

실무 테스트 품질 개선을 위한 실증적 연구

박준호*, 박진호**

*숭실대학교 컴퓨터학과

**숭실대학교 소프트웨어학과

e-mail:park1058@ssu.ac.kr

An Empirical Study for Quality Improvement in Practical Testing

Joon-Ho Park*, Jin-Ho Park**

*Dept of Computer Science, Soongsil University

**Dept of Software engineering, Soongsil University

요 약

소프트웨어 개발 사업에서 가장 중요한 부분이 시스템 테스트이다. 이를 위해서는 테스트 방법, 테스트 기간, 절차 등이 확보가 되어야 한다. 하지만, 실무에서는 테스트 절차를 이행하는데 많은 문제점을 가지고 있다. 이를 위해 테스트 문제점과 이를 보완할 수 있는 절차를 본 논문에서 제시한다.

1. 서론

소프트웨어 개발 사업 및 솔루션 구축 사업에서 고객에게 인도되기 전에 수행하는 것이 테스트이다. 일반적인 실무 프로젝트에서는 테스트 검증을 위해 단위 테스트, 시스템 테스트, 통합 테스트를 진행한다. 하지만, 테스트가 중요한 프로젝트의 경우에는 테스트 시나리오가 다양하여 테스트 기반이 충분히 확보되지 않으면, 안정적인 시스템을 오픈하는 것이 어렵다. 복잡한 서비스를 하는 시스템의 경우에는 많은 문제점을 도출할 수 있다. 본 논문에서 실무에서 테스트 품질을 높이기 위한 절차를 제시한다.

2. 관련연구

본 논문에서는 테스트 방법론과 실무에서 사용하는 테스트 수행 절차를 비교하여 이론에서 제시하는 것과 실무에서 적용하는 것에 대한 문제점을 도출하고 이를 보완할 수 있는 절차를 도출한다.

2.1 V-Model

V-Model은 그림 1과 같이 구성된다. V-Model은 소프트웨어 개발 프로세스를 폭포수 모델의 확장된 형태 중 하나로 볼 수 있다. V-Model은 검증(Verification) 단계와 유효화(Validation) 단계로 구분할 수 있다. 본 논문에서는 유효화(Validation) 단계를 분석한다. V-Model의 유효화(Validation) 단계에는 단위 테스트, 통합 테스트, 시스템 테스트, 인

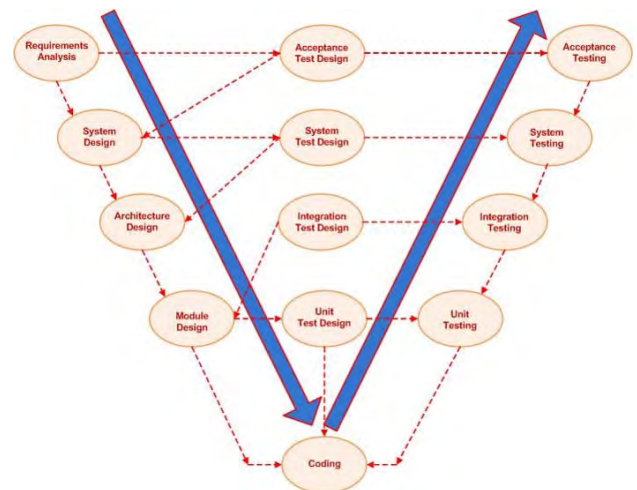


그림 1 V-Model

수 테스트로 진행된다. 단위 테스트(화이트 박스 테스트)는 모듈 설계 단계에서 준비된 테스트 시나리오를 이용하여, 코드를 개발한 개발자가 직접 수행한다. 통합 테스트(블랙박스 테스트)는 각각의 개발된 모듈을 통합하여, 통합된 컴포넌트 간의 인터페이스와 상호작용 상의 오류를 발견하는 작업을 수행한다. 통합 테스트는 코드를 직접 확인하는 형태는 아니며, 테스트 시나리오를 기반으로 개발자가 수행한다. 시스템 테스트는 실제 구현된 시스템과 계획된 사양(Specification)을 서로 비교하는 작업이다. 이를 위해 자동화 도구를 이용하며, 시스템 레벨의 에러(Error, Defect)들이 이 테스트 단계를 통해 많이 발견된다. 테스트 수행은 개발자와 별도의 다른

팀을 통해 수행한다. 인수 테스트는 시스템이나 시스템의 일부 또는 특정한 비기능적(Non-Functional)인 특성에 대해 확신(Confidence)을 얻는 단계이다. 인수 테스트는 시스템을 배포하거나 실제 운영이 가능한 사항인지에 대한 평가를 하는 것이다. 인수 테스트의 전형적인 형태는 사용자 인수 테스트, 운영상의 인수 테스트 등으로 구분된다. V-Model은 시스템 검증과 테스트를 강조한 모델이다. V-Model을 사용할 경우 오류를 줄일 수 있는 장점이 있다. 하지만 단점으로는 반복적인 테스트가 아닌 단계별 테스트 수행으로 변경 사항이 발생할 경우 처음 단계부터 다시 진행해야 하는 문제점이 있다.

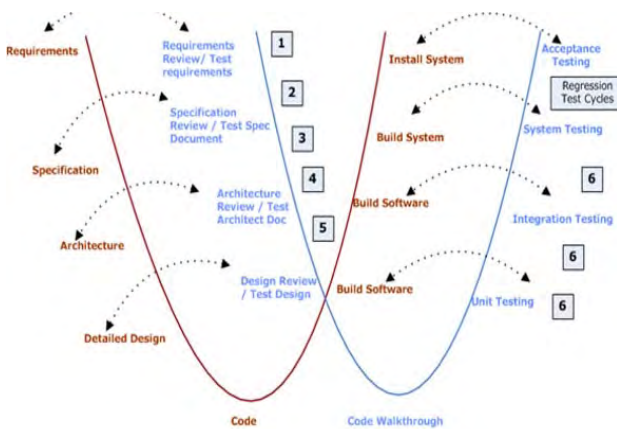


그림 2 W-Model

2.2 W-Model

W-Model은 V-Model에서 수행하지 않는 정적 테스트를 제시하고 있다. 그림 2의 W-Model은 V-Model과는 다른 수행 절차를 제시하고 있다. W-Model에서는 6개의 다른 포인트를 제시하고 있다.

포인트 1은 테스트 계획 및 테스트 전략을 수립한다. 포인트 2는 테스트 시나리오를 설계한다. 포인트 3과 4는 설계 문서와 기능 사양 명세서 (Specification)를 바탕으로 테스트 케이스를 준비한다. 포인트 5는 테스트 케이스에 대한 검토 및 검토 의견에 대해 업데이트를 진행한다. W-Model은 위 5가지 포인트를 통해 정적 테스트를 수행한다. 마지막으로 포인트 6은 다양한 테스트 방법론(예 : 단위/통합/시스템, 경로 테스트, 보안 테스트, 사용자 테스트, 성능 테스트 등)을 제시하고 있다. W-Model은 구조화된 테스트 절차라 할 수 있다.

2.3 실무 테스트 기법

실무에서 테스트하는 방법은 그림 3과 같이 수행

한다. 일반적으로 실무에서는 장비가 납품되는 경우, 솔루션이 납품되는 경우, 소프트웨어 개발이 된 경우에 따라 사전에 테스트 하는 방법이 달라진다. 예로 들어 장비 납품의 경우에는 장비 입고 및 확인 작업이 수행된다. 각 장비별로 정상 동작이 되는지에 대한 부분을 확인한다. 솔루션의 경우에는 각 지정된 하드웨어 설치하여 정상적으로 동작이 되는지 확인한다. 본 논문에서는 소프트웨어 개발에 대한 테스트 절차를 분석한다. 실무에서는 V-Model를 사용한다.

단위 테스트는 단위 기능별 테스트를 수행한다. 수행 항목으로는 기능성(Functionality), 사용성(Usability), 신뢰성(Reliability) 테스트를 단위 기능 중심으로 테스트를 한다. 통합 테스트는 업무 프로세스별 테스트를 진행한다. 통합 테스트는 시나리오에 정의된 프로세스 및 테스트 케이스 바탕으로 진행하며, 통합 프로세스 관점의 완전성 테스트이다.

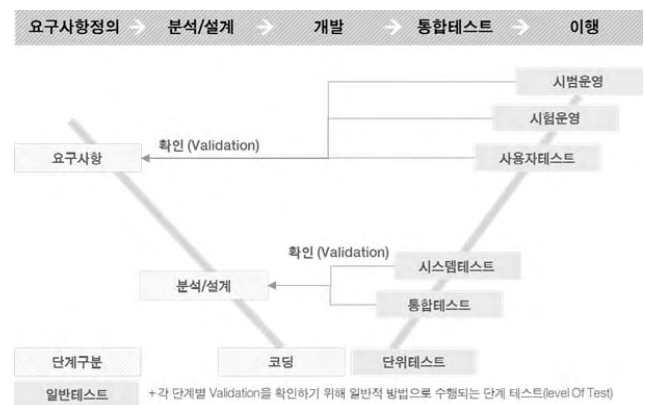


그림 3 실무 테스트 모델

2.4 실무 현황 분석

실무에서 진행하는 프로젝트를 분석하여 테스트 절차를 검토 했다. 현재 분석한 실무 프로젝트의 경우에는 일반적인 모델에서 제시하는 사항과는 다르게 수행이 되고 있다. 일반적으로 통합 테스트 시나리오 작성은 분석, 설계 업무를 작업한 담당자가 작성해야 하지만, 현장 실무에서는 개발자가 작성하고 있다. 이는 본 기능을 개발한 개발자가 시나리오를 작성하여 오류율을 최소화 할 수 있는 시나리오를 작성하게 된다. 또한 개발자의 관점에서 작성이 되기 때문에 연계 및 전체적인 시스템을 이해하지 못한 사항에서 작성되어 업무 흐름이 잘못 작성되는 경우가 발생한다.

국내의 경우에는 테스트를 진행하기 이전에 고객과의 인터뷰 및 제안 요청서에서 별도로 제시하는

테스트 성능 지표에 대해 표기하는 경우가 매우 저하다. 이로 인한 테스트 지표 및 결과에 대한 지표가 무의해 지는 경우가 발생하고 있다.

요구사항 고유번호	TER-001	
요구사항 명칭	테스트 방안	
요구사항 분류	테스트 요구사항	
요구사항 상세설명	정의	테스트 방안
	세부내용	<ul style="list-style-type: none"> 개발 및 유지보수 환경에서 사용 가능한 테스트용 라이선스 제공 웹 표준 기술 적용 완료 및 개발 또는 수정 완료한 운영 프로그램의 성능테스트를 실시 최종 개발된 프로그램은 부하테스트 및 성능시험을 통하여 운영 시스템의 부하 Peak시에 정상적인 업무처리와 기술아키텍처 및 응용 아키텍처 적정수준의 성능을 내는지를 검증한 후에 실 운영에 적용해야 한다. 단위 테스트, 통합테스트, 테스트환경, 방법, 절차 등 테스트를 위한 각 단계별 구체적인 방안을 제출 각 단계별 테스트 결과를 공단에 제출

그림 4 일반적인 테스트 요구사항

일부 공공기관에서는 그림 5와 같이 테스트 진행 사항이나 전문 테스트 인력 투입 등에 대한 구체적인 사항을 제시하고 있다.

요구사항 고유번호	TER-001
요구사항 명칭	테스트 계획 수립
분류	테스트 요구사항
요구사항 상세 설명	<ul style="list-style-type: none"> 각 단계별 시스템의 기능을 명확히 하여 타당성 있고 현실성 있는 테스트 일정을 수립·제시하여야 한다. 대상 업무별 단위테스트, 통합테스트, 시스템테스트, 인수테스트 등에 대한 방안을 제시하여야 한다. 테스트 계획서에는 테스트 인력, 테스트 데이터, 테스트 절차/방법, 테스트 일정/주기 등을 포함하여 테스트를 체계적이고 효율적으로 진행할 수 있는 방안을 제시하여야 한다. 제안서는 사용자 요구 및 프로그램 오류, 성능을 점검하는 단순 시험부터 실제 사용자 환경에서 실 업무처리에 이르기까지 종합적인 시험방법 및 시험 시나리오 등 세부 시험절차를 수립하여 제시해야 한다. 발생 가능한 상황에 대해서 실제 data오류 데이터 포합을 인력하여 테스트하여야 하며 각종 유효성 테스트 계획서를 구체적으로 작성하여 제출해야 한다. 프로그램 개발 품질을 50% 이상 진행시 질별 개발된 프로그램에 대해 단위 테스트 실시 방안을 수립하여 제시하여야 함 전문 AA(Application Architecture)가 투입되어, 개발 규칙 가이드 준수여부 등을 확인하여, 조치 결과를 평가에 제출하여야 함 준비된 테스트 시나리오에 따라 제3자 테스트가 투입되어 프로그램 기능단위 별 테스트를 진행하여야 함 크로스 브라우저(IE 7.0~11.0, FF, Safari, Opera, chrome, 모바일 등)별 테스트 계획을 수립·제시하여야 한다.

그림 5 구체적인 테스트 요구사항

테스트 품질이 낮은 프로젝트는 전문 테스트팀 조직이 없고, 요구 분석과 설계가 잘 못 경우가 있다. 또한 테스트를 위한 가이드와 시나리오 작성 가이드가 부족하다.

개발자의 인터뷰를 통해 도출된 사항으로는 테스트 시나리오 작성 방법에 대해 명확한 이해가 떨어지며, 무엇을 기반으로 작성을 해야 하는지 모르는 경우가 많았다.

실무에서 그림4와 같이 제시된 사업의 경우에는 고객 인터뷰 및 시스템 현황 분석을 통해 시스템 규모에 따른 테스트 일정 산정과 테스트 성능 지표가 필요한 요구사항의 경우에는 초기 시나리오 기반(그

림 6)으로 작성한다. 마지막으로 테스트 진행 후 에러(error)가 발생할 경우 대응 방안 에 대한 시나리오 도 작성한다.

요구사항	수강 신청 기간의 동시접속 보장
자극의 근원	최종 사용자
자 극	동시 접속 사용자가 보내는 request
대 상	대학정보 시스템
관 경	정상동작
응 답	접속처리
응답측정	동시 접속자 1,000명을 기준으로 각 사용자당 평균 6회이하의 페이지 호출 및 DB조회란 request를 발생시킬때, CPU사용율과 메모리 사용율은 매 5분 마다 측정하고 그 결과의 평균값이 50%이하를 유지한다.
비 고	

그림 6 초기 시나리오 작성 예시

3. 결론

본 논문에서는 이론적인 모델과 실무에서 수행하고 있는 모델을 분석하여 실무에서 품질 높은 테스트를 수행 할 수 있는 방안을 제시한다. 이를 위해 테스트 품질을 높이기 위한 프로세스적인 측면과 테스트 도구를 제시한다.

실무 환경 인터뷰 및 분석을 통한 결과 개발자 이해 부족이 많이 도출되었다. 이를 위한 개선 절차로 테스트 시나리오 작성 가이드를 제공한다. 시나리오와 결과서를 하나의 문서를 통해 관리 할 수 있도록 한다.

그림 7은 시스템 테스트 시나리오 및 결과서를 보여준다.

시스템 테스트 시나리오									
테스트 구분	장소	인원	목적	테스트 프로세스	작성	작성 일자	작성 부서	검정 일자	검정 부서
요구사항ID	담당자	관련 계획/일정	번호	단계별 주요수행내용	테스트 데이터	예상 수행 일자 (일정/일정/일정)	담당자	결과/비고/비고	관련 부서/테스트 일자
REQ-DEV-001	홍길동	DBO Editor Query Test	1	요구사항 분석 시 CPU 사용률, 평균 동시 사용자 수 측정 (Windows 10 기준: 메모리 2G, CPU 2G 기준)	DBO Editor Query Tester 0 명 시			CPU 사용률은 각 사용자당 평균 6회 이하를 유지하며, 메모리 사용률은 50% 이하를 유지한다.	
REQ-DEV-002	홍길동	DBO Editor Query Test	2	요구사항 분석 시 CPU 사용률, 평균 동시 사용자 수 측정 (Windows 10 기준: 메모리 2G, CPU 2G 기준)	DBO Editor Query Tester 0 명 시			CPU 사용률은 각 사용자당 평균 6회 이하를 유지하며, 메모리 사용률은 50% 이하를 유지한다.	시스템테스트 결과서
REQ-DEV-003	홍길동	DBO Editor Query Test	3	요구사항 분석 시 CPU 사용률, 평균 동시 사용자 수 측정 (Windows 10 기준: 메모리 2G, CPU 2G 기준)	DBO Editor Query Tester 0 명 시			CPU 사용률은 각 사용자당 평균 6회 이하를 유지하며, 메모리 사용률은 50% 이하를 유지한다.	

그림 7 시스템 테스트 시나리오 / 결과서

소프트웨어 개발에서 테스트 품질을 높이기 위해서는 기존의 개발 방법론을 사용하며, 고객의 제안 요청서에 명확한 지표가 없는 경우 기존의 고객 인터뷰에 추가로 그림6과 같이 기능에 대한 시나리오를 통해 테스트 방안을 도출한다.

테스트 도구는 상용 툴을 사용하는 것이 좋지만, 비용적인 측면을 고려하여 무료로 사용할 수 있는 툴을 제시한다. 실무에서 가장 많이 사용하는 테스트 툴은 JMeter와 JUNIT 툴이다. 이와 함께 HttpUnit,

LoadUI, SoapUI를 제시한다. LoadUI는 웹 서비스의 로드 테스트에 사용할 수 있다. 또한 SoapUI툴과도 연동이 가능하다. SoapUI는 웹 서비스의 호출, 개발, 시뮬레이션, 부하, 보안 테스트가 가능하다. 본 툴은 윈도우, 리눅스, Mac OS, 유닉스 환경에서 사용이 가능하다.

본 논문에서는 프로젝트 규모 및 특성을 고려하여 무료 툴을 실무에서 사용하였으며, 기존 상용 제품 (LoadRunner)을 대체가 가능했다. 대규모 시스템이 아닌 경우에는 무료 툴로 가능하며, 회사 테스트 조직 운영을 위해서 무료 툴을 기반으로 운영이 된다면 프로젝트 특성에 맞는 테스트가 가능하며 이는 테스트 품질 개선 효과로 이어질 수 있다.

참고문헌

- [1] Lind J., Iterative software engineering for multiagent systems : the MASSIVE method, vol. 1994. Berlin : Springer, 2001
- [2] A. Spillner: “ The W-Model srengthening the Bond Between Development and Test”, Orlando 2002
- [3] S. Schula, K.J. Buchenrieder, “Multilevel Testing for Design Verification of Embedded systems,” Embedded systems, IEEE Design & Test of Computers, pp 60-69, March 2002
- [4] M. Ricketts, “Testing and Configuration Management,” The 16th meeting of the European Working Group on Operational Workstations(EGOWS), Exeter, GB, June 2005