

1. 소프트웨어공학 강의 목차 2. 기출문제 출제경향 분석 ('99 ~ '08)

SE01-소프트웨어공학, 생명주기

12%

SE02-프로젝트 관리

24%

SE03-구조적 개발 방법론

36%

SE04-객체지향 개발 방법론

15%

SE05-발전적 주제

13%

1. 소프트웨어 공학(Software Engineering) 이해하기 ★★☆☆☆

* 정의 : 가장 경제적으로 신뢰도 높은 S/W를 만들기 위한 방법, 도구와 절차들의 체계화 한 학문
- 프로그램 코딩(X) -> 프로그램을 잘 만드는 방법 연구 -> 목표 : 투자 대비 성능 우수

* 나는 개발부 팀장입니다. 조금 전, 고객만족팀으로부터 고객관리프로그램 개발 의뢰를 받았습니다.
어떻게 프로그램을 개발해야 할까요?

* S/W (프로젝트) 개발 절차 : 요구 분석 -> 설계 -> 구현(코딩) -> 테스트(시험) -> 유지보수

1) 요구 분석 : 어떻게 만들어 줄까? (무엇)

-> 분석 도구

2) 설계 : 요구 분석 결과를 가지고 구체적인 기능과 구조를 체계화 (어떻게)

-> 설계 기법

3) 구현 : 프로그램 언어를 선정하고, 설계 명세서를 컴퓨터가 이해할 수 있도록 표현

-> 프로그램 언어 선정 기준, 코딩 표준화

4) 테스트 : 요구 사항에 맞게 작동하는가?

-> 테스트 기법

5) 유지보수 : 버전 업데이트 및 새로운 기능 추가 (S/W 개발 비용 70% 차지)

-> 유지보수 과정 (유지보수 요구 -> 현 시스템 이해 -> 수정, 테스트)

2. 소프트웨어 위기 ★★☆☆☆

1) 정의 : 소프트웨어 개발 속도가 하드웨어 개발 속도를 따라가지 못해 소프트웨어에 대한 사용자들의 요구 사항을 처리할 수 없는 문제가 발생함을 의미
-> 소프트웨어 공학이 나타나게 된 배경

2) 위기의 결과

- 개발인력의 부족 -> 인건비 상승 -> 개발기간 지연 및 개발 비용 증가
- 성능 및 신뢰성 부족 -> 품질 저하
- 유지보수의 어려움

3) 좋은 소프트웨어의 조건

- 남이 알아보기 쉬워야 한다.
- 경제적, 문서화가 잘 되어 있어야 한다.
- 독창적 (X)

4) 소프트웨어 공학의 기본 원칙

- 현대적인 프로그래밍 기술 적용
- 지속적인 검증 시행
- 결과에 대한 명확한 기록 유지
- 충분한 인력 투입 (X)

1. 소프트웨어 생명 주기

1) 정의 : 소프트웨어를 개발하기 위해 정의, 개발, 유지보수 과정을 각 단계별로 나눈 것
- 표현 형태 : 폭포수 모형, 프로토타입 모형, 나선형 모형, 4GT

2) 소프트웨어 생명 주기 단계

- ① 정의단계 : 타당성 검토 단계, 계획 단계, 요구사항분석 단계
- ② 개발단계 : 설계 단계, 구현 단계, 테스트 단계
- ③ 유지보수단계 : 가장 비용이 많이 요구되는 단계

3) 역할

- 프로젝트 비용산정과 개발 계획수립의 기본 골격
- 프로젝트 진행방향을 명확히 한다
- 용어, 기술의 표준화 가능 -> 일관성 유지
- 문서화가 충실한 프로젝트 관리 용이
- 단계별 종료 시점은 변동될 수 있음 (명확 X)

[SE 1강]-소프트웨어 공학, 생명주기

2. 폭포수 모형 : 순차적(고정) -> 요구분석 불만족



1) 개발 단계

: 타당성 검토 -> 계획 -> 요구 분석 -> 설계 -> 구현(코딩) -> 시험(검사,테스트) -> 운용 -> 유지보수

2) 특징

- 가장 오래되고 폭넓게 사용된 전통적인 소프트웨어 생명주기
- 물이 위에서 아래로 떨어지듯이 단계가 순차적으로 진행되고 단계별 정의가 분명
- 두 개 이상의 과정이 병행 수행되거나 이전 단계로 넘어가는 경우가 없음
- 개발 과정 중에 발생하는 새로운 요구나 경험을 설계에 반영하기 어려움 (요구사항 변경 X)
- 제품의 일부가 될 매뉴얼 작성 필요
- 각 단계가 끝난 후 결과물이 명확히 나옴

* 사용자의 요구사항 분석 작업이 어려운 이유

- 개발자와 사용자 간의 지식이나 표현의 차이가 커서 상호 이해가 쉽지 않다.
- 사용자의 요구사항이 모호하고 부정확하며, 불완전하다.
- 개발하고자 하는 시스템 자체가 복잡하다.

[SE 1강]-소프트웨어 공학, 생명주기

3. 프로토타입 모형 : 모형(가변) -> 요구분석 만족



* 개념 이해하기 : 모델하우스를 고객에게 보여주고 요구에 맞도록 건물을 시공한다.

1) 정의 : 시스템의 일부 혹은 시스템의 모형을 만드는 과정으로서 요구된 소프트웨어의 일부를 구현하며, 추후 구현단계에 사용될 골격코드가 되는 모형

2) 개발 단계

요구 수집 -> 빠른 설계 -> 프로토타입 구축 -> 고객평가 -> 프로토타입 조정 -> 구현

3) 특징

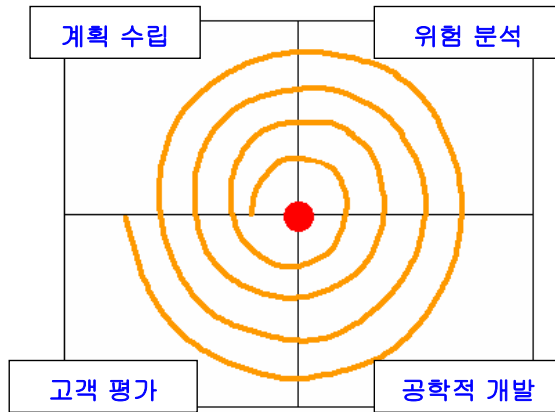
- 실제 상황이 나오기 전에 가상으로 시뮬레이션을 통해 최종 결과물에 대한 예측을 할 수 있음
- 개발 단계에서 오류 수정을 할 수 있음
- 요구사항을 충실히 반영
- 실제 개발된 시스템 견본을 미리 만들어 최종 결과물을 예측하는 모형 -> 비용 증가

[SE 1강]-소프트웨어 공학, 생명주기

4. 나선형 모형 : 폭포수 장점 + 프로토타입 장점 ★★☆☆☆

1) 개발 단계

: Planning → Risk Analysis → Engineering → Customer Evaluation



2) 특징

- 점증적 생명주기 모델
- 위험분석 단계에 초점
- Boehm(보헴) 제안
- 비용이 많이 들고, 시간이 많이 소요되지만 완성도 높으므로 대규모 프로젝트에 유리
- 개발 단계에서 유지보수 (X)

[SE 1강]-소프트웨어 공학, 생명주기

5. 4GT (4th Generation Techniques) : 4세대 기법

1) 개발 단계

: 요구사항 분석 -> 설계, 구현 -> 제품화

2) 특징

- 4세대 언어(비주얼 베이직) 이용 -> 원시 코드를 자동으로 생성
- 설계 단계 단축 -> 개발 시간 감소 (소규모 개발 시 효율적)

3) 개념 이해하기

[SE 1강]-소프트웨어 공학, 생명주기

1. 시스템의 구성 요소에 해당되지 않는 것은?

가. 입력 나. 출력 다. 제어 라. 상태

2. 컴퓨터의 발달 과정에서 소프트웨어의 개발 속도가 하드웨어의 개발 속도를 따라가지 못해 사용자들의 요구사항을 감당할 수 없는 문제가 발생함을 의미하는 것은?

가. 소프트웨어의 위기(Crisis)
나. 소프트웨어의 오류(Error)
다. 소프트웨어의 버그(Bug)
라. 소프트웨어의 유지보수(Maintenance)

3. 소프트웨어의 위기 현상과 거리가 먼 것은?

가. 유지보수의 어려움
나. 개발 인력의 급증
다. 성능 및 신뢰성 부족
라. 개발 기간의 지연 및 개발비용의 증가

4. 공학적으로 잘 작성된 소프트웨어의 특성이 아닌 것은?

가. 소프트웨어는 신뢰성이 높아야 하며 효율적이어야 한다.
나. 소프트웨어는 사용자가 원하는 대로 동작해야 한다.
다. 소프트웨어는 편리성이나 유지보수성에 점차 비중을 적게 두는 경향이 있다.
라. 소프트웨어는 잠재적인 에러가 가능한 적어야 하며 유지 보수가 용이해야 한다.

5. 좋은 소프트웨어의 조건이라고 할 수 없는 항목은?

가. 남이 알아보기 쉬워야 한다.
나. 경제적이어야 한다.
다. 문서화가 잘 되어 있어야 한다.
라. 프로그램이 독창적이어야 한다.

6. 소프트웨어 공학이 나타나게 된 배경과 관계가 먼 것은?

가. S/W 비용의 증가
나. 유지보수 비용의 감소
다. S/W 품질과 생산성의 재고
라. 특정 개인에 의존한 시스템 개발

[정답] 1.라 2.가 3.나 4.다 5.라 6.나



[SE 1강]-소프트웨어 공학, 생명주기

7. 소프트웨어 라이프 사이클 단계 중 가장 오랜 시간이 걸리며, 대부분의 비용을 차지하는 단계는?

가. 타당성 검토 단계 나. 운용 및 유지보수 단계
다. 기본 설계 단계 라. 실행 단계

8. 현재 소프트웨어 개발 중 가장 많은 비용이 요구되는 단계는?

가. 분석 나. 설계 다. 구현 라. 유지보수

9. 소프트웨어 수명 주기 모형 중 폭포수 모형(Waterfall Model)의 개발 단계로 옳은 것은?

가. 계획 - 분석 - 설계 - 시험 - 구현 - 유지보수
나. 계획 - 분석 - 설계 - 구현 - 시험 - 유지보수
다. 계획 - 설계 - 분석 - 구현 - 시험 - 유지보수
라. 계획 - 분석 - 설계 - 구현 - 시험 - 설치

10. 폭포수 모델에 대한 설명으로 옳지 않은 것은?

가. 소프트웨어 개발 과정의 각 단계가 순차적으로 진행 된다.
나. 앞 단계에서 발견하지 못한 오류를 다음 단계에서 발견 했을 때 오류 수정이 용이하다.
다. 두 개 이상의 과정이 병행 수행되거나 이전 단계로 넘어가는 경우가 없다.
라. 개발 과정 중에 발생하는 새로운 요구나 경험을 설계에 반영하기 힘들다.

11. 시스템의 일부 혹은 시스템의 모형을 만드는 과정으로서, 요구된 소프트웨어의 일부를 구현하여, 추후 구현 단계에서 사용될 골격 코드가 되는 모형은?

가. 폭포수 모형 나. 점층적 모형
다. 프로토타입 모형 라. 계획 수립 모형

12. 다음은 프로토타입(원형) 모형의 개발에 필요한 작업을 기술한 것이다. 작업 순서대로 옳게 나열한 것은?

① 빠른 설계 ② 프로토타입 구축 ③ 프로토타입 조정
④ 요구 수집 ⑤ 구현 ⑥ 고객 평가

가. ④ - ② - ① - ③ - ⑥ - ⑤
나. ④ - ① - ② - ⑤ - ⑥ - ③
다. ④ - ① - ② - ③ - ⑥ - ⑤
라. ④ - ① - ② - ⑥ - ③ - ⑤

[정답] 7.나 8.라 9.나 10.나 11.다 12.라



13. 실제 상황이 나오기 전에 가상으로 시뮬레이션을 통해 최종 결과물에 대한 예측을 할 수 있는 소프트웨어 수명 주기 모형은?

- 가. 집중적 모형(Spiral Model)
- 나. 프로토타입 모형(Prototype Model)
- 다. 코코모 모형(Cocomo Model)
- 라. 폭포수 모형(Waterfall Model)

14. 프로토타입 모형의 장점으로 가장 적절한 것은?

- 가. 프로젝트 관리가 용이하다
- 나. 노력과 비용이 절감된다.
- 다. 요구사항을 충실히 반영한다.
- 라. 관리와 개발이 명백히 구분된다.

15. 소프트웨어 수명 주기 모형 중 나선형(Spiral) 모형의 단계와 그 순서가 올바르게 구성된 것은?

- 가. Planning → Requirement Analysis → Development → Maintenance
- 나. Planning → Risk Analysis → Engineering → Customer Evaluation
- 다. Requirement Analysis → Planning → Design → Maintenance
- 라. Requirement Analysis → Risk Analysis → Development → Maintenance

16. Boehm이 제안한 나선형 모형의 태스크(Task)에 해당되지 않는 것은?

- 가. 계획 수립(Planning)
- 나. 위험 분석(Risk Analysis)
- 다. 객체 구현(Object Implementation)
- 라. 고객 평가(Customer Evaluation)

[정답] 13.나 14.다 15.나 16.다



17. 소프트웨어 생명주기(Life Cycle) 모델 중 아래 보기가 설명하는 모형은?

- a. 고객과의 의사소통(Communication)을 통해 계획수립과 위험분석, 구축, 고객평가의 과정을 거쳐 소프트웨어를 개발한다.
- b. 가장 큰 장점인 위험분석 단계에서 기술과 관리의 위험요소들을 하나씩 제거해 나감으로써 완성도 높은 소프트웨어를 만들 수 있다.
- c. 반복적인 작업을 수행하는 점증적 생명주기 모델이다.
- d. 비용이 많이 들거나 시간이 많이 소요되는 대규모 프로젝트나 큰 시스템을 구축할 때 유리하다.

- 가. 프로토타입(Prototype) 모델
- 나. 폭포수(Waterfall) 모델
- 다. 나선형(Spiral) 모델
- 라. RAD 모델

18. 프로토타이핑 모형(Prototyping Model)에 대한 설명으로 옳지 않은 것은?

- 가. 최종 결과물이 만들어지기 전에 의뢰자가 최종 결과물의 일부 또는 모형을 볼 수 있다.
- 나. 개발단계에서 오류 수정이 불가하므로 유지보수 비용이 많이 발생한다.
- 다. 프로토타입은 발주자나 개발자 모두에게 공동의 참조 모델을 제공한다.
- 라. 프로토타입은 구현단계의 구현 골격이 될 수 있다.

19. 다음 중 전통적인 소프트웨어 개발 방법론이 폭포수형(waterfall) 모델에서 개발 순서가 옳은 것은?

- 가. 타당성 검토 → 계획 → 분석 → 구현 → 설계
- 나. 타당성 검토 → 분석 → 계획 → 설계 → 구현
- 다. 타당성 검토 → 계획 → 분석 → 설계 → 구현
- 라. 타당성 검토 → 분석 → 계획 → 구현 → 설계

[정답] 17.다 18.나 19.다

