
인터넷기반 원격시험시스템의 객체지향적 분석 및 설계

김윤호* · 박운영**

An Object-Oriented Analysis and Design
for Remote Examination System on Internet

Yun-Ho Kim* · Un-Young Park**

요 약

본 논문에서는 UML에 기반한 객체지향적 모형화에 의하여 인터넷에 기반한 원격 시험 시스템의 체계적인 분석 및 설계를 제시한다. 본 시스템은 기본적으로 교수가 문제를 출제하는 문제 출제부와 학생이 문제에 응시하는 문제 풀이부로 구성된다. 객체지향에 기반하여 설계하고 설계과정에서 UML을 사용하여 설계 내용을 표현함으로써 객체지향적 설계의 장점인 체계적인 설계와 재사용성, 특정 프로그래밍 언어에의 의존성 등을 확보함으로써 개발과 테스트, 수정, 보완에 있어서 효율성과 비용 절감을 얻을 수 있다. 또한 본 논문에서의 원격 시험 시스템은 원격 교육 시스템에 적용시킬 수 있으며, 기존의 오프라인 시험의 대치 또는 인터넷에 기반한 설문조사를 시행하는 시스템으로의 전환도 가능하여 다양한 분야에 적용이 가능하다.

ABSTRACT

This Paper presents an object-oriented analysis and design for Remote Examination System on Internet. The UML is used for describing the process of analyzing and designing the system. The system is composed of the parts: Question Generation part that professors make questions, Question Answer part that students answer the questions. By object-oriented analysis and design, the system inherits the advantages of object-oriented methodology. All the results are retained on database on server-side. This system can be integrated with the Cyber-University System, and substituted for off-line paper-based tests or adopted to on-line poll applications on Internet.

키워드

Internet Computing, OOA/D/P, Cyber-University

1. 서 론

인터넷 사용자의 급속한 증가와 통신망의 고속화로 인하여 인터넷 응용분야는 날로 증가하고 있다. 특히, 최근에 주목받고 있는 분야 중에 하나가 인터넷 기반의 원격 교육에 관한 연구이다. 원격교육에서

는 교수와 학생간의 강의와 학습, 평가를 위한 시험의 시행이 원격으로 이루어지게 된다. 현재의 연구 동향은 주로 원격 강의 서비스에 초점이 맞춰져 있다. 그러나 원격교육에서 교육의 효과를 평가하기 위해서는 교육의 내용에 대한 시험을 시행할 수 있는 기능 또한 강의만큼 중요한 기능이다. 따라서 시험의

* 안동대학교 전자정보산업학부

** 컴포넌트뱅크 연구원

시행 또한 강의 서비스 시스템과 통합되어 운영되는 것이 바람직하다.

한편, 인터넷의 급속한 보급으로 인해 소프트웨어의 개발과 운영 환경이 국제화, 분산화, 멀티미디어화 될 것이 요구되고 있으며, 사용자의 요구와 개발의 비용이 날로 증가하고 있다. 따라서 이로 인한 소프트웨어의 설계 비용을 절감하고 신뢰성과 보편성을 갖춘 개발 방법론 역시 절실히 요구되고 있다. 이러한 요구에 대안으로 제시된 것이 객체지향 소프트웨어 개발 방법론 [1,2]이다. 또한, 객체지향에 기반한 문제의 분석 및 소프트웨어의 설계 및 구현의 체계적인 수행을 위하여 UML(Unified Modeling Language)[3]이 제안되었다. 따라서 소프트웨어 프로젝트 수행시, UML에 기반한 객체지향 소프트웨어 개발은 현시점에서 가장 적절한 개발 방법론으로 받아들여지고 있다.

본 논문에서는 UML에 기반한 모형화를 통하여 인터넷에 기반한 원격 시험 시스템(remote testing system)의 체계적인 분석 및 설계를 제시하고자 한다. 이 시스템은 기본적으로 교수가 문제를 작성하는 문제 출제부와 학생이 문제에 응시하기 위한 문제 풀이부로 구성되었다. 출제된 문제 지 및 학생의 응답지는 서버에 존재하는 데이터베이스에 저장되어 출제 및 응시가 처리되도록 하였다. 이러한 원격 시험 시행 시스템은 원격 교육의 성적 평가를 위해서 뿐만 아니라, 시스템에 대한 약간의 수정을 통하여 기존의 오프라인 시험 또는 복잡한 형태의 설문조사를 온라인에서 시행할 수 있는 시스템으로의 전환도 가능하여 폭넓은 적용력을 가지고 있다.

II. 시스템의 분석 및 설계

본 논문에서는 UML을 이용하여 원격 시험 시스템의 문제 분석 및 시스템 설계를 수행한다. UML은 소프트웨어 프로젝트를 기술하기 위한 표준으로서 제안되었으며, 대규모 소프트웨어 시스템의 가시화와 명세, 구성, 문서화에 사용된다[4]. UML에서의 Use Case는 정적 모형화(static modeling)의 성격을 가지며 시스템의 요구사항을 가시화하기 위하여 사용되며 분석 단계뿐만 아니라 설계 및 구현, 나아가서 사

용자 설명서에도 이용될 수 있다[5,6]. 시스템의 문제 분석은 Use Case를 이용하여 분석되며, Use Case는 선행조건, 이벤트의 흐름, 후행조건 등으로 이루어진다. 또한 Use Case는 GUI 화면 설계도 포함시킬 수 있다. 따라서, 객체지향에 기반한 시스템의 분석 및 설계를 위하여 Use Case를 이용하여 문제의 분석 및 설계를 하는 방법이 바람직하다[7].

2.1 Use Case 모형화

Use Case를 작성하기 위해서는 시스템의 경계 내에서 Actor와 이와 관련된 사용례를 판별하는 것이 주요한 사항이다. Actor는 시스템의 외부에서 시스템을 사용하는 역할을 하며, Use Case는 Actor가 시스템이 하기를 원하는 기능들을 기술한다. 원격 시험 시스템에서는 시스템의 Actor로서 교수와 학생이 있고, 교수는 문제의 출제와 관리의 기능이 그리고 학생은 문제의 풀이와 성적의 확인이 사용례에 해당한다.

따라서 Use Case 분석에 있어서 교수와 학생이 Actor로 설정한다. 그리고 문제지 관리, 성적 처리, 문제지 작성, 문제의 수정 등이 교수 Actor의 Use Case로 설정하며, 문제풀이와 성적 확인과 같은 사항들이 학생의 Use Case로 설정한다. 원격 시험 시스템의 총괄적인 Use Case Diagram을 보이면 그림 1과 같다.

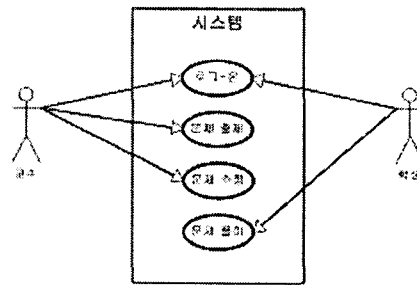


그림 1. 시스템의 총괄적 Use Case Diagram
Fig. 1 Overview of System Use Case Diagram

본 논문에서 설정한 Use Case는 교수와 학생의 Actor에 기초하여 다음과 같다. Use Case는 필요에

따라 더 세분화, 상술화하는 것이 가능하다.

- 로그-온 Use Case
- 문제 출제 Use Case
- 문제 수정 Use Case
- 문제 풀이 Use Case

아래에서 표들을 통하여 각 Use Case에 대한 상세한 내용을 보인다. 여기서 P는 Priamry Scenario를 나타낸다. 이벤트의 흐름은 Primary Secnario와 Alternative Scenario로 나누어지는데, Primary Scenario는 Use Case의 정상적인 흐름에 대한 각 단계를 나열하는 것이고, Alternative Scenario는 기본적인 흐름에서 분기되거나 비정상적인 흐름에 대한 이벤트들의 흐름을 나열하는 것이다. 각 Use Case는 이벤트의 흐름의 각 단계에서 다른 Use Case를 호출할 수 있으며, 여러 Use Case들에 공통된 단계들을 추출하여 새로운 Use Case를 생성시킬 수 있다.

표 1. 로그-온 Use Case 시나리오

Table 1. Scenario of Log-on Use Case

Use Case Name	로그-온
Actor	교수, 학생
Flow of Event	P1: 사용자가 로그-온할 때 시작 P2: 시스템은 로그-온 대화화면을 출력 P3: 사용자는 ID와 PW를 입력 P4: 시스템은 입력된 정보를 확인 P5: 시스템은 접근을 허용 P6: 종료.
Alternative Flows	A1: Invalid ID(계정)가 입력되면 재입력 요구 A2: Invalid PW(비밀번호)가 입력되면 재입력 요구

다음의 표 1에서는 로그-온 Use Case의 시나리오를 보여주며, 표 2는 문제 출제 Use Case의 시나리오를 보인다. 표 2에서는 P2단계에 로그-온 Use Case를 호출하며, P4단계가 반복 수행되면서 문제의 출제가 진행된다.

표 2. 문제출제 Use Case 시나리오

Table 2. Scenario of Question Use Case

Use Case Name	문제 출제
Actor	교수
Flow of Event	P1: 교수의 로그-온에 의해 시작 P2: 시스템은 로그-온 Use Case를 실행 P3: 교수는 출제할 과목을 선택 P4(반복): 교수는 문제의 출제를 수행 p0: 출제된 문제의 일람 p1: 문제의 유형 선택(주관식/객관식) p2-1: 주관식의 경우 단답식/서술식의 유형 선택 p2-2: 객관식의 경우 선택지문을 입력 P5: 종료
Alternative Flows	A1: 출제할 과목이 없을 때 재선택 요구 A2: 문제 출제가 완료되지 않고 종료될 때 문제지의 저장 불가 A3: 문제의 수정을 위한 기능으로의 전환

표 3. 문제수정 Use Case 시나리오

Table 3. Scenario of Modification Use Case

Use Case Name	문제 수정
Actor	교수
Flow of Event	P1: 교수가 문제 수정을 선택할 때 시작 P2: 시스템은 과목 선택창을 표시 P3: 교수는 과목과 문제를 선택 P4: 교수는 문제의 유형과 항목을 수정(삭제) P5: 교수가 다른 기능을 선택하면 종료
Alternative Flows	A1: 수정 후 저장을 확인

여기서 문제의 유형이 주관식인 경우 단답식일 때는 Text Field형태의 폼을 제공하고 기술식일 때는 Text Area형태의 폼을 제공한다. 객관식일 경우 Radio Button, Combo Box, Check Box 등의 형태의 폼을 제공하는데, 지문 중에서 하나의 답만을 선

택해야 하는지 두 개 이상의 지문을 선택할 수 있는 지에 따라 Radio Button 형태의 지문과 다른 형태의 지문이 구분되어 사용된다.

표 3에서는 문제 수정 Use Case의 시나리오를 보인다. 여기서는 문제의 유형과 질의문, 항목에 대한 수정과 문제 또는 문항의 삭제도 포함된다. 또한 표 4에서 학생 Actor에 의해 수행되는 문제 풀이 Use Case의 시나리오를 보인다. 표 4의 P2에서는 로그-온 Use Case를 호출하여 유효한 사용자인지를 확인하며, 시스템이 응시가능한 문제지를 제시하면 학생은 문제풀이를 진행하는 과정을 거치게 된다.

표 4. 문제풀이 Use Case 시나리오
Table 4. Scenario of Answering Use Case

Use Case Name	문제 풀이
Actor	학생
Flow of Event	P1: 학생이 문제풀이를 선택하면 시작 P2: 학생이 로그-온 기능을 선택하면 로그-온 Use Case가 실행 P3: 시스템은 학생이 응시가능한 문제지를 출력 P4: 학생은 응시할 문제지를 선택 P5: 학생은 문제를 풀이 P6: 학생은 문제 풀이를 마치면 그 결과의 저장을 선택 P7: 학생이 다른 사항을 선택하면 종료
Alternative Flows	A1: 학생이 응시가능한 문제지가 없을 경우 오류 메시지를 출력 A2: 학생이 풀이 결과를 저장하지 않고 종료를 시도하면 경고 메시지를 출력

2.2 시퀀스 모형화

UML에서 시스템의 정적인 면을 모형화하기 위해서는 Class Diagram과 Object Diagram 등을 이용하며, 동적인 면의 모형화를 위해서는 Interaction Diagram을 사용한다. Interaction Diagram에는 Sequential Diagram과 Collaboration Diagram이 있는데, 전자는 메시지의 시간순서에 따른 흐름을 보여 주며 후자는 상호작용에 관여하는 객체들의 구성을 보여준다.

로그-온 Use Case에 대한 Sequence Diagram을 아래의 그림 2에서 보인다. 좌측의 Actor는 교수 또는 학생 Actor를 나타낸다. 그림2에서 보는 바와 같이 Sequence Diagram은 교수나 학생이 접속을 시도하면서 로그-온 과정에서 일어나게 되는 객체들과 메시지의 흐름을 시간 순서에 따라 보여주고 있다. 사용자가 자신의 ID와 비밀번호를 입력하게 되면 데이터베이스에 저장된 내용과 비교하여 그 유효성과 비유효성을 검사하여 처리하도록 한다.

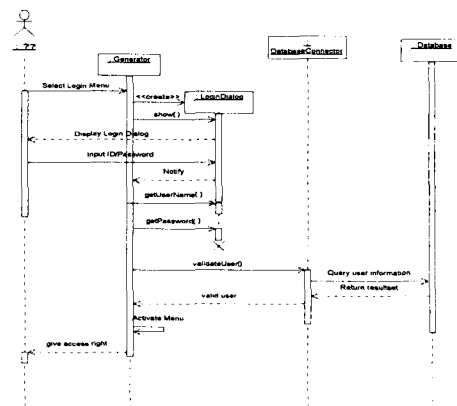


그림 2. 로그-온 Sequence Diagram
Fig. 2 Sequence Diagram of Log-On

문제 출제 Use Case에 대한 처리과정은 다음과 같다. 여기서는 교수가 로그-온 Use Case의 통과하였을 때 교수가 선택할 수 있는 과목을 사용자에게 출력하여 준다. 교수는 출력된 과목 중 하나를 선택하고, Generator 객체는 해당되는 문제들을 데이터베이스로부터 입수하여 사용자에게 문제지를 출력한다. 교수는 새로운 문제를 출제하거나 기존 문제에 대한 삭제를 포함한 수정을 할 수 있다. 문제 출제 Use Case에 대한 Sequence Diagram을 그림 3에서 보인다. 좌측의 Actor는 교수 Actor를 나타낸다.

문제 풀이 Use Case에 대한 Sequence Diagram을 보이면 그림 4와 같다. 학생은 원격 시험 시스템에 로그-온하고 이를 통과하여 접근 권한을 갖게 되면, 응시 가능한 문제지를 선택할 수 있게 된다. 선택된 문제지에 대하여 풀이를 하고 난 뒤에 풀이한 결과들을 저장하고 시스템에서 로그아웃하게 된다.

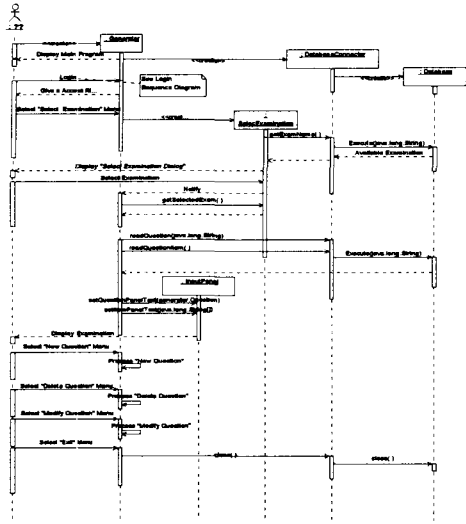


그림 3. 문제 출제 Sequence Diagram
Fig. 3 Sequence Diagram of Question Use Case

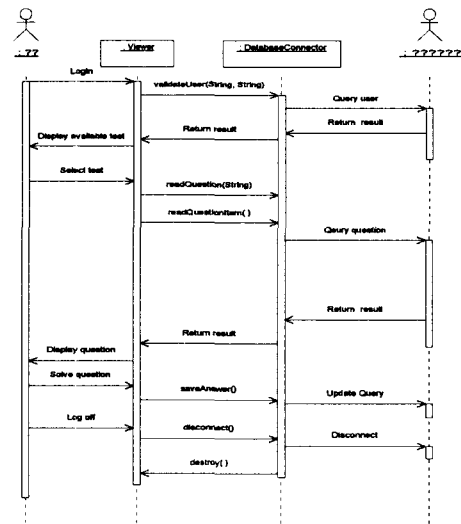


그림 4. 문제 풀이 Sequence Diagram
Fig. 4 Sequence Diagram of Answering Use Case

응시하는 문제지는 viewer 객체에 의해 처리되며 응답한 결과는 databasehandler 객체에 의하여 저장된다. 좌측의 Actor는 학생 Actor를 나타내며, 우측의

Actor는 데이터베이스 Actor를 나타낸다.

2.3 클래스 모형화

클래스 모형화의 목적은 클래스를 식별해내고 클래스 자신에 대한 속성과 메소드에 대한 정의를 하며, 이들 클래스들 간의 상호관계성을 밝혀내기 위한 것이다. 본 논문에서의 시스템을 위한 클래스들에 대한 패키지들을 그림 5의 Package Diagram에 보인다. 그림 5에서 보듯이 시스템에 속한 클래스들은 examination, generator, viewer, databasehandler의 패키지에 속하게 되며, 각 패키지에 속한 개별 클래스들은 더욱 세분화, 상술화되어 설계될 수 있다.

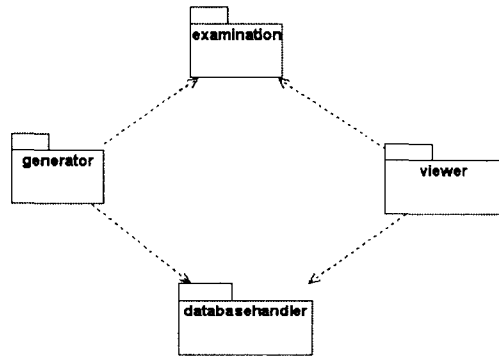


그림 5. 클래스들의 Package Diagram
Fig. 5 Package Diagram of Classes

그림 6에서 examination 패키지에 속한 클래스들과 클래스들 사이의 상관관계를 표시한 Class Diagram을 아래에서 보인다. 이 패키지에는 Question, QuestionItem, QuestionList의 3가지 클래스가 있다. Question과 QuestionItem 클래스는 하나의 문제와 그 문제들을 포함하는 문제지에 대응되고, QuestionItem 클래스는 한 문제에 속하는 지문에 대응된다.

다음에는 generator 패키지에 속한 클래스들과 이들 클래스 간의 상관관계에 대한 내용을 그림으로 보인다. 그림 7에서 보듯이 generator 패키지에는 문제지와 문제, 문항들을 작성하여 이들을 생성하는 기능들에 관련된 클래스들이 포함되어 있다. 이 패키지는 특히 응시자의 문제 풀이의 편의성을 도모하기 위하여 다양하고 유려한 GUI를 제공하는 화면설계와 깊은 관련이 있는 클래스들이 포함되는 패키지이

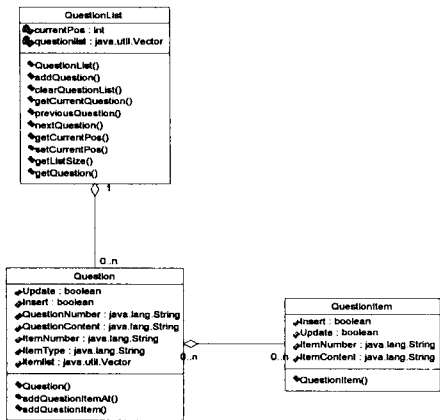


그림 6. examination 패키지의 Class Diagram
Fig. 6 Class Diagram of examination Package

터베이스의 종류에 독립적인 구현을 원한다면 Factory Pattern[8,9]을 이용하여 설계하는 것이 바람직하다.

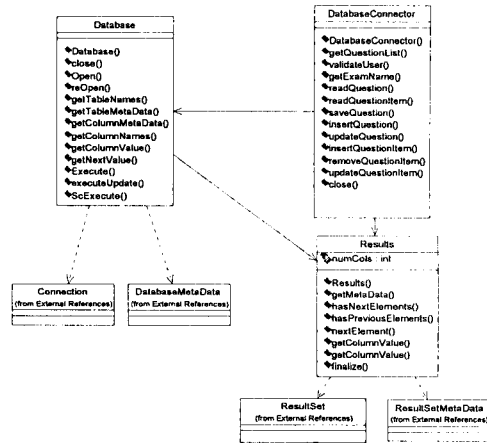


그림 8. databasehandler 패키지의 Class Diagram
Fig. 8 Class Diagram of databasehandler Package

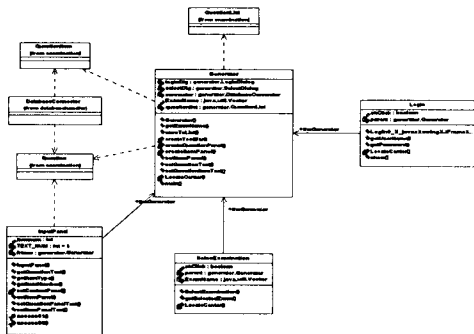


그림 7. generator 패키지의 Class Diagram
Fig. 7 Class Diagram of generator Package

며, 단순히 텍스트 형식의 문제형태뿐만 아니라 그림 또는 동영상과 같은 형태의 문제형식이 지원될 수 있는 부분이다. 또한 문제지의 작성이 이루어지는 부분이므로 응시자의 편의성을 도모함과 함께 철저한 보안성도 요구되는 부분이다.

databasehandler 패키지에 속한 클래스들과 이들의 상관관계에 대한 내용은 그림 8과 같다. databasehandler 패키지에 속한 클래스들은 데이터베이스 연결을 위해 사용되는 클래스들로 Database, DatabaseConnector, Results 클래스들로 구성되어 있다. 데이터베이스에의 연결을 위해 Database와 Results라는 두 개의 클래스를 설계함에 있어 데이

그림 9는 viewer 패키지에 속한 클래스들과 클래스들 간의 상관관계를 보이는 Class Diagram이다. viewer 패키지에 속한 클래스들은 출제된 문제를 화면에 출력하는 기능을 담당하고 있으며 문제 풀이 Use Case의 구현하는 역할을 하는 클래스들로 구성되어 있다.

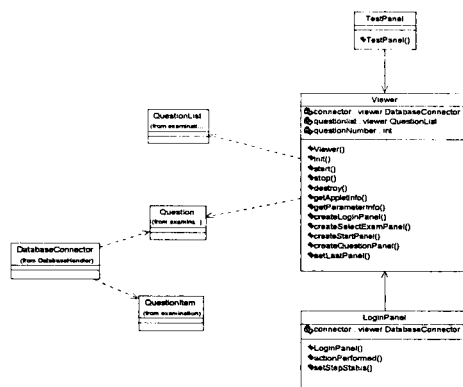


그림 9. viewer 패키지의 Class Diagram
Fig. 9 Class Diagram of viewer Package

2.4 데이터베이스 모형화

UML에서는 데이터베이스의 설계에 관한 내용도 Class Diagram으로 표현할 수 있다. 클래스 모형화의 결과로 식별된 클래스들 중 지속적인 데이터의 유지를 필요로 하는 클래스들에 대해 데이터베이스로의 사상(mapping)을 함으로써 데이터베이스에 대한 설계를 수행한다. 다음의 그림 10은 설계된 데이터베이스의 테이블들의 구조와 이들 간의 관계를 Class Diagram으로 표시한 것이다. 데이터베이스의 테이블을 나타내기 위하여 각 클래스들은 <<Table>> stereo type[3]으로 표시한다. 주요 테이블을 살펴보면 Question 테이블은 출제된 문제를 저장하는 역할을 하며, QuestionItem 테이블은 각 문제의 지문을 저장하는 역할을 한다. 교수와 학생에 대한 정보는 ProfessorInformation 과 StudentInformation 테이블에서 관리된다.

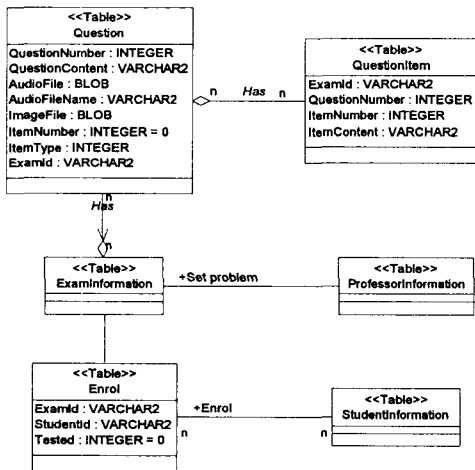


그림 10. 데이터베이스 테이블의 Class Diagram
Fig. 10 Class Diagram of Database Tables

III. 결 론

본 논문에서는 UML에 기반한 객체지향적 모형화에 의하여 인터넷에 기반한 원격 시험 시스템의 체계적인 분석 및 설계를 제시하였다. 본 시스템은 기본적으로 교수가 문제를 출제하는 문제 출제부와 학

생이 문제에 응시하는 문제 풀이부로 구성되었다. 출제된 문제지 및 학생의 응답지는 데이터베이스에 저장되어 출제 및 응시의 결과가 처리되도록 하였다. 객체지향에 기반하여 설계하고 설계과정에서 UML을 사용하여 설계내용을 표현함으로써 객체지향적 설계의 장점인 체계적인 설계와 재사용성, 특정 프로그래밍 언어에의 의존성 등을 확보함으로써 개발과 테스트, 수정, 보완에 있어서 효율성과 비용 절감을 얻게 되었다. 또한 본 논문에서의 원격 시험 시스템은 원격 교육 시스템에 적용시킬 수 있으며, 기존의 오프라인 시험의 대치 또는 인터넷에 기반한 설문조사를 시행하는 시스템으로의 전환도 가능하여 다양한 적용력이 기대된다.

참고문헌

- [1] G. Booch, "Object-Oriented Development," IEEE Trans. Software Engineering, vol. SE-12, no.2, pp. 211-221, 1986.
- [2] G. Booch, Object-Oriented Analysis and Design with Applications, Addison-Wesley, 2nd Ed., pp. 3-26, 1994.
- [3] J. Rumbaugh, I. Jacobson and G. Booch, The Unified Modeling Language Reference Manual, Addison-Wesley, pp. 3-11, pp. 449-452, 1999.
- [4] G. Booch, J. Rumbaugh and I. Jacobson, The Unified Modeling Language User Guide, Addison-Wesley, pp.13-35, 1999.
- [5] G. Schneider and J. P. Winters, Applying Use Cases: A Practical Guide, Addison-Wesley, pp. 83-132, 1998.
- [6] D. Rosenberg, Kendall Scott, Use Case Driven Object Modeling with UML: A Practical Approach, Addison-Wesley, pp. 37-56, 1999.
- [7] R. G. Fireman and C. F. Kemerer, "Object-Oriented and Conventional Analysis and Design Methodologies," Computer, vol. 25, no. 10, pp. 22-39, 1992.
- [8] E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, pp. 107-

116, 1995.

- [9] M. Grand, Patterns in Java, vol. 1, Wiley, pp. 89-98, 1998.

저자소개



김운호(Yun-Ho Kim)

1983. 2. 경북대학교 전자공학과
공학사

1993. 2. 경북대학교 대학원 컴
퓨터공학과 공학석사

1997. 2. 경북대학교 대학원 컴
퓨터공학과 공학박사

1997. 8.-현재 안동대학교 전자정보산업학부 조교수
※관심분야: VLSI 구조 및 알고리즘, 분산병렬처리,
인터넷 컴퓨팅, 객체지향 분석/설계/프로그래밍

박운영(Un-Young Park)

2000. 3. 안동대학교 컴퓨터공학과 공학사

2002. 2. 안동대학교 컴퓨터공학과 공학석사

2002. 3.-현재 컴포넌트 뱅크 연구원

※관심분야: 객체지향, 인터넷컴퓨팅