소프트웨어 분야의 리스크경영시스템 도입방안에 관한 연구

On The Adoption of Risk Management System in Software Industries

김 종 걸*·김 형 만**·김 인 희**

Jong-Gurl, Kim*· Hyung-man, Kim**· In-Hee, Kim**

Abstract

다양한 리스크 문제가 발생하고 있는 환경 속에서 리스크에 대한 적절한 대응을 하고 안정화를 도모함과 동시에 리스크 문제가 표면화되어 초래하는 영향을 극소화 하기 위한 새로운 패러다임의 경영시스템 도입이 주요 전략과제로 대두되고 있다.

본 연구에서는 리스크 경영시스템(Risk Management System)인 IEC 61508의 리스크 규격을 중심으로 소프트웨어 분야가 가지고 있는 리스크 문제를 효과적으로 관리할 수 있는 대안 모색을 가능하도록 하고 소프트웨어 분야에서의 리스크 제로와 안전강화를 위하여 기초 자료를 제시하였다.

keyword : 리스크경영시스템(Risk Management System), 소프트웨어 품질관리(Software Quality Control)

C&C(Computer & Communication) 혁명 이후 오늘날 사회에서 소프트웨어가 차지하는 비중은 날로 높아져가고 있다. 소프트웨어의 규모나 수행능력, 용량이 날로 복잡화, 대형화되어감에 따라 고신뢰성과 고품질의 소프트웨어를 개발하기 위한 체계적이고 합리적인 소프트웨어개발시스템 구축이 어느때보다 절실히 요구되고 있다[3]. 이에 따라 “소프트웨어 공학”이라는 학문이 점점 활성화되어가고 있으며, 소프트웨어 품질에 관한 전반적인 관심도 한층 부각되어지고 있다. 이러한 소프트웨어 품질에 대한 체계의 확립과 방법론에 대한 관심은 앞으로도 더욱 높아지리라 예상된다[3].

* 성균관대학교 시스템경영공학과

** 성균관대학교 산업공학과

1. 서론

국내 실정에 적합한 품질에 관련된 소프트웨어 기초 이론 분야가 보다 발전하기 위해서는 우리 설정에 맞는 소프트웨어 품질평가 방법론과 자동화 도구 개발의 기초 연구가 선행되어야 한다. 그러나 국내에서는 소프트웨어 품질 평가 방법론과 도구의 필요성을 충분히 인식하면서도 실질적인 연구가 활발히 진행되지 못한 실정이며, 현재의 연구 현황은 많은 문제점과 과제를 안고 있는 것이 사실이다[3]. IT융합의 진전으로 소프트웨어 결함으로 인한 열차, 선박, 항공기, 의료장비 등의 사고 발생우려, 석유화학 공장, 철도 자동차, 항공분야 등 고도의 소프트웨어기능 안전성이 필요한 분야 및 하드웨어 분야와 달리 소프트분야는 신뢰성이 정량적 확보 방안이 아직 초보 단계의 문제점들이 그 예이다.

2. IEC 61508 고찰

2.1 적용범위

이 국제표준은 E/E/PE(전기적/전자적/프로그램 할 수 있는 안전관련 시스템)이 안전 기능을 수행하는데 사용되어지곤 할 때와 같은 측면을 포함한다[6]. 이는 전적으로 적용분야 요구에 대해서도 사용 가능하고 그것에 의해서 발생하는 특별한 분야에도 적용가능하다.

2.2 용어 및 정의

유해, 위해, 위해상태, 위해사건, 리스크, 정황리스크 등 총 78개의 용어에 대해 정의되어있다[8].

2.3 리스크 경영을 위한 요구사항

요구에 대한 정도를 결정하는 하나의 요소를 결정하는 것은 일반적으로 가능하지 않다. 그것은 많은 요인에 의존적이고, 일반적으로 상술하면, E/E/PES나 소프트웨어 안전수명주기 단계와 활동에 의존적이다. 요소들은 다음을 포함한다[6].

- 위험요인의 본질
- 안전도 수준
- 결과와 리스크 절감
- 이행기술의 종류
- 시스템의 크기

- 포함된 팀의 수
- 실제적인 분배
- 설계의 참신성

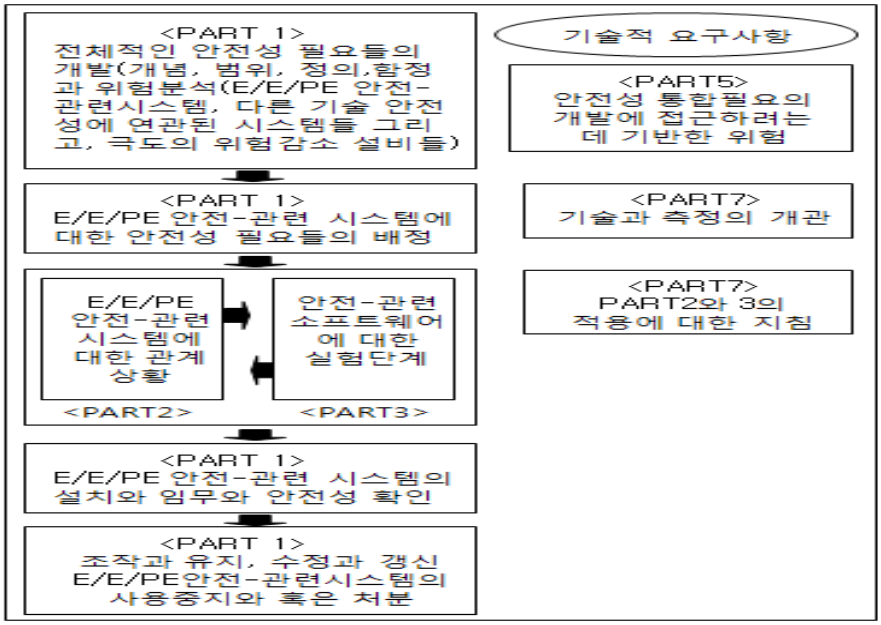
2.4 주요내용

이 국제표준은 전기, 전자, 프로그램화 할 수 있는 전자제품(E/E/PES)으로 구성된 시스템이 안전기능을 수행하기 위한 모든 안전 수명주기 활동을 위한 일반적 접근방법을 강조한다. 이통합화된 접근은 전자에 기반을 둔 안전에 관련된 모든 시스템이 개발되기 위해 채택되어 지고 있다. 주된 목적은 표준 적용의 활성화를 수행하는 것이다. Table 1은 전체적인 구성을 보여준다.

[표 1]. 국제적인 표준

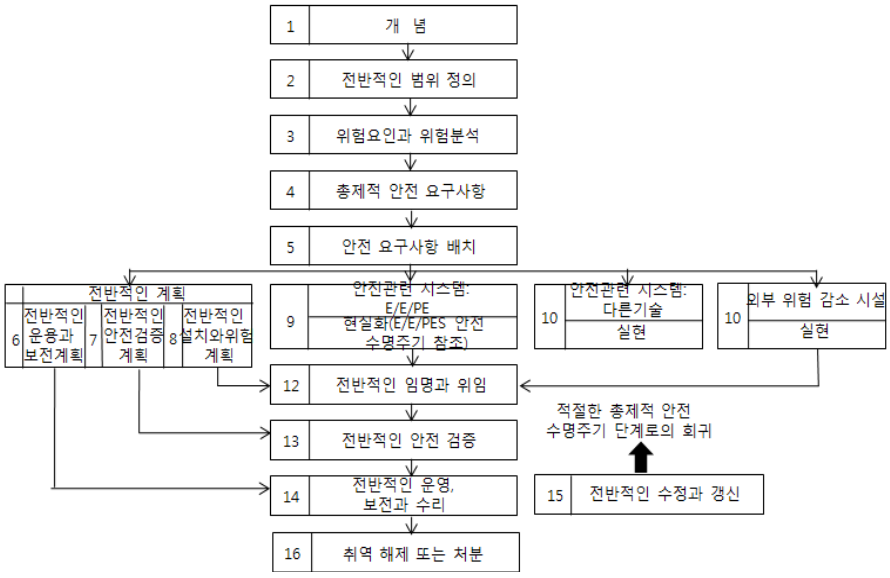
구 분	개 념
61508-1	- 1998, E/E/PE안전관련 시스템의 기능적 안전성 - PART1 : 일반적 요구사항
61508-2	- 2000, E/E/PES안전관련 시스템의 기능적 안전성 - PART2 : 전기/전자/프로그램 할 수 있는 전자 시스템의 요구사항들
61508-3	- 1998, E/E/PES안전관련 시스템의 기능적 안전성 - PART3 : 소프트웨어 요구사항들
61508-4	- 1998, E/E/PES안전관련 시스템의 기능적 안전성 - PART4 : 정의와 약어
61508-5	- 1998, E/E/PES안전관련 시스템의 기능적 안전성 - PART5:안전도 수준의 정의를 위한 방법의 예시들
61508-6	- 2000, E/E/PES안전관련 시스템의 기능적 안전성 - PART6 : PART2와 3의 적용에대한 가이드라인
61508-7	- 2000, E/E/PES안전관련 시스템의 기능적 안전성 - PART7 : 기법과 척도에 대한 개관

그림 1은 IEC 61508-1에서 7까지 IEC 61508의 종합적인 구성을 보여준다.



[그림 1]. IEC 61508 종합적인 구성[6]

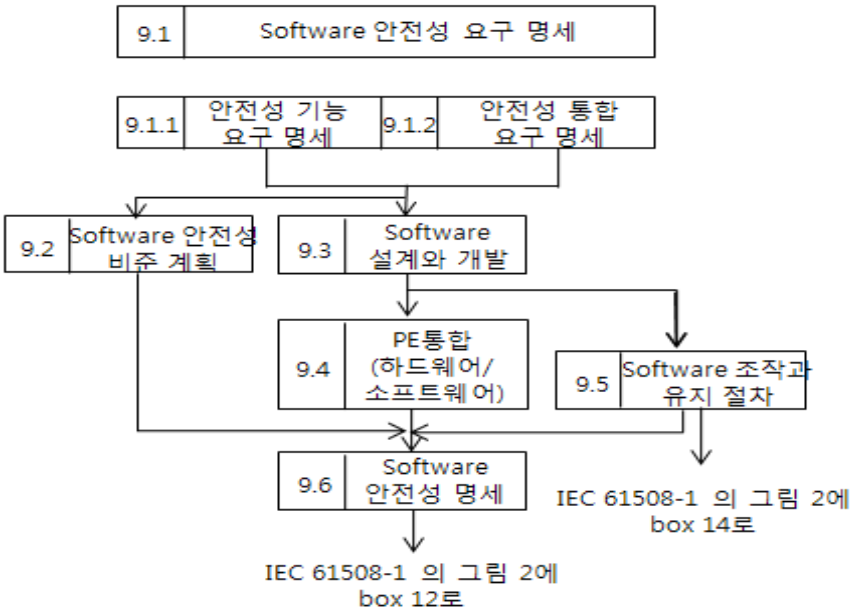
2.5 리스크경영 개념과 절차



[그림 2]. 전반적인 안전수명주기[6]

E/E/PE 안전 관련 시스템에 대해 요구되는 안전도 수준을 이루기 위해 필요한 모든 활동들과 함께 체계적 방법으로 취급하기 위해 이 규격은 기술적인 구조로서 전반적인 안전수명주기(그림 2)를 채택한다.

전체 리스크 경영시스템 가운데 중간 시스템으로써 IEC 61508에서 다루는 소프트웨어 수명주기는 다음과 같다.



[그림 3]. 소프트웨어 안전 수명주기[7]

3. 소프트웨어 품질 고찰

3.1 소프트웨어 정의

IEC 61508에 의하면 소프트웨어는 프로그램, 절차, 데이터 또는 규칙과 어떤 관련된 문서와 이에 부속되는 시스템을 운영상의 데이터로 구성되는 지각적 창조물로 정의하였다.

한편, Pressman에 의하면 소프트웨어는 원하는 기능을 수행하는 컴퓨터 프로그램과 프로그램이 설계, 이용, 개발, 추진, 보수하는데 필요한 문서 체계라고 정의하였다[1].

이와 같이 소프트웨어는 “실행할 때 원하는 기능과 성능을 제공해 주는 명령어, 프로그램이 정보를 알맞게 조작하도록 해주는 자료구조, 프로그램의 연산과 사용을 설명해주는 문서” 라는 차원에서 정의되어진다.

소프트웨어를 이해하기 위해서는 사람들이 다른 분야에서 구축했던 것과 다르게 만들어진 소프트웨어 특성 등을 조사해보는 것이 중요하다. 하드웨어를 구축할 때, 인간의 창조적 과정(분석, 설계, 구축, 검사)이 최종에는 물리적인 형태로 변환된다. 만약에 우리

가 새로운 컴퓨터를 구축하려면 초기에 윤곽을 그리고, 공식적인 설계도를 작성하고, 일반적인 원형을 물리적인 제품(VLSI 칩, 회로판, 전력공급 등)으로 발전시켜 나간다.

소프트웨어는 물리적인 시스템 요소라기보다는 논리적인 요소이다. 그러므로 소프트웨어는 하드웨어의 특성에 비해 상당이 다른 특성들을 갖고 있다[3].

3.2 소프트웨어 품질 정의

소프트웨어 개발자들은 소프트웨어가 대형화함에 따라 소프트웨어 개발 방법과 유지보수 비용 이외에도 소프트웨어 품질 보증(Quality Assurance)에 대한 큰 관심을 갖고 있다.

기존에는 좋은 소프트웨어라고 하는 것은 신뢰성이 높은 소프트웨어라고 생각해 왔다. 이 때문에 프로그램 테스트 기간 중에 가능한 한 많은 오류를 찾아내어 수정하고 제거하여 좋은 소프트웨어를 만들어 왔다. 소프트웨어 품질이 좋다고 말하는 것은 그 본질 자체가 명확하게 정의되어 있지 않아 객관적으로 평가할 수 없다. 품질관리 용어에서 품질이라는 것은 상품 혹은 서비스가 사용목적에 만족되고 있는지를 결정하기 위한 평가의 대상이 되는 고유성질로 정의하고 있다. 소프트웨어 품질은 보는 사람들의 견해가 다르다. 예를 들면 프로젝트 관리자에게 있어서는 품질은 제한된 비용과 기간 내에 기능적 요구사항을 구현하는 것을 말한다. 그리고 사용자에게 품질이라는 것은 사용하기가 용이하고 빠른 응답시간을 갖는 것이다. 또한 개발자에게 품질이라는 것은 수행을 올바르게 하고 프로그램 표준에 맞도록 하는 것이다[4].

3.3 소프트웨어 품질 평가

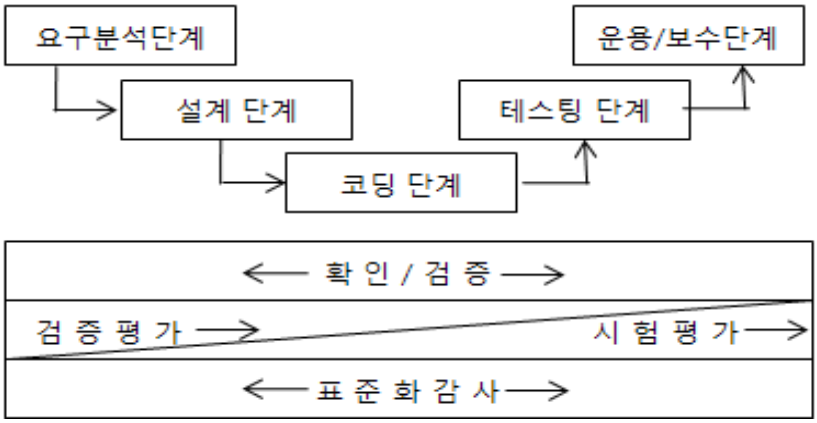
소프트웨어 생명주기의 첫 단계인 요구분석 단계에서부터 마지막 단계인 유지 보수 단계까지의 총비용중 유지보수의 비용은 전체 비용의 약 80%에 이르고 있음을 많은 연구보고서에서 지적해 왔으며, 또한 유지보수 비용을 절감하기 위한 많은 개발 방법론이 연구되어 오고 있다[3].

이러한 유지보수 비용을 절감하기 위해서는 무엇보다도 신뢰성 있는 소프트웨어를 개발하는 것이 중요하며, 이러한 신뢰성 있는 소프트웨어를 개발하기 위한 방법론으로 프로그램 테스팅, 품질평가, 구조적 설계, 객체지향 설계 등 다양한 방법론들이 연구되어져 오고 있다. 이중 소프트웨어 품질평가는 소프트웨어 개발 종료 후, 개발과정에서 생성된 생산물과 최종 생산물인 소프트웨어를 평가함으로써 생명주기 전 과정에서 발생된 오류를 검출할 수 있고, 이를 바탕으로 소프트웨어가 실제 요구자의 요구를 어느 정도 충족시키고 있는지를 평가함으로써 소프트웨어의 신뢰성을 평가하는 평가 방법론이다.

본 연구에서는 평가 기법을 크게 확인/검증과 표준화 감사로 구분하였다. 확인/검증의 주된 관점은 요구사항의 적합성을 입증하기 위한 것이고, 표준화 감사는 개발에 관련된 제반 표준 및 지침에 대한 적합 여부를 판정하기 위한 것이다[3]. 확인이란 어느 단계의 개발 제품이 최초의 사용자 요구 또는 소프트웨어 요구에 적합한지를 입증하

기 위한 활동을 의미하며, 검증은 어느 단계의 개발 제품이 이전 단계서 설정된 개발 규격을 충족시키는지의 여부를 판단하기 위한 활동이다[3].

확인/검증은 다시 검증평가와 시험평가로 분류할 수 있다. 검증평가는 요구사항의 정의나 설계와 같이 초기 단계에서 작성되는 개발 규격서의 충분성 평가를 위주로 해왔으며, 시험평가에서는 프로그램자체를 대상으로 하여, 구체적인 시험 사례를 입력한 그 결과를 분석하는 평가 기법이다[3]. 이와 같은 평가기법을 수명주기에 적용시켜 보면 아래 그림과 같다.



[그림 4]. 소프트웨어 품질 평가 기법의 적용 관계 [3]

4. 리스크경영시스템의 소프트웨어 분야 적용단계

IEC 61508 리스크경영시스템은 소프트웨어의 안전 수명주기 과정에서 일어나는 모든 사항에 주의를 기울여 소프트분야에서의 리스크를 초래할 수 있는 문제점들을 개선해 나가는 노력에 초점을 두고 있다. 이러한 과정은 소프트웨어에서의 리스크경영시스템 과정으로서 소프트웨어의 리스크경영시스템을 실제 기업에서 적용하기 위한 과정을 제시하면 다음과 같다.

4.1 IEC 61508 리스크경영시스템의 구축

IEC 61508 리스크경영시스템을 위한 계획과 실행은 실제적으로 조직의 전략적 계획과 부합되어야 한다. 아울러 조직의 설립목적, 활동내용, 전망 및 목표에 대한 품질경영의 원칙과 절차 및 과정이 잘 융합되어야 한다[2]. IEC 61508을 위한 조직화는 조직평가, 사용자 이해, 전망 및 원칙의 설정을 하는 과정이다.

우선 조직평가는 IEC 61508을 도입하기 이전에 소프트웨어 분야가 IEC 61508 리스크경영시스템에 적합한 문화적 환경인지의 여부를 평가하는 것이다[2]. IEC 61508 리스크경영시스템에 적합한 문화적 환경은 리스크 과정의 형성과 리스크행동의 조장에

관한 분명한 가치와 믿음을 가진 조직이며, 이들 가치와 믿음은 IEC 61508 리스크경영시스템을 지원한다.

또한 사용자에 대한 이해는 전체 IEC 61508 리스크경영시스템 과정의 핵심적 요소로서, 사용자 서비스와 만족에 초점을 맞춘다.

마지막으로 전망 및 원칙의 개발은 소프트웨어 분야가 추구하는 미래와 미래 소프트웨어 분야가 수행해야 할 활동의 안내를 위한 원칙이 무엇인가에 관한 최고 경영자와 모든 조직구성원간의 일반적인 이해를 도모하는 행위이다.

이러한 합의는 조직의 비전과 가치의 공식적인 표현의 기초가 된다. 비전은 시간으로 조직의 원하는 것이 무엇인가를 분명하게 적극적이고, 강력하게 제시해야 한다. 원칙과 활동에 대한 안내를 지원하는 잘 설정된 비전은 조직이 초점을 두고 있는 일반 목표를 향한 강력한 도구가 된다.

4.2 소프트웨어 분야에서의 IEC 61508 리스크경영시스템 실행

리스크경영시스템의 계획 및 조직적 활동과 이에 따른 실제적인 적용간에 활동들은 소프트웨어분야에서 제공하는 소프트웨어 자체의 구조나 기능, 개발자와 사용자의 구별이 포함된다.

이 과정에서 IEC 61508-3에서 시행되는 실현단계에 있는 소프트웨어 안전수명주기 단계는 소프트웨어분야와 관련된 조직의 가장 중요한 부분이 된다. 안전수명주기를 파악하여 소프트웨어의 목적을 계속적으로 만족시키기 위해 안전수명주기에서 요구되는 요구가 무엇인지를 정확하게 파악하여야 한다.

소프트웨어를 평가의 목적은 개발 완료된 소프트웨어에 대한 품질 평가만을 의미하는 것이 아니라, 소프트웨어 생명주기 전과정에 걸쳐 개발중인 소프트웨어의 품질을 측정하여, 나타난 문제점을 즉시 보완해 나감으로서, 품질목표를 효율적으로 향상시키는데 있다.

또한 IEC 61508-3에서 제안하는 소프트웨어의 안전무결과 개발의 수명주기의 과정을 중시, 사용자에게 어려운 경험을 요구하거나 세심한 주의가 요구되는지의 여부, 자금절약이나 시간절약 등 낭비제거 요소를 고려하여, 작업흐름도를 측정하고 확인하여야 한다.

4.3 IEC 61508 리스크경영시스템 소프트웨어분야에서의 적용방안

IEC 61508 리스크경영시스템에서의 소프트웨어 리스크평가를 통한 리스크 향상을 기대하기 위해서는 우선적으로 소프트웨어 리스크에 대한 국제 표준을 인식하고 수용하도록 보급하는 것이 필요하다. IEC 61508에 대한 보급방안으로는 첫째, 규격 도입에 따른 기업의 부담해소를 제안한다. 새로운 제도를 도입하기 위하여 기업은 그 비용과 인력에 대한 부담으로 인한 인증의 참여에 망설임을 가질 수 있다[5]. 이러한 점을 해결하고자 본 제도의 도입에 따른 효과를 홍보하고 해외의 소프트웨어 분야에서의 리스크 경영시스템 효과 자료를 제공하여야 한다.

또한, 기업에 대하여 리스크 경영진단 수단을 활용한 소프트웨어 리스크관리 자가진단 시스템을 구축하여 규격 도입의 효과를 간접적으로 평가할 수 있도록 해야 한다.리스크경영시스템의 네트워크를 구성하여 정보의 교류 및 비용 절감방안을 제시하고 기존의 유사시스템 인증 업체에 대하여 시스템부분의 심사를 생략하는 등 기업의 참여에 따른 부담을 최소화하는 것도 하나의 방안으로 활용될 수 있다.

[표 2]. 리스크경영시스템 제도 도입시 문제점 및 해결방안

도입시 예상 문제점	해결방안 및 인식제고
IEC61508 리스크 경영시스템 도입에 따른 기업의 비용 및 인력 부담	- 선진국의 IEC61508리스크 경영시스템 효과 조사 및 홍보
	- 기존 유사시스템 인증업체의 경우 심사시 시스템 부분은 생략
	- 리스크경영 진단 툴을 활용한 리스크 낭비비용의 자가진단 시스템 구축
	- 선진국의 IEC61508 리스크경영 효과 조사 및 홍보
	- 기존 유사시스템 인증업체의 경우 심사시 시스템 부분은 생략
형식적인 제도로의 변질 가능	- 효과적인 리스크 관리기법을 지속적으로 제공함
	- 엄격한 인증 및 사전관리 심사
	- 인증을 받은 기업에 대한 평가 및 우수 기업시상
기존 인증제도와 유사하여 혼동 가능	- 리스크경영과 기존제도와의 차이점 조사 및 홍보
	- 성과중심의 제도가 될 수 있도록 운영
	- 인정 및 인증기관을 기존 기관과 차별화 함
	- 리스크개선팀 상시 운영
	- 인증 및 심사 기준을 엄격히 통제

두 번째는, 규격의 효용성 증대이다. 효과적인 소프트웨어 분야에서의 리스크 관리기법을 지속적으로 제공하여 참여기업에 대한 유인책으로써 사용하여야 하며, 엄격한 인증과 사전 관리에 대한 심사를 통하여 본 제도 인증을 받은 기업의 수준을 일정수준 이상으로 유지하도록 해야 한다.

또한, 참여 기업에 대한 평가를 통하여 우수기업에 대하여 시상을 하는 등 사전관리에 많은 비중을 두어야만 제도가 안정화될 수 있다. 세 번째는, 소프트웨어 분야에서의 IEC61508을 통한 리스크진단 의무화 제도이다. 정부에서는 장기간 지속되고 있는 소프트웨어 분야에서의 IEC61508을 통한 리스크에 대한 리스크진단을 제도화해야 한다.

리스크진단이란 리스크관련 전문 기술장비 및 인력을 구비한 진단기관으로부터 소프트웨어에서의 요구/분석단계, 설계단계, 코딩단계, 테스팅단계, 운영/보수단계 등 소프트웨어 수명주기 전반에 걸쳐 소프트웨어의 수명주기 흐름을 파악하여, 손실요인 발굴 및 리스크절감을 위한 대책등 최적의 개선안을 제시하는 기술 컨설팅 제도를 말한다.

4.4 리스크경영시스템의 통합

소프트웨어에서의 리스크경영시스템은 모든 소프트웨어의 조직 구성원들에 의해 이해되는 운영방법 및 리스크가치로 통합되어야 한다. 리스크경영시스템의 시범실시 결과는 모든 소프트웨어의 조직구성원에 의해 평가하고 표현된 결과이다.

여기서 제시된 제안은 문제 해결과정이나 팀훈련에 적용하여 계속적인 리스크 증진을 도모할 수 있는 기회로 삼아야 한다. 아울러 전략적인 리스크 경영계획을 창안하거나 부서별, 분야별 계획을 수립하여 리스크경영시스템을 전체 조직으로 확대, 통합하여야 한다.

5. 결 론

본 논문에서는 리스크관리 부재 시 대형사고가 수반될 수 있는 소프트웨어 분야의 전략 중 하나인 리스크경영시스템의 구축과 운영에 대한 기초자료를 제시하기 위해 리스크경영시스템인 IEC 61508을 분석하였다. 분석을 통해 도출된 소프트웨어 분야의 적용사항을 현업에 적용한다면 요구분석단계, 설계단계, 코딩단계, 테스팅단계, 운영/보수단계의 수명주기 전 과정에서 안전성강화와 리스크 제로를 위한 기초 자료가 될 것이다.

이 자료를 토대로 한국의 소프트웨어 분야의 안전수준이 세계수준으로 발전하고, 국가 경쟁력을 강화하기 위해서는 향후 소프트웨어 분야에서의 리스크관리 연구가 계속 지속되어야 할 것이다.

6. 참 고 문 헌

[1] 권오탁, “ 소프트웨어 품질관리를 위한 품질평가 기술”, The Journal of Information Systems Review, pp.1-16, 1994

[2] 박재용, “ 정보지원센터의 정보서비스 질향상을 위한 품질경영시스템 도입방안에 관한 연구”, 경영정보연구, pp.1-23, 2002

[3] 김종걸, 소프트웨어 품질관리, 성균관대학교, 2007

[4] 김신홍, 소프트웨어 공학, 내하출판사, 2002

[5] 임기추, 주요국의 에너지경영시스템 추진현황 및 국내 도입방안 연구, 에너지경제연구원, 2007

[6] IEC, IEC 61508-1, General Requirements, 1998

[7] IEC, IEC 61508-3, Software Requirements, 1998

[8] IEC, IEC 61508-4, Definitions and Abbreviations, 1998

저 자 소 개

김 종 걸
서울대학교 계산통계학에서 석사
한국과학기술원 산업공학과에서 박사학위
현재 한국품질보증/PL 연구회 회장으로 활동
성균관대학교 시스템경영공학과 교수로 재직
주 소 : 경기도 수원시 장안구 천천동 300번지 성균관대학교 시스템경영공학과 27416호실

김 형 만
상지대학교 산업공학과를 졸업
성균관대학교 산업공학과 석사
성균관대학교 산업공학과 박사수료
현 에틱스아카데미 e-Learning 강사로 활동
상지대학교 시스템경영공학과 외래교수
관심분야: 신뢰성공학, 품질공학, TRIZ, 제품개발
주 소 : 경기도 수원시 장안구 천천동 300번지 성균관대학교 시스템경영공학과 26418B호실

김 인 희
남서울대학교 산업공학과를 졸업
현 성균관대학교 산업공학과 석사재학
관심분야: 신뢰성공학, 품질공학, 소프트웨어 품질관리, 리스크 경영공학, SPC
주 소 : 경기도 수원시 장안구 천천동 300번지 성균관대학교 시스템경영공학과 26418B호실