

# MVC 모델을 적용한 자동 문제 출제 시스템

이선숙 박문화<sup>o</sup>

성신여자대학교 컴퓨터정보학부  
everblue2@nate.com, mpark@sungshin.ac.kr

## Automated Selection of Test Questions using MVC Model

Seonsook Lee, Moonhwa Park<sup>o</sup>  
School of Computer Science & Engineering, Sungshin Women's University

### 요 약

학습자가 중심이 되고 선생님과 부모님들은 조력자 역할을 하는 자기주도 학습이 주로 이뤄지는 E-learning 환경에서 학습도구의 한가지인 자동 문제 출제 시스템의 역할은 매우 중요하다. 이런 학습 환경에서의 학습자의 참여도는 학습내용은 물론이고 화면구성이나 디자인에 따라서도 크게 좌우된다. 그러므로 화면구성과 디자인은 학습자의 관심과 흥미를 지속적으로 유지할 수 있도록 주기적으로 변경하여 학습자의 호응도를 높일 수 있게 구성하는 것이 바람직하다. 하지만 기존에 연구된 자동 문제 출제 시스템은 디자인과 프로그램이 분리되어 있지 않아 작은 사용자 인터페이스의 변화에 대처하는데 많이 비용이 소요될 수 있다. 본 논문에서는 SUN의 J2EE Patterns Catalog의 일부인 MVC 모델을 적용하여, 디자인과 프로그램을 분리하여 유지관리가 편리한 자동 문제 출제 시스템을 설계하였다.

### 1. 서 론

현재의 교육은 학습자 중심으로 이루어지고 있으며, 인터넷 사용의 일반화로 교육 분야에도 적극적으로 인터넷을 활용하고 있다. 이러한 흐름과 맞물려 학습 콘텐츠의 표준화 작업, 맞춤형 학습시스템 등의 E-learning 연구가 활발하게 진행되고 있다.

E-learning은 오프라인에서 이루어지는 교육활동에 비해 학습자의 참여도에 따라 학습효과가 크게 좌우된다. E-learning 환경에서 학습자가 가장 먼저 접하게 되는 것은 화면 디자인을 포함한 UI(User Interface)이며, 매력적(attractive)으로 구성된 UI는 학습자의 호응도를 높여 학습 동기유발의 요인으로 작용할 수 있다. 이때 UI는 시대적 상황이나 환경을 고려한 디자인이 융통성 있게 적용될 수 있어야 하는데, 이러한 특성은 UI의 변경이 빈번하게 발생할 수 밖에 없음을 시사한다. 이때, 빈번한 UI변경은 개발 생산성 및 프로그램 재사용성 저하, 프로그래머와 디자이너 간의 불화 야기, 시스템 유지·관리의 비효율성 등이 예상된다. 따라서 SUN의 J2EE Patterns Catalog 일부인 MVC 모델의 적용을 해결방안으로 제안한다.

본 논문에서는 LMS (Learning Management System)에서 사용하는 평가시스템 중, 자동 문제 출제 시스템에 MVC 모델을 적용한다. 그러므로 UI와 프로그램이 분리되지 않았던 기존의 시스템과는 달리, 개발된 코드의 재사용성을 저하시키는 문제를 해결하고, 디자이너와 프로그래머 간의 업무 분장 요소 제거한다. 그러므로 시스템을 보다 효과적이고 능률적으로 유지·관리 할 수 있는 시스템을 설계한다.

설계에 사용되는 개발 방법론에 활용 된다. Model은 응용 프로그램의 내부 상태, 즉 비즈니스 자료(business data)나 비즈니스 로직(business logic)과 같은 것을 나타낸 것으로 어플리케이션의 데이터를 표현하고, 데이터에 접근을 제어하는 작업을 수행한다. View는 Model의 콘텐츠를 어떻게 보여주는가 하는 규정으로, 클라이언트에 출력되는 사용자 인터페이스와 관련이 있다. Controller는 어플리케이션의 동작을 정의한다. 사용자의 입력(input)과 어플리케이션 프로그램 간의 상호작용 처리와 데이터 저장과 조회 등과 같은 Model과의 상호작용도 처리한다. [1][2][3][4]

MVC 모델의 핵심은 비즈니스 로직을 처리하는 Model과 사용자에게 결과화면을 보여주는 View가 분리되어 있으며, 어플리케이션의 흐름 제어나 사용자의 처리 요청은 Controller에 집중되어 있다는 것이다. 다시 말하면 View와 Model이 완벽하게 분리되어 있기 때문에 Model의 내부 로직이 변경되어도 View는 영향을 받지 않으며, 또한 Model이 결합되어 있지 않기 때문에 View를 쉽게 변경할 수 있다.[3][5]

### 2.2 자동 문제 출제 시스템

자동 문제 출제 시스템이란 방대한 양의 문제들을 체계적으로 분류하여 데이터베이스에 등록해 놓고 필요시 일부의 문제를 자동으로 추출하여 학습자의 학습 성과 평가에 이용할 수 있도록 하는 시스템이다.

초기의 문제 출제 시스템은 고정 출제나 무작위 출제 방식으로 교수자의 출제 의도가 거의 반영되지 못하였지만, 현재는 이 분야의 연구가 활발히 진행되어 기존의 시스템에 비해 교수자의 의도가 잘 반영되고 있다. 예를 들면, 초기 문제 출제시 문제의 난이도를 부여하고, 학습자의 테스트 결과를 난이도 재조정된 근거자료로 사용하는 것처럼 피드백 학습 방법을 통해 교수자가 기대하는 난이도를 반영할 수 있도록 한다. 또, 학습자의 테스트

### 2. 관련 연구

#### 2.1 MVC 모델

MVC 모델은 Model, View, Controller와 같이 각 계층 간의 결합력이 최소화 되도록 어플리케이션 영역을 나눈 것으로 Web-Tier Application을 포함한 소프트웨어

결과를 바탕으로 학습자의 학습 패턴을 분석하고, 그 분석 결과를 이용하여 학습자에게 맞는 문제가 출제될 수 있도록 하는 다양한 알고리즘이 연구 발표되고 있다.[6][7][8][9]

거쳐 문제 관리, 문제 출제, 회원 관리에 접근이 가능하다. 채점 및 평가는 시스템에서 자동적으로 수행되므로 사용자가 관여할 필요는 없다.

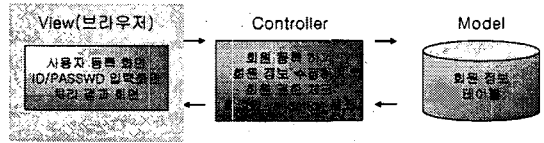
### 3. MVC 모델을 이용한 자동 문제 출제 시스템 설계

#### 3.1 기존 자동 문제 출제 시스템의 문제점

기존의 시스템은 웹 디자이너가 설계한 사용자 인터페이스 템플릿과 각종 프로그래머가 개발한 각종 비즈니스 로직과 스크립트 언어가 한 화면에 공존한다. 이러한 경우 정적 언어(HTML)의 한계로 인하여, 화면 디자인과 UI 구성에 맞춰 프로그램 개발이 이루어질 수 밖에 없다. 그러한 까닭으로 웹 디자이너와 프로그래머간의 충돌이 야기될 수 있는 문제와, 객체지향 프로그래밍 언어가 추구하는 프로그램 재사용성이 떨어뜨리는 문제가 발생할 수 있다.

이 문제를 해결하기 위해 많은 연구가 이루어지고 있으며, 그 연구들의 대부분의 결과는 UI와 프로그램 분리이다. 본 논문에서는 기존 시스템의 문제 해결 방안으로 J2EE Patterns Catalog의 일부인 MVC 모델을 제시적용하여 UI와 프로그램을 분리시켜, 디자이너와 프로그래머 간의 분쟁요소를 없애고, 프로그램의 재사용성을 높일 수 있는 자동 문제 출제 시스템을 설계한다.

#### 3.2.1 회원 인증 모듈



[그림 2] MVC 모델을 적용한 회원 인증 모듈

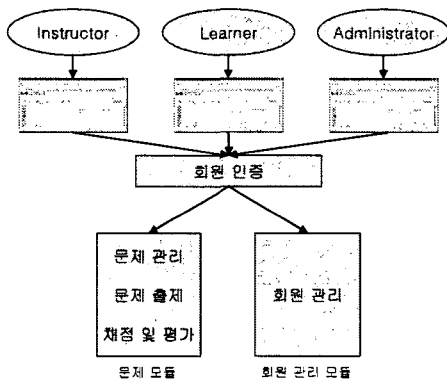
회원 인증모듈은 회원 가입과 로그인을 처리하는 모듈로 사용자 인터페이스를 담당하는 사용자 입력화면 (view)과 이를 처리하는 업무프로세스(controller), 그리고 그 데이터를 관리하는 테이블(Model)로 구성한다. 본 시스템은 회원가입과 인증을 거쳐야만 사용 가능한 시스템이다. 사용자가 ID와 Password를 입력하면 controller에서 로그인 체크를 하고, 또 사용자의 접근 권한을 세팅한다. 최고 관리자(supervisor)는 시스템 구축 시 등록하도록 하여 별도의 가입 절차가 필요 없도록 하지만, 그 외의 모든 사용자는 관리자의 승인을 거쳐 회원으로 등록한다.

#### 3.2 MVC 모델을 적용한 자동 문제 출제 시스템 설계

본 논문에서 설계하는 시스템은 웹을 기반으로 자동 문제 출제 시스템으로 다음과 같다. UI는 별도의 클라이언트가 필요 없이 웹 브라우저만으로도 접근 가능하도록 하고, 자동 문제 출제 시스템의 알고리즘은 피드백 학습 제공으로 신뢰성을 높인 [6]을 따르며, 개발하는 방법에는 MVC 모델을 채택한다.

#### 3.2.2 문제 모듈

[표 1] 문제 테이블



[그림 1] 시스템 전체 구성도

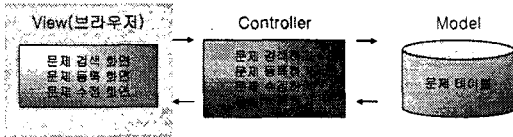
이 시스템은 [그림1]과 같이 시스템을 크게 사용자 (Instructor, Learner, Administrator), 회원인증 모듈, 문제 모듈, 회원관리 모듈로 나누었다. 사용자의 권한에 따라 각 모듈의 하위 평가의 접근 권한이 정해진다.

먼저 교수자(Instructor)는 회원 인증을 거쳐 문제 출제, 문제 관리를 사용할 수 있으며, 학습자(Learner)는 회원 인증을 거쳐 문제 출제를, 그리고 관리자는 회원 인증을

필드	필수 여부	비고
문제 ID	필수	테이블의 primary key 값으로 sequence 사용
문제 과목	필수	과목 코드
문제 단원	필수	단원 코드
문제 내용	필수	문제 내용
보기 1	필수	문제의 보기1
보기 2	필수	문제의 보기2
보기 3	필수	문제의 보기3
보기 4	필수	문제의 보기4
보기 5	필수	문제의 보기5
문제 정답	필수	문제의 정답 번호
문제 해설	선택	학습자의 이해증진을 위한 문제해설
난이도	필수	자동 문제 출제 시 참조할 정보로 문제의 난이도 저장 (상, 상중, 중, 중하, 하)를 (A, B, C, D, E)로 저장
총 응시 수	필수	해당 시점까지 출제된 횟수 (누적 값)
정답 수	필수	해당 시점까지 학습자가 정답을 맞춘 횟수 (누적 값)
문제 등록자	필수	문제를 등록한 사람의 USER ID
문제 등록일	필수	문제를 등록한 날짜 및 시간
문제의 상태	필수	문제사용 여부로 등록 및 사용 가능 상태는 'Y', 해당 문제 삭제 시 'N'으로 변경

문제 모듈은 '문제 관리', '문제 출제', '채점 및 평가'로 구성하며, 그 내용은 다음과 같다.

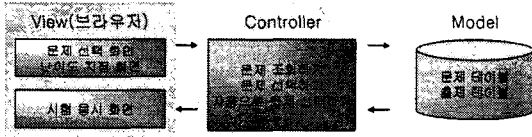
1) 문제 관리



[그림 3] MVC 모델을 적용한 문제 관리 모듈

문제의 등록/조회/수정/삭제 등을 관리하는 모듈로 관리자와 교수자만 접근가능하다. 문제 등록 시 문제 테이블(model)에서 요구하는 데이터를 사용자 인터페이스를 담당하는 브라우저(view)에서 입력하는데, 이때 교수의 판단에 따라 난이도를 상, 상중, 중, 중하, 하의 5 단계로 나누어 입력하도록 한다.[6][8] 문제 조회의 조건은 과목, 단원, 키워드, 난이도 등으로 사용자 인터페이스(view)를 구성하도록 한다. 이러한 사용자 인터페이스를 지원하는 업무 프로세스는 controller에서 구현하며, 그 업무 프로세스는 검색, 등록, 수정, 삭제로 문제 테이블의 상태를 변화시키는 역할을 한다. 여기서 문제의 수정 및 삭제 시 발생할 수 있는 권한은 해당 문제의 등록자와 관리자로 제한하도록 한다.

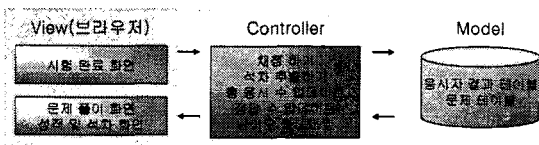
2) 문제 출제



[그림 4] MVC 모델을 적용한 문제 출제 모듈

문제 출제는 교수가 임의로 선택하여 시험지를 구성하는 출제자 지정방식과 난이도를 부여하여 시스템이 시험지를 구성하는 자동 문제 출제 방식을 모두 지원한다. 출제자 지정방식은 사용자가 교수자와 관리자만 사용이 가능한 화면으로 구성하여, 그 사용자들이 문제를 직접 선택하여 출제테이블에 등록한다. 자동 문제 출제 시스템의 경우엔 모든 사용자가 사용할 수 있으며, 이는 사용자가 입력한 예상평균 점수에 따라 5단계 난이도로 구성된 문제를 적절하게 조합하여 출제테이블을 구성하도록 한다.[6]

3) 채점 및 평가



[그림 5] MVC 모델을 적용한 채점 및 평가 모듈

채점 및 평가 모듈은 시험 완료 액션이 발생하면 시스템이 자동으로 채점을 하고, 그 결과를 바로 보여주는 역할을 한다. 이때 시스템 내부적(controller)으로 결과테이블과 문제 테이블에 데이터를 입력/수정 한다.

평가를 하기 위해 각 응시자의 결과테이블에 점수와 석차 정보를 입력하고, 난이도 재조정을 위해 문제 테이블에서 각각의 문제에 대해 총 응시 수와 정답 수를 업데이트하고, 이 정보를 토대로 난이도 정보도 업데이트하도록 시스템을 구현한다.

3.3 MVC 모델의 적용 결과

웹 환경에서의 자동 문제 출제 시스템에 MVC 모델을 적용하여 설계 및 개발을 하면, UI와 비즈니스 로직이 자연스럽게 분리되는 것을 알 수 있다. UI와 비즈니스 로직이 분리되면, 디자인을 포함한 UI가 변경된다 하더라도 비즈니스 로직은 그대로 사용할 수 있으며, UI가 그대로 유지되고 비즈니스 로직이 변경된다면 비즈니스 로직만을 새롭게 적용하면 되므로 개발 생산성 향상은 물론이고 프로그램의 재사용성도 증대시킬 수 있다.

4. 결론 및 향후 연구방향

E-learning 환경에서 학습자의 참여도는 학습내용은 물론이고, 화면구성이나 디자인에 따라서도 크게 좌우된다. 그러므로 화면구성과 디자인은 학습자의 관심과 흥미를 지속적으로 유지할 수 있도록 주기적으로 변경하여 학습자의 호응도를 높일 수 있게 구성하는 것이 바람직하다. 본 논문에서는 MVC 모델을 적용하여, 디자인과 프로그램을 분리하는 자동 문제 출제 시스템을 설계하였다. 그 결과, 개발 생산성 향상과 프로그램 재사용성을 증가시켜, 유지 관리를 편리하게 하는 성과를 얻을 수 있었다. 앞으로 소프트웨어 구조뿐만 아니라 하드웨어를 구조를 설계함에 있어서도, 2-tier가 아닌 Multi-tier로 구성하여, 서버의 확장 및 유지관리에 기여할 수 있는 시스템을 연구할 예정이다.

5. 참고문헌

- [1] Rich Green, "Design Enterprise Applications with the J2EE Platform, Second Edition", Sun Microsystems, 2002
- [2] James W. Cooper, "Java Design Patterns", Addison-Wesley, 2002
- [3] 유주현, "MVC 기반의 웹 컴포넌트 컨테이너 시스템에 관한 연구", 순천대학교 대학원 이학석사학위논문, 2003
- [4] 김철민, "Object Pool 패턴을 이용한 WIPI기반 MVC모델 개선에 관한 연구", 전남대학교 대학원 석사학위논문, 2005
- [5] Andrew Hessey and David Carington, "Comparing Two User-Interface Architecture : MVC and PAC", Software Verification Research Center, 1996
- [6] 김경아, "웹 기반 자동 문제 출제 시스템", 동국대학교 교육대학원 석사학위논문, 2000
- [7] 이현주, 이미숙, 홍승미, 이찬희, 정순호, "학습자의 수준평가를 이용한 웹 기반 자동 문제 출제 시스템", 한국정보처리학회 논문지, 2003
- [8] 김은정, "웹기반 학습시스템의 평가 문제에 대한 출제 방법 및 난이도 재조정에 대한 연구", 한국정보처리학회, 2005