

Korea Conference on Software Engineering 2012
(KCSE-2012)

2012

한국 소프트웨어공학 학술대회 논문집

- 일시: 2012년 2월 8일(수) ~ 10일(금)
- 장소: 강원도 보광 휘닉스파크內 한화리조트



소프트웨어공학
소사이어티

제 14 권 제 1 호
Vol. 14 No. 1



- 주최: 한국정보과학회, 한국정보처리학회
주관: 한국정보과학회 소프트웨어공학 소사이어티
한국정보처리학회 소프트웨어공학 연구회
한국전자통신연구원
- 후원: 비트컴퓨터,
고려대학교 고신뢰 융합소프트웨어 연구센터,
단국대학교 금융 IT 를 위한 소프트웨어공학연구센터,
서강대학교 SW 요구 및 검증공학기술 연구센터,
포항공대 융합소프트웨어개발 연구센터,

KCSE 2012 프로시딩 목차

SW 개발방법론

Agile 방법론에 대한 동양 문화 관점에서의 고찰 김민호, 조성범, 서승현, 이창호, 민상윤 (KAIST)	1
XML 기반 신뢰성있는 관점 지향 기법 (우수논문) 김은선, 이병정, 이재호 (서울시립대학교)	12
파라메트릭 소프트웨어 매트릭스 기반의 LOC 정의 (short) 신원, 장천현 (건국대학교)	14
글로벌 소프트웨어 개발 방법 적용 사례 연구: A-SQUARE 프로젝트 (산업체논문) 박태현, 강성원 (KAIST)	18

시멘틱웹

모바일 상황 인식 서비스 환경에서 향상된 추론 결과를 획득하기 위한 온톨로지 모델 선택 기법 (우수발표논문) 권오현, 박준석, 염근혁 (부산대학교)	42
온톨로지 재사용을 위한 온톨로지 저장소와 SWRL 구성 도구의 구현 및 적용 이인환, 남태우, 류자치, 염근혁 (부산대학교)	44
OWL Ontologies 를 이용한 XML Schema 파일 관리 (short) 박은해, 김정선 (한양대학교 컴퓨터공학과)	50
감정 분석을 이용한 동적 기능 제공 프레임워크: 모바일 기기 기반 (short) 황정우, 이윤규, 인호 (고려대학교)	54

요구 공학

Goal 지향 유스케이스 기반의 요구사항 추출에 관한 연구 박보경, 문소영, 김동호, 서채연, 김영철 (홍익대학교)	58
A Study on Research Trends in Requirement Engineering (short) 최수진, 박수용 (서강대학교)	65
UML 기반 결함 트리 자동생성과 런타임 결함식별 김준희, 강신욱, 이은석 (성균관대학교)	73

SW 개발 환경에서의 암묵지 전달/측정을 위한 지식관리 프로세스에 대한 고찰 이성연, 권용찬, 고민석, 정로미, 민상윤 (KAIST)	<u>148</u>
TPI Next 와 CMMI 의 접목을 통한 효율적 테스트 프로세스 개선 접근 방법론 김명호, 배승진, 이충희, 이준형, 민상윤 (KAIST)	<u>156</u>
TMN 관리기능을 이용한 USN 기반 데이터 통합운용관리시스템의 품질평가모델 개발 김성희, 권기현 (경기대학교), 한주연 (한국정보통신기술협회 시험인증연구소)	<u>164</u>

테스팅 1

소프트웨어 역공학을 통한 HeliScope OFP 의 테스트 케이스 생성 이종훈, 이동아, 유준범, 송승화, 김두현 (건국대학교)	<u>172</u>
Pre-Testing 를 위한 M&S 기반 테스트 케이스 추출 연구 (short) 우수정, 손현승, 김우열, 김재승, 김영철 (홍익대학교)	<u>181</u>
지능형로봇 테스트를 위한 상태기반 모델링 유지동, 김대곤, 장영원, 노혜민, 유철중 (전북대학교)	<u>184</u>

SW 프로세스 2

모바일 도메인의 피드백 환경 구축을 위한 상호작용 모델링 방법 신수혜, 박준석, 염근혁 (부산대학교)	<u>191</u>
원자력 해체시설 특성관리 시스템을 위한 CBD 프로세스의 적용 방안 (short 우수논문) 조운형, 박승국, 최윤동, 문제권 (한국원자력연구원)	<u>198</u>
프로젝트의 성공적 수행을 위한 상주감리의 위상 정립 및 향후 발전방향에 관한 연구 (short) 전재영, 조희준, 권호열 (강원대학교)	<u>200</u>
CMMI 기반의 품질보증 프로세스 개선 지침 모델 개발 (short) 김정민, 도성룡, 한혁수 (상명대)	<u>204</u>

테스팅 2

스마트 폰 앱 호환성 테스트를 위한 기존 GUI 테스트 오라클의 한계 이정욱, 김태연, 채홍석 (부산대학교)	<u>208</u>
-----------------------------------------------------------------	------------

Goal 지향 유스케이스 기반의 요구사항 추출에 관한 연구

박보경*, 문소영*, 김동호*, 서채연*, 김영철**

홍익대학교 일반대학원 소프트웨어공학전공
홍익대학교 세종캠퍼스

{bk*, whit2*, dong*, seo*, bob**} @selab.hongik.ac.kr

요약: 정확한 요구사항은 성공적인 소프트웨어 개발에 필요한 중요한 요소이다. 구현 단계에서 발생한 에러 수정은 많은 시간과 노력 및 비용이 발생하므로 초기 요구사항을 정확하게 판단해 요구사항의 오류를 줄여야 한다. 이를 위해 정확한 요구사항 추출에 관한 연구가 계속되고 있다. 하지만 기존의 방법들은 과거 구조적 방법론을 기반으로, 현재의 객체지향 패러다임에 적합하지 않다. 본 논문은 객체지향 패러다임에 적용 가능한 요구사항 추출 기법을 제안한다. 제안한 방법은 Goal 지향의 유스케이스 방법을 이용하여 각각의 유스케이스에서 Goal 을 식별하고, Goal 지향 유스케이스 명세서를 작성하여 유스케이스 별로 요구사항을 추출한다. 이 과정을 통해 보다 정확한 고객 요구사항을 추출할 수 있고, 현재의 객체지향 패러다임에 적용이 가능하다. 사례연구로는 자동차 물품관리 시스템의 한 사례를 사용하였다.

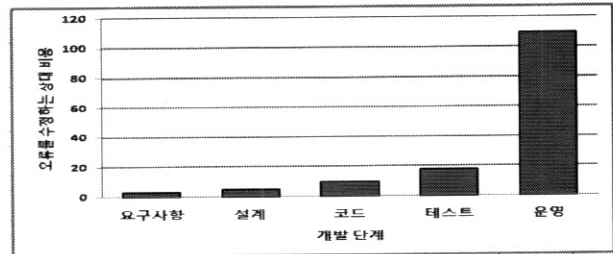
핵심어: 요구사항, 소프트웨어 개발, 가치혁신 요구공학, 객체지향 패러다임, Goal 지향 유스케이스

1. 서론

요구사항 수집 및 분석은 성공적인 소프트웨어 개발을 위한 가장 핵심적인 요소이다. 요구사항 문제로 인해 완료된 작업을 다시 작업하게 되는 일이 빈번히 발생한다. 재작업은 전체 개발비용의 30~50%가 소요되며, 오류사항 에러는 재작업 비용의 70~85%를 차지한다[5]. 그림 1 에서와 같이 프로젝트 후반에 발견되는 오류 수정비용은 각각의 개발단계에서 즉시 수정하는 것보다 더 많은 비용이 든다. 초기 요구사항을 정확하게 판단하고 관리한다면 비용 및 시간을 절약할 수 있다. [5]

이를 위해 요구사항을 측정하여 보다 정확한 요구사항을 추출하고자 하는 연구가 계속되어 왔다. 예를 들면, 기존의 요구사항 측정 방법으로는 Value 기반의 요구공학 방법[2]과 Risk 기반의 요구공학 방법[7] 그리고 Goal 기반의 요구공학 방법[1,8,9]

등이 있다. 이들 방법은 과거 구조적 방법론을 기반으로 하는 개발이다. 하지만 현재에는 널리 사용중인 객체지향 패러다임의 개발 방법에 적용하는 것은 어렵다.



[그림 1] 요구사항 오류 발견 시기에 따른 오류 수정 상대비용[5]

객체지향 패러다임의 개발 방법에 요구사항 추출 방법을 적용하기 위해 Goal 지향 유스케이스 방법을 사용하였다. 이는 개발하고자 하는 시스템의 Goal 에 맞는 요구사항을 추출함으로써, 더 명확하고 정확한 요구사항을 시스템에 반영하고자 하는 것이다. Cockburn[1]이 제시한 Goal 지향의 유스케이스 방법을 적용하여 유스케이스를 도출하고, 수많은 유스케이스들에 대한 Goal 을 식별한다. 우리는 Cockburn 이 제안한 아이디어를 이용하여, 각 Goal 에 맞는 고객 요구사항을 추출한다. 이러한 과정을 통해 현재의 패러다임에 맞는 요구사항을 추출함으로써 보다 정확한 요구사항 추출이 가능할 것이다.

본 논문은 다음과 같이 구성된다. 2 장에서는 관련연구로 가치혁신 요구공학(ViRE: Value-Innovative Requirements Engineering)과 Goal 지향의 유스케이스 방법론(Goal Oriented Use Cases Approach)에 대해서 언급한다. 3 장에서는 2 장에서 소개 한 방법론을 기반으로 하는 고객 요구사항 (Customer Requirements) 추출 과정을 제시한다. 4 장은 사례 연구로 3 장에서 제시한 프로세스를 적용하여 요구사항을 추출하는 방법을 보인다. 마지막으로 5 장에서는 결론 및 향후 연구에 대해서 기술한다.

이 논문은 2011 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업(2011-0004203)과 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구결과임.

2. 관련연구

가치혁신 요구공학은 매우 좋은 아이디어이나 우리가 이 방법을 분석한 결과로 볼 때, 과거 구조적 방법을 기반으로 하는 프로세스이기 때문에 현재의 객체지향 패러다임에 적용하기 어려운 점이 있으리라 본다.

사용자의 요구에 대해 시스템 개발 관점에서 정제한 것을 요구사항(Requirements)이라고 한다. Brian Lawrence 는 요구사항을 “설계를 선택하게 하는 모든 것이다.”라고 정의하였고, IEEE 표준 용어집(1990)에서는 요구사항을 다음과 같이 정의하고 있다.[4]

1. 사용자가 문제를 해결하거나 목적을 달성하기 위하여 사용자가 필요로 하는 조건이나 역량
2. 시스템 또는 시스템 컴포넌트가 계약, 표준, 명세 또는 정규화된 문서를 충족하기 위하여 반드시 충족시켜야 하는 조건 또는 역량
3. 1,2항과 같은 조건과 역량을 표현한 문서

이와 같은 정의는 요구사항에 대한 사용자 및 개발자의 관점을 모두 포함하고 있다.

2.1 가치혁신 요구공학

블루오션 전략을 기반으로 하는 가치혁신 요구공학[2,3]은 고객의 기대와 요구를 만족시키는 방법론이다. ERRC(Eliminate, Reduce, Raise, Create) 분석과 공리적 요구사항 검증을 통해 새로운 고객가치를 창출할 수 있다. 고객의 가치를 분석하고, 절차를 명세화하며, 고객가치가 적절하게 반영되었는지 평가하는 반복된 절차를 통해 요구사항을 추출할 수 있다.

가치혁신 요구공학 프로세스는 목표 설정, 새로운 가치 식별, ERRC 분석, 요구사항 재정의, 요구사항 확인의 총 5 단계로 구성되어 있으며, 모든 단계는 적어도 3번 반복해야 한다.

ERRC 결정 매트릭스는 그림 2 와 같이 고객 가치의 우선순위화, 시스템 요소로 요구사항 매핑 그리고 고객 가치를 결정하는데 사용된다. ERRC 결정 매트릭스는 제안된 ERRC 행위를 통해 새로운 시스템 요소들을 재결정하는데 그 목적이 있다.

가치혁신 요구공학에서는 요구사항의 충돌을 찾아내고 검증하기 위한 접근법으로, Axiomatic 방법을 제안하였다. 이 접근법에서는 요구사항과 설계 프로세스 중 고객 도메인과 시스템 도메인을 포함한다. Axiomatic 방법에서 가장 중요한 것은 매핑 프로세스 동안 2 가지는 만족되어야 한다는 것이며, 2 가지 axiom 은 다음과 같다.

- Axiom 1: 기능 요구 사항의 독립성 유지

- Axiom 2: 정보 콘텐츠의 최소화

		SEs (How)					Customer Importance Degree (Ci)		
		SE ₁	SE ₂	SE ₃	SE _j	SE _n	Extremely important	5	
							Very important	4	
							Important	3	
							Little important	2	
							Not important	1	
	CR ₁						CI		
CRs (What)	CR ₂	Correlation					Customer Importance		
	CR ₃						CI _i		
	CR ₄				R _{ij}				
	CR _n								
ERRC Decision (ED _i)	BI					BI _j	Correlation Degree (R _{ij})		
Eliminate X	RC					RC _j	Strong ●	9	
Reduce -		ERRC Decision						Moderate ⊙	3
Raise +	ED					ED _j	Weak ○	1	
Create o									

[그림 2] ERRC 결정 매트릭스[2]

먼저 고객 요구사항이 정의되면 시스템 파라미터가 식별된다. 그리고 고객 도메인과 시스템 도메인 간의 매핑 프로세스는 $CRs = [R] \cdot SEs$ 로 나타낼 수 있다.

ERRC 분석 단계에서 ERRC 결정 매트릭스의 Correlation 은 CR(Customer Requirements)들 간의 충돌과 부적절한 SE(System Elements)를 점검하기 위해 사용된다. 초기에 정리되지 않은 매트릭스에서 요소들의 위치를 변경함으로써 Diagonal Matrix, Triangular Matrix, 및 그 외 Matrix 의 3 가지 형태로 변경할 수 있다. 시스템 요소가 잘못 선정되었거나 충돌을 일으키는 경우에는 CR 과 SE 를 재조정해야 한다.

가치혁신 요구공학은 요구사항을 분석하고 ERRC 분석과 공리적 검증을 통해 새로운 고객의 가치를 찾는데 도움이 된다. 또한 ERRC 결정 매트릭스를 통해 우선순위화 및 가치를 부여하였기 때문에 고객 요구사항을 추출하는데 있어 체계적인 접근이 가능하다.

하지만 과거 구조적 방법론을 기반으로 하는 프로세스이기 때문에 현재의 객체지향 패러다임에 적용하기에는 어려운 점이 있으리라 본다. 또한 시스템 요소가 소프트웨어와 하드웨어 요소를 포함하고 있기 때문에 소프트웨어만을 개발할 때의 기준이 모호하다. 그리고 자세한 요구사항 및 시스템 요소를 수동으로 계산하기 때문에 절차가 복잡하고, ERRC 행렬을 만드는 것 또한 어렵다.

이와 같은 문제점을 극복하기 위해 객체지향 패러다임의 Goal 지향 유스케이스 방법을 적용하고자 한다. Goal 지향의 유스케이스 방법은 수많은 유스케이스들에 대한 Goal 을 식별함으로써, 각각의 Goal 에 적합한 고객 요구사항을 추출한다. 이는 모든 가능한 요구사항 추출보다는 적합한 요구사항을 추출하고자 하는 것이다.

2.2 Goal Oriented Use Case Approach

위에서 언급한 바와 같이 요구사항은 많은 사람들에게 의해 다양한 정의가 내려져 있지만, 일반적인 요구사항의 정의가 불분명하다. 또한 중복된 요구사항을 정의하는 것은 부가적인 의미들을 일관성 있게 적용하기 어렵다.

이 논문에서는 효율적인 요구사항을 기술하기 위해 Cockburn 이 제안한 Goal 지향 요구사항 기술을 채택하여 시나리오와 유스케이스를 기술할 수 있는 형식을 정의한다. 시나리오와 유스케이스에 관한 정의는 다음과 같다[1].

• 시나리오

: 상호작용의 순서는 주요 액터의 목표를 달성하기 위해 특정 조건에서 발생하고, 그 목표에 관한 특정 결과를 가진다. 상호작용은 실행 동작에서 시작하고 목표가 전달되거나 끝날 때까지 계속된다. 그리고 시스템은 상호작용과 관련된 모든 책임을 완료한다. 시나리오에 대한 특징적인 정보는 주요 액터, Goal, 시나리오 발생 시 조건, 시나리오 결과이고 이를 확인할 수 있다.

• 유스케이스

: 시스템과 외부 액터 사이에서 시나리오를 수집하는 것은 주요 액터의 목표가 전달될 수도 있고, 전달되지 못할 수도 있는 방법을 보여주는 것이다. 이를 통해 유스케이스에 대한 특징적인 정보가 있는지 확인할 수 있다. 특징적인 정보는 주요 액터, Goal, 사용된 시나리오를 나타낸다.

시나리오는 발생 조건에 따라 구분되고, 같은 목표를 가지고 있기 때문에 함께 그룹화 된다. 그림 3 은 시나리오의 예를 보여준다. 시나리오의 형식은 짧고 간단하다. 여기서 설명된 상호작용들은 쉽게 검토될 수 있고 필요할 때 확장될 수 있다.

System under discussion	the insurance company
Primary Actor	me, the claimant
Goal	I get paid for my car accident
Condition	Everything is in order
Outcome	Insurance company pays claim
1. Claimant submits claim with substantiating data.	
2. Insurance company verifies claimant owns a valid policy (Failure here probably means goal failure).	
3. Insurance company assigns agent to examine case.	
4. Agent verifies all details are within policy guidelines. (An interaction between the agent and secondary actors)	
5. Insurance company pays claimant. (Implies all preceding goals managed to pass)	

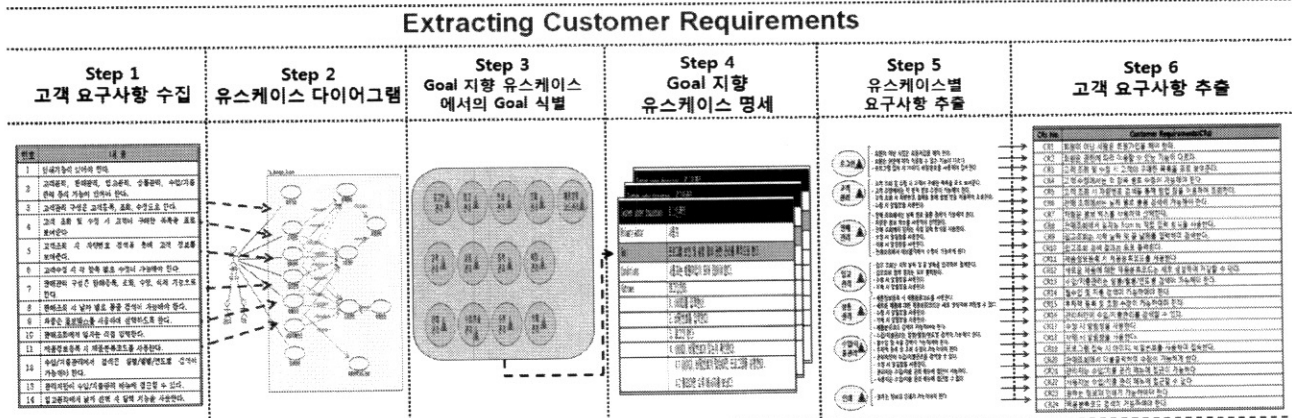
[그림 3] Goal 지향 요구사항 템플릿

3. 고객 요구사항 추출

기존의 가치혁신 요구공학에서는 요구사항을 분석하고자 할 때, ERRC 산출 방식을 통해 새로운 요구사항을 추출한다. 하지만, ERRC 산출 방식은 결과가 정량적인 수치로 표현되지 않는다. 그러므로 산출되는 데이터의 정량화가 어렵다[6].

구조적 방법론은 프로세스 중심의 하향식 방법론이다. 단순한 시스템 개발에는 효과적이지만 데이터 분석과 설계 부분에 취약하고, 개발 단계의 변화에 많은 노력이 소요된다. 따라서 대규모의 복잡한 시스템 개발에는 적합하지 않다. 객체지향 방법론은 시스템 구성요소를 찾아내고 구성요소 간의 연관성을 증명하는 상향식 방법이다. 소프트웨어의 확장과 변화가 쉽고, 재사용이 가능하다. 기존의 가치혁신 요구공학 방법은 구조적 방법론을 기반으로 한 개발자 중심의 개발 방법이다. 따라서 현재의 객체지향 패러다임 개발 방법에 적용하기 어려운 점이 있으리라 본다.

본 논문에서는 객체지향 패러다임의 개발 방법인 유스케이스 방법을 이용한다. Cockburn 의 아이디어를 채택하여, 수많은 유스케이스들에 대한 Goal 을 식별함으로써 각각의 목표에 맞는 고객 요구사항을 추출한다. 그림 4 는 고객 요구사항 추출



[그림 4] 고객 요구사항 추출 과정

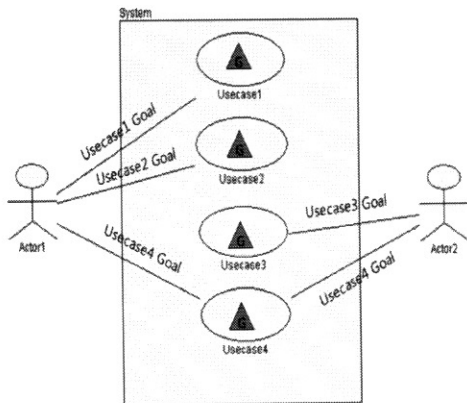
과정을 나타낸 것이다. 고객 요구사항을 총 6 단계로 구성되어 있다.

• Step 1: 고객 요구사항 수집

요구사항 수집은 이해관계자의 요구사항을 정의하고 문서화하는 것이다. 고객이 어떠한 기능을 원하는지 분석하기 위해서는 전체 도메인에 대한 정의와 세부적인 도메인을 식별해야 한다. 그리고 식별된 도메인에 대한 조사계획을 수립한다. 직/간접적인 데이터 추출방법을 통해 방대한 양의 데이터를 수집한다[6,10]. 이러한 정보를 토대로 시스템 요구사항을 정의한다. 또한 기능 별 요구사항을 도출하고 정리한다.

• Step 2: 유스케이스 다이어그램

요구사항 문서를 정의한 후, 유스케이스 다이어그램을 식별하기 위해서 시스템을 확인하고 액터를 식별해야 한다. 또한 행위자들 간의 책임을 확인하고, 유스케이스 간의 관계를 파악해 놓아야 한다. 고객 요구사항 정의 문서를 바탕으로 유스케이스 다이어그램을 도출하는데, 각각의 유스케이스에서 Goal 을 식별한다. 액터와 유스케이스 간의 관계를 고려하여 유스케이스마다 Goal 을 파악한다. 각각의 유스케이스에 포함되는 Goal 은 ▲의 기호로 표시한다. 아래의 그림은 Goal 지향의 유스케이스 다이어그램을 나타낸 것이다.



[그림 5] Goal 지향 유스케이스 다이어그램

• Step 3: Goal 지향 유스케이스에서의 Goal 식별

이 단계에서는 Goal 지향 유스케이스를 각각 도출한다. 2 단계에서 도출한 각각의 Goal 지향 유스케이스에서 Goal 을 추출한다. 예를 들어 “문서 상신” 유스케이스에서는 “결재할 문서를 결재자에게 상신한다.”란 Goal 을 추출할 수 있다. Goal 이 식별된 유스케이스를 추출하여 정리한다.

• Step 4: Goal 지향 유스케이스 명세

3 단계에서 식별한 Goal 정보를 바탕으로 Goal 지향 요구사항 시나리오/유스케이스 명세서를 작성한다. 표 1 에서와 같이 식별한 Goal 정보를 기술하고, 주요 액터, 선행 조건 등도 기술한다.

[표 1] Goal 지향 요구사항 시나리오/유스케이스 명세

System under discussion	전자결재 시스템
Primary Actor	상신자
Goal	결재할 문서를 결재자에게 상신한다.
Condition	상신자는 문서상신을 위한 문서작성을 요청한다.
Outcome	상신자는 결재자에게 최종 상신한다.
	1. 상신자는 문서상신을 위한 문서작성을 요청한다.
	2. 상신자는 상신할 문서의 제목과 내용을 작성한다. (첨부파일이 있을 경우 파일을 첨부한다.)
	3. 상신자는 결재자를 지정하여 요청한다. (결재자가 다수일 경우 결재자를 추가한다.)
	4. 상신자는 문서를 상신한다.
	5. 상신자는 상신 의견을 최종적으로 작성하고 결재자에게 최종 상신한다.

• Step 5: 유스케이스 별 요구사항 추출

유스케이스 별로 식별된 Goal 과 Goal 지향 요구사항 시나리오/유스케이스 명세서를 이용하여 요구사항을 추출한다. 이는 각각의 Goal 에 맞는 요구사항을 유스케이스 별로 추출함으로써 어떠한 요구사항이 어떠한 Goal 에 더 적합한지 판단할 수 있다. 초기에 정의한 요구사항을 각 Goal 과 연관된 유스케이스에 매칭시킴으로써 유스케이스 별로 요구사항을 정리한다.

• Step 6: 고객 요구사항 추출

5 단계에서 추출된 요구사항을 종합하여 고객 요구사항을 도출한다. 추출된 요구사항을 순서대로 고객 요구사항 번호를 지정 하여 유스케이스 별로 추출된 요구사항을 기술한다.

4. 사례 연구

[표 2] 고객 요구사항

번호	내용
1	인쇄기능이 있어야 한다.
2	고객관리, 판매관리, 입고관리, 상품관리, 수입/지출 관리 등의 기능이 있어야 한다.
3	고객관리 구성은 고객등록, 조회, 수정으로 한다.
4	고객 조회 및 수정 시 고객이 구매한 목록을 표로 보여준다.

5	고객조회 시 차량번호 검색을 통해 고객 정보를 보여준다.
6	고객수정 시 각 항목 별로 수정이 가능해야 한다.
7	판매관리 구성은 판매등록, 조회, 수정, 삭제 기능으로 한다.
8	판매조회 시 날짜 별로 물품 검색이 가능해야 한다.
9	차종은 콤보 박스를 사용하여 선택하도록 한다.
10	판매조회에서 일자는 직접 입력한다.
11	제품정보등록 시 제품분류코드를 사용한다.
12	수입/지출관리에서 검색은 일별/월별/연도별 검색이 가능해야 한다.
13	관리자만이 수입/지출관리 메뉴에 접근할 수 있다.
14	입고관리에서 날짜 선택 시 달력 기능을 사용한다.
⋮	⋮

제안한 고객 요구사항 추출 과정을 통해 자동차 물품관리 시스템에 적용하였다. 자동차 물품관리 시스템을 개발하기 위해서는 고객이 어떠한 기능을 원하는지 분석해야 한다. 이를 위해 고객의 요구사항이 무엇인지 정의해야 하는데, 표 2 는 수집된 고객 요구사항을 나타낸다.

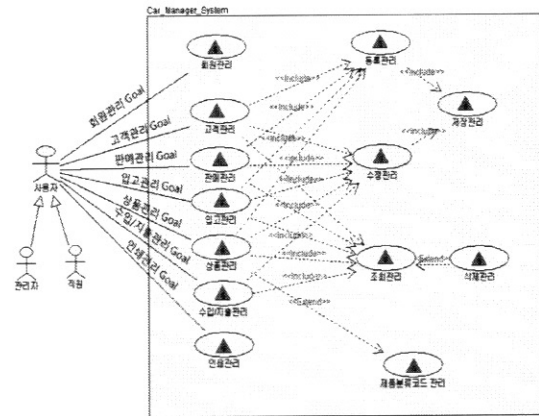
위의 요구사항을 분석하고 정리하여 자동차 물품관리 시스템의 기능을 도출한다. 기능 별 요구사항은 표 3 과 같다.

[표 3] 기능 별 요구사항

기능범주	사용자	기능(유스케이스)
로그인	사용자 관리자	<ul style="list-style-type: none"> 회원 가입(여부 체크) 기능 로그인(아이디/비번체크)기능
고객관리	사용자 관리자	<ul style="list-style-type: none"> 고객등록 기능 고객조회 기능 고객수정 기능 팝업 창 기능 표 출력 기능(구매 목록) 저장 기능
판매관리	사용자 관리자	<ul style="list-style-type: none"> 판매등록 기능 판매조회 기능 판매수정 기능 판매삭제 기능 콤보박스 선택 기능(차종) 날짜 별 물품 선택 기능 저장 기능 날짜 직접 입력
입고관리	사용자 관리자	<ul style="list-style-type: none"> 입고등록 기능 입고조회 기능 입고수정 기능 입고삭제 기능 표 출력 기능(입고 검색 결과) 달력(날짜 선택)기능 날짜 직접 입력 저장 기능
상품관리	사용자 관리자	<ul style="list-style-type: none"> 상품등록 기능 상품조회 기능 상품수정 기능 상품삭제 기능 제품분류코드 생성 기능 저장 기능
수입/지출	관리자	<ul style="list-style-type: none"> 월수입/지출, 검색 기능 월수의 검색 기능

관리		<ul style="list-style-type: none"> 투자액 등록 기능 투자액 조회 기능 투자액 수정 기능 일별/월별/연도별 검색 기능 저장 기능 달력(날짜 선택) 기능
파일	사용자 관리자	<ul style="list-style-type: none"> 인쇄기능

기능 별 요구사항을 도출한 후, 유스케이스 다이어그램을 작성한다. 요구사항 정의 문서 와 기능 별 요구사항 표를 참고하여 유스케이스를 추출한다. 또한 유스케이스 추출과 함께 각 유스케이스 별로 Goal 을 분석하고 식별한다. 그림 6 은 자동차 물품관리 시스템의 Goal 지향 유스케이스 다이어그램을 나타낸 것이다.



[그림 6] 자동차 물품관리 시스템의 Goal 지향 유스케이스 다이어그램

Goal 지향 유스케이스 다이어그램에서 Goal 을 식별하고 정리한다. 이를 표 4 와 같이 유스케이스와 Goal 로 기술 한다.

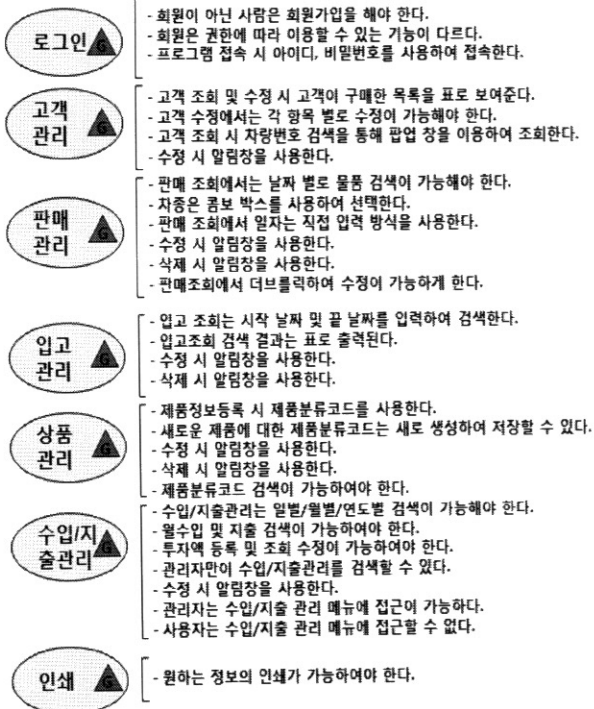
[표 4] 각각의 유스케이스 별로 식별된 Goal

유스케이스	Goal(목표)
로그인 관리	프로그램 보안 및 실행 등의 권한 관리를 목적으로 한다.
고객관리	신규 고객 등록, 고객 조회 및 고객 정보 수정 등의 관리를 목적으로 한다.
판매관리	판매된 물품의 등록, 조회, 수정 및 삭제 등의 관리를 목적으로 한다.
상품관리	상품등록, 조회 및 잘못된 상품 정보 수정, 삭제 등의 관리를 목적으로 한다.
수입/지출관리	투자액 등록/조회/수정 및 월수입/지출 상황 검색 등의 관리를 목적으로 한다.
입고관리	입고된 물품의 등록 조회 및 잘못된 상품의 수정, 삭제 등에 관한 관리를 목적으로 한다.
제품분류코드 관리	제품분류코드를 통해 물품을 분류하여 관리하는 것을 목적으로 한다.
인쇄관리	물품 정보의 인쇄를 목적으로 한다.

유스케이스에서 식별된 Goal 을 도출한 후, Goal 지향의 유스케이스 명세서를 작성한다. 표 4 에서 식별된 Goal 과 액터 및 선행 조건 등의 정보를 기술한다. 표 5 는 Goal 지향 유스케이스 명세서 중 고객관리에 관한 명세서를 나타낸 것이다.

[표 5] 고객관리 유스케이스 명세서

System under discussion	고객관리
Primary Actor	사용자
Goal	신규고객등록, 고객조회 및 고객정보수정 등의 관리를 목적으로 한다.
Conditions	1. 고객이 프로그램에 등록되어있지 않다. 2. 고객정보 및 구매목록을 확인한다. 3. 고객정보를 변경한다.
Outcome	고객등록을 한다. 1. 고객등록의 각 항목을 입력한다. 2. 저장한다. 고객조회를 한다. 1. 조회하고자 하는 고객의 차량번호를 입력한다. 2. 차량번호를 검색한다/ 3. 조회하고자 하는 고객정보 및 구매목록을 확인한다. 고객수정을 한다. 1. 변경하고자 하는 고객정보를 변경한다. 2. 고객정보를 확인한다.



[그림 7] Goal 지향 유스케이스 요구사항 추출

유스케이스 별로 식별된 Goal 과 유스케이스 명세서를 사용하여 요구사항을 추출한다. 초기에 정의한 요구사항을 각 Goal 과 연관된 유스케이스에 매칭시켜 요구사항을 정리한다. 그림 7 은 Goal 지향 유스케이스에 적합한 요구사항을 나열한 것이다.

각각의 Goal 에 맞는 요구사항을 각 Goal 과 연관된 유스케이스에 매칭시킨다. 유스케이스 별로 요구사항을 정리하여 표 6 과 같이 도출할 수 있다.

[표 6] 추출된 고객 요구사항

CRs No.	Customer Requirements(CRs)
CR1	회원이 아닌 사람은 회원가입을 해야 한다.
CR2	회원은 권한에 따라 이용할 수 있는 기능이 다르다.
CR3	고객 조회 및 수정 시 고객이 구매한 목록을 표로 보여준다.
CR4	고객 수정에서는 각 항목 별로 수정이 가능해야 한다.
CR5	고객 조회 시 차량번호 검색을 통해 팝업 창을 이용하여 조회한다.
CR6	판매 조회에서는 날짜 별로 물품 검색이 가능해야 한다.
CR7	차종은 콤보 박스를 사용하여 선택한다.
CR8	판매조회에서 일자는 from to 직접 입력 방식을 사용한다.
CR9	입고조회는 시작 날짜 및 끝 날짜를 입력하여 검색한다.
CR10	입고조회 검색 결과는 표로 출력된다.
CR11	제품정보등록 시 제품분류코드를 사용한다.
CR12	새로운 제품에 대한 제품분류코드는 새로 생성하여 저장할 수 있다.
CR13	수입/지출관리는 일별/월별/연도별 검색이 가능해야 한다.
CR14	월수입 및 지출 검색이 가능하여야 한다.
CR15	투자액 등록 및 조회 수정이 가능하여야 한다.
CR16	관리자만이 수입/지출관리를 검색할 수 있다.
CR17	수정 시 알림창을 사용한다.
CR18	삭제 시 알림창을 사용한다.
CR19	프로그램 접속 시 아이디, 비밀번호를 사용하여 접속한다.
CR20	판매조회에서 더블클릭하여 수정이 가능하게 한다.
CR21	관리자는 수입/지출 관리 메뉴에 접근이 가능하다.
CR22	사용자는 수입/지출 관리 메뉴에 접근할 수 없다.
CR23	원하는 정보의 인쇄가 가능하여야 한다.
CR24	제품분류코드 검색이 가능하여야 한다.

5. 결론

본 논문에서는 각각의 Goal 에 적합한 고객 요구사항을 추출하여 시스템에 반영하기 위한 방법을 정의하였다. Cockburn 이 제시한 Goal 지향 유스케이스 방법론을 고객 요구사항 추출 과정에 적용하였고, 추출된 유스케이스에 대한 Goal 을 식별하였다. 또한 각각의 유스케이스에서 Goal 을 식별함으로써, 각 Goal 에 적합한 고객 요구사항 추출이 가능하였다.

기존의 구조적 방법론을 기반으로 하는 개발 방법론에 비해 본 논문에서 제안한 방법은 현재의 객체지향 패러다임을 기반으로 하고 있다. 또한 요구사항 추출 과정에 Goal 지향의 유스케이스 방법론을 적용함으로써 보다 정확한 요구사항의 추출이 가능할 것이다.

하지만 본 논문에서 제시한 방법은 사용자의 요구가 아닌 고객의 요구만을 고려하여 추출하였기 때문에 보다 정확한 요구사항이라고 보기 힘들다. 또한 본 논문에서는 단순히 요구사항을 추출하는

과정만을 제시하였기 때문에 어떠한 요구사항이 더 중요한지 측정하고 판단하기 어렵다. 앞으로의 연구에서 이를 보완해야 할 것이다. 차후 연구에서는 고객의 요구사항과 더불어 사용자의 요구사항도 반영할 수 있는 방법에 대해서 연구할 것이다. 또한 기존의 고객 요구사항과 사용자 요구사항을 포함하는 풍부한 요구사항 도출하고, 요구사항 우선순위 및 중요도 측정에 대해서도 연구하고자 한다.

참고문헌

- [1] Alistar Cockburn, "Goals and Use Cases", J, Object-Oriented Programming, Vol. 10, No. 7, Sept, 1997, pp. 35-40
- [2] 김상수, "Value-Innovative Requirements Engineering", 고려대학교, 2008
- [3] 박용식, 김상수, 인호, "개선된 BORE 프로세스를 적용한 모바일 디바이스 개발", 한국정보처리학회 추계학술발표대회, 제 13 권, 제 2 호, 2006
- [4] 안성빈, "유스케이스 패러다임 상에서 요구사항 기반 테스트", 홍익대학교 대학원, 2011
- [5] Karl E. Wiegers, "Software Requirements", 정보문화사, 2003
- [6] 안성빈, 김동호, 서채연, 김영철, 정지홍, "사용자 행위 분석 기반 요구추출 방법에 대한 연구", 소프트웨어공학회, Vol.12, No.1, 2010
- [7] 송미경, "소프트웨어 리스크 카테고리 항목을 이용한 리스크 값 측정 및 테스트 연계 방안 연구", 이화여자대학교 대학원, 2010
- [8] Axel van Lamsweerde, "Goal-Oriented Requirements: A Guided Tour", Invited mini-tutorial paper, appeared in
- [9] Lee, Y, Ryoo, J, Stach, J. F, & Park, E. K. (2000). "Composing requirements for goal-oriented system behaviors using an extended form of use case." Proceedings 3rd IEEE Symposium on Application Specific Systems and Software Engineering Technology
- [10] 안성빈, 박보경, 김동호, 서채연, 김영철, "사용자 행위 분석 기반 비기능적 요구추출에 대한 연구", KSEJW, Vol.8, No.1, 2010