

논문 2019-2-2 <http://dx.doi.org/10.29056/jsav.2019.12.02>

# ISO/IEC 9241.10 표준에 기초한 소프트웨어 완성도-하자 감정 기법 연구

김도완\*†

## Software Completeness Evaluation based on ISO/IEC9241.10

Do-Wan Kim\*†

### 요 약

한국저작권위원회에 의뢰된 소프트웨어 저작물 감정 대상 중 25% 이상은 소프트웨어 완성도-하자 감정이 다. 기존 소프트웨어 완성도-하자 감정 사례의 대부분은 해당 소프트웨어의 기능성에 국한하여 계약서에 포함된 또는 고객이 원하는 요구사항들이 구현되어 작동되는지 확인하는 방식으로 이루어졌다.

본 논문은 소프트웨어 완성도 정의에 부합할 수 있도록, 보다 체계적이고 합리적인 완성도-하자 감정 기법을 제안한다. ISO/IEC 9241.10 표준은 소프트웨어 품질 제고를 위한 설계표준이라 할 수 있다. ISO/IEC 9241.10 표준은 준수되어야 하는 7개 항목을 규정하고 있으며, 작업을 위한 기능상의 완전성과 작업 능력 효율화를 위한 사용상의 완전성을 요구하고 있다.

본 논문에서 제시된 소프트웨어 완성도-하자 감정 방법론은 소프트웨어의 질적 품질에 대한 완성도 감정으로 기존 기능구현-작동 여부 감정 방법론은 보완하고 있다.

### Abstract

More than 25% of software copyright evaluations commissioned to the Korea Copyright Commission are software completeness-defective evaluations. Most of the existing software completeness-defective evaluation cases were made to ensure that the requirements contained in the contract or customer's desired requirements were implemented and able to run, limited to the functionality of the software.

This paper proposes a more systematic and reasonable maturity-correction technique to meet software completeness definitions. The ISO / IEC 9241.10 standard is a design standard for improving software quality. The ISO / IEC 9241.10 standard specifies seven items that must be complied with and requires functional integrity for work and operational integrity for work efficiency.

The software completeness-defective methodology presented in this paper complements the existing function's implementation-functioning methodology with completeness evaluation on the quality of software.

**한글키워드** : 소프트웨어 완성도, ISO/IEC9241.10, 사용성평가, 기능성평가, 소프트웨어 품질평가

**keywords** : Software completeness, ISO/IEC9241.10, Usability evaluation, Functionality evaluation, Software quality evaluation

\* 배재대학교 정보통신공학과 교수

† 교신저자: 김도완(e-mail: dwkim@pcu.ac.kr)

접수일자: 2019.10.21. 심사완료: 2019.12.10.

게재확정: 2019.12.20.

## 1. 서론

소프트웨어 관련 분쟁에 있어서 한국저작권위

원회에 의뢰된 SW 저작물 감정 분야 중 25% 이상은 개발된 소프트웨어의 완성도 또는 하자 감정이다[1]. 소프트웨어 완성도 또는 하자 분쟁 사례 대부분은 개발자와 발주자 관계에서, 분쟁원인에 대한 법적 책임관계를 밝히기 위하여 감정 의뢰된다.

소프트웨어 완성도-하자 분쟁의 원인은 계약 관계에서 개발된 소프트웨어에 대한 발주자(수요자)의 불만족이다. 기 개발된 소프트웨어에 대해 만족, 불만족에 대한 표출은 발주자(수요자)측면의 ‘고객(Customer)’, ‘의뢰인(Client)’, ‘사용자(User)’로부터 표출될 수 있다. 고객은 발주기업 또는 기관과 같이 해당 SW를 사용할 조직체, 의뢰인 또는 사용자일 수 있으며, 사용자는 실제 작업 현장에서 해당 SW를 사용하는 최종 사용자(End-User)를 의미한다.

기 개발된 SW에 대하여 만족 또는 불만족 등 그 품질을 결정하는 척도는 고객, 의뢰인, 사용자에게 따라 조금씩 다를 수 있다. 일반적으로 고객에게 좋은 SW를 판별하는 기준은 성능(Performance), 유지보수용이성(Maintainability), 신뢰성(Dependability), 효율성(Efficiency)이 될 수 있으며, 특정한 작업을 수행하는 사용자에게는 기능성(Functionality)과 사용성(Usability)이 중요한 척도이다[2].

J. McManus & T. Wood-Harper 연구에 따르면 23.8%의 SW 프로젝트가 실패하거나 중도에 포기되어지며, 수행 완료된 SW 프로젝트 중 30% 이상 프로젝트는 개발자와 발주자 사이에 품질, 개발기간초과, 비용초과 등으로 인한 심각한 문제를 일으킬 수 있는 상태로 종결되었다[3].

기존 SW 완성도-하자 감정 사례의 대부분은 해당 SW의 기능성에 국한하여 계약서에 포함된 또는 고객이 원하는 요구사항들이 구현되어 작동되는지 확인하는 방식으로 이루어졌다. 본 논문에서는 고객과 사용자 관점을 함께 고려한 새로

운 SW 완성도-하자 감정 기법을 ISO/IEC 9241.10 설계 표준 지침을 고려하여 제시한다.

## 2. 관련연구

SW 실제 사용자 그룹인 사용자에게 SW 품질 척도는 얼마나 우수한 사용성을 해당 SW가 가지고 있는가? 이다. 아무리 기능적으로 뛰어난 SW라고 할지라도 그 사용성이 떨어진다면, 사용자로부터 불만족의 아우성을 받을 수밖에 없다. SW 완성도-하자 분쟁의 사례에서도 개발자가 주장하는 기능의 완전한 구현과, 사용자가 표출하는 SW 사용성의 결여에 의한 불만에 따른 갈등을 쉽게 찾아 볼 수 있다. 사용자 입장에서 기능의 완전한 구현은 기본적인 전제일 뿐이며, 사용자는 원하는 기능을 얼마나 쉽고 효율적으로 사용할 수 있는가 여부가 중요할 뿐이다. 따라서, SW 사용자 인터페이스는 기능의 호출과 실행을 위하여 사용자가 일차적으로 사용하는 SW 구성 요소로서, 사용자 관점에서는 SW 완성도를 결정하는 중요 요소이다.

사용자인터페이스 중심의 SW 사용성 평가에 관한 연구는 SW 품질 평가 방법으로 널리 연구되었다[8]. ISO/IEC 9126은 SW 품질특성에 대한 추상적 레벨의 표준 규정이다[4]. ISO/IEC 9241.10은 SW 품질제고를 위한 설계 표준으로 작업적합성(Suitability for the task), 자명성(Self descriptiveness), 조종성(Controllability), 기대부합성(Conformity with user expectations), 에러건고성(Error tolerance), 개별화성(Suitability for individualization) 그리고 학습성(Suitability for learning) 7개 속성을 규정하고 있다[5]. 위 표준의 7개 속성은 사용자의 SW 사용성 뿐만 아니라 소프트웨어의 완성도를 결정짓는다 할 수 있다.

J. Nielsen은 그의 연구에서 적절한 피드백을 사용자에게 줄 것(Visibility of system status), 사용자에게 시스템에 대한 완전한 통제권 부여(User control and freedom), UI일관성 유지를 통한 학습성 제고(Consistency and Standards), 오류 상황에서 쉬운 원상회복 및 예방(Error prevention), 사용자 선호에 따른 인터페이스 조정(Flexibility and efficiency of use) 등의 SW 사용성 속성을 제시하였다[6]. J.Nielsen의 제안은 ISO/IEC 9241.10 설계표준 속성과 대부분 일치한다고 할 수 있다.

일반적으로 SW 완성도-하자 감정은 납품된 SW의 품질에 대한 불만족에 기인하는 법적 갈등에 의하여 의뢰된다[7]. 위 연구 및 표준 규정에 언급된 SW 품질 특성의 하나로서 사용성과, SW 완성도-하자 감정에서 SW 사용성은 떼어서 규정하기 쉽지 않다. 왜냐하면 SW 완성도-하자 감정은 SW 설계-구현-실행 전체에 대한 평가일 수 있기 때문이다. 따라서 SW 완성도-하자 감정이라는 것을 고려하면 오히려 ISO/IEC 9241.10에 규정된 7개 속성 모두가 SW 완성도를 제고하기 위한 설계표준으로 SW 완성도 - 하자 감정 척도가 될 수 있을 것으로 보인다.

### 3. ISO/IEC 9241.10 기초한 소프트웨어 완성도-하자 감정 기법

한국저작권위원회 컴퓨터프로그램 완성도 감정 메뉴얼은 완성도를 다음과 같이 정의하고 있다[13];

“프로그램의 완성도는 프로그램이 질적으로 완성된 정도, 혹은 프로그램이 사용자의 요구를 만족한 정도”

대법원 판례에 따르면 SW 도급관계에서 제작

물의 공급 업무가 완성되었음을 판단하는 기준은 다음과 같다[10];

“제작물 공급계약에서 일이 완성되었다고 하려면 당초 예정된 최후의 공정까지 일단 종료하였다는 점만으로는 부족하고 목적물의 주요 구조 부분이 약정된 대로 시공되어 사회통념상 일반적으로 요구되는 성능을 갖추고 있어야 함으로 ...”

즉, SW 완성도라고 함은 프로그램이 도급자(사용자)의 요구를 ‘질적으로 만족시키는 정도’라 정의할 수 있으며, ‘질적으로 만족시키는 정도’란 다시금 도급 계약관련 문서에 포함될 수 있는 ‘표현될 수 있는 기능적 요구사항’을 만족시키는 정도와, 도급 계약관련 문서에 포함되기 어려운 ‘표현되기 어려운 사용상의 요구사항’을 만족시키는 정도로 나누어 볼 수 있다. ‘표현되기 어려운 사용상의 요구사항’은 “사회통념상 일반적으로 요구되는 성능”을 의미한다. 예를 들어 메뉴 또는 아이콘의 디자인 및 작동 방식 등 사용자 그룹의 선지식(Pre-knowledge)을 최대한 활용하여 쉽고 빠르게 효율적으로 SW를 사용할 수 있도록 지원하는 것 등이 SW 사용상의 완성도 또는 품질을 결정한다.

소프트웨어 공학적으로 소프트웨어 개발은 크게 요구분석(Requirements Specification)-개발(Development)-평가(Validation)-개선(Evolution) 단계를 거친다[6]. 이때 SW 설계는 요구분석단계에서 고객의 요구사항을 반영하는 SW Design 과 실제 코딩 및 부분적 테스트가 실시되는 개발 단계에서 적용된다고 할 수 있다. 즉 SW 설계는 발주자의 지원 아래 개발자가 주도적으로 작업을 수행하며, 고객은 개발자에게 원하는 요구사항을 전달하는 것이 일반적이다. 반면에 개발이 끝나고 인수를 위한 시험 및 평가 단계에서는 고객이 원하는 SW가 개발되어 납품되는지 평가

하게 된다. 이때 고객은 요구한 기능들이 개발된 SW 인터페이스 상에서 정상적으로 작동하는지? 원하는 결과를 얻기 위하여 해당 기능을 문제없이 실행시킬 수 있는지? 평가한다.

우리는 본 논문에서 SW 완성도-하자 감정을 위해 ISO/IEC 9241.10 SW 품질제고를 위한 설계 표준 7개 속성을 활용할 것을 제안한다. 작업 적합성, 자명성, 조종성(Controllability), 기대부합성, 에러견고성(Error tolerance), 개별화성 그리고 학습성이 그것이다.

작업적합성(Suitability for the task)은 기능의 완성도 평가 척도이다. 작업적합성은 SW 환경이 사용자의 작업수행에 불필요한 어려움을 가중시키는 것 없이 사용자를 지원하여야 한다는 규정이다. 즉 작업적합성은 본질적으로 시스템의 기능에 관계된다. 현행 SW 완성도-하자 감정에서는 요구되는 기능이 완전하게 작동하는가? 또는 SW 기능을 실행하여 원하는 기능상의 결과를 얻을 수 있는가? 평가한다. 따라서 ISO/IEC 9241.10 작업적합성 설계 표준 항목은 SW 기능의 완성도 평가 규정으로 활용할 수 있다.

위 작업적합성을 제외한 나머지 6개 ISO/IEC 9241.10 설계 표준 항목은 SW 사용 완성도 평가를 하는데 적용된다. 구현된 SW 기능을 실행시키기 위하여 이용하는 사용자 인터페이스의 자명성, 조종성, 기대부합성, 에러견고성, 개별화성 및 학습성 척도는 사용자가 판단하는 SW 사용성이다. 즉 사용자 입장에서 SW 사용성에 관한 완성도라고 할 수 있다. 위 6개 항목의 평가 척도는 사용자 인터페이스 구현 표준지침으로 SW 사용성에 관한 완성도 평가에 적용된다[9].

본 논문은 SW 완성도-하자 감정을 위하여 SW 기능 완성도와 SW 사용 완성도를 각각 기능 별로 평가 감정하고 그 결과를 병합하는 방식을 제안한다. SW 사용 완성도 평가는 실제 사용자테스트를 하는 방법과, SW 인간공학

(Ergonomics) 전문가에 의한 사용자인터페이스 테스트를 하는 방법이 있다. 이때 사용자인터페이스 디자인 표준화 지침이 평가 척도로 활용된다.

### 3.1 SW 기능 완성도-하자 감정평가

SW 기능 완성도-하자 감정 평가의 직접적인 근거는 ‘요구사항명세서(Requirements Analysis Documents)’이다. ‘요구사항명세서’는 발주자가 개발사에 제시하거나, 개발사가 발주자의 지원 아래 분석 작성하기도 한다. 요구사항에 대한 개진은 도급자인 발주자가 수급자인 개발자에게 하는 것이 일반적인데, 요구사항은 시스템의 기능 및 성능과 직간접적으로 관계되며, 도급계약의 핵심 문서라고 할 수 있다. 요구사항명세서는 ‘기능적 요구사항’과 ‘비기능적 요구사항’을 포함한다. 시스템 기능구현과 직접적으로 관계되는 요구사항을 ‘기능적 요구사항’이라 정의할 수 있으며, 기능구현에 참작되어야 하는 요구사항을 ‘비기능적 요구사항’이라 정의할 수 있다. ‘비기능적 요구사항’은 일반적으로 구현된 시스템의 기능적 성능 수치, 표준지침 준수, 발주자의 정책 등이 될 수 있다.

한국저작권위원회에서 발간한 “저작권관련 감정사건 판례집”에 나타난 SW 완성도-하자 평가의 대부분은 SW 기능 완성도-하자 감정이다 [11][12]. 기능 완성도 감정은 “위탁 개발된 컴퓨터프로그램에서 발생하는 기능상의 문제점, 단위 프로세스 결과에 대한 신뢰상의 문제점, 운용상 문제점 등을 대상 프로그램, 제안서, 개발계약서, 시스템 설계도 및 개발작업 명세서 등의 자료 분석과 실제 작동 시스템을 검증하여 완성 및 하자 정도를 판단”으로 정의되었다[13]. 이때 완성도에 영향을 미치는 하자는 ‘기능의 미완성’, ‘기능의 미작동 또는 부분작동’ 및 ‘기능의 불완전한 작

동'을 의미한다[13]. SW 기능 완성도 감정 프로세스는 ① 제시되거나 확보된 문서를 중심으로 감정요청사항 분석 및 쟁점확인 ②감정기준 및 방법 설정, ③감정수행, ④감정결과 산출의 단계를 거친다[13]. ③감정수행 단계에서 ① 단계에서 확보된 문서를 근거로 테스트 시나리오를 작성하고, 테스트 환경을 구축하여 '요구분석명세서'에 리스팅되어 있는 기능 하나 하나를 평가하여 완성 정도를 '완성(정상동작)', '미완성(미구현)', '미작동(미동작)', '부분작동(부분구현, 부분동작)', 또는 '오류(오작동, 오동작)'으로 감정한다[13]. ④감정결과 산출 단계에서 완성도는 전체 기능 중 '완성(정상작동)'으로 감정된 기능의 비율과 기능의 중요도에 따르는 가중치 등을 고려하여 산정된다[13].

위 [13] 한국저작권위원회 컴퓨터프로그램 감정 매뉴얼에서 SW 완성도 감정은 SW 기능 완성도-하자 감정을 표준화하여 제시하고 있으며, SW 기능 완성도-하자 감정을 위하여 한국저작권위원회 SW 완성도 감정 매뉴얼을 적용하여 ISO/IEC 9241.10 SW 품질제고를 위한 설계 표준에 명시된 '작업적합성' 완성도를 감정 평가한다.

### 3.2 SW 사용 완성도-하자 감정평가

SW '사용성(Usability)은 SW 기능을 실행시킬 때 사용자가 판단하는 만족도(User Satisfaction) 및 사용의 수월성(ease of understanding and learning)이다. 국제표준에서 사용성에 대한 정의는 다음과 같다.

"The capability of the software product to be understood, learned, used and liked by the user, when used under specified conditions.

NOTE 1 Some aspects of functionality, reliability and efficiency will also affect

usability, but for the purposes of ISO/IEC 9126 are not classified as usability.

NOTE 2 Users may include operators, end users and indirect users who are under the influence of or dependent on the use of the software. Usability should address all of the different user environments that the software may affect, which may include preparation for usage and evaluation of results." [14]

"The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use"[15]

"The ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component"[16]

ISO/IEC 9241.10 SW 품질제고를 위한 설계 표준은 SW 사용 완성도를 감정 평가하는 표준으로 적합하다. 왜냐하면 ISO/IEC 9241.10은 SW 품질제고를 위한 설계 표준으로, 도급관계에서 개발된 SW가 국제 설계 표준을 얼마나 준수하고 있는가 여부를 감정인의 전문가적 식견으로 평가할 수 있기 때문이다. ISO/IEC 9241.10 SW 품질제고를 위한 설계 표준 항목 중 '자명성', '조종성', '기대부합성', '에러견고성(Error tolerance)', '개별화성' 그리고 '학습성'이 어느 정도 높은 수준의 SW 사용성을 사용자에게 제공하고 있는지 평가이다.

자명성(Self-descriptiveness)은 소프트웨어 시스템의 사용자가 자신의 액션에 대한 직접적이고 관계되는 시스템의 반응(Reaction)을 어려움 없이 이해할 수 있어야 한다는 것을 의미한다. 즉 시스템이 사용자의 액션을 정확히 이해하고, 그에 합당하는 정보를 제공하는 것 뿐만이 아니라, 사용자의 도움말 요청에 대하여 실제 문제상황에

서 해결에 직접적으로 관계되는 상황인지 정보를, 사용자의 지적 수준에 맞추어 제공하는 것을 포함한다.

학습성(Suitability for learning)은 만약 시스템이 시스템의 다양한 사용단계에서 사용자의 지식 습득 시간을 최소화 할 수 있다면 이 소프트웨어 시스템은 학습성을 지원한다.

기대부합성(Conformity with user expectations)이란 소프트웨어 시스템의 대화 흐름이 사용자의 기대에 일치하여야 한다는 것을 의미한다. 이는 사용자가 HCI에서 다른 시스템과의 경험상의 지식을 활용할 수 있도록 하며, 사용자에게 지식습득을 쉽게하고 부담감을 감소시킨다. 기대부합성에 대한 규정은 시스템의 일관성, 투명성을 보장하여 시스템을 사용하기 위한 사용자의 심성모델(Mental model) 형성을 촉진시킨다.

조작성(Operability & Controllability)은 소프트웨어 시스템이 사용자에게 사용자의 작업취향에 맞도록 시스템의 흐름 및 작업환경에 영향을 미칠 수 있도록 허용하는 것을 의미한다. 이는 사용자가 HCI에서 주도적으로 시스템을 컨트롤 할 수 있도록 허용하여, 사용자가 상호작용(Interaction)을 질적으로 양적으로 조종할 수 있도록 규정하고 있다.

개별화성(Suitability for individualization) 항목은 사용자의 필요나 주어진 작업에 대한 사용자의 숙련도에 따라 소프트웨어 시스템이 변경되어질 수 있다면, 이 소프트웨어 시스템은 개별화성을 지원한다고 할 수 있다. 개별화성은 따라서 조작성과 깊이 관계되며, 개별화를 위한 사용자의 선지식(Pre-knowledge)을 요구한다.

에러견고성(Error tolerance): 만약 사용자의 에러에 대하여 에러 수정을 위한 최소의 소모로서 원하는 작업결과를 가져올 수 있다면 그 시스템은 에러에 대하여 견고하다고 말할 수 있다.

감정인은 위 6개 표준 설계 항목에 대하여, 사용자인터페이스 디자인의 완성도를 평가한다. 이때 평가의 근거는 운영체제, 개발언어, 응용프로그램 도메인 등에 따른 사용자인터페이스 디자인 가이드라인이 준수되고 있는지이다. OSF/Motif의Style Guide[17], Apple사의 Macintosh Human Interface Guidelines[18][19][20], ISII(Interfaced Systems International Inc.)의 Graphical User Interface Style Guide[21][20], Microsoft Designer Guide[23] 등 운영체제, 개발언어, 응용도메인, 실행환경에 따라 준수되어야 하는 범용적인 가이드라인이 존재한다. 이러한 가이드라인이 준수되었을 때 높은 수준의 ISO/IEC 9241.10 SW 품질제고를 위한 설계 표준 항목 중 위 6개 감정 항목에 대한 SW 사용 완성도가 달성될 수 있다.

#### 4. ISO/IEC 9241.10 표준에 기초한 소프트웨어 완성도-하자 감정 방법 결론 및 향후 연구

한국저작권위원회컴퓨터 프로그램 감정 메뉴얼에 정의된 “프로그램의 완성도는 프로그램이 질적으로 완성된 정도, 혹은 프로그램이 사용자의 요구를 만족한 정도”란 기능의 양적인 구현뿐만 아니라, 질적인 수준을 포함한다. 기존 SW 기능 구현 여부를 감정하여 SW 완성도-하자를 평가하는 방식은 ‘완성도’ 정의의 한 부분 ‘기능’ 구현 여부 감정에 치중되어 있었다.

본 논문은 3장에서 살펴본 바와 같이 ISO/IEC 9241.10 표준에 명시된 7개 항목을 SW 완성도-하자 감정의 표준 근거로서 삼아, 요구명세서에 적시된 기능들이 얼마나 완전하게 구현되어 작동하는가 (SW 기능 완성도)? 구현된 기능의 사용상의 질적 품질이 얼마나 만족스러운가 (SW 사

용 완성도)?를 평가하는 방법을 제안한다. 대체적으로 SW 기능 완성도 감정이 정량적인 평가 결과를 도출시킨다면, SW 사용 완성도는 정성적인 평가 결과를 도출시킨다.

SW 기능 완성도 산출을 위해서 기존 감정 기법 및 프로세스가 그대로 활용될 수 있다. SW 사용 완성도 감정은 운영체제, 프로그래밍언어, 응용도메인, 실행환경, 국제 디자인 표준 등이 사용자인터페이스 디자인에 있어 올바르게 적용되어 있는지 여부가 감정인의 전문적 판단을 뒷받침할 수 있는 근거이다. 이를 위하여 향후 보다 세밀한 연구가 필요하다.

향후 연구 과제 중 하나는 SW 완성도-하자' 감정에서 '기능 완성도'와 '사용 완성도'의 비중을 어떻게 배분하여 전체 완성도를 산정할 것 인가?이다. SW 특수성에 따라 특정 SW 응용 도메인에서는 SW를 사용하기 위한 일정한 수준의 트레이닝이 필요하기도 하며, 일반 범용 SW는 사전 트레이닝이 없이도 사용 할 수 있을 수준의 높은 사용성을 요구한다. 또한 기능의 완전성이 없다면 SW 사용은 불가하며, 기능의 완전성이 담보될 때 사용 완성도는 그 의미를 찾을 수 있다. 웹, 모바일 앱 그리고 일반 범용 SW의 경우 표준적인 디자인 스타일 가이드를 준수하여 사용성을 높이고 있다. 예를 들어 이러한 응용 실행 환경 및 응용도메인에서 SW 완성도는 '기능 완성도' 70%, '사용 완성도' 30% 비중을 두어 산정될 수도 있겠다. 또는 '기능 완성도'는 정량적으로 산정하고, '사용 완성도'는 정성적으로 서술하는 방식도 고려될 수 있다.

향후 연구 주제 중 다른 하나는 운영체제, 개발언어, 응용도메인 등에 따라, 사용자인터페이스 상에서의 '사용 완성도'를 감정하는데 요구되는 평가 항목을 규정하는 것이다. 주요 다국적 SW 기업들은 자체적인 디자인 스타일 가이드를 가지고 이를 준수할 것을 규정하고 있지만, 이는 준

수 지침으로 한정된다. SW 인간공학적 관점에서 표준화 할 수 있는 HCI 요소를 찾아내어 사용자 인터페이스 상에서의 '사용 완성도'를 감정하는데 요구되는 평가 항목을 도출하는 연구가 이루어질 것이다.

## 참 고 문 헌

- [1] 김시열, “저작물 감정제도의 효과 및 개선방안 연구”, 한국저작권위원회, 2013.
- [2] I. Sommerville, “Software Engineering”, 9th edition, Addison wesley, 2009.
- [3] J. McManus and T. Wood-Harper, “Understanding the sources of information project failure”, Management services, 2007.
- [4] ISO/IEC 9126, Information Technology-Software quality characteristics and metrics
- [5] ISO/IEC 9241.10, Dialogue principles
- [6] J. Nielsen, “10 Usability heuristics for user interface design”, 1994.
- [7] SW 감정인 워크숍, 한국저작권위원회, 2018.
- [8] 이하용, 양해술, “UI 중심의 소프트웨어 사용성 평가 방법”, The Journal of Digital Policy & Management, 2013.
- [9] 김도완, 인터페이스 표준화
- [10] 대법원 2006. 10.13 선고 2004다21862판결
- [11] 2013 저작권관련 감정사건 판례집, 한국저작권위원회, 2013.
- [12] 감정사건 판례집, 한국저작권위원회, 2016.
- [13] 컴퓨터프로그램 감정 매뉴얼, 한국저작권위원회
- [14] ISO/IEC 9126-1, 2000.
- [15] ISO/IEC 9241.11, 1998.
- [16] IEEE Std. 610.12-1990
- [17] OSF/Motif, Style Guide, Open Software Foundation, Cambridge
- [18] Macintosh Human Interface Guidelines, Addison-Wesley Publishing Company
- [19] Mac OS8 Human Interface Guidelines,

- Developer Note, Technical Publications,  
Apple co.
- [20] AQUA Human Interface Guidelines,  
Preliminary, Apple
- [21] Graphical User Interface Style Guide,  
Interfaced Systems International Inc.
- [22] The User Interface Design Process  
-Overview, Interfaced Systems  
International Inc.
- [23] Microsoft Windows Interface and Design  
guidelines, MicroSoft

————— 저 자 소 개 —————



김도완(Do-Wan Kim)

1990년 독일 Regensburg 대학교 전산정보  
공학 학사

1993년 독일 Regensburg 대학교 전산 정보  
공학 석사

1996년 독일 Regensburg 대학교 전산 정보  
공학과 박사 (Ph. D.)

1996년 한국전자통신연구원 선임연구원

2004년 Southampton Uni. (영) 방문교수

1997년~현재 배재대학교 정보통신공학과  
교수

<주관심분야> Semantic Web Service,  
Information Management, Knowledge-  
based System