

# 웹기반 소프트웨어 품질평가 모델 개발

오영배, 정영은, 신석규

수원여대 컴퓨터응용학부, TTA SW시험인증센터

yboh@swc.ac.kr, {yejung, skshin}@tta.or.kr

## Development of a Quality Evaluation Model for Web-based Software

### 요 약

소프트웨어 어플리케이션들이 점차 다양한 응용분야에서 광범위하게 사용됨에 따라 소프트웨어 품질의 중요성이 높아지고 있다. 특히 웹기반 소프트웨어는 전통적 소프트웨어에 비해 보다 높은 품질 수준을 요구하고 보안이 중요시되며 시스템 통합과 유지보수 및 재사용에 대한 품질 요소가 중요시된다. 웹기반 소프트웨어의 품질을 객관적이고 정량적으로 명세하고 평가하기 위하여, 객관적인 근거를 제시할 수 있는 지표로서 평가 항목들이 정의되어야 한다. 본 논문은 웹기반 소프트웨어의 품질 평가를 위한 모델을 제안하고 평가항목 및 평가방법과 절차를 제시하고자한다.

## 1. 서 론

오늘날 소프트웨어는 점차 다양한 응용분야에서 광범위하게 사용되고 있으며, 그 정확한 작동은 기업을 성공으로 이끌고 인간의 안전과도 깊은 관련을 갖는다. 따라서 고품질의 소프트웨어 제품을 개발하거나 선정 획득하는 일은 매우 중요하다.

웹기반 소프트웨어의 품질은 웹을 기반으로 동작하는 소프트웨어 제품이 사용자의 요구사항이나 만족도를 얼마나 충족시키는가를 나타낸다[1].

웹기반 소프트웨어의 특징은 익명의 사용자가 서버에 접속하여 작업을 수행한다는 것이다. 웹기반 소프트웨어에서는 특정한 사용자를 가정할 수 없으며 사용자의 편리성과 소프트웨어의 성능, 효율성 등이 중요한 품질 요소가 되고 또한 전자상거래를 중심으로 시스템을 통합하는 추세이므로 시스템의 통합성, 상호운용성 및 보안성이 중요시되며, 클라이언트와 서버 상에 다양한 기술이 혼재되어 있으므로 소프트웨어 관리가 어려워지는 특성이 있어 유지보수성 및 유연성이 필요한 속성으로 고려되고 있다.

본 논문은 웹기반 소프트웨어의 품질을 평가하기 위한 지침으로서 평가모델 및 방법을 제시한다.

## 2. 관련 연구

ISO/IEC 9126[2] 소프트웨어 제품 품질 모델이 제시된 이래 지금까지의 소프트웨어 품질 평가 모델에 대한 연구는 소프트웨어의 일반적 품질특성을 정의하는데 초점을 두고 있다. ISO/IEC 9126 모델이 다양한 소프트웨어 분야에 적용할 수 있는 포괄적인 특성을 담고 있으나 세분화 되고 있고 새로운 특성들이 생성되는 영역별 소프트웨어에 모델을 적용하는 것이 점점 어려워지고 있다. 특히 최근의 인터넷 기반 소프트웨어의 급격한 증가에 따라 웹 기반 소프트웨어의 특성을 반영하여 적용할 수 있는 평가모델의 필요성이 요청되고 있다. 이러한 필요성에 따라 기존에 연구되었던 웹기반 평가모델을 제시하고 그 특성을 비교평가하면 다음과 같다.

### 2.1 Omnibus

Omnibus 모델[3]은 소프트웨어 제품의 평가모델로서 제품운용, 제품변경 및 제품설치/이전 등의 세가지 관점에서 품질속성을 정의하고 있다. 본 모델은 웹기반 소프트웨어뿐만 아니라 일반적인 소프트웨어 품질을 포괄할 수 있도록 구성되어있고 품질속성이 상세히 정의되어있다. 이 모델은

ISO/IEC 9126 품질모델의 모태가 되었다. 본 모델은 웹기반 소프트웨어의 구체적인 특성을 반영하고 있지 못하여 웹기반 소프트웨어에 적용하기에는 어려움이 따른다.

## 2.2 WebQEM

WebQEM(Web Quality Evaluation Method) 모델[4]은 웹개발 프로젝트 또는 웹어플리케이션 운영 단계에서 웹사이트 및 웹어플리케이션이 품질 요구사항을 만족하는지를 평가하기 위한 모델이다. 본 모델은 네비게이션, 접근성, 검색 메커니즘, 콘텐츠, 신뢰성, 성능 등 인터페이스 관련 설계 및 구현 특성에 대한 문제점을 발견하는데에도 유용하다. 이 모델은 웹사이트와 웹어플리케이션의 전반적인 특성을 평가할 수 있지만 어플리케이션의 네비게이션, 인터페이스 및 신뢰성 등 사용성 위주의 제품 특성을 평가하는 초점을 두고 있으며, 코드의 설계 및 품질 관련 속성을 평가하기에는 다소 미흡하다. 즉 일반적인 방문자 관점에서의 웹사이트의 특성과 속성을 고려하고 평가하는데 주안점을 두고 있다.

## 2.3 Web Testing Handbook

Web Testing Handbook[5]은 웹기반 어플리케이션의 시험유형과 시험절차를 제시하고 있다. 시험유형으로는 Cross-Platform 시험, 사용자 인터페이스 시험, 성능 시험, 품질 시험 및 데이터베이스 시험으로 구분하고 있으며 시험절차로서는 시험계획, 시험사례 설계, 시험 개발, 시험 수행 및 평가, 다음 단계에 대한 준비 등으로 구분하고 있다. 각 유형 및 단계별 시험 방법을 상세히 설명하고 있으나 웹기반 시스템의 품질평가 모델 및 품질평가를 포함하는 메트릭을 제시하지는 못하고 있다.

# 3. 평가 모델

## 3.1 고려 사항

웹기반 소프트웨어가 패키지 소프트웨어와 다른 차이점은 패키지 소프트웨어가 제품으로 제공되는데 반해 웹기반 소프트웨어는 서비스로 존재한다는 것이다. 따라서 웹기반 소프트웨어를 평가하기 위하여 고려하여야 하는 요소는 다음과 같다. 첫째, 웹 환경에서는 사용자를 한정할 수 없고 익명의 사용자들이 서버에 접속하여 작업을 수행하는 것이다.

이러한 환경에서는 사용자에게 소프트웨어의 사용자 매뉴얼을 제공하기가 어렵고 사용방법에 대한 교육도 하기가 어려워, 소프트웨어를 쉽게 사용할 수 있도록 사용 용이성을 높이는 것이 필요하다. 둘째, 웹 환경에서는 사용자의 접속수를 예측하기 어려우며 특정 시점에 사용자가 몰려 부하가 집중될 수 있어 피크타임을 대비한 소프트웨어의 성능 및 효율성을 고려하여야 한다는 것이다. 셋째, 시스템이 외부에 공개되어 있는 특성상 해커 및 부정사용자들에게 공격의 가능성에 노출되어 있기 때문에 보안성 확보가 필수적이다. 넷째, 웹기반 소프트웨어는 웹 상에서 다양한 타시스템과의 접속, 연동 및 통합이 필요하므로 시스템의 상호운용성과 통합성이 필요하다. 다섯째, 웹기술의 급격한 발전 속도에 따라서 시스템의 신기술 수용 능력과 소프트웨어 업그레이드, 소프트웨어 관리 및 유지보수가 용이하게 이루어져야 한다.

## 3.2 모델 개발

본 평가모델은 웹기반 소프트웨어의 품질을 평가하기 위한 지침으로서 기준 및 방법을 제시하기 위하여 ISO/IEC 9126 모델과의 일관성을 유지하면서 웹기반 소프트웨어의 특성을 고려한 품질 특성을 도출하여 개발하였다.

본 모델은 웹기반 소프트웨어에 적용하기 위한 것이다. 예를 들면 전자상거래, 원격강의, 에이전트 기반 시스템 등 웹을 통해 지원되는 소프트웨어들에 적용될 수 있다.

본 모델은 웹기반 소프트웨어 품질을 여섯가지 특성(characteristics)으로 구분하고, 이를 다시 부특성들(sub-characteristics)로 세분화한 평가 모델을 명세한다. 이러한 부특성들은 소프트웨어가 컴퓨터 시스템의 일부로 사용될 때 외부로 나타나며, 내부적인 소프트웨어 속성들의 결과이다.

정의된 특성은 컴퓨터 소프트웨어와 데이터를 포함하여 모든 종류의 소프트웨어에 적용할 수 있다. 특성과 부특성은 소프트웨어 품질에 대한 일관성 있는 용어를 제공하며, 소프트웨어에 대한 품질 요구사항을 명세하고, 또한 사용성, 효율성, 안정성, 상호운용성, 보안성 및 유지보수성과 같은 웹기반 소프트웨어의 평가를 위한 지침을 제공한다.

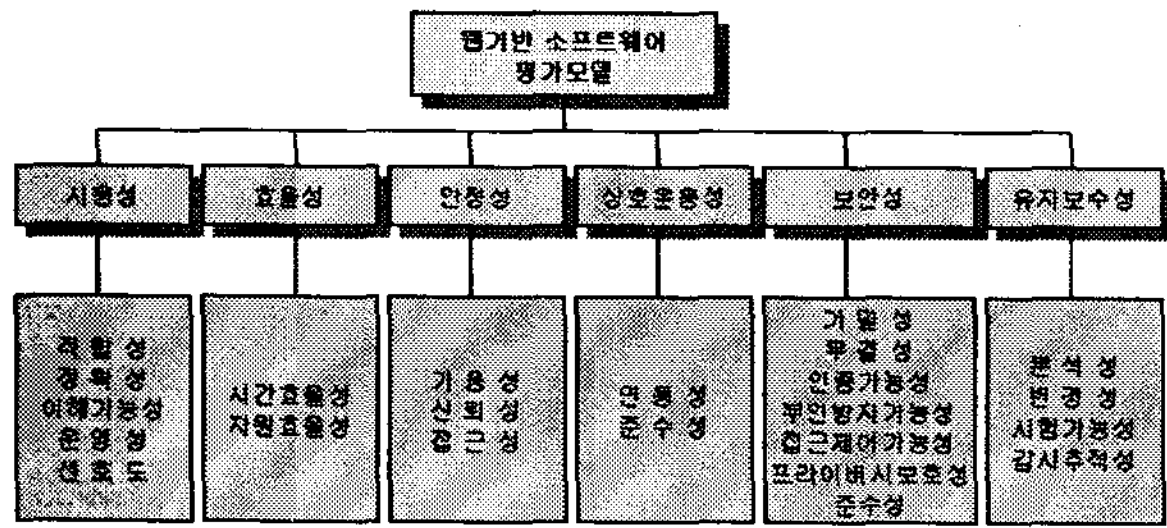
소프트웨어 제품의 품질을 평가하기 위한 일반적인 평가모델은 TTASK.O11.0049 “패키지 소프트웨어 품질평가 항목”[6] 표준에 제시되어 있다. 이것은 소프트웨어를 제품으로 보았을 때 고려할

수 있는 소프트웨어의 품질평가 기준을 제공한다.

웹기반 소프트웨어는 설치위주의 소프트웨어 제품과는 다른 서비스 지향적이고 공개적인 소프트웨어로서의 특성을 가지고 있으며 본 모델에서는 이러한 특성을 반영한 품질특성을 제시하고 있다. 본 평가모델은 패키지 소프트웨어 품질평가 모델과 일관성을 가지고 있으며 이를 웹기반 소프트웨어의 특성에 적합하게 확장 보완한 모델이다.

본 모델의 사용자는 요구사항에 따라 본 평가항목에 새로운 항목을 추가하거나 기존 항목을 변경하여 적용할 수 있다.

웹기반 소프트웨어 평가 모델은 웹기반 소프트웨어 품질을 측정·평가하기 위하여 소프트웨어의 품질요소와 특성을 정의하고 있다. 본 모델은 [그림 1]과 같이 여섯 개의 품질특성으로 구분하며, 이러한 품질특성별로 세부 평가항목에 대한 평가 지침을 제시한다.



[그림 1] 웹기반 소프트웨어 품질평가 모델

## 4. 평가항목

### 4.1 평가항목의 요소

[그림 1]의 평가모델의 품질특성 및 부특성은 품질측정을 위한 평가항목으로 구분되며 평가항목을 구성하는 각 요소는 다음과 같다.

- 가) 특성: 품질모델 상에서 평가항목이 위치하는 대분류
- 나) 부특성: 품질모델 상에서 평가항목이 위치하는 중분류
- 다) 평가항목명: 해당 평가항목의 이름
- 라) 평가 목적: 평가항목에 대하여 측정하기 위한 목적을 표현
- 마) 평가방법: 해당 평가항목에 대하여 측정하기 위한 방법을 제시
- 바) 사용자: 해당 평가항목을 사용하는 사용자 부류를 제시

## 4.2 평가항목의 구성

품질 측정을 위한 평가항목은 품질모델의 6개의 품질특성 및 각 특성별 부특성에 따른 측정항목을 도출하여 항목별 특성 및 평가방법을 제시한다. 본 모델의 전체 평가항목의 구성은 [그림 2]와 같다.

<b>1 사용성</b> 1.1 적합성 1.1.1 기능 구현 완전성 1.1.2 기능 충분성 1.1.3 기능 적절성 1.2 정확성 1.2.1 기능 구현 정확성 1.2.2 정밀성 1.3 이해 가능성 1.3.1 기능 이해도 1.3.2 인터페이스 이해도 1.3.3 도움말 이해도 1.3.4 입출력 데이터이해도 1.3.5 인터페이스 일관성 1.3.6 사용자 안내성 1.3.7 메시지 이해 용이성 1.4 운영성 1.4.1 운영 절차 조정 가능성 1.4.2 운영 절차 일관성 1.4.3 진행상태 파악 가능성 1.4.4 오류 복구 용이성 1.4.5 문제 해결 정보 제공	<b>4 상호운용성</b> 4.1 연동성 4.1.1 데이터 교환성 4.2 준수성 4.2.1 표준 준수율
<b>2 효율성</b> 2.1 시간 효율성 2.1.1 평균 반응 시간 2.1.2 평균 처리율 2.1.3 평균 처리 시간 2.2 자원 효율성 2.2.1 입출력 자원 사용률 2.2.2 메모리 사용률 2.2.3 데이터 전송률 2.2.4 CPU 사용률	<b>5 보안성</b> 5.1 기밀성 5.1.1 데이터 기밀성 5.2 무결성 5.2.1 데이터 무결성 5.3 인용 가능성 5.3.1 사용자 인용 가능성 5.4 무인방지기능성 5.4.1 통신사실 무인방지기능성 5.5 접근제어 가능성 5.5.1 접근정책 가능성 5.5.2 접근정책 보호성 5.6 프라이버시 보호성 5.6.1 프라이버시보호 가능성 5.7 준수성 5.7.1 표준 준수율
<b>3 안정성</b> 3.1 가용성 3.1.1 이용 가능성 3.2 신뢰성 3.2.1 결함 회피율 3.2.2 고장 회피율 3.2.3 오류 방지성 3.3 접근성 3.3.1 접근 가능성	<b>6 유지보수성</b> 6.1 분석성 6.1.1 진단 가능 지원률 6.1.2 상태 모니터링 제공률 6.1.3 감사 추적 가능성 6.2 변경성 6.2.1 변경 가능성 6.2.2 소프트웨어 변경통제 가능성 6.2.3 변경 용이성 6.3 시험 가능성 6.3.1 내장형 시험 기능 보유성 6.4 감사 추적성 6.4.1 감사 추적 가능성

[그림 2] 웹기반 소프트웨어 평가항목의 구성

각각의 평가항목은 4.1절의 평가요소에 따른 평가 및 측정 방법을 제시한다. [그림 3]은 사용성 품질특성에 대한 평가항목을 보여주고 있다.

번호	특성	부특성	평가항목명	평가항목의 목적	평가방법	사용자
1	사용성	적합성	기능구현 완전성	문서에 기술되어 있는 기능의 구현 여부	소프트웨어에 구현된 기능 수와 사용자 문서에 기술된 기능의 수를 비교하여 평가	개발자, 품질보증자
2	사용성	적합성	기능 충분성	소프트웨어에 필요한 필수적인 기능이 충분히 구현되어 있는지 여부	소프트웨어에 요구되는 필수기능과 실제 소프트웨어에 구현된 기능의 수를 비교하여 평가	개발자, 품질보증자
3	사용성	적합성	기능 적절성	평가된 기능이 적절하게 구현되어 있는지 여부	평가된 기능과 평가시 문제가 발견된 기능의 수를 비교하여 평가	개발자, 품질보증자
4	사용성	정확성	기능 구현 정확성	소프트웨어 사용자 문서에 기술되어 있는 기능이 정확하게 동작하도록 구현되어 있는지 여부	소프트웨어에 정확히 구현된 기능과 평가할 모든 기능의 수를 비교하여 평가	개발자, 사용자
5	사용성	정밀성	정밀성	소프트웨어의 결과값이 사용자 문서에 기술되어 있는 결과값의 정밀도와 동일하게 구현되어 있는지 여부	요구되는 정밀도 수준과 동일하지 않은 결과의 경우를 평가	개발자, 사용자
6	사용성	이해 가능성	기능 이해도	제품설명서와 사용자문서를 읽고 소프트웨어가 제공하는 기능을 이해할 수 있는 정도	문서를 읽고 기능 이해 수준을 판단	개발자, 사용자, 시험자
7	사용성	이해 가능성	인터페이스 이해도	소프트웨어의 메뉴 및 기타 인터페이스를 보고 기능을 이해할 수 있는 정도	메뉴 및 인터페이스를 보고 기능의 이해 수준을 판단	개발자, 사용자, 시험자
8	사용성	이해 가능성	도움말 이해도	제품에서 제공하는 도움말(데모/튜토리얼)을 쉽게 이해할 수 있는 정도	도움말(또는 데모/튜토리얼)을 읽고 이해 수준을 판단	개발자, 사용자, 시험자
9	사용성	이해 가능성	입출력 데이터 이해도	소프트웨어의 입력 및 출력에 사용되는 데이터를 쉽게 이해할 수 있는 정도	입/출력에 해당하는 화면 및 문서를 보고 내용의 이해수준을 판단	개발자, 사용자, 시험자
10	사용성	이해 가능성	인터페이스 일관성	인터페이스 요소들 간에 일관성있게 구현된 정도	인터페이스를 보거나 작동하여 불일치 않는 정도를 판단	개발자, 사용자, 시험자

[그림 3] 사용성 평가항목

### 5. 평가과정

#### 5.1 평가방법

[그림 3]의 평가항목은 소프트웨어 측정·평가를 위한 지침으로 사용할 수 있다. 이 평가항목은 [그림 4]와 같은 측정가능한 평가 메트릭(metrics)으로 변환하여 적용할 수 있으며 각각의 평가항목은 평가치를 구하기 위한 테스트케이스를 작성하여 시험하여야 한다.

평가항목	부속성	평가항목	측정단위	측정대상	점검표	서명여부		
적합성		기능 정보 제공	scale	D/P	기능 점검표	Yes		
		기능 구현 완전성	scale	D/P	기능 점검표	Yes		
		경계값 정보 제공	scale	D/P	경계값 점검표	Yes		
		경계값 처리율	scale	D/P	경계값 점검표	Yes		
정확성		기능 구현 정확성 정보 제공	scale	D/P	기능 점검표	Yes		
		기능 구현 정확성	scale	D/P	기능 점검표	Yes		
이해가능성		메뉴 지식 정보 제공	Y/N/NA	D		No		
		기능 이해도	scale	D	기능 점검표	Yes		
		인터페이스 이해도	scale	P	기능 점검표	Yes		
		도움말 이해도	scale	D/P	도움말 이해 점검표	Yes		
		입/출력 데이터 이해도	scale	D/P	입/출력 데이터 점검표	Yes		
		인터페이스 일관성	scale	D/P	인터페이스 일관성 점검표	Yes		
		내용 일관성	scale	D/P	내용 일관성 점검표	Yes		
		사용자 안내성	Y/N/NA	D/P		No		
		운영성		오류 복구 용이성	scale	P	오류 복구 점검표	No
				오류 방지성	scale	D/P	오류 방지 점검표	Yes
메시지 이해 용이성	scale			D/P	메시지 이해 용이성 점검표	Yes		
운영원차 조정 가능성	Y/N/NA			D/P		No		
선호도		진행상태 파악 가능성	scale	D/P	진행상태 파악 점검표	Yes		
		인터페이스 조정 가능성	scale	D/P	인터페이스 조정 가능성 점검표	No		
		인터페이스 선호도	scale	P	인터페이스 선호도 점검표	No		

[그림 4] 사용성 품질 메트릭

평가항목별 메트릭을 구성하여 각각의 테스트케이스(점검표)를 작성한 후 테스트한 결과는 [그림 5]와 같은 점검표로 결과를 기술한다.

순번	경계값	종류 (D/P)	상세설명	결과 (Y/N)	비고
1	30	D	Remote Manager 사용자 등록시 30까지 가능	Y	경계값 처리는 있으나 '메인 메시지 사용자 등록이 거부됨' 이라고 나옴
2	16 or 8	D	Remote Manager 사용자 등록시 아이디는 20자 이하는 16자까지 가능함	N	Linux에서 숫자로 아이디와 패스워드 등록이 안됨
3	0<x<100	P	로그파일의 크기가 제한됨	Y	
4	500	P	Telnet에서 윈도우 사이즈가 아무의 높은 값을 넣어도 500으로 제한됨	Y	
5	65535	P	Remote Manager에서 서버정보의 포트에 대한 한계값	Y	
6	65535	P	Telnet에서 서버정보의 포트에 대한 한계값	Y	
7	65535	P	Ftp에서 서버정보의 포트에 대한 한계값	Y	
8	10000	P	Telnet에서 Set up screen에서 Line수가 한계값	Y	
9		P	Ftp에서 서버정보의 시간초과에 대한 경계값 처리	Y	시간초과에 대한 경계값 처리가 안됨
Y의 개수				8	
N의 개수				1	
결과				0.889	

[그림 5] 경계값 점검표

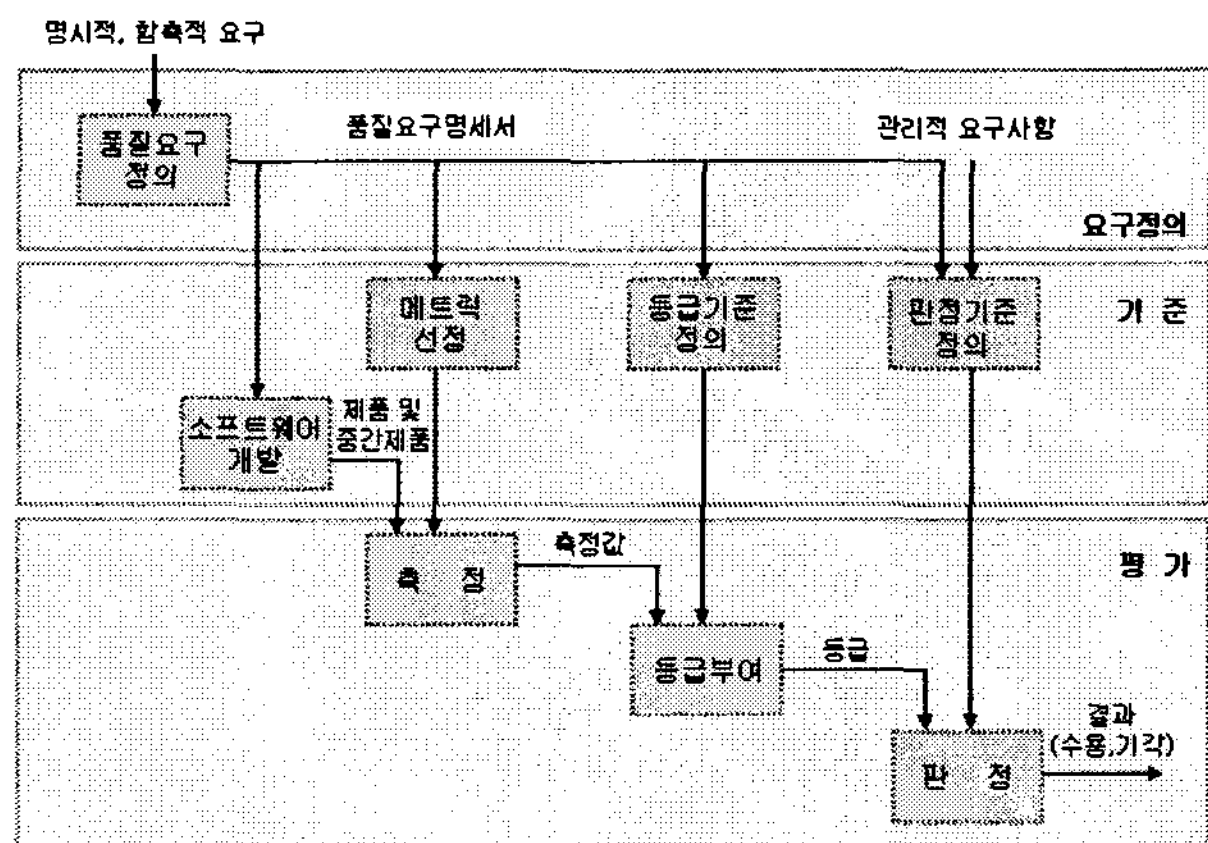
각 평가항목에 대한 테스트 결과를 집계하여 평가결과를 작성하면 [그림 6]과 같은 평가결과가 도출된다.

프로그램 및 데이터				
품질특성	부속성	평가항목	기준값	측정값
사용성	적합성	기능정보 제공	Scale	0.97
		기능 구현 완전성	Scale	0.98
		경계값 정보 제공	Scale	0.22
		경계값 처리율	Scale	0.89
	정확성	기능 구현 정확성 정보 제공	Scale	0.97
		기능 구현 정확성	Scale	0.98
	이해가능성	인터페이스 이해도	Scale	1.00
		도움말 이해도	Scale	0.95
		입/출력 데이터 이해도	Scale	0.90
		인터페이스 일관성	Scale	0.77
내용 일관성		Scale	0.96	
사용자 안내성		Y/N	N/A	
운영성	오류 방지성	Scale	0.81	
	메시지 이해 용이성	Scale	0.73	
	운영원차 조정 가능성	Y/N	NA	
	진행상태 파악 가능성	Scale	0.73	
선호도		인터페이스 조정 가능성	Scale	NA

[그림 6] 평가결과

#### 5.2 평가절차

[그림 7]은 소프트웨어 품질평가에 대한 평가절차를 보여주고 있다. 소프트웨어 요구정의 단계에서는 개발할 소프트웨어의 품질요구사항을 정의한다. 품질요구사항이 정의되면 기준설정 단계에서 소프트웨어의 품질을 평가하기 위한 품질모델에 따른 메트릭을 선정하며, 메트릭에 따른 평가결과에 따라 품질등급 기준을 정의하고, 도출된 등급의 품질수준을 수용할 것인지에 따른 판정기준을 정하게 된다. 기준설정 단계에서 기준을 설정하고 소프트웨어가 개발되면 측정단계로 넘어가게 된다. 측정단계에서는 메트릭에 따른 소프트웨어의 품질을 측정하고, 도출된 측정결과에 따라서 등급기준에서 정의된 품질등급을 부여하고, 판정기준에 따라 개발된 소프트웨어의 품질을 수용할 것인지 또는 기각할 것인지를 판정하게 된다.





## 6. 결론

본 논문은 웹기반 소프트웨어의 품질을 평가하기 위한 지침으로서 기준 및 방법을 제시하기 위하여 ISO/IEC 9126 패키지 소프트웨어 품질 모델과의 일관성을 유지하면서 웹기반 소프트웨어의 특성을 고려한 품질 특성을 도출하여 개발하였다. 웹기반 소프트웨어의 품질 평가를 위한 평가모형을 제안하고 평가항목 및 평가방법과 평가 절차를 제시하였다.

본 연구는 급격히 증가하고 있는 웹기반 소프트웨어 시장에서 웹에 기반한 소프트웨어 평가모형이 부재한 상황에서 ISO/IEC 9126 모델과의 일관성을 유지하면서 웹의 특성을 반영한 평가모형을 제시하였다는데 의의가 있다.

본 연구 결과는 국내 개발업체에서 생산하는 웹기반 소프트웨어의 품질을 객관적으로 평가할 수 있는 평가항목 및 평가방법을 제시하여 소프트웨어의 품질을 향상시킬 수 있는 기반을 마련함으로써 웹기반 소프트웨어 시장을 활성화하는데 기여하였다고 본다.

## 7. 참고문헌

- [1] Hung Quoc Nguyen, Testing Applications on the Web, John Wiley & Sons Inc, 2001
- [2] ISO/IEC, ISO/IEC 9126-1 Product quality-Part 1 : Quality model, 2001
- [3] G.Gordon Schulmeyer, James I McManus, Handbook of Software Quality Assurance, Prentice Hall, 1999
- [4] Luis Olsina, Gustavo Rossi, Measuring Web Application Quality with WebQEM, IEEE Software, Oct. 2002
- [5] Steven Splaine, Stefan P. Jaskiel, The Web Testing Handbook, STQE Publishing, 2001
- [6] TTA, TTAS.KO-11.0049, 패키지 소프트웨어 품질평가 항목, 2005