

산업용 내장형 소프트웨어를 위한 품질 시험 모듈의 개발

이선원¹, 이하용², 양해술^{3*}

Development of Quality Test Module for Industrial Embedded Software

Sun-Won Lee¹, Ha-Yong Lee², Hae-Sool Yang^{3*}

요 약 산업용 소프트웨어의 유형은 크게 패키지형, 내장형, 시스템형으로 구분할 수 있다. 패키지형 소프트웨어는 산업용 ERP를 의미하며, 내장형 소프트웨어는 산업용 기기에 내장되어 사용되고, 시스템형은 제조 및 생산 시스템의 전반적인 제어를 위한 소프트웨어이다. 본 논문에서는 산업용 내장형 소프트웨어를 대상으로 하여 품질시험 및 평가를 수행할 수 있는 평가모듈의 개발에 관해 기술하였다. 평가모듈의 객관성 및 타당성을 제고하기 위해서는 국제표준을 기반으로 한 체계 구축이 필수적이다. 관련된 국제표준으로는 소프트웨어 제품의 요구사항 및 평가에 관한 표준인 ISO/IEC 12119와 9126이 있으며, 평가모듈의 구성 형식을 규정하고 있는 ISO/IEC 14598-6이 있다. 이러한 표준들을 기반으로 하여 산업용 내장형 소프트웨어의 품질 요구사항을 추출하고 품질 요구의 만족 수준을 평가할 수 있는 평가모듈을 구축하였다.

키워드 : 산업용 소프트웨어, 평가모듈, 메트릭, 내장형 소프트웨어

Abstract The types of industrial software are classified into package, embedded and system type. The Package type software means industrial ERP, the embedded software is built in industrial devices and the system type software controls the manufacturing and production system. In this paper, we described about the development of evaluation modules which can implement quality test and evaluation for industrial embedded software. If you are planning to enhance the objectivity and the validity of evaluation modules, it is necessary to construct the system based on international standard. There are two international standard, ISO/IEC 12119 and ISO/IEC 9126 related to quality evaluation of software. Also ISO/IEC 14598-6 is the standard that defines the construction form of the evaluation modules. Based on these standards, We extracted the quality requirements of industrial embedded software and constructed the evaluation modules that can evaluate the content level of the quality requirements.

Key words : Industrial Software, Evaluation Module, Metric, Embedded Software

1. 서 론

오늘날 소프트웨어 산업은 다양한 분야에서 급격히 발전하고 있다. 특히 산업용 소프트웨어 분야는 높은 부가가치로 인해 관심의 대상이 되고 있다. 그러나 산업기술

시험원의 설문조사 결과에서는 국내 산업용 소프트웨어의 품질에 대한 견해로서 70% 정도가 보통 이하라는 의견을 보이고 있다[1].

다른 소프트웨어 분야와 마찬가지로 산업용 소프트웨어의 경우에도 몇 가지 유형으로 구분할 수 있다. 산업용 소프트웨어의 유형은 크게 패키지형, 내장형, 시스템형으로 구분할 수 있다. 산업용 패키지 소프트웨어는 산업용 ERP(Enterprise Resource Planning)를 의미하며, 산업용 내장형 소프트웨어는 산업용 기기에 내장되어 특정한 목적을 위해 사용되는 소프트웨어이며, 산업용 시스템형 소프트웨어는 제조 및 생산 시스템의 전반적인 제어를

본 연구는 정보통신부 지원 ITRC 프로그램의 지원을 받아 수행되었음.

¹ 서울벤처정보대학원대학교 경영학과(박사과정)

² 서울벤처정보대학원대학교 컴퓨터응용기술학과(전임강사)

³ 호서대학교 벤처전문대학원 컴퓨터응용기술학과(교수)

*교신저자: 양해술(hsyang@office.hoseo.ac.kr)

위해 사용되는 소프트웨어를 말한다[2].

일반 사무용 패키지 소프트웨어에 관해서는 품질평가 방법에 관한 연구를 통해 시험모듈의 구축 및 품질 측정 방법에 관한 연구가 진전되어 제도화의 기틀을 마련하였다[3, 4]. 그러나 산업용 내장형 소프트웨어 분야의 경우에는 시험방법에 대한 연구가 미흡하여 고부가가치를 창출할 수 있는 산업용 내장형 소프트웨어 분야의 품질수준을 확보하지 못하는 결과를 초래하고 있다.

본 논문에서는 산업용 내장형 소프트웨어를 대상으로 하여 품질시험 및 평가를 수행할 수 있는 평가모듈의 개발에 관해 기술하였다. 평가모듈의 객관성 및 타당성을 제고하기 위해서 본 논문에서는 소프트웨어 제품 평가에 관한 국제표준을 기반으로 하였다. 적용된 국제표준으로는 소프트웨어 제품의 요구사항 및 평가에 관한 표준인 ISO/IEC 12119[5]와 ISO/IEC 9126[6]이 있으며, 소프트웨어 품질평가 프로세스에 관한 국제표준인 ISO/IEC 14598[7] 중에서 소프트웨어 품질평가에 관한 제반 방법에 대해 문서화하는 것을 의미하는 평가모듈의 구성 형식을 규정하고 있는 ISO/IEC 14598-6이 있다. 이러한 표준들을 기반으로 하여 산업용 내장형 소프트웨어의 품질 요구사항을 추출하고 품질 요구의 만족 수준을 평가할 수 있는 평가모듈을 구축하였다. 본 논문의 제2장에서는 산업용 소프트웨어에 대해 소개하고 그 특성을 기술하였으며 제 3장에서는 산업용 내장형 소프트웨어의 품질에 관한 요구사항에 대해 분석하였으며 제4장에서는 산업용 내장형 소프트웨어의 시험을 위한 기준을 제시하고 제5장에서는 산업용 내장형 소프트웨어의 시험을 위한 방법을 문서화한 시험모델에 대해 기술하였다. 제6장에서는 본 논문에서 제시하는 평가방법과 타 평가방법과의 차이점 및 장단점을 분석하였으며 마지막으로 결론 및 향후 연구 과제를 제시하였다.

2. 산업용 소프트웨어 유형과 특성

이 절에서는 산업용 소프트웨어의 유형과 산업용 소프트웨어가 가지고 있는 특성에 대해 소개한다.

2.1 산업용 소프트웨어의 유형

(1) HMI(Human Machine Interface) 시스템

MMI(Man Machine Interface)라고 불리기도 하는 이 기술은 광범위하게는 인간의 모든 감각기관을 이용한 정보의 표출과 외부 정보에 대한 인지행위를 통해 기계와 통신하는 모든 인터페이스를 칭한다. HMI는 빠르게 변화

하는 산업 환경에 부합되는 시스템으로 PLC와 같은 연결된 외부 디바이스를 통해 끊임없이 기계와 통신하면서 기계를 제어하고, 그 상태를 파악하여 빠르게 보여주고 경고하는 등의 기능을 갖는다. 이러한 HMI 시스템의 유형은 [표 1]과 같이 크게 두 가지로 나누어진다.

표 1. HMI 시스템의 유형

| 유형 | 내용 |
|------------------|--|
| 터치패널 내장형 HMI 시스템 | <ul style="list-style-type: none"> 패널 형태로 구성되어 현장기기에 근접하여 설치되어 대부분 전용의 하드웨어 플랫폼에 HMI 소프트웨어가 내장 제어설비의 PLC 제어기와 직접 연결되어 운전자는 그래픽 터치 패널을 통하여 설비를 실시간으로 모니터링하고 제어할 수 있도록 인터페이스를 제공 전용 시스템을 사용하여 신속한 처리가 가능하고 높은 신뢰성을 가짐 |
| PC 기반의 HMI 시스템 | <ul style="list-style-type: none"> 기존의 전통적인 산업 제어 시스템들을 대신하여 등장 일반적인 컴퓨터 운영체제에서 실행되는 HMI 소프트웨어를 통해 산업현장의 제어기와 네트워크로 연결하여 각종 기능을 실현하는 통합 시스템 일반 운영체제에서 실행되는 다른 응용 소프트웨어와 통합이 용이하고 사용자의 요구사항에 따라 유연하게 응용하여 사용 가능 |

(2) SCADA(Supervisory Control and Data Acquisition) 시스템

SCADA 시스템은 원격지에 있는 설비를 운영자가 지정한 명령에 따라 설비를 제어할 수 있는 능력과 제어대상 기기가 지시한 대로 제어되고 있는지의 상태를 감시하는 설비이다. 원방감시 제어는 운영자가 여러 곳에 분산 설치되어 있는 원격 단말 장치들을 감시하고 필요에 의해 제어할 수 있도록 구축하는 시스템을 말한다.

SCADA 시스템은 Remote Terminal Units(RTU), 통신장치, Master Station(MS) 또는 Master Terminal Unit(MTU)로 구성되어 있다. MTU는 제어실에 위치하며 운영요원들에 의해 운영된다. 운영 요원들은 MTU에 의해 작동되어지는 CRT나 시각적인 표시기(Lamps, LED 또는 LCD)를 가진 Map Board 또는 Projection Type의 Wall Board를 사용하여 전체 시스템의 현재 상태를 감시한다.

운영요원들은 시스템 운영에 필요한 매개변수의 조절 또는 운영 중 발생된 문제를 해결하거나 시스템의 상태를 조작 또는 변경하기 위하여 시스템을 사용하며, 시스템의 제어는 일반적으로 키보드나 마우스 등 MTU의 입력 장비를 사용하여 이루어진다.

RTU들은 현장에 설치된 각종 기기들의 정보를 취득하기 위하여 원격지에 설치된다. MTU와 RTU의 관계는 Master and Slave와 유사하며 이들 RTU와 MTU 사이의 데이터 전송을 위한 통신을 구현하는데 광통신, 공중전화망 등을 이용하여 구현될 수 있으나 일반적으로 전용선을 이용한 모델들을 통하여 이루어진다.

(3) DCS(Distributed Control System)

초기의 컴퓨터를 이용한 제어 개념은 한 대의 컴퓨터에 의한 DDC(Direct Digital Control) 시스템과 현장 아날로그 연결에 의한 제어방식이 주종을 이루고 있었다. DDC 시스템이란 한 대의 컴퓨터에 프로세스 데이터의 입력, 출력 및 플랜트의 감시, 조작, 제어 등을 모두 집중화 시켜 관리하는 시스템이다. DDC의 경우 모든 제어기능이 한 대의 컴퓨터에 집중되어 있으므로 컴퓨터에 이상이 발생하면 공정 전체가 제어 불능 상태가 되어 신뢰성 저하가 문제가 되었다. DCS는 DDC의 단점을 보완하기 위하여 하나의 중앙처리장치를 여러 개의 작은 중앙처리장치로 나누어 기능별로 분리하고 작은 용량의 중앙처리장치를 갖는 각각의 컴퓨터를 통신네트워크로 연결시켜 전체 시스템을 구성하도록 되어 있다.

DCS의 기본 개념은 공정제어에 적용되는 시스템을 각 플랜트에 알맞은 단위 서브시스템으로 분리하고 각 소단위 시스템에서는 각각의 주어진 역할을 수행하며, 상호간에 통신이 가능하도록 한 것으로서 DCS는 소형 DDC 시스템 여러 개를 유기적으로 연결하여 전체 시스템을 구성한 것이라 볼 수 있다. DCS의 기본 특징은 프로세스 제어 기능을 여러 대의 컴퓨터에 분산시켜서 신뢰성을 향상시키고 이상발생시 그 파급 효과를 최소화시키며 프로세스 정보처리 및 운전 조작 그리고 분산 설치된 컴퓨터들의 관리 기능 등은 중앙의 주 컴퓨터(DOC : Distributed Operate Console)에 집중화시킴으로서 자료처리 및 운영관리를 원활하게 하는데 있다.

2.2 산업용 내장형 시스템의 특징

산업용 내장형 시스템은 한정된 목적을 가지며 실시간 처리를 요구하고 열악한 환경에서의 강한 내구성 등 그 특징을 일일이 언급하기는 어렵지만 몇 가지 특징을 요약하면 [표 2]와 같이 정리할 수 있다[8, 9, 10, 11].

표 2. 내장형 시스템의 특징

| 특징 | 내 용 | 비고 |
|--------|---|----|
| 목적의 한정 | <ul style="list-style-type: none"> • 범용 시스템은 일반적인 프로그램이 빠르게 실행될 수 있는 구조를 갖게 되며, 이렇게 설계된 후 고정됨 • 범용 프로세서의 성능은 운영체제 및 컴파일러 등 많은 구성 소프트웨어에 의해 평가 • 내장형 시스템은 빈번한 연산의 보다 빠른 수행을 위하여 때론 그 기능을 하드웨어로 구현해야 될 필요가 있음 • 소프트웨어와 하드웨어가 동시에 개발되면서, 어떤 기능이 하드웨어로 구현되어야 하는지의 문제를 먼저 고려 • 특정한 목적을 효과적으로 달성하기 위한 최소의 하드웨어와 최소의 소프트웨어를 이용한 시스템을 설계해야 한다는 문제를 내장형 시스템 설계자가 해결해야 함 | |
| 실시간 처리 | <ul style="list-style-type: none"> • 내장형 시스템은 과제의 처리 기한이 주어지는 실시간 처리가 많으며 실행 기한(Deadline)이 주어져 있어 그 실행 기한 안에 데이터가 처리되지 않으면, 동영상이나 인터넷 전화에서 정보 전달이 제대로 되지 않음 • 실시간 처리는 실행 기한 내에서만 처리된다면, 무조건 빨리 처리되는 것이 바람직하다고 할 수 없기 때문에, 처리해야 하는 모든 작업들의 우선순위를 효과적으로 조정하여 모든 작업이 각각의 제한 시간 안에 처리되도록 하는 것이 중요 | |
| 대량 생산 | <ul style="list-style-type: none"> • 내장형 시스템은 대량생산을 목적으로 최소 비용에 최대 효과를 내야하는 기기이며 최종 제품의 성능에 의해 평가됨 | |
| 강한 내구성 | <ul style="list-style-type: none"> • 많은 내장형 시스템은 고온이나 다습한 환경, 또는 충격이 가해지거나 일부 기능에 이상이 있어도 기본 기능은 계속 동작하도록 요구되는 경우가 많음 • 많은 내장형 시스템이 험난한 환경에서 동작함과 동시에 절대적으로 오작동을 허락하지 않는 미사일 제어 시스템이라든가 ABS 등의 아주 중요한 작업을 수행하는 경우가 빈번 | |

3. 산업용 내장형 소프트웨어의 품질 요구사항

산업용 내장형 소프트웨어의 품질에 관한 요구사항을 정립하기 위해서는 소프트웨어 제품평가를 위한 품질특성의 개념에서 출발하여 산업용 내장형 소프트웨어와 일반 사무용 소프트웨어와의 차이를 명확히 해야 한다. 산업용 내장형 소프트웨어의 요구사항을 도출하기 위해 본 논문에서는 패키지 소프트웨어의 품질 요구사항과 시험에 관한 표준인 ISO/IEC 12119의 품질특성 체계를 바탕으로 하여 산업용 내장형 소프트웨어의 요구사항을 먼저 추출하고 이들 요구사항을 품질특성 체계에 맞추어 분류정리한 후, 각 요구사항에 따른 메트릭을 구축하였다. 다음에 산업용 내장형 소프트웨어가 갖추어야 할 요구사항에 대해 주요 내용을 요약하였다. 기능성 측면에서 산업용 내장형 시스템은 하드웨어와 소프트웨어적인 기능에

(1) 기능성 측면

기능성은 소프트웨어가 특정 조건에서 사용될 때, 명시된 요구와 내재된 요구를 만족하는 기능을 제공하는 소프트웨어 제품의 능력이라고 할 수 있다.

표 3. 기능성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|----------------------|---|-----|
| 소프트웨어/하드웨어 기능에 대한 언급 | 산업용 내장형 시스템의 경우, 빈번하게 반복되면서 고성능을 요구하는 기능은 하드웨어로 구현하며 향후 기능의 업그레이드가 요구되는 기능은 소프트웨어로 구현된다. 향후, 업그레이드를 위해 기능에 대해 소프트웨어 또는 하드웨어적 기능인지를 명시할 필요가 있다. | 적합성 |
| 사용환경에 대한 명세 | 산업용 내장형 시스템은 거친 환경에서 반복적인 충격이 발생하는 환경에서 동작해야 하는 경우도 있으므로 그 사용 환경에 맞는 강한 내구성을 가질 필요가 있다. 따라서, 산업용 내장형 소프트웨어에 대해 어떤 환경 조건에서 정상적으로 동작할 수 있는가에 대해 명확히 명세함으로써 적절한 사용 환경을 조성하도록 하거나 사용 환경에 맞는 산업용 소프트웨어를 선택할 수 있도록 해야 한다. | 적합성 |
| 비인가자에 대한 접근통제 | 산업용 내장형 시스템은 결함 발생이나 운영 실수로 인한 가동 중지시 막대한 손해를 발생시킬 수 있으므로 비인가자의 시스템 접근을 철저히 통제해야 한다. | 보안성 |
| ... | ... | ... |

대해 명확히 구분하고 명세된 사용환경에서 정확히 동작하며 인가되지 않은 사용자의 접근을 차단할 수 있어야 한다. [표 3]에 기능성에 관한 요구사항의 예를 나타내었다[2].

(2) 신뢰성 측면

신뢰성이란 명세된 조건에서 사용될 때 성능 수준을 유지할 수 있는 소프트웨어 제품의 능력이라고 할 수 있다. 신뢰성 측면에서 산업용 내장형 소프트웨어는 제3자 입장에서 평가수행이 어려운 점을 고려하여 자체적인 테스트 결과를 제공하고 실질적인 활용상황에서 결함을 최소화할 수 있어야 한다. [표 4]에 신뢰성에 관한 요구사항의 예를 나타내었다[2].

표 4. 신뢰성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|-----------------------|---|--------|
| 테스트 수행 결과의 제공 | 산업용 내장형 소프트웨어의 경우, 제조 및 제어 등의 산업 현장에서 사용되는 특성을 고려하여 현장과 유사한 환경에서 테스트한 결과와 그 적정성 여부를 명세함으로써 사용자의 선택에 도움을 줄 수 있는 정보를 제공해야 한다. | 성숙성 |
| 실적용 환경에서의 결함 및 오류의 회피 | 산업용 내장형 소프트웨어는 제조 및 제어 등의 산업 현장에서 결함 및 오류 발생을 유발할 수 있는 테스트 케이스를 시험하여 결함 및 오류 발생이 적정 수준을 유지하고 있음이 확인되어야 한다. | 결함 허용성 |
| 신뢰성 수준에 관한 정보 제공 및 준수 | 산업용 내장형 소프트웨어의 신뢰성 품질은 제조되는 물품이나 제어되는 공정의 품질에 큰 영향을 미치게 되므로 제조 및 제어 등의 산업 현장에서 보증될 수 있는 신뢰성 수준에 관한 정보가 제공되어야 하며 제공되는 정보에 따라 신뢰성 수준을 준수하여야 한다. | 준수성 |
| ... | ... | ... |

(3) 사용성 측면

사용성이란 명시된 조건에서 사용될 경우, 사용자에 의해 이해되고 학습되고 사용되고 선택될 수 있는 소프트웨어 제품의 능력이라 할 수 있다. 사용성 측면에서 산업용 내장형 소프트웨어는 이해하기 쉬운 편리한 인터페이스와 가동중에 제공되는 메시지의 이해용이성, 산업 환경의 변화에 대한 탄력적인 대응 등이 가능해야 한다. [표 5]에 사용성에 관한 요구사항의 예를 나타내었다[2].

표 5. 사용성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|-----------------|---|--------|
| 인터페이스의 이해도 | 산업용 내장형 소프트웨어의 편리한 인터페이스는 사용 용이성을 향상시켜 생산성 향상의 결과로 나타나게 된다. 따라서 이해 용이한 인터페이스로 구성되어 있는 산업용 내장형 소프트웨어는 사용성 측면에서 고품질 확보를 위한 기본 특성이라 할 수 있다. | 이해 가능성 |
| 제공되는 메시지의 이해용이성 | 산업용 내장형 소프트웨어는 운영 과정에서 발생하는 다양한 상황에 대한 다수의 메시지를 간결한 방법으로 시각적으로 제공하므로 제공되는 메시지를 얼마나 직관적으로 쉽게 이해할 수 있는 형태로 제공하는가는 다양한 상황의 변화나 문제 발생에 신속하게 대처할 수 있는 가능성을 높이는 요인이 될 수 있다. | 운영성 |
| 운영절차 조정 가능성 | 산업용 내장형 시스템은 제조환경이나 제어환경의 다양한 변화에 대처하여 운영절차에 대해 다양한 변화를 줄 수 있도록 설계되어야 한다. | 운영성 |
| ... | ... | ... |

(4) 효율성 측면

효율성이란 명시된 조건에서 사용되는 자원의 양에 따라 요구된 성능을 제공하는 소프트웨어 제품의 능력이라 할 수 있다. 효율성 측면에서 산업용 내장형 소프트웨어는 실시간 특성을 고려하여 처리기한 명세 및 높은 처리 성능 등이 주로 요구된다고 할 수 있다. [표 6]에 효율성에 관한 요구사항의 예를 나타내었다[2].

표 6. 효율성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|---------------|--|--------|
| 처리기한의 명세 및 준수 | 산업용 내장형 소프트웨어는 제조 및 제어 공정에서 동작하면서 다른 공정과의 타이밍을 적절히 조절하여 생산이나 제어공정에 차질을 주지 않도록 성능상에 문제가 발생하지 않아야 한다. 따라서 내장형 소프트웨어의 기능에 대해 처리기한이 명시되어 있어야 하며 명시한 처리기한을 준수해야 한다. | 시간 효율성 |
| 높은 처리율 | 산업용 내장형 시스템은 고성능이 요구되는 핵심 기능에 대한 하드웨어적인 구성을 통해 성능요구를 만족할 수 있도록 구성되어야 한다. | 시간 효율성 |
| 데이터 전송 성능 | 산업용 내장형 소프트웨어는 고성능의 데이터 전송 성능을 통해 다른 기기와의 원활한 동작이 가능해야 한다. | 자원 효율성 |
| ... | ... | ... |

(5) 유지보수성 측면

유지보수성이란 소프트웨어 제품이 변경되는 능력이라 할 수 있다. 유지보수성 측면에서 산업용 내장형 소프트웨어는 가동중 결함으로 인해 중단되었을 때 결함 내용을 신속하게 파악하여 복구할 수 있어야 하며 동작상황을 모니터링할 수 있는 정보의 저장 기능이 요구된다. [표 7]에 유지보수성에 관한 요구사항의 예를 나타내었다[2].

표 7. 유지보수성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|-----------------|---|-----|
| 진단 기능 | 산업용 내장형 소프트웨어의 경우 운영 과정에서 결함이 발견되었을 경우, 신속하게 결함의 원인을 찾아내고 문제를 해결하여 재가동할 수 있어야 한다. 따라서 자체적으로 내장된 진단기능을 제공하여 신속하고 정확한 결함의 진단과 해결방법이 제시되어야 한다. | 분석성 |
| 운영 중 활동 기록 및 제시 | 산업용 내장형 소프트웨어 사용 중에 진행되는 동작이 무엇인지 명확히 제시하고 있으며 정상적인 동작에 문제가 발생한 경우, 발생된 문제에 대한 가이드를 제시해야 한다. | 분석성 |
| ... | ... | ... |

(6) 이식성 측면

이식성이란 한 환경에서 다른 환경으로 전이될 수 있는 소프트웨어 제품의 능력이라 할 수 있다. 이식성 측면에서 산업용 내장형 소프트웨어는 다양한 환경에 대응하여 용이한 이식 및 설치가 가능해야 한다. [표 8]에 이식성에 관한 요구사항의 예를 나타내었다[2].

표 8. 이식성에 관한 요구사항

| 요구사항 | 내용 | 비고 |
|--------|---|--------|
| 이식 용이성 | 산업용 내장형 소프트웨어가 유사한 유형의 서로 다른 기기에 적용이 가능하다고 할 때 별다른 노력 없이 쉽게 이식이 가능해야 한다. | 적용성 |
| 설치 용이성 | 산업 내장형 소프트웨어는 해당 산업용 기기에 쉽게 설치가 가능해야 한다. 산업용 기기를 다루는 사용자는 일반적으로 소프트웨어 분야의 관련 전문 지식을 가지고 있지 못한 경우가 많을 것이므로 설치가 용이하면 재설치가 필요할 때 산업용 기기의 유희시간을 최소화할 수 있는 장점이 있다. | 설치 가능성 |
| ... | ... | ... |

4. 산업용 내장형 소프트웨어의 시험 기준

이 절에서는 품질특성의 관점에서 산업용 내장형 소프트웨어의 품질시험에 적용할 수 있는 시험기준과 방법에 대해 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성 등 소프트웨어 제품평가에 관한 표준인 ISO/IEC 9126을 기준으로 제시하였다.

4.1 기능성에 관한 시험기준

기능성은 소프트웨어가 특정 조건에서 사용될 때, 명시된 요구와 내재된 요구를 만족하는 기능을 제공하는 소프트웨어 제품의 능력이라고 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 기능정보제공, 기능구현완전성, 기능충분성, 데이터교환성 등을 들 수 있으므로 [표 9]와 같은 시험기준을 작성하였다[2].

표 9. 기능성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|-----|-----------|--|---|
| 1 | 기능 정보 제공 | 산업용 내장형 소프트웨어에서 제공하는 모든 기능에 대한 설명을 제공하고 있어야 한다. | 산업용 내장형 소프트웨어에서 제공하는 모든 기능의 수를 파악하고 그 중에서 문서에 언급된 기능수를 측정 |
| 2 | 기능 구현 완전성 | 문서에 기술된 기능이 사용가능하도록 실제 소프트웨어 및 기기에 구현되어 있어야 한다. | 문서에 기술된 모든 기능을 식별하고 소프트웨어 및 기기에 현되어 있는가 확인 |
| 3 | 기능 충분성 | 산업용 내장형 소프트웨어가 그 목적에 맞는 필수적인 기능들을 구현하고 있어야 한다. | 산업용 내장형 소프트웨어 유형에 따른 필수기능점검표를 만들고 소프트웨어에서 필수기능항목을 식별 |
| 4 | 데이터 교환성 | 산업용 내장형 소프트웨어가 데이터 교환이 요구되는 다른 소프트웨어나 산업기기와의 데이터 교환이 명세된 대로 가능해야 한다. | 산업용 내장형 소프트웨어의 기능 명세에 따른 데이터 교환 기능의 목록을 작성하여 테스트 데이터를 생성하고 교환 가능 수행 |
| ... | ... | ... | ... |

4.2 신뢰성에 관한 시험기준

신뢰성이란 명세된 조건에서 사용될 때 성능 수준을 유지할 수 있는 소프트웨어 제품의 능력이라고 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 사용환경 시험여부, 결함회피율, 다운회피율 등을 들 수 있으므로 [표 10]과 같은 시험기준을 작성하였다[2].

표 10. 신뢰성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|-----|-------------|--|--|
| 1 | 사용 환경 시험 여부 | 산업용 소프트웨어 유형에 따라 적절한 사용 환경에서 테스트를 수행한 결과를 제공해야 한다. | 실제 사용환경에서 테스트를 성공적으로 수행한 결과를 제공하는가를 확인 |
| 2 | 결함 회피율 | 산업용 내장형 소프트웨어의 사용 중 발생하는 결함의 수가 규정된 값을 넘지 않아야 하며 결함이 발생한 경우, 결함 발생 사실을 알려줄 수 있어야 한다. | 허용 가능한 결함수를 규정하고 일정시간 시스템을 운영하여 허용된 결함수 이내로 결함이 나타나는가 확인 |
| 3 | 다운 회피율 | 산업용 내장형 소프트웨어의 결함으로 인해 시스템이 다운(정지)되는 경우가 발생하지 않아야 한다. | 산업용 소프트웨어를 운영하여 결함발생 상황을 식별하고 다운되지 않는가를 확인 |
| ... | ... | ... | ... |

4.3 사용성에 관한 시험기준

사용성이란 명시된 조건에서 사용될 경우, 사용자에게 의해 이해되고 학습되고 사용되고 선호될 수 있는 소프트웨어 제품의 능력이라 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 기능이해도, 입출력데이터이해도, 오류 복구용이성 등을 들 수 있으므로 [표 11]과 같은 시험기준을 작성하였다[2].

표 11. 사용성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|-----|-------------|--|---|
| 1 | 기능 이해도 | 산업용 내장형 소프트웨어나 시스템의 사용자 수준에서 제품설명서와 사용자 문서를 통해 제공하는 기능을 충분히 이해할 수 있도록 설명되어 있어야 한다. | 사용자 집단을 구성하고 테스트케이스를 작성하여 이해 정도를 측정 |
| 2 | 입출력 데이터 이해도 | 산업용 내장형 소프트웨어가 사용자 수준에서 입출력 화면 및 데이터를 쉽게 이해할 수 있도록 작성되어 있어야 한다. | 사용자 집단을 구성하고 입출력 화면 및 데이터에 대한 이해 정도를 측정 |
| 3 | 오류 복구 용이성 | 산업용 내장형 소프트웨어의 오류는 기기의 리콜을 유발할 수도 있으므로 쉽게 복구될 수 있어야 한다. | 오류목록을 작성하여 복구를 시도하고 정해진 시간 내에 복구된 오류의 수를 측정 |
| ... | ... | ... | ... |

4.4 효율성에 관한 시험기준

효율성이란 명시된 조건에서 사용되는 자원의 양에 따라 요구된 성능을 제공하는 소프트웨어 제품의 능력이라 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 데이터전송률, 평균처리율, 메모리사용률 등을 들 수 있으므로 [표 12]와 같은 시험기준을 작성하였다[2].

표 12. 효율성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|-----|---------|--|--|
| 1 | 데이터 전송률 | 영상 데이터 같은 대용량의 데이터를 충분히 빠르게 전송할 수 있어야 한다. | 산업용 내장형 소프트웨어가 전송하는 특정 영상 데이터에 대한 전송 속도를 규정하고 규정에 따르는 가를 확인 |
| 2 | 평균 처리율 | 산업용 내장형 소프트웨어가 처리 평균량의 한계값에 대해 주어진 시간 내에 성공적으로 작업을 수행할 수 있는 평균처리량이 한계값에 접근해야 한다. | 규정된 처리평균량의 한계값에 대한 테스트 케이스를 통해 계산된 처리평균량의 비율을 이용하여 산출 |
| 3 | 메모리 사용률 | 동일 유형의 산업용 내장형 시스템에 장착되는 평균 메모리량을 초과하지 않아야 한다. | 동일 유형의 산업용 내장형 시스템에 장착되는 메모리의 평균량에 대한 평가대상 시스템에 설치되어 있는 메모리의 양을 측정 |
| ... | ... | ... | ... |

4.5 유지보수성에 관한 시험기준

유지보수성이란 소프트웨어 제품이 변경되는 능력이라 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 진단기능정보제공, 진단기능지원률, 변경가능률 등을 들 수 있으므로 [표 13]과 같은 시험기준을 작성하였다[2].

표 13. 유지보수성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|----|-------------|---|--|
| 1 | 진단 기능 정보 제공 | 산업용 내장형 소프트웨어를 사용할 때, 발생하는 오류의 증상 및 해결할 수 있는 진단기능에 관한 정보를 제공해야 한다. | 문서에서 산업용 내장형 시스템의 진단 기능에 대해 기술하고 있는가를 확인 |
| 2 | 진단 기능 지원률 | 문서에 기술된 진단 기능정보에 따라 산업용 내장형 시스템의 정상 작동 여부를 진단할 수 있는 진단기능이 제공되어야 한다. | 문서에 명세된 진단 기능이 실제 구현되어 있는가를 확인 |

| | | | |
|-----|--------|--|---|
| 3 | 변경 가능성 | 산업용 내장형 시스템이 문서상의 환경 설정 변경에 관한 정보대로 구현되어 있어야 한다. | 산업용 내장형 시스템에서 환경설정 변경에 관한 항목에 따라 테스트케이스를 생성하여 설정 변경이 성공한 경우를 측정 |
| ... | ... | ... | ... |

4.6 이식성에 관한 시험기준

이식성이란 한 환경에서 다른 환경으로 전이될 수 있는 소프트웨어 제품의 능력이라 할 수 있다. 따라서 이를 시험하기 위한 특성으로는 이식편리성, 설치가능률, 이식 표준준수정보제공 등을 들 수 있으므로 [표 14]와 같은 시험기준을 작성하였다[2].

표 14. 이식성에 관한 산업용 내장형 S/W 시험기준

| 번호 | 시험할 특성 | 시험기준 | 시험방법 |
|-----|------------|--|---|
| 1 | 이식 편리성 | 사용자가 내장형 소프트웨어를 시스템 환경에 쉽게 적용시킬 수 있도록 구현되어 있어야 한다. | 사용자가 소프트웨어의 셋업을 변경하여 사용 환경에 쉽게 적용시킬 수 있는지 확인 |
| 2 | 설치 가능성 | 사용자 설치가 가능한 경우, 설치 정보에 따라 시스템에 내장형 소프트웨어의 설치가 가능해야 한다. | 설치 시도 횟수에 따른 성공 횟수를 측정 |
| 3 | 이식표준준수정보제공 | 산업용 내장형 소프트웨어의 호환성을 위한 표준이나 규정을 준수하도록 구현되어 있어야 한다. | 산업용 내장형 소프트웨어가 준수해야 할 호환성 관련 표준에 대한 체크리스트를 구성하여 준수하고 있는 항목의 수를 체크 |
| ... | ... | ... | ... |

5. 산업용 내장형 소프트웨어 시험모듈

본 연구를 통해 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성에 대한 부특성 27개에 대해 [표 15]와 같이 산업용 내장형 소프트웨어의 시험모듈을 개발하였다.

표 15. 개발된 메트릭의 구성

| 특성 | 부특성 | 메트릭 | 갯수 |
|-------|-------|--|----|
| 기능성 | 적합성 | 기능정보제공, 데이터정보제공, 사용환경명세제공, 기능구현완전성, 기능충분성, 경제 값정보제공, 경제값처리율 | 7 |
| | 정확성 | 기능분류명확성, 기능구현정확성정보제공, 기능구현정확성 | 3 |
| | 상호연용성 | 연결가능성, 데이터교환정보제공수준, 데이터교환성 | 3 |
| | 보안성 | 접근통제정보제공, 접근통제가능성 | 2 |
| | 준수성 | 기능표준준수정보제공, 기능표준준수율 | 2 |
| 소계 | | | 17 |
| 신뢰성 | 성숙성 | 문제해결이력정보제공, 문제해결률, 사용환경시험여부, 사용환경시험여부의 적절성, 결함회피율 | 5 |
| | 결함허용성 | 다운 회피율, 고장 회피율, 오조작 회피율 | 3 |
| | 회복성 | 데이터회복정보제공, 데이터회복률, 복구가능률 | 3 |
| | 준수성 | 신뢰성수준정보제공, 신뢰성수준준수율 | 2 |
| 소계 | | | 13 |
| 사용성 | 이해가능성 | 예비지식정보제공, 기능이해도, 인터페이스이해도, 입출력데이터이해도, 내용일관성, 사용자안내성 | 6 |
| | 학습성 | 학습용이성, 도움말접근용이성 | 2 |
| | 운영성 | 오류복구용이성, 메시지에히용이성, 운영절차조정정보제공, 진행상태 파악가능성 | 4 |
| | 선호도 | 인터페이스조정가능성, 인터페이스선호도 | 2 |
| | 준수성 | 사용성표준준수정보제공, 사용성표준준수율 | 2 |
| 소계 | | | 16 |
| 효율성 | 시간효율성 | 처리기한명세, 평균처리율, 평균처리시간 | 3 |
| | 자원효율성 | 메모리사용률, 데이터전송률 | 2 |
| | 준수성 | 효율성표준준수정보제공, 효율성표준준수율 | 2 |
| 소계 | | | 7 |
| 유지보수성 | 분석성 | 진단기능정보제공, 진단기능지원률, 진단구분률, 문제해결정보제공, 문제해결구현률 | 5 |
| | 변경성 | 환경설정변경정보제공, 변경가능률 | 2 |
| | 안정성 | 환경설정변경안전성정보제공, 환경설정변경안전성 | 2 |
| | 시험가능성 | 내장형시험기능정보제공, 내장형시험기능구현율 | 2 |
| | 준수성 | 유지보수표준준수정보제공, 유지보수표준준수율 | 2 |
| 소계 | | | 13 |
| 이식성 | 적용성 | 이식편리성 | 1 |
| | 설치가능성 | 설치정보제공, 설치가능률, 설치용이성 | 3 |
| | 대체성 | 데이터지속정보제공, 데이터지속가능률 | 2 |
| | 공존성 | 공존가능정보제공, 공존가능률 | 2 |
| | 준수성 | 이식표준준수정보제공, 이식표준준수율 | 2 |
| 소계 | | | 10 |
| 총계 | | | 76 |

5.1 품질모델의 구성

산업용 내장형 소프트웨어의 구성요소에는 패키지 소프트웨어가 만족시켜야 할 권고나 요구사항 또는 규제를 포함하고 있는 요구사항 문서와 소프트웨어 속성을 설명하는 문서로서 잠재적인 구매자가 제품 구입에 앞서 스스로 그 제품의 적합성을 평가할 수 있도록 하는 제품설명서, 제품의 사용을 위해 제공되며 제품 사용에 필요한 모든 정보가 포함되는 사용자 문서, 한 가지 이상의 매체를 통해 제공되는 실행 프로그램과 관련 데이터가 있다.

ISO/IEC 12119를 산업용 소프트웨어의 시험에 적용하기 위해서는 산업용 소프트웨어를 구성하는 각 항목들에 대해 적용할 품질 모델을 구성해야 한다. 본 논문에서

는 ISO/IEC 9126과 ISO/IEC 12119의 품질특성을 기반으로 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성의 6가지 품질특성을 기반으로 하였다.

5.2 시험 모듈의 체계와 개발 내역

시험모듈의 구성에 대해 소프트웨어 제품평가 프로세스에 관한 국제표준인 ISO/IEC 14598의 부분 6에서 형식을 제시하고 있으며 이러한 형식을 기반으로 본 연구에서 구축한 산업용 내장형 소프트웨어 품질시험 모듈의 전반적인 체계는 [그림 1]과 같이 모듈의 전반적인 개요, 적용에 관한 사항, 참조 문서, 메트릭, 적용 절차, 시험 결과에 대한 해석 등으로 구성되어 있다.

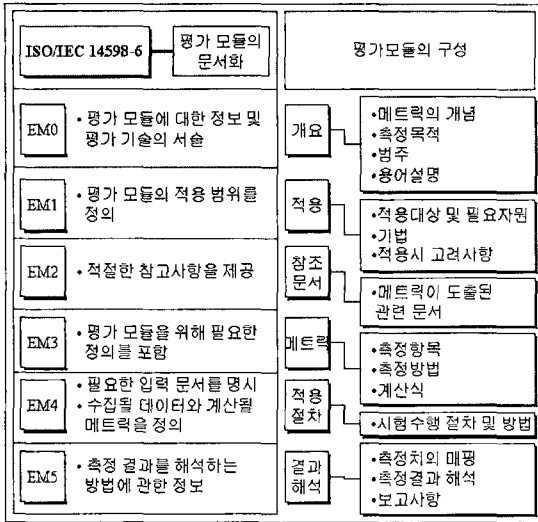


그림 1. ISO/IEC 14598-6과 평가모듈의 구성

본 연구를 통해 [표 16]과 같이 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성에 대한 부특성 27개에 대해 총 76개의 메트릭을 개발하였다.

표 16. 시험모듈 개발 내역

| 특성 | 부특성 결과 | | | | | 계 |
|-------|--------|-----|--------|-----|-----|----|
| | 적합성 | 정확성 | 상호 운영성 | 보안성 | 준수성 | |
| 기능성 | 7 | 3 | 3 | 2 | 2 | 17 |
| | 5 | 3 | 3 | 2 | | |
| 신뢰성 | 5 | 3 | 3 | 2 | | 13 |
| | 6 | 2 | 4 | 2 | 2 | |
| 사용성 | 6 | 2 | 4 | 2 | 2 | 16 |
| | 3 | 2 | 2 | | | |
| 효율성 | 3 | 2 | 2 | | | 7 |
| | 5 | 2 | 2 | 2 | 2 | |
| 유지보수성 | 5 | 2 | 2 | 2 | 2 | 13 |
| | 1 | 3 | 2 | 2 | 2 | |
| 이식성 | 1 | 3 | 2 | 2 | 2 | 10 |
| | | | | | | |
| 계 | | | | | | 76 |

5.4 품질검사표

품질검사표는 품질 시험을 수행하는 과정에서 편리하

게 참조할 수 있도록 필요한 사항들 추출하여 요약한 표이다. 이러한 품질검사표의 예를 [표 17]에 나타내었다.

표 17. 품질검사표의 예

| | | |
|---------|--|---|
| 메트릭명 | 산업용 내장형 시스템이 제한 받고 있는 경계값의 범위를 벗어난 입력에 대한 예외처리를 하고 있습니까? | |
| 경계값 처리율 | 경계값 확인 대상 항목 수 | |
| 측정 항목 | A | - 사용자 문서에 명시된 경계값에 대한 테스트 케이스를 작성 |
| | B | - 각 항목별 테스트케이스 성공률의 합 - 테스트케이스에 따라 테스트를 수행하여 명세와 일치하는 경우를 체크 |
| 계산식 | $- \text{경계값 처리율} = B/A$ $- B = \sum_{i=1}^n \frac{\text{Success_TC}_i}{\text{Total_TC}_i}$ <ul style="list-style-type: none"> - Success_TC : i 번째 경계값 처리 기능 확인을 위해 수행한 테스트케이스 중 성공한 건 수 - Total_TC : i 번째 경계값 처리 기능 확인을 위해 수행한 테스트케이스 수 | |
| 결과 영역 | $0 \leq \text{경계값 처리율} \leq 1$ | 결과값 |
| 문제점 | | |

품질검사표에는 기본적으로 메트릭명과 메트릭이 측정하고자 하는 내용에 대한 문장이 포함되어 있다. 측정 항목은 계산식을 통해 메트릭을 구성하는 요소로 1개 또는 그 이상의 개수로 구성되며 항목 개요와 측정 방법에 대한 기술을 포함한다.

결과 영역은 계산식에 의해 산출되는 값이 나타날 수 있는 영역으로 메트릭 중 대부분이 0과 1사이의 값으로 사상되나 명확한 영역을 규정할 수 없는 경우도 있는데, 이러한 경우에는 별도의 사상 영역에 따른 판정 기준을 설정하게 된다.

5.5 점검표

점검표는 품질검사표를 이용하여 측정항목에 대한 측정을 수행하기 위해 작성된 테스트 케이스의 시험 목록이다. 예를 들어 [표 17]의 품질검사표에 있는 “경계값 처리율” 메트릭에 대한 측정항목의 점검표는 [표 18]과 같이 작성될 수 있다.

표 18. 점검표의 예

| 순번 | 경계값 | 경계값 정보제공 (Y) | 설명 | 경계값 처리율 | 비고 |
|-------|-------------------|--------------|----------------------------------|---------|----|
| 1 | Log file 최대 보관 일수 | Y | Log file 최대 보관일수 설정 가능 | Y | |
| 2 | 연결시도 최대 횟수 | Y | 네트워크 연결을 위해 시도하는 최대 시간을 설정할 수 있음 | Y | |
| ... | ... | ... | ... | ... | |
| Y의 갯수 | | | | | |
| N의 갯수 | | | | | |
| 결 과 | | | Y/(Y+N) | | |

5.6 시험결과서

점검표의 테스트 케이스를 사용하여 품질검사표에 대한 측정이 수행되면 각 측정값들을 산출할 수 있으며 메트릭으로 정의된 계산식을 이용하여 측정값으로부터 결과를 도출하면 메트릭별 측정 결과가 산출될 수 있다. 이 측정 결과를 품질특성, 부특성, 메트릭의 분류체계에 따라 정리하면 [표 19]와 같은 시험결과서로 정리된다.

표 19. 시험결과서의 예

| 제품설명서 및 사용자 문서 | | | | |
|----------------|---------|-----------|-------|------|
| 품질특성 | 부특성 | 메트릭 | 결과유형 | 측정값 |
| 1. 기능성 | 1.1 적합성 | 기능정보제공 | Y/N | 0.80 |
| | | 기능구현 완전성 | Scale | 0.95 |
| | | 경계값 정보 제공 | Y/N | 0.75 |
| | | 경계값 처리율 | Scale | 0.78 |
| | 1.2 정확성 | 기능구현 정확성 | Y/N | 0.63 |
| | 정보제공 | | | |
| | 데이터 교환성 | Scale | 0.86 | |
| ... | ... | ... | ... | ... |
| 6. 이식성 | 적용성 | 이식편리성 | Scale | 1.00 |
| | ... | ... | ... | ... |

6. 평가 방법 비교 분석

이 절에서는 본 논문에서 제시한 평가 방법과 타 평가 방법과의 비교를 통해 본 평가 방법의 장단점을 기술하였다.

소프트웨어의 품질평가에 관한 연구 분야는 그 기준이나 평가 결과에 대한 타당성을 검증하기가 매우 곤란한 측면이 있다. 본 연구에서는 연구 결과에 대해 최대한의 객관성과 타당성을 확보하기 위해 국제적인 합의 또는 권고라고 할 수 있는 패키지 소프트웨어의 품질시험 및 요구사항에 관한 표준을 기반으로 관련 평가모듈과 평가 프

로세스를 구축하였다.

품질평가 방법과 평가 결과에 대한 타당성을 검증하기 위해서는 다수의 소프트웨어를 대상으로 한 평가 결과의 축적을 통해 합리적인 평가 기준의 도출과 부적절한 평가 모듈의 개선 및 보완, 평가자(또는 기관)와 평가 대상 제품을 개발하여 평가를 의뢰하는 개발자(또는 기관)의 합의가 요구된다고 할 수 있다.

따라서, 이 절에서는 소프트웨어의 품질평가 방법에 관련된 다양한 형태들을 살펴보고, 본 논문에서 제시한 평가 방법의 장단점을 분석해 보도록 하겠다.

6.1 평가 방법의 형태

(1) 소스 코드의 복잡도 평가

McCabe의 복잡도 척도를 기반으로 하는 툴이나 C언어 소스 코드에 대한 품질평가 툴[12]은 소스코드를 분석하여 다양한 형태의 결과를 제시한다. 사용자는 이러한 툴을 통해 소스 코드의 구조와 흐름, 복잡도를 파악하고 소프트웨어의 품질검증 및 테스트에 대한 체계적이고 정량적인 방법론을 제공한다.

(2) 소스 코드의 코딩 룰 체크

(주)Genesis의 PQQA QAC/C++/J는 파싱 기반의 코딩 룰 체크 도구이다. 대부분의 유틸리티 성격의 코딩 룰 체크 도구와는 달리 파싱 기반으로 다양한 코딩룰의 정의 및 적용이 가능하다. PQQA QAC/C++/J는 도구 내에 1,000개 이상의 코딩 룰이 레벨별로 정의되어 있으므로 선별적으로 적용이 가능하고 조직과 프로젝트에 맞는 룰셋을 개발할 수 있다. 아울러 코드의 구조분석 및 메트릭을 제공하여 중요하고 복잡한 코드에 관심과 노력을 집중할 수 있다.

(3) 체크리스트에 의한 품질 검토

체크리스트에 의한 품질 검토 방법은 ISO/IEC 9126-3의 내부 메트릭을 기반으로 하여 요구분석, 설계, 구현 단계 등, 생명주기 각 단계별로 개발된 체크리스트를 사용하여 각 단계가 마무리된 후, 다음 단계로 진행해도 좋은지를 검토하는 방법이다. 각 단계별로 개발된 산출물을 대상으로 충실도를 검토하여 결정함으로써 최종 소프트웨어 제품에 대해 사용자의 요구사항이 단계적으로 충실히 반영될 수 있도록 하는 방법이다.

6.2 평가 방법의 비교

본 논문을 통해 제시한 품질평가 방법의 장단점을 정리하면 [표 20]과 같다.

표 20. 품질평가 방법의 장단점 비교

| 평가 방법 | 장점 | 단점 |
|-----------------|--|---|
| 소스 코드 복잡도 평가 도구 | 평가를 통해 복잡도를 개선함으로써 소프트웨어의 유지보수를 용이하게 하기 때문에, 지속적인 버전업을 요구하는 소프트웨어의 경우 유용하게 활용 가능 | 유지보수성은 소스 프로그램 자체에만 관련되는 것은 아니며 S/W 개발 생명주기 전 과정에 걸친 산출물이 S/W의 유지보수에 관련된다고 할 수 있다. 또한, 복잡도 평가 도구의 경우, 유지보수자가 바뀌는 경우에는 효과가 크지 않음 |
| 코딩을 체크 도구 | 조직과 프로젝트에 맞는 코딩 룰 셋을 개발할 수 있다는 점에서 개발 목적에 최적화된 소스의 개발이 가능 | 소스 코드에 국한 |
| 체크리스트에 의한 품질 검토 | 생명주기 전 단계에 걸친 소프트웨어 개발 산출물에 대한 품질 수준을 체크하여 각 단계별로 최상의 수준에서 다음 단계로 진행할 수 있기 때문에 생명주기 전반에 걸쳐 품질 수준을 확보하기 위한 활동을 수행 | 체크리스트를 통해 도출된 개발산출물의 품질이 최종 소프트웨어 제품의 품질을 정확히 대변하지 못하며 추정값에 불과함 |
| 품질평가 모듈 | 최종 S/W 제품을 대상으로 하고 있으므로 S/W 제품의 실질적인 품질값을 제시하며 개발된 최종 S/W 제품의 제품설명서, 사용자 문서, 프로그램과 데이터에 대해 ISO/IEC 9126-2의 외부메트릭과 ISO/IEC 12119의 패키지 S/W 시험 요구사항을 기반으로 하여 산업용 내장형 S/W의 품질에 대한 평가 근거 확보 | 최종 제품의 평가에 한정하여 적용 |

본 논문에서는 평가자 관점에서 활용할 수 있는 소프트웨어 품질평가 방법을 개발하는 것을 목적으로 하였기 때문에, 전문적인 지식이나 평가 방법이 요구되는 복잡도 평가나 코딩 룰 체크 방법은 연구 목적과 차이가 있으며, ISO/IEC 9126-3의 내부 메트릭을 기반으로 하는 체크리스트 사용은 최종 소프트웨어 제품보다는 소프트웨어 개발 기관 자체에서 개발 과정 전단계에 걸쳐 활용할 수 있는 방법이므로 본 논문의 평가 방법과는 차이가 있다고 할 수 있다.

소스 코드에 대한 평가 도구의 경우에는 지원하지 않는 프로그래밍 언어를 사용한 소스의 경우에는 적용이 불가능한 단점이 있으며, 체크리스트를 활용하는 경우에는 다양한 경우에 탄력적으로 적용이 가능하다는 장점이 있으며 최종 소프트웨어 제품에 대해 품질평가 모듈을 적용하는 경우에도 마찬가지이다.

산업용 내장형 소프트웨어의 품질평가를 위해 ISO/IEC 9126과 ISO/IEC 12119를 활용한 평가 방법은 향후, 산업용 내장형 소프트웨어의 다양한 유형에 응용할 수 있을 뿐만 아니라, 다른 유형의 소프트웨어에 대해서도 기본 모델로 하여 변환함으로써 적용이 가능하다.

7. 결론

소프트웨어 산업의 발전으로 사용자는 다양한 유형의 소프트웨어들 중에서 자신이 컴퓨터를 사용하는 목적과 용도에 알맞은 소프트웨어를 선택할 수 있게 되었으며 이로 인해 올바른 선택 방법에 대한 중요성이 대두되고 있다.

특히, 최근 급격히 발전하고 있으며 높은 부가가치를

창출할 수 있지만 많은 문제점과 오류들이 노출되고 있는 산업용 소프트웨어에 대한 인증 필요성이 증대되고 있다. 산업용 소프트웨어는 갈수록 복잡해지고 복합적인 다양한 기능을 요구하게 됨에 따라 산업용 소프트웨어의 품질 평가에 대한 중요성이 높아지고 있다. 따라서 국내 실정에 적합한 산업용 소프트웨어 제품 시험 체계의 구축이 매우 시급한 문제가 되고 있다.

국내에서는 소프트웨어 제품 인증에 대한 관련 기반이 아직 미흡하며 국가적인 시험인증 체계의 구축이 완전하지 못한 상태이다. 특히, 정보통신 분야의 시험인증 관련 법령 및 규정이 미흡하며 최근에 시험인증 체계의 확장을 위한 연구가 진행되고 있는 실정이다. 정부차원의 시험인증 기관은 어느 정도 정착되어 있으나 민간 차원의 기관은 아직 없는 실정이며 소프트웨어 제품에 대한 인증을 국외의 기관을 통해 인증 받는 사례도 있는 실정이다.

오늘날 몇몇 소프트웨어 개발 선진국이 개인용 컴퓨터에서 사용하는 패키지 소프트웨어의 대부분을 공급하고 있으며 개발 기술에서 우위를 점하고 있다. 그러나 산업용 소프트웨어의 경우에는 특정 기술적 우위를 점하고 있는 기업 및 국가가 독점할 수 없는 분야이며 다양한 산업용 소프트웨어가 공존할 가능성이 크다. 이러한 현실에서 산업용 소프트웨어 산업에 대한 중요성의 증대와 점유율 면에서의 우위를 확보하기 위한 품질 확보의 필요성이 요구됨에 따라 산업용 소프트웨어에 대한 품질 시험 체계 구축과 측정 기준에 대한 연구 수행이 요구된다.

이 논문에서는 소프트웨어 제품 품질평가에 관한 국제 표준을 기반으로 하여 산업용 내장형 소프트웨어의 품질 요구사항을 추출하고 품질 요구의 만족 수준을 평가할 수 있는 평가모듈을 구축하였다.

산업용 내장형 소프트웨어에 대한 품질평가 방법에 관한 평가기준을 개발하고 가이드라인을 구축하여 향후 실질적인 활용을 통해 고품질 산업용 내장형 소프트웨어의 개발을 촉진하고 아울러 상대적으로 낙후되어 있는 산업용 내장형 소프트웨어 산업에 대한 국제 경쟁력을 향상시킬 수 있을 것으로 기대한다.

참고문헌

- [1] 산업기술시험원, "산업용 S/W 품질평가 관련 설문 조사 결과", <http://swtest.ktl.re.kr>, 2006. 1.
- [2] 양해술, "산업용 소프트웨어 유형별 품질평가 모듈의 개발", 산업기술시험원 2차년도 중간보고서, 2006. 3.
- [3] 양해술, "소프트웨어 품질 측정 기록 및 지원툴킷 개발", ETRI 컴퓨터·소프트웨어 기술연구소 위탁연구과제, 1차년도최종보고서, 1997. 11.
- [4] 양해술, "소프트웨어 시험평가 모듈 개선 연구", ETRI 컴퓨터·소프트웨어 기술연구소 위탁과제, 최종보고서, 2001. 11.
- [5] ISO/IEC 12119, "Information Technology - Software Package - Quality requirement and testing".
- [6] ISO/IEC 9126, "Information Technolog - Software Quality Characteris- tics and metrics - Part 1, 2, 3.
- [7] ISO/IEC 14598, "Information Technology - Software product evaluation - Part 1, 2, 3, 4, 5, 6.
- [8] Edward A. Lee, "What's Ahead for Embedded Software?", IEEE Computer, September 2000, p. 18-26.
- [9] Steve Heath, "Embedded systems Design", Newnes, 1997.
- [10] Jane Liu, "Real-Time Systems", McGraw Hill, 2001.
- [11] Jean J. Labrosse, "Embedded Systems Building Blocks; Complete and Ready-to-Use modules in C", CMP books, 2000.
- [12] 情報處理振興事業協會技術センター, "ソフトウェア品質評価モデルの調査報告書(その2)", 1987.

이 선 원(Sun-Won Lee)

[정회원]



- 1996년 상주대학교 의상디자인학과 졸업(미술학사)
- 1999년 국민대학교 대학원 경영학과 마케팅 전공(경영학석사)
- 2004년~현재 서울벤처정보대학원대학교 박사과정
- 2005년~현재 국가기술자격정책심의위원

- 2006년~현재 산업인력공단 직종별 전문위원

<관심분야>

S/W 품질보증과 품질평가, S/W 개발 방법론

이 하 용(Ha-Yong Lee)

[정회원]



- 1993년 강원대학교 전자계산학과 졸업(이학사)
- 1995년 강원대학교 대학원 전자계산학과 소프트웨어 공학 전공(이학석사)
- 2005년 호서대학교 벤처전문대학원 컴퓨터응용 기술학과졸업(공학박사)

- 1996년~2005년 경희대, 경원대, 선문대, 호서대 컴퓨터공학부 강사
- 1995년~2002년 한국S/W품질연구소 선임연구원
- 2005년~현재 서울벤처정보대학원대학교 교수

<관심분야>

소프트웨어공학(특히, S/W 품질보증과 품질평가, 품질관리, 객체지향 프로그래밍, 객체지향 분석과 설계, 컴포넌트 기반 S/W 개발방법론, 품질평가)

양 해 술(Hae-Sool Yang)

[정회원]



- 1975년 홍익대학교 전기공학과 졸업(학사)
- 1878년 성균관대학교 정보처리학과 졸업(석사)
- 1991년 日本 오사카대학 기초공학부 정보공학과 소프트웨어공학 전공(공학박사)
- 1975년 ~ 79년 육군중앙경리단 전자계산실 시스템분석장교

- 1980년~95년 강원대학교 전자계산학과 교수
- 1986년~87년 日本 오사카대학교 객원연구원
- 1994년~95년 한국정보처리학회 논문편집위원장
- 1995년~2002년 한국S/W품질연구소 소장
- 2001년~현재 한국정보처리학회 부회장
- 1999년~현재 호서대학교 벤처전문대학원 교수

<관심분야>

소프트웨어공학(특히, S/W 품질보증과 품질평가, 품질관리, 품질컨설팅, OOA/ OOD/OOP, CASE, SI), S/W 프로젝트관리, 컴포넌트 기반 개발방법론과 품질평가