

3주 2강

수학적 산정 기법



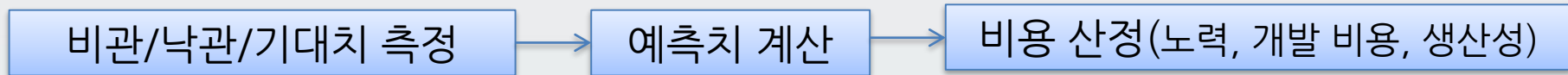
이번 주차에는...

소프트웨어 개발 계획

- 비용 산정 기법3 : 수학적 산정 기법

1. 원시 코드 라인수 기법

▪ LOC 기법



- 낙관적 : 한 모듈의 라인 수를 가장 적게 생각할 때의 예상 라인 수(가중치 1 부여)
- 비관적 : 한 모듈의 라인 수를 가장 많이 생각할 때의 예상 라인 수(가중치 4 부여)
- 보통 : 한 모듈의 라인 수를 보통이라고 생각할 때의 예상 라인 수(가중치 1 부여)
- 추정 LOC : $(\text{낙관치} + (4 \times \text{중간치}) + \text{비관치}) / 6$

2. 개별 단계 노력(M/M) 기법

- LOC
 - 개발하려는 소프트웨어의 총 코드 라인 수를 예측하여 구현 단계에 대한 M /M을 산정
- 실제 소프트웨어 개발
 - 코딩뿐 아니라 요구 분석, 설계 등의 단계에서도 인력과 자원이 많이 필요



LOC 보완

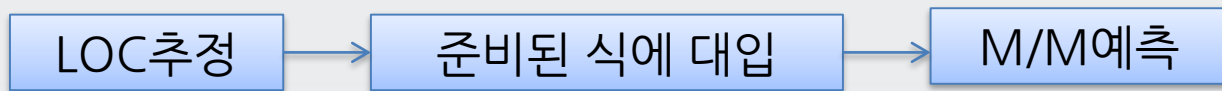
- 개발 단계별 노력 기법
 - M/M을 소프트웨어 개발 생명 주기의 각 단계에 적용하여 단계별로 산정
 - 장점: 코딩만 대상으로 산정하는 LOC보다 더 정확

3. 비용 산정 기법 3 : 수학적 산정 기법

- 상향식 비용 산정 기법
- 경험적 추정 기법 또는 실험적 추정 기법
- ① COCOMO 방법
 - SW 규모(LOC) 예측한 후 SW 종류에 따라 각 비용 산정 공식에 대입하여 비용 산정
- ② Putnam 방법
 - 소프트웨어 생명주기의 전 과정에 사용될 노력의 분포를 **가정해 주는**
- ③ 기능 점수(FP) 방법
 - 기능 점수를 구한 후 이를 이용해 비용 산정

4. COCOMO 방법(1)

- COCOMO 방법
 - 라인수 중심의 개발비 산정



- 개발비 산정 시 고려 사항
 - 프로그램 유형(난이도)에 따른 가중치

표 3-1 투입 인력 산출 공식과 프로젝트 유형에 따른 상수 a, b 값

$$PM = a \times (KDSI)^b$$

상수	유형별 값
a	단순형(2.4), 중간형(3.0), 내장형(3.6)
b	단순형(1.05), 중간형(1.12), 내장형(1.20)

5. COCOMO 방법(2)

■ 네 가지 특성에 따른 15가지 분류

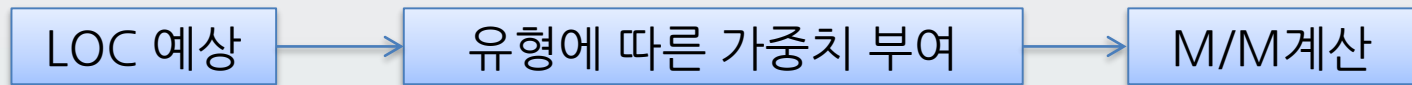
표 3-3 COCOMO 방법에서 사용되는 노력 승수 값

특성	비용 승수 요소	승수 값					
		매우 낮음	낮음	정상	높음	매우 높음	극히 높음
제품 특성	요구되는 신뢰도	0.75	0.88	1.00	1.15	1.40	
	데이터베이스 크기	0.94	1.00	1.08	1.16		
	제품의 복잡도	0.70	0.85	1.00	1.15	1.30	1.65
컴퓨터 특성	실행 시간 제약			1.00	1.11	1.30	1.66
	주기억 장치의 제약			1.00	1.06	1.21	1.56
	HW/SW의 안정성		0.87	1.00	1.15	1.30	
	처리 시간		0.87	1.00	1.07	1.15	
개발자 특성	분석가의 능력	1.46	1.19	1.00	0.86	0.71	
	응용 분야 경험	1.29	1.13	1.00	0.91	0.82	
	컴퓨터와 친숙성	1.21	1.10	1.00	0.90	-	
	프로그래머 능력	1.42	1.17	1.00	0.86	0.70	
	프로그램 언어의 경험	1.14	1.07	1.00	0.95		
프로젝트 특성	소프트웨어 공학 기술 사용	1.24	1.10	1.00	0.91	0.82	
	소프트웨어 도구의 사용	1.24	1.10	1.00	0.91	0.83	
	요구되는 개발 일정	1.23	1.08	1.00	1.04	1.10	

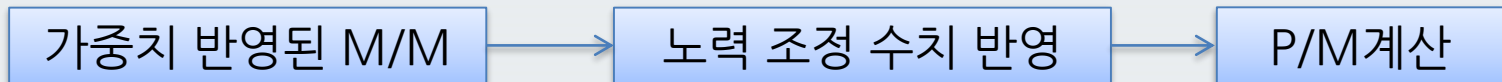
6. COCOMO 방법을 이용한 총 개발 기간 산정 과정

▪ ① 가중치 반영하기

- 프로그램의 규모 LOC에 가중치가 반영된 초기 개발 인건비 M/M 산정



▪ ② 보정 계수 반영 하기



▪ ③ 총 개발 기간 산정하기



7. 프로젝트 유형

- 단순형 프로젝트
 - 복잡도와 난이도가 비교적 높지 않은 업무용 소프트웨어
 - 크기는 중소 규모 정도, 크기는 50KDSI 이하
- 중간형 프로젝트
 - 규모나 복잡도가 중간급 정도, 크기 300KDSI 이하
 - 운영체제, 데이터베이스 관리 프로그램
- 내장형 프로젝트
 - 자동화 기기, 전자 제품과 같은 하드웨어와 밀접하게 관련 있는 소프트웨어
 - 크기 300KDSI 이상
-

8. 가중치 반영하기(1)

■ 개발 인건비의 초기 예측 값 산출 방법

표 3-2 프로젝트 유형에 따른 투입 인력 산출 공식

프로젝트 유형	공식
단순형	$PM = 2.4 \times (KDSI)^{1.05}$
중간형	$PM = 3.0 \times (KDSI)^{1.12}$
내장형	$PM = 3.6 \times (KDSI)^{1.20}$

- $PM^{\text{Person/Month}}$: 소프트웨어 개발에 필요한 인력(인원/월)
- $KDSI^{\text{Kilo Delivered Source Instruction}}$: 소프트웨어의 최종 원시 코드 라인 수

9. 가중치 반영하기(2)

- 유형별 M/M

- 단순형 < 중간형 < 내장형

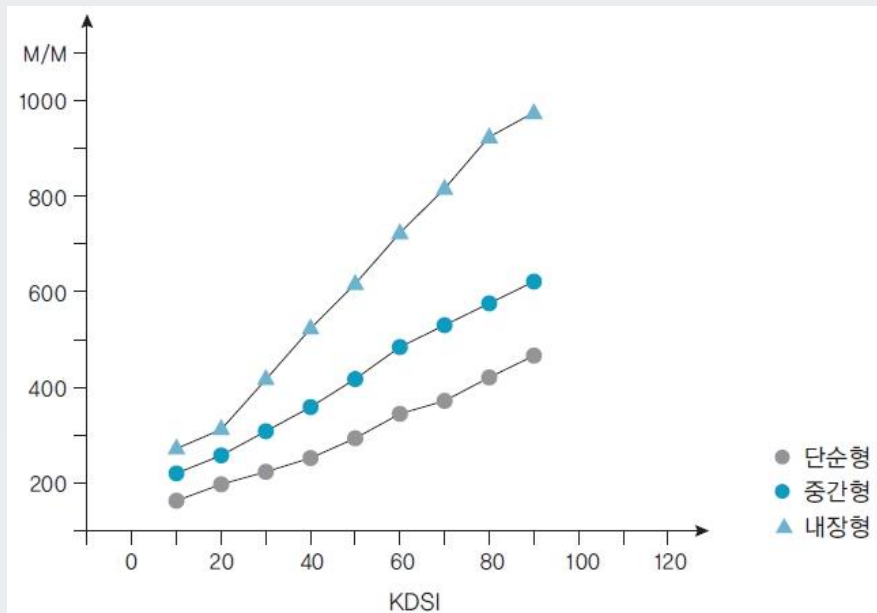


그림 3-4 COCOMO 방법에 의한 비용 예측

10. 보정 계수 반영하기(1)

- 노력 조정 수치 EAF: Effort Adjustment Factor

- 보정에 사용되는 값

노력조정 수치^{EAF} = 필요한 각 항목에 승수 값을 모두 곱한 값

개발하려는 소프트웨어에 요구되는 신뢰도가 높고(1.15), 매우 복잡하며(1.30), 소프트웨어공학 기술을 거의 사용하지 않고(1.24), 요구되는 개발 일정이 매우 촉박하고(1.10), 다른 요소들은 보통(1.00)일 경우의 노력 조정 수치?

$$EAF = 1.15 \times 1.30 \times 1.24 \times 1.10 = 2.04$$

가중치 반영된 M/M

노력 조정 수치 반영

P/M계산

11. 보정 계수 반영하기(2)

표 3-3 COCOMO 방법에서 사용되는 노력 승수 값

특성	비용 승수 요소	승수 값					
		매우 낮음	낮음	정상	높음	매우 높음	극히 높음
제품 특성	요구되는 신뢰도	0.75	0.88	1.00	1.15	1.40	
	데이터베이스 크기	0.94	1.00	1.08	1.16		
	제품의 복잡도	0.70	0.85	1.00	1.15	1.30	1.65
컴퓨터 특성	실행 시간 제약			1.00	1.11	1.30	1.66
	주기억 장치의 제약			1.00	1.06	1.21	1.56
	HW/SW의 안정성		0.87	1.00	1.15	1.30	
	처리 시간		0.87	1.00	1.07	1.15	
개발자 특성	분석가의 능력	1.46	1.19	1.00	0.86	0.71	
	응용 분야 경험	1.29	1.13	1.00	0.91	0.82	
	컴퓨터와 친숙성	1.21	1.10	1.00	0.90	-	
	프로그래머 능력	1.42	1.17	1.00	0.86	0.70	
	프로그램 언어의 경험	1.14	1.07	1.00	0.95		
프로젝트 특성	소프트웨어 공학 기술 사용	1.24	1.10	1.00	0.91	0.82	
	소프트웨어 도구의 사용	1.24	1.10	1.00	0.91	0.83	
	요구되는 개발 일정	1.23	1.08	1.00	1.04	1.10	

12. 보정 계수 반영하기(3)

- 노력 조정 수치가 반영된 노력(P/M)

표 3-4 노력 조정 수치가 반영된 유형별 노력^{P/M}

프로젝트 유형	공식
단순형	$PM = 2.4 \times (KDSI)^{1.05} \times EAF$
중간형	$PM = 3.0 \times (KDSI)^{1.12} \times EAF$
내장형	$PM = 3.6 \times (KDSI)^{1.20} \times EAF$

(예)

만일 개발하려는 소프트웨어의 KDSI가 60이고, 소프트웨어는 중간형이며, 노력 조정 수치(EAF)가 2.04라면 노력(E)?

$$PM = 3.0 \times (KDSI)^{1.12} \times EAF = 3.0 \times (60)^{1.12} \times 2.04 = 600.179$$

13. 총 개발 기간 산정

■ 총 개발 기간^{TDEV}: Total DEvelopment time

표 3-5 총 개발 기간 산정 공식

프로젝트 유형	공식
단순형	$TDEV = 2.5 \times (PM)^{0.38}$
중간형	$TDEV = 2.5 \times (PM)^{0.35}$
내장형	$TDEV = 2.5 \times (PM)^{0.32}$

(예)

만일 개발하려는 소프트웨어의 KDSI가 60이고, 소프트웨어는 중간형이며, 노력 조정 수치(EAF)가 2.04라면 노력(E)=600.179이다. 이 때 총 개발 기간은?

$$TDEV = 2.5 \times (PM)^{0.35} = 2.5 \times (600.179)^{0.35} = 23.461$$

14. COCOMO II 방법

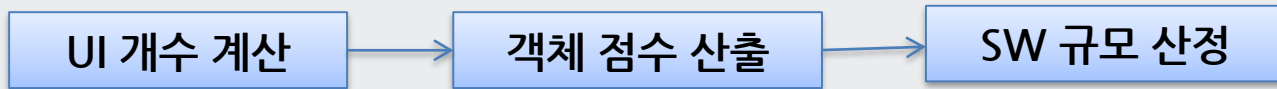
COCOMO의 문제점: (계획 단계)원시코드 라인수 예측



COCOMOII: 단계별로 값을 예측한 후 인건비 예측

15. 프로젝트 진행 정도에 따른 분류

- [1단계] 애플리케이션 합성 모델 application composition model



- [2단계] 초기 설계 모델 early design model
 - 초기 설계 단계에서 예측 값을 구한다.
 - 1단계보다 더욱 정확한 예측이 가능
- [3단계] 구조 설계 이후 모델 post-architecture model
 - 기능 점수를 바탕으로 한 LOC를 추정하여 소프트웨어 규모를 산정

16. 기능 점수 산정 방법(1)

- 산정 근거
 - 기능(입·출력, 데이터베이스 테이블, 인터페이스, 조회 등)의 수
 - → 라인 수와 무관하게 기능이 많으면 규모도 크고 복잡도도 높다고 판단
 - 사용자관점에서 소프트웨어의 기능 정량화 => 개발 비용 산정
- 기능 점수function point
 - 소프트웨어 기능의 크기를 측정하는 단위
 - (즉) 소프트웨어의 기능이 얼마나 복잡한가를 상대적인 점수로 표현하는 것

17. 기능 점수 산정 방법(2)

■ 용도

- 개발 시 비용 산정
- 유지보수 비용 산정
- 개발 시 필요한 자원 산정

■ 특징

- 소프트웨어 규모를 측정하는 방법
- 기능적 요구사항이 중심이 되는 측정 방법
- 소프트웨어의 요구사항 복잡도를 측정
- 구현 관점 아닌 사용자 관점의 요구 기능을 정량적으로 산정
- 측정의 일관성 유지를 위해 개발 기술, 개발 방법, 품질 수준 등은 고려하지 않음
- 소프트웨어 개발에 사용되는 언어와 무관
- 소프트웨어 개발 생명주기의 전체 단계에서 사용 가능

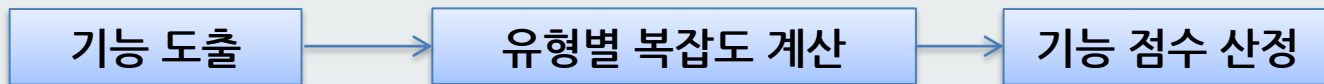
18. 소프트웨어 기능 분류



19. 기능 점수 산정 방법

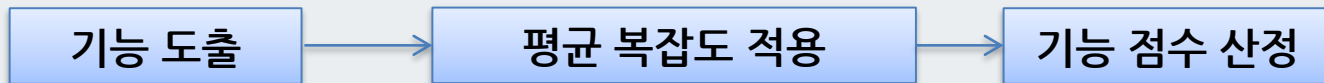
- 정규 기능 점수 법

- 설계 단계 이후에 사용하면 유용



- 간이 기능 점수 법

- 기획 및 발주 단계에서 사용



20. 기능 점수 산정 방법의 장점

- 사용자의 요구 사항만으로 기능을 추출하여 측정
 - 실제 구현 방법과 무관
- 객관적인 요구 사항만으로 측정
 - 개발 방법이나 개발 팀과 무관
- 모든 개발 단계에서 사용
 - 계획 단계뿐 아니라, 분석, 설계, 구현 단계에서도 사용 가능

21. 기능 점수 산정 방법의 단점

- 높은 분석 능력 필요
 - 요구 사항으로부터 기능을 도출하려면 상당한 분석 능력이 필요
- 기능 점수 전문가 필요
 - 이 방법을 잘 사용할 수 있는 기능 점수 전문가가 필요
- 내부 로직 위주의 소프트웨어에는 다소 부적합
 - 사용자가 알 수 있는 기능으로 측정하기 때문에 내부 로직 위주의 SW 측정에는 부적합
- 개발 규모 측정에 적합
 - 개발 규모를 예측하는 데 적합

22. 간이 기능 점수법 산정

- 평균 복잡도 가중치
 - 5가지 유형에 적용된 복잡도에 대해 계산한 가중치의 평균값
 - 5가지 유형: 내부 논리 파일, 외부 연계 파일, 외부 입력, 외부 출력, 외부 조회

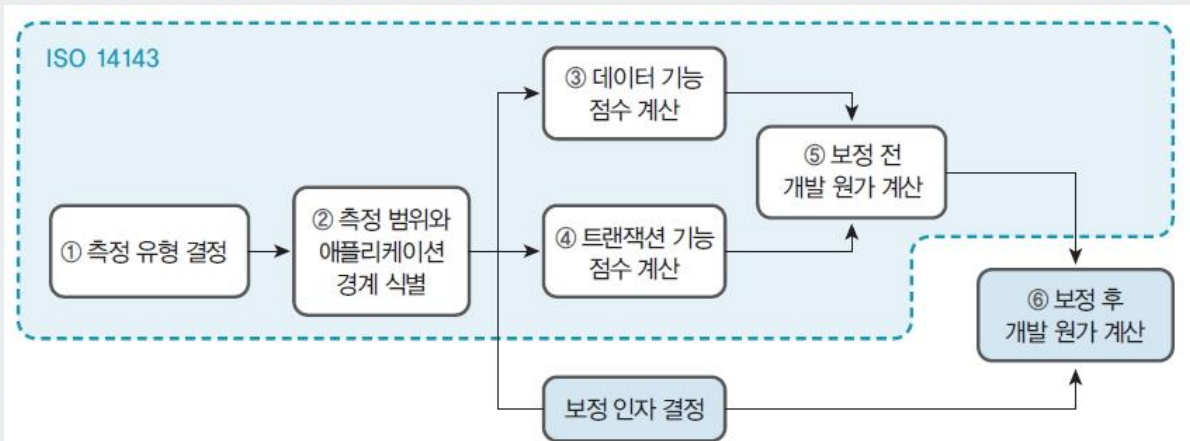
표 3-6 평균 복잡도 가중치

유형	내부 논리 파일	외부 연계 파일	외부 입력	외부 출력	외부 조회
가중치	7.5	5.4	4.0	5.2	3.9

출처 : SW사업 대가산정 가이드(2015년 개정판), 한국소프트웨어산업협회

- 간이 기능 점수법
 - 프로젝트 초기 단계에서 평균 복잡도 가중치를 사용하여 소프트웨어 기능의 크기를 측정

23. 간이 기능 점수 산정 절차



① 측정 유형(개발, 개선, 애플리케이션) 결정

② 개발하려는 소프트웨어의 측정 범위와 애플리케이션 경계 설정

③ 데이터 기능 점수 계산: 데이터 기능(내부 논리 파일, 외부 연계 파일) 도출 → 도출된 데이터 기능(내부 논리/외부 연계 파일)의 개수×평균복잡도 가중치

④ 트랜잭션 기능 점수 계산: 트랜잭션 기능(외부 입·출력, 외부 조회) 도출 → 도출된 트랜잭션 기능(외부 입·출력, 외부 조회)의 개수×평균복잡도 가중치

⑤ 보정 전 개발 원가 계산: 총 기능 점수×기능 점수당 단가(총 기능 점수=데이터 기능 점수+트랜잭션 기능 점수)

⑥ 보정 후 개발 원가 계산: 보정 전 개발 원가×4가지 보정 계수(규모, 애플리케이션 유형, 언어, 품질 및 특성)

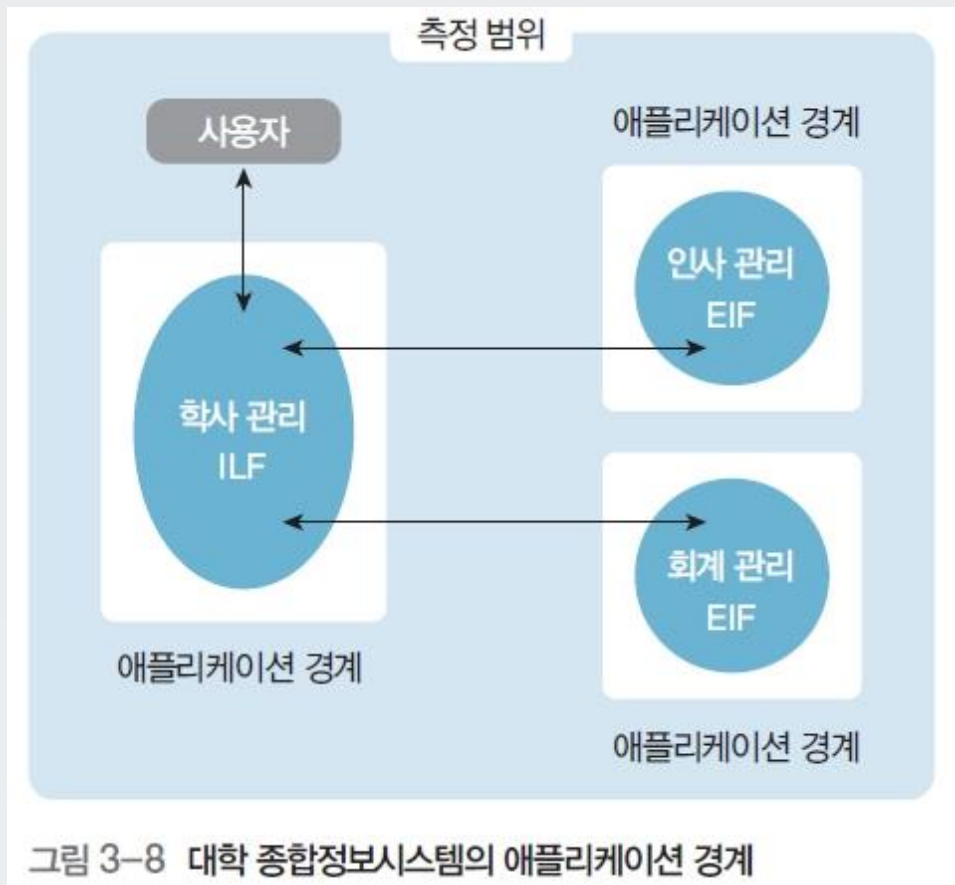
그림 3-7 간이 기능 점수 산정 절차

출처: CPM 4.1, IFPUG, 1999

24. 측정 유형 결정

- 개발 프로젝트 기능 점수(개발 규모 측정)
 - 프로젝트가 완료되어 최초 설치했을 때의 기능 크기를 산정
 - (즉) 프로젝트에서 사용자를 위해 제공된 모든 기능을 측정
- 개선(enhancement) 프로젝트 기능 점수(변경 규모 측정)
 - 사용 중인 소프트웨어에 변경이 발생했을 때 변경된 부분의 기능을 측정
 - (즉) 완료된 프로젝트에서 추가, 수정, 삭제된 부분만 그 크기를 측정
- 애플리케이션 기능 점수(응용 규모 측정)
 - 현재 운용 중인 애플리케이션의 기능을 측정
 - 개발프로젝트 기능 점수에 개선 프로젝트 기능 점수까지 포함된 것

25. 측정 범위와 애플리케이션 경계 식별(1)



26. 측정 범위와 애플리케이션 경계 식별(2)

- 측정 범위에 포함될 요소(서브시스템)의 식별
 - 측정 범위: 기능 점수를 측정하고자 하는 대상
 - 측정 대상: 개발하고자 하는 전체 소프트웨어 또는 소프트웨어의 일부 시스템
- 애플리케이션 경계
 - 금번 기능 점수를 계산하는 대상과 다른 애플리케이션이나 외부 사용자를 구분하는 경계
- 애플리케이션 경계 식별 시 주의 점
 - 사용자 관점에서 경계 구분
 - 개발자 관점(클라이언트/서버, 서버/비즈니스 로직/클라이언트 계층)으로 구분하면 안됨
 - 애플리케이션의 경계의 크기가 적당해야 함

27. 데이터 기능 점수 계산(1)

- 데이터 기능 점수 계산

- 내부 논리 파일(ILF)의 개수와 외부 연계 파일(EIF)의 개수를 계산하여 각각의 평균 복잡도(가중치)에 따라 데이터 기능 점수 결정

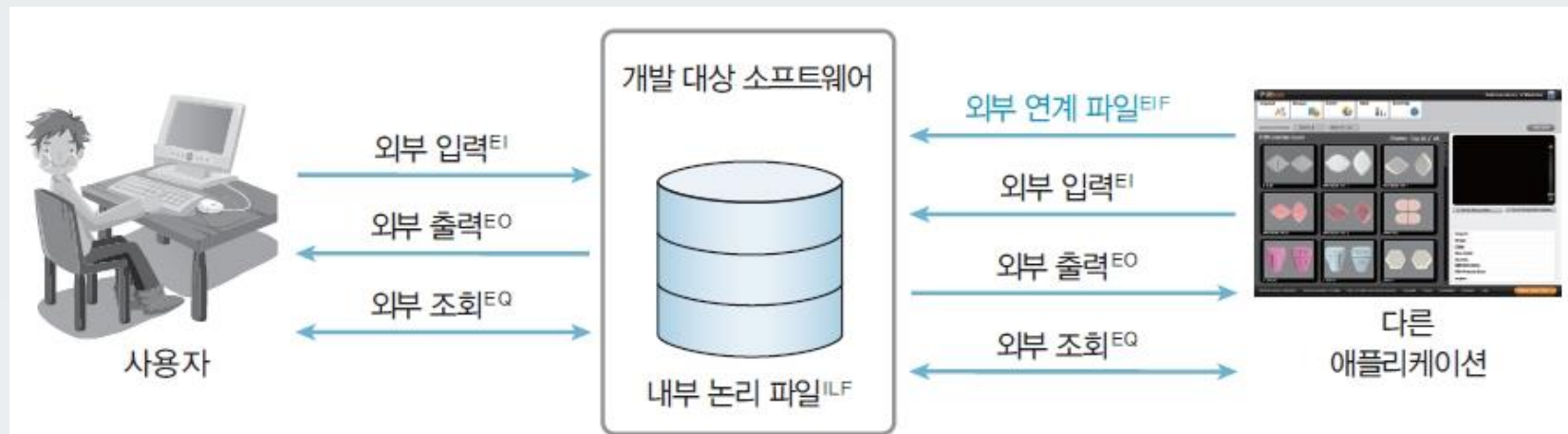


그림 3-9 사용자 관점에서 본 기능 점수의 5가지 유형

28. 데이터 기능 점수 계산(2)

■ 내부 논리 파일 ILF: Internal Logical File

- 사용자가 등록/수정/삭제/조회를 하기 위한 대상
- 데이터베이스에 존재하는 데이터 모임(데이터베이스의 정보)
- 데이터베이스 정보: 기능 점수 측정 대상으로 애플리케이션 내부에서 파일로 유지
- 데이터베이스 테이블, 시스템 내부에서 다루는 파일, 클래스 등
- 애플리케이션에 존재하는 데이터를 논리적으로 모아놓은 것
- 금번 프로젝트에서 생성하여 관리하는 데이터

■ 외부 연계 파일 EIF: External Interface File

- 측정 대상 애플리케이션에서는 참조만 하고 다른 애플리케이션에서 유지되는 파일
- 다른 프로젝트에서 생성하였으나 금번 프로젝트에서 참조하는 데이터

29. 데이터 기능 점수 계산(3)

표 3-7 외부 연계 파일과 내부 논리 파일의 예

기능	기능 유형	기능 명	기능 유형
학생 관리	학생 정보 관리	학생 기본 정보	내부 논리 파일 ^{ILF}
		학생 등록금 정보	외부 연계 파일 ^{EIF}

데이터 기능 점수 = {(ILF 개수×7.5) + EIF 개수 + 5.4}

7.5: 내부 논리 파일의 평균 복잡도

5.4: 외부 연계 파일의 평균 복잡도

30. 트랜잭션 기능 점수 계산(1)

- 트랜잭션 기능
 - 입력, 출력, 조회
- 트랜잭션 기능 측정
 - 외부 입력, 외부 출력, 외부 조회의 횟수를 세는 것
- 트랜잭션 기능 점수
 - {외부 입력, 외부 출력, 외부 조회}의 개수 X 각각의 평균복잡도(가중치)

31. 기능 유형(2)

- **외부 입력**external input
 - 데이터베이스에 데이터를 등록하거나, 수정·삭제하는 것
 - (예) 학생 정보 등록, 수정, 삭제
- **외부 출력**external output
 - 계산하는 로직을 거쳐 데이터나 제어 정보를 사용자에게 보여주는 기능
 - 수학적 계산 로직이 하나 이상 존재하며 그에 따른 파생 데이터도 존재
 - (예) 학생 학점 조회
- **외부 조회**external inquiry
 - 로직이 필요 없고 DB에 존재하는 데이터를 찾아 그대로 표시만 해주는 기능
 - (예) 학생 주소 검색, 학생 정보 조회

32. 트랜잭션 기능의 예(3)

표 3-8 트랜잭션 기능의 예

기능 명	기능 유형	기능 명	기능 유형
학생 관리	학생 정보 관리	학생 정보 등록	외부 입력 ^{TI}
		학생 정보 수정	외부 입력 ^{TI}
		학생 정보 조회	외부 조회 ^{EQ}
		학생 정보 삭제	외부 입력 ^{TI}
		학생 학점 조회	외부 출력 ^{EO}
		학생 정보 테이블	내부 논리 파일 ^{ILF}
		학생 등록금 테이블	외부 연계 파일 ^{EIF}
	비밀번호 관리	비밀번호 조회	외부 조회 ^{EQ}
		비밀번호 변경	외부 입력 ^{TI}

트랜잭션 기능 점수 = {(TI개수 × 4.0) + (EO개수 × 5.2) + (EQ개수 × 3.9)}

4.0: 외부 입력의 평균 복잡도

5.2: 외부 출력의 평균 복잡도

3.9: 외부 조회의 평균 복잡도

33. 미조정 기능 점수 계산

- 미조정 기능 점수UFP: Unadjusted Function Point

- 데이터 기능 점수 + 트랜잭션 기능 점수
- 단순히 기능적인 요구 사항에 대해서만 계산
- 여러 가지 특성에 대한 고려를 하지 않음

표 3-9 미조정 기능 점수의 예

기능		기능 수	평균 복잡도	기능 점수	
데이터 기능 점수	ILF	1	7.5	7.5	12.9
	EIF	1	5.4	5.4	
트랜잭션 기능 점수	EI	4	4.0	16	29
	EO	1	5.2	5.2	
	EQ	2	3.9	7.8	
계				41.9	

34. 보정 전 개발 원가 계산

- 보정 전 개발 원가
 - 미조정 기능 점수 X 기능 점수당 단가
 - 기능 점수당 단가: 519,203원(출처: SW사업대가산정 가이드)
 - 소프트웨어 개발 전체에 대한 기능 점수

표 3-10 소프트웨어 개발 단계별 기능 점수 가중치

단계	분석	설계	구현	테스트	합계
평균 복잡도 가중치	0.19	0.24	0.32	0.25	1.00
단가	98,648원	124,609원	166,145원	129,801원	519,203원

출처: SW사업 대가산정 가이드(2015년 개정판), 한국소프트웨어산업협회

35. 보정 후 기능 점수 계산(1)

- 규모 보정 계수
 - 규모에 따라 값을 보정
 - 대규모: 복잡도 증가 생산성 하락 보정 계수를 높여 보정
 - 기능 점수에 따라 달라진다.
- 애플리케이션 유형 보정 계수
 - 소프트웨어 유형에 따라 복잡도가 달라 생산성도 달라지는 것을 보정

표 3-11 애플리케이션 유형에 따른 보정 계수

애플리케이션 유형	애플리케이션 종류	보정 계수
업무 처리용	인사, 회계, 급여, 영업 등 경영 관리 및 업무 처리용 소프트웨어 등	1.0
과학 기술용	과학 계산, 시뮬레이션, 스프레드시트, 통계, OR, CAE 등	1.2
멀티미디어용	그래픽, 영상, 음성 등 멀티미디어 응용 분야 지리 정보 시스템, 교육/오락용 등	1.3
지능 정보용	자연어 처리, 인공지능, 전문가 시스템	1.7
시스템용	운영체제, 언어 처리 프로그램, DBMS, 인간/기계 인터페이스, 윈도우즈 시스템, CASE, 유틸리티 등	1.7
통신 제어용	통신 프로토콜, 에뮬레이션, 교환기 소프트웨어, GPS 등	1.9
공정 제어용	생산관리, CAM, CIM, 기기 제어, 로봇 제어, 실시간, 내장형 소프트웨어 등	2.0
지휘 통제용	군·경찰 등의 장비, 인력의 지휘 통제를 요하는 소프트웨어	2.2

출처: SW사업 대가산정 가이드(2015년 개정판, 한국소프트웨어산업협회)

36. 보정 후 기능 점수 계산(2)

- 언어 보정 계수
 - 언어에 따라 생산성도 달라지므로 개발 언어에 따른 보정 계수 적용

표 3-12 언어 유형에 따른 보정 계수

언어 유형	보정 계수
어셈블리어, 기계어, 자연어	1.9
C, C++, C#, Java	1.2
코볼 ^{Cobol} , 포트란 ^{Fortran} , 파스칼 ^{Pascal}	1.0
델파이 ^{Delphi} , HTML, 파워빌더 ^{Power Builder} , 비주얼 베이직 ^{Visual Basic} , XMLm Script(JSP, ASP, PHP, 플래시 ^{Flash} 등)	0.8
엑셀 ^{Excel} , 스프레드시트 ^{Spreadsheet}	0.6

출처: SW사업 대가산정 가이드(2015년 개정판), 한국소프트웨어산업협회

37. 보정 후 기능 점수 계산(3)

■ 품질/특성 보정 계수

표 3-13 품질/특성 보정 계수

품질/특성 요소		판단 기준	영향도
분산 처리	애플리케이션이 구성 요소 간에 데이터를 전송하는 정도	분산 처리에 대한 요구 사항이 명시되지 않음.	0
		클라이언트/서버 및 웹 기반 애플리케이션과 같이 분산 처리와 자료 전송이 온라인으로 수행됨.	1
		애플리케이션상의 처리 기능이 복수 개의 서버 또는 프로세서에서 동적으로 상호 수행됨.	2
성능	응답 시간 또는 처리율에 대한 사용자 요구 수준	성능에 대한 특별한 요구 사항이나 활동이 명시되지 않으며, 기본적인 성능이 제공됨.	0
		<ul style="list-style-type: none"> 응답 시간 또는 처리율이 피크 타임 또는 모든 업무 시간에 중요함. 연동 시스템의 처리 마감 시간에 대한 제한이 있음. 	1
		성능 요구 사항을 만족하기 위해 설계 단계부터 성능 분석이 요구되거나, 설계/개발/구현 단계에서 성능 분석 도구가 사용됨.	2
신뢰성	장애 시 미치는 영향의 정도	신뢰성에 대한 요구 사항이 명시되지 않으며, 기본적인 신뢰성이 제공됨.	0
		고장 시 쉽게 복구할 수 있는 수준으로 약간 불편한 손실이 발생함.	1
		고장 시 복구가 어려우며, 재정적 손실이 많이 발생하거나 인명 피해 위험이 있음.	2
다중 사이트	상이한 H/W, S/W 환경을 지원하도록 개발되는 정도	<ul style="list-style-type: none"> 설계 단계에서 하나의 설치 사이트에 대한 요구 사항만 고려됨. 애플리케이션이 동일한 H/W 또는 S/W 환경 하에서만 운용되도록 설계됨. 	0
		<ul style="list-style-type: none"> 설계 단계에서 하나 이상의 설치 사이트에 대한 요구 사항이 고려됨. 애플리케이션이 유사한 H/W 또는 S/W 환경 하에서만 운용되도록 설계됨. 	1
		<ul style="list-style-type: none"> 설계 단계에서 하나의 설치 사이트에 대한 요구 사항만 고려됨. 애플리케이션이 상이한 H/W 또는 S/W 환경 하에서만 운용되도록 설계됨. 	2

출처 : SW사업 대가산정 가이드(2015년 개정판), 한국소프트웨어산업협회

38. 보정 후 개발 원가 계산(1)

보정 후 개발 원가
= 보정 전 개발 원가 × (규모 보정 계수 × 애플리케이션 보정 계수
× 언어 보정 계수 × 품질/특성 보정 계수)

■ 학사 관리 개발 원가 계산의 예

표 3-14 기능 점수 계산을 위한 애플리케이션

애플리케이션	기능	세부 기능
학사 관리	학적	관리 학적 자료 등록/수정/삭제/조회
	교육 과정	관리 교육 과정 등록/수정/삭제/조회, 유사/동일 과목 조회
	수업 관리	개설 강좌 등록/수정/삭제/조회
	수강 관리	수강 과목 등록/수정/조회/삭제, 시간표 등록/수정/삭제/조회, 이수 구분 변경
	성적 관리	성적 등록/수정/조회
사용자 관리	교수 정보 관리	교수 정보 등록/수정/삭제/조회
	학생 정보 관리	학생 정보 등록/수정/삭제/조회

39. 보정 후 개발 원가 계산(1)

표 3-15 학사 관리 애플리케이션에 대한 개발비 계산

단계	기능			개수	평균 가중치	기능 점수
①	데이터 기능	ILF	학적 테이블, 교육 과정 테이블, 개설 강좌 테이블, 수강 과목 테이블, 성적 테이블	5	7.5	37.5
		EIF	교수 정보 테이블, 학생 정보 테이블	2	5.4	10.8
	트랜잭션 기능	EI	학적 자료 등록/수정/삭제, 교육 과정 등록/수정/삭제, 개설 강좌 등록/수정/삭제, 수강 과목 등록/수정/삭제, 시간표 등록/수정/삭제, 이수 구분 변경, 성적 등록/수정	18	4.0	72.0
		EO	유사/동일 과목 조회, 성적 조회	2	5.2	10.4
		EQ	학적 자료 조회, 교육 과정 조회, 개설 강좌 조회, 수강 과목 조회, 시간표 조회, 교수 정보 조회, 학생 정보 조회	7	3.9	27.3
	기능 점수의 합		158			
②	보정 전 개발 원가		$158 \times 519,203\text{원} = 82,034,074\text{원}$			
③	보정 계수		규모 보정(0.65), 애플리케이션 유형 보정(1.0), 언어 보정(0.8), 품질/특성 보정(1.0)			
	보정 후 개발 원가		$82,034,074\text{원} \times (0.65 \times 1.0 \times 0.8 \times 1.0) = 42,657,718\text{원}$			



다음 시간

일정 계획, 위험 분석



송실사이버대학교

송실사이버대학교의 강의콘텐츠는
저작권법에 의하여 보호를 받는바, 무단
전재, 배포, 전송, 대여 등을 금합니다.

*사용서체 : 나눔글꼴