

IEEE 829-2008 중심 소프트웨어 테스트 문서 최적화 연구

Study of optimal software test
documentation through IEEE 829-2008



컴퓨터정보통신공학과
박정훈
xelion@korea.ac.kr

- **배경과 목적** : 분산화, 경량화 되어가는 소프트웨어 애플리케이션의 추세에 따라, 소프트웨어 프로젝트 문서 중 테스트 관련 문서 영역도 경량화 될 필요가 존재.
- **연구 결과** : 프로젝트 전체의 결과 문서는 정상적으로 사용하고 있으나, 이를 계획하는 활동은 미흡함. 또한 과정 도출을 위한 도구 사용에 대한 기대는 높으나, 이를 기록하고 관리하는 문서에 대한 효용은 낮게 평가됨.
- **연구 성과** : 소프트웨어 개발 시 필요한 테스트 문서의 최적화로 SW개발 중 필요한 곳에 집중할 수 있는 문서화를 통해 소프트웨어 품질을 개선.

연구의 배경

▣ 소프트웨어 공학의 연구 기반

- Mission Critical Project
- Safety Critical System
- ➔ 대규모 자본 및 기술, 위험관리 기반

▣ 최근의 시류

- 스마트폰의 등장이 후 Mobile 시장의 Big-bang
- 저용량 Device에서 운용 가능한 소프트웨어의 강세
- 클라우드 컴퓨팅 등 정보관리 체계의 변화
- 경량화된 개발방법론들의 대두
- ➔ SW공학의 경량화 추세

연구의 목적

▣ 소프트웨어 문서 작성의 필요성

- 위험 절감
- 의사소통 낭비 감소
- 재 작업 감소

▣ 소프트웨어 테스트 문서화에 대한 연구

- 외국의 SW공학자들도 필요성은 인식, 사례 공유 존재
- **SW테스트 문서 관련 정량화된 연구 결과가 없음**
- ➔ SW 테스트 영역을 (1)계획, (2)도구와 절차, (3)결과와 보고서라는 세 영역으로 나누어 현황과 개선안 도출

▣ 문서화

- 인간이 개발한 의사소통 접근 방법 중 가장 오래되고 확실한 접근 방법

▣ 소프트웨어 문서화

- SW시스템과 관련 사항에 대한 체계적인 의사소통을 목적으로 산출
- 명확히 고려된 계획을 통해 산출 필요
- 프로젝트 전체에 투입되는 비용과 노력이 감소

▣ 소프트웨어 문서화 기존연구 및 사례

- 모든 소프트웨어 개발 시 문서화가 필요함이 이전 연구를 통해 증명됨 : 오픈 소스, 엔터프라이즈 시스템, 유지보수 등
- 최근의 시류인 애자일 영역에서도 소프트웨어 문서화가 투자자들의 이익을 위해 중요한 산출물임이 증명됨
- 우리나라에서는 국가 프로젝트 중 소프트웨어 테스트 활동의 체계화를 통해 프로젝트 전체 비용이 감소된 사례가 있음.

▣ 연구 필요 부분

- 소프트웨어 개발생명 주기 동안 발생하는 전체 문서를 통한 품질 향상에 대한 연구는 있으나 테스트 영역에 대한 연구는 미흡

▣ IEEE 829-2008

(Standard for Software Test Documentation)

- 소프트웨어 테스트에 관련된 문서들을 설명하고, 해당 문서들이 가지고 있어야 하는 내용들에 대해 목차로 가이드라인 제시
- ISEB 등 소프트웨어 테스트 지식 체계에서도 IEEE 829의 내용을 참고

▣ IEEE 829-1998 vs. IEEE 829-2008

- IEEE 829-1998 이후 10년 만에 개정
- IEEE 829-2008에서는 소프트웨어 개발생명주기 전체의 테스트 지침과 방법을 요약하여 기술하는 Master Test Plan 문서 추가
- 기존에는 Test Plan을 하나로 작성하도록 가이드 하였으나, Master Test Plan 및 각 개발생명주기 단계별 Level Test Plan을 분리
- 결과 보고서도 Master Test Report와 Level Test Report로 분리
- 그 외 사항들도 전체와 단계로 분리
- 용어 변경 : 다수
예) Incident report → Anomaly report

▪ Software & System Integrity Level

- IEEE 829-2008에서는 모든 소프트웨어에 똑같은 수준의 문서화를 요구하기보다는, 위험관리 기법을 토대로 무결성 단계를 측정하여 소프트웨어의 성격에 맞는 문서화 수준을 수행할 것을 권고.

▪ 연구 모형

- 기존문헌 및 관련연구를 기반으로 자료 수집 및 설문 조사 수행을 통한 연구 접근방안.

소프트웨어와 시스템 무결성 단계 설명

4단계 - 대재앙(Catastrophic)

소프트웨어의 오류로 발생하는 결과로 인해 완화 불가능한 수준의 대재앙을 불러오는 경우. (생명과 연관된 고장, 시스템 고장, 환경에 영향을 주는 고장, 경제나 사회적 물의를 일으키는 고장 등)

3단계 - 치명적임(Critical)

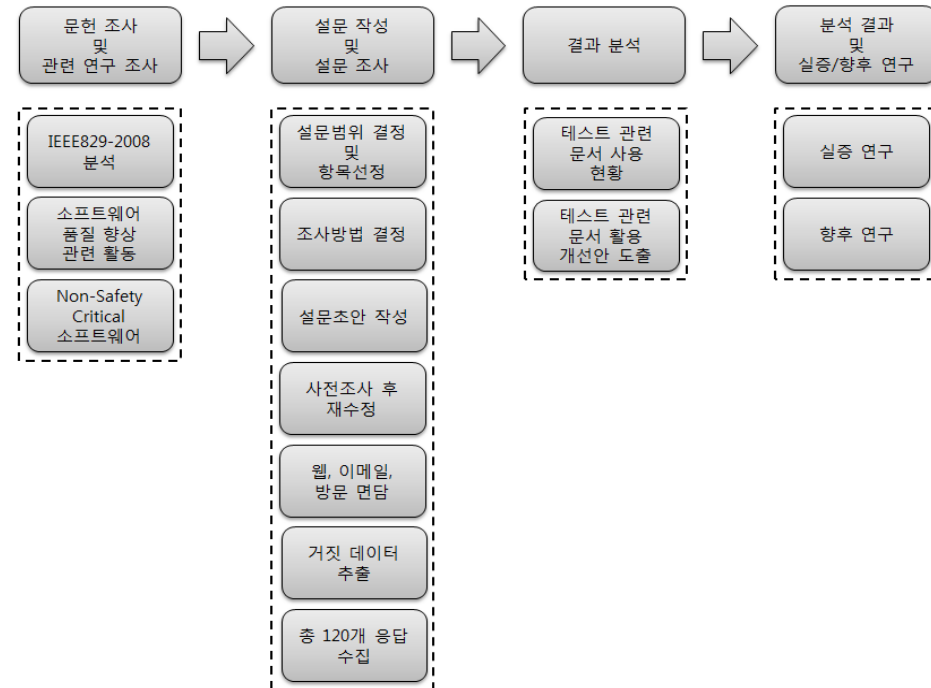
소프트웨어의 오류로 발생하는 결과로 인해 치명적인 수준의 결과가 나타나는 경우. 위험에 대한 부분적인 완화가 가능하지만, 완전한 완화는 불가능한 경우. (영구 손상, 주요 시스템 저하, 환경 피해, 경제나 사회적 영향 등)

2단계 - 보잘것없음(Marginal)

소프트웨어의 오류로 발생하는 결과로 인해 의도한 기능들이 작은 결함을 보이는 경우. 위험관리를 통해 완전한 보완이 가능한 경우.

1단계 - 하찮음(Negligible)

소프트웨어의 오류로 발생하는 결과들이 하찮은 수준의 결함인 경우. 위험관리를 통한 보완책이 필요하지 않은 경우.



설문 조사 방식 및 질문의 구성

- 구글 문서 도구 활용.
- 자기기입식 방식으로 진행.
- 양자택일식 질문으로 구성.
- 테스트 분야의 전문적인 용어는 설문 문항에 주석으로 설명을 기입.

항목 구분

테스트 계획

- Master Test Plan
- Level Test Plan
- Level Test Design

테스트 도구와 절차

- Level Test Case
- Level Test Procedure

테스트 결과와 보고서

- Level Test Log
- Anomaly Report
- Level Interim Test Status Report
- Level Test Report
- Master Test Report

조사 대상 선정 및 거짓 데이터 제거

- 총 154개의 데이터 중 34개의 데이터 제거. 120개의 데이터를 분석에 활용.

사전 조사 활동

- SW개발 분야 12년 이상의 경력을 보유한 전문가 3인으로부터 설문 문항에 대한 사전 검증을 통해 설문 문항 수정.

→ **문항의 전문성** : SW 테스트 분야의 전문성으로 인해 설문 문항이 어렵다는 지적이 있어 전문 용어들에 대한 설명 추가.

→ **설문 결과 예측 시 신뢰성 확보** : 설문 문항의 전문성 때문에 응답자가 설문 문항을 정확히 이해하고 응답하였다고 보기 어려우므로, 양자택일 방식으로 질문 구성.

→ **거짓 데이터 추출 기준** : 거짓데이터를 추출할 수 있도록 기준을 수립하여 추가적인 질문을 구성.

조사 기간

- 2012년 7월~8월 : 약 60일

연구 목표의 설정

참조 가설 부재

▪ 소프트웨어 개발 프로젝트 전체의 문서화는 기존 연구가 존재하지만, 테스트 영역만 따로 분리하여 정량적인 결과를 도출한 연구는 존재하지 않으므로, 참조 가설 없음.

목표 설정

▪ 현업 실무자들을 대상으로 한 설문으로 SW테스트 문서 사용 현황 및 개선안에 대해 도출.

| # | 연구 목표 | 분야 |
|---|---|--------|
| 1 | 테스트 활동에 대한 문서를 작성하고 있는가 | 현황 분석 |
| 2 | 향후 테스트 문서 표준을 활용할 경우 프로젝트에서의 소프트웨어 품질 개선될 것이라 생각하는가 | 개선안 제시 |

※ 이 후의 표는 모두 이 목표를 기반으로 수행된 내용.

응답자 분석

| 구분 | 범위 | 빈도 | 비율 |
|----------------------|--------------|-----|------|
| 직종 분류 | 품질관리자 | 55 | 46% |
| | 프로젝트 관리자 | 21 | 18% |
| | 프로그래머 | 28 | 23% |
| | 기타 (전산공무원 등) | 7 | 6% |
| | 컨설턴트 | 5 | 4% |
| | 선계자 | 4 | 3% |
| 회사 개발 | 합계 | 120 | 100% |
| | 30명 이하 | 44 | 44% |
| | 31명~50명 | 18 | 18% |
| | 51명~100명 | 8 | 8% |
| | 101명 이상 | 11 | 11% |
| | 합계 | 82 | 100% |
| 현재 프로젝트 개발 참여자의 수 | 10명 이하 | 65 | 54% |
| | 11명~30명 | 29 | 24% |
| | 31명~50명 | 12 | 10% |
| | 51명~100명 | 7 | 6% |
| | 101명 이상 | 7 | 6% |
| | 합계 | 120 | 100% |
| 단일 프로젝트를 위한 테스트 팀의 수 | 0명 | 45 | 37% |
| | 1명 | 9 | 7% |
| | 2명~5명 | 45 | 37% |
| | 6명~10명 | 14 | 12% |
| | 11명~20명 | 4 | 4% |
| | 21명 이상 | 3 | 3% |
| | 합계 | 120 | 100% |
| 업종 구분 | SI | 43 | 36% |
| | SI성 SM | 19 | 16% |
| | SM | 11 | 9% |
| | 게임 | 11 | 9% |
| | Mobile 관련 | 9 | 8% |
| | 포털&웹 | 7 | 6% |
| | 공공&국방 | 7 | 6% |
| | QA&테스팅 | 3 | 3% |
| | 교육 | 3 | 3% |
| | 기타 | 4 | 3% |
| | 합계 | 120 | 100% |

「중소기업법」을 기초로 하여, 30명 이하 팀을 중소기업 규모 소프트웨어를 개발하는 팀이라고 가정하고 분석 진행

테스트 계획

▣ 테스트 계획

- 소프트웨어 개발 생명주기에서 테스트 계획을 논리적으로 설계하여 기술해두는 문서.
- 테스트 활동은 수행도 중요하지만, 테스트를 올바르게 계획하는 것이 버그를 예방할 수 있는 가장 좋은 방법.

▣ 테스트 계획 분야로 분류한 문서

- 최상위 테스트계획.
- 단계별 테스트 계획.
- 단계별 테스트 기획.

테스트 계획 문서 현황

| | 예 | 아니오 | 모르겠음 |
|-----------------------------------|-----|-----|------|
| 품질관리계획 (Quality Plan) | 52 | 35 | 7 |
| | 55% | 37% | 7% |
| 최상위 테스트계획 (Master Test Plan) | 17 | 68 | 9 |
| | 18% | 72% | 10% |
| 단계별 테스트계획 (Level Test Plan) | 36 | 51 | 7 |
| | 38% | 54% | 7% |
| 단계별 테스트 기획 (Level Test Design) | 24 | 62 | 8 |
| | 26% | 66% | 9% |

※ 이 중 Quality Plan은 PMBOK의 영역이나,
Master Test Plan과의 차이를 확실히 하기 위해 설문에 삽입

테스트 계획 문서 개선안

| | 예 | 아니오 | 모르겠음 |
|------------|-----|-----|------|
| 품질관리계획 | 75 | 10 | 9 |
| | 81% | 10% | 9% |
| 최상위 테스트계획 | 55 | 8 | 31 |
| | 59% | 9% | 33% |
| 단계별 테스트계획 | 80 | 3 | 11 |
| | 85% | 3% | 12% |
| 단계별 테스트 기획 | 49 | 4 | 41 |
| | 52% | 4% | 44% |

결론 : 테스트 계획 문서

▣ 현황 파악

- 프로젝트 전체의 계획은 작성하지만 테스트 활동에 대한 계획은 거의 하지 않음.

▣ 개선안 제시

- 단계별 테스트 문서 작성이 필요.
- 최상위 테스트 계획은 Software and System Integrity Level이 확보된 후 수행되어야 하는 부분
이므로, 이를 적용하기 위한 추가적인 연구가 필요.
- 단계별 테스트 기획은 단계별 테스트 계획을 세 부적으로 기술하는 문서로 하나로 통합 가능.

테스트 도구와 절차

▣ 테스트 도구와 절차

- 소프트웨어 테스트 활동을 위해 필요한 접근 방법이나 도구 사용 등을 기술해 두는 문서.
- 최근의 테스트케이스는 테스트 항목 이상의 의미를 담고 있으며, 테스트에 필요한 사항, 수행 절차, 기대결과 등 테스트케이스 하나로 시스템에 필요한 테스트 하나를 수행할 수 있도록 구성.

▣ 테스트 도구와 절차 분야로 분류한 문서

- 테스트 케이스.
- 테스트 절차.

테스트 도구와 절차 문서 현황

| | 예 | 아니오 | 모르겠음 |
|----------------------------|-----|-----|------|
| 테스트케이스 (Test case) | 54 | 36 | 4 |
| | 57% | 38% | 4% |
| 테스트 절차 (Test Procedure) | 28 | 59 | 7 |
| | 30% | 63% | 7% |

테스트 계획 문서 개선안

| | 예 | 아니오 | 모르겠음 |
|--------|-----|-----|------|
| 테스트케이스 | 80 | 0 | 14 |
| | 85% | 0% | 15% |
| 테스트 절차 | 74 | 2 | 18 |
| | 79% | 2% | 19% |

결론 : 테스트 도구와 절차 문서

▣ 현황 파악

- 소프트웨어의 모듈 및 컴포넌트 테스트의 커버리지 확보를 위한 테스트 케이스는 어느 정도 사용하고 있으나, 이를 체계적으로 수행하기 위한 절차는 거의 사용하지 않음.

▣ 개선안 제시

- 체계적인 테스트 케이스 사용을 통해 소프트웨어 품질 향상 필요.
- 체계적인 테스트 절차를 이용하여 소프트웨어 품질 향상 필요.

테스트 결과와 보고서

▣ 테스트 결과와 보고서

- 소프트웨어 테스트 활동을 통해 도출된 결과들을 기록하고 요약해 두는 문서

▣ 테스트 도구와 절차 분야로 분류한 문서

- 테스트 로그
- 예외 보고서
- 테스트 중간 상태 보고서
- 테스트 결과 보고서

※ IEEE 829-2008에는 결과 보고서가 Master와 Level로 나뉘어 있지만, 전문가 검토를 통해 “따로 작성하는가”에 대한 질문만 하는 것이 낫다고 판단.

테스트 결과와 보고서 현황

| | 예 | 아니오 | 모르겠음 |
|---|-----|-----|------|
| 테스트 로그 (Test Log) | 24 | 67 | 3 |
| | 26% | 71% | 3% |
| 예외보고서 (Anomaly Report) | 40 | 49 | 5 |
| | 43% | 52% | 5% |
| 테스트 중간 상태 보고서 (Level Interim Test Status Report) | 15 | 73 | 6 |
| | 16% | 78% | 6% |
| 테스트 결과보고서 (Test Report) | 52 | 37 | 5 |
| | 55% | 39% | 5% |
| 단계별 보고서와 최상위 결과 보고서 분할 | 9 | 74 | 11 |
| | 10% | 79% | 12% |

테스트 결과와 보고서 개선안

| | 예 | 아니오 | 모르겠음 |
|---|-----|-----|------|
| 테스트 로그 (Test Log) | 53 | 26 | 15 |
| | 56% | 28% | 16% |
| 예외보고서 (Anomaly Report) | 65 | 10 | 19 |
| | 69% | 11% | 20% |
| 테스트 중간 상태 보고서 (Level Interim Test Status Report) | 47 | 21 | 27 |
| | 49% | 22% | 28% |
| 테스트 결과보고서 (Test Report) | 83 | 2 | 9 |
| | 88% | 2% | 10% |
| 단계별 보고서와 최상위 결과 보고서 분할 | 26 | 11 | 57 |
| | 28% | 12% | 60% |

결론 : 테스트 결과와 보고서

▣ 현황 파악

- 테스트 결과를 도출하기 위한 중간 상태의 문서들을 거의 작성하지 않음.
- 결과보고서를 분할하지 않고, 수행 내역에 대한 결과보고서만 작성하는 것으로 나타남.

▣ 개선안 제시

- 테스트 결과보고서 사용을 통해 소프트웨어 품질 향상 필요.
- 추가적으로, 예외 보고서는 소프트웨어 품질을 체계적으로 개선하도록 해 준다는 기존의 연구가 있으므로 사용을 고려해야 함.

그 외 항목

▣ 추가 질문

- 전문가 검토를 통해 도출된 문제점인 전문성, 신뢰성 확보, 거짓데이터 추출을 위해 추가한 항목
- 묻고자 하는 질문과 배치되는 응답이 나오거나, 선행되지 않으면 현실적으로 활동을 수행하기 어려운 문항들을 고려하여 문항 구성.

▣ 추가 문항으로 분류한 부분

- 테스트 위험관리
- 테스트 도구
- 테스트 형상관리
- 테스트 결과와 보고서 분석

그 외 항목 현황

| | 예 | 아니오 | 모르겠음 |
|-------------------------|-----|-----|------|
| (테스트 활동에 대한) 위험 관리 | 19 | 65 | 10 |
| | 20% | 69% | 11% |
| 테스트 도구 | 32 | 58 | 4 |
| | 34% | 62% | 4% |
| (테스트 문서의) 형상관리 | 26 | 63 | 5 |
| | 28% | 67% | 5% |
| 예외보고서의 이슈에 대한 영향도 분석 | 30 | 54 | 10 |
| | 32% | 57% | 11% |
| (결과보고서 기반) 근본 원인 분석 | 34 | 52 | 8 |
| | 36% | 55% | 9% |
| 교훈 공유 (Lessons Learned) | 39 | 48 | 7 |
| | 41% | 51% | 7% |

그 외 항목 개선안

| | 예 | 아니오 | 모르겠음 |
|----------------------|-----|-----|------|
| (테스트 활동에 대한) 위험 관리 | 64 | 9 | 21 |
| | 68% | 10% | 22% |
| 테스트 도구 | 74 | 7 | 13 |
| | 79% | 7% | 14% |
| (테스트 문서의) 형상관리 | 55 | 17 | 22 |
| | 59% | 18% | 23% |
| 예외보고서의 이슈에 대한 영향도 분석 | 70 | 10 | 14 |
| | 74% | 11% | 15% |

결과 : 그 외 항목

▣ 현황 파악

- 테스트 결과를 도출하기 위한 중간 상태의 문서들을 거의 작성하지 않음.
- 테스트 활동에 대한 기술적/관리적인 활동은 거의 수행하지 않는 것으로 나타남.

▣ 개선안 제시

- 테스트 도구를 사용하여 소프트웨어 품질 향상을 위한 개선 필요
- 예외 보고서의 이슈에 대한 영향도를 분석하는 작업을 추가하여 소프트웨어 품질 향상을 위한 개선 필요

결론

▣ 연구 결과

- 테스트 활동 현황 : 수행에만 집중됨.
- 계획 수립과 중간 산출물에 대한 도출 활동이 부족.
- 테스트 도구나 절차를 계획하지 않음.
- 결과 보고서 작성은 하지만, 결과를 도출해 내는 과정에 대한 개선은 배제됨.
- 개선을 위해 아래의 문서/활동이 필요.

| 구분 | 문서 | 현황 | 개선안 | 개선 필요 |
|-------------|------------|-----|-----|-------|
| 테스트 계획 | 품질관리계획 | 55% | 81% | ○ |
| | 단계별 테스트계획 | 38% | 85% | ○ |
| 테스트 도구와 절차 | 테스트케이스 | 57% | 85% | ○ |
| | 테스트 절차 | 30% | 79% | ○ |
| 테스트 결과와 보고서 | 예외 보고서 | 43% | 69% | ○ |
| | 테스트 결과 보고서 | 55% | 88% | ○ |

- 이 결과는 IEEE 829-2008에서 권고하는 Software and System Integrity Level 중 1단계에도 미치지 못하는 수준으로, 현재 우리나라의 소프트웨어 개발팀은 국제 표준에 못 미치는 문서화 역량을 보유한 것으로 나타남.
- 문서화를 통해 프로젝트 전체의 비용이 감소함을 인식하고 체계화된 테스트 문서 작성을 위한 노력이 필요함.

연구의 한계 및 향후 연구

▣ 연구의 한계

- 프로젝트 규모나 팀의 성숙도, 특정 도메인 분야의 테스트 문서 사용은 고려 안됨. 연구의 결과가 특정 상황을 배제한 일반적인 테스트 현황만을 대변하는 한계가 존재함.

▣ 향후 연구

- 최적화된 테스트 문서를 활용하여 SW품질이 향상되는 지에 대한 추가 연구 필요.
- 도메인 별, 팀 성숙도 별 연구도 필요.

주요 참고문헌

※ 발표용 자료에서는 주요 문헌만 기재

IEEE 829-2008
 IEEE 829-1998
 IEEE Explorer.org
 Documentation Engineering
 Software testing 2nd edition
 SWEBOK
 PMBOK
 Wikipedia

