

객체지향모델링 과정에서 계층적 유즈케이스(Use Case) 분석을 통한 클래스 도출 및 정의

이재우^{1*}

¹경북대학 컴퓨터정보과

Design of Class Model Using Hierarchical Use Case Analysis for Object-Oriented Modeling

Jae-Woo Lee^{1*}

¹Division of Computer Science & Information, Kyungbok College

요 약 유즈케이스 다이어그램은 사용자와 시스템간의 상호작용을 정의하는데 사용되어 진다. 소프트웨어 개발에 있어서 사용자의 요구사항을 도출하고 정의하는 것은 가장 중요한 일중에 하나이다. 본 논문에서는 클래스 추출 및 정의를 위하여 계층화 된 유즈케이스 모델링 과정을 제안하였다. 본 논문에서 제안하는 모델링 과정에서는 먼저, 여러 계층에 걸쳐서 유즈케이스 다이어그램을 정의하고 다음단계로서 각 계층별 클래스를 추출하고자 하였다. 즉, 소프트웨어 개발에 있어서의 사용자 요구사항 정의단계에서 도출된 유즈케이스 다이어그램과 클래스 다이어그램들의 상호작용을 비교 검증함으로써 보다 효율적인 객체지향모델링 과정을 수행할 수 있게 된다.

Abstract Use case diagram is used for defining inter-action between users and systems in object-oriented modeling. It is very important to defining users' requirements for efficient software development. In this paper, we propose a object-oriented modeling process using hierarchical use case analysis for designing class model. First, We define many use case diagrams by several hierarchical modeling level. And next, we can also design class model using the use case diagrams. Our proposed modeling process provides interaction between use case model and class model. That can make us to check the modeling process during the software development. Using the proposed object-oriented modeling we can develop software based on users' requirements. It is very useful for class modeling.

Key Words : Use Case, Class, UML Diagram

1. 서론

객체지향 모델링 과정은 사용자와 시스템간의 상호작용을 기술하는 유즈케이스(Use Case) 다이어그램으로 시작한다. 이는 효율적인 소프트웨어를 개발하기 위해서는 사용자의 요구사항 도출이 가장 중요하기 때문이다. 최근 정보기술의 눈부신 발전에 따라 많은 기업이나 조직의 정보화 과정에서 다양한 정보기술들이 적용되고 있기는 하지만 실제로 소프트웨어 개발에 있어서 사용자의 요구를 정확하게 도출하고 이러한 요구사항을 효율적으로 설계하여 구축하는 일이 가장 어렵고 힘든 작업 중에 하나일 것이다. 이는 정보기술이 아무리 발전한다고 하더라도

다양한 정보제공을 위해서는 표준 모델링 과정을 수행하고 이 결과가 정보모델에 표현이 되어야 하기 때문이다[1,2].

UML(Unified Modeling Language)은 이러한 요청에 의하여 국제적으로 표준화된 모델링언어로 제정되었으며 유즈케이스 다이어그램을 포함하여 객체를 도출하여 분석하고 이를 구축해가는 가장 효율적인 모델링 도구 중에 하나이다. 특히 유즈케이스 다이어그램은 개발하고자 하는 소프트웨어 사용자들의 요구사항 분석을 위한 객체지향 개발과정의 시작단계에서 기술되는데 소프트웨어 개발에 있어서 가장 필수적이고도 중요한 과정이다. 이 과정은 먼저 시스템을 사용할 액터(Actor)들을 모두 도출

*교신저자 : 이재우(jwlee@kyungbok.ac.kr)

접수일 09년 09월 24일

수정일 (1차 09년 11월 09일, 2차 09년 12월 03일)

게재확정일 09년 12월 16일

하고 액터와 시스템간의 상호작용인 유즈케이스를 정의하게 되는 것으로 이는 객체도출에 앞서 시스템의 전체적인 구성과 개발범위를 사용자의 요구사항 도출에 의하여 확정하게 되는 가장 중요한 단계이다. 액터와 유즈케이스가 모두 도출되고 나면 구체적인 소프트웨어 개발을 위하여 유즈케이스명세서를 작성하게 된다[1-3].

유즈케이스명세서는 각각 도출된 액터와 유즈케이스에 대한 상세 시나리오를 정의하는 과정인데 사용자 요구사항을 시스템에 적용하기 위하여 사용자들의 업무처리 규칙 및 각종 기업의 비즈니스, 회사 전반에 있어서 정보처리 규정 등 소프트웨어 개발을 위한 시스템정의에 있어서 가장 중요한 부분을 차지하고 있다. 이처럼 개발될 소프트웨어가 기업영역을 잘 지원하기 위해서는 효율적으로 분석, 설계되어야 하며 현실 업무에 따라서 올바르게 구축된 시스템들은 관리가 용이하고 이를 이용하여 좋은 정보를 제공할 수 있는 중요한 정보자산이 된다. 객체지향 개발과정은 이처럼 현실세계의 현상이나 복잡한 업무처리 규칙 등을 개념적으로 이해하기 쉽게 표현하고자 모델링을 수행하는 과정이다[4].

따라서 이러한 효율적이고 정확한 모델링을 위해서는 유즈케이스 다이어그램에 대한 보다 상세하고 구체적인 기술이 필요한데 본 논문에서는 소프트웨어 개발에 있어서 보다 체계적이고 정확화 된 사용자 요구분석을 위하여 계층화 된 유즈케이스 추출기법을 제안하고자 한다. 즉, 유즈케이스(Use Case)를 정의하는 과정에서 먼저 개략적인 시스템 범위를 기술하는 유즈케이스다이어그램을 작성하고 이를 다시 세분화하여 계층화, 상세화 시켜 나가는 과정을 제안하였다.

본 논문의 구성은 2장에서는 객체지향 모델링 관련연구를 살펴보고 3장에서는 사용자 요구사항에 대한 정확한 정의를 위하여 계층화 된 유즈케이스 정의 과정을 제안하고 이를 통한 유즈케이스명세서 작성 수준 등 제안된 모델링의 특징 및 장점을 살펴본다.

4장에서는 계층적 유즈케이스 다이어그램을 기반으로 적용사례를 통하여 객체를 추출하고 유즈케이스 모델과 비교해 봄으로써 제안모델의 활용을 평가한다.

마지막으로 5장 결론 부분에서 객체지향모델링 과제를 설정하고 향후 연구과제를 살펴본다.

2. 관련 연구

2.1 유즈케이스 다이어그램(Use Case Diagram)

현실세계(Real World)는 매우 복잡하고 다양하기 때문

에 소프트웨어 개발에 있어서 사용자의 요구를 정확하게 반영하거나 직접적인 관찰을 통하여 명확하게 표현하기란 그리 쉬운 일은 아닐 것이다. 하지만 어떤 지식이나 정보를 적절한 추론절차나 상세화 도구를 통하여 표현할 수만 있다면 이를 통하여 우리는 각종 다양한 소프트웨어의 구현을 가능하게 할 수 있을 것이다. 현실세계는 우리가 볼 수 있는 어떤 객체들과 그들 간의 관계성에 의하여 규명될 수 있고, 그러한 객체들의 행위(Behavior)를 중심으로 표현하고 명세화 할 수 있다. 이렇게 눈에 잘 보이지 않는 현상이나 실체를 어떤 절차에 의하여 명세화한 것을 모델이라고 한다. 객체모델링은 현실세계를 이해하기 위하여 표현하는 절차나 과정을 말한다[3,4].

유즈케이스 다이어그램은 액터와 시스템 사이의 상호작용을 정의하는 것으로서 가장 먼저 액터를 추출하고 정의하게 된다. 액터는 시스템에 동작을 주어 상호작용을 하게 하는 것으로서 시스템과 메시지를 주고받는다. 이때 액터와 시스템간의 상호작용을 담당하는 것이 유즈케이스(Use Case)이다. 따라서 유즈케이스는 액터가 요청하는 시스템과의 상호작용을 정확하게 수행 할 수 있어야 하며 이를 위해서 각 유즈케이스에 대한 상세한 시나리오인 사용자례명세서가 기술되어야 한다[1-3].

본 논문에서는 이러한 유즈케이스 도출 및 정의에 정확성을 높이기 위하여 유즈케이스를 계층화 시켜 정의함으로써 객체지향 개발과정 진행에 따라 점차적으로 가시화 시키고 반복적인 과정을 통하여 유즈케이스 다이어그램을 정의하고자 하였다. 먼저 최상위 계층구조로서 유즈케이스들을 도출하고 각 유즈케이스별로 다음 하위계층으로 세분화하여 최종단계까지 분할해 나감으로써 궁극적으로는 최하위단계의 유즈케이스 다이어그램을 통하여 객체를 도출하고자 하였다. 객체지향 모델링이란 복잡한 현실세계를 객체와 객체간의 구조로 변환하여 모델링 도구를 사용하여 표현한 것을 말하며 현실세계의 모습을 쉽게 이해할 수 있도록 지원한다.

2.2 관련 연구

현실세계의 지식 및 정보 모델링과 관련하여 많은 연구들이 있어 왔는데 지식 및 정보의 구조를 정의하고 이를 현실세계와 유사하게 객체로 표현함으로써 보다 효율적으로 소프트웨어 설계 및 개발에 적용하고 하였다.

소프트웨어 설계 및 개발에 있어 설계방법 표준화를 위해 UML 도입에 영향을 주는 요인에 관한 연구가 있었는데 소프트웨어 설계방법의 표준화로 UML(Unified Modeling Language)을 사용할 때 영향을 미칠 수 있는 객체지향언어의 사용정도, UML 도구 공급자의 지원, 조직원의 신기술 학습의욕, 최고경영자의 지원, 프로젝트의

규모 등에 대한 연구가 있었다[5].

기업체의 비즈니스 시스템 구축을 위한 UML 기반의 분석 모델링의 문제점을 분석한 연구가 있었는데 이 연구에서는 사용자 요구분석을 위한 유즈케이스 모델링의 절차와 분석과정을 살펴보고 UML 기반의 모델링을 적용하여 분석할 경우 고려해야 될 사항들을 연구하였다[6].

국내 대형 SI 업체의 UML(Unified Modeling Language)의 도입과 활용 사례를 소개한 연구가 있었는데 소프트웨어 모델의 기본 목적인 복잡한 소프트웨어를 추상적 수준에서 이해할 수 있게 표현하기 위하여 사용자 요구의 정확한 반영, 개발팀원간의 업무분담, 개발비의 추정, 유지보수에 유용하게 쓰일 수 있도록 제시하였다.[7]

UML은 각종 과제나 소프트웨어 개발 프로젝트를 수행할 때 청사진을 제공하는 대표적인 모델링 언어로 활용되어 오고 있고 UML을 사용한 소프트웨어 모델링의 새로운 요구 사항에 따른 UML 2.0 기반 객체지향 모델링 프로세스 및 지침에 대한 연구가 있었다[8].

UML 모델링 도구에 사용자 정의형 패턴을 쉽게 적용할 수 있도록 하는 기능을 설계하여 구현한 결과를 제시한 연구도 있었는데 소프트웨어 디자인 패턴(Design Pattern)은 좋은 설계나 아키텍처의 재사용을 도와주며, 이미 만든 시스템의 유지보수나 문서화도 개선해 준다. 사용자가 필요시 디자인 패턴을 쉽고 편하게 UML 모델링 도구에 적용할 수 있도록 소프트웨어의 생산성 및 품질 개선을 제시하였다[9].

유즈케이스 다이어그램에 대한 유즈케이스 명세서 작성을 위한 연구도 있었는데 유즈케이스 기술서 작성 및 편집을 체계적으로 하기 위한 도구를 제시하였고 또한, 시나리오의 기술에 있어서 1컬럼 방식과 2컬럼 방식 모두를 지원하도록 하였으며, 시나리오 각 단계별 작성과 편집의 처리에 있어서 자동화된 기기를 지원하여 작성의 편의성을 높여 주고자 하였다[10].

유즈케이스 모델링은 궁극적으로는 사용자 요구사항에 대한 도출을 통하여 정확한 소프트웨어를 개발하고자 함에 있는데 이러한 관점에서 유즈케이스 모델링을 위한 요구사항 중복 및 불일치 분석에 대한 연구가 있었다. 이 연구는 요구사항의 중복관계 및 불일치를 도출할 수 있을 뿐만 아니라 다중 사용자관점의 부분적인 요구사항들을 일관성을 갖는 통합된 요구사항 모델로 전환함으로써 소프트웨어의 행위와 기능을 명확하게 규명할 수 있음을 보여 주었다[11].

이밖에 UML의 클래스다이어그램을 확장하여 객체관계형 데이터베이스 설계를 위한 UML 스테레오타입을 기술하며, 다양한 복합관계 및 복합객체를 지원하기 위한

설계기법을 제안한 연구와 UML을 응용한 소프트웨어 아키텍처를 쉽게 문서화할 수 있는 방법을 제안한 연구, 그리고 소프트웨어 개발 생명주기에서 소프트웨어 생산성 향상을 위한 효율적인 객체지향 모델링 방법을 제안한 연구 등이 있었다[12-14].

3. 계층화된 유즈케이스 모델링

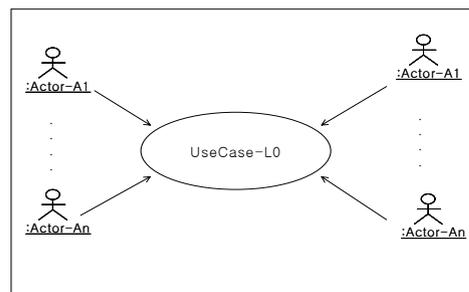
3.1 액터(Actor) 추출

본 논문에서는 사용자 요구분석을 통하여 정보모델링을 수행하고자 유즈케이스 다이어그램을 계층화 하는 과정을 제안하였다. 제안된 방법은 먼저 액터를 추출하기 위하여 최 상위 레벨의 유즈케이스 다이어그램 작성을 제안하고자 하였다. 최 상위 수준의 유즈케이스 다이어그램은 해당 소프트웨어 개발을 위하여 사용자 요구분석을 토대로 한 모든 액터를 추출하고 유즈케이스는 최 상위 수준으로 단 1개로 표현한다.

[정의1] 임의의 유즈케이스는 계층화하여 상세화 수준으로 표현한다. L_i 은 유즈케이스 모델링에서의 계층화로 표현되며 계층화 수준 i 는 유즈케이스에 대한 계층화 레벨을 나타낸다. i 가 0일때 최 상위 수준의 유즈케이스로 표현되며 상세화 됨에 따라 하위레벨로 표현된다. 여기서 i 는 0, 1, 2, ..., n 레벨로 표현할 수 있다.

그림 1에서 보느냐와 같이 추출된 액터는 해당 소프트웨어에서 사용할 모든 종류의 액터를 추출하여 표현되며 최 상위레벨의 유즈케이스는 L_0 로 표현하였다. 이렇게 하는 이유는 개발하고자 하는 소프트웨어에 대한 모든 액터를 우선 추출해 봄으로써 해당 소프트웨어의 사용자 요구분석 대상자에 대한 명확한 정의가 가능해 지기 때문이다.

최 상위수준에서 추출된 액터들은 현 단계에서는 전체적인 구성만을 보여 주지만 유즈케이스를 계층화하여 세분화 하게 되면 보다 구체적으로 시스템의 상호작용을 파악할 수 있게 된다.



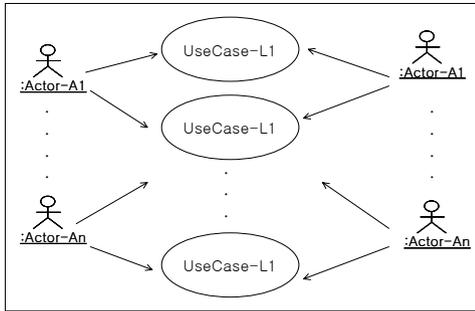
[그림 1] 액터추출을 위한 유즈케이스모델링(L0)

3.2 유즈케이스(Use Case) 계층화

앞 절에서 정의한대로 우선 최 상위 수준의 유즈케이스 모델링이 이루어지고 나면 다음 수준의 유즈케이스 모델링을 위하여 레벨-0수준의 유즈케이스를 상세화하여 레벨-1수준으로 계층화 시키는 과정을 수행한다. 레벨-0 수준의 유즈케이스는 몇 개의 레벨-1수준의 유즈케이스로 세분화 될 수 있으며 이에 따라 액터 역시 상세화 되어 계층화 된 유즈케이스 모델링이 수행 된다.

[정의2] 유즈케이스 레벨-0수준(L0)은 사용자 요구분석에 따라 상세화 되어 레벨-1수준으로 계층화를 반복하여 진행 된다. 즉 유즈케이스 레벨-0수준은 여러 개의 1레벨의 유즈케이스들의 전체 집합으로 표현된다.

유즈케이스 계층화 모델링이 진행됨에 따라 유즈케이스 최상위 수준인 L0는 유즈케이스 레벨-1수준인 L1들로 세분화 된다.



[그림 2] 유즈케이스 계층화 모델링(L1)

이렇게 하여 도출된 레벨-1수준의 유즈케이스들은 다시 레벨-2, 3, 4, ... , n 수준으로 상세화, 계층화하여 유즈케이스 모델링 과정을 반복하여 거치고 나면 더 이상 세분화 할 수 없는 수준까지 계층화 되고 사용자 요구분석 과정은 상세화 될 수 있다.

각 계층화 된 유즈케이스모델링 과정에서는 해당 되는 유즈케이스별로 하위계층의 유즈케이스들을 모델링 하고 이를 반복하는 과정을 수행하게 되는데 이러한 과정은 그림 3과 같이 정의될 수 있다.

```

UseCase(Hierarchical Modeling)
Extract Ai; // Extract Actor
Define L0; // UseCase Modeling Level 0
While(Level<Requirements Level){
  Define Li; // UseCase Modeling Level i
  For j=1 to m
    Define Lij; // Extract UseCase in Level i
  End;
} // End of While
    
```

[그림 3] 유즈케이스 계층화 모델링 과정

3.3 유즈케이스명세서 작성

본 논문에서 제안하는 계층화 된 유즈케이스 모델링 방법은 앞절에서 기술한 대로 레벨-0수준의 유즈케이스 모델링을 시작으로 하여 상세화, 계층화하여 유즈케이스들을 도출해가는 과정으로서 이렇게 함으로써 소프트웨어 개발에 있어서 사용자 요구에 대한 보다 상세하고 정확한 반영이 가능해 진다. 이렇게 하여 최하위 수준까지 유즈케이스 모델링이 수행되고 나면 해당 최하위 수준의 유즈케이스에 대한 명세서를 작성하여 클래스 도출 등 다음 단계의 UML 모델링 과정을 보다 용이하게 수행할 수 있게 된다.

유즈케이스 명세서	
유즈케이스레벨	XX-XX-XX
유즈케이스명	○○○○○
유즈케이스명세	1. 2.
기타요구사항	○○○○○

[그림 4] 유즈케이스 명세서 작성

유즈케이스명세서는 유즈케이스 다이어그램에 대한 상세한 업무처리 규칙 및 알고리즘 등을 기술하는 UML 도구로서 그림 4에서 보는 바와 같이 상세화되어 최하위 수준까지 도출된 유즈케이스에 대하여 기술함으로써 사용자 요구사항을 보다 명확하게 반영할 수 있게 된다.

3.4 클래스(Class) 도출

본 논문에서 제안한 계층화 된 유즈케이스 모델링방법은 사용자 요구에 대한 정확한 반영을 위하여 최하위 수준까지 유즈케이스를 도출하는 과정을 제안하였다. 또한 클래스 추출을 위하여 최하위 수준의 유즈케이스 명세서를 작성하는 과정에서 클래스의 도출이 보다 용이하게 된다.

유즈케이스 모델링 및 객체추출	
유즈케이스명	요구되는 객체/클래스 명
유즈케이스1	객체1, ...
유즈케이스2	
.....	
유즈케이스j	
.....	
유즈케이스n	

[그림 5] 유즈케이스 모델링 및 객체추출

클래스를 도출하기 위해서는 앞절에서 정의된 각 최하위 수준의 유즈케이스별로 요구되는 객체를 도출하고 이를 클래스로 정의함으로써 계층화 된 유즈케이스 모델링 과정을 수행한다. 그림 5에서 보는 바와 같이 각 유즈케이스와 요구되는 객체들을 모두 도출해 봄으로써 클래스 추출과정을 보다 쉽게 수행할 수 있게 된다.

3.5 유즈케이스와 클래스(Class) 모델링 검증

객체지향 모델링 과정에서 가장 중요한 것은 정확한 클래스의 도출일 것이다. 정확한 객체모델링을 위하여 본 논문에서는 최하위 수준까지 계층화 한 유즈케이스모델과 각 유즈케이스별로 추출된 객체/클래스에 대한 상관분석/검증과정을 제안하였다. 각 유즈케이스별로 추출한 객체/클래스를 모두 나열하고 이를 각 유즈케이스들과 비교 검증해 봄으로써 전체 개발하고자 하는 소프트웨어에 대한 요구분석 오류를 줄일 수 있게 된다. 그림 6에서 보는 바와 같이 본 논문에서 제안한 방법을 통하여 각각 정의되고 추출된 유즈케이스들과 객체/클래스를 상호 검증하는 체크리스트를 사용하였다. 이렇게 하는 이유는 각 유즈케이스에서 요구되어 지는 객체들을 일반화 시키고 모든 유즈케이스별로 모든 객체가 도출되었는지를 검증하기 위함이다.

	객체1	객체2	객체i	객체n
유즈케이스1		V	V			
유즈케이스2	V					V
.....		V				
					V	V
유즈케이스i				V		
.....						V
유즈케이스n	V		V			

[그림 6] 유즈케이스 및 클래스 모델링 검증

4. 제안모델의 적용사례 평가

객체지향 소프트웨어 개발과정 적용사례를 통하여 계층적 유즈케이스 모델에 대한 평가를 해 보고자 대학의 학사관리를 분석해 보았다. 요구사항 분석을 통하여 정의된 유즈케이스 모델은 학생관리, 교수관리, 커리큘럼 및 수강관리 등으로 정의될 수 있다.

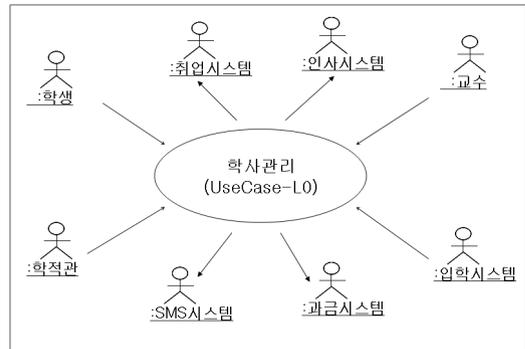
먼저, 유즈케이스를 요구사항에 맞게 상세화 시켜 나가면 그림 7과 같이 계층화 할 수 있는데 본 적용사례에서는 3계층으로 세분화하여 최하위 레벨에서 액터를 추

출해 보았다.

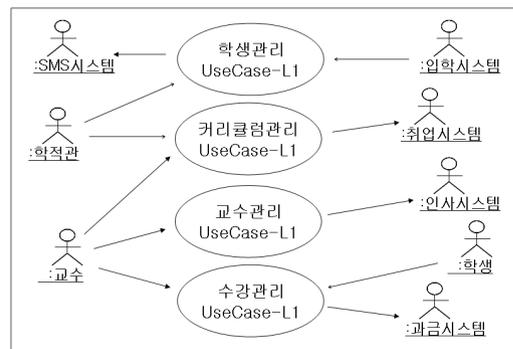
레벨-0	레벨-1	레벨-2	...	액터추출
	학생관리	학적관리 장학관리 진로지도	...	교수, 학적관, SMS시스템, 입학시스템, 취업시스템 ..
학사관리	교수관리	강좌선택 출석관리 강의평가	...	교수, 인사시스템 ..
	커리큘럼	과목편성 강좌관리 과목평가	...	학적관, 교수
	수강관리	수강신청 성적관리	...	학생, 교수, 과금시스템 ..

[그림 7] 유즈케이스 계층화 사례

사용자요구 상세화에 따라 그림 7은 보다 더 계층화 할 수도 있고 이러한 유즈케이스 계층화를 통하여 시스템과 상호작용하는 액터들에 대한 상세화가 가능하다. 유즈케이스 레벨-0와 레벨-1 수준으로 작성된 유즈케이스 다이어그램에 대한 사례는 그림 8과 그림 9와 같이 상세화, 계층화 하여 나타낼 수 있다.



[그림 8] 유즈케이스 모델링(L0) 사례



[그림 9] 유즈케이스 모델링(L1) 사례

다음 과정은 각 유즈케이스별로 상세한 명세서를 작성하게 되고 이 과정에서 사용자요구에 적합한 객체들을 도출하게 된다. 계층화하여 상세한 유즈케이스별로 객체 도출 및 정의를 함으로써 보다 정확한 요구정의에 도움이 된다.

유즈케이스명	요구되는 객체/클래스 명
0.1.1학적관리	학생, 학적, 지도교수 ...
0.1.2장학관리	학생, 장학종별, 장학기준 ...
.....	
0.4.1수강신청	과목, 수강학생, 강의교수 ...
0.4.2성적관리	과목, 과제, 성적처리 ...

[그림 10] 최하위 유즈케이스별 객체추출 사례

본 논문에서 제안하는 계층화 된 유즈케이스 모델의 가장 큰 목적은 사용자 요구정의 과정에서 계층화 된 유즈케이스모델과 추출된 객체/클래스에 대한 정확성을 높이기 위하여 상관분석을 통하여 검증/보완하기 위함이다. 즉, 각 유즈케이스에서 생성되거나 참조되어야 하는 객체/클래스들을 그림 11과 같이 상호 비교해 가면서 체크해봄으로써 사용자요구에 맞는 모든 객체가 도출되었는지를 검증하기 위함이다.

	학생	학적	교수	과목
0.1.1학적관리	V	V	V		
0.1.2장학관리	V	V			
.....					
0.4.1수강신청	V			V	
0.4.2성적관리	V		V	V	

[그림 11] 유즈케이스/객체 상관분석 사례

5. 결론

본 논문에서는 소프트웨어 개발에 있어서 보다 체계적이고 정형화 된 사용자 요구분석을 위하여 계층화 되고 상세화 된 유즈케이스 다이어그램 작성방법을 제안하였다. 유즈케이스(Use Case)를 정의하는 작업은 객체지향 모델링 과정의 시작점으로 객체지향 모델링 과정에 있어서 가장 중요한 작업과정 중에 하나이다. 본 논문에서는 먼저 개략적인 시스템 범위를 기술하는 최 상위 수준의 유즈케이스 다이어그램을 작성하고 이를 다시 세분화하여 계층화, 상세화 시켜나가는 과정을 제안하였다. 이 과정을 통하여 정의된 최하위 수준의 유즈케이스별로 객체/클래스를 추출하고 이를 상호검증 함으로써 개선된 객체

지향 모델링 과정의 수행이 가능하다. 향후과제로는 유즈케이스모델을 기반으로 한 클래스 다이어그램의 정확성 검증을 통한 객체지향 모델링 과정에 대한 체계적인 평가방법에 대하여 많은 연구가 필요할 것이다.

참고문헌

- [1] Leszek A. Maciaszek, "Requirements analysis and system design Developing information systems with UML", Addison-Wesley Pub. Co., 2001.
- [2] Martin Fowler, Kendall Scott, "UML Distilled Second Edition : A Brief Guide to the Standard Object Modeling Language", Addison-Wesley Pub. Co., 2000.
- [3] 박현철, "UML 이해와 실제", 한국소프트웨어연구원, 2005.
- [4] 북병학, 이재우, "시스템통합의 핵심기술", 대청미디어, 2001.
- [5] 서영석, 한영춘, 김기수, "UML 도입에 영향을 주는 요인에 관한 연구 - 국내 S/W 개발업체를 중심으로 -", 한국정보시스템학회 2005년도 춘계학술대회 발표논문집, pp.109-122, 2005.
- [6] 이혜선, 박재년, "비즈니스 시스템을 위한 UML 기반의 분석 모델링의 문제점", 한국정보처리학회 2005년도 제23회 춘계학술발표대회, pp.301-304, 2005.
- [7] 김영기, "UML 활용 사례연구", 한국경영과학회/대한산업공학회 2005년도 춘계공동학술대회 발표논문, pp.294-299, 2005.
- [8] 강현구, 천두완, 김수동, "UML 2.0 기반 객체지향 모델링 프로세스 및 지침", 한국정보과학회 2004년도 가을 학술발표논문집, Vol.31, No.2(2), pp.307-309, 2004.
- [9] 이장우, 이민규, "UML 모델링 도구의 사용자 정의형 패턴 적용을 위한 설계 및 구현", 한국정보과학회 2004년도 가을 학술발표논문집, Vol.31, No.2(2), pp.310-312, 2004.
- [10] 최환복, 김윤호, "유즈케이스 기술서 작성 도구의 개발", 한국해양정보통신학회논문지, v.13, no.2, pp.243-252, 2009.
- [11] 최진재, 황선영, "유즈케이스 모델링을 위한 요구사항 중복 및 불일치 분석", 한국정보과학회논문지:소프트웨어및응용, v.31, no.7, pp.869-882, 2004.
- [12] 김인철, 김영웅, "확장된 UML 클래스 다이어그램을 이용한 객체 관계형 데이터베이스 설계 기법", 한국정보처리학회 2005년도 제24회 추계학술발표대회, pp.91-94, 2005.
- [13] 궁상환, "UML을 응용한 GLORY 소프트웨어 아키텍처

텍처의 표현", 한국산학기술학회논문지, v.10, no.8, pp.1970-1976, 2009.

- [14] 김영규, 양해술, 최형진, "객체지향 환경에서 소프트웨어 생산성 향상을 위한 프레임워크 모델", 한국산학기술학회논문지, v.9, no.6, pp.1678-1689, 2008.

이 재 우(Jae-Woo Lee)

[정회원]



- 1997년 8월 : 서울산업대학교 전자계산학과 (공학석사)
- 2004년 8월 : 고려대학교 컴퓨터학과 (이학박사)
- 1987년 1월 ~ 1999년 2월 : 쌍용정보통신 선임연구원
- 1996년 12월 ~ 현재 : 정보관리기술사
- 1999년 3월 ~ 현재 : 경북대학교 컴퓨터정보과 교수

<관심분야>

소프트웨어공학, 데이터베이스, 프로젝트관리, 정보통신