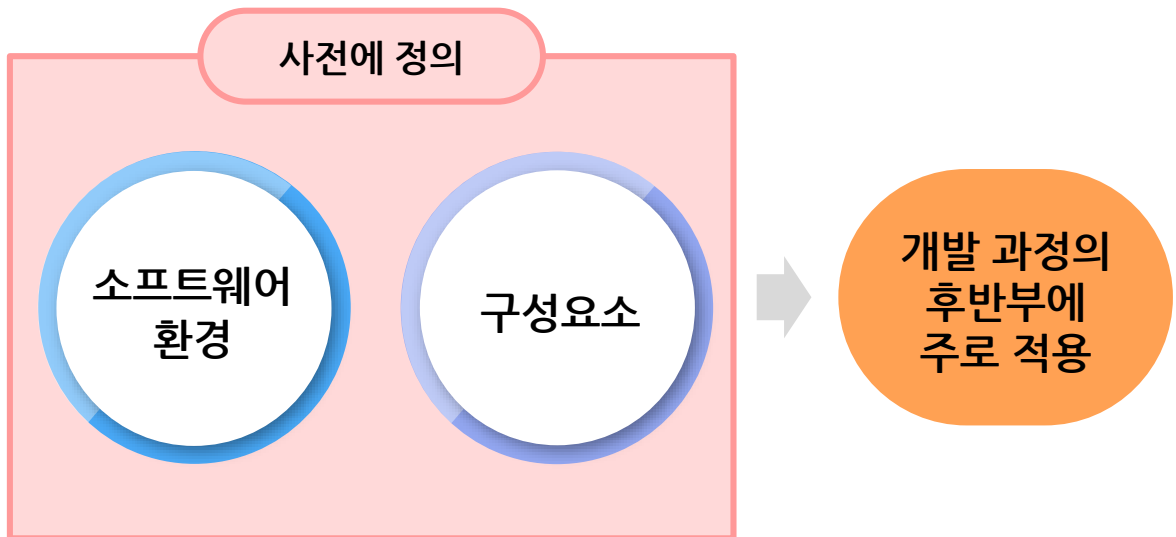
3 COCOMO 모형의 종류

▪ 발전형(Detailed) COCOMO



발전형
COCOMO란?

- 발전형 COCOMO는 중간 (Intermediate)형 COCOMO를 보완하여 만들어진 방법
- **개발 공정별로 보다 자세하고 정확하게 노력을 산출하여 비용을 산정하는 모형**



비용 산정기법



3 수학적 산정

2 COCOMO(ConStructive COst MOdel) 방법

3 COCOMO 모형의 종류

▪ 발전형(Detailed) COCOMO

비용 산정 공식

[중간형 COCOMO 산정 공식을 그대로 사용]

- 개발 노력(m/m) = (기본COCOMO의 m/m) × 요인별 노력 승수
- 개발 기간($TDEV$) = $c \times (m/m)^d$
- 적정 투입 인원(FPS) = $(m/m) / TDEV$
- 인적 비용($COST$) = $(m/m) \times 1\text{인당 월평균 급여}$

[노력 승수는 다음과 같이 적용하여 산정]

- 노력 승수 = 개발 공정별 노력 승수 × 개발 공정별 가중치

비용 산정기법

3 수학적 산정

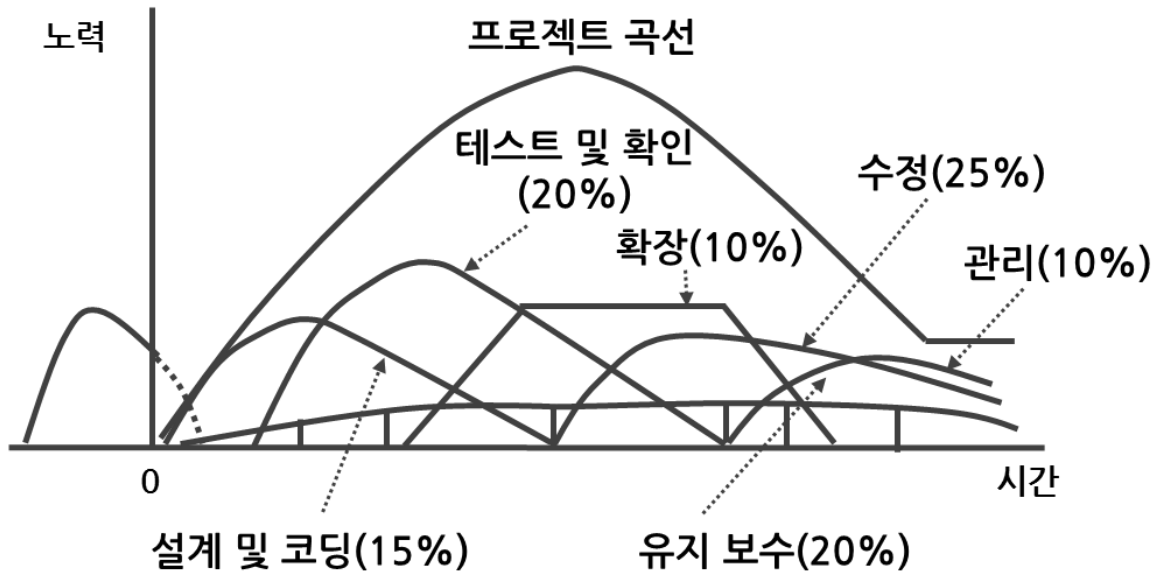
3 Putnam 모형

1 정의 및 특징

소프트웨어 생명 주기의 전 과정 동안에 사용될 인월(m/m)의 분포를 가정해 주는 모형

푸트남(Putnam)이 제안한 것으로 생명 주기에측모형이라고도 함

시간에 따른 함수로 표현되는 Rayleigh-Norden 곡선의 분포도를 기초로 함



비용 산정기법



3 수학적 산정

3 Putnam 모형

1 정의 및 특징

대형 프로젝트의 인월(m/m) 분포 산정에 이용되는 기법

개발 기간이 늘어날수록 프로젝트 적용 인원의 인월(m/m)의 노력이 감소함

2 비용 산정 공식

비용 산정 공식

- 개발 인월(m/m) = $L^3 / (C_k^3 * Td^4)$

* L : 원시 코드 라인 수

Td : 개발 시간

C_k : 환경 상수

(빈약 환경 = 2,000, 좋은 환경 = 8,000, 최적 환경 = 12,000)

비용 산정기법



3 수학적 산정

4 기능 점수(Functional Point) 모형

1 정의 및 특징

Albrecht가 제안



비용 산정기법 가운데 **최선의 평가**를 받음

비용 산정기법



3 수학적 산정

4 기능 점수(Functional Point) 모형

2 비용 산정 공식

비용 산정 공식

- 기능점수(FP) = 총 기능점수 × {0.65 + (0.1 × 총 영향도)}

3 기능별 가중치

자료입력(입력양식)	단순(3), 보통(4), 복잡(6)
정보 출력(출력보고서)	단순(4), 보통(5), 복잡(7)
명령어(사용자 질의 수)	단순(3), 보통(4), 복잡(6)
데이터 파일	단순(7), 보통(10), 복잡(15)
필요한 외부 루틴과의 인터페이스	단순(5), 보통(7), 복잡(10)

학습정리

1. 비용 구성요소



- 개발 비용 산정은 단위작업 공수(비용)를 통한 총 공수(총비용)를 산정하며 발주자가 프로젝트 발주 시 예정 가격의 산정과 개발업자가 수주한 개발 용역에 대하여 적정한 대가를 산정할 수 있는 기준임
- 규모 산정이란 소프트웨어 개발 프로젝트 계획 수립 시 소요 비용 및 기간을 산출하며 아웃소싱 즉 계약의 근거가 됨
- 소프트웨어 비용 결정 요소
 - 프로그래머의 자질
 - 소프트웨어 복잡도
 - 소프트웨어 크기
 - 가용 시간
 - 요구되는 신뢰도의 수준
 - 기술 수

학습정리

2. 비용 산정기법



- 비용 산정기법은 상향식 산정기법, 하향식 산정기법, 수학적 산정 방법으로 나눌 수 있음
- 상향식 산정 방법은 측정하기 쉽고 이해하기 쉽기 때문에 기존의 많은 프로젝트 측정모델들의 주요 입력으로 원시코드 라인 수 (LOC) 기법이 있음
- 하향식 산정방법은 업무 분류 구조로 정의가 되며 각 구성요소에 대한 산정을 독립적으로 실시한 후 이를 집계하여 산정함
 - 프로젝트 전체 비용을 산정한 후 작업별로 비용을 세분화함
 - 하향식 산정 방법으로는 전문가 감정기법과 델파이 기법이 있음
 - 전문가 감정기법은 조직 내에 경험이 많은 두 명 이상의 전문가에게 비용 산정을 의뢰하는 기법임
 - 델파이 기법은 전문가 감정기법의 주관적인 편견을 보완하기 위해 더 많은 전문가의 의견을 종합하여 산정함
- 수학적 산정기법은 상향식 비용 산정기법 중 한 가지로 경험적 추정기법 또는 실험적 추정기법이라고 함
 - COCOMO 방법은 소프트웨어 규모(LOC) 예측한 후 소프트웨어 종류에 따라 각 비용 산정 공식에 대입하여 비용 산정함
 - Putnam 모형은 대형 프로젝트의 인월(m/m) 분포 산정에 이용되는 기법임
 - 기능점수 모형(FP)은 소프트웨어 기능을 증대시키는 요인별로 가중치를 부여하고, 요인별 가중치를 합산하여 총 기능 점수를 산출하여 총 기능 점수와 영향도를 이용하여 기능점수를 구한 후 이를 이용하여 비용을 산정하는 기법임