

자바를 이용한 자동 문제 출제 시스템의 설계 및 구현

박운영^o, 김윤호
안동대학교 컴퓨터공학과

A Design and Implementation of Dynamic Test Generating System Using Java

Un-Young Park^o, Yun-Ho Kim
Dept. of Computer Engineering, Andong National University

요 약

인터넷 사용자의 급속한 증가와 통신망의 고속화는 교육 분야에 있어서도 많은 변화를 가져왔다. 최근 주목을 받는 분야중에 하나는 인터넷 기반의 가상대학이다. 현재의 동향은 주로 원격 강의 시스템에 치우쳐있다. 가상대학에서 해결해야 할 또 다른 중요한 부분은 학생의 수학 능력을 평가하는 시스템이다. 본 논문에서는 자바를 이용한 자동화된 문제 출제 시스템과, 출제된 문제를 학습자에게 출력하는 시스템을 제안하고 구현한다. 편리한 사용자 인터페이스를 가지고, 멀티미디어를 가미한 문제 출제 시스템으로의 확장성을 제공하는 동시에 현재 강의 시스템과 병행하여 최대한의 교육 효과를 내는데 그 목적이 있다.

1. 서론

인터넷 사용자의 급속한 증가와 통신망의 고속화는 교육 분야에 있어서도 많은 변화를 가져왔다. 최근 주목을 받는 분야중에 하나는 인터넷 기반의 가상대학이다. 가상대학에서는 교수와 학생 간의 교수와 학습, 평가와 응시가 원격으로 이루어지게 된다. 현재의 동향은 주로 원격 강의 시스템에 치우쳐있다. 가상대학에서 해결해야 할 또 다른 중요한 부분은 학생의 수학 능력을 평가하는 시스템이다. 현재 가상대학중 시험시스템에서는 문제출제를 위해 웹 페이지를 작성하는 형태를 띄고 있고 웹페이지의 작성은 문제 출제의 번거로움을 야기시키게 된다. 동적인 문제출제에 관련된 기존 연구들은 Script나 HTML로 문제출제시스템을 구성함으로써 사용자 인터페이스부분에서 취약한 측면을 가지고 있다[1, 2, 3, 4]. 본 논문에서는 자바를 이용한 자동화된 문제 출제 시스템과, 출제된

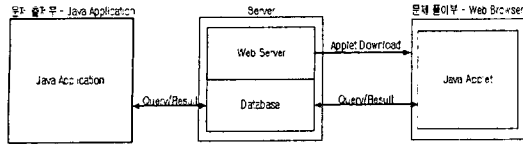
문제를 학습자에게 출력하는 시스템을 제안하고 구현한다.

자바 스왈의 사용자 인터페이스를 사용하여 구현하고, 컴퓨터 비전공들은 웹 페이지의 작성을 통해 문제 출제를 하는 것이 아니라 문제 출제 시스템의 간단한 입력으로 문제를 출제할 수 있으며 자바의 JDBC[5]를 이용하여 데이터베이스에 독립적인 문제 출제 시스템을 구성한다. 편리한 사용자 인터페이스를 가지고, 멀티미디어를 가미한 문제출제 시스템으로의 확장성을 제공하는 동시에 현재 강의 시스템과 병행하여 최대한의 교육 효과를 낼 수 있다.

2. 자동 문제 출제 시스템의 설계

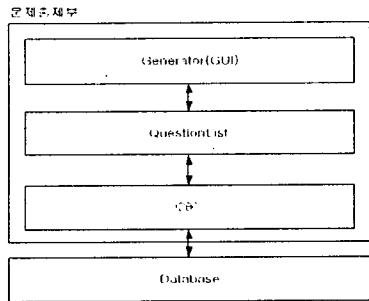
2.1 전체시스템의 구성

자동 문제 출제 시스템의 전체적인 구성은 그림1과 같다.



[그림1] 전체 시스템 구성도

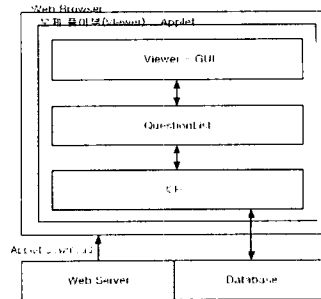
자동 문제 출제 시스템은 크게 문제출제부와 문제풀이부, 서버로 나누어진다. 문제출제부는 Java Application의 형태로 구현되며 JDBC 드라이버를 통해서 문제 출제 데이터베이스를 액세스한다. 문제풀이부는 Applet의 형태로 홈페이지 내에서 운영된다. 문제풀이자는 HTTP 프로토콜을 이용하여 문제풀이부에 접속요청을 하게 되며 서버의 애플릿이 문제풀이자의 컴퓨터에 다운로드되어 실행된다. 문제풀이부에서도 JDBC 드라이버를 통해 데이터베이스를 액세스한다. 서버는 문제풀이부를 운영하기 위한 웹 서버와 문제출제자가 출제한 문제와 문제 풀이자가 풀이한 답을 저장하기 위한 데이터베이스로 구성된다. 자바에서 애플릿 구현시 가장 문제가 되는 보안 문제는 웹서버와 데이터베이스 서버를 물리적으로 동일한 머신으로 동으로서 해결한다.



[그림2] 문제출제부의 구성

그림2는 문제출제부의 구성을 보여준다. 문제출제자는 문제출제부(Java Application)를 통하여 문제를 동적으로 출제하게 된다. 객관식과 단답형 주관식, 서술형 주관식의 문제 유형중 하나를 선택할 수 있고, 객관식의 경우 항목의 개수도 선택하여 출제할 수 있다. 단답형 주관식은 자바 소정의 사용자 인터페이스 컴포넌트인 JTextField에, 서술형 주관식은 JTextArea에, 객관식은 JComboBox, JCheckBox, JRadioButton에 대응된다. 또한 이미 출제된 문제의 형태를 바꾸는 것이 가능하며 객관식의 경우에는 항목의 개수도 변경 가능하다. 그림3은 문제풀이부의 구조를 나타낸다. 문제출제부는 Application의 형태로 구현되지만 문제풀이부

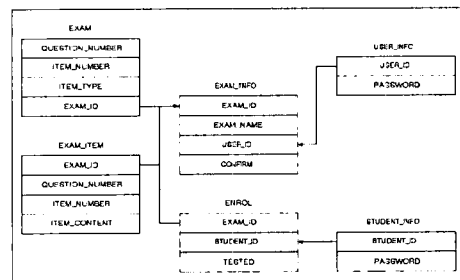
는 다중 사용자를 위해 Applet 형태로 구현된다.



[그림3] 문제풀이부 구성도

문제풀이자는 웹서버에 접속하여 문제풀이부를 실행시키게 된다. 문제풀이자가 HTTP프로토콜을 사용하여 접속요청을 하게 되면 웹서버의 애플릿은 웹브라우저내에서 실행되게 된다. 문제풀이자는 사용자 확인 과정을 거치며, 각 문제풀이자가 풀이할 수 있는 문제지중 하나를 선택할 수 있다. 문제풀이자가 문제 풀이를 마쳤을 경우, 풀이한 답을 문제출제자가 문제출제를 마쳤을 때 생성되는 테이블에 저장하게 되며 문제풀이자는 이후 동일한 문제지에 대해 풀이를 할 수 없다.

2.2 데이터베이스 설계



[그림4] 데이터베이스 테이블 연관 관계

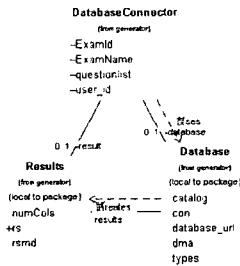
그림4는 자동 문제 출제 시스템에 사용되는 테이블들과 그들 사이의 연관 관계를 보여준다. EXAM 테이블과 EXAM_ITEM 테이블은 문제의 내용과 각 문제의 객관식 항목들을 저장한다. 문제출제자의 정보를 저장하는 테이블과 문제풀이자의 정보를 저장하는 테이블, 각 문제지별 문제풀이자의 등록정보, 문제지 정보를 저장하는 테이블로 구성된다.

문제번호	QUESTION_NUMBER	문제번호	QUESTION_NUMBER
문제내용	QUESTION_CONTENT	항목번호	ITEM_NUMBER
항목개수	ITEM_NUMBER	항목내용	ITEM_CONTENT
출력형태	ITEM_TYPE	문제지ID	EXAM_ID
문제지ID	EXAM_ID		

표1. 문제테이블과 문제항

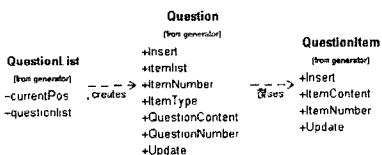
목테이블

표1은 문제출제부에서 사용되는 문제(EXAM) 테이블과 문제항목(EXAM_ITEM) 테이블을 보여준다. 문제테이블과 문제항목테이블의 경우 문제를 동적으로 출제함에 따라서 빈번하게 변동되는 사항들이 많은 내용들을 포함한다. 문제의 작성과 수정이 빈번하게 일어나게 되므로 데이터베이스를 매번 직접 액세스하는 것보다는 문제출제자가 삽입하거나, 수정하는 내용들을 저장하였다가 일괄 처리하는 방법을 택하였다.



[그림5] 데이터베이스연결을 위한 클래스 다이어그램

그림5는 데이터베이스 연결을 위해 사용되어지는 클래스들을 나타낸다. JDBC 연결을 위해 Database와 Results라는 두 개의 클래스를 FACADE Pattern을 사용하여 구현하며 DatabaseConnector 클래스가 두 개의 클래스를 사용하도록 구성하였다. DatabaseConnector 클래스를 이용하여 문제출제자가 기존에 작성한 문제들을 데이터베이스로부터 읽어와 그림6에 나타난 QuestionList 클래스에 저장한다.



[그림6] 문제와 항목을 저장하는 클래스 다이어그램

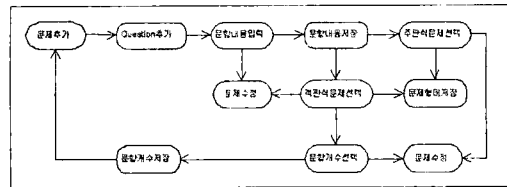
Question 클래스는 각 문제를 저장하고 있으며 문제별로 항

목을 저장하는 클래스인 QuestionItem의 리스트를 가지고 있다. Question 클래스에는 새롭게 추가되는 문제와 변경되는 문제를 구분하기 위해 Insert와 Update 플래그를 가지고 있다. 또한 QuestionItem에는 데이터베이스 저장시 각 항목의 변경사항을 확인하기위한 Update 플래그를 가지고 있다.

문제출제부에서는 문제출제자의 ID를 입력받아 각 문제출제자에 해당되는 문제지 이름을 데이터베이스로부터 읽어오게 되며 문제출제자는 문제지를 선택하여 문제를 출제할 수 있다. 문제출제자가 입력한 내용들은 표1의 문제테이블과 문제항목테이블에 저장된다. 문제출제완료후 문제출제완료버튼을 누르게 되면 문제출제자가 입력한 내용에 맞게 학습자의 답안을 저장할 수 있는 테이블이 작성된다. 문제의 형태에 따라서 학습자의 답안을 저장할 수 있는 테이블의 구성이 달라지기 때문에 문제출제가 완료된 후에는 문제를 수정할 수 없다.

3. 문제의 추가, 수정 및 저장

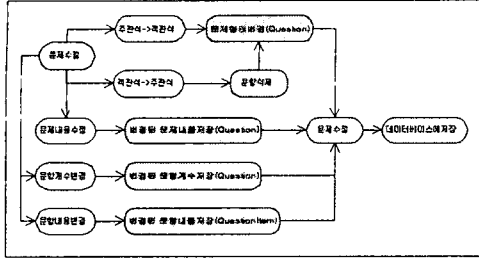
그림7은 문제출제자가 새로운 문제를 생성할 때 제어의 흐름을 나타내고 있다.



[그림7] 문제 추가 흐름도

문제출제자가 “새문제추가”라는 메뉴를 선택하게 되면, 문제출제자로부터 문제내용, 문제형태, 객관식의 경우 문항의 개수등을 입력받아 QuestionList에 저장하게 된다. 새롭게 추가되는 문제의 경우에는 그림6에서 나타난 Question 클래스의 Insert 플래그를 토글시켜 저장시킨다.

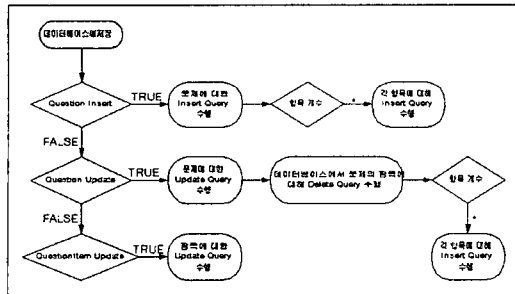
그림8은 문제출제자가 문제를 수정할 경우 문제출제부의 제어흐름을 나타낸다. 문제수정은 문제의 형태를 객관식에서 주관식으로, 주관식을 객관식으로 변경하는 경우와, 문제내용, 객관식의 경우 문항의 개수를 변경하는 경우, 문항내용을 변경하는 경우로 나누어 볼 수 있다.



[그림8] 문제 수정 흐름도

문제내용을 수정하는 경우, 문제형태를 변경하는 경우, 문항 개수를 변경하는 경우에는 Question 클래스의 Update 플래그를, 문항내용변경의 경우에는 QuestionItem 클래스의 Update 플래그를 토글시킨다. 객관식에서 주관식으로 문제의 형태를 변경할 때에는 문항의 내용을 삭제하고 변경된 문제의 형태를 Question 클래스에 저장한다.

그림9은 “문제저장”을 선택하였을 경우 문제출제자가 작성한 문제가 데이터베이스에 저장되는 흐름을 나타낸다.



[그림9] 데이터베이스 저장 흐름도

그림6에서 살펴본 QuestionList에는 문제출제자가 작성한 문제들이 저장된다. 또한 Question 클래스는 문제의 내용, 문제의 형태, 항목의 개수와, 문제항목리스트를 저장한다. Question 클래스에는 문제가 새롭게 추가된 문제인지, 기존문제를 수정한 것인지 알 수 있는 Insert 플래그와 Update 플래그를 가지고 있다. QuestionItem 클래스에는 항목내용이 변경되었다는 것을 알 수 있는 Update 플래그가 있다. 문제출제자가 작성한 문제들을 저장할 경우 먼저 QuestionList에 저장되어 있는 각 Question들에 대해 Insert 플래그의 토글 여부를 확인한다. 이 때 Insert 플래그가 참값을 가지게 되면 이 문제는 새롭게 수정된 문제임을 나타내며

로 이 문제에 대해서는 Insert Query를 이용하여 데이터베이스에 저장한다. 또한 이 문제의 항목개수를 조사하여 항목의 개수가 0 이 아닐 때, 즉 객관식 문제의 경우에는 문제의 각 항목에 대해서도 Insert Query를 수행하여 데이터베이스에 저장한다. Insert 플래그가 FALSE 값을 가질 경우에는 Question 클래스의 Update 플래그를 검사한다. 이 때 Update 플래그가 참값을 가지게 되면 Question의 내용들은 Update Query를 통해 데이터베이스에 저장된다. 다음 과정으로 데이터베이스에서 이 문제에 해당하는 항목들이 있는 경우 모두 Delete Query를 수행해 삭제시키고 항목의 개수를 조사하여 각 항목들에 대해 Insert Query를 수행한다. 마지막으로 QuestionItem의 Update 플래그를 검사하여 참값을 가지면 QuestionItem에 대하여 Update Query를 수행한다.

4. 결론

본 논문에서는 자바를 이용한 자동 문제 출제 시스템을 설계하고 구현하였다. 본 문제 출제 시스템은 문제 출제 과정의 번거로움을 없애고 손쉽게 대량의 문제를 출제할 수 있는 방법을 제공한다. 문제 출제 시스템에서 자바를 이용함으로써 유연한 사용자 인터페이스를 구현하였다.

향후 자바의 멀티미디어 지원 기술을 사용하여 현재의 문제 출제 시스템에 이미지, 동영상 등과 같은 멀티미디어 요소를 추가한 시스템으로의 확장이 필요하다.

[참고문헌]

- [1] 오민권, 윤여창, “웹을 이용한 온라인 학습 평가 시스템”, 한국정보처리학회, 2000 추계학술발표논문집 제7권 제1호, 2000
- [2] 박기석, 이재영, 김동한, “자바를 이용한 웹 기반의 문제 출제 시스템”, 한국정보과학회, 1999 가을 학술발표논문집 제26권 제2호, 1999
- [3] 오민권, 윤여창, “웹을 이용한 여론조사 시스템”, 한국정보처리학회, 1999 추계 학술발표논문집 제6권 제2호, 1999
- [4] 최돈은, 서현진, 박기석, 이재영, “동적인 문제 출제 시스템의 설계 및 구현”
- [5] JDBC, <http://java.sun.com/products/jdbc/>.