

유 원 경 교수지도

석사학위 청구논문

# XML기반 온라인 수업관리 System

2005

성신여자대학교 교육대학원

교육학과 전자계산교육전공

안 미 숙

# XML기반 온라인 수업관리 System

유 원 경 교수지도

이 논문을 석사학위논문으로 제출함.

2005년 5월

성신여자대학교 교육대학원

교육학과 전자계산교육전공

안 미 숙



# 인 준 서

안 미 숙의 석사학위 논문으로 인준함.

심사위원 \_\_\_\_\_ (인)

심사위원 \_\_\_\_\_ (인)

심사위원 \_\_\_\_\_ (인)

성신여자대학교 교육대학원



## 논문 개요

오늘날, XML을 이용한 연구가 활발히 이루어지고 있으며, 이러한 연구들 중에서 교육에 관련된 연구는 최근 몇 년 동안 많이 이루어졌다. 교육에 관련된 연구들 대부분이 아직은 HTML을 단순히 XML로 대체해서 교육 콘텐츠를 웹에서 제공하는 수준이거나 또는 XML을 이용하여 교육시스템을 구현하더라도 XML Schema보다는 DTD를 그리고, 관계형 데이터베이스를 많이 이용하고 있다. 그러나, 기존 DBMS기반을 이용해 XML 데이터를 저장하면 XML문서를 관계형 모델 혹은 객체지향 모델로 변환하여 시스템에 저장해야 하므로 impedance mismatch문제가 있고, 여러 테이블에 동일한 데이터가 중복될 수 있다.

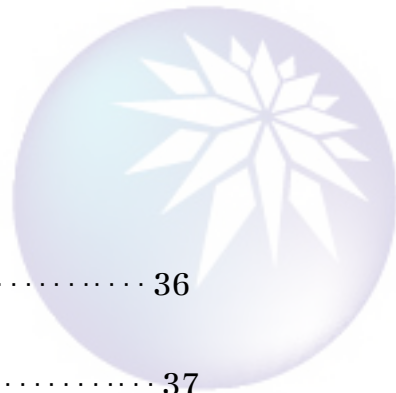
본 논문에서는 교수가 강좌를 수강하는 학생에게 강좌에 대한 강의자료, 과제물, 시험 등에 대한 정보를 제공하고, 학생은 제공되는 정보를 통해 강좌에 대한 정보도 얻고, 과제물도 제출하고, 시험도 볼 수 있다. 이런 과정을 통해 교수와 학생이 서로 강좌에 대한 상호작용을 한다. 이와 같은 교육시스템에 관련된 문서를 효과적으로 통합 관리, 공유, 검색 할 수 있도록 XML과 XML Schema 그리고, XML 문서 관리의 효율성을 높이기 위해 XML 전용 데이터베이스인 eXcelon을 이용하여 XML기반 온라인 수업관리 System을 구현하였다.



# 목 차

## 논문개요

제 1 장 서론 .....	1
제 2 장 관련연구 .....	4
제 3 장 수업관리 System 분석 .....	9
3.1. 시스템 전반적인 개요 .....	9
3.2. 개발 환경 .....	21
제 4 장 수업관리 System 설계 .....	24
4.1. 전체 구조 .....	24
4.2. 사용자 설계 .....	25
4.2.1. 학생 .....	26
4.2.2. 교수 .....	28
4.2.3. 관리자 .....	30
4.3. XML Schema 설계 .....	31
4.3.1. 학생 .....	32
4.3.2. 교수 .....	32
4.3.3. 교과목 .....	33
4.3.4. 수강 .....	33
4.3.5. 강좌 .....	34
4.3.6. 과제물 .....	34
4.3.7. 과제물 제출 .....	34
4.3.8. 강의자료 .....	35
4.3.9. 시험지 .....	35



4.3.10. 답안지 .....	36
제 5 장 수업관리 System 구현 .....	37
5.1. 학생 .....	37
5.2. 교수 .....	40
5.3. 관리자 .....	45
제 6 장 결론 및 향후 연구방향 .....	48

참고 문헌 및 사이트

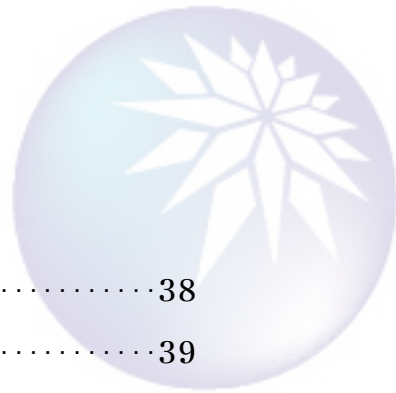
ABSTRACT

부록



## 그림 목 차

[그림 3-1] 학생 흐름 .....	10
[그림 3-2] 학생 메뉴들의 흐름 .....	11
[그림 3-3] 교수 흐름 .....	13
[그림 3-4] 강의계획서메뉴 흐름 .....	13
[그림 3-5] 강의자료메뉴 흐름 .....	14
[그림 3-6] 과제물메뉴 흐름 .....	15
[그림 3-7] 출석부메뉴 흐름 .....	16
[그림 3-8] 시험메뉴 흐름 .....	17
[그림 3-9] 성적메뉴 흐름 .....	18
[그림 3-10] 관리자 흐름 .....	18
[그림 3-11] 학생과 교수 흐름 .....	19
[그림 3-12] 교과목과 강좌 흐름 .....	20
[그림 4-1] XML기반 수업관리 System 전체 흐름 .....	25
[그림 4-2] 학생 메뉴 .....	26
[그림 4-3] 교수 메뉴 .....	28
[그림 4-4] 관리자 메뉴 .....	30
[그림 4-5] 전체 데이터구조 .....	32
[그림 5-1] 교과목 목록을 보여주는 화면 .....	37
[그림 5-2] 강의자료 목록을 보여주는 화면 .....	37
[그림 5-3] 첨부된 강의자료 파일 다운로드 하는 화면 .....	38



[그림 5-4] 과제물 제출 화면	38
[그림 5-5] 시험 보는 화면	39
[그림 5-6] 성적확인 화면	39
[그림 5-7] 강의계획서 화면	40
[그림 5-8] 강의자료 등록 화면	41
[그림 5-9] 강의자료 관리 화면	41
[그림 5-10] 과제물내용 변경화면	42
[그림 5-11] 과제물채점 화면	42
[그림 5-12] 시험지작성 화면	43
[그림 5-13] 시험지관리 화면	43
[그림 5-14] 성적입력 화면	44
[그림 5-15] 출석부 화면	44
[그림 5-16] 교과목목록 화면	45
[그림 5-17] 교과목등록 화면	45
[그림 5-18] 교과목관리 화면	46
[그림 5-19] 학생관리 화면	46
[그림 5-20] 학생정보 화면	47
[그림 5-21] 강좌목록 화면	47



# 표 목 차

[표 3-1] 개발 환경 .....23

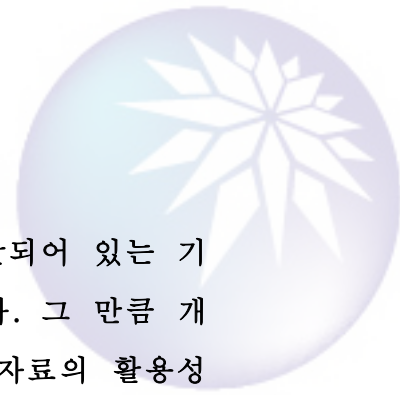


## 제 1 장 서론

오늘날의 교육환경은 인터넷과 컴퓨터 기술의 발달로 예전처럼 교실에 학생들을 모아 놓고 교사가 수직적으로 지식을 전달하는 정적인 교육이 아니라, 학생들과 교사가 교실뿐만 아니라 교실 외의 공간 즉, 사이버 공간에서도 서로 대화하고 정보를 주고 받을 수 있는 동적인 교육을 요구하는 방향으로 흘러가고 있다. 즉, 교육이 장소를 초월해서 이루어진다. 요즘 초고속 인터넷을 활용한 사이버대학, 원격연수, 웹 기반 코스웨어, 온라인 학습지 등이 개발되어 활용 중이고, 좋은 반응을 보이고 있다. 사교육비 절감대책으로 교육인적자원부에서 추진하고 있는 교육방송의 웹 기반 VOD 수능 강의도 많은 학생들이 회원으로 가입하여 활용되고 있다.

웹 기반 교육은 개개의 학습자를 중심으로 학습활동이 이루어진다는 점에서 교실수업에 비해 효율성을 보장해 줄 뿐만 아니라 학습자의 개인차를 반영한 자기주도적 개별학습의 학습도구로써 활용될 수 있다. 학습자의 학습상황을 점검하고 학업성취도를 파악하여 학생의 수준에 맞는 학습 자료를 제공하는, 즉 교사의 역할을 대리하는 교사에이전트 기능이 부가된 웹 기반 지능형 교육시스템이 운영되고 있다[11]. 그러나 교사에이전트의 기능을 뛰어나게 구현한다고 해도 교사의 역할을 다 할 수는 없다.

제7차 교육과정에서는 학교 교육과정 운영 시 교과용 도서 이외에 교육방송, 시청각 교재, 각종 학습자료 등을 활용할 것과 교과 중심의 교육에서 탈피하여 컴퓨터를 중심으로 한 정보 통신 기술 매체를 활용

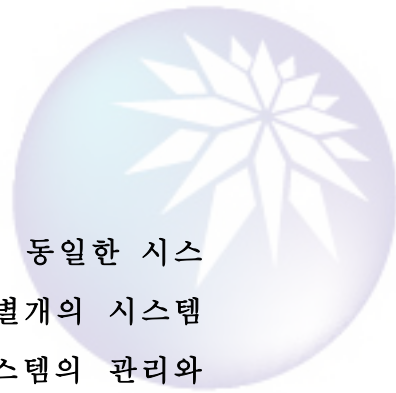


교육이 활성화되도록 하였다[13]. 그러나 현장에서 개발되어 있는 기존의 학습자료도 수업에 활용되지 못하고 있는 실정이다. 그 만큼 개발된 학습자료에 대한 관리가 소홀해지고, 개발된 학습자료의 활용성도 떨어질 수 밖에 없다.

이러한 시점에서 좀더 효율적이고, 편리하게 사이버 공간에서 학생들과 교사가 수업에 대한 정보를 주고 받을 수 있는 방법을 생각하지 않을 수 없다. 그리고, 이는 데이터 관리를 학교 운영 시스템에 맞게 의미를 부여하여 자유롭게 구성할 수 있어야 하고, 후에 저장 데이터를 검색, 통합, 변경 등을 함에 있어 효율성 및 편리성이 또한 보장되어야 한다.

이런 면에서 XML(Extensible Markup Language)문서는 유용하다. XML문서는 문서의 내용과 스타일을 분리함으로써 모든 컴퓨팅 환경에서 XML문서를 공통적으로 읽을 수 있고, 문서를 생성한 플랫폼이나 응용프로그램에 종속될 필요가 없다. XML이라는 표준화된 기술은 기업들간에 데이터 교환이 쉬워지고, 하나의 XML문서에 다양한 스타일을 적용함으로써 다양한 형태의 문서를 얻을 수 있다[15]. XML Schema는 문서의 구조와 내용을 제한하는 메커니즘을 제공하고, 문서의 클래스들에 대해 더 강력한, 또는 더 느슨한 제한을 허용하며, 복수의 네임스페이스에 속해 있는 마크업들로 구성된 문서에 대한 유효성 검증 능력을 가지고 있다. 또한 요소, 속성 및 데이터형 정의를 상속할 수 있게 하는 메커니즘을 가지고 있고, 원시 데이터형을 지원한다[20].

이와 같은 기술들을 수업관리 System에 도입하여, 검색·통합·변경 등 문서 관리를 편리하고 효율적으로 하기 위해 XML문서를 database에 저장할 필요가 있으며, 여기에 이를 적용시킨 수업관리 System이 필요하다.



이러한 수업관리 System을 이용한다면, 교사와 학생이 동일한 시스템을 함께 활용함으로써 긴밀한 상호작용이 형성되어, 별개의 시스템으로 분리되어 있을 때보다 수업의 질이 향상되고, 시스템의 관리와 학습문서의 업그레이드가 원활히 이루어질 것이다.

본 논문에서는 온라인 상에서 보다 효과적으로 학생과 교수가 상호작용을 하고, 그 상호작용의 결과인 데이터들을 관리하고자 XML기반 온라인 수업관리 시스템을 구현하고자 한다. 이를 통해서 문서를 효율적으로 통합 관리, 공유할 수 있도록 문서를 논리적으로 기술하는 XML Schema를 적용시킨 XML문서를 만들어 데이터를 저장하고, 그 문서를 XML전용 데이터베이스인 eXcelon에 저장한다.

본 논문의 구성은 다음과 같다. 제1장에서는 연구의 목적 및 필요성에 관한 내용을 기술하고, 제2장에서는 수업관련 시스템의 구현에 관련된 연구를 기술하며, 제3장에서는 XML기반 온라인 수업관리 시스템의 구현에 관한 분석을 기술한다. 제4장에서는 분석의 결과를 갖고, 시스템을 설계하고, 스키마에 대한 자세한 내용은 부록을 참조한다. 제5장에서는 설계를 이용해서 시스템을 구현하다. 마지막으로 제6장에서는 결론 및 향후 연구방향을 제시한다.



## 제 2 장 관련 연구

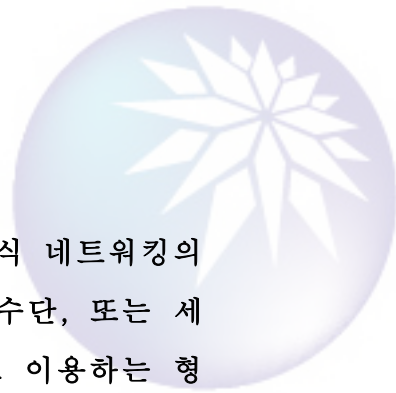
웹 기반 수업은 최근에 각광을 받고 있는 인터넷을 교수 도구로 하여 학습이 이루어지는 교수모형으로 Relan과 Gillani은 인터넷의 자원과 속성을 활용하는 건설적이고, 협력적인 학습 환경 내에서의 교육적 전략에 기초한 인식의 적용이라 하고 있으며, Clark에 의하여 WBT(Web Based Training)는 ‘공용 또는 개인 컴퓨터 네트워크에 의해 전달되고 브라우저에 의해 디스플레이 되는 개별적인 교수’라고 정의된다[3][6].

새로운 교육 방법의 형태로서 지금까지의 교실 수업과는 다른 교수 모형인 웹 기반 수업은 특정한 그리고, 미리 계획된 방법으로 학습자의 지식이나 능력을 육성하기 위한 의도적인 상호작용을 웹을 통해 전달하는 활동이라고 정의 내릴 수 있다.

다양한 매체 중심의 교육을 제공하기 위해 웹 브라우저와 대중들의 인터넷 접속을 확대 시킨다는 장점을 지니고 있는 웹 기반 수업은 어떠한 주제이건 학습자 주도적이고 학습자의 속도에 맞는 교수법을 제공한다.

웹 기반 수업의 발달은 컴퓨터 네트워크 공학의 발달과 교육적 활용에 그 바탕을 두고 있다고 볼 수 있다. 컴퓨터 네트워크가 교육에 활용된 형태는 크게 세 가지 정도로 구분 된다[16].

첫째, 컴퓨터 네트워크가 면대면 교육이나 원격교육 등에서 하나의 보조적 매체로 활용되는 형태가 있으며, 둘째로는 컴퓨터 네트워크가 전체 강좌나 강좌의 일부를 가르치는 주된 매체로 활용되는 형태가 있



다. 셋째 형태로는 컴퓨터 네트워크를 보다 자유로운 지식 네트워킹의 장, 토론에서의 참여수단, 온라인 데이터베이스 활용의 수단, 또는 세계의 흩어진 전문가나 동료들과의 정보교환 수단 등으로 이용하는 형태이다. 컴퓨터 네트워크가 수업을 위한 하나의 이용 매체로 이용되는 경우, 서로 다른 지역에 있는 교실들이 연결되어 자료 교환, 사회적 상호작용, 공동 프로젝트의 진행 등에 이용되는데, 이는 초·중·고등학교는 물론 대학, 성인 교육에 이르기까지 교육에 컴퓨터 네트워크를 가장 널리 활용하는 방식이라고 하겠다. 사회학, 컴퓨터 과학, 통계학 등 다양한 강좌에서 시도된 컴퓨터 네트워크 수업은 학업 성취 면에서 기존 교실 수업의 경우와 차이를 보이고 있지 않으며, 오히려 학생들의 높은 만족도를 보여주고 있다.

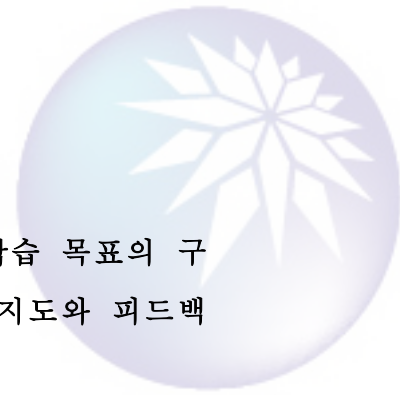
이 형태의 수업에서는 통신망을 통해 수업내용이 전달되고, 시험을 치르게 되며, 학생들은 E-mail, 게시판, 컴퓨터 컨퍼런싱 등을 통해 상호작용을 하게 된다.

웹 기반 수업에서는 다양한 멀티미디어의 상호작용을 통한 학습이 진행 되는데, 웹 기반수업의 교수설계 환경은 학습자, 수행, 상호작용 멀티미디어 프로그램의 세 요소로 구성된다고 볼 수 있다[18].

이 세 가지 구성 요소들은 각각 학습자, 교사, 문서의 역할과 책임을 가지며, 그 중에서 문서는 학습자와 교사를 연결하는 매체 역할을 한다. 웹 기반 수업의 교수 설계 시 이 세가지 요소들의 역할과 전략이 고려되어야 한다.

교수는 특정 학습을 신장시키기 위해 정보들을 신중히 조직하고 진술하는 활동이다. 그러므로 웹 기반 수업의 설계에 있어서도 교수 설계의 여러 이론들을 고려하여 적절한 설계가 이루어져야 한다.

Dick & Reiser의 교수설계이론은 교수설계를 하는데 있어 대체로 7



가지의 원리가 적용된다. 그 원리는 학습자의 동기화, 학습 목표의 구체화, 선행학습의 상기, 적극적인 교수 활동 참여, 학습지도와 피드백 제공, 평가, 학습강화 및 교정 제공이다[1].

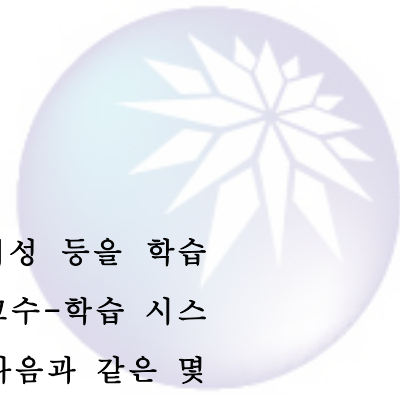
Dick & Carey의 체계접근 교수 설계 원리는 효과적인 교수 프로그램을 개발하기 위하여 교수 개발 과정이 하나의 체제과정(the process of system)처럼 각 부분 또는 단계의 유기적 상호작용을 통해서 일차적인 교수 프로그램을 산출하고, 그 다음 그 프로그램을 평가하여 피드백하고 피드백에 따라 프로그램을 수정하여, 최종 결과물로서의 교수 프로그램이 교수 요구나 교수 문제해결을 보장케 하려는데 있다[2].

웹 기반 교수-학습에 관련된 연구는 개인용 컴퓨터 보급과 인터넷의 발달로 웹 상에서 누구나 간단한 절차만으로 시간과 장소에 구애 받지 않고, 멀티미디어 자료를 손쉽게 활용함으로써 현장수업에서의 어려움을 충분히 보완할 수 있는 새로운 교수-학습 형태로 웹을 기반으로 한 교육(WBI, Web-Based Instruction)이라는 새로운 교수-학습 모형을 탄생시켰다.

웹 기반 교육이란 하이퍼미디어를 기반으로 하는 교육 프로그램으로서 학습자의 지식이나 능력을 육성하기 위한 의도적인 상호작용을 위해 WWW(World Wide Web)의 특징과 자원 등을 교육에 활용하는 것이라 정의할 수 있다[4].

Web을 이용해서 교수-학습 활동을 할 때에는 다른 교수매체에 비해 여러 측면에서 유용하다.

첫째, 전달되는 정보들의 역동성과 상호작용성을 들 수 있다. 둘째, Web자체가 하나의 커다란 정보 제공원이 된다. 셋째, Web을 이용해서 교수활동을 해 나갈 때에는 정보나 자료를 수시로 수정 및 보완할 수 있다[5].



WBI 학습은 정보의 다양성, 역동성, 상호작용성, 즉시성 등을 학습자에게 제공하는 장점이 있다. 그러나 현재의 웹 기반 교수-학습 시스템은 학습을 위한 웹 문서의 선정과 교수-학습 설계에 다음과 같은 몇 가지 문제점을 안고 있다.

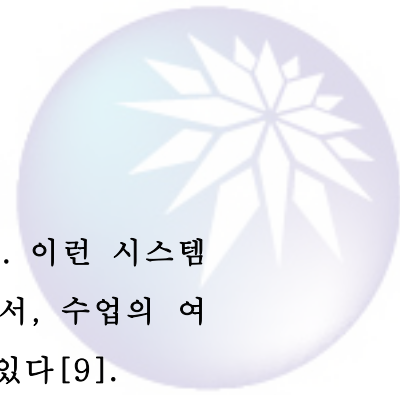
첫째, 교수자에 의해 선정된 웹 문서는 효용가치가 매우 높음에도 이를 체계적으로 관리하는 도구가 존재하지 않아 같은 문서를 반복해서 검색하거나 제작을 한다. 둘째, 교수자가 학습 목표에 적합한 웹 문서를 검색, 제작한 후 학습자에게 제시하기 위해서는 별도의 교수 설계 과정이 필요하다. 셋째, 학습자는 학습 결과와 웹 문서의 관련성 분석이 매우 어렵다[7].

이와 같이, 웹 기반 교수-학습은 다양한 학습 자료를 제공하여 학습자의 수준별 학습을 도울 수 있는 하나의 효과적인 교수-학습 매체로 활용이 되지만, 지나친 학습 자료의 개방으로 인하여 학습자는 지식획득은 선행된 학습내용을 기반으로 하여 이루어진다는 구성주의 학습이론에 비추어 볼 때 학습자의 학습활동을 저하시킬 수 있으며, 학습자들로 하여금 학습목표를 상실시키는 요인이 될 수도 있다[10].

지금까지 구현된 온라인상에서의 수업에 관련된 시스템들을 살펴보면, 다음과 같은 여러 부족한 부분들이 있다.

첫 번째는 수업자료를 단순히 HTML을 이용해 작성한 후 웹에 제공하는 경우이다. 이런 시스템은 교사와 학생들의 상호작용이 전혀 이루어질 수 없고, 서로의 의견 또한 주고 받을 수 없으므로 학습자 주도의 열린 수업을 할 수 없다. 그리고, 제공된 자료들이 데이터베이스에 저장되어 있지 않아 자료의 활용도가 떨어진다. 즉, 한정된 내용의 자료만 웹 페이지로 만들어 단순히 수업자료만 제공하고 있다[8][14].

두 번째는 프로그래밍 언어와 데이터베이스를 이용해 시스템을 구현



하더라도 한 가지의 한정된 기능만을 제공하는 경우이다. 이런 시스템은 교사와 학습자 간의 상호작용 방법을 제한하고 있어서, 수업의 여러 정보를 제공하고 공유하는 등의 여러 활동에 제약이 있다[9].

세 번째는 시스템의 효율성에 대한 고려가 부족한 경우이다. 이 경우, 데이터의 유효성 검증이나 중복 체크, 그리고 시스템의 수행속도가 떨어진다[12].

본 논문에서는 위에서 살펴본 기존의 시스템이 갖는 여러 부족한 점들을 개선하고, 웹 기반 학습시스템의 효율성을 높이기 위해 XML과 XML Schema를 사용하며, 또한 XML 전용 데이터베이스를 통해 시스템의 수행속도와 효율성을 높이도록 한다. 웹 기반 학습시스템에서 무엇보다 중요한 교사와 학습자 간의 상호작용을 원활히 하기 위해 교사가 학생에게 학습에 대한 자료와 과제물, 시험, 성적 등을 제공하고 이를 관리하면, 학생은 교사가 제공한 학습자료와 과제물, 시험 등을 통해 해당 교과목에 대해 서로 상호작용을 할 수 있도록 설계하고 구현한다.



## 제 3 장 수업관리 System 분석

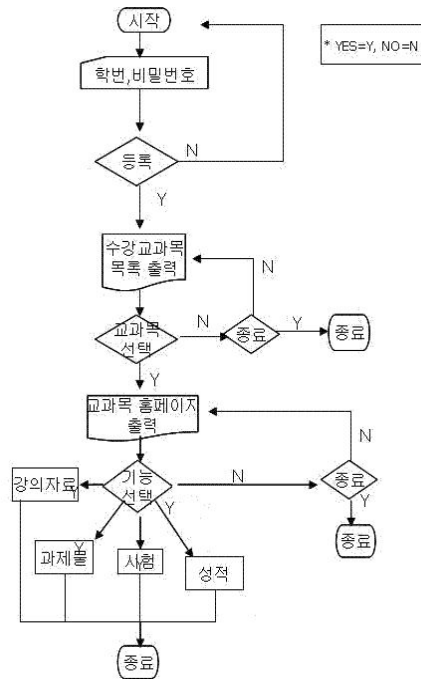
### 3.1. 시스템 전반적인 개요

본 논문에서 구현하고자 하는 수업관리 System은 교실 수업의 보조 도구로서 교실에서 제공되는 수업내용의 한계를 극복하기 위해서 웹 상에서 강의에 대한 자료를 제공하며, 그 강의에 대한 점검을 할 수 있도록 과제물을 내주고, 그 과제물 제출은 온라인을 통해 할 수 있도록 한다. 또한, 쪽지시험의 개념으로 강의에 대한 이해도를 측정할 수 있는 시험을 볼 수 있는 기능을 가지고 있으며, 수업에 대한 전반적인 관리를 각각의 사용자 즉, 학생, 교수, 관리자 입장에서 수업에 적극적으로 참여할 수 있도록 하는 System이라고 말할 수 있다.

이 System의 각각의 사용자들에 대해 살펴보면 다음과 같다.

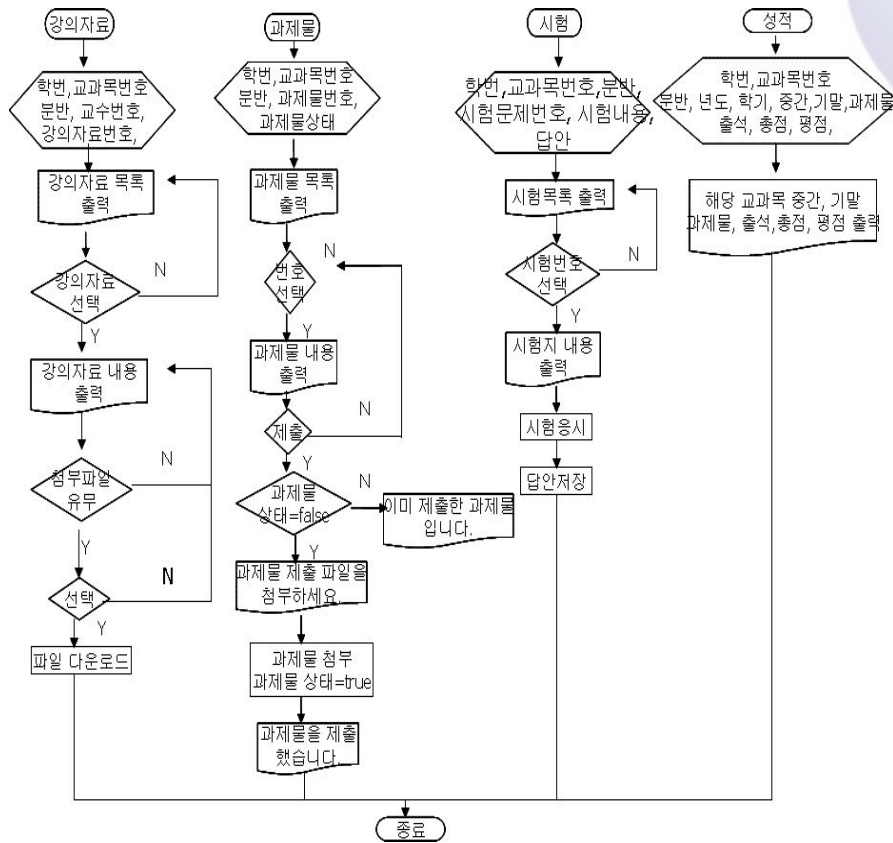
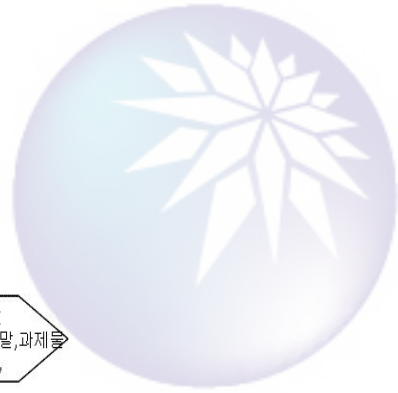
학생은 로그인을 통해 강좌를 등록하거나, 등록된 강좌에 대한 선택을 할 수 있고, 원하는 강좌에 대한 정보를 얻을 수 있다. 또한 과제물을 제출하고, 해당 강좌에 대한 시험을 볼 수 있으며, 성적 확인을 할 수 있다.

학생 흐름에 대해 살펴보면 다음 [그림 3-1]과 같다.



[그림 3-1] 학생 흐름

등록된 학생은 로그인을 통해 수강한 교과목에 대한 정보를 얻고, 원하는 과목을 선택하면, 해당 과목에 대한 강좌페이지로 이동해서 여러 정보를 메뉴를 통해 얻을 수 있다. 이들 메뉴의 기능에 대한 흐름은 [그림 3-2]와 같다.



[그림 3-2] 학생 메뉴들의 흐름

[그림 3-2]에 나와 있는 각 메뉴에 대한 흐름을 설명하면 다음과 같다.

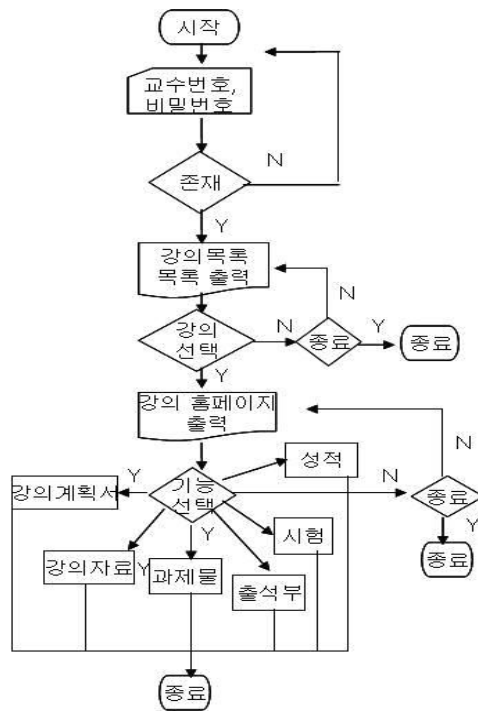
- ① 강의자료 : 해당 강좌에 대한 강의자료를 제공하는 것으로 선택 시 강의자료 목록이 나오고, 그 목록 중 원하는 자료를 하나 선택하면 좀더 자세히 강의자료 내용을 볼 수 있다. 강의자료가 파일로 첨부되어 있는 경우에는, 해당 파일을 다운로드 받아 내용을 볼 수 있다.



- ② 과제물 : 해당 강좌에 대해 교수가 내준 과제물 내용을 제공하는 메뉴로서, 선택 시 과제물 목록을 보여주며, 자세한 내용은 원하는 과제물을 선택하여 확인할 수 있다. 또한, 과제물에 대한 내용이 파일로 첨부되어 있는 경우, 그 파일을 다운로드 하여 내용을 확인할 수 있다.
  
- ③ 시험 : 교수가 강좌에 대한 시험을 출제했을 경우, 시험목록이 보여지고, 학생이 그 목록 중 하나를 선택하면, 해당 시험에 응시할 수 있게 된다. 시험을 본 후, 제출 버튼을 누르면 시험에 대한 답안이 저장된다.
  
- ④ 성적 : 학생이 응시한 시험에 대한 성적을 확인할 수 있는 기능을 제공한다.

교수는 강좌가 등록되어 있는 교수에 한하여 강좌에 대한 강의계획서, 강의자료 등 강좌에 대한 정보를 올릴 수 있고, 과제물을 학생들에게 내줄 수 있다. 또한, 과제물에 대한 채점을 할 수 있으며, 해당 강좌에 대한 시험지 작성을 할 수 있다.

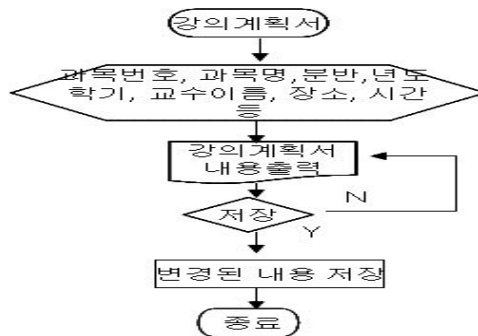
교수에 대한 흐름을 살펴보면 다음 [그림 3-3]과 같다.



[그림 3-3] 교수 흐름

강좌가 등록된 교수는 로그인을 통해 교과목들에 대한 정보를 얻을 수 있고, 그 중 원하는 과목을 선택하면, 해당 과목의 강좌페이지로 이동해 강좌에 대한 여러 정보를 메뉴를 통해 얻을 수 있다.

여러 메뉴 중 강의계획서 메뉴에 대한 흐름은 [그림 3-4]와 같다.



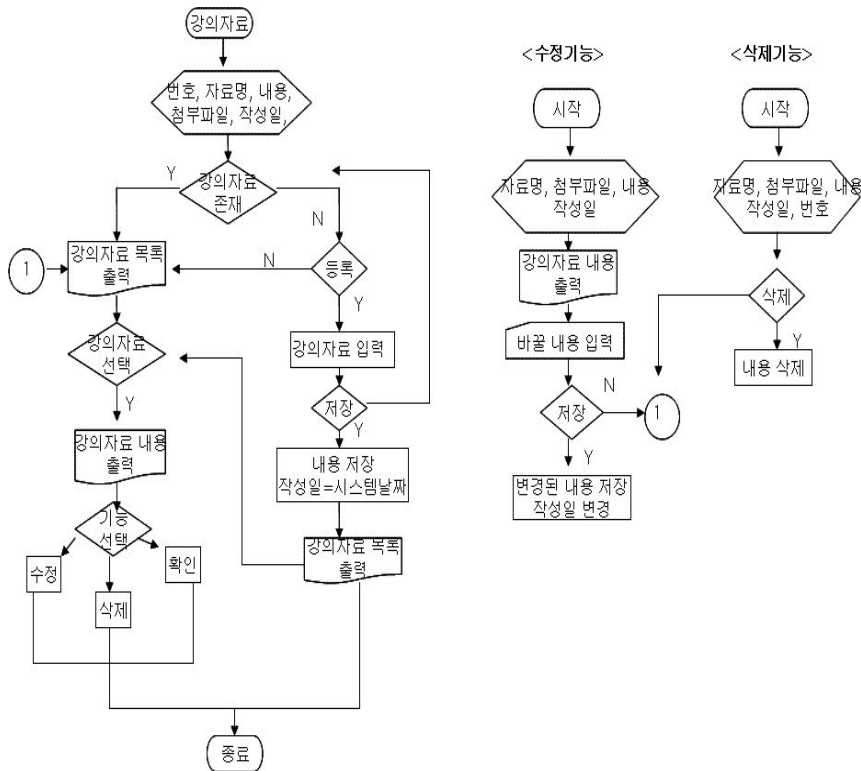
[그림 3-4] 강의계획서메뉴 흐름



강의계획서메뉴 흐름에 대한 설명은 다음과 같다.

- ① 강의계획서 : 해당 강좌에 대한 강의계획서를 선택하면 강의계획서의 기본적인 내용이 출력된다. 교수가 자세한 강의계획서를 원할 경우, 내용을 입력하여 저장할 수 있고, 입력한 내용의 변경을 원할 경우 변경할 수 있는 기능을 제공한다.

[그림 3-5]는 강의자료 메뉴에 대한 흐름은 나타낸 것이다.



[그림 3-5] 강의자료메뉴 흐름

강의자료메뉴 흐름에 대한 설명은 다음과 같다.

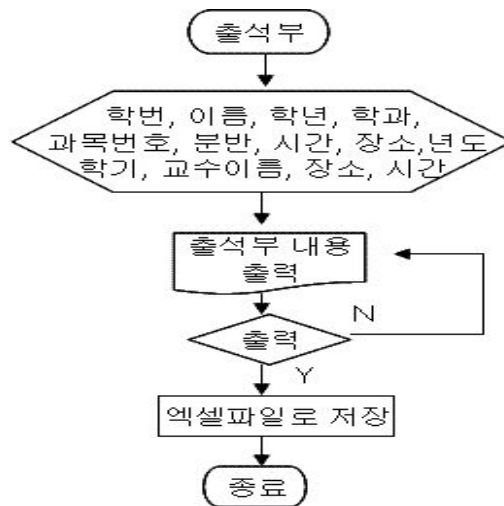




과제물메뉴 흐름에 대한 설명은 다음과 같다.

- ③ 과제물 : 해당 강좌에 대한 과제물을 학생들에게 부과하기를 원할 경우 과제물을 등록할 수 있다. 등록된 과제물은 언제든지 수정과 삭제가 가능하며, 학생들이 제출한 과제물 결과를 확인한 후 이를 채점할 수 있다.

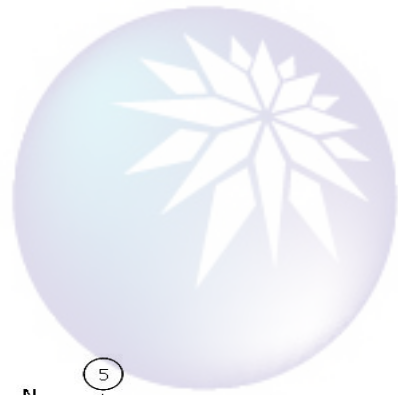
[그림 3-7]은 출석부 메뉴의 흐름에 대한 것이다.



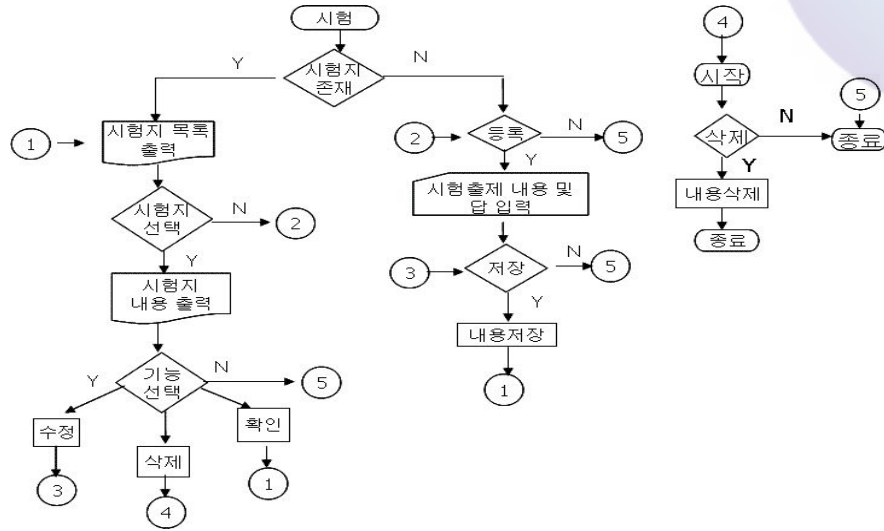
[그림 3-7] 출석부메뉴 흐름

출석부메뉴 흐름에 대한 설명은 다음과 같다.

- ④ 출석부 : 해당 강좌를 수강한 학생들의 출석부를 확인하고, 엑셀 파일로 저장할 수 있는 기능을 제공한다.



[그림 3-8]은 시험 메뉴의 흐름에 대한 것이다.

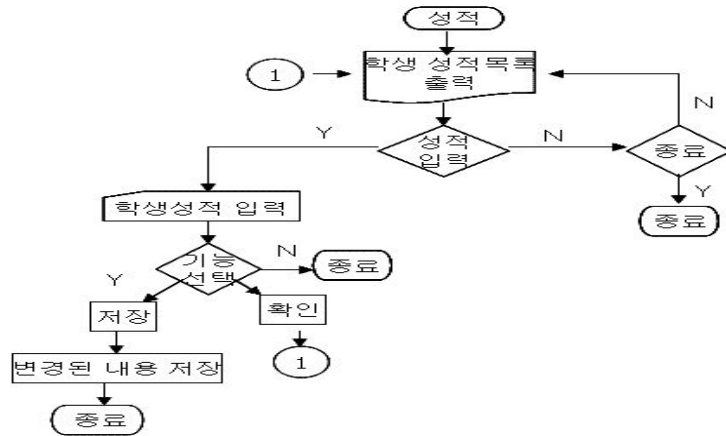


[그림 3-8] 시험메뉴 흐름

시험메뉴 흐름에 대한 설명은 다음과 같다.

- ④ 시험 : 해당 강좌에 대한 시험지를 등록할 수 있으며, 등록된 시험지에 대해서 해당 답안을 작성할 수 있다. 또한, 등록된 시험지의 내용은 수정과 삭제가 가능하다. 물론, 답안도 수정과 삭제가 가능하다.

[그림 3-9]는 성적메뉴의 흐름에 대한 것이다.



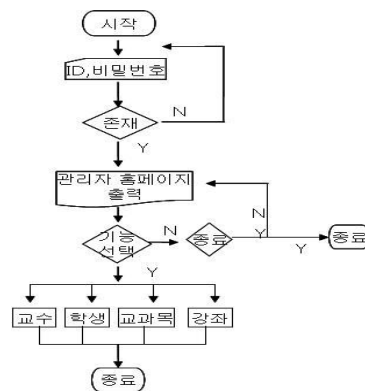
[그림 3-9] 성적메뉴 흐름

성적메뉴 흐름에 대한 설명은 다음과 같다.

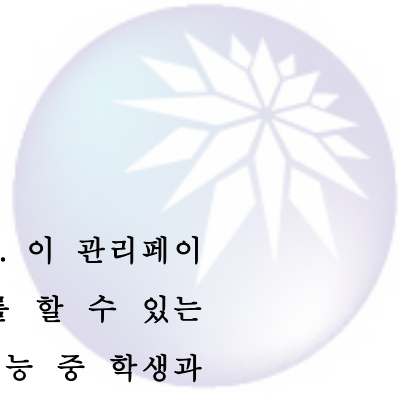
- ⑥ 성적 : 해당 강좌를 수강하는 학생들에 대한 성적을 등록할 수 있으며, 등록된 성적은 수정과 삭제가 가능하다.

관리자는 학생, 교수, 교과목, 강좌에 대한 일을 하며 시스템이 제대로 운영될 수 있도록 지원하고, 또한 자료에 대한 백업 관리를 한다.

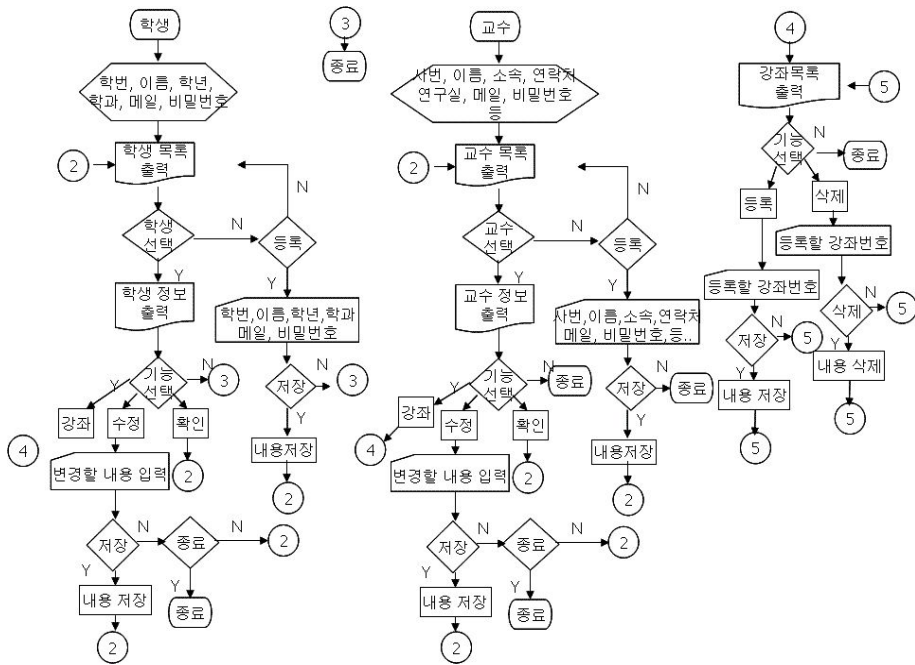
좀더 자세히 관리자에 대해 살펴보면 다음 [그림 3-3]과 같다



[그림 3-10] 관리자 흐름



관리자는 로그인을 통해 관리페이지로 이동할 수 있다. 이 관리페이지에는 교수, 학생, 교과목, 강좌, 출석부에 대한 관리를 할 수 있는 메뉴가 있어서 여러 정보들을 관리한다. 이들 메뉴의 기능 중 학생과 교수에 대한 흐름은 [그림 3-11]과 같다.



[그림 3-11] 학생과 교수 흐름

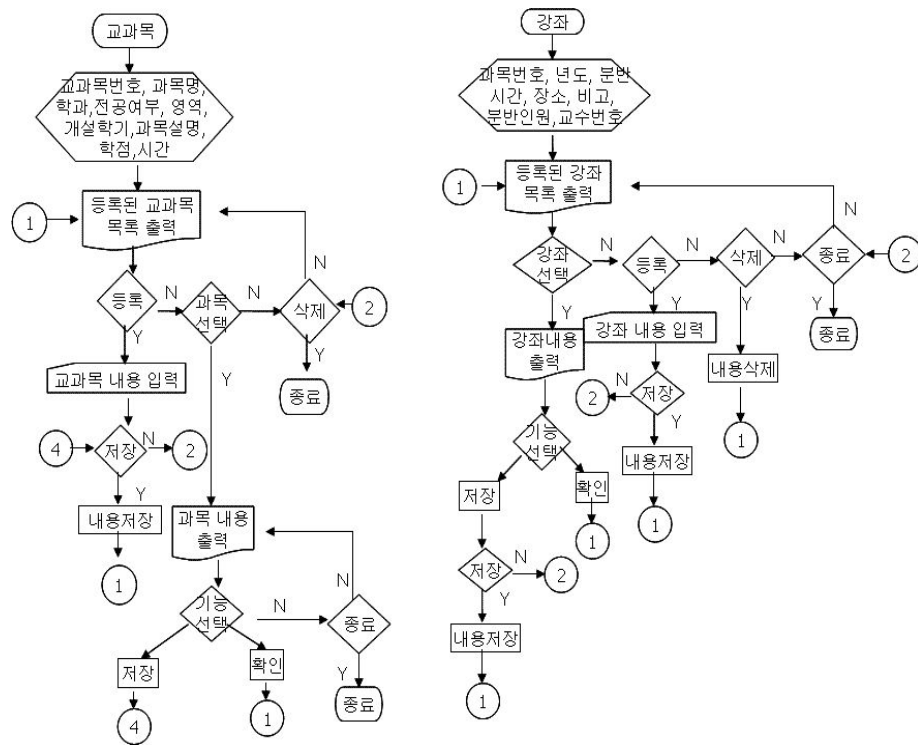
학생과 교수 메뉴의 흐름에 대해 설명하면 다음과 같다.

- ① 학생 : 학생을 등록하거나, 등록된 정보를 수정할 수 있는 기능을 갖고 있다. 또한, 학생이 등록한 강좌 목록을 보고 해당 강좌를 삭제하거나, 새로운 강좌 등록이 가능하다.



- ② 교수 : 교수를 등록하거나, 등록된 정보를 수정할 수 있는 기능을 갖고 있다. 또한, 교수가 강의하는 강좌에 대한 목록을 제공해주고, 해당 강좌를 삭제할 수 있다.

[그림 3-12]는 교과목과 강좌메뉴에 대한 흐름을 나타낸 것이다.



[그림 3-12] 교과목과 강좌메뉴 흐름

교과목과 강좌메뉴의 흐름에 대해 설명하면 다음과 같다.

- ③ 교과목 : 교과목 정보를 등록하거나, 등록된 정보를 수정, 삭제할 수 있는 기능을 갖고 있다.



- ④ 강좌 : 강좌 정보를 등록하거나, 등록된 정보를 수정, 삭제할 수 있는 기능을 갖고 있다.

## 3.2. 개발 환경

분석 내용에서 다른 사용자들에 대한 정보를 추가하거나 변경하는 등과 같은 작업을 좀더 쉽게, 그리고, 어떤 시스템에서도 정보를 주고 받을 수 있도록 하기 위해서 본 논문에서는 XML을 사용하였다.

본 논문에서 사용한 XML은 여러가지 장점을 가지고 있다. 그 장점을 정리하면 다음과 같다.

첫째, XML은 관련업계의 표준이다. 데이터의 공유와 그에 따른 전송으로 인해 출현하게 되었다.

둘째, XML은 벤더(Vendor)와 벤더간에 독립적이다. 그래서 어떠한 벤더에게 종속되지 않고 순수한 파일로 데이터를 저장하고 보관한다.

셋째, 자료의 영속성이다. 두 번째에 거론했던 것과 비슷한 맥락이지만, 사용자가 어떤 벤더를 사용하든지 간에 상관없이 언제 어디서나 같은 자료를 볼 수 있다. XML은 자료를 가장 근본적인 파일단위로 저장하는 방식이기 때문이다.

넷째, 풍부한 인력과 많은 지원 소프트웨어를 들 수 있다. XML은 산업계 전반에 걸쳐서 사용이 될 것이고 그렇다면 엄청난 프로그래머와 이들이 쓸 많은 소프트웨어들이 등장할 것이다. 지금도 매년 해마다 엄청난 양의 소프트웨어들이 쏟아지고 있는 것만 보아도 알 수 있다 [15]. 이렇게 여러 장점을 갖고 있는 XML을 본 연구에 사용하였다.



XML이 많이 사용되면서 XML전용 데이터베이스의 필요성 또한 대두되었다. XML전용 데이터베이스 중 하나로 eXcelon이 있는데, 그 특징은 다음과 같다.

첫째, XML문서를 DOM형태로 저장한다. 그래서, 필요할 때마다 XML을 Parsing할 필요 없이 빠르게 데이터를 얻을 수 있고, XML 전체뿐만 아니라 일부분만의 정보를 가져 올 수 있다. 따라서, XML 정보는 다수의 사용자에게 빠르게 서비스가 되고, 대용량의 XML 데이터의 서비스가 가능하다.

둘째, Well-formed XML 문서를 저장한다. 하나의 응용프로그램이 다양한 구조의 XML문서를 다룰 수 있고, 저장된 XML문서에 Element 또는 Attribute 단위로 수정이 가능하다. 따라서 응용프로그램의 요구에 따라 XML Schema의 수정이 자유롭다.

셋째, 산업 표준을 지원한다. W3C XML1.0 with namespaces, W3C DOM Level 1.2, Xpath 1.0 query language, XSLT 1.0 with Java extension functions, 세계 언어를 위한 Standard Document Encodings를 지원한다.

넷째, 빠른 XSLT 기능을 제공한다. XSL도 XML 문서임으로 DOM구조로 저장된다. 즉, XSLT수행 중 Parsing 없이 바로 Transformation 수행한다.

다섯째, 사용하기 편리하다. XML 응용프로그램 개발 전과정에 사용되는 개발 툴(Stylus Studio)을 제공한다. 또한 관리자 및 개발자를 위한 관리도구 툴(DXE Manager)을 지원한다.

여섯째, 기관 시스템과의 통합이 용이하다. J2EE서버와 통합이 쉽다(JCA제공). 그리고, RDB와 데이터 교환이 용이하다(XSLT 사용)[17].

본 연구에서는 위와 같이 여러 장점을 갖고 있는 eXcelon을 또한



이용하였다.

본 연구를 위해 사용된 개발 환경을 정리하면 [표 3-1]과 같다.

하드웨어	
CPU	Intel Pentium 4 Mobile CPU 1.60GHZ
Memory	256 Mbytes
소프트웨어	
운영체제	Microsoft Windows XP Professional
데이터베이스	eXcelon 3.1
프로그래밍언어	JSP, 자바스크립트
웹 브라우저	Microsoft Internet Explorer 6.0

[표 3-1] 개발 환경



## 제 4 장 수업관리 System 설계

### 4.1. 수업관리 System의 전체 구조

[그림 4-1]에서 보듯이 XML기반 온라인 수업관리 System처럼 부하가 적은 중소규모의 시스템 환경에 적합한 응용프로그램 개발 형태는 Web Server와 Java Servlet/JSP Engine을 이용한다. eXcelon의 XIS(eXtensible Information Server)는 번들로 Tomcat Servlet/JSP Engine을 제공한다. 또한 대부분의 Dynamic XML Engine(DXE) Java Client기능을 수행하는 Wev Servlet Framework(WSF)를 제공한다.

WSF는 Java Servlet으로 다음과 같은 기능을 제공한다[17].

첫째, XML 문서 전체 추출.

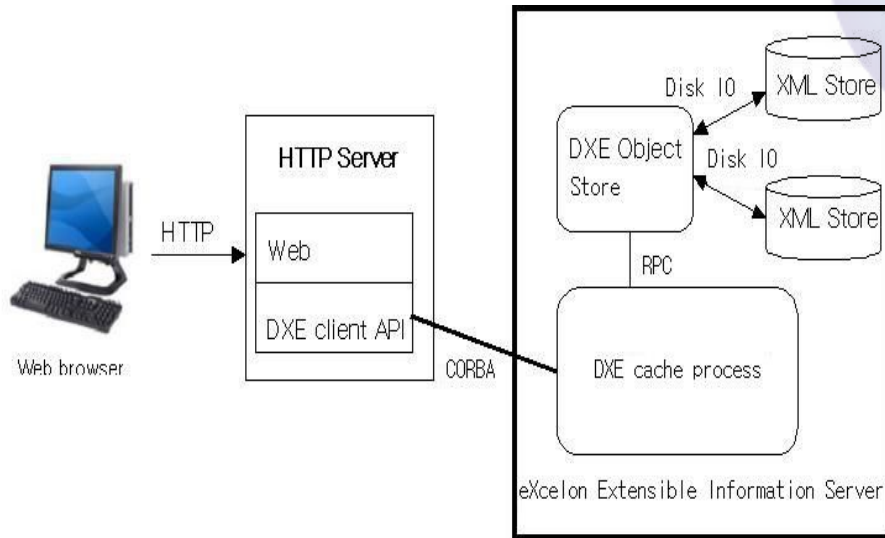
둘째, XML Query.

셋째, XML Update.

넷째, XSLT Transformation

다섯째, XIS Server에 등록된 모듈 호출 및 결과 처리

여섯째, Directory 문서 생성.



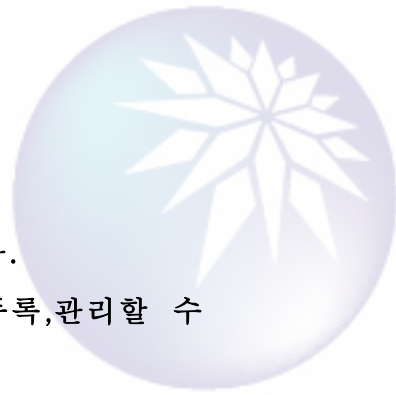
[그림 4-1] XML기반 수업관리 System 전체 흐름

## 4.2. 사용자 설계

본 논문은 수강신청이 이미 이루어졌다는 가정하에 수업에 대한 시스템을 설계하고, 구현하였다. 그래서, 수강신청에 대한 내용은 다루지 않고 있다. 본 논문에서의 사용자는 학생, 교수, 관리자이다. 사용자에 대해 살펴보면 다음과 같다.

학생은 수강 등록된 과목목록 중 자세한 정보를 원하는 과목을 선택해서 원하는 과목의 강좌에 대한 정보를 얻을 수 있으며, 해당 강좌에 대한 과제물을 제출하고, 또한 시험을 볼 수 있다.

교수는 여러 과목에 등록된 강좌에 대한 강의계획서, 강의자료, 과제물 및 시험지를 작성할 수 있다. 또한, 과제물에 대해 학생이 제출한



과제를 검사하고, 채점할 수 있고, 성적을 입력할 수 있다.

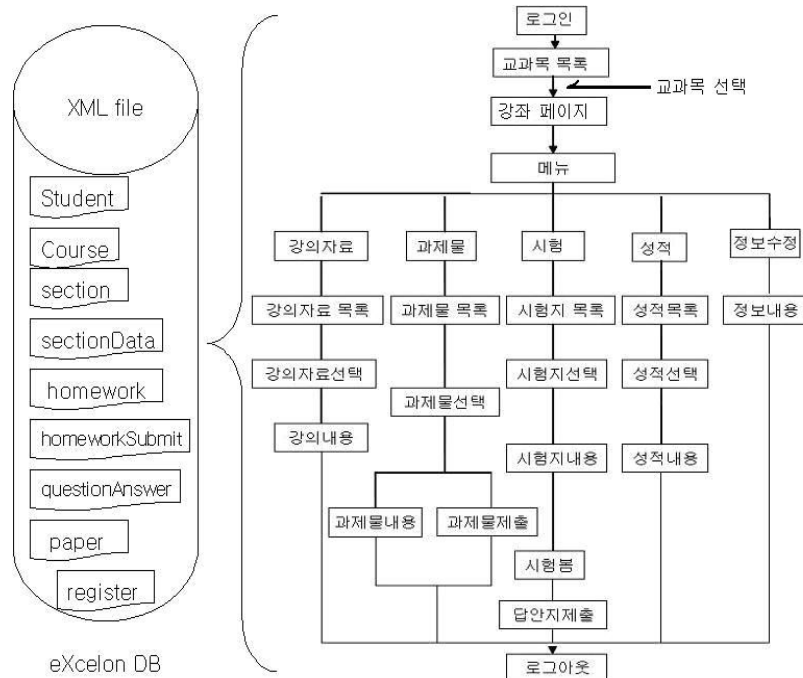
관리자는 학생, 교수, 교과목, 강좌에 대한 정보를 등록,관리할 수 있다.

좀더 자세히 사용자 설계에 대해 살펴보자.

#### 4.2.1. 학생

수강 신청한 학생에 한하여 로그인을 할 수 있다. 로그인 이후에는 학생이 수강한 교과목 목록이 나오고, 그 중 자세한 내용을 원하는 과목을 선택하면 해당 교과목의 강좌페이지로 이동한다.

강좌페이지에서는 강좌에 대한 정보를 5개의 메뉴를 통해 제공한다. 그 내용은 다음 [그림 4-2]와 같다.





[그림 4-2] 학생 메뉴  
메뉴에 대해 자세히 설명하면 다음과 같다.

① 강의자료

해당 강좌에 대한 강의 자료를 저장하고 있는 sectionData.xml 문서에서 정보를 얻어 제공하며, 첨부된 파일이 있으면 다운로드할 수 있다.

② 과제물

해당 강좌에 대한 과제물을 저장하고 있는 homework.xml 문서에서 정보를 얻어 제공하며, 첨부된 파일이 있으면 다운로드할 수 있다. 학생은 과제에 대한 과제물을 제출할 수 있다. 이때, 파일로 과제물을 제출해야 하는 제한조건이 있다.

③ 시험

해당 강좌에 대한 시험지 목록을 보여주고, 선택한 시험지에 대한 시험을 볼 수 있다.

④ 성적

③에서 치룬 시험에 대한 성적뿐만 아니라, 중간고사, 기말고사, 출석점수에 대한 정보를 제공한다.

⑤ 정보수정

학생 스스로 자신의 변경된 정보를 직접 수정할 수 있다.

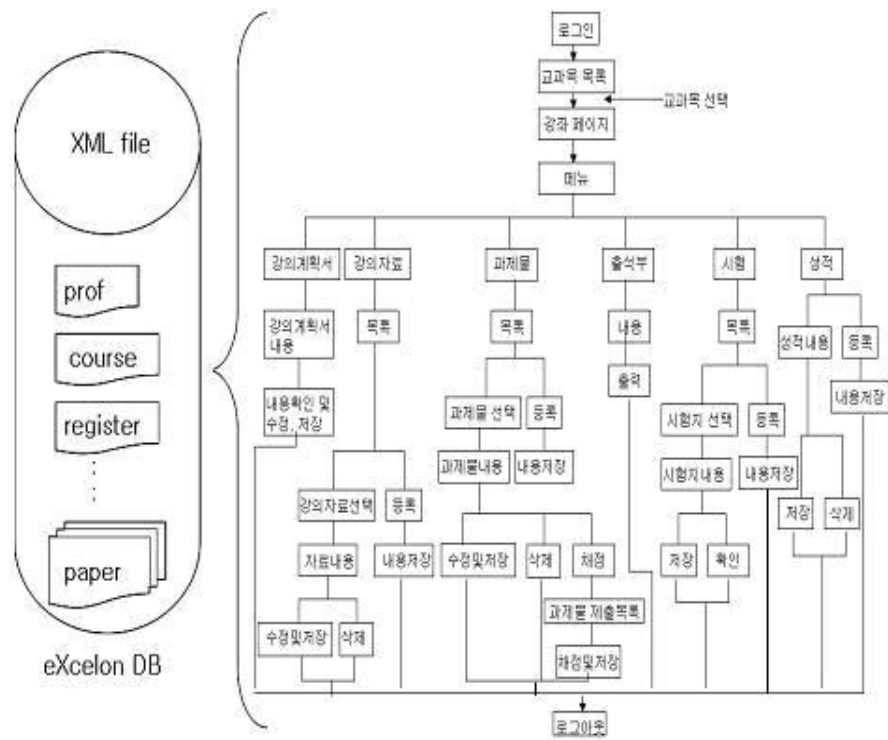


## 4.2.2. 교수

등록된 강좌가 있는 교수에 한하여 로그인을 할 수 있다. 로그인 이후에는 교수의 등록된 교과목 목록이 나오고, 그 중 자세한 내용을 원하는 과목을 선택하면 해당 교과목의 강좌페이지로 이동한다.

강좌페이지에서는 강좌에 대한 정보를 6개의 메뉴를 통해 제공한다.

그 내용은 다음 [그림 4-3]와 같다.



[그림 4-3] 교수 메뉴



메뉴에 대해 자세히 설명하면 다음과 같다.

① 강의계획서

강좌에 대한 정보를 갖고 있는 section.xml문서에서 해당 강좌에 대한 강의계획서 정보만을 추출해 그 내용을 보여주는 메뉴로서, 내용이 존재하지 않을 경우, 내용을 입력할 수 있고, 내용이 존재하는 경우에는 그 내용을 수정할 수 있다.

② 강의자료

강좌에 대한 강의자료를 등록하거나 변경 혹은 삭제 할 수 있는 메뉴로서, 학생들에게 강의에 대한 보다 폭넓은 자료를 제공할 수 있다. 이때, 자료를 일일이 입력할 수도 있고, 또는 파일을 업로드 해서 정보를 제공할 수 있다.

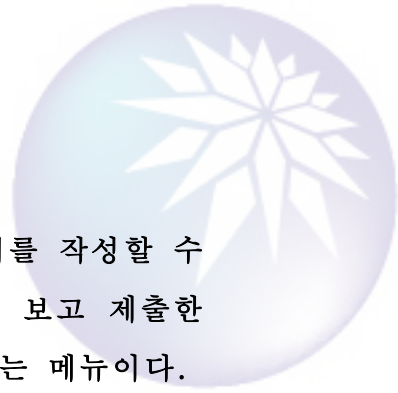
③ 과제물

강좌에 대한 과제물을 학생들에게 내줄 수 있도록 과제물을 등록, 수정, 삭제할 수 있는 기능을 부수적으로 갖고 있고, 학생들이 해당 과제물에 대해 제출한 과제물 내용을 확인하고, 채점할 수 있는 기능을 제공하는 메뉴로서, 과제물 작성은 파일형태로도 가능하다.

④ 출석부

강좌에 대해 수강 등록한 학생의 출석부를 교수의 컴퓨터에 엑셀 파일 형태로 저장할 수 있는 기능을 제공하는 메뉴이다.

⑤ 시험



강좌에 대한 강의 내용을 시험 볼 수 있도록, 시험지를 작성할 수 있는 기능을 제공하며, 또한, 학생들이 해당 시험을 보고 제출한 답안지를 검사하고, 채점할 수 있는 기능까지 제공하는 메뉴이다.

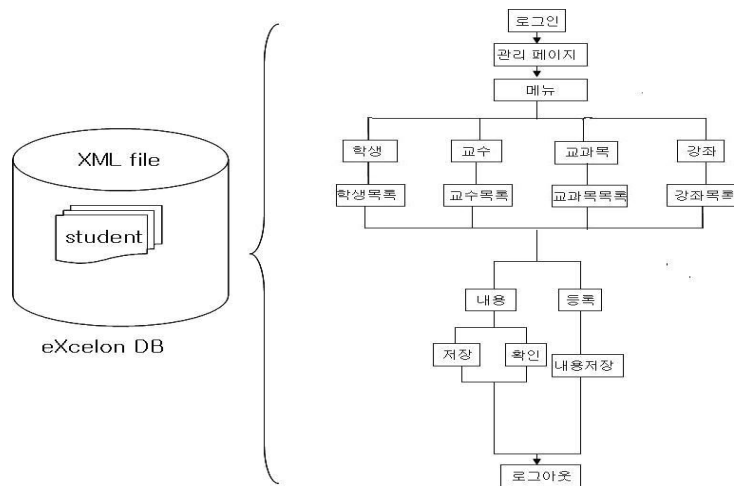
⑥ 성적

강좌를 수강하는 학생들에 대한 중간고사, 기말고사, 과제물 점수 등을 입력하거나, 입력된 성적을 수정할 수 있는 기능까지 제공하는 메뉴이다.

### 4.2.2. 관리자

관리자가 로그인을 하면 관리페이지로 이동하게 되고, 관리페이지에서는 4개의 메뉴를 통해 시스템의 관리를 돕는다.

그 내용은 다음 [그림 4-4]와 같다.



[그림 4-4] 관리자 메뉴



메뉴에 대해 자세히 설명하면 다음과 같다.

① 학생

학생정보를 등록하거나, 변경 및 저장할 수 있는 기능을 제공하는 메뉴이다.

② 교수

교수정보를 등록하거나, 변경 및 저장할 수 있는 기능을 제공하는 메뉴이다.

③ 교과목

교과목정보를 등록하거나, 변경 및 저장할 수 있는 기능을 제공하는 메뉴이다.

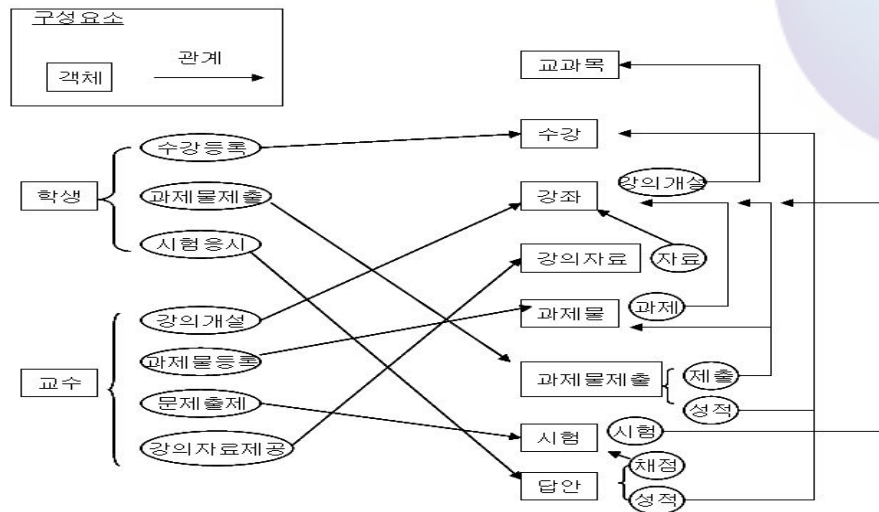
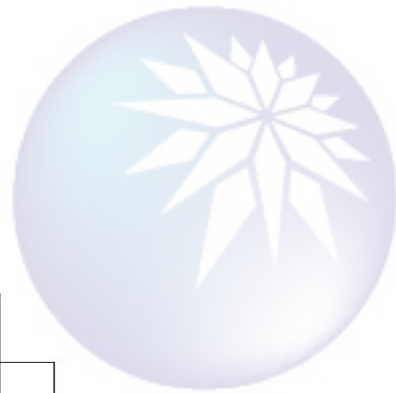
④ 강좌

강좌정보를 등록하거나, 변경 및 저장할 수 있는 기능을 제공하는 메뉴이다.

### 4.3. XML Schema 설계

스키마 설계에 앞서 전체 데이터 구조에 대해 알아보자.

그 내용은 [그림 4-5]와 같다.



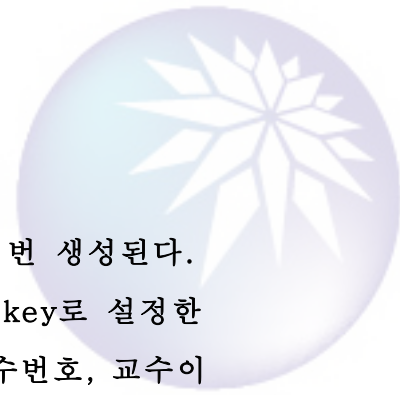
[그림 4-5] 전체 데이터구조

[그림 4-5]의 전체 데이터구조를 바탕으로 스키마를 설계하였다. 그 내용은 다음과 같다.

### 4.3.1. 학생

학생정보가 추가 될 때마다 <student> element 가 매번 생성된다. 학생정보에서 값이 유일해야 하고, 학생들을 구분할 수 있는 학번을 key로 설정한다. 이는 기존의 데이터베이스에서의 primary key와 같다. <student> element 가 갖는 하위 element에는 학번, 학년, 학과 이름, 이메일, 비밀번호가 있다. 자세한 설명은 부록에서 [student.xsd]에 나타나있다.

### 4.3.2. 교수



교수정보가 추가 될 때마다 <professor> element가 매번 생성된다. 교수정보에서 값이 유일해야 하는 교수번호<prof\_no>를 key로 설정한다. <professor> element 가 갖는 하위 element에는 교수번호, 교수이름, 소속학과, 핸드폰, 이메일, 연구실전화, 연구실방번호, 비밀번호가 있다.자세한 설명은 부록에서 [prof.xsd]에 나타나 있다.

### 4.3.3. 교과목

교과목 정보를 담당하는 스키마에서는 새로운 타입을 만들어 사용하고 있다. 학과에 대한 type을 ‘컴퓨터정보학부’와 ‘미디어정보학부’로 한정하여 새로운 type인 DEPT\_kind를 만들었고, 또한 학기에 대한 type을 ‘1학기’, ‘2학기’로 한정하여 새로운 type인 SEMESTER\_kind를 만들었다. 교과목 정보가 추가 될 때마다 <course> element가 생성되고, <course>element가 갖는 하위element에는 교과목번호, 교과목이름, 학과, 전공여부, 영역, 개설학기, 과목설명, 학점, 시간이 있다. 자세한 설명은 부록에서 [course.xsd]에 나타나 있다.

### 4.3.4. 수강

학생이 강좌를 수강한 정보를 담당하는 스키마로서, [student.xsd]와 [section.xsd]를 include하고 있다. <registerinfo>element가 갖는 하위 Element는 <register>이다. <register>element가 갖는 하위 element <sectionID>와 <student\_no>는 다른 schema에서 key로 지정한 것을 keyref로 참조하고 있다. <register>element가 갖는 하위 element에는 강좌번호, 학번, 중간점수, 기말점수, 출석점수, 과제물점



수, 기타 시험점수, 총점, 학점이다. 자세한 설명은 부록에서 [register.xsd]에 나타나 있다.

#### 4.3.5. 강좌

교과목에 대해 개설된 강좌 정보를 담당하는 스키마로 강좌에 대한 강의계획서 내용도 포함하고 있다. 강좌가 하나씩 개설될 때 마다 <section>element가 하나씩 증가한다. <section>element가 갖는 하위 element에는 강좌번호, 교과목번호, 교수번호, 년도, 분반, 분반인원, 시간, 장소, 비고, 강의계획서이다. 자세한 설명은 부록에서 <section.xsd>에 나타나 있다.

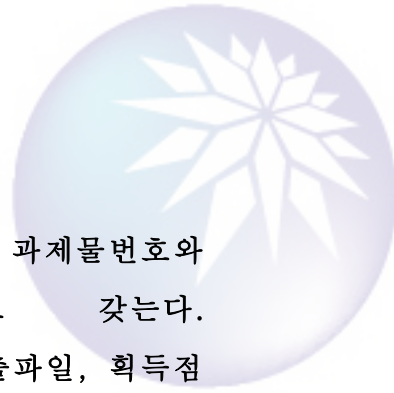
#### 4.3.6. 과제물

과제물에 대한 정보를 담당하는 스키마로 강좌에 대한 과제물이 증가할 때마다 <homework>element가 하나씩 증가된다.

<homework>element가 갖는 하위 element에는 과제물번호, 과제물명, 첨부파일, 내용, 최고점수, 제출기한이 있다. <homework>의 상위 element <homeworkinfo>는 attribute로 강좌번호를 갖는다. 자세한 설명은 부록에서 <homework.xsd>에 나타나 있다.

#### 4.3.7. 과제물 제출

학생이 수강한 강좌에 대한 과제물제출 정보를 담당하는 스키마로 <homeworkSubmitInfo>element                    하위                    element로



<HomeworkSubmit>이 있고, 이 element는 과제물번호와 <HomeworkSubmitContent>를 하위 element로 갖는다. <HomeworkSubmitContent>는 학번, 과제물제출일, 제출파일, 획득점수를 하위 element로 갖는다. <homeworkSubmitInfo>element는 attribute로 강좌번호를 갖는다. 자세한 설명은 부록에서 <homeworkSubmit.xsd>에 나타나 있다.

#### 4.3.8. 강의자료

교수가 강좌에 대해 제공하는 강의자료 정보를 담당하는 스키마로 <lectureNoteInfo>라는 최상위 element는 attribute로 강좌번호를 element로는 <lectureNoteContent>를 갖는다. <lectureNoteContent>는 하위 element로 자료번호, 자료이름, 자료내용, 첨부파일, 작성일을 갖는다. 자세한 설명은 부록에서 <lectureNote.xsd>에 나타나 있다.

#### 4.3.9. 시험지

교수가 강좌를 수강하는 학생들에 대해 강좌에 대한 이해도를 측정하고, 평가하기 위해 시험을 볼 경우, 시험지에 대한 정보를 담당하는 스키마로 최상위 element인 <paperInfo>는 attribute로 강좌번호를 하위 element로는 <paper>를 갖는다. <paper>는 하위 element로 시험지구분번호, 시험명, <paperContents>를 갖는다. <paperContents>는 문제번호, 문제내용, 배점, 정답, <instance>를 하위 element로 갖으며, <instance>는 보기번호, 보기내용을 하위 element로 갖는다. 자세한 설명은 부록에서 <paper.xsd>에 나타나 있다.



#### 4.3.10. 답안지

교수가 출제한 시험을 학생이 볼 경우, 학생이 작성한 답안지에 대한 정보를 담당하는 스키마로 <section.xsd>, <student.xsd>, <paper.xsd>를 include하고 있다. 최상위 element인 <QuestionAnswerInfo>는 attribute로 강좌번호를 갖으며, 하위 element로는 <question\_answer>를 갖는다. <question\_answer>는 attribute로 시험지구분번호를 갖고, element로는 <QA\_contents>를 갖는다. <QA\_contents>는 attribute로 학번을 갖으며, 하위 element로는 <contents>를 갖는다. <contents>는 문제번호, 작성한답, 취득점수를 하위 element로 갖는다. 자세한 설명은 부록에서 <QuestionAnswer.xsd>에 나타나 있다.



## 제 5 장 수업관리 System구현

### 5.1. 학생

학생이 수업관리 System을 이용하여 강의자료, 과제물, 시험, 성적의 메뉴를 선택할 때 나타나는 화면들이다.

[그림 5-1]은 학생이 로그인을 해서 수업관리 시스템으로 들어간 뒤 처음으로 나타나는 화면으로 학생이 수강한 교과목 목록을 볼 수 있다.

당신이 수강한 교과목 목록입니다.

교과번호	교과이름
NAD0100	자료구조

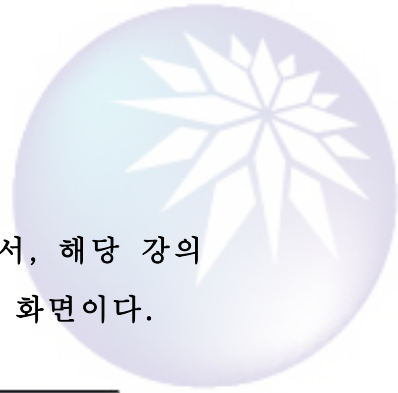
[그림 5-1] 교과목 목록을 보여주는 화면

[그림 5-2]는 교과목을 선택한 후 제공되는 강좌 홈페이지에서 강의자료 메뉴를 선택했을 경우 보여주는 강의자료 목록이다.

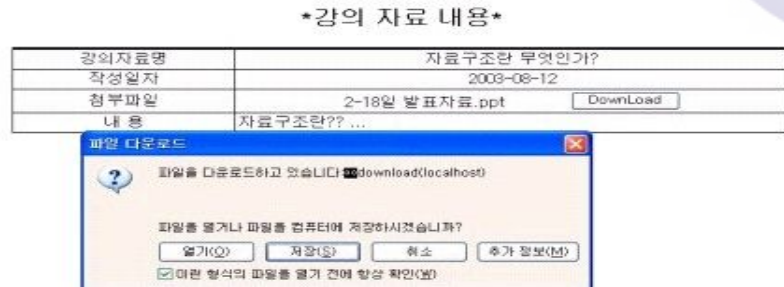
\*강의 자료\*

순서	강의자료명	작성일자
1	자료구조란 무엇인가?	2003-08-12
2	스택, 큐란 무엇인가?	2003-08-14

[그림 5-2] 강의자료 목록을 보여주는 화면



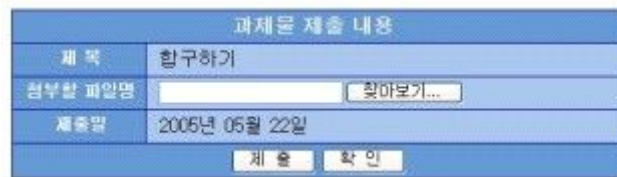
[그림 5-3]은 강의자료 중 원하는 강의자료를 선택해서, 해당 강의 자료 내용을 자세히 본 후, 첨부된 파일을 다운로드 하는 화면이다.



[그림 5-3] 첨부된 강의자료 파일 다운로드 하는 화면

과제물 메뉴도 강의자료와 마찬가지로 과제물 목록 중 하나의 과제물을 선택하면 과제물 내용을 볼 수 있고, 첨부된 파일이 있으면 다운로드 받을 수 있다.

[그림 5-4]는 과제물 내용을 본 후, 제출버튼을 선택해 과제물을 제출하는 화면이다. 과제물 제출은 파일을 첨부해서 한다.



[그림 5-4] 과제물제출 화면

[그림 5-5]는 시험 메뉴를 선택해서 시험을 보는 화면이다. 시험지는 한 번만 볼 수 있다. 문제를 풀지 않고 화면을 닫거나, 종료를 하면 시험을 본 것으로 간주해서 다시는 시험지를 볼 수 없다. 이러한 경고



메시지는 시험지 내용을 보여주기 전에 이루어진다.



### 시험지 내용

번호	내용	보기	배정/정답
1	배열 선언이 올바르지 않는 것을 고르시오. (자바)	<pre> 1)int a; 2)int[] a; 3)int s[]; 4)int [] s;</pre>	1 <input type="checkbox"/>
2	다음 설명이 무엇을 말하는지 고르시오. [설명] 선형리스트의 양쪽 끝에서 입출력을 모두 할 수 있는 구조.	<pre> 1)스택 2)큐 3)데크 4)리스트</pre>	2 <input type="checkbox"/>
3	다음은 스택에로의 삽입을 나타낸 코드이다. 빈칸(?)에 들어갈 알맞은 것을 고르시오. [코드] void add(int *top, element item) { if(*top >= MAX_STACK_SIZE-1) { stack_full(); return; } stack[(?)] = item; }	<pre> 1)*top++ 2)++*top 3)*top++ 4)++(*top)</pre>	2 <input type="checkbox"/>

제출

[그림 5-5] 시험 보는 화면

[그림 5-6]은 성적을 확인하는 화면이다.

### 성적

중간고사	기말고사	과제점	출석	기타시험	총점	학점
0	0	0	0	0	0	0

[그림 5-6] 성적확인 화면



## 5.2. 교수

교수도 학생과 같이 로그인을 하면 강의하는 강좌에 대한 목록이 나오고, 그 중 하나를 선택하면 해당 홈페이지로 이동한다. 다음은 교수가 선택한 강좌 홈페이지에서 강의계획서, 강의자료, 과제물, 시험, 성적, 출석부 메뉴를 사용할 때 나타나는 화면들이다.

[그림 5-7]은 강의계획서 메뉴를 선택해서 강의계획서 내용을 보는 화면이다. 강의계획서 내용은 수정이 가능하다.

**수업 계획서**

2006년 제 1학기 교과구분 98년 이전 입학자 :  
98년 이후 입학자 : 전공

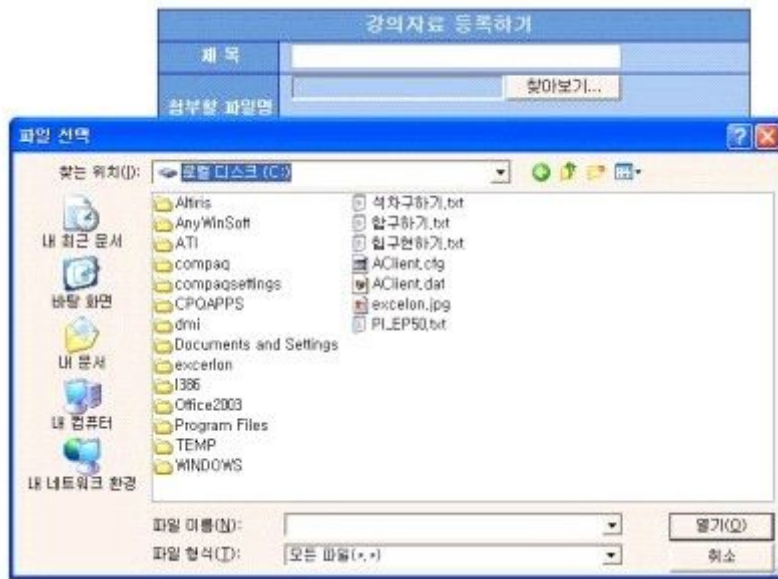
교과목명	자료구조	3학점 3시간	학수번호 NA00100	영역구분	3
담당교수	홍길동 연구실	수정권 A동 30	상담시간 화 13:00-15:00		
E-mail	hong12@sungshin.ac.kr			전화	02-920-1234
학습목표	본 강의는 소프트웨어를 프로그래밍할 때 미리 결정하여야 하는 자료구조에 대한 강의이다.				
주교재	이석호, "자료구조와 C", 경익사, 2004				
부교재 및 참고도서	이석호 역, "C로 쓴 자료구조론", 홍경당, 1999				
수업방법	이론중심강의 <input checked="" type="checkbox"/> 실험, 실기, 실습 <input checked="" type="checkbox"/> 토론 <input checked="" type="checkbox"/> 세미나 <input checked="" type="checkbox"/> 과제발표 <input type="checkbox"/> 현장답사 <input type="checkbox"/> 질의응답 <input type="checkbox"/> 개별지도 <input type="checkbox"/> 립지도 <input type="checkbox"/> 기타 <input type="checkbox"/>				
교육기자재	비디오 <input type="checkbox"/> 오디오 <input type="checkbox"/> O.H.P <input checked="" type="checkbox"/> Slide <input checked="" type="checkbox"/> Computer <input checked="" type="checkbox"/> 기타 <input type="checkbox"/>				
평가방법	중간고사 : 40% 기말고사 : 40% 과제물 : 10% 출석 : 20% 기타 : 0%				
<b>주 별 수업 내 용</b> <small>(교재별론 수업지형 방식, 과제물, 보고서, 세미나 주제 등을 기재)</small>					

[그림 5-7] 강의계획서 화면

[그림 5-8]은 강의자료메뉴를 선택해 내용을 등록하면서, 파일도 첨부

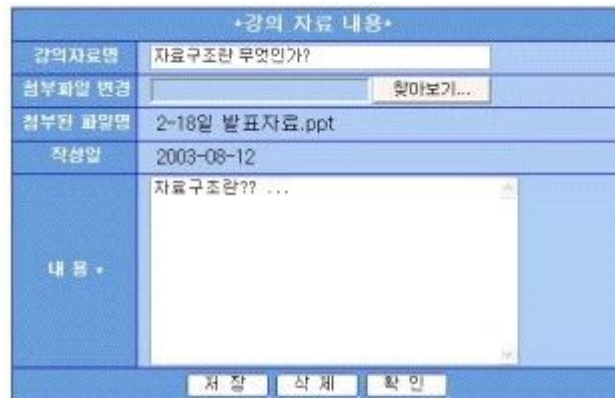


부하는 화면이다.

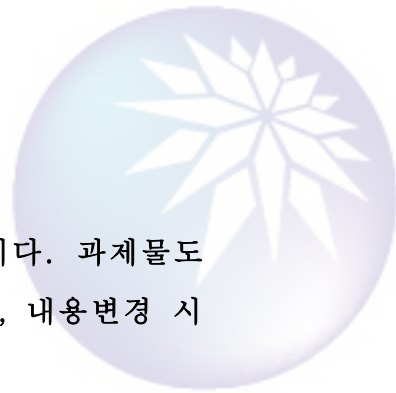


[그림 5-8] 강의자료등록 화면

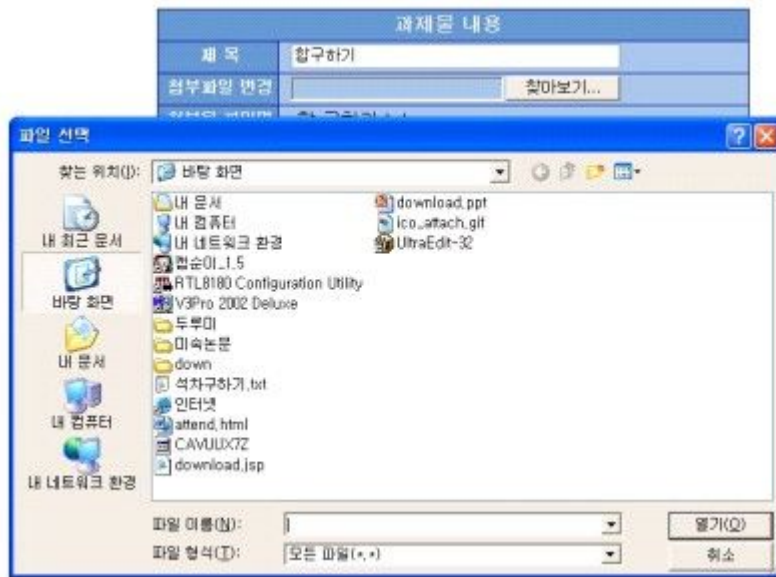
[그림 5-9]는 등록된 강의자료를 관리하는 화면으로, 강의자료 내용과 첨부된 파일을 변경할 수 있다. 또한, 삭제도 가능하다.



[그림 5-9] 강의자료 관리 화면



[그림 5-10]는 등록된 과제물내용을 변경하는 화면이다. 과제물도 강의자료와 마찬가지로 파일을 첨부해서 등록할 수 있고, 내용변경 시 첨부파일 변경도 가능하다.



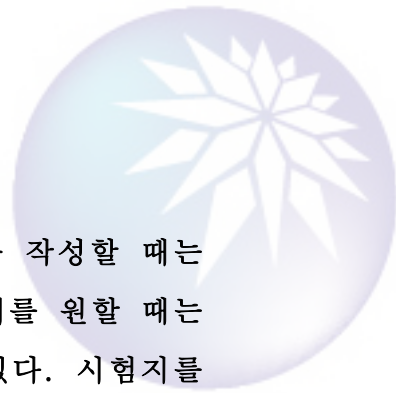
[그림 5-10] 과제물내용 변경 화면

[그림 5-11]은 해당 과제물을 제출한 학생에 대해 채점하는 화면이다. 이때, 학생이 과제물로 제출한 파일을 확인 할 수 있다.

#### 과제물 채점



[그림 5-11] 과제물채점 화면



[그림 5-12]는 시험지를 작성하는 화면이다. 시험지를 작성할 때는 문항 수를 하나씩 입력하는 공간을 추가할 수 있고, 삭제를 원할 때는 체크박스를 이용해 삭제할 문항을 선택해 삭제할 수 있다. 시험지를 작성할 때는 정답과 문항별 배점도 입력한다.

### 문제 출제

번호	내용	보기	정답	배점
		1) [입력칸]		
		2) [입력칸]		
		3) [입력칸]		
		4) [입력칸]		

[추가] [삭제] [저장]

[그림 5-12] 시험지작성 화면

[그림 5-13]은 작성된 시험지 내용을 관리하는 화면으로 시험지 내용을 변경할 수 있다.

### 시험지 내용

번호	내용	보기	정답	배점
1	배열선언이 올바르지 않는 것을 고르시오. (자바) [설명]	1) int s; 2) int[] s; 3) int s[]; 4) int    s;	1	1
2	다음 설명이 무엇을 말하는지 고르시오. [설명] 선형리스트의 양쪽 끝에서 압축력을 모두 할 수 있는 구조.	1) 스택 2) 큐 3) 데크 4) 리스트	3	2
3	다음은 스택에로의 삽입을 나타낸 코드이다. 빈칸(?)에 들어갈 알맞은 것을 고르시오. [코드] void add(int *top, element item) {	1) top++ 2) **top 3) *top++ 4) **(*top)	4	2

[저장] [확인]

[그림 5-13] 시험지관리 화면



[그림 5-14]는 해당 강좌를 수강하는 학생 목록이 나와 성적을 입력하는 화면이다. 과제물 성적과 시험 성적은 시스템에서 자동으로 채점되어 누적된 점수를 화면에 출력한다. 교수는 과제물과, 시험을 제외한 나머지 점수만 입력하면 된다.

### 성적

학번	중간고사	기말고사	과제물	출석	기타시험	총점	학점
200343001	<input type="text"/>	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	<input type="text"/>

[그림 5-14] 성적입력 화면

[그림 5-15]는 출석부 메뉴를 선택해서 출석부 내용을 보고, 원하면 excel파일로 저장하는 화면이다. 출석부는 강좌를 수강한 학생전체의 목록이 나온다.

## 2005 학년도 1학기 학기 출석부

학수번호: NK00100      교과목명: 자료구조      분반: 01      담당교수: 홍길동

학 과	10 주	11 주	12 주	13 주	14 주	15 주	16 주	출석인수
컴퓨터정보학								

[그림 5-15] 출석부 화면



### 5.3. 관리자

관리자가 교과목, 강좌, 학생, 교수 메뉴를 사용할 때 나타나는 화면들이다.

[그림 5-16]은 교과목 메뉴를 선택하면 교과목 목록이 나오는 화면이다.

+교과목 목록+

교과목번호	교과목이름	개설 학과	개설 학기	학점
NA00100	자료구조	컴퓨터정보학부	1학기	3
NA00200	프로그래밍	컴퓨터정보학부	1학기	3
NA00300	프로그래밍 언어론	컴퓨터정보학부	1학기	3

[그림 5-16] 교과목목록 화면

[그림 5-17]은 [그림 5-16]에서 등록버튼을 선택했을 경우, 보여지는 교과목등록 화면이다.

교과목 정보

교과목번호

교과목이름

개설 학과

전공 여부

열 면

개설 학기

교과목설명

학 점

시 간

표시 항목은 필수 기입사항입니다.

[그림 5-17] 교과목등록 화면

[그림 5-18]은 관리자가 교과목을 관리하는 화면이다. 관리자는 등



특된 교과목을 선택하여 내용을 볼 수 있다. 또한, 정보를 변경할 수 있다.

[그림 5-18] 교과목관리 화면

교과목메뉴와 강좌메뉴는 기능적인 면이 매우 유사하다. 강좌메뉴도 교과목메뉴처럼 메뉴를 선택하면 강좌목록이 나오고, 강좌를 새로 등록, 삭제할 수 있으며, 또한 강좌를 선택하면 해당 강좌 정보를 볼 수 있고, 내용을 변경하여 저장할 수 있다.

[그림 5-19]은 학생메뉴 선택 시 보여지는 학생관리 화면이다. 관리자는 이 곳에서 학생을 등록, 삭제할 수 있다. 학생이름을 선택하면 해당 학생의 정보가 [그림 5-20]처럼 보여진다. 이때, 학생정보를 수정할 수 있다.

\*수강신청한 학생 목록\*

학번	이름	학과
200343001	김소연	컴퓨터정보학부
200343002	김승희	컴퓨터정보학부
200343003	나연희	컴퓨터정보학부
200343004	나선희	컴퓨터정보학부

[등록] [삭제]

[그림 5-19] 학생관리 화면



· 학번	200343001
· 비밀번호	****
· 학년	3
· 성명	김소연
· 학과	컴퓨터정보학부
· Email	sykim1234@sungshin.ac.kr

※ 표시 항목은 필수 기입항목입니다.

저장   강좌   확인

[그림 5-20] 학생정보 화면

[그림 5-21]은 [그림 5-20]에서 강좌버튼을 선택했을 경우, 학생이 수강 등록한 강좌목록을 보여주는 화면이다. 관리자는 학생에 대한 강좌를 등록·삭제를 통해 관리할 수 있다.

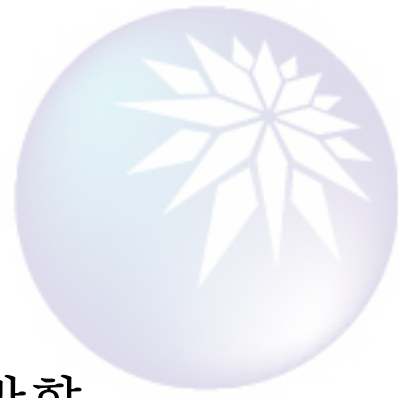
\*수강신청한 강좌 목록\*

교과목번호	교과목이름	분반
NA00100	자료구조	01

등록   삭제

[그림 5-21] 강좌목록 화면

교수메뉴도 학생메뉴와 기능적으로 매우 유사하다. 교수메뉴를 선택하면, 학생메뉴처럼 교수목록이 나오고 원하는 교수를 선택하면 교수 정보를 볼 수 있다. 물론, 교수정보에 대한 수정이 가능하고, 교수가 강의하는 강좌를 등록·삭제할 수 있다.



## 제 6 장 결론 및 향후 연구방향

인터넷의 급속한 발달과 주 5일제 수업 등이 맞물려 교육의 흐름이 바뀌고 있다. 이제는 학습자가 언제, 어디서나 시간과 장소에 구애 받지 않고 학습할 수 있는 시스템을 요구하고 있다. 웹 기반 교육이 수업의 보조도구 이상의 영역으로 자리잡는데 이러한 변화는 큰 몫을 하였다. 학교에서도 7차 교육과정에서 ICT활용 교육을 활성화하여 교수-학습 방법을 다양화하고 개선하여 학습의 효율을 제고하도록 하고 있다. 교육학술정보원에서는 ICT활용 교수-학습 과정안 모형을 제시하기도 하였다.

본 논문에서는 이러한 시대적 흐름에 발맞추어 오프라인에서 이루어지는 수업의 한계를 극복하고, 시간과 공간에 구애 없이 학습자가 학습을 할 수 있는 학습의 장을 만들어 학습의 효율성을 높이기 위해 XML기반 온라인 수업관리 System을 구현하였다.

XML기반 수업관리 System을 활용하였을 경우 기대되는 효과를 살펴보면 다음과 같다.

첫째, 강의시간적 제약으로 제시하지 못 하였거나, 부족했던 강의 내용을 수업관리 System을 통해 제공해 줄 수 있다.

둘째, 과제물을 작성해 일일이 학생들이 모아 교수님께 제출하지 않고, 수업관리 System을 이용해 제출함으로써, 간편하게 과제물을 제출할 수 있고, 과제물을 모으는 수고도 덜 수 있으며, 과제물을 안 갖고



왔을 경우나 잃어버렸을 경우의 불이익을 피할 수 있다. 또한, 교수에게는 많은 양의 과제물을 간편하게 채점하고, 처리할 수 있어 시간의 절약과 편리함을 제공한다.

셋째, 학생들이 수업 내용을 잘 이해하고 잘 따라오는지 수시로 확인 할 수 있도록 체크하는 것을 수업관리 System의 시험을 이용하면 학생과 교수 모두 쉽게 확인할 수 있고, 평가할 수 있다.

본 논문이 더욱더 능동적인 수업관리 System이 될 수 있도록, 전체 메일 기능이나 정해진 시간에 인터넷 채팅을 통해 상담을 할 수 있는 기능 등에 대한 후속 연구가 기대된다. 또한, 본 논문은 사용자의 입장에서 얼마나 많은 도움이 되는지 설문조사가 되어 있지 아니하므로 그 효과성과 실용성에 대한 검증 측면에 대한 후속 연구 또한 기대된다.



## 참고문헌 및 사이트

[1] Dick, W. & Reiser, R. Planning effective instruction, Englewood Cliffs, NJ: Prentice Hall, 1989.

[2] Dick, W. & Carey, L. The Systematic Design of Instruction (3rd ed.). Harper Collins Publishers, 1990.

[3] Gillani, B. B., & Relan, A. Incorporating interactivity and multimedia into web-based instruction. In Khan, B. D. (Ed.). Web-Based Instruction. Englewood Cliffs, NJ: Educational Technology Publications, 1997.

[4] Khan, B. H. Web-based instruction(WBI): What is it and why is it? In B. H. Khan(Ed.), *Web-based instruction*(pp. 5-18). Englewood Cliffs, NJ: Educational Technology Publications, 1997.

[5] McManus, Thomas Fox. "Delivering instruction on the world wide web. [On-Line]. Available", 1995.

[6] Ruth Colvin Clark, "Using Web-Based Training Wisely", Training, pp. 51-56, 1999.



[7] 김갑수, 이진민, ‘웹 에이전트를 이용한 웹 기반 교수-학습 시스템의 설계 및 개발’, 한국정보교육학회 제5권 제1호 pp69-81, 2001.

[8] 김영희, ‘웹기반 수업의 설계 및 구현(일과 에너지 단원을 중심으로)’, 건국대학교 교육대학원 석사학위논문, 2003.

[9] 이덕수, ‘수업 보조 도구로서의 온라인 강의실 시스템의 설계와 구현’, 홍익대학교 교육대학원 석사학위논문, 2001.

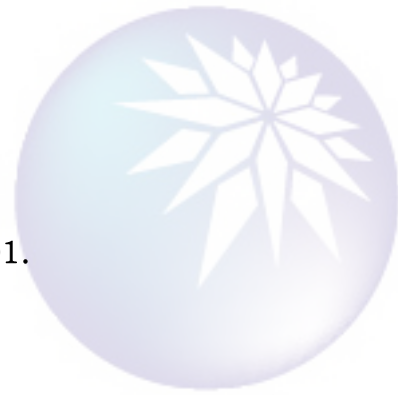
[10] 이종희, 이근왕, ‘멀티 에이전트를 이용한 코스 반복 학습시스템의 설계 및 구현’, 정보처리학회논문집B 제8-B 제6호, pp595-600, 2001.

[11] 이현희, ‘교사에이전트를 활용한 웹 기반 단계별 원격 교육 시스템의 설계와 구현’, 전남대학교 석사학위논문, 2001.

[12] 우홍권, ‘XML을 이용한 ICT활용 학습지도안 지원 교육시스템의 설계 및 구현’, 동의대학교 교육대학원 석사학위논문, 2004.

[13] 양경민, ‘XML기반 코스웨어 설계 및 구현’, 원광대학교 교육대학원 석사학위논문, 2000.

[14] 양원용, ‘웹을 이용한 여수 향토사 학습시스템’, 여수대학교 교육대학원 석사학위논문, 2004.



- [15] 김은중, '클릭하세요! JSP와 XML', 대림출판사, 2001.
- [16] 나일주, 정인성, '교육공학의 이해', 학지사, 1999.
- [17] 데이텍주식회사, 'XIS 교육교재', 데이텍주식회사, 2002.
- [18] 백영균, 강신천, '웹기반 학습환경의 준비와 개발도구', 원미사, 1999.
- [19] 최종명, 유재우, 최재영, '자바개발자를 위한 XML', 홍릉과학출판사, 2002.
- [20] Jon Duckett의 8인 공저/천주석 역, 'professional XML Schema', 정보문화사, 2002.



# ABSTRACT

## Design and Implementation of a XML based Online Class Management System

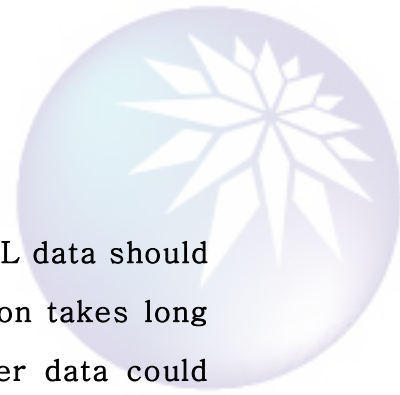
Ahn, Mi Sook

Major in Computer Science Education

Graduate School of Education

Sungshin Women's University

Numerous research projects which are related to XML are being conducted actively. Specially, there were many research projects for the online education systems based on XML. Most of them simply replace HTML with XML DTD instead of XML Schema, and use general relational databases to store XML contents. But, when



they store XML data by use of relational database, XML data should be converted to relation model schema. The conversion takes long time, and has impedence mismatch problem. Moreover data could be duplicated throughout the several tables.

In this paper, our system operates as follows. Teachers provide students with the materials on lectures, homeworks and examinations. Students can get the information and take the examination through the system. To implement this system efficiently, we have chosen XML schema to represent the logical structure of XML contents and have used “eXcelon database” (XML Customized Database) to make it possible to organize, share and search more efficiently than general RDBMS does.



## 부 록

학생스키마[student.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 학생 정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:
5:   <xs:element name="studentinfo">
6:     <xs:complexType>
7:       <xs:sequence>
8:         <xs:element ref="student" minOccurs="0" maxOccurs="unbounded"/>
9:       </xs:sequence>
10:    </xs:complexType>
11:    <xs:key name="KeyStudentNO">
12:      <xs:selector xpath="student"/>
13:      <xs:field xpath="student_no"/>
14:    </xs:key>
15:  </xs:element>
16:
17:  <xs:element name="student">
18:    <xs:complexType>
19:      <xs:sequence>
20:        <xs:element name="student_no" type="xs:integer"/>
21:        <xs:element name="student_pw" type="xs:string"/>
22:        <xs:element name="student_year" type="xs:integer"/>
23:        <xs:element name="student_name" type="xs:string"/>
24:        <xs:element name="student_dept" type="xs:string"/>
25:        <xs:element name="student_email" type="xs:string"/>
26:      </xs:sequence>
27:    </xs:complexType>
28:  </xs:element>
29: </xs:schema>
```



## 교수 스키마 [prof.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 교수정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:
5:
6:   <xs:element name="professorInfo">
7:     <xs:complexType>
8:       <xs:sequence>
9:         <xs:element ref="professor" minOccurs="0" maxOccurs="unbounded"/>
10:      </xs:sequence>
11:    </xs:complexType>
12:
13:    <xs:key name="KeyProfessorNO">
14:      <xs:selector xpath="professor"/>
15:      <xs:field xpath="prof_no"/>
16:    </xs:key>
17:
18:  </xs:element>
19:
20:  <xs:element name="professor">
21:    <xs:complexType>
22:      <xs:sequence>
23:        <xs:element name="prof_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
24:        <xs:element name="prof_name" type="xs:string" minOccurs="1" maxOccurs="1"/>
25:        <xs:element name="prof_pw" type="xs:string" minOccurs="1" maxOccurs="1"/>
26:        <xs:element name="prof_dept" type="xs:string" minOccurs="1" maxOccurs="1"/>
27:        <xs:element name="prof_room" type="xs:string" minOccurs="1" maxOccurs="1"/>
28:        <xs:element ref="prof_contact" minOccurs="1" maxOccurs="1"/>
29:      </xs:sequence>
30:    </xs:complexType>
31:  </xs:element>
32:
33:  <xs:element name="prof_contact">
34:    <xs:complexType>
35:      <xs:sequence>
36:        <xs:element ref="prof_cellularPhone" minOccurs="1" maxOccurs="1"/>
37:        <xs:element ref="prof_email" minOccurs="1" maxOccurs="1"/>
38:        <xs:element ref="prof_officeTel" minOccurs="1" maxOccurs="1"/>
39:      </xs:sequence>
40:    </xs:complexType>
41:
42:  </xs:element>
43:  <xs:element name="prof_cellularPhone" type="xs:string"/>
44:  <xs:element name="prof_email" type="xs:string"/>
45:  <xs:element name="prof_officeTel" type="xs:string"/>
46: </xs:schema>
```



## 교과목 스키마 [course.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 교과목정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:     <xs:simpleType name="DEPT_kind">
5:         <xs:restriction base="xs:string">
6:             <xs:enumeration value="컴퓨터정보학부"/>
7:             <xs:enumeration value="미디어정보학부"/>
8:         </xs:restriction>
9:     </xs:simpleType>
10:    <xs:simpleType name="SEMESTER_kind">
11:        <xs:restriction base="xs:string">
12:            <xs:enumeration value="1학기"/>
13:            <xs:enumeration value="2학기"/>
14:        </xs:restriction>
15:    </xs:simpleType>
16:    <xs:simpleType name="NO_kind">
17:        <xs:restriction base="xs:string">
18:            <xs:pattern value="NA[0-9]{5}|NB[0-9]{5}/>
19:        </xs:restriction>
20:    </xs:simpleType>
21:    <xs:element name="courseinfo">
22:        <xs:complexType>
23:            <xs:sequence>
24:                <xs:element name="course" minOccurs="0" maxOccurs="unbounded">
25:                    <xs:complexType>
26:                        <xs:sequence>
27:                            <xs:element name="course_no" type="NO_kind" minOccurs="1" maxOccurs="1"/>
28:                            <xs:element name="course_name" type="xs:string" minOccurs="1" maxOccurs="1"/>
29:                            <xs:element name="course_dept" type="DEPT_kind" minOccurs="1" maxOccurs="1"/>
30:                            <xs:element name="course_type" type="xs:string" minOccurs="1" maxOccurs="1"/>
31:                            <xs:element name="course_province" type="xs:integer" minOccurs="1" maxOccurs="1"/>
32:                            <xs:element name="course_semester" type="SEMESTER_kind" minOccurs="1" maxOccurs="1"/>
33:                            <xs:element name="course_explanation" type="xs:string" minOccurs="1" maxOccurs="1"/>
34:                            <xs:element name="course_credit" type="xs:integer" minOccurs="1" maxOccurs="1"/>
35:                            <xs:element name="course_hour" type="xs:integer" minOccurs="1" maxOccurs="1"/>
36:                        </xs:sequence>
37:                    </xs:complexType>
38:                </xs:element>
39:            </xs:sequence>
40:        </xs:complexType>
41:        <xs:key name="KeyCourseNO">
42:            <xs:selector xpath="course"/>
43:            <xs:field xpath="course_no"/>
44:        </xs:key>
45:    </xs:element>
46: </xs:schema>
```



## 수강 스키마 [register.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 수강정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:   <xs:include schemaLocation="student.xsd"/>
5:   <xs:include schemaLocation="section.xsd"/>
6:
7:   <xs:element name="registerinfo">
8:     <xs:complexType>
9:       <xs:sequence>
10:        <xs:element ref="register" minOccurs="0" maxOccurs="unbounded"/>
11:      </xs:sequence>
12:
13:    </xs:complexType>
14:    <xs:keyref name="RefSectionNo_Register" refer="KeySectionNO">
15:      <xs:selector xpath="register"/>
16:      <xs:field xpath="sectionID"/>
17:    </xs:keyref>
18:    <xs:keyref name="RefStudentNo_Register" refer="KeyStudentNO">
19:      <xs:selector xpath="register"/>
20:      <xs:field xpath="student_no"/>
21:    </xs:keyref>
22:  </xs:element>
23:  <xs:element name="register">
24:    <xs:complexType>
25:      <xs:sequence>
26:        <xs:element name="sectionID" type="xs:string" minOccurs="1" maxOccurs="1"/>
27:        <xs:element name="student_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
28:        <xs:element name="reg_attendance" type="xs:integer" minOccurs="1" maxOccurs="1"/>
29:        <xs:element name="reg_homework" type="xs:integer" minOccurs="1" maxOccurs="1"/>
30:        <xs:element name="reg_midtern_exam" type="xs:integer" minOccurs="1" maxOccurs="1"/>
31:        <xs:element name="reg_final_exam" type="xs:integer" minOccurs="1" maxOccurs="1"/>
32:        <xs:element name="reg_other_paper" type="xs:integer" minOccurs="1" maxOccurs="1"/>
33:        <xs:element name="reg_total" type="xs:integer" minOccurs="1" maxOccurs="1"/>
34:        <xs:element name="reg_grade" type="xs:string" minOccurs="1" maxOccurs="1"/>
35:      </xs:sequence>
36:    </xs:complexType>
37:  </xs:element>
38:
39: </xs:schema>
```

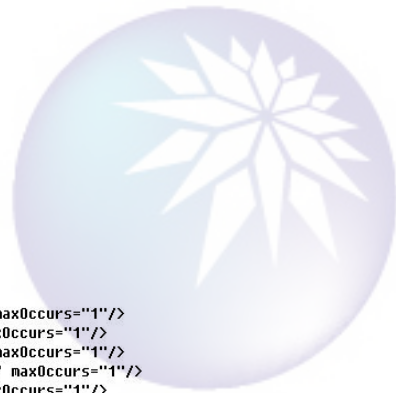


## 강좌 스키마[section.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 강좌정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:   <xs:include schemaLocation="course.xsd"/>
5:   <xs:include schemaLocation="prof.xsd"/>
6:
7:   <xs:simpleType name="ClassType">
8:     <xs:restriction base="xs:string">
9:       <xs:pattern value="[0-9]{2}"/>
10:    </xs:restriction>
11:  </xs:simpleType>
12:  <xs:simpleType name="YearType">
13:    <xs:restriction base="xs:integer">
14:      <xs:pattern value="2[0-9]{3}"/>
15:    </xs:restriction>
16:  </xs:simpleType>
17:  <xs:element name="section_hour">
18:    <xs:complexType>
19:      <xs:sequence>
20:        <xs:element name="hour" type="xs:string" minOccurs="1" maxOccurs="1"/>
21:        <xs:element name="location" type="xs:string" minOccurs="1" maxOccurs="1"/>
22:      </xs:sequence>
23:    </xs:complexType>
24:  </xs:element>
25:  <xs:element name="sectionInfo">
26:    <xs:complexType>
27:      <xs:sequence>
28:        <xs:element name="section" minOccurs="0" maxOccurs="unbounded">
29:          <xs:complexType>
30:            <xs:sequence>
31:              <xs:element name="section_no" type="xs:string" minOccurs="0" maxOccurs="1"/>
32:              <xs:element name="course_no" type="xs:string" minOccurs="0" maxOccurs="1"/>
33:              <xs:element name="prof_no" type="xs:integer" minOccurs="0" maxOccurs="1"/>
34:              <xs:element name="section_year" type="YearType" minOccurs="0" maxOccurs="1"/>
35:              <xs:element name="section_classno" type="ClassType" minOccurs="0" maxOccurs="1"/>
36:              <xs:element name="section_complement" type="xs:integer" minOccurs="0" maxOccurs="1"/>
37:              <xs:element ref="section_hour" minOccurs="0" maxOccurs="unbounded"/>
38:              <xs:element name="section_note" type="xs:string" minOccurs="0" maxOccurs="1"/>
39:              <xs:element ref="lecturePlan" minOccurs="0" maxOccurs="1"/>
40:            </xs:sequence>
```



```
41:         </xs:complexType>
42:     </xs:element>
43: </xs:sequence>
44: </xs:complexType>
45:
46: <xs:key name="KeySectionNO">
47:   <xs:selector xpath="section"/>
48:   <xs:field xpath="section_no"/>
49: </xs:key>
50: <xs:keyref name="RefCourseNO_Section1" refer="KeyCourseNO">
51:   <xs:selector xpath="./section"/>
52:   <xs:field xpath="course_no"/>
53: </xs:keyref>
54: <xs:keyref name="RefProfNO_Section" refer="KeyProfessorNO">
55:   <xs:selector xpath="./section"/>
56:   <xs:field xpath="prof_no"/>
57: </xs:keyref>
58: </xs:element>
59:
60: <!-- lecturePlan 강의계획서 내용 -->
61: <xs:element name="lecturePlan">
62:   <xs:complexType>
63:     <xs:sequence>
64:       <xs:element name="lectureplan_consultation" type="xs:string" minOccurs="0" maxOccurs="1"/>
65:       <xs:element name="lectureplan_goal" type="xs:string" minOccurs="0" maxOccurs="1"/>
66:       <xs:element name="lectureplan_materialbook" type="xs:string" minOccurs="0" maxOccurs="1"/>
67:       <xs:element name="lectureplan_referencematerial" type="xs:string" minOccurs="0" maxOccurs="1"/>
68:       <xs:element ref="lectureplan_way" minOccurs="0" maxOccurs="1"/>
69:       <xs:element ref="lectureplan_resource" minOccurs="0" maxOccurs="1"/>
70:       <xs:element ref="lectureplan_evaluation" minOccurs="0" maxOccurs="1"/>
71:       <xs:element ref="lectureplan_contents" minOccurs="0" maxOccurs="1"/>
72:     </xs:sequence>
73:   </xs:complexType>
74: </xs:element>
75:
76: <xs:element name="lectureplan_way">
77:   <xs:complexType>
78:     <xs:sequence>
79:       <xs:element name="theory" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
80:       <xs:element name="practice" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
81:       <xs:element name="discussion" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
82:       <xs:element name="seminar" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
83:       <xs:element name="publication" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
84:       <xs:element name="exploration" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
85:       <xs:element name="questionAnswer" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
86:       <xs:element name="direction" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
87:       <xs:element name="teamteaching" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
88:       <xs:element name="others" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
89:     </xs:sequence>
90:   </xs:complexType>
91: </xs:element>
92:
93: <xs:element name="lectureplan_resource">
94:   <xs:complexType>
95:     <xs:sequence>
96:       <xs:element name="video" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
97:       <xs:element name="audio" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
98:       <xs:element name="ohp" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
99:       <xs:element name="slide" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
100:      <xs:element name="computer" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
101:      <xs:element name="toolothers" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
102:    </xs:sequence>
103:   </xs:complexType>
104: </xs:element>
```



```
105:
106: <xs:element name="lectureplan_evaluation">
107:   <xs:complexType>
108:     <xs:sequence>
109:       <xs:element name="midyearsRate" type="xs:integer" minOccurs="0" maxOccurs="1"/>
110:       <xs:element name="finalsRate" type="xs:integer" minOccurs="0" maxOccurs="1"/>
111:       <xs:element name="homeworkRate" type="xs:integer" minOccurs="0" maxOccurs="1"/>
112:       <xs:element name="attendanceRate" type="xs:integer" minOccurs="0" maxOccurs="1"/>
113:       <xs:element name="othersRate" type="xs:integer" minOccurs="0" maxOccurs="1"/>
114:     </xs:sequence>
115:   </xs:complexType>
116: </xs:element>
117:
118: <xs:element name="lectureplan_contents">
119:   <xs:complexType>
120:     <xs:sequence>
121:       <xs:element ref="week" minOccurs="0" maxOccurs="16"/>
122:     </xs:sequence>
123:   </xs:complexType>
124: </xs:element>
125:
126: <xs:element name="week">
127:   <xs:complexType>
128:     <xs:sequence>
129:       <xs:element name="week_no" type="xs:string" minOccurs="1" maxOccurs="1"/>
130:       <xs:element name="week_content" type="xs:string" minOccurs="1" maxOccurs="1"/>
131:     </xs:sequence>
132:   </xs:complexType>
133: </xs:element>
134: </xs:schema>
```



## 과제물 스키마 [homework.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 과제물정보 스키마 -->
3:
4: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
5: <xs:include schemaLocation="section.xsd"/>
6:
7: <xs:element name="homeworkinfo">
8:   <xs:complexType>
9:     <xs:sequence>
10:       <xs:element ref="homework" minOccurs="0" maxOccurs="unbounded"/>
11:     </xs:sequence>
12:     <xs:attribute ref="sectionID"/>
13:   </xs:complexType>
14:
15:   <xs:keyref name="RefSectionN01" refer="KeySectionN0">
16:     <xs:selector xpath="homeworkinfo"/>
17:     <xs:field xpath="@sectionID"/>
18:   </xs:keyref>
19:
20: </xs:element>
21:
22:
23: <xs:element name="homework">
24:   <xs:complexType>
25:     <xs:sequence>
26:       <xs:element name="homework_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
27:       <xs:element name="homework_title" type="xs:string" minOccurs="1" maxOccurs="1"/>
28:       <xs:element name="homework_contents" type="xs:string" minOccurs="1" maxOccurs="1"/>
29:       <xs:element name="homework_perfectscore" type="xs:integer" minOccurs="1" maxOccurs="1"/>
30:       <xs:element name="homework_senddate" type="xs:date" minOccurs="1" maxOccurs="1"/>
31:       <xs:element name="homework_writedate" type="xs:date" minOccurs="1" maxOccurs="1"/>
32:       <xs:element name="homework_file" type="xs:anyURI" minOccurs="1" maxOccurs="1"/>
33:     </xs:sequence>
34:   </xs:complexType>
35:
36:   <xs:key name="KeyHomeWorkN0">
37:     <xs:selector xpath="homework"/>
38:     <xs:field xpath="homework_no"/>
39:   </xs:key>
40: </xs:element>
41:
42: <xs:attribute name="sectionID" type="xs:string"/>
```



## 과제물 제출 스키마[homeworkSubmit.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 과제물제출 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:
5:     <xs:include schemaLocation="student.xsd"/>
6:     <xs:include schemaLocation="section.xsd"/>
7:     <xs:include schemaLocation="homework.xsd"/>
8:
9:     <xs:element name="homeworkSubmitInfo">
10:         <xs:complexType>
11:             <xs:sequence>
12:                 <xs:element ref="HomeworkSubmit" minOccurs="1" maxOccurs="unbounded"/>
13:             </xs:sequence>
14:             <xs:attribute name="sectionID" type="xs:string"/>
15:         </xs:complexType>
16:
17:         <xs:keyref name="RefSessionNO3" refer="KeySessionNO">
18:             <xs:selector xpath="homeworkSubmitInfo"/>
19:             <xs:field xpath="@sectionID"/>
20:         </xs:keyref>
21:     </xs:element>
22:
23:     <xs:element name="HomeworkSubmit">
24:         <xs:complexType>
25:             <xs:sequence>
26:                 <xs:element name="homework_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
27:                 <xs:element ref="HomeworkSubmitContent" minOccurs="1" maxOccurs="unbounded"/>
28:             </xs:sequence>
29:         </xs:complexType>
30:
31:         <xs:keyref name="RefStudentNO1" refer="KeyStudentNO">
32:             <xs:selector xpath="HomeworkSubmit"/>
33:             <xs:field xpath="homework_no"/>
34:         </xs:keyref>
35:     </xs:element>
36:
37:
38:     <xs:element name="HomeworkSubmitContent">
39:         <xs:complexType>
40:             <xs:sequence>
41:                 <xs:element name="student_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
42:                 <xs:element name="homeworksubmit_date" type="xs:date" minOccurs="1" maxOccurs="1"/>
43:                 <xs:element name="homeworksubmit_score" type="xs:integer" minOccurs="1" maxOccurs="1"/>
44:                 <xs:element name="homeworksubmit_file" type="xs:anyURI" minOccurs="1" maxOccurs="1"/>
45:             </xs:sequence>
46:         </xs:complexType>
47:
48:         <xs:keyref name="RefHomeWorkNO1" refer="KeyHomeWorkNO">
49:             <xs:selector xpath="HomeworkSubmitContent"/>
50:             <xs:field xpath="student_no"/>
51:         </xs:keyref>
52:     </xs:element>
53:
54:
55: </xs:schema>
```



## 강의자료 스키마[lectureNote.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 강의자료 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:   <xs:include schemaLocation="section.xsd"/>
5:
6:   <xs:element name="lectureNoteInfo">
7:     <xs:complexType>
8:       <xs:sequence>
9:         <xs:element ref="lectureNoteContent" minOccurs="1" maxOccurs="unbounded"/>
10:      </xs:sequence>
11:      <xs:attribute name="sectionID" type="xs:string"/>
12:    </xs:complexType>
13:    <xs:keyref name="RefSectionNo_LectureNote" refer="KeySectionNO">
14:      <xs:selector xpath="lectureNoteInfo"/>
15:      <xs:field xpath="@sectionID"/>
16:    </xs:keyref>
17:  </xs:element>
18:
19:  <xs:element name="lectureNoteContent">
20:    <xs:complexType>
21:      <xs:sequence>
22:        <xs:element name="lectureNote_no" type="xs:string" minOccurs="1" maxOccurs="1"/>
23:        <xs:element name="lectureNote_title" type="xs:string" minOccurs="1" maxOccurs="1"/>
24:        <xs:element name="lectureNote_contents" type="xs:string" minOccurs="1" maxOccurs="1"/>
25:        <xs:element name="lectureNote_date" type="xs:date" minOccurs="1" maxOccurs="1"/>
26:        <xs:element name="lectureNote_file" type="xs:anyURI" minOccurs="1" maxOccurs="1"/>
27:      </xs:sequence>
28:    </xs:complexType>
29:  </xs:element>
30: </xs:schema>
```



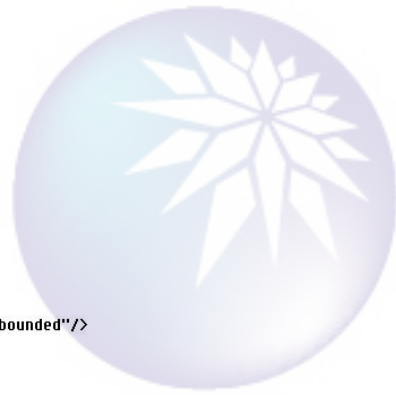
## 시험지 스키마[paper.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 시험지정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:   <xs:include schemaLocation="course.xsd"/>
5:   <xs:include schemaLocation="section.xsd"/>
6:
7:   <xs:element name="paperInfo">
8:     <xs:complexType>
9:       <xs:sequence>
10:        <xs:element ref="paper" minOccurs="0" maxOccurs="unbounded"/>
11:      </xs:sequence>
12:      <xs:attribute name="sectionID" type="xs:string"/>
13:    </xs:complexType>
14:
15:    <xs:keyref name="RefSessionN06" refer="KeySectionN0">
16:      <xs:selector xpath="paperInfo"/>
17:      <xs:field xpath="@sectionID"/>
18:    </xs:keyref>
19:  </xs:element>
20:
21:  <xs:element name="paper">
22:    <xs:complexType>
23:      <xs:sequence>
24:        <xs:element name="paper_no" type="xs:string" minOccurs="0" maxOccurs="1"/>
25:        <xs:element name="paper_title" type="xs:string" minOccurs="0" maxOccurs="1"/>
26:        <xs:element ref="paperContents" minOccurs="0" maxOccurs="unbounded"/>
27:      </xs:sequence>
28:    </xs:complexType>
29:
30:    <xs:key name="KeyPaperNo">
31:      <xs:selector xpath="paper"/>
32:      <xs:field xpath="paper_no"/>
33:    </xs:key>
34:
35:  </xs:element>
36:
37:  <xs:element name="paperContents">
38:    <xs:complexType>
39:      <xs:sequence>
40:        <xs:element name="question_no" type="xs:integer" minOccurs="1" maxOccurs="1"/>
41:        <xs:element name="question_content" type="xs:string" minOccurs="1" maxOccurs="1"/>
42:        <xs:element name="question_score" type="xs:integer" minOccurs="1" maxOccurs="1"/>
43:        <xs:element name="question_answer" type="xs:string" minOccurs="1" maxOccurs="1"/>
44:        <xs:element ref="instance" minOccurs="0" maxOccurs="4"/>
45:      </xs:sequence>
46:    </xs:complexType>
47:  </xs:element>
48:  <xs:element name="instance">
49:    <xs:complexType>
50:      <xs:sequence>
51:        <xs:element name="instance_no" type="xs:string" minOccurs="1" maxOccurs="1"/>
52:        <xs:element name="instance_cont" type="xs:string" minOccurs="1" maxOccurs="1"/>
53:      </xs:sequence>
54:    </xs:complexType>
55:  </xs:element>
56: </xs:schema>
```



## 답안지 스키마[QuestionAnswer.xsd]

```
1: <?xml version="1.0" encoding="ISO-10646-UCS-2"?>
2: <!-- 답안지정보 스키마 -->
3: <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
4:
5:     <xs:include schemaLocation="section.xsd"/>
6:     <xs:include schemaLocation="student.xsd"/>
7:     <xs:include schemaLocation="paper.xsd"/>
8:
9:     <xs:element name="QuestionAnswerInfo">
10:        <xs:complexType>
11:            <xs:sequence>
12:                <xs:element ref="question_answer" minOccurs="0" maxOccurs="unbounded"/>
13:            </xs:sequence>
14:
15:            <xs:attribute name="sectionID" type="xs:string"/>
16:        </xs:complexType>
17:        <xs:keyref name="RefSessionN07" refer="KeySectionN0">
18:            <xs:selector xpath="QuestionAnswerInfo"/>
19:            <xs:field xpath="@sectionID"/>
20:        </xs:keyref>
21:    </xs:element>
22:
23:    <xs:element name="question_answer">
24:        <xs:complexType>
25:            <xs:sequence>
26:
27:                <xs:element ref="QA_contents" minOccurs="0" maxOccurs="unbounded"/>
28:            </xs:sequence>
29:            <xs:attribute name="paper_no" type="xs:integer"/>
30:        </xs:complexType>
31:
32:        <xs:keyref name="RefPaperN0" refer="KeyPaperN0">
33:            <xs:selector xpath="question_answer"/>
34:            <xs:field xpath="@paper_no"/>
35:        </xs:keyref>
36:    </xs:element>
37:
```



```
38:
39: <xs:element name="QA_contents">
40:   <xs:complexType>
41:     <xs:sequence>
42:       <xs:element ref="contents" minOccurs="0" maxOccurs="unbounded"/>
43:     </xs:sequence>
44:     <xs:attribute name="studentID" type="xs:integer"/>
45:   </xs:complexType>
46:
47:   <xs:keyref name="RefStudentN02" refer="KeyStudentN0">
48:     <xs:selector xpath="QA_contents"/>
49:     <xs:field xpath="@studentID"/>
50:   </xs:keyref>
51: </xs:element>
52:
53: <xs:element name="contents">
54:   <xs:complexType>
55:     <xs:sequence>
56:       <xs:element name="question_no" type="xs:integer" minOccurs="0" maxOccurs="1"/>
57:       <xs:element name="write_answer" type="xs:string" minOccurs="0" maxOccurs="1"/>
58:       <xs:element name="getPoint" type="xs:integer" minOccurs="0" maxOccurs="1"/>
59:     </xs:sequence>
60:   </xs:complexType>
61: </xs:element>
62:
63: </xs:schema>
```