

# 분석단계의 테스트를 위한 체크리스트의 개발

양해술  
호서대학교 벤처전문대학원  
hsyang@office.hoseo.ac.kr

이하용<sup>U</sup>  
한국소프트웨어품질연구소  
insq@unitel.co.kr

## Development of Checklist for Test of Analysis Phase

Hae-Sool Yang  
Graduate School of Venture, Hoseo Univ.

Ha-Yong Lee<sup>U</sup>  
Institute of Software Quality

### 요 약

소프트웨어 분야의 급격한 발전으로 인해 소프트웨어 품질의 중요성이 더욱 높아지고 있다. 그러나 많은 소프트웨어 개발 프로젝트들이 정해진 일정 때문에 충분한 품질관리가 이루어지지 못하고 잠재적인 위험 요소들을 지닌 채 제품화되어 출시되고 있는 것이 현실이다. 이러한 위험 요소들은 소프트웨어의 결함으로 잠재하면서 언젠가 고장(failure)을 일으킬 수 있는 가능성을 지니고 있다. 소프트웨어 개발 과정에서 발생하는 오류는 초기 단계의 오류일수록 최종 소프트웨어 제품에 미치는 영향이 크므로 초기 단계에 문제점을 점검하는 것이 더욱 중요한 문제일 뿐 아니라 생명주기 전 단계에 걸쳐 각 단계에 적합한 시험 항목을 개발하여 적용하는 것이 고품질의 소프트웨어를 개발하는 최선의 방법이 될 수 있을 것이다. 본 연구에서는 생명주기 단계 중 분석 단계에 관한 테스트 항목을 개발하여 분석 단계에서 야기될 수 있는 문제점들을 최소화할 수 있는 방안을 마련하였다.

### 1. 서론

소프트웨어가 발전함에 따라 품질의 중요성은 더욱 높아지고 있다. 그러나 많은 소프트웨어 개발 프로젝트들이 충분한 시간을 할당하여 품질관리를 수행하거나 검토가 이루어지지 못한 채 잠재적인 결함 발생 위험 요소들을 지닌 상태에서 제품화되고 있다. 일반적으로 소프트웨어의 결함이나 오류를 최소화하여 고품질의 소프트웨어를 개발하기 위한 방안으로 적용하는 방법은 개발된 소프트웨어에 대해 결함이나 오류를 찾아내기 위해 시험 사례를 개발하여 소프트웨어를 직접 실행해 보면서 검토하는 것이 일반적이다.

그러나 소프트웨어 개발 과정에서 발생하는 오류는 초기 단계의 오류일수록 최종 소프트웨어 제품의 품질에 미치는 영향이 크므로 초기 단계에 문제점을 점검하는 것이 더욱 중요한 문제일뿐 아니라 생명주기 전 단계에 걸쳐 각 단계에 적합한 시험 항목을 개발하여 적용하는 것이 고품질의 소프트웨어를 개발하는 최선의 방법이 될 수 있을 것이다.

본 연구에서는 생명주기 전 단계 중 분석 단계에 관한 테스트 항목을 개발함으로써 소프트웨어 품질 향상을 위한 기반을 구축하였다.

### 2. 소프트웨어의 개발 공정

소프트웨어의 개발은 공정의 순환으로 이루어진다. 공정의 순환은 Plan-Do-Check-Action의 4가지 요소로 구성된다. 개발팀은 프로젝트 계획을 세우고 계획한 일정에 따라 소프트웨어를 개발하며, 테스트하는 사람은 소프트웨어가 수요자와 사용자의 요구사항에 부합하는지를 결정하기 위한 점검을 수행한다. 만약 수요자와 사용자의 필요에 부합되지 않는다면, 테스트하는 사람은 결점들을 개발팀에게 보고하며, 개발팀은 처리되지 않은 결점들을 바로잡는다.

테스트 하는 사람은 테스트를 통해 자신의 책임을 완수함으로써 소프트웨어 생명주기 전과정을 통해 품질 향상 업무에 참여하게 된다.

### 3. 소프트웨어 결함의 유형

소프트웨어의 결함이란 원하는 제품의 속성과 불일치를 보이는 것이다. 결함에는 제품 명세의 결함과 고객이나 사용자의 기대와 불일치하는 경우와 같이 두 가지 종류가 있다.

#### 3.1 제품 명세의 결함(defect)

제품 명세의 결함이란 만들어진 제품이 제품에 대한 명세와 다른 경우를 말한다. 즉, 제품을 만드는 알고리즘이 명세와

다른 경우를 말한다. 즉, 제품을 만드는 알고리즘이 명세와 다르다면 결함이 있는 것으로 생각할 수 있다.

**3.2 고객이나 사용자의 기대와 불일치**

이러한 불일치는 만들어진 제품에 사용자가 원하는 것이 없다가 만들어진 제품에 포함될 수 있도록 명세되지 않았다는 것이다. 이러한 잘못은 명세서를 작성하는 과정에서 또는 요구 과정에서 발생하거나 구현된 요구가 만족스럽지 못한 경우에 발생한다. 결함은 일반적으로 다음과 같은 세 가지 부류 중 하나의 형태로 나타난다.

유형	의미
잘못(wrong)	명세가 부정확하게 구현된 경우에 해당한다. 이 결함은 고객이나 사용자의 명세와 불일치하는 것
누락(missing)	명세된 사항이나 사용자의 요구가 만들어진 제품에 구현되어 있지 않은 경우
여분(extra)	명세되지 않았던 요구가 제품에 포함된 경우

**3.3 결함과 실패**

결함은 소프트웨어 시스템에 포함된다. 결함은 잘못(wrong), 누락(missing), 여분(extra)으로 분류된다. 결함은 소프트웨어 자체에서 발견되거나 매뉴얼과 문서화를 지원할 때 발견될 수 있다. 결함이 소프트웨어 시스템에 포함된 문제점이기 해도, 그것이 사용자, 고객, 운영 시스템에 영향을 미치지 전까지는 큰 피해를 주지는 않는다.

동작 중에 발생하는 오류나 사용자, 구매자에게 부정적인 영향을 미치는 결함을 실패(failure)라 한다. 결함이 갖는 주된 관심사는 결함이 실패로 바뀔 수 있다는 것이다.

조직에 손해를 끼치는 것은 실패(failure)이다. 어떤 결함은 전혀 실패로 나타나지 않을 수도 있다. 다시 말해서, 하나의 결함이 많은 실패를 야기할 수도 있다는 것이다.

**4. 분석단계의 체크리스트 개발**

본 연구에서는 분석 단계의 시험을 수행하기 위한 체크리스트를 개발하였다.

정보시스템을 개발하는 과정에서 테스트는 개발 요구가 시작된 시점부터 함께 해야 한다. 사용자 요구를 파악하고 분석하는 과정에서 오류가 발생한다면 전체 어플리케이션에 오류가 존재하게 될 것이다. 시스템에 관한 요구 사항을 테스트함으로써 사용자의 요구를 올바르게 수용한 정보 시스템이 될 가능성을 증가시키게 될 것이다.

이런 관점에서 분석단계의 테스트는 요구 조건이 적절하게 기록되는 것을 보장하고, 이런 요구 조건은 프로젝트 팀의 서비스 정보에 의해서 정확하게 해석되어야 한다. 요구 조건은 부서의 가이드 라인이나 기준, 절차에 따라 기록될 것이다.

**4.1 방법론 테스트 요소**

요구를 정의하고 문서화하기 위한 프로세스는 요구 단계 동안에 수행되어야만 한다. 이러한 절차가 제대로 규정되어 있을 수록 테스트 프로세스는 더욱 쉬워진다.

요구 프로세스는 사실 수집, 분석, 결정, 그리고 설계에서 사용하기 위해 사전에 정의된 방법으로 요구를 기록하는 과정이다. <표 2>는 분석단계의 방법론 테스트 요소에 관한 시험을 위한 체크리스트의 예를 나타낸 것이다.

<표 2> 방법론 테스트 요소에 관한 체크리스트

#	테스트기준	평가			제안된 테스트
		매우 적합	적합	부적합	
1	이용할 수 있는 조직의 정책과 절차는 명시되어 있는가?				정책과 절차를 개발할 책임이 있는 사람과 모든 적용 가능한 정책들이 확인되었는가를 확인하라.
2	요구사항들은 이런 정책과 절차에 따르는가?				정책과 절차에 따르는가를 보증하기 위해 요구를 재검토하라.
3	요구사항들은 요구 방법론에 따라 문서화되어 있는가?				모든 필요한 문서가 완전히 갖추어져 있는지 확인하기 위해 요구를 검토하라.
4	---	---	---	---	---

**4.2 정확성 테스트 요소**

사용자의 만족은 시스템 목표가 달성될 때 보증될 수 있다. 이러한 목표의 달성은 목표가 측정 가능할 때에만 측정될 수 있다. '사용자에 대한 서비스 개선'과 같은 정성적인 목표는 측정할 수 없는 목표이다. 반면에 '4시간 동안에 사용자의 요구를 처리한다'와 같은 정량적인 목표는 측정 가능하다.

<표 3>은 분석단계의 정확성 테스트 요소에 관한 시험을 위한 체크리스트의 예를 나타낸 것이다.

<표 3> 정확성 테스트 요소

#	테스트기준	평가			제안된 테스트
		매우 적합	적합	부적합	
1	어플리케이션이 요구하는 데이터가 요구된 수준의 신뢰성을 가지고 수집될 수 있는가?				데이터가 요구된 수준의 신뢰성을 가지고 생성될 수 있다는 것을 데이터를 생성할 사람과 함께 확인
2	명시된 기간 내에 데이터가 수집될 수 있는가?				데이터가 요구된 시간 프레임 내에 수집될 수 있도록 데이터를 발생시키는 사람을 확인
3	사용자의 요구가 기록으로 정의되었는가?				기록된 요구가 완전한가를 확인
4	---	---	---	---	---

**4.3 사용용이성 테스트 요소**

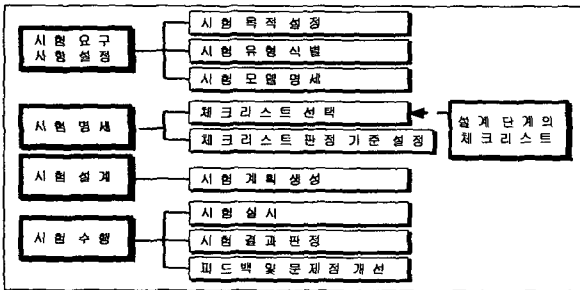
시스템을 사용하기 위해 필요한 노력의 양과 기술 수준은 요구 단계 동안에 정의되어야 한다. 경험에 의하면, 사용하기 어려운 어플리케이션이나 특징들은 자주 사용되지 않는다는 것을 알 수 있다. 반면에 사용하기 쉬운 기능적인 시스템들은 빈번하게 사용된다. 만일 명세에 포함되지 않았다면, 사용용이성 명세는 시스템 분석가나 프로그래머가 디폴트로 만들 수 있다. <표 4>는 사용용이성 테스트 요소에 관한 시험을 위한 체크리스트의 예를 나타낸 것이다.

<표 4> 사용용이성 테스트 요소에 관한 체크리스트

#	테스트기준	평가				제안된 테스트
		매우 적합	적합	부 적합	N/A	
1	사용자 함수들이 확인되어 있는가?					사용자 함수들이 요구사항에 정의되어 있는가를 사용자와 함께 확인
2	사용자의 기술 레벨이 식별되어 있는가?					사용자 기술 레벨을 기술하고 있는 요구사항 문서를 시험
3	기대된 수준의 감독이 식별되어 있는가?					기대된 수준의 감독을 기술하고 있는 요구사항 문서를 시험
4	---	--	--	--	--	---

5. 체크리스트의 적용을 위한 절차

분석 단계의 시험을 위한 체크리스트는 고품질의 개발 공정 구축 및 문서 개발을 위해 개발 기관 스스로가 도입하여 적용하는 것이 바람직하다. 따라서 개발 기관은 시험을 위한 효과적인 절차를 구축하고 구축된 절차에 따라 분석 단계 시험을 위한 체크리스트를 적용할 수 있도록 함으로써 고품질 소프트웨어 개발의 목표에 접근할 수 있을 것이다. 이러한 절차를 구축하기 위해 도입하여 적용할 수 있는 표준 절차로서 ISO/IEC 14598-3의 개발자를 위한 품질평가 프로세스를 들 수 있다. ISO/IEC 14598-3은 ISO/IEC 14598 전반에 적용되는 기본 평가 프로세스를 바탕으로 품질 요구를 명백히 하고 소프트웨어 품질 측정의 구현과 분석에 대한 가이드라인을 제공함으로써 개발자에게 개발 전과정 동안 소프트웨어 품질 측정과 평가의 적용을 지원한다. (그림 1)은 ISO/IEC 14598의 기본 평가 프로세스를 바탕으로 설계 단계 시험에 적용할 수 있는 절차를 나타낸 것이다.



(그림 1) 분석 단계 시험을 위한 프로세스

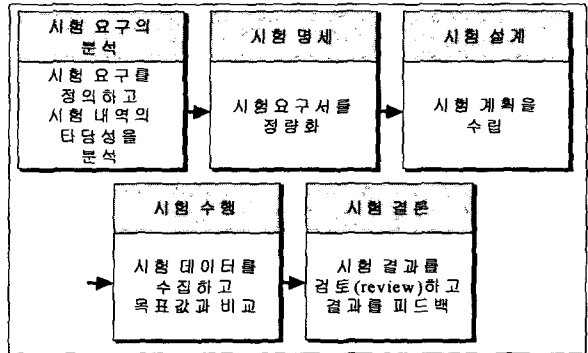
(그림 2)에 ISO/IEC 14598-3 개발자를 위한 프로세스를 기초로 설계 단계 시험을 위한 절차를 나타내었다.

6. 결론

소프트웨어 개발 과정에서 오류가 발생하는 것은 피할 수 없는 문제이다. 그러나 오류를 최소화함으로써 오류로 인한 손실 발생 가능성을 최소화할 수 있다.

소프트웨어 개발 과정에서 초기 단계의 오류일수록 최종 소

프트웨어 제품에 미치는 영향이 크므로 초기 단계의 시험이 매우 중요하다.



(그림 2) 개발자를 위한 프로세스

본 연구에서는 고품질의 소프트웨어 제품을 개발하기 위한 방안으로 생명주기 단계 중 초기 단계인 분석단계에서 적용할 수 있는 테스트 항목을 개발하였다.

소프트웨어 제품의 오류는 초기 단계에서 발생할수록 큰 영향이 미친다는 점을 고려할 때 분석단계 및 설계단계 등의 효과적인 테스트가 매우 중요한 의미를 갖는다.

따라서 본 연구를 통해 개발된 분석단계의 시험을 위한 체크리스트를 활용하여 분석단계에 대한 효과적인 시험 공정이 이루어질 수 있을 것이다. 향후, 테스트 항목을 추가하고 구체적인 테스트 방법에 대한 보완이 이루어질 수 있도록 지속적인 연구가 진행되어야 할 것이다.

참고 문헌

- [1] Moller, K. H. and Paulish, D. J., "Software Metrics", Chapman & Hall(IEEE Press), 1993.
- [2] Wallmuller, E., "Software Quality Assurance A practical approach", Prentice Hall, 1994.
- [3] ISO/IEC 12119, "Information Technology - Software Package - Quality requirement and testing".
- [4] 吉澤. 東. 片山, "소프트웨어의 품질관리와生産技術", 日本規格協會, 1990. 5.
- [5] 水野幸男, "소프트웨어의綜合的品質管理", 日科技連出版, 1993.
- [6] 양해술, 이하용, "설계단계에서의 품질평가 툴킷(ESCORT-D)의 설계 및 구현", 한국정보과학회논문지(C), Vol. 3, No. 3, 1997. 6.
- [7] 양해술, "한진해운 신정보(영업 및 물류)시스템의 품질보증과 품질평가", 한진해운(주) 구현단계 확인평가, 1998. 9. 7.
- [8] 양해술, "소프트웨어 제품 평가 지원도구의 개발", ETRI 컴퓨터·소프트웨어 기술연구소 용역과제, 3차년도최종보고서, 1999. 10.