

소프트웨어 신뢰성 품질 평가 방법론에 대한 연구

정혜정*

*평택대학교 정보통계학과

e-mail : jhjung@ptu.ac.kr

A Study on the Method of Software Reliability Quality Testing

Hye-Jung Jung*,

*Dept of Information Statistics, Pyeong-Taek University

요 약

ISO/IEC 9126에 의하면 소프트웨어 품질 특성은 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성의 6가지 품질 특성으로 나누어 평가를 실시하고 있다. 각 특성은 부특성으로 나누어지고 이러한 부특성에 따른 평가 항목을 메트릭으로 정의하고 있다. 그러나 품질 특성 중 신뢰성의 경우는 ISO/IEC 9126에서 제시한 부특성인 성숙성, 오류허용성, 복구성, 준수성에 따라 품질 평가 항목이 메트릭으로 제시되어져 있다. 그러나 제시되어져 있는 신뢰성 평가 항목은 소프트웨어 신뢰성 성장 모델을 이용하여 신뢰도를 평가하고 메트릭 값을 얻도록 구성되어 있으나 수리적인 어려움과 데이터 수집에 따르는 문제점으로 인하여 적용하고 있지 못한 실정이다. 이러한 문제점을 고려하여 고장 데이터를 이용하여 소프트웨어 신뢰성을 평가할 수 있는 방안을 검토하고 현재 연구가 진행되어지고 있는 국제 표준 문건 ISO/IEC 25000 시리즈를 국내 표준으로 적용할 수 있는 방안에 대해서 연구한다.

1. 서론

소프트웨어 산업이 확대되어지고 대부분의 생활 속에 접하는 우리 삶 속에서 소프트웨어에 대한 영향이 커지고 있다. 특히 내장형 소프트웨어의 증가로 인하여 소프트웨어의 안전성에 대한 관심이 증대되어지고 있다. 이러한 시점에 특히 안전성을 요하는 산업용 소프트웨어, 철도청 소프트웨어, 항공기 소프트웨어 등에 대한 신뢰성 평가 방안에 대한 연구가 진행되어지고 있다. 소프트웨어 품질에 대해서 학문적 접근으로 품질을 향상시킨 일본의 경우도 소프트웨어 신뢰성 평가에 많은 연구를 진행하고 있다. 현재 국내의 시험기관에서 적용하고 있는 국제 표준 ISO/IEC 9126[3]의 경우 소프트웨어 품질 특성을 6가지 관점에서 평가하게 구성되어져 있다. 기능성(Functionability), 신뢰성(Reliability), 사용성(Usability), 유지보수성(Maintainability), 이식성

(Portability), 효율성(Efficiency)의 6가지 품질 특성에 대한 평가를 실시하고 있으나 신뢰성의 경우 소프트웨어 신뢰성 성장 모델(Software Reliability Growth Model)에 적용하여 고장데이터(Failure Data)를 이용하여 평가하여야 하는 어려움이 있다. 국내의 소프트웨어 업체들은 대체적으로 소프트웨어 개발 인력을 보유하고 있는 상태에서 소프트웨어 품질향상이란 측면에서 소프트웨어 제품의 품질에 대한 인식은 상당히 낮은 편이다. 현재 소프트웨어는 국내 시장에서 국외 시장으로 마케팅 전략을 변환하면서 소프트웨어 품질에 대한 인식변화가 일기 시작하였다. 특히 5년 전부터 소프트웨어 품질향상의 중요점을 인식하고 소프트웨어 시험 평가를 담당하고 있는 한국정보통신기술협회(TTA)의 노력과 산업용 소프트웨어 품질 평가를 담당하고 있는 산업자원부 기술표준원(ATS)의 역할로 인하여 국내 소프트웨어 업체의 인식변화가 일어나고 있다. 이러한 소프트웨어 품질에 대한 중요성의 인식은 앞으로 더

속 확대되어질 전망이다. 2006년에 각 지자체에서 내세운 도시계획 부분에서 각 지자체별 2010년을 겨냥한 전 도시의 유비쿼터스를 강조하고 있다. 이러한 시스템은 대체적으로 내장형 소프트웨어에 의해서 만들어진 제품이므로 소프트웨어 품질은 상당히 중요한 역할을 하게 되어질 것이다. 특히 철도청이나 항공기의 소프트웨어에 의한 고장은 큰 재앙을 동반할 수 있으므로 소프트웨어 신뢰성은 상당히 중요한 연구과제가 되어질 것이다. 본 연구에서는 2장에서 소프트웨어 신뢰성 평가 모델에 대해서 소개하고 ISO/IEC 9126[3]에서 소프트웨어 신뢰성에 대한 메트릭을 소개하고 3장에서는 ISO/IEC 25000 시리즈에 대해서 연구 진행 상황을 간단히 소개하고 ISO/IEC 25000[6,7,8,9,10,11,12]에서 소개한 품질 측정 프리미티브(Measurement Primitive)를 소개한다. 4장에서는 소프트웨어 신뢰도 성장 모델에 적용하여 신뢰도 평가를 한 사례를 소개한다. 5장에서는 본 연구의 결론 및 향후 연구과제에 대해서 검토한다.

2. ISO/IEC 9126 소개

소프트웨어 품질과 관련된 국제 표준화 활동은 ISO/IEC JTC1/SC7/WG6에서 주도적으로 연구활동하고 있으며 1998년 이후로 ISO/IEC 9126[3]을 기반으로 하는 국내 표준이 제정되었다. 소프트웨어 품질과 관련된 국제 표준으로는 산업자원부 기술표준원이나 한국 표준 협회 등에서 제정되어진 내용과 한국정보통신기술협회에서 제정한 내용들을 들 수 있다.

<표 1>소프트웨어 품질관련 TTA 표준

표준번호	제목
TTAS.IS-9126.1	소프트웨어 품질 특성 및 메트릭-품질 특성 및 부특성(1998. 10)
TTAS.IE-1061	소프트웨어 품질 메트릭 방법론 표준 (1999.12)
TTAS.KO-11.0013	소프트웨어 품질 프로그램 심사지침 (1999.12)
.....
TTAS.IE-730.1	소프트웨어 품질 보증 계획 수립을 위한 지침(2003.12)
TTAS.IE-1008	소프트웨어 단위 시험 표준(2001.12)

한국정보통신기술협회에서는 1998년 이후로 소프트

웨어 품질과 관련된 국내 표준을 다수 제정하였으며 현재 소프트웨어 품질 평가를 위한 기초자료로 활용하고 있다

<표 2>소프트웨어 품질관련 KS표준

표준번호	제목
KS X 2206	소프트웨어 문서화 관리 지침 (1989. 05)
KS X 2216-1	정보기술-소프트웨어 제품 품질-제1부:품질모델 (2001. 04)
KS X 2221	소프트웨어 패키지의 품질 요구사항 및 시험(1998. 12)
.....
KS X 2223-2	정보기술-소프트웨어 제품평가-제2부:계획과 유지보수(2001.04)

<표 3>소프트웨어 품질관련 KICS 표준

표준번호	제목
KICS.IS-12207	소프트웨어 생명주기 공정 표준-ISO/IEC 12207(1996.12)
KICS.IS-8402	소프트웨어 품질의 측정, 평정 및 심사를 위한 기술지원서(1994.12)
.....
KICS.IS-9126	소프트웨어 패키지의 품질요구사항과 시험에 관한 기술지원서(1996.05)

위의 <표1>, <표2>, <표3>에서는 소프트웨어 품질과 관련된 국내 표준에 대한 내용이다. 대체적으로 ISO/IEC 9126을 기반으로 하여 국내 표준을 제정하였다. 현재 국제 표준화 활동에서 연구하고 있는 ISO/IEC 25000 시리즈에 대한 내용도 연구 검토하여 국내에서 적용할 수 있는 방안에 대한 검토와 표준 제정에 대한 검토가 요구되어진다.

현재 국내의 시험기관에서 소프트웨어 품질 평가를 위한 국제 표준모델로 ISO/IEC 9126을 참조하고 있다. ISO/IEC 9126의 참조 모델은 총 4개의 부분으로 구성되어져 있으며 ISO/IEC 9126-2[3]가 소프트웨어 제품에 대한 평가방법을 제시하고 있다. 본 표준 문건에 제시 되어 있는 소프트웨어 품질은 총 6개의 품질 특성에 대해서 평가메트릭을 제시하고 있다. 6개 품질 특성으로는 기능성(Functionability), 신뢰성(Reliability), 사용성(Usability), 유지보수성(Maintainability), 효율성(Efficiency), 이식성(Portability) 등으로 구성되어져 있으며 대체적으로 기능성에 대한 평가와 사용성에 대한 평가는 소프트웨어 제품에 대해서 개발업체에서 제시한 사용자

문서나 제품설명서를 통해서 테스트케이스를 구성하여 평가하고 있다. 그러나 신뢰성이나 유지보수성의 경우 소프트웨어 개발업체에서 시험기간 동안 제시한 고장데이터를 이용하여 평가를 하여야 하나 현재 국내의 소프트웨어 개발업체는 소프트웨어를 평가하기 위한 전문적인 시험인력을 구성하고 있지 않으므로 시험을 통한 고장데이터를 얻는 것은 상당히 어려움이 있다. 소프트웨어 제품에 대한 평가의 어려움이 있어서 ISO/IEC 9126에서 제시하고 있는 신뢰성의 평가 메트릭을 적용하는 데에는 어려움이 있다. 특히 ISO/IEC 9126에서 제시하고 있는 신뢰성의 성숙성은 기존의 시험자료를 이용하여 앞으로 시험을 통해서 얻게 되어질 소프트웨어의 고장과 결함에 대한 자료를 예측하도록 제시되어져 있다. 시험기간 동안 얻어진 고장과 결함의 자료를 이용하여 소프트웨어 신뢰성 성장 모델(Software Reliability Growth Model)에 적용하여 예측을 하여야 하는 문제가 있으므로 이론적으로 해결되어져야 할 어려움이 있다. 아래의 표는 ISO/IEC 9126에서 소프트웨어 신뢰성 중 성숙성에 관련된 평가메트릭이다.

<표 4>ISO/IEC9126의 신뢰성 중 성숙성 평가

메트릭명	세부항목	계산식	값범위
예상잠재 고장밀도	NPFI 예상된 잠재고장의 수	X=ABS(NPFI-NAFI)/SIZE	0<=X
	NAFI 실제 검출된 고장의 수		
	SIZE 제품크기		
예상잠재 결함밀도	FPFU 예상된 잠재결함의 수	X=ABS(NPFI-NAFU)/SIZE	0<=X
	NAFU 실제 검출된 결함의 수		
	SIZE 제품크기		
고장밀도 (결함밀도)	NFAI 검출된 고장의 수	X=NFAI/SIZE	0<=X
	NAFU 검출된 결함의 수	Y=NAFU/SIZE	0<=Y
	SIZE 제품크기		
고장해결	NRFI 해결된 고장의 수	X=NRFI/NAFI Y=NRFI/NPFI	0<X<1
	NAFI 실제로 검출된 고장의 수		0<=Y
	NPFI 예견된 잠재적 고장의 수		
결함해결	NCFU 해결된 결함의 수	X=NCFU/NAFU Y=NCFU/NPFU	0<X<1
	NAFU 실제로 검출된 결함의 수		0<=Y
	NPFU 예견된 잠재적 결함의 수		
평균고장 발생시간	TOPT 프로그램 작동시간	X=TOPT/NAFI Y=TSIB/NAFI	0<X
	TISB 고장발생사이의 시간간격		0<Y
	NFAI 실제로 발견된 고장수		
테스트 적용범위	A 실제로 수행된 시험수	A/B	0~1
	B 요구사항충족 수행될 시험수		

<표 4>에서 제시되어져 있는 신뢰성의 성숙성은 소프트웨어 신뢰성 성장 모델에 적용하여 남아있는 오

류에 대한 값을 예측하여야 한다.

현재 국내의 소프트웨어 개발업체는 시험에 대한 인력이나 장비 등이 미비한 상태이므로 국가 기관에서 이것을 전담하여 소프트웨어 시험을 실시하고 있으나 <표 4>에 제시되어져 있는 소프트웨어 신뢰성의 성숙성은 다소 평가의 어려움이 있다.

소프트웨어 신뢰성 평가라는 관점에서 소프트웨어 신뢰도 성장 모델을 제시하고 신뢰도 평가에 연구를 시작한 것은 Jelinski-Moranda(JM:1972)[2]에 의해서 이루어졌으며 JM에 의해서 제시된 모델은 지수함수의 성질을 이용하였다. 지수함수는 하드웨어 신뢰성 평가에 적용되어졌던 모델로써 하드웨어의 특성과 다른 소프트웨어의 특성을 고려하여 모수의 범위를 만족하는 조건을 제시하고 신뢰성을 평가할 수 있는 방안에 대해서 제시하였다.

$$f(t_i) = \phi(N-i+1)exp(-\phi(N-i+1)t_i)$$

ϕ : 실패강도

N: 총 고장 수

위의 모델에서 실패율 $\lambda(i) = (N-i+1)\phi$ 이고 이러한 실패율에 따라서 여러 가지 모델이 제시되어졌다

$$\lambda(i) = ((N-i+1)\phi)^2$$

$$\lambda(i) = exp(-(N-i+1)\phi)$$

소프트웨어 신뢰도 성장 모델은 지수함수의 경우에도 위에 제시된 실패율에 따라서 다양하게 제시되어질 수 있으며 고장 데이터의 형태에 따라서도 다양한 연구 결과가 발표되어져 있다. 위의 소프트웨어 신뢰도 성장 모델에서 ϕ, N 은 추정되어져야 할 모수이므로 최우추정법에 의해서 추정이 되어진다. 비동질적 포아송 분포 모델을 제안한 Goel-Okumoto[1]는 포아송 과정의 성질을 이용한 소프트웨어 신뢰도 성장 모델을 제안하였으며 이 경우 평균값 함수는 아래와 같다

$$m(t) = a(1 - exp(-bt)), a > 0, b > 0$$

여기에서 a는 시험과정에서 발견되어질 수 있는 전체 결함의 수이고 b는 실패발생 비율이다. 비동질적 포아송과정의 소프트웨어 신뢰도 성장 모델에서 모수는 최우추정법을 이용하였다. 모수 추정 방법도 계속적으로 연구되어지고 있는 과제중에 하나로써 최우추정법, 최소자승추정법, 깁스샘플링방법 등 다

양한 연구 결과가 발표되어져 있다. 이와 같이 추정된 모수를 이용하여 예상잠재 고장수와 결함수를 계산할 수 있게 되어진다. 계산된 결과는 소프트웨어 신뢰도 평가 메트릭인 예상잠재 고장밀도와 예상잠재 결함밀도를 측정할 수 있으며 신뢰성 값을 평가할 수 있다. 그러나 이러한 모델의 복잡성과 모수 추정에 대한 수리적인 어려움으로 인하여 현재는 소프트웨어 시험 결과에 활용되어지고 있지 못한 실정이다.

3. ISO/IEC 25000 연구 동향

ISO/IEC 25000 프로젝트는 SQuaRE(Software Quality Requirement and Evaluation)라는 이름으로 연구가 진행되어지고 있다. ISO/IEC JTC1/SC7/WG6는 소프트웨어 품질 평가와 관련되어 국내 전문가들이 연구하는 그룹으로 현재 많은 관심과 연구를 진행 중에 있으며 국내의 시험 사례들을 다수 적용할 수 있는 방안을 검토하고 있다. 현재 국제 표준 프로젝트의 연구 진행은 곧 국내의 소프트웨어 품질 관련 표준에 대한 제정 연구가 진행되어져야 할 것으로 보여진다. 소프트웨어 제품의 품질 평가 모델인 ISO/IEC 9126[3]과 소프트웨어 평가절차에 대한 국제표준 ISO/IEC 14598[4]을 새롭게 통합하여 내용을 보강하는 소프트웨어 품질 평가 모델 SQuaRE(Software Quality Requirement and Evaluation)에 대한 연구가 현재 진행중에 있다. 1994년 이후로 분리되어 사용되어졌던 국제표준 ISO/IEC 9126과 ISO/IEC 14598을 통합하기 위한 연구로 두 개의 표준에서 불일치하는 부분들을 새롭게 정리하여 국제 표준을 새롭게 정리하기 위한 프로젝트이다. SQuaRE 프로젝트의 5개 부분에 대한 주요 내용으로는 Quality Management Division, Quality Model Division, Quality Metrics Division, Quality Requirement Division, Quality Evaluation Division이다.

SQuaRE 매트릭스는 측정개념을 명확히 하고 측정 프리미티브를 정의하고 기능적 요구사항을 포함하는 품질 메트릭을 구성하고 있다.

SQuaRE 매트릭스에서도 ISO/IEC 9126과 같이 6가지 품질특성인 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성의 품질요구사항을 반영하고 있으며 측정 프리미티브 클래스(Measurement Primitive Class:MP)를 구성하여 나타내고 있다.

SQuaRE 프로젝트에서 ISO/IEC 25020[9]에서는 기본적인 측정 프리미티브에 대한 소개와 개요를 설명하고 있다. 아래의 <표 5>는 ISO/IEC 25000 시리즈에서 제시한 측정 프리미티브 클래스 내용의 예제를 제시한 것이다. SQuaRE 프로젝트에서는 소프트웨어 품질 평가를 좀더 정량적으로 하기 위해서 측정 프리미티브를 제시하고 적용할 수 있는 방안을 검토하고 있다고 밝히고 있다. 본 연구에서 계속적인 연구 과제로 소프트웨어 신뢰성 평가를 위해서 ISO/IEC 25000 시리즈에 제시되어져 있는 측정 프리미티브를 구성하고 구성된 내용을 중심으로 하여 소프트웨어 신뢰성 평가 메트릭을 제안하는 방안을 추진하려 하고 있다

<표 2> 측정 프리미티브 클래스

1. Number of Functions	The count of all the functions that satisfy the condition given in the MP definitions of this class. Note: the functions can be for example, required, implemented, tested, essential, optional, or any combination of these and more.
2. Number of Failures	The count of all failures which occur in a given time span and which also satisfy the condition given in the MP definitions of this class. <i>Examples of MPs:</i> # of expected failures, # of detected failures, # of resolved failures, # of failures of a given severity level.
3. Number of Faults	The count of software product faults detected (or estimated) in a given software product component and satisfy the condition given in the MP definitions of this class. Note: in the MPs, the # fault of a given category, # faults of a given severity, # faults successfully corrected, etc.

위의 <표 5>에서와 같이 측정 프리미티브를 구성하기 위해서는 시험자, 사용자, 개발자 모두가 소프트웨어의 특성을 고려하여 구성하여야 한다.

특히 ISO/IEC 25000 시리즈는 사용자 관점의 요구사항을 소프트웨어 품질 평가에 충분히 반영하기 위한 기초적 내용을 연구하고 있다. 현재 진행되어지고 있는 ISO/IEC 25000 시리즈의 경우 연구 프로젝트 내용이 소프트웨어 품질 평가를 위해서 구체적이지 못하다는 지적을 받고 있다. 소프트웨어 신뢰성 평가를 위해서는 구체적인 소프트웨어 측정 프리미티브의 구성에 대한 연구가 진행되어져야 할 것으로 보여진다. 아래의 표는 ISO/IEC 25000 시리즈 중에서 25000 표준 문서의 연구 진행사항에 대한 내용이다. 본 문건은 국제 표준으로 투표과정을 완료한 상태이다

<표 6> ISO/IEC 25000 시리즈의 연구동향

NP	WD	CD	2CD	FCD	FDIS
		2002.10	2003.4	2004.5	2005.10

현재 진행되어지고 있는 ISO/IEC 25000 시리즈의 경우 소프트웨어 품질 평가에 있어서 많은 변화가 있을 것으로 기대되어진다. ISO/IEC 9126과 비교하여 좀더 정량적인 품질 평가 모델을 제시하고 있으며 측정을 위한 측정 프리미티브를 제시하여 평가를 할 수 있는 방안을 제시하고 있다. 이러한 관점에서 현재 진행되어지고 있는 국제 표준 문건의 검토와 연구를 통해서 국내에서 적용할 수 있는 방안을 검토하여야 할 것이다.

4. 신뢰성 평가의 적용사례

본 자료는 소프트웨어 시험 기관에서 6일간 시험을 거쳐서 발견된 고장 수에 관한 자료이다. 관측 방법은 총 시험 시간에 대해서 발견된 오류 수를 예측하는 방법으로 실시하였다. 발견된 오류 수에 대한 데이터는 아래의 표와 같다

<표7> 시험날짜별 고장수에 대한 자료

첫째	둘째	세째	네째	다섯째	여섯째
8	12	19	11	5	2

위의 오류수에 대한 고장 발견 시간은 포아송 과정의 평균을 함수를 이용하여 예측하였다. 각각 시험 날짜별로 1시간당 발생된 고장수를 중심으로 하여 고장 발견시간에 대한 자료를 예측하였다. 그결과 예측된 6일간의 고장시간에 대한 자료는 아래와 같다.

<표 8> 고장시간의 예측자료

첫째	63,60, 65, 61, 60, 61, 59, 49
둘째	48, 52, 35, 31, 33, 41, 36, 35, 41, 36, 41, 43
세째	22, 25, 34, 21, 32, 34, 31, 31, 31, 24, 29, 20, 35, 27, 22, 27, 23, 19, 25
네째	49, 35, 39, 48, 34, 41, 34, 30, 51, 48, 37
다섯째	115, 89, 105, 82, 96
여섯째	237, 245

위의 고장시간에 대한 자료를 소프트웨어 신뢰성 성장 모델을 이용하여 신뢰성을 평가한 결과 실패고장률은 0.000355이고 신뢰도는 0.49로 낮게 측정되었다. 그리고 현재 까지 발견된 고장 수 57개를 이용하여 전체 총 고장수를 예측한 결과 88개로 예측되

었다. 소프트웨어 신뢰도를 예측하기 위해서는 신뢰도 성장 모델에 따라서 다소의 차이가 있을 수 있으므로 정확한 모델을 설정하여 모수를 추정하고 추정 결과를 이용하여 신뢰도를 평가하여야 한다.

5. 결론 및 연구 방향

국내 소프트웨어 품질 관련 전문가들은 소프트웨어의 품질 향상이란 관점에서 소프트웨어 품질 평가를 위한 기초 자료를 구성하여야 할 것으로 보여진다. 이와 같은 문건은 국내 표준으로 제정되어야 하며 국제 표준으로 제안할 수 있도록 연구되어야 할 것이다. 특히 현재 진행 중에 있는 ISO/IEC 25000 시리즈에 맞추어서 국내의 표준도 제정되어질 수 있도록 병행하여 연구가 진행되어야 할 것으로 보여지며 특히 신뢰성 평가를 위해서 시험기관의 시험 경험과 테스트 케이스 등을 조사하여 사용자관점, 개발자관점, 시험자관점의 요구조건을 만족하는 평가항목을 구성하여야 할 것으로 보여진다. ISO/IEC 25000 시리즈에서 제시하고 있는 측정 프리미티브 클래스를 좀더 구체적으로 제시하고 적용할 수 있는 방안에 대한 검토가 필요할 것으로 보여진다.

국내의 소프트웨어가 국내 시장을 겨냥하는 것이 아닌 국제 시장에서의 경쟁을 위해서는 국외적으로 인정 받을 수 있는 소프트웨어 품질 평가에 대한 표준 제정이 필요할 것으로 보여지며 벤치마킹을 통해서 국내 시험기관의 시험 결과에 대한 내용을 국외시장에서 인정할 수 있는 제도적인 내용이 필요할 것으로 보여진다.

참고문헌

- [1] Goel..A.L, & Okumoto.K, "Time Dependent Error Detection Rate Model for Software Reliability and Other Performance Measures", IEEE Trans. Reliability, R-28, pp.206-211, 1979
- [2] Jelinski.Z, & Moranda.P.B, 'Software Reliability Research, In Statistical Computer Performance Evaluation', New York, Academic Press, pp.465-484, 1972.
- [3] ISO/IEC 9126, "Information Technology Software Quality Characteristics and metrics-Part 1,2,3.

- [4] ISO/IEC 14598, "Information Technology Software Product Evaluation -Part 1,2,3,4,5,6.
- [5] ISO/IEC 12119, "Information Technology Software Package Quality requirement and testing".
- [6] ISO/IEC 25000 (Software and System engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Guide to SQuaRE), 2005. 10, FDIS
- [7] ISO/IEC 25001(Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Planning and management), 2005. 5, CD.
- [8] ISO/IEC 25010(new) Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Quality model
- [9] ISO/IEC 25020(Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement reference model and guide)
- [10] ISO/IEC 25021(Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement primitives)
- [11] ISO/IEC 25030(Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Quality requirements)
- [12] ISO/IEC 25040(Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation reference model and guide)
- [13] 정혜정, "의료용 소프트웨어의 평가기준 개발", 식품의약품안전청, 최종보고서,2002. 12.
- [14] 정혜정, 정원태, "S/W 신뢰도 평가 기술 및 품질관리 적용방안", 산업자원부 기술표준원, 2003. 10.
- [15] 정혜정, 정원태, "S/W 신뢰성 성장 모델을 적용한 신뢰성 평가 기술", 산업자원부 기술표준원, 2004. 9.
- [16] 정혜정, 정원태, 정영은, 신석규, 양해술 "소프트웨어 품질 표준화 동향 조사", 한국정보처리학회 추계학술발표논문집, 2005. 11.
- [17] 정혜정, 정원태, 이하용 "국내외 S/W 품질 평가 기술 및 표준 동향 연구",한국정보통신기술협회최종연구보고서, 2005, 11,
- [20] 정혜정., 소프트웨어 신뢰성 평가를 위한 평가 방안, 산업자원부, 2005. 5
- [21] 정혜정, "소프트웨어 품질 평가 표준에 관한 연구",사단법인 한국정보처리학회 소프트웨어 공학논문집, 2006.01.
- [22] 정혜정, 정원태, 정영은, 신석규 "소프트웨어 신뢰성 품질 평가에 대한 표준화 연구", 사단법인 한국정보처리학회 춘계학술발표논문집, 2006, 5