



2024-2학기

UNIFIED
MODELING
LANGUAGE™




게임 지향 설계와 분석을 위한


UML
기초와 응용



Unified Modeling Language



소프트웨어 공학
이론과 실제



게임소프트웨어공학(GSE)

[2주차-1] 이론 2장 소프트웨어 품질

1. 소프트웨어 품질의 중요성


2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

6. 과제/진도/Q&A

 한국과학기술대학교

이 강의자료는 여러분들을 위한 담당교수의 경험/생각과 한빛아카데미㈜에서 제공한 교재 자료를 활용해서 작성하였습니다

1

[이론 2장] 소프트웨어 품질

■ 주요 내용

1. 소프트웨어 품질의 중요성

2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

■ 학습 목표

1. 소프트웨어의 품질에 대한 정확한 의미 이해

2. 소프트웨어 품질을 결정하는 요소 이해

3. 인공지능 소프트웨어 품질요소 이해

4. 소프트웨어 품질 관련 표준 이해











 한국과학기술대학교

이론 2장 소프트웨어 품질

2

Copyrighted by 석진원(2024-2학기)

1



UNIFIED
MODELING
LANGUAGE™

이론교재

2장

소프트웨어 품질



1. 소프트웨어 품질의 중요성
2. 소프트웨어 품질요소
3. 인공지능 소프트웨어 품질
4. 소프트웨어 품질 모델 및 표준
5. 소프트웨어 품질관리
6. 과제/진도/Q&A



이론 2장 소프트웨어 품질

3

소프트웨어 품질(Quality)

- 어떤 스마트폰을 선택할 것인가?
 - 선택의 기준은 무엇인가?
 - 사용자는 무엇을 기대하고 있는가?




그림 2-1 미래에 나타날 스마트폰의 예(Google 검색)

소프트웨어 품질

■ 소프트웨어 품질의 특징

- 소프트웨어는 물리적으로 존재하지 않아 품질을 눈으로 확인하기 어렵다.
- 개발 초기에 사용자의 요구는 건축물의 설계 도면과 같이 정확히 알 수 없다.
- 시간이 지날수록 사용자가 원하는 품질 수준이 점점 높아진다.
- 소프트웨어와 하드웨어 기술이 급진적으로 발전한다.
- 특정 제품의 품질은 100점, 90점과 같이 절대적으로 평가할 수 없다.
- 품질은 다양한 관련자들을 고려해야 하는 다차원적인 것이다.
- 소프트웨어 품질은 개발자의 능력, 예산, 개발 기간 등에 따라 달라질 수 있다.
- 품질은 이해관계자가 수락할 수 있는 수준에 존재해야 한다.

 한국공학대학교


이론 2장 소프트웨어 품질

5

소프트웨어 품질

■ 소프트웨어 품질의 정의

- **미국 국방부** Department of Defense(1985): 사용자가 의도한 기능을 수행할 수 있는 소프트웨어 속성에 대한 제공 정도
- **키친햄** Kitchenham(1986): 요구에 대한 적합성= 의도한 목적에 맞는 좋은 솔루션
- **IEEE**(1990): 시스템, 구성 요소 또는 프로세스에 명시된 요구사항을 충족시키는 정도 → 고객 및 사용자의 요구 및 기대를 충족시키는 정도
- **글래스** Glass(1992): 대상물, 즉 소프트웨어의 우수한 정도
- **ISO** International Standards Organization(1999): 요구사항을 만족하는 소프트웨어 제품의 능력
- **프레스만** Pressman(2004): 명시적인 기능 및 성능 요구사항, 명시적으로 문서화된 개발 표준, 개발된 소프트웨어에서 기대되는 묵시적인 특성에 대한 적합성

 한국공학대학교

이론 2장 소프트웨어 품질

6

소프트웨어 품질의 중요성

보여지는 증상

감춰진 증상

외적 품질

내적 품질

정확성
신뢰성

효율성
상호 운영성

견고성
기타 외적 품질

가시성
결합력
코딩 규칙
검증 가능성
유지보수성

융통성
재사용성
적시성
프로그램 구조
가독성
기타 내적 품질

비용
복잡도

그림 2-2 소프트웨어 품질 특성을 나타내는 품질 아이스버그

한국공학대학교

이론 2장 소프트웨어 품질

7

소프트웨어 품질의 중요성

사용자가 만족하는 품질의 소프트웨어

↓

품질 좋은 소프트웨어

↓

품질 좋은 제품

↓

시장에서의 선호도 향상

↓

기업의 이익 창출 증대

한국공학대학교

이론 2장 소프트웨어 품질

8

Copyrighted by 석진원(2024-2학기)

4

UNIFIED
MODELING
LANGUAGE™

UML

이론교재

2장

소프트웨어 품질

이론과 실제

소프트웨어 공학

이론과 실제

소프트웨어 공학

1. 소프트웨어 품질의 중요성

2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

6. 과제/진도/Q&A

한국공학대학교

이론 2장 소프트웨어 품질

9

소프트웨어 이해관계자(Stakeholder)

■ 소프트웨어 이해관계자의 3가지 역할: 스폰서, 사용자(고객), 유지보수자

사용자/고객
관점의 기대 사항

스폰서(sponsor)
관점의 기대 사항

유지보수자
관점의 기대 사항

그림 2-3 소프트웨어 품질 요소에 대한 스테이크홀더 간 관심사 및 연관성

한국공학대학교

이론 2장 소프트웨어 품질

10

Copyrighted by 석진원(2024-2학기)

5

소프트웨어 이해관계자(Stakeholder)

이해관계자별 기대치(관심사)

역할	기대치	대응 품질 요소
스폰서 (Sponsor)	<ul style="list-style-type: none">적은 비용으로 소프트웨어가 개발되기를 원한다.개발된 소프트웨어가 다른 응용에 사용되기를 원한다.기존 컴포넌트의 재사용을 통해 개발되기를 원한다.비즈니스 과정에 이득이 되기를 원한다.	<ul style="list-style-type: none">낮은 비용적응성(Adaptability)재사용성비용 효율성
사용자 (User)	<ul style="list-style-type: none">소프트웨어의 기능이 정확히 동작하기를 원한다.고장이 나지 않기를 바란다.사용하기 쉬워야 한다.적은 비용으로 구매하기를 원한다.	<ul style="list-style-type: none">정확성신뢰성사용성(Usability)낮은 비용
유지보수자 (Maintainer)	<ul style="list-style-type: none">소스 코드를 이해하기 쉬워야 한다.표준 코딩 스타일에 따라 코드가 개발되기를 원한다.변경 영향이 한 부분으로 국한되기를 원한다.수정된 코드를 쉽게 테스트하기를 원한다.코드와 일관성 있는 문서가 제공되어야 한다.	<ul style="list-style-type: none">가동성코딩 표준 준수성프로그램 구조검증 가능성문서화

한국공학대학교

이론 2장 소프트웨어 품질

11

[실습 2-1] 이해관계자의 기대치 식별

실습 2-1

이해관계자 stakeholder 기대치 식별하기

페이스북, 인스타그램과 같은 SNS 플랫폼 형태의 '가상 가족Virtual Family'을 개발하고자 한다. 이 플랫폼은 가상 공간에서 가족의 형성을 지원하는 기능을 담은 플랫폼으로, 불특정 다수의 인터넷 사용자를 대상으로 개발하고자 한다. 이때 이 플랫폼에 대하여 관련 스테이크홀더(스폰서, 사용자, 운영 관리자)의 기대치가 무엇인지 식별하시오.

스테이크홀더	기대치
스폰서	
사용자	
운영 관리자	

한국공학대학교

이론 2장 소프트웨어 품질

12

소프트웨어 외적 품질요소

■ 정확성(Correctness)

주어진 명세서의 내용을 하나씩 테스트하여 원하는 결과를 생성하는지 여부로 판단

Correctness(P)= (A/B)

P: 정확성을 알고 싶은 프로그램

B: 사용자의 전체 요구사항 개수

A: 테스트를 통해서 명세 내용대로 동작하지 못한 기능 수

■ 신뢰성(Reliability)

소프트웨어를 사용하는 동안 나타나는 오류(정확하게는 고장) 발생 정도로 판단

소프트웨어 고장의 빈도수와 그 치명도로 나타냄

예) MTBF (Mean Time Between Failure)= 10,000H

한국공학대학교

이론 2장 소프트웨어 품질

13

소프트웨어 외적 품질요소

■ 견고성(Robustness)

사용자가 제시한 요구사항 명세에 정의하지 않은 조건이나 환경에서도 소프트웨어가 합리적으로 동작해야만 견고한 것임

항공기 제어 소프트웨어, 원자력 발전소 제어 소프트웨어처럼 안전을 담보하기 위해 위험한 임무를 수행해야 하는 시스템에서 특히 중요

■ 성능(Performance)

소프트웨어의 효율성(Efficiency)을 의미

소프트웨어를 수행하기 위해 필요한 메모리의 양(Byte), 총 실행 시간(msec) 등이 척도, 알고리즘의 성능분석에 사용(빅오 표기법, O(n))

■ 사용자 친숙성(User Friendliness)

소프트웨어가 사용하기 편리한가를 나타내는 품질요소

편의성 지원 기능은 얼마나 제공되는지 측정

예) 소프트웨어 인터페이스에서 도움말 말풍선, 스크롤 바, 핫 키(Hot Key) 등의 기능 개수

한국공학대학교

이론 2장 소프트웨어 품질

14

Copyrighted by 석진원(2024-2학기)

7

소프트웨어 외적 품질요소

가용성(Availability)

- 서버와 네트워크, 프로그램 등의 정보 시스템이 정상적으로 사용 가능한 정도, 예) 장비 가동율
- 사용자가 소프트웨어를 정상적으로 사용 가능한 시간(Usable Time)을 소프트웨어의 전체 운영 시간(Operation Time)으로 나눈 값이 클수록 가용성이 높아짐

보안성(Security)

- 외부의 악의적인 공격이나 해커(Hacker)의 위협을 소프트웨어가 막아낼 수 있도록 구현하여 잠재적인 공격이 예측되는 상황에서도 소프트웨어가 올바르게 동작
- 소프트웨어 시스템에서 발견된 취약점의 개수, 사고 통계, 보안 취약으로 인한 연간 손실액 등이 척도

 한국공학대학교


이론 2장 소프트웨어 품질

15

소프트웨어 내적 품질요소

검증 가능성(Verifiability)

- 소프트웨어가 지닌 속성이 올바르게다는 것을 안전하게 확인 가능
- 정형 검증(Formal Verification)과 테스트로 평가
 - 정형 검증: 검증 대상을 형식 언어로 표현하고 이를 풀거나 시뮬레이션 함으로써 검증 대상이 정확하게 동작한다는 것을 보이는 방법
 - 테스트: 소프트웨어의 정확한 동작을 확인하기 위하여 적절하게 생성된 테스트 데이터를 이용하여 실행시키는 방법
- 다양한 정형 기법을 혼합하거나 테스트를 통해 시스템 속성을 확인 필요

 한국공학대학교

이론 2장 소프트웨어 품질

16

소프트웨어 내적 품질요소

■ 유지보수성(Maintainability)

• 수정 유지보수(Corrective Maintenance)

개발된 소프트웨어를 사용하는 동안 오류가 발생하는 경우에 이루어지는 활동

• 적응 유지보수(Adaptive Maintenance)

소프트웨어 운영 조건에 대한 변화를 수용하는 유지보수 활동

• 완전 유지보수(Perfective Maintenance)

사용 중인 소프트웨어에 대한 가독성 및 이해성을 높이는 재구조화를 목적으로 수행되는 활동

• 예방 유지보수(Preventive Maintenance)

소프트웨어가 정확하게 동작할 것인지를 테스트하고 점검

임무 중심(Mission-Critical) 및 안전 중심(Safety-Critical) 소프트웨어의 경우 매우 중요

소프트웨어 유지보수

수정 유지보수

적응 유지보수

완전 유지보수

예방 유지보수

그림 2-4 소프트웨어 유지보수 활동에 대한 구분

한국공학대학교

이론 2장 소프트웨어 품질

17

소프트웨어 내적 품질요소

■ 재사용성(Reusability)

• 새로운 소프트웨어를 개발하기 위해 기존 소프트웨어 컴포넌트를 다시 사용하는 정도

• $Reusability(P) = LOC(R)/LOC(S)$

P : 특정 대상

LOC(S) : 전체 개발된 산출물의 양

LOC(R) : 재사용에 의해 개발된 부분의 양

■ 이식성(Portability)

• 얼마나 다양한 하드웨어 플랫폼을 지원하는가와 얼마나 다양한 버전의 소프트웨어 플랫폼을 지원하는가 하는 것

한국공학대학교

이론 2장 소프트웨어 품질

18

Copyrighted by 석진원(2024-2학기)

9

소프트웨어 내적 품질요소

■ 생산성(Productivity)

외적 품질요소인 성능(Performance)의 속성을 적용한 내적 품질 속성

주어진 시간 내에 얼마만큼의 성과를 내고 있는가를 나타내는 척도 사용

■ 상호운용성(Interoperability)

서로 다른 소프트웨어들이 협업을 수행할 수 있는 능력을 충분히 제공하는 것

IoT(Internet of Things) 기술을 근간으로 하는 스마트 시티 환경에서 매우 중요함

Network-Centric Hub

Diagram showing a central hub connected to various nodes including Air Force, Navy, and Army, illustrating network-centric operations.

Connection

Diagram showing various devices (laptop, smartphone, tablet, etc.) connected to a central cloud, illustrating the concept of connectivity and interoperability.

그림 2-5 상호 운용성 품질의 필요성을 나타내는 개념도 (Google 검색)

한국과학기술대학교

이론 2장 소프트웨어 품질

19

소프트웨어 내적 품질요소

■ 가시성(Visibility)

소프트웨어 개발 단계의 상태 정보(산출물 리뷰 결과 및 품질 정보)와 산출물(요구사항 명세서, 설계서 등) 등을 체계적으로 유지·관리

프로젝트 진척에 대한 의사결정이 가능해지고, 개발일정을 조정하거나 관리 가능

■ 기타 품질요소

적응성(Adaptability), 무결성(Integrity), 회복성(Recoverability), 변경성(Changeability), 추적성(Traceability) 등

한국과학기술대학교

이론 2장 소프트웨어 품질

20

Copyrighted by 석진원(2024-2학기)

10

프로세스 품질

프로세스 모델 적합성

어떤 프로세스 모델을 적용할 것인가

개발 방법론 적합성

객체지향 방법, 구조적 방법론, 정보공학 방법론 등

도구 적합성

소프트웨어 개발 과정에서 많은 지원 도구(CASE 툴)가 개발 환경에 적합한지, 팀 기반 개발을 충분히 지원할 수 있는지

표준 준수성

선택한 표준이 적절한지, 선택한 표준을 준수하여 프로젝트가 진행되고 있는지

프로젝트 데이터 관리 수준

프로젝트 진행과 관련된 상세 데이터가 정보 저장소에 저장되어 프로젝트 내에서 혹은 추후 신규 프로젝트의 예측 활동에서 활용할 수 있는지

한국공학대학교

이론 2장 소프트웨어 품질

21

UNIFIED
MODELING
LANGUAGE™

UML

이론교재

2장

소프트웨어 품질

이론 2장 소프트웨어 품질

이론 2장 소프트웨어 품질

1. 소프트웨어 품질의 중요성

2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

6. 과제/진도/Q&A


한국공학대학교

이론 2장 소프트웨어 품질

22

전통적인 소프트웨어 vs. 인공지능(AI) 소프트웨어

- 학습 모델의 입력과 결과 사이의 관계는 입력 데이터의 일부에 대해서만 정의
- 캡슐화 및 모듈화 같은 소프트웨어 엔지니어링의 일반적인 개발 원칙은 그대로 적용하기 어려움
- 기계학습을 포함하는 컴포넌트의 개발과 통합의 접근방법은 매우 다양한 방식에 의해 이루어짐
- 학습기능을 수행하는 알고리즘은 학습 및 테스트에 사용하는 데이터보다 훨씬 덜 중요하게 고려
- 모바일, 국방, 금융, 정부, 우주과학, 게임 등의 SW 특성은?


 한국과학기술대학교

이론 2장 소프트웨어 품질

23

인공지능 소프트웨어의 품질 특성

- 투명성Transparency과 책임Accountability
 - 기계학습 기반 시스템은 동일한 입력에 대하여 서로 다른 결과를 제공할 수 있음
 - AI시스템이 제공하는 출력은 시간에 따라 변할 수 있다는 가능성 때문에 출력에 대한 해석 가능성 및 설명 가능성이 중요
 - 출력 결과가 부정적인 결과를 유발하는 경우, 이를 사용자에게 알려주는 보고 기능이 부가적으로 제공되는 것이 필요
- 다양성Diversity, 공정성Fairness 그리고 사회적 웰빙well-being
 - AI 시스템 개발에서 학습 모델의 구성과 이에 필요한 의사결정 정책 등은 특정 요소에 편중하여 출력을 제공하는 오류가 있어서는 안됨
 - 모든 가능한 데이터들을 수용하여 출력을 제공할 수 있는 모델 설계 필요
 - 다양한 이해관계자의 참여를 통해 AI시스템은 환경 친화적 방향으로 구축되어 사회적 웰빙을 제공해야 함


 한국과학기술대학교

이론 2장 소프트웨어 품질

24

인공지능 소프트웨어의 품질 특성

- **보안Security과 안전성Safety**
 - AI 시스템의 응용 범위가 넓어져 개인 정보의 누출이나 프라이버시 침해 소지가 있음
 - 정확한 문제 해결을 위해서 사용되는 데이터가 좋은 품질과 무결성을 가져야 하는 것은 물론, 데이터에 존재할 수 있는 오류, 노이즈Noisy, 정체 불명이거나 악의적인 데이터들에 대처할 수 있어야 함
- **기술적 견고성Robustness과 신뢰성Reliability**
 - 안전하고 신뢰할 수 있는 시스템을 개발하는 것이 중요
 - 유해한 입력, 오류가 있는 입력에 대해서도 믿을 만하고 이해할 만한 결과가 제공되어야 함
- **법적Legal, 윤리적Ethical 측면**
 - AI 시스템이 인간이 해온 일들을 대행하는 에이전트Agent 역할을 수행하거나 인간이 하는 일들을 감독하고 모니터링 할 수 있으므로 인간에게 적용되는 법적, 윤리적 문제들을 보장할 수 있도록 개발되어야 함

 한국공학대학교

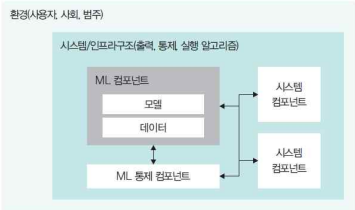
이론 2장 소프트웨어 품질

25

시스템 관점별 품질요소


- **기계학습(ML) 시스템의 구성 관점**
 - **모델 관점**
 - 학습 모델과 관련되며, 분류나 차원 축소 같은 작업을 수행하기 위해 데이터에 대한 훈련이 이루어지는 부분
 - **데이터 관점**
 - 모델에 입력되는 실데이터와 관련된 부분으로
 - **시스템 관점**
 - 기계학습 컴포넌트(모델과 데이터)들을 연결하고, 형상Configuration을 정의하는 부분
 - **인프라 관점**
 - 어떻게 구현되는가에 초점을 맞춘 부분으로 시스템 관점과 밀접한 관계가 있음
 - **환경 관점**
 - 시스템 외부 관점으로 시스템과 사용자가 어떻게 상호작용하는가를 표현하는 부분

환경(사용자, 사회, 범주)



시스템(인프라구조(클라우드, 통제, 실행 알고리즘))

그림 2-6 기계학습 기반 소프트웨어 시스템의 구성 관점

 한국공학대학교

이론 2장 소프트웨어 품질


26

시스템 관점별 품질요소

■ 관점별 품질요소: 모델, 데이터, 시스템, 인프라 및 환경 관점

표 2-2 모델 관점의 품질 요소

측정 대상	모델 관점 품질 요소
모델 유형	모델 타당성: 정의된 기능을 수행하기 위해 선정된 모델 유형이 적합한가
훈련된 모델	적합성(Fitness): 정의된 기능이 개발된 데이터를 기반으로 정확하게 수행될 수 있는가
	견고성(Robustness): 누락 혹은 오류가 있는 데이터를 잘 처리할 수 있는가
	안정성(Stability): 서로 다른 데이터에 대하여 반복적인 결과를 생성할 수 있는가
	공정성(Fairness): 모델의 출력이 공정한 결정을 제시하는가
	해석능력(Interpretability): 훈련된 모델의 내용을 사람이 해석할 수 있는가

 한국공학대학교

이론 2장 소프트웨어 품질

27

시스템 관점별 품질요소

표 2-3 데이터 관점의 품질 요소

측정 대상	데이터 관점 품질 요소
개발 데이터	대표성(Representativeness): 데이터가 모집단을 대표하는가
	정확성(Correctness): 데이터가 오류 없이 개발되었는가
	완전성(Completeness): 누락된 데이터가 없는가
	유통성(Currentness): 데이터가 최신 내용을 포함하는가
	독립성(Independence): 훈련 데이터와 테스트 데이터가 상호 독립적인가
개발 및 운용 데이터	일관성(Consistency): 서로 다른 데이터셋에 대하여 형식, 표본 추출 등이 일관성이 있는가

표 2-4 환경 관점의 품질 요소

측정 대상	환경 관점 품질 요소
훈련 과정	환경 영향(Env. Impact): 훈련 과정이 환경에 영향을 미치는 정도
활용 집단	사회 영향(Social Impact): ML 컴포넌트가 사회에 영향을 미치는 정도
범위	범위 준수(Scope Compliance): ML 컴포넌트가 의도된 사용 범위 내에 존재하는 정도

 한국공학대학교

이론 2장 소프트웨어 품질

28

시스템 관점별 품질요소	
표 2-5 인프라 관점의 품질 요소	
측정 대상	인프라 관점 품질 요소
인프라구조	인프라 적합성(Suitability): 인프라 구성 요소가 ML 컴포넌트의 요구를 충족하는가
훈련 알고리즘	훈련 효율성(Training Efficiency): 학습 모델을 훈련하기 위한 자원이 유용한가
실행 알고리즘	실행 효율성(Execution Efficiency): 훈련된 모델을 실행시키기 위한 자원이 유용한가
표 2-6 시스템 관점의 품질 요소	
측정 대상	시스템 관점 품질 요소
출력 제어	효과성(Effectiveness): 출력 제어 알고리즘이 오류 출력을 탐지하는 능력
	제어 효율성(Supervision Efficiency): ML 컴포넌트를 모니터링하기 위해 사용되는 자원이 유용한가
범주 제어	효과성(Effectiveness): 범주 제어 알고리즘이 문맥 변경을 탐지하는 능력
	제어 효율성(Supervision Efficiency): 응용 범주를 모니터링하기 위해 사용되는 자원이 유용한가

[실습 2-2] 새로운 품질요소 설계하기	
실습 2-2	새로운 품질 요소 설계하기
<p>사회적·기술적 변화에 따라 소프트웨어 기능도 복잡해지고 다양해지고 있다. 1990년대 컴퓨터는 다량의 데이터에 대한 연산을 고속으로 처리하여 정확한 결과를 제공하는 것이 목적이었다. 그러나 2020년대 컴퓨터는 의사, 판사 등 전문가들의 역할을 대체하거나 그림 그리기, 작곡하기 등 인간의 창작 영역에까지 도전하고 있다.</p> <p>이처럼 변화된 소프트웨어의 특성에 맞춰 품질 요소를 고려하는 것은 좋은 소프트웨어를 개발하기 위한 기초가 된다. 새로운 소프트웨어의 품질 요소와 해당 품질 요소를 측정하기 위한 척도^{Metric}를 개발하는 일은 소프트웨어 엔지니어가 해야 할 일 중 하나다. 새로운 소프트웨어 품질 요소를 개발하기 위한 방법으로 GQM^{Goal-Question-Metric} 방법이 있다.</p> <p>(1) GQM 방법에 대해 조사하고, 그 활용 방법을 이해한다.</p> <p>(2) SNS 플랫폼 '가상 가족'을 개발하는 데 고려해야 하는 새로운 품질 요소를 설계하시오.</p>	

UNIFIED
MODELING
LANGUAGE™

UML

이론교재

2장

소프트웨어 품질

이론 2장 소프트웨어 품질

1. 소프트웨어 품질의 중요성
2. 소프트웨어 품질요소
3. 인공지능 소프트웨어 품질
4. 소프트웨어 품질 모델 및 표준
5. 소프트웨어 품질관리
6. 과제/진도/Q&A

UML 기초와 응용

소프트웨어 공학 이론과 실제

한국공학대학교

이론 2장 소프트웨어 품질

31

소프트웨어 품질 모델: McCall의 FCM 모델

FCM의 기본 개념

• **Factors(요소):** 사용자에게 보이는 소프트웨어의 외적 특성 기술, 시스템의 동작 특성

• **Criteria(기준):** 개발자에게 보이는 소프트웨어의 내적 특성 기술, 소프트웨어 개발 및 설계와 관련된 품질요소 정의

• **Metrics(지표):** 소프트웨어의 내적·외적 특성을 측정하기 위한 기본 단위와 방법 정의 및 설명

소프트웨어 품질

품질요소1, 품질요소2, 품질요소3

품질기준1, 품질기준2, 품질기준3

품질 지표(척도, 메트릭)

그림 2-7 McCall의 품질 모델

한국공학대학교

이론 2장 소프트웨어 품질

32

Copyrighted by 석진원(2024-2학기)

16


소프트웨어 품질 모델: McCall의 FCM 모델

■ McCall의 Factor Model Tree

- 운영관점, 개선 관점, 전환 관점으로 구분

표 2-7 McCall이 정의한 품질 요소의 분류

구분	세부 품질 요소
제품 운영	정확성(Correctness), 신뢰성(Reliability), 효율성(Efficiency), 무결성(Integrity), 사용성(Usability)
제품 개선	시험 가능성(Testability), 융통성(Flexibility), 유지보수성(Maintainability)
제품 전환	이식성(Portability), 재사용성(Reusability), 상호 운용성(Interoperability)

 한국공학대학교


이론 2장 소프트웨어 품질

33

소프트웨어 품질 모델: HP의 FURPS 모델

■ FURPS의 구성 요소

- F(Functionality)
 - 소프트웨어가 수행하는 기능Feature Set과 이러한 기능의 일반성Generality 및 보안성Security 요소 포함
- U(Usability)
 - 소프트웨어의 미적Aesthetic 구성 및 특성, 일관성, 문서화 등의 요소 포함
- R(Reliability)
 - 소프트웨어 고장의 빈도와 치명도, 고생 발생 주기, MTBF, 고장 회복력, 출력 정확도 등의 요소 포함
- P(Performance)
 - 처리 속도 및 응답 시간, 자원 사용률 등의 요소 포함
- S(Supportability)
 - 소프트웨어 확장, 적용, 수정 등과 관련된 요소 포함

 한국공학대학교

이론 2장 소프트웨어 품질

34

소프트웨어 품질 모델: ISO 9126 품질 모델(표준)

ISO/IEC 9126: Software Engineering – Product Quality(2001)

기능성
Functionality

신뢰성
Reliability

사용성
Usability

효율성
Efficiency

유지보수성
Maintainability

이식성
Portability

사용자가 요구하는 기능이 소프트웨어에 구현되었는가?

소프트웨어가 얼마나 믿을 만한가?

소프트웨어가 사용하기 쉬운가?

소프트웨어가 효율적으로 동작하는가?

소프트웨어가 수정하기 용이한가?

소프트웨어가 다른 환경에서 사용할 수 있도록 쉽게 전환 가능한가?

그림 2-8 ISO 9126의 Six(6) 품질 특성 모델

한국공학대학교

이론 2장 소프트웨어 품질

35

소프트웨어 품질 모델: ISO 9126 품질 모델(표준)

표 2-8 ISO 9126에서 정의하는 세부 품질 요소의 분류

품질 특성	세부 품질 요소
기능성	적합성(Suitability), 정확성(Accuracy), 상호 운용성(Interoperability), 준수성(Compliance), 보안성(Security)
신뢰성	성숙성(Maturity), 결함 허용성(Fault-Tolerance), 회복성(Recoverability)
사용성	이해성(Understandability), 학습 용이성(Learnability), 운영성(Operability)
효율성	시간 효율성(Time behaviour), 자원 효율성(Resource behaviour)
유지보수성	분석성(Analyzability), 변경성(Changeability), 안정성(Stability), 시험 가능성(Testability)
이식성	적용성(Adaptability), 설치 용이성(Installability), 부합성(Conformance), 대체 가능성(Replaceability)

한국공학대학교

이론 2장 소프트웨어 품질

36

소프트웨어 품질 모델: ISO 25010 품질 모델(표준)

- ISO/IEC 25010: Systems and software engineering-Systems and software Quality Requirements and Evaluation(SQuaRE) -System and software quality models(2011)

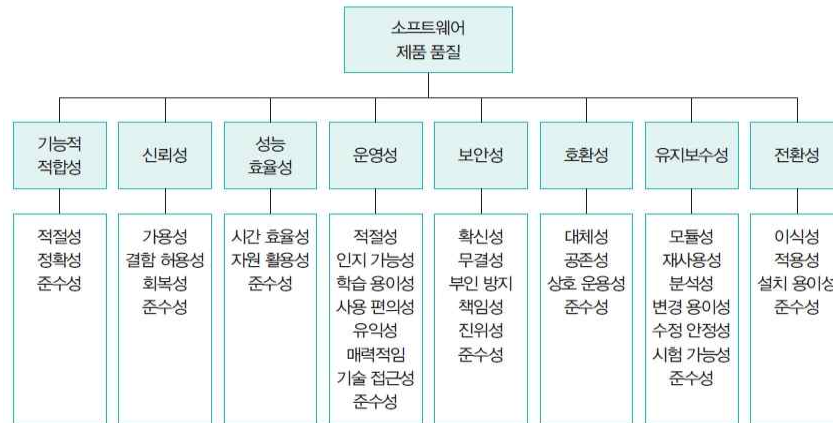


그림 2-10 ISO 25010에서 정의하는 Eight(8) 품질 특성 모델

소프트웨어 품질 모델: 다양한 품질 모델과 관계성

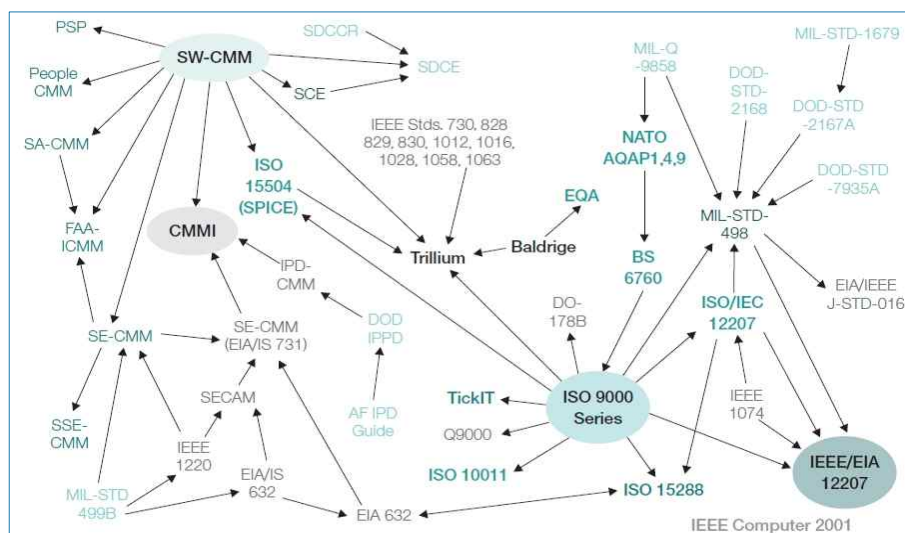


그림 2-12 소프트웨어 품질 표준의 진화 및 관계성

UNIFIED
MODELING
LANGUAGE™

UML

이론교재

2장

소프트웨어 품질

이론과 실제

소프트웨어 공학

이론과 실제

소프트웨어 공학

1. 소프트웨어 품질의 중요성

2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

6. 과제/진도/Q&A

한국공학대학교

이론 2장 소프트웨어 품질

39

정량적 품질 개선

소프트웨어 개발 과정에서 적용되는 정량적 품질관리 프로세스

준비 단계

관리 단계

평가 단계

정보 지원

개선 활동

사용자 피드백

결과 분석

결과 분석 및 성능 척도

측정 계획

새로운 이슈

척도 조정 단계

측정 단계

그림 2-13 소프트웨어 품질 측정을 위한 상위 프로세스

한국공학대학교

이론 2장 소프트웨어 품질

40

Copyrighted by 석진원(2024-2학기)

20


정량적 품질 개선

■ 준비 단계

- 프로젝트 혹은 소프트웨어 개발조직 전체 차원에서 품질관리 활동이 이루어진다는 것을 공식화
- 품질관리 담당자 지정 등 팀 구성과 역할을 정의
- 품질관리 활동을 위해 요구되는 적용 표준, 지원 도구, 환경 등을 구축

■ 척도 조정 단계

- 프로젝트의 다양한 품질요소 중에서 어떤 요소를 측정할 것인지 결정
- 프로젝트의 특성과 응용 소프트웨어에 대한 요구사항을 바탕으로 적용 대상 품질요소 확정
- 각 품질요소를 적용할 단계와 대상물 정의(품질요소의 정량화 방법 포함)
- 모든 척도 조정 정보는 품질 측정계획서에 기술

 한국공학대학교

이론 2장 소프트웨어 품질

41


정량적 품질 개선

■ 측정 단계

- 품질 측정 계획서를 바탕으로 별도의 업무부담 없이 프로젝트 데이터 수집 및 정량화 방법에 의해 분석
- 새로운 중요 데이터가 발견되거나 하나의 척도를 분리해야 하는 상황이 발생하면 새로운 척도 마련 및 적용

■ 평가 단계

- 측정된 품질요소들이 올바른 품질 정보를 제공하는지 점검하고, 품질 수준 평가
- 품질 수준 평가는 유사 프로젝트의 품질요소 수준과 상대적인 비교를 통해 진행되거나, 조직에서 정의한 품질 척도에 근거해서 진행하고, 품질 측정 과정이 올바르게 진행되었는지 점검
- 품질 데이터 수집 시점, 데이터 임의 변경 및 조작 등을 점검
- 수집된 데이터와 측정 데이터는 조직 차원의 프로젝트 데이터베이스에 저장

 한국공학대학교

이론 2장 소프트웨어 품질

42

정량적 품질 개선


■ 관리 단계

• 단위 프로젝트의 품질관리 활동이라기보다 조직 차원에서 수행하는 품질 및 기술 관리 프로세스에 해당

• 적절한 품질 표준을 선택

• 품질관리 활동에 대한 다양한 양식 및 가이드라인 제공

• 교육 등을 통한 조직 차원의 품질관리 활동을 관리·지원

 한국공학대학교

이론 2장 소프트웨어 품질


43


참고: 국가품질상 제도

■ 대한민국 국가품질상(KNQA: Korea National Quality Award)

• 대한민국을 대표하는 품질에 대한 국가포상제도, 1975년 제정, 품질경영 혁신활동에 탁월한 경영성과를 창출하여 국가산업 경쟁력 향상에 크게 기여한 공로자, 우수기업 및 단체에 수여

• 법적근거: 산업표준화법 제31조의 4 및 동법 시행령 제30조의 4(품질경영 우수기업의 선정, 포상 및 지원)



 한국공학대학교


이론 2장 소프트웨어 품질

44

참고: 세계의 국가품질상 제도


■ 말콤볼드리지상(Malcolm Baldrige National Quality Award), 미국

- 1987년 말콤볼드리지 국가품질개선법안(Public Law 100-107)에 의해 제정된 상으로서 미국 산업계에 품질의 중요성을 일깨우기 위해 우수한 품질을 성취한 기업들을 발굴하여 수여




■ 유럽품질상(The European Quality Award), EU(유럽연합)

- 전사적 품질경영(TQM) 활동에서 최고의 성과를 거둔 유럽기업에 수여되는 품질상으로 유럽 품질경영재단(EFQM)에서 운영, 조직의 경쟁력 확보를 위해 제품 품질의 확보가 최우선이라는 인식 하에 1992년에 제정




■ 데밍상(The Deming Prize), 일본

- W. Edwards Deming 박사가 일본에 통계적 품질관리 기법을 전파시킨 공적을 기리기 위해 1951년 일본과학기술연맹(JUSE)이 제정한 상, 전사적 품질관리(CWQC)의 수행에 뛰어난 성과를 이룩한 기업에 수여



★ 미국, 일본 등을 비롯한 80여 개국에서 국가품질상 제도를 운영 중, 품질의 중요성을 일깨우고 산업의 품질경쟁력을 높여 국가경쟁력 향상에 기여

 한국공학대학교

이론 2장 소프트웨어 품질

45

[이론 2장] 소프트웨어 품질 요약

■ 소프트웨어 품질

- 사용자의 기대치에 대한 충족 정도

■ 소프트웨어 품질요소


- 외적 품질요소 vs. 내적 품질요소
- 프로덕트 품질요소 vs. 프로세스 품질요소

■ 인공지능(AI) 소프트웨어의 품질요소

- 인공지능 모델 관점의 품질요소: 적합성, 견고성 등
- 데이터 관점의 품질요소: 대표성, 정확성 등
- 환경 관점의 품질요소 와 인프라 관점의 품질요소
- 시스템 관점의 품질요소: 효과성, 제어 효율성 등

■ 소프트웨어 품질 관리 및 표준

- ISO/IEC 9126(2001) → ISO/IEC 25010(SQuaRE) (2011)

 한국공학대학교

이론 2장 소프트웨어 품질

46

[프로젝트 II] 대상시스템의 품질요소 추정

프로젝트 II


대상 시스템 품질 요소 추정하기

1장의 프로젝트 정의서에서 제시한 핵심 기능과 유사 시스템 분석 결과를 기반으로, 대상 시스템에서 중요하게 고려해야 할 품질 요소를 추정하고 정리합니다. 이 사항은 추후 프로젝트 관리 계획서 작성에 반영됩니다.

(1) 사용자가 원하는 기대치는 무엇인가?

(2) 개발자 혹은 유지보수 담당자가 원하는 기대치는 무엇인가?


(3) 개발 조직의 관리자가 원하는 기대치는 무엇인가?

한국공학대학교

이론 2장 소프트웨어 품질

47

UNIFIED
MODELING
LANGUAGE™



이론교재

2장

소프트웨어 품질

UML
기초와 응용

소프트웨어 공학
이론과 실제

1. 소프트웨어 품질의 중요성

2. 소프트웨어 품질요소

3. 인공지능 소프트웨어 품질

4. 소프트웨어 품질 모델 및 표준

5. 소프트웨어 품질관리

6. 과제/진도/Q&A


한국공학대학교

이론 2장 소프트웨어 품질

48

6. 과제/진도/Q&A: 2주차 과제 없음

■ 2주차 과제 없음


 한국공학대학교

이론 2장 소프트웨어 품질

49

6. 과제/진도/Q&A: 2주차/전체

주차	강의 내용	수업 유형	학습 활동
1	공통1장 강제인자+이론1장 소개표+O&A 객체 지향 설명지 작성	대면수업(이론/실습)	대면수업/실습, 과제 해결
2	이론2장 SW 품질+실습1장 UML 이해+12장 starUML 모델링도구 설치 및 사용법+피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
3	이론3장 SW 개발프로세스+실습2장 UML 구성요소/뷰+프로젝트 팀편성/주제선정/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
4	이론4장 SW 개발방법론(DevOps+UP)+실습3장 유스케이스 다이어그램+문제기술서(SOP) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
5	국경일(개천절) 휴강(15주차 보장)	국경일 휴강	국경일 휴강
6	이론5장 프로젝트 관리+실습4장 클래스 다이어그램+프로젝트정의서(PC) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
7	이론6장 SW 비용산정+실습5장 순차 다이어그램+프로젝트관리계획서(PMP) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
8	이론7장 요구사항 도출+실습6장 통신 다이어그램+요구사항정의서(SRD)/중간발표(PT+PMR) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
9	중간고사(필기+개인)+프로젝트 중간발표(PT+PMR+팀별)/피드백	대면수업(시험/발표)	서술형 필기시험/구두발표
10	이론8장 객체지향 분석+실습7장 활동 다이어그램+요구사항추적표(RTM) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
11	이론9장 모듈화 설계+실습8장 상태 다이어그램+1. 요구사항명세서(SRS) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
12	이론10장 설계 패턴+이론11장 객체지향 설계+실습9장 컴포넌트 다이어그램+설계기술서(SDD) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
13	이론12장 인스펙션+이론13장 코딩+실습10장 배치 다이어그램+구현계획서(SIP) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
14	이론14장 화이트박스 테스트+이론15장 블랙박스 테스트+실습 11장 패키지 다이어그램+시험계획서(STP)/시험설계서(STD) 작성/피드백	대면수업(이론/실습)	대면수업/실습, 과제 해결
15	이론16장 SW 개발 적용 기술+실습12장 깃과 깃허브 활용 방법+구현결과서(SIR)/시험결과서(STR)/최종발표(PT+PCR) 작성/피드백	대면수업(이론/실습) (5주차 보장)	대면수업/실습, 과제 해결 (5주차 보장)
16	기말고사(L&L+개인중간고사(필기+개인))+프로젝트 중간발표(PT+PCR+팀별)/피드백+최종보고서(PCR) 제출	대면수업(시험/발표)	서술형 필기시험/구두발표

 한국공학대학교

이론 2장 소프트웨어 품질

50

6. 과제/진도/Q&A: 2주차(결과)-3주차(계획)

2주차 강의 진행 결과					
주차	주요학습내용	학습성과 학습목표	수업운영방법	학습준비사항	교재, 참고도서 (page)
2주차	<ul style="list-style-type: none">이론 2장 SW의 품질실습 1장 UML의 이해실습 12장 StarUML을 이용한 SW 개발방법	<ol style="list-style-type: none">SW 품질의 이해UML의 이해StarUML 모델링 도구사용법 이해	<ul style="list-style-type: none">대면강의+실습	교재 준비(이론, 실습) 및 이론 2장/실습 1장, 12장 읽어 보기	강의계획서+이론/실습 교재/참고도서+강의자료
3주차	<ul style="list-style-type: none">이론 3장 SW 개발프로세스실습 2장 UML 구성요소/뷰팀 편성/구축+프로젝트주제 선정 (주제 선정/개요 작성	<ol style="list-style-type: none">SW 개발프로세스의 이해UML 구성요소와 뷰의 이해팀 편성 및 구축 방법, 주제선정 방법/개요 작성 이해	<ul style="list-style-type: none">대면강의+실습[과제#2] 팀 편성/역할 분장+프로젝트 주제 작성 제출	교재 준비(이론, 실습) 및 이론 3장/실습 2장 읽어 보기	강의계획서+이론/실습 교재/참고도서+강의자료


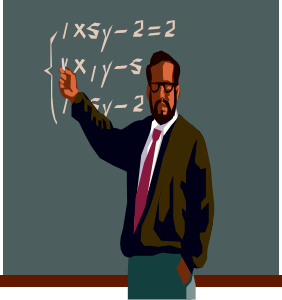
6. 과제/진도/Q&A: [진도] 3주차 계획

- 강의계획서는 **잘(정확히)** 숙지하고, 매주 강의 **진도 확인**하기
- 과제 **없음**
- 강의교재의 이론 3장, 실습 2장 읽어보기

👉 3주차: 이론3장 SW개발 프로세스+실습2장 UML 구성요소/뷰+팀 편성/프로젝트 주제 선정/개요 작성+피드백


□ 궁금하면 질문하자!

▪ LMS의 질의응답, 댓글, 쪽지 및 메일 기능을 적극 활용하기 바랍니다!




학습(學習)은 배우고 익히라는 것이다.
배우기만 힘 쓰면 스스로 할 수 없는 사람이 된다!
항상 배우고 익혀야 한다!

Q & A

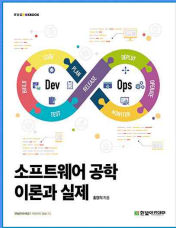
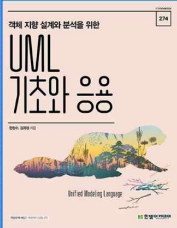
 한국과학기술대학교

이론 2장 소프트웨어 품질

53




UNIFIED
MODELING
LANGUAGE™



[GSE 2주차-1] 수고했습니다!
다음 시간에도 반갑게 만납시다~~

이 과목에서 사용되는 일부 자료, 영상물 등은 강의 내용을 보충하기 위해 교육 목적으로 활용하였습니다. 이 과목의 강의 자료 및 영상물의 불법적 이용, 무단 전재·배포는 법적으로 금지되어 있으니, 학생 여러분은 학습 외 용도의 사용을 주의하시기 바랍니다.

 한국과학기술대학교

이론 2장 소프트웨어 품질

54

Copyrighted by 석진원(2024-2학기)

27