

A Study for Domain Categorization and Estimation of Complexity for Reliability Improvement of Domain Analysis

Lee Eun-Ser[†]

ABSTRACT

Domain analysis is an important component for reliability of development project. Domain analysis error have an effect in the whole system. As a result, the system reliability will be deteriorated. Therefore, we need a methodology to analyze domain characteristic for a reliable analysis in the domain analysis phase. In this paper, we propose a methodology for domain categorization and estimation of complexity for reliability improvement of domain analysis.

Keywords : Domain Analysis, Domain Categorization, Domain Complexity

도메인 분석의 신뢰성 향상을 위한 도메인 분류와 복잡도 측정에 관한 연구

이 은 서[†]

요 약

도메인 분석은 신뢰성 있는 프로젝트 개발의 중요한 요소가 된다. 도메인 분석에서 발생하는 오류는 전체 시스템에 영향을 주게 되고, 그 결과 고객의 만족도가 낮아진다. 따라서 요구사항 단계에서 신뢰성 있는 분석을 위하여 도메인의 특성을 분석할 수 있는 방법이 필요하게 된다. 본 논문에서는 이와 같은 문제를 해결하기 위하여 도메인 분석의 신뢰성 향상을 위한 도메인 분류와 복잡도 측정방법을 제시하고자 한다.

키워드 : 도메인 분석, 도메인 분류, 도메인 복잡도

1. 서 론

도메인 분석은 프로젝트를 개발하는데 있어서 중요한 요인이 된다. 도메인 분석을 기반으로 요구사항을 정의하게 되는데, 시스템에 대한 요구사항은 시스템에 의해 제공되는 서비스와 그것의 운영상의 제약에 관한 기술이다.

도메인 요구사항은 개발하려는 분야의 전문 지식에 기반을 둔 내용이다. 즉, 개발되는 시스템에서의 요구사항과는 내용이 다르다. 이와 같은 노력은 좋은 소프트웨어를 만들기 위한 초석이 된다.

소프트웨어 공학자는 시스템에서 소프트웨어의 중요성 때문에 사회, 기술적 시스템과 시스템 공학에 대한 지식을 가져야 한다[6, 7].

소프트웨어 공학자는 이와 같은 지식을 기반으로 기능을 추출하게 되는데, 이 과정에서 반드시 도메인의 지식을 필수 요소로 활용하게 된다. 도메인 지식은 기능 추출의 중요한 요소이며 개발 범위와 내용을 결정하는 중요한 요소가 된다.

본 논문에서는 도메인 분석을 정확히 수행하기 위하여 도메인 특성에 의한 분류를 수행하였으며, 그 결과 도메인 복잡도를 측정하여 전반적인 도메인 분석의 신뢰성이 높아졌음을 제시하고자 한다.

2. 기반 연구

2.1 도메인 분석

도메인에 대한 분석 작업은 해당 영역에 해당되는 특화된 패턴들을 생성하여 분석 모델을 만들 수 있도록 해준다. 이와 같은 내용은 개발 비용의 감소의 효과를 얻을 수 있게 된다.

[†] 종신회원 : 안동대학교 컴퓨터공학과 부교수
Manuscript Received : November 19, 2015
First Revision : December 10, 2015
Accepted : December 10, 2015
* Corresponding Author : Lee Eun-Ser(eslee@anu.ac.kr)

소프트웨어 도메인 분석은 특히 애플리케이션 도메인 내의 다수의 프로젝트들의 재사용을 위한 특정 애플리케이션 도메인에서 나온 공통된 요구사항의 구별, 분석, 그리고 명세서이다[1].

특정 애플리케이션 도메인은 항공 전자 공학부터 은행 업무, 멀티미디어 비디오 게임부터 의학 기기에 내장된 소프트웨어에 이르기 까지 다양하다. 도메인 분석의 목표는 정확하다: 분석 클래스들이 광범위하게 적용될 수 있어 재사용될 수 있는 분석 패턴들을 발견하거나 창조하는 것이다[2, 8, 9].

도메인 분석에서는 사용자의 요구사항을 추출하는 기능적인 요구사항과 그 외에 환경 및 하드웨어와 연관된 비기능적 요구사항의 분석을 목적으로 한다.

도메인 분석은 소프트웨어 프로세스를 위한 상부 액티비티로 보여질 수도 있다. 이를 통해, 도메인분석은 어떤 한 개의 소프트웨어 프로젝트에 연결되지 않은 현재 진행형인 소프트웨어 공학활동이라는 것이다. 어떤 의미에서, 도메인 분석가의 역할은 중공업 환경에서 툴스미스(toolsmith)마스터의 역할과 유사하다. 툴스미스의 업무는 유사한 작업을 하는 많은 사람들이 사용하는 툴들을 설계하고 생성하는 것이다. 도메인 분석가의 역할은 분석 패턴들, 분석 클래스들, 그리고 비슷한 애플리케이션에서 작업하는 대다수의 사람들에게 의해 사용될 수 있는 관련 정보를 발견하고 정의하는 것이다.

도메인 분석 프로세스의 주요 입력과 출력물을 나타낸다. 도메인지식의 원천은 도메인에 걸쳐 재사용될 수 있는 객체들을 구별하기 위한 시도로 조사된다. 고객들이 더욱 기술적으로 수준이 높아짐에 따라 무엇에 만큼이나 어떻게 명세서를 작성하느냐에 대한 경향이 있다. 그러나 주요 초점은 무엇에 있어야 한다[3, 10].

도메인 분석의 보완적인 관점은 “도메인을 모델링하는 것과 연관되어 소프트웨어 공학자들과 다른 스테이크 홀더들이 그것에 대해 더 잘 배울 수 있다[4].

도메인의 이해는 도메인 전문가에 국한된 내용이 아니다. 도메인에 대한 이해를 하지 못한다면 소프트웨어 공학자에 의한 개발 과정들은 사용자를 위한 소프트웨어 개발이 아닌 개발자만의 소프트웨어를 만들게 되는 것이다.

2.2 도메인 추상화

도메인 분석은 개발 과정에서 패턴이나 컴포넌트와 같은 형태로 나타난다. 이와 같은 형식들은 도메인 지식이 함축되어 있으면 여러 기능들을 분석하여 공통된 기능과 가변적인 기능들을 추출해야 한다.

일반적으로 내부적으로 개발된 컴포넌트는 즉각적으로 재사용할 수는 없다. 내부적으로 개발된 컴포넌트는 다른 응용 시스템에서는 필요할 것 같지 않은 응용 시스템에 특별한 특성과 인터페이스를 포함한다. 따라서 더욱 일반적이고, 그러므로 더욱 재사용가능한 버전을 생성하기 위해서는 이러한 컴포넌트를 개조하고 확장해야한다. 명백히 이것은 관련된 비용을 갖는다. 먼저 컴포넌트가 재사용될 것인지의

여부를 판단하고, 다음에는 재사용을 통한 비용 절감이 컴포넌트를 재사용가능하게 만드는 비용을 정당화하는지의 여부를 판단해야 한다. 안정적인 도메인 추상화란 매우 느리게 변화하는 응용 도메인에서의 기본적인 개념이다. 예를 들어, 은행 시스템에서 도메인 추상화는 계좌, 예금주, 계좌 내역을 포함한다. 병원 관리 시스템에서 도메인 추상화는 환자, 진료, 간호사를 포함한다. 이러한 도메인 추상화를 때로는 비즈니스 객체라고 한다. 만일 컴포넌트가 공통적으로 사용되는 비즈니스 객체 혹은 관련된 객체들의 그룹의 구현이라면, 컴포넌트는 아마도 재사용될 수 있을 것이다[5].

도메인 추상화를 통하여 재사용성을 향상시킬 수 있으며, 분석과 설계 과정에서 발생하는 결함들을 줄일 수 있는 방법이기도 하다.

3. 본 론

3.1 도메인 특성에 의한 분류 (형태별로 구분)

도메인 분석은 프로젝트 수행과 완성을 위하여 중요한 요소가 된다. 도메인 분석을 통하여 요구사항 정의 및 분석, 설계, 구현, 테스트를 수행할 수 있다. 도메인 분석은 프로젝트 개발 생명주기와 상품화 생명주기의 기초가 되며, 전체 개발 프로젝트의 신뢰성에도 많은 영향을 미치게 된다.

객체지향을 위한 일반적인 도메인 분석의 과정은 다음과 같다[1].

1. 조사하려는 도메인을 정의한다.
2. 도메인에서 추출한 항목을 분류한다.
3. 도메인 내의 대표적인 응용사례를 수집한다.
4. 사례에서 각 응용을 분석하고 분석 클래스를 정의한다.
5. 클래스의 요구사항 모델을 개발한다.

이와 같은 도메인 분석 방법은 도메인의 특성을 고려하지 않은 분석 방법이다. 정확한 도메인 분석을 위하여 도메인에 따른 분석 방법이 활용되어야 한다. 도메인에 따라 분석을 하기 위하여 도메인을 분류하기 위한 기준을 마련하고자 한다.

선행되어야 할 작업으로는 도메인 분류를 위하여 도메인 분석의 목적에 따라서 공통적으로 적용되는 부분과 도메인의 특성에 따라 차별화되는 부분을 선별해야 한다.

도메인 분석을 위한 목적의 분류는 크게 비기능적인 관점과 요구사항에 기반을 둔 기능적인 관점으로 분류하였다.

비기능적인 관점의 목적은 주로 성능과 안정성 요소가 중심이 되고 있다. 비기능적 관점의 목적은 다음과 같다.

- ◆ 빠른 속도 지향 - 개발하려는 도메인이 빠른 실행을 목적으로 분석이 필요한 경우이다.
- ◆ 안정성 향상 - 시간의 흐름이 지속되어도 일관성 있는 성능을 필요로 하는 경우이다.
- ◆ 보안성 향상 - 내부 및 외부의 위협 요소로부터 시스

템을 보호하기 위한 경우이다.

기능적인 관점의 목적은 사용자 중심의 분석을 요구하는 경우이다. 비기능적 관점의 목적은 다음과 같다.

- ◆ 재사용성 향상 - 고객의 요구사항을 시스템에 반영하는 과정에서 재사용되는 기능을 추출하여 컴포넌트와 같은 형태로 분석을 하는 경우이다. 이 경우에는 소스 코드의 재작업이 주가 되지 않도록 기능을 재사용 가능한 형태로 구성을 하게 된다. 재사용성을 강조하기 위한 형태는 소스코드를 공개하지 않는 블랙박스 형태와 소스코드를 개방하여 수정이 가능하게 하는 화이트박스의 두 경우를 모두 포함한다.
- ◆ 변경되는 사항의 수용 - 도메인의 환경 요소가 변경되는 경우와 시스템의 구성요소가 변경되는 경우가 발생하기도 한다. 이를 위하여 도메인 분석 시에 변경을 수용하여 기존의 기능 품질에 영향을 최소화하기 위한 경우이다.

본 논문에서는 기능적인 관점에서 변경되는 사항의 수용의 신뢰성 있는 도메인 분석하기 위한 목적으로 도메인 분석 시에 활용할 수 있는 기준을 제시하고자 한다.

도메인 분석 시에 변경되는 사항의 수용을 위하여 도메인 분석 단계에서 이와 같은 사항을 정의해야 한다. 변경 사항은 도메인 환경의 변화나 요구사항의 변경이 빈번이 발생하는 도메인의 특성일 수도 있지만 일반적인 기능 개선과 같은 업그레이드의 형태도 포함하고 있다. 따라서 변경 사항의 수용을 통하여 시스템의 신뢰성과 유지보수의 편의성도 제공할 수 있게 된다.

3.2 도메인 복잡도 측정을 위한 기준 설계

도메인 분류를 기반으로 도메인의 복잡도를 개발 전에 예측하여 개발의 전체적인 일정과 품질, 비용에 대한 신뢰성을 제공하고자 한다. 도메인의 복잡도 측정은 일정측면, 품질측면, 비용측면에서 계획을 수립할 시점에 활용이 가능하다. 이와 같은 자료의 활용은 프로젝트 관리의 현실화를 얻을 수 있으며, 전반적인 프로젝트의 완성도를 높일 수 있다. 또한 도메인의 복잡도를 구조적인 관점과 사용 형태적인 관점 등으로 분석을 하여 정량적으로 복잡도를 측정하고자 한다. 산출된 값은 전체 시스템의 복잡도를 증가시키는 위험요소를 개발이 진행되기 전에 제공한다. 제공된 정보를 기반으로 복잡도가 증가되는 부분을 요구사항 분석이나 설계가 진행되기 전에 수정할 수 있는 기회가 된다. 이러한 수정은 형태적인 변경을 통하는 경우에 많은 효과를 얻을 수 있게 된다.

형태적인 변경의 예로는 기능의 상호작용 방법의 변경등을 들 수 있다. 따라서 전체적인 시스템의 복잡도를 낮추고 발생하는 오류의 가능성을 줄일 수 있게 된다. 각 항목의 총합은 1.0으로 산정을 한다. 각 항목의 산출 값은 전체 항목 중에 차지하는 가중치를 곱하여 총 복잡도의 산정 값은 각 항목의 합으로 산출한다. 총 복잡도의 산정 값은 1.0을

기준으로 한다. 즉, 총 복잡도가 1.0이면 복잡도가 가장 많은 도메인으로 분석할 수 있다. 도메인의 분석과정은 기존의 객체지향을 위한 일반적인 방법을 활용한다[1]. 기존의 과정에 본 논문에서는 기준을 설계하여 변경 사항의 수용을 원활히 하고자 한다. 기준의 설정은 본 논문에서는 변경 사항의 수용에 기반을 두고 있다.

도메인 복잡도 측정을 위한 특성의 분류 방법은 다음과 같은 내용을 기반으로 한다.

1) 도메인의 특성을 결정하고 가중치를 부여한다.

도메인의 정보 흐름의 특성에 의하여 다음과 같이 가중치를 부여하였다.

- ◆ 단방향 정보 제공 - 도메인에서 구축하려는 시스템의 정보가 질의에 의한 결과를 제공하는 것이 아니라 일방적인 정보를 제공하는 도메인에 해당된다. 이때 정보 제공시에 공용성과 가변성 여부를 판별한다. 단방향 정보 제공의 경우 복잡도 산정은 정보 중에서 가변성이 30%이내를 차지하면 하, 31~70%를 차지하면 중, 71~100%를 차지하면 상으로 측정한다. 상은 0.5, 중은 0.3, 하는 0.1로 수치화 한다.
- ◆ 단방향과 양방향 정보 제공을 모두 사용 - 도메인에서 구축하려는 시스템의 정보가 일방적인 정보제공과 질의에 의한 결과를 제공하는 두 가지 방식의 방법을 혼용하는 도메인에 해당된다. 이때 정보 제공시에 공용성과 가변성 여부를 판별한다. 정보 제공의 경우 복잡도 산정은 정보 중에서 가변성이 30% 이내를 차지하면 하, 31~70%를 차지하면 중, 71~100%를 차지하면 상으로 측정한다. 상은 0.7, 중은 0.5, 하는 0.4로 수치화 한다.
- ◆ 양방향 정보 제공 - 도메인에서 구축하려는 시스템의 정보가 질의에 의한 결과를 제공하는 도메인에 해당된다. 이때 정보 제공시에 공용성과 가변성 여부를 판별한다. 양방향 정보 제공의 경우 복잡도 산정은 정보 중에서 가변성이 30%이내를 차지하면 하, 31~70%를 차지하면 중, 71~100%를 차지하면 상으로 측정한다. 상은 0.8, 중은 0.6 하는 0.4으로 수치화 한다.

단방향의 경우, HTML과 같이 일방적인 정보를 제공하는 형태이다. 양방향의 경우, 질의어를 받고 그 결과를 제공하는 형태가 될 수 있다.

Table 1. Domain Character by Information Flow

	High	Middle	Low
Type of one way date	0.5	0.3	0.1
Type of one and both way date	0.7	0.5	0.4
Type of both way date	0.8	0.6	0.4

2) 도메인에서 추출할 항목을 선정하고 분류한다.

도메인에서 추출할 항목을 선정하고 분류를 명확히 해야 한다. 이와 같은 작업을 수행하는 것은 기능의 명세의 중요한 요소가 되고, 시스템의 만족도를 결정하는 중요한 요인이 된다. 요추출 항목은 다음과 같다.

- 입력 - 항목의 입력물
- 출력 - 항목의 결과물
- 사용되는 자료(DB등) - 사용되는 자료를 정의
- 요구되는 하드웨어 - 필요한 하드웨어를 정의
- 인터페이스 타입과 개수 - 도메인 분석을 통한 인터페이스의 타입과 개수 추출
- 시작조건 - 추출 항목의 시작 조건 정의
- 종료조건 - 추출 항목의 시작 조건 정의

위 항목들을 추출할 수 있는 정도가 추출할 항목 선정과 분류의 복잡도를 측정하는 중요한 요인이 된다. 1-3개 추출은 하, 4-5추출은 중, 6-7개 추출은 상으로 세분화 하였다. 상은 0.2, 중은 0.5, 하는 0.8로 수치화 하였다.

Table 2. Domain Character by Extraction Items

	High	Middle	Low
Number of extraction items	0.2	0.5	0.8

3) 도메인 전문가의 능력에 의하여 도메인 내의 대표적인 응용사례를 수집을 한다.

도메인 내의 응용 사례는 분석 과정에서 시나리오로 활용이 된다. 시나리오를 기반으로 설계가 수행되므로 응용 사례는 중요한 시스템의 핵심 내용이 된다. 응용 사례에 대한 조사는 도메인 전문가의 노하우를 얻어서 정리하게 된다. 따라서 이 항목에서는 중요한 요소가 도메인 전문가와 응용 사례의 구체성이다. 도메인 전문가의 해당 도메인의 경력이 5년 이상이면 상, 3~5년 이면 중, 3년 미만이면 하로 산정하였다. 산정 값은 상은 0.3, 중은 0.5, 하는 0.8로 수치화하였다.

Table 3. Domain Character by Ability of Domain Expert

	High	Middle	Low
Ability of domain expert	0.2	0.5	0.8

1), 2), 3) 항목에 가중치를 곱하여 전체 산정 값을 결정한다. 이때 각 항목의 가중치의 합은 1을 넘지 않는다. 따라서 도메인 복잡도 산정 값도 1을 넘지 않는다. 도메인 산정 값의 수식은 다음과 같다.

$$\text{도메인 복잡도} = (1\text{항} \times \text{가중치}) + (2\text{항} \times \text{가중치}) + (3\text{항} \times \text{가중치})$$

도메인 복잡도의 활용은 각 항목의 산정 값을 기준으로 초기 도메인의 복잡도가 발생하는 부분을 인지하여 도메인 분석과 요구사항 분석 과정에서 해당 부분의 복잡도를 경감시키기 위한 방법으로 활용될 수 있다. 예를 들면, 1번 항목에서 양방향에 치중된 도메인 정보 흐름의 복잡도가 높게 측정된다면 독립적이고 공용성이 있는 불변의 자료를 분류하여 단방향 정보흐름으로 유도를 하여 복잡도를 낮추는 방법이다.

공용성은 모든 도메인에서 변경없이 공통으로 사용할 수 있는 부분이다. 가변성은 다른 도메인과 차별화가 되는 부분으로서 독자적인 특성을 갖는 부분이다.

4. 사례 연구

4.1 개발 배경

- ◆ 스마트폰 사용자가 점점 증가하면서 노년층과 유아층 사용자도 함께 증가하고 있다. 하지만, 그런 사용자들을 위한 앱은 찾기 쉽지 않다.
- ◆ 소외계층의 사용자에게 카카오톡, 네이버 등을 설치하고 추가적으로 사용할 수 있는 앱을 제작하고자 한다.
- ◆ 노년층을 위한 앱은 고등학생이 작성한 기초 자료를 활용, 아이디어를 구현해가는 과정을 함께 포함하고 있다

4.2 요구사항 관리

1) 목적

요구사항 관리는 프로젝트 진행 전 기간에 걸쳐서 지속적으로 관리되어야 하는 항목이며, 최초 프로젝트를 시작했을 당시에 나왔던 요구사항부터 프로젝트 진행에 따라 변경되는 요구사항까지 모든 항목들을 관리하여, 프로젝트에서 변경되어야 할 부분은 변경시킬 수 있도록 하기 위해 유지 관리 하는 것에 그 목적이 있다.

각 단계별로 요구사항관리에 대한 작업이 진행되어야 한다.

2) 적용 범위

본 프로젝트의 사업범위에 해당하는 모든 범위에서 적용할 수 있다. 단 변경은 최대한 코딩이 완료되기 전에 이루어질 수 있도록 한다.

개발 과제는 노년층을 위한 스마트폰 앱을 개발하는 것으로서 기능은 돋보기, 사진 보정, 음성 출력의 기능을 대상으로 하고 있다. 개발 과제의 유스케이스는 다음과 같다.

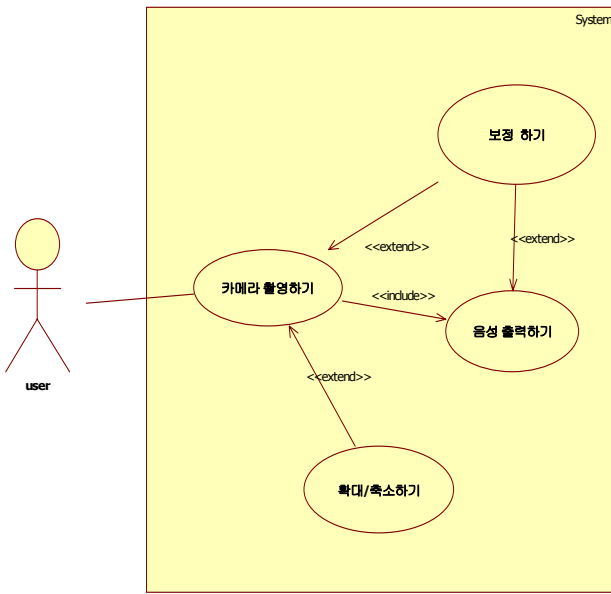


Fig. 1. Use Case Diagram

◆ 시나리오

카메라 촬영하기-사용자는 application을 실행시켜 카메라를 촬영한다.

음성 출력하기-사용자는 보정된 이미지를 음성으로 출력한다.

확대/축소하기-사용자는 촬영된 이미지를 확대/축소한다.

보정하기-사용자는 카메라 촬영할 때 보정하기 기능으로 올바른 이미지를 출력 한다.

사용자는 보정하기 기능으로 촬영된 이미지를 음성으로 출력한다.

개발 과제의 내용을 기반으로 3장에서 제시한 이론을 기반으로 도메인 복잡도를 측정하고자 한다. 도메인의 특성은 단방향과 양방향이 병행된 형태로 분석되었다. 수준은 중으로 분석되어서 0.5로 수치화 되었다. 이 항목은 프로젝트에서 의존성이 높다. 따라서 50%의 가중치를 부여하였다. 도메인에서 추출항목의 개수는 중으로 분석되었으며 0.8로 수치화되었다. 이 항목은 프로젝트에서 의존성이 낮다. 따라서 25%의 가중치를 부여하였다.

도메인 전문가의 능력에 대한 내용으로는 하로 분석되었으며 0.8로 수치화 되었다.이 항목은 프로젝트에서 의존성이 높다. 따라서 25%의 가중치를 부여하였다.

산출된 결과를 기반으로 도메인 복잡도는 다음과 같이 산출된다.

$$\begin{aligned} \text{도메인 복잡도} &= (0.5 \times 0.5) + (0.8 \times 0.25) + (0.8 \times 0.25) \\ &= 0.25 + 0.2 + 0.2 = 0.65 \end{aligned}$$

전체적인 도메인의 복잡도는 0.575로 중간 단계의 복잡도가 산출되었다.

결과적으로, 도메인 추출항목의 개수와 도메인 전문가의

능력 부분에서 많은 복잡도를 내재하고 있었다. 따라서 산출 결과를 기반으로 도메인 분석 과정에서 복잡도를 낮추기 위한 활동을 수행할 수 있다. 이와 같은 활동은 프로젝트 개발 시 전체 시스템의 복잡도를 낮추고 신뢰성을 높일 수 있게 된다. 개선 내용은 다음과 같다.

◆ 도메인 복잡도 산출 항목과 개선 사항

- 문자 관독을 위한 담당 기능의 불명확성 ↔ 메시지를 전달할 수 있는 기능 마련

- 이미지 편집기능에 대한 품질 측정의 불명확성 ↔ 품질 측정 기준의 마련

- 사용자가 기능을 시작할 수 있는 방법이 없음 ↔ 기능 시작의 명령어 전달 함수 마련

개선 사항을 기반으로 도메인 복잡도에서 산출되었던 부분을 개선하여 요구사항 분석, 설계, 코딩에서 이와 같은 문제를 경감시키고 발생되지 않도록 수행하였다. 따라서 전반적인 시스템에서 개선 효과를 얻을 수 있게 된다.

5. 결론 및 향후 연구 방향

본 논문에서는 도메인의 특성에 따라서 개발하려는 분야가 복잡해 진다. 따라서 도메인에서 복잡도를 발생하는 요인을 특성에 의하여 미리 찾아서 실제 시스템 개발 시에 이와 같은 사항이 발생되진 않도록 하여 전체 시스템의 신뢰성을 향상시키고자 한다.

기존의 기능 점수 방식과 제시하고자 하는 이론은 다음과 같은 차이점이 있다.

기능 점수는 함수의 형태를 5가지(입력, 출력, 질의, 내부 파일, 외부 파일)로 구분하여 가중치를 산정하게 되어 있다. 본 논문의 이론에서는 함수 부분이 중심이 아니고 상위 단계에서 분석을 하는 부분에 초점을 맞추고 있다. 따라서 기능 점수 방식보다는 그 이전 개발 단계에서 신뢰성을 얻고자 한다.

향후 연구로는 도메인 복잡도를 산출하기 위한 기준의 측정값을 많은 사례를 기반으로 검증이 필요하다. 또한 검증을 통하여 기준의 재 정의가 필요하다. 그리고 본 논문의 이론을 웹과 앱을 통하여 연동할 수 있는 도구의 개발이 필요하다.

References

[1] Kim sung kyu, Software engineering, Han san, p.301, 2011.
 [2] D. G. Firesmith, "Object-oriented requirements analysis and logical design: A Software Engineering Approach," John Wiley & Sons, Inc. New York, NY, USA, 1993.
 [3] R. S. Arnold, "Software Restructuring," *Proc. IEEE*, Vol.77, No.4, pp.607-617, 1989.
 [4] T. Lethbridge and R. Laganie, "Object-Oriented Software Engineering Practical Software Development Using UML and Java," McGraw-Hill, 2001.

- [5] Sommerville, "Software engineering," Pearson, 2011, p.83.
- [6] S. A. White and M. Alford, et al., "System engineering of computer-based systems," *IEEE Computer*, Vol.26, No.11, pp.54-65, 1993.
- [7] Yoon chung, "Successful Software development methodology," Life power press, 1999.
- [8] Shari lawrence, "Software engineering forth edition," Pearson, p.208, 2010.
- [9] Choi eun man, "Software engineering," Jungik publishing co, 2011, p.156.
- [10] Han huck su, "Introduction of software engineering," Hongrung, 2011, p.117.



이 은 서

e-mail : eslee@anu.ac.kr

2001년~현 재 ISO/IEC 15504 국제 선임
심사원

2004년 중앙대학교 컴퓨터공학과(박사)

2004년~현 재 임베디드 산업협회 전문
위원

2004년~현 재 한국정보통신기술협회 위원

2012년~현 재 안동대학교 컴퓨터공학과 부교수

관심분야: CBD, Formal method, Quality model, SPI(Defect
Analysis)