Journal of The Korea Society of Computer and Information
Vol. 18, No. 6, June 2013
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci.2013.18.6.063

안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론 -역할기반 접근제어를 중심으로-

주 경 수*, 우 정 웅*

An Object-Oriented Analysis and Design Methodology for Secure Database Design -focused on Role Based Access Control-

Kyung-Soo Joo *, Jung-Woong Woo *

요 약

IT의 발전에 따라 다양하고 복잡한 기능을 가진 응용 시스템들이 요구되고 있다. 이러한 응용 시스템들은 대부분 데이터를 효율적으로 관리하기 위해 데이터베이스를 기반으로 구축된다. 그러나 응용 시스템 개발을 위하여 사용되고 있는 대부분의 객체지향 분석·설계 방법론들은 데이터베이스와의 상호 연관성을 명확하게 제공하지 못하고 있다.

보안과 관련된 요구사항들이 증가되면서 보안에 대한 중요성 역시 점차 증가되고 있다. 하지만 이러한 보안은 대부분 개발 마지막 과정에서 고려되기 때문에 요구사항 분석부터 구현에 이르기까지 시스템 개발 전 주기에 따른 일관된 보안 적용은 어려운 실정이다.

따라서 본 논문에서는 요구사항 분석부터 구현에 이르기까지, 보안이 강조된 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'을 제안한다. 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'은 대부분의 객체지향 분석·설계 방법론들이 제시하지 못했던 데이터베이스와의 상호 연관성을 제공할 뿐만 아니라, 보안이 강조된 모델링 언어인 UMLsec을 사용하여 보안이 데이터베이스 설계에 반영토록 하였다. 아울러 보안에 따른 구현을 위하여 관계형 데이터베이스의 역할기반 접근제어(RBAC; Role Based Access Control)를 사용한다.

▸ Keywords : 객체지향 분석·설계, 데이터베이스, RBAC, 보안, UMLsec

Abstract

In accordance with the advancement of IT, application systems with various and complex functions are being required. Such application systems are typically built based on database in

•제1저자 : 주경수 •교신저자 : 우정웅
•투 고 일 : 2013. 5. 5, 심사일 : 2013. 5. 31, 게재확정일 : 2013. 6. 10.
* 순천향대학교 컴퓨터소프트웨어공학과(Dept. of Computer Software Engineering, SoonChunHyang University)

order to manage data efficiently. But most object-oriented analysis·design methodologies for developing web application systems have not been providing interconnections with the database.

Since the requirements regarding the security issues increased, the importance of security has become emphasized. However, since the security is usually considered at the last step of development, it is difficult to apply the security during the whole process of system development, from the requirement analysis to implementation.

Therefore, this paper suggests an object-oriented analysis and design methodology for secure database design from the requirement analysis to implementation. This object-oriented analysis and design methodology for secure database design offers correlations with database that most existing object-oriented analysis and design methodologies could not provide. It also uses UMLsec, the modeling language, to apply security into database design. In addition, in order to implement security, RBAC (Role Based Access Control) of relational database is used.

▸ Keywords : Object-Oriented Analysis and Design, Database, RBAC, Security, UMLsec

I. 서 론

IT의 발전에 따라 다양하고 복잡한 기능을 가진 응용 시스템들이 요구되고 있다. 이러한 응용 시스템들은 대부분 데이터를 효율적으로 관리하기 위해 데이터베이스를 기반으로 구축된다[1,2,3].

보안과 관련된 요구사항들이 증가되면서 보안에 대한 중요성 역시 점차 증가되고 있다. 따라서 모델링 과정부터 보안 대책을 도출하고 이를 시행하는 것이 필수이다[1,4,5]. 이를 위하여, 데이터베이스는 역할기반 접근제어를 통해 보안 방안을 지원하고 있지만, 이러한 기술들이 대부분 분석·설계의 결과로 사용된 것이 아니기 때문에 일관성이 없어, 보안에 취약한 데이터베이스 응용 시스템으로 개발될 가능성이 매우 높다[1,6,7,8].

이에 따라 본 논문에서는 객체지향 분석·설계 방법론 중의 하나인 CBD(Component Based Development) 방법론을 기반으로, 요구사항 분석부터 구현에 이르기까지, 시스템 개발 전 주기에 걸쳐 보안에 대한 일관성을 제공하는, 보안이 강조된 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'을 제안한다. 아울러 보안에 대한 구현은 데이터베이스의 역할기반 접근제어를 이용한다.

본 논문의 구성은 다음과 같다. 2장에서는 제안한 방법론의 이해를 돕기 위한 관련 연구들을 소개하고, 3장에서는 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'을 설명한다. 그리고 마지막 4장에서는 제안한 객체지향 분석·설계 방법론의 평가 및 결론을 제시한다.

II. 관련연구

1. 객체지향 분석·설계 방법론

객체지향 분석·설계 방법론 중 대표적인 UP(Unifi-ed-Process)의 특징은 유스케이스 기반, 아키텍처 중심, 반복 및 점증적이며, 도메인 모델, 유스케이스 모델, 분석 모델, 설계 모델 그리고 구현 모델 등으로 작성된다는 것이다[9].

또한 CBD 방법론은 컴포넌트를 기반으로 소프트웨어 시스템을 개발함으로써 사용자의 요구사항 변화에 신속하고 유연하게 대처하고자 하는 것을 목표로 한다[10].

한편 기존의 객체지향 분석·설계 방법론으로 도출된 개념적 모델은 클래스 다이어그램을 바탕으로 객체지향 프로그래밍 코드를 생성할 수 있지만, 보안에 대한 일관된 분석·설계 방법론은 제시하지 못하고 있다[8].

2. UMLsec을 이용한 보안 유스케이스 모델링

UMLsec은 UML(Unified Modeling Language)에서 보안과 관련된 정보를 통합하여 확장된 모델링 언어이다. 아울러 사용자 데이터의 기밀성, 무결성 등의 보안 요구사항을 분석·설계에 반영하는 표준화된 기호를 제공하고 설계의 검증을 지원하며, 프로파일(Profile)의 속성 및 제약에 대한 명세를 통해 정의된다.

보안과 관련한 분석·설계 방법으로는, 기존의 객체지향 분석·설계 방법론과 보안 요구사항을 통합한 UML 기반의 개발 방법론이 제시되었다[5]. 이 연구에서는 보안에 대해 확장된 UMLsec을 이용해서 보안이 중요한 응용 시스템 개발을 위한 일관된 객체지향 분석·설계 방법론을 제시하고는 있지만, 관계

형 데이터베이스와의 상호 연관성은 제공하지 못하고 있다.

3. 관계형 데이터베이스 설계 방법론

대표적인 관계형 데이터베이스 설계 방법론으로는 정보공학방법론(Information Engineering Methodology)이 있으며, 정보공학방법론의 특징은 업무영역을 데이터 중심으로 정보시스템을 구축케 하며, 데이터 중심의 업무 절차 및 환경 변화에 유연한 장점을 가지고 있는 안정적인 방법론 중의 하나이다[11]. 그러나 정보공학방법론은 기존의 객체지향 분석·설계 방법론과 상호 연관성을 제공하지 못하고 있다. 아울러 데이터베이스 시스템은 보안과 관련하여 역할기반 접근제어 기술들을 지원하고 있으나, 정보공학방법론은 설계 초기 단계에서 이를 고려하고 있지 않기 때문에 개발과정 전체에 걸친 보안에 대한 일관성을 제공하지 못하고 있다.

III. 안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론

본 논문에서 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'은 그림 1과 같이 요구사항 분석 단계에서 비기능적 요구사항 중 하나인 보안에 대한 정의를 추가하였으며, 추가된 요구사항은 UMLsec을 이용하여 정의하였다. 아울러 시스템 분석 및 설계 단계에서도 UMLsec을 이용하여 보안이 강조된 분석·설계를 표현하였다. 또한 마지막 구현 단계에서는 분석·설계의 결과를 바탕으로, 보안에 대한 요구사항을 SQL의 DDL(Data Definition Language)과 DCL(Data Control Language)구문을 통해 역할기반 접근제어로 구현하였다. 한편 요구사항 중 기능적 요구사항 분석과 시스템 분석 및 설계는 기존의 CBD 방법론을 적용하여 수행하였다. 아울러 시스템 분석 및 설계과정 중 데이터베이스의 설계는 O-R(Object-Relational) 매핑을 통해 수행하였다.

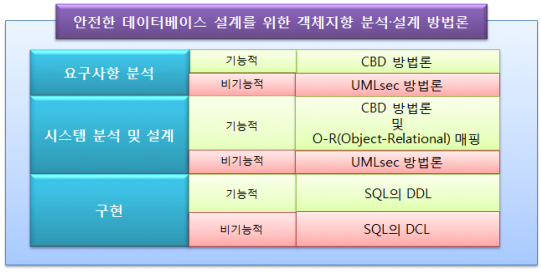

그림 1. 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'의 과정
Fig. 1 Process of Object-oriented analysis and design methodology for secure database design proposed

1. 요구사항 분석

1.1 요구사항 리스트 작성

요구사항 정의는 사용자들이 소프트웨어에 기대하는 기능 및 비기능적 요구를 도출하고 검증하는 활동을 뜻한다.[1,12]. 표 1은 비기능적 요구사항 중 보안 요구사항 정의가 포함되어 있는, 온라인 뱅킹 시스템의 일부분에 대한 요구사항 리스트이다.

표 1. '온라인 뱅킹 시스템'을 위한 요구사항 리스트
Table 1. Requirement list for On-line banking system

1. 사용자는 조회 서비스를 이용할 수 있다.
2. 조회 서비스는 잔액 확인, 거래 목록 확인, 지난 기록 확인 및 다운로드 기능이 있다.
3. 사용자는 요금 지불 서비스를 이용할 수 있으며, 각종 세금을 납부하는 기능이다.
4. 사용자는 거래 서비스를 이용할 수 있다.
5. 거래 서비스는 자금 이체와 같은 기능이 포함된 기능 이다.
6. 관리자는 관리 기능을 통해 시스템의 전체적인 접근권한을 가지고 있으며, 또한 새로운 계좌에 대한 생성 및 삭제, 잔액 수정, 거래 취소, 사용자 등급을 설정할 수 있다.
7. 특정 사용자에 대한 시스템 사용권한을 설정할 수 있다.
8. 해당 시스템을 사용하기 위해서는 로그인이 필요하다.
9. 데이터 관리 및 보호를 위한 기능이 필요하다.

표 2는 표 1의 내용 중 보안 요구사항만을 정리한 내용이며, 표 2의 1번은 관리자 권한에 대한, 2번은 인증에 대한, 3번은 인가에 대한 보안 요구사항이다. 그리고 4번은 비밀보장 및 데이터 무결성에 해당하는 보안 요구사항이다.

표 2. 보안 요구사항 정의
Table 2. Defining security requirements

유형	설 명
보안	1. 관리자는 관리 기능을 통해 시스템의 전체적인 접근권한을 가지고 있으며, 또한 새로운 계좌에 대한 생성 및 삭제, 잔액 수정, 거래 취소, 사용자 등급을 설정할 수 있다. 2. 해당 시스템을 사용하기 위해서는 로그인이 필요하다. 3. 관리자는 특정 사용자에 대한 시스템 사용권한을 설정 할 수 있다. 4. 데이터 관리 및 보호를 위한 기능이 필요하다.

1.2 유스케이스 작성

유스케이스는 시스템이 어떤 일을 수행하기 위해 거쳐야 하는 단계들을 말하며, 또한 새로 만들 시스템이나 소프트웨어 변경사항에 대한 요구사항을 찾아내는 방법이다[1,12].

표 1에서 정의된 사용자 요구사항 리스트를 기반으로, 유스케이스를 작성한다. 다만 보안 요구사항이 있는 유스케이스

의 경우에는 UMLsec 방법론에 따라 유스케이스를 확장해야 한다[5]. 표 3은 온라인 뱅킹 시스템에 대한 유스케이스 목록의 일부분이며, 표 4는 보안을 위해 UMLsec 방법론에 따라 확장된 유스케이스를 보여준다.

표 3. 유스케이스 목록
Table 3. Use case list

유스케이스명	설 명
회원가입	각 사용자는 시스템을 사용하기 위해서 회원가입을 할 수 있다.
로그인	각 사용자는 시스템을 사용하기 위해서 로그인 할 수 있다.
계정확인	시스템이 사용자의 계정을 확인할 수 있다.
잔액확인	각 사용자는 계좌의 잔액을 확인할 수 있다.
거래목록확인	사용자의 거래 목록을 확인할 수 있다.
거래목록다운로드	사용자의 거래 목록을 다운로드 할 수 있다.
요금지불	시스템을 통해 각종 세금을 납부할 수 있다.
계좌생성	관리자는 새로운 계좌를 생성할 수 있다.
계좌삭제	관리자는 기존 계좌를 삭제할 수 있다.
잔액수정	관리자 및 직원은 모든 일반 사용자의 잔액정보를 수정할 수 있다.
거래취소	관리자는 사용자가 수행한 거래를 취소할 수 있다.
등급설정	관리자는 각 사용자에 대한 접근권한을 설정할 수 있다.

표 4. 보안 요구사항이 있는 유스케이스 - 등급설정 유스케이스
Table 4. Use case having security requirement; Use case for rating set-up

Use Case : 등급설정

※ 액터와 관련된 위험성
- 고객은 자신과 관련된 정보만 확인할 수 있어야 한다. 관리자는 모든 사용자의 정보를 확인 및 수정할 수 있다.

※ 보안이 요구되는 입출력 데이터와 보안이 요구되지 않는 입·출력 데이터

보안이 요구되는 I/O	보안이 요구되지 않는 I/O
ID	결과 출력
패스워드	-

※ 변경된 시스템의 행동
- 사용자는 회원가입을 해야 한다.
- 시스템은 로그인을 통해 인증 절차를 거쳐야 하며, 그렇지 않을 경우 사용자는 시스템을 사용할 수 없다.
- 본인 인증 과정에서 입력 정보가 틀릴 경우 시스템은 관련 오류 메시지를 출력해야 한다.
- 관리자는 사용자의 등급을 설정한다.
- 시스템은 사용자에게 결과를 출력해 준다.

1.3 유스케이스 모델 상세화

유스케이스 상세화 활동에서는 직전 활동에서 도출된 각 유스케이스 별로 개요, 관련 액터, 우선순위, 선행/후행 조건, 시나리오, 비기능적 요구사항 항목으로 구성된 유스케이스 명세서를 작성해야 한다[12]. 또한 보안이 요구되는 유스케이스의 경우에는 비기능적 항목에서 보안에 대한 정의를 간결하면서 명확하게 정의하기 위해, 표 2를 참조하여 작성한다.

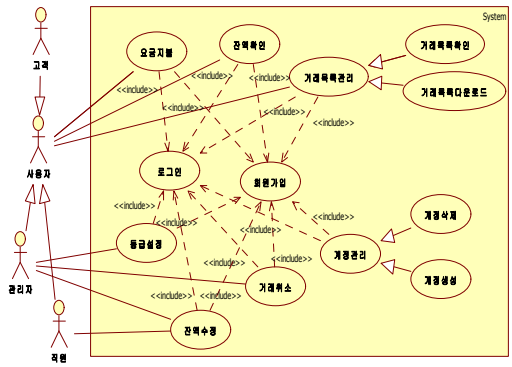

그림 2. '온라인 뱅킹 시스템'을 위한 유스케이스 모델
Fig 2. Use case model for on-line banking system

표 5는 보안이 요구되는 '등급설정' 유스케이스 명세서이다. 또한 유스케이스 명세서를 통해 해당 유스케이스의 다양한 상황, 즉 시나리오를 작성한다[12]. 표 6은 '등급설정' 유스케이스의 기본 시나리오에 해당한다.

표 5. '등급설정' 유스케이스 명세서
Table 5. Use case description for rating set-up

항 목	설 명		
이름	등급설정		
개요	관리자는 각 사용자에 대한 접근권한을 설정할 수 있다.		
관련 액터	주액터	관리자	
우선 순위	1	중요도	1(상)
		난이도	1(상)
선행 조건	관리자로 로그인이 되어 있어야 한다. 설정하고자 하는 사용자가 회원가입이 이루어진 상태이어야 한다.		
후행 조건	로그인 상태가 유지 되어야한다. 시스템은 관리자에게 변경된 사용자의 정보를 보여준다. 시스템은 사용자의 등급을 기록한다.		
시나리오	기본 시나리오	액터와 시스템 간의 기본 시나리오	
비기능적 요구사항	보안 요구사항 - 관리자는 시스템의 전체적인 접근권한을 가지고 있다. - 관리자는 특정 사용자에 대한 시스템 사용권한을 설정할 수 있다.		

표 6. '등급설정' 유스케이스의 기본 시나리오
Table 6. Basic scenario of rating set-up use case

1. 사용자는 회원가입이 되어 있어야한다. 2. 사용자는 로그인 화면에서 ID와 패스워드를 입력하고 로그인 버튼을 누른다. 3. 시스템은 관리자화면을 보여준다. 관리자는 관리자화면에서 등급설정을 선택한다. 4. 등급설정화면에서 해당 사용자의 등급을 확인할 수 있으며, 등급을 수정하기 위해서는 등급설정버튼을 누른다. 5. 시스템은 상세한 등급정보화면을 보여준다. ※ 상세한 등급정보화면 : ID, 이름, 등급 6. 관리자는 등급을 수정한 후, 확인버튼을 누른다. 시스템은 수정된 데이터를 기록하고 상세한 등급정보 화면을 갱신한다. 7. 전 화면으로 되돌아가기 위해서는 취소버튼을 누른다.

1.4 유스케이스 모델 작성

유스케이스 모델 작성은 시스템이 제공할 개별 기능을 유스케이스로 표현하고, 유스케이스와 상호작용을 하는 시스템 외부의 존재를 액터로 표현한다. 그리고 시각적인 표현을 위해 UML의 유스케이스 다이어그램을 사용하여, 액터와 유스케이스 간의 연관 관계를 표현함으로써 어떤 액터가 어떤 유스케이스를 이용하는지를 기술한다[12]. 그림 2는 온라인 뱅킹 시스템의 유스케이스 모델 작성을 보여준다.

2. 시스템 분석 및 설계

시스템 분석 및 설계 단계는 사용자의 요구사항을 충족시킬 수 있도록 시스템의01 구성 요소를 파악하는 것을 목표로 하며, 요구사항 모델을 바탕으로 수행되어야 한다[12].

제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'의 시스템 분석 및 설계 과정은 그림 3과 같다.

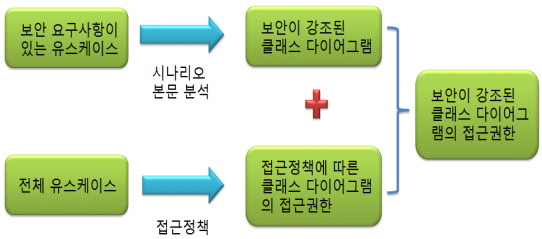

그림 3. 접근정책에 따른 보안이 강조된 클래스 다이어그램의 생성과정
Fig 3. Creating process of security emphasized class diagram depending on access policy

2.1 유스케이스 본문 분석

유스케이스 본문 분석은 사용자로부터 얻은 요구사항 정보들을 토대로 작성된 유스케이스의 기본 시나리오 내용을 본문 분석하여, 시스템에 필요한 클래스들을 추출해 내는 작업을 말한다[1,12].

본문 분석을 통해 추출할 수 있는 클래스는 경계 클래스(Boundary Class)와 제어 클래스(Control Class), 그리고 엔티티 클래스(Entity Class)가 있다[12].

2.2 접근정책 작성

다음은 접근정책 작성 활동으로, 각 액터들이 각각의 유스케이스에 대한 접근권한을 작성해야 한다[5]. 접근정책 작성 과정은 앞서 작성한 보안이 강조되어야 하는 유스케이스 명세서와 보안이 강조될 필요가 없는 일반 유스케이스 명세서를 토대로 유스케이스에 대한 접근권한을 명확하게 명시할 수 있다[3].

작성된 접근정책은 이후 도출될 클래스 다이어그램에 대한 접근권한을 나타낸다. 표 7은 온라인 뱅킹 시스템의 일부 유스케이스에 대한 접근정책을 정의한 것이다.

표 7. 액터에 따른 유스케이스 접근정책
Table 7. Use case access policy according to an actor

	고객	직원	관리자
회원가입	X	X	X
로그인	X	X	X
계정확인	P	X	X
잔액확인	P	X	X
거래목록확인	P	X	X
거래목록 다운로드	P	X	X
요금지불	X	-	X
계좌생성	-	-	X
계좌삭제	-	-	X
잔액수정	-	X	X
거래취소	-	-	X
등급설정	-	-	X
범례 : 모든 권한(X), 일부 권한(P), 권한 없음(-)			

2.3 분석 클래스 다이어그램 작성

접근정책 작성 활동 이후, 분석 클래스 다이어그램의 작성은 유스케이스 시나리오를 본문 분석하여 클래스 다이어그램을 작성하는 활동이다[12].

보안 요구사항이 있는 유스케이스로부터 도출된 클래스들은 보안이 강조되는 클래스들이며, 각 클래스들은 표 7을 참고하여 접근정책에 따른 접근권한을 UMLsec 방법론에 따라 〈〈secrecy〉〉 스테레오 타입을 이용하여 작성한다.

2.4 분석 클래스 다이어그램의 상세화

분석 클래스 다이어그램의 상세화에서는 직전 활동에서 도출된 보안이 강조된 클래스 다이어그램을 바탕으로 유스케이스 시나리오를 추가 본문 분석하여, 각 분석 클래스들의 속성들과 연산들을 정의한다[1,12]. 그림 4는 〈〈secrecy〉〉 스테레오 타입을 이용한 상세화된 분석 클래스 다이어그램이다.

2.5 MVC 패턴 적용

상세화된 분석 클래스 다이어그램에 다음과 같이 MVC 패턴을 적용한다.

① 〈〈entity〉〉 스테레오 타입을 사용한 클래스는 Model로 대응 시킨다.

② 〈〈boundary〉〉 스테레오 타입을 사용한 클래스는 View로서 JSP 등으로 구현한다.

③ 〈〈control〉〉 스테레오 타입을 사용한 클래스는 Controller 로서 서블릿 등으로 구현한다.

④ 〈〈secrecy〉〉 스테레오 타입을 사용한 클래스는 보안이 강조 되어야 하는 클래스이며, 〈〈entity〉〉 스테레오 타입과 같이 사용되었다면 데이터베이스의 역할기반 접근 제어를 이용하여 구현한다.

2.6 관계형 데이터베이스 설계방법

관계형 데이터베이스 설계방법은 O-R 매핑을 통하여 진행된다. O-R 매핑은 객체지향 분석·설계 방법론으로 도출된 분석 클래스 다이어그램을 관계형 데이터 모델링으로 변환이 가능하다[13]. 표 8을 바탕으로 클래스 다이어그램을 관계형 데이터베이스 스키마로의 변환 작업을 수행할 수 있다[14].

표 8. 관계형 데이터베이스 스키마로의 변환방법(O-R 매핑)
Table 8. Method for transformed to Relational database schema(O-R Mapping)

① 클래스는 테이블이 됨.
② 클래스의 속성(attribute)은 테이블의 열(column)이 됨.
③ 클래스의 속성 타입은 테이블의 열 타입이 됨.
④ 일반화가 없는 클래스를 위해서는 integer 기본키를 생성하고, {oid}를 위해서는 기본키 제약 조건에 {oid} 태그 열을 추가함.
⑤ 자식 클래스(Subclasses)들은, 각 부모 클래스의 키를 기본키와 외부키 제약조건에 추가함.
⑥ 속성에 {nullable} 태그가 있으면 테이블 속성에 NULL 또는 NOT NULL을 추가함.
⑦ 속성이 초기 값을 가지면, 열에 DEFAULT 문을 추가함.
⑧ 연관 클래스들은, 각 역할-실행 테이블에 대한 기본키를 기본키와 외부키 제약 조건에 추가함.
⑨ 만일 {alternate oid = 〈n〉} 태그이면, UNIQUE 제약 조건에 대한 열을 추가함.
⑩ 각 명시된 제약에 대해 CHECK를 추가함.
⑪ 0..1, 1..1 규칙의 연관 관계에서 참조하는 테이블에 외부키를 생성함.
⑫ 집합 테이블(CASCASE와 같이)의 외부키를 갖는 복합집합을 위해서 기본키를 생성한다 : 기본키를 위해 추가적인 열(column)을 추가함.
⑬ 이진 연관 클래스를 적당한 "N"쪽 테이블로 이동함으로써 최적화 함.
⑭ 연관 클래스가 아닌 3원 연관은 N : N에 대한 테이블로 생성함.
⑮ N : N, 3원 연관에서 역할-실행 테이블의 키로부터 기본키와 외부키 제약 조건을 생성함.
⑯ 연관 클래스 없는 다대다(many-to-many) 연관을 위해 기본키와 외부키를 생성함.

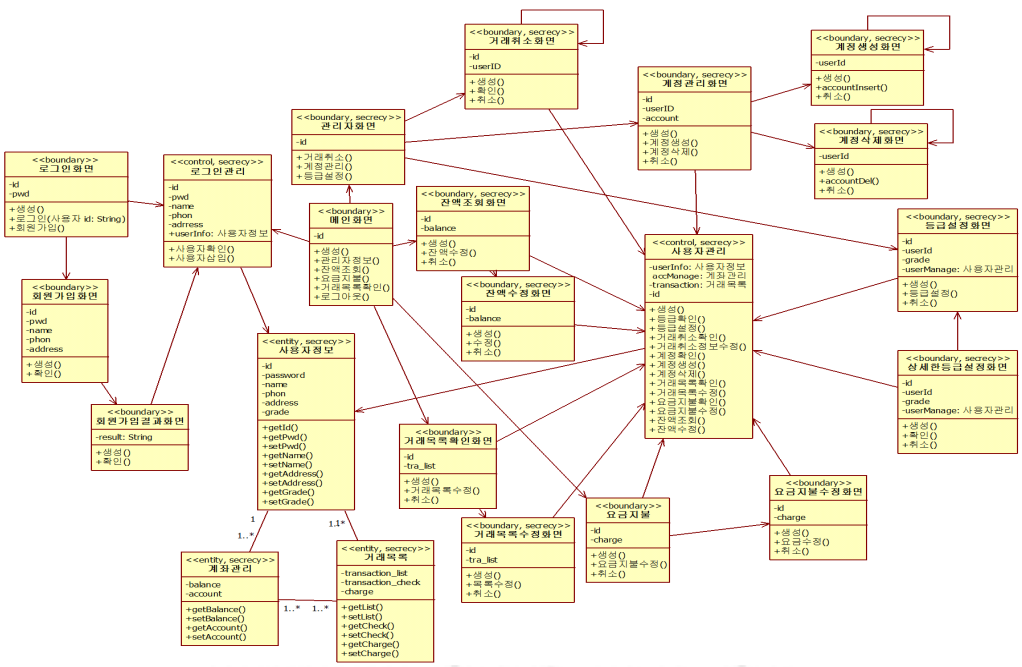

그림 4. 상세화된 분석 클래스 다이어그램
Fig 4. Detailed analysis class diagram

3. 구현

3.1 SQL의 DDL

표 9는 O-R 매핑을 통해 변환된 테이블 스키마 중 '사용자 정보' 테이블을 SQL의 DDL구문 중 'CREATE TABLE' 명령어를 사용하여 생성한 것이다.

표 9. '사용자정보' 테이블 스키마
Table 9. User info table schema

```
CREATE TABLE user_info （
pid INTEGER PRIMARY KEY,
id VARCHAR(10) NOT NULL,
password INTEGER NOT NULL,
name VARCHAR(12) NOT NULL,
phon VARCHAR(15) NOT NULL,
address VARCHAR(30) NOT NULL,
grade VARCHAR(6) NOT NULL,
acc_num INTEGER REFERENCE account,
tran_num INTEGER REFERENCE transaction,
CONSTRAINT puzzleinfo_PK PRIMARY KEY(pid, acc_num,
tran_num) );
```

3.2 SQL의 DCL

생성된 테이블들의 클래스들은 설계과정에서 모두 〈〈secrecy〉〉 스테레오 타입이 사용되었기 때문에 보안이 강조되는 테이블들이다. 이에 따라 각 테이블에 대한 역할기반 접근제어 권한을 SQL의 DCL구문 중 "GRANT" 명령어를 통해 설정할 수 있다. 표 10은 '사용자' 및 '관리자' 역할에 대한 접근제어 스키마이다.

표 10. '사용자' 및 '관리자' 역할에 대한 접근제어 스키마
Table 10. Access control schema for user and administrator role

```
CREATE ROLE user_entry;
GRANT user_entry TO user_view;
GRANT ALL ON
user_info TO user_entry;
GRANT SELECT ON
account TO user_entry;
GRANT SELECT ON
transaction TO user_entry;
CREATE ROLE admin_entry;
GRANT admin_entry TO admin_view;
GRANT ALL ON
user_info TO admin_view;
GRANT ALL ON
account TO admin_view;
GRANT ALL ON
transaction TO admin_view;
```

IV. 평가 및 결 론

본 논문에서 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'은 기존의 객체지향 분석·설계 방법론이 명확하게 제시하지 못했던 보안에 대한 분석·설계를 보안이 강조된 모델링 언어인 UMLsec과 통합하여 사용자 요구사항 정의부터 구현까지, 각 단계마다 보안이 강조된 객체지향 분석·설계 방법론으로 제안하였다. 아울러 분석·설계의 결과를 바탕으로 보안에 대한 구현은 데이터베이스의 SQL 구문 중 DDL과 DCL 구문을 사용하여 각 역할에 따른 접근을 제어할 수 있는 역할기반 접근제어로 구현하였다. 따라서 본 논문에서 제안한 '안전한 데이터베이스 설계를 위한 객체지향 분석·설계 방법론'은 대부분의 객체지향 분석·설계 방법론이 제시하지 못했던 관계형 데이터베이스와의 상호 연관성을 지원하고 있을 뿐만 아니라, 기존의 데이터베이스 설계방법론이 명확하게 제시하지 못했던 보안을, UMLsec을 사용하여 개발 초기부터 강조하여 보안이 데이터베이스 설계에 반영토록 하였다.

이에 따라 기존의 객체지향 분석·설계 방법론과 보안 그리고 관계형 데이터베이스와의 상호 연관성을 제시하여, 데이터베이스 시스템 개발 전 주기에 대한 안전하고 일관된 객체지향 분석·설계가 가능하다.

참고문헌

[1] Joo Kyung-Soo, Woo Jung-Woong, "A Development of the Unified Object-Oriented Analysis and Design Methodology for Security-Critical Web Applications Based on Object-Relational Data - Forcusing on Oracle 11g-", Korea Society of Computer Infomation, Vol. 17, No. 12, pp. 169-177, 2012

[2] Han Jeong-Su, Kim Gwi-Jeong, Song Yeong-Jae, "Introduction to UML : Object-Oriented Design as in a friendly learning", Hanbit Media. Inc, pp. 58-66, 2009.

[3] Brett D. McLaughlin, Gary Pollice, David West, "Head First Object Oriented Analysis & Design", Hanbit Media. Inc, pp. 96-103, 2007.

[4] Eduardo Fernaʹndez-Medinaa, Juan Trujillob,

Rodolfo Villarroelc and Mario Piattinia, "Developing secure data warehouses with a UML extension", Journal Information Systems archive, vol. 32 No. 6, pp.826-856, 2007

〔5〕 G.Popp, J. Jurjens, G.Wimmel, R. Breu, "Security-Critical System Development with Extended Use Case", Asia-Pacific Software Engineering Conference, 5-1 self, 2003.

〔6〕 Madan, s, "Security Standards Perspective to Fortify Web Database Applications From Code Injection Attacks", International Conference on Intelligent Systems, Modelling and Simulation(ISMS), vol. 10, pp. 226-230, 2010.

〔7〕 lqra Basharat, Farooque Anam, Abdul Wahab Muzaffar, "Database Security and Encryption: A Survey Study", International Journal of Computer Application, vol. 47, No. 12, pp28-34, 2012

〔8〕 David Basin, Jürgen Doser and Torsten Lodderstedt, "Model Driven Security: from UML Models to Access Control Infrastructures", ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 15 No. 1, pp39-91, 2006

〔9〕 Cho Wan-Su, "UML 2 & UP Object-Oriented Analysis&design", pp.189-205, Hongrung Publishing Company, 2005.

〔10〕 Jeon Byeong-Seon, "CBD, WHAT&HOW", Wowbooks, pp. 189-205, 2005.

〔11〕 Joo Kyung-Soo, Jho Do-Hyung, "Development of Integrated Design Methodology for Relational Database Application -Focusing on Object-Oriented Analysis and Design Methodology-", Korea Society of Computer Information, Vol. 16, No. 11, pp. 25-34, 2011.

〔12〕 Chae Heung-Seok, Object-oriented CDB Project for UML and Java as learning, Hanbit Media. Inc, pp. 84-112, 2009.

〔13〕 Cho Jeong-Gil, "An Efficient Transformation Techn ique from Relational Schema to Redundancy Free XML Schema", Korean Society for Internet Information, Vol. 11, No. 6, pp. 123-133, 2010

〔14〕 Bang Seung-Yun, Joo Kyung-Soo, "Design Metho dology for XML Schema Application based on UML", Soonchunhyang Univ, pp.71-75, 2003.

저 자 소 개

주 경 수

1993: 고려대학교 전산학과 공학박사.

현 재: 순천향대학교 컴퓨터소프트웨어공학과 교수

관심분야: Database System, XML, System Integration, Object Oriented System, Cloud Databases, BigData Databases,

Email : gsoojoo@sch.ac.kr

우 정 웅

2012: 순천향대학교 컴퓨터학과 공학사.

현 재: 순천향대학교 컴퓨터소프트웨어공학과 석사과정

관심분야: Database System, UML, Object Oriented System, aaCloud Databases, BigData Databases,

Email : jyone0715@gmail.com