

XML기반의 수강신청 관리 시스템 설계 및 구현

윤호군
목원대학교 컴퓨터교육과

Design and Implementation of Student Taking a course application
Management System Based on XML

Ho-Kun Yun
Mokwon University

요 약

효율적인 데이터 관리 및 처리를 위하여 XML이 웹 기반 응용 시스템에 적용되었다. 또한, ASP는 쉬운 웹 기반 프로그래밍의 구현을 가능하게 하였으며, UML은 시스템 모델링 기법의 새로운 대안으로 연구되고있다.

이에 관하여, 본 논문에서는 XML, ASP, UML을 이용하여 학생 수강신청 관리 시스템을 구현하였다. 이는, 학생모드와 관리자 모드로 구성되었으며, 효과적인 데이터 관리 및 처리를 위한 XML을 이용하였다. 또한, 시스템 분석 및 설계를 위하여 UML을 이용하였으며, 전체 시스템은 ASP로 구현하였다. 그리고, 데이터베이스는 MS-ACCESS를 이용하였으며, 웹 문서의 효율적인 활용을 위하여 Java Script 언어를 사용하였다.

키워드 : XML, ASP, 학생 수강관리 시스템, UML

Abstract

XML was applied to web-Based application system for efficient data supervision and processing. Also, ASP permitted embodiment of easy web-Based programming, and UML is studied by the new alternative of system modelling techniques.

Thus, In this paper see embodied student taking a course application management system using XML, ASP, UML. This was composed, and used XML for effective data supervision and processing by student mode and administrator mode. Also, It used UML for systems analysis and design, and whole system implemented by ASP. And, Database implemented to used in MS-ACCESS, and it used Java Script language for efficient practical use of web document.

Keyword : XML, ASP, Student taking a course application management system, UML

1. 서론

인터넷의 급속한 보급과 발전은 정보처리 사회의 혁명을

일으켰으며 웹을 기반으로 하는 지식사회로 만들었다. 웹은 우리에게 정보표현능력과 정보가공능력을 제공하면서 대중성을 확보하게 되었다[1]. 그리하여, 이를 교육에도 활

용하고자 하는 많은 노력이 전세계적으로 교육자들에 의해 이루어지고 있다[2]. 이에 따라, 현재 대학의 학사행정업무의 많은 부분을 웹 기반 서비스로 도입하고 있으며 학사업무 전반에 걸쳐 그 효율을 높이고자하는 많은 노력을 기울이고 있다. 현재의 학사업무를 이루는 구성요소로는 크게 성적처리, 수강신청, 각종증명확인 및 처리, 학생 및 직원관리 등을 들 수 있다. 이들 중 성적처리와 수강신청은 교수와 전국각지에서 모여드는 학생들을 연결하며, 이러한 일을 인터넷을 활용하게되면 많은 시간의 낭비와 비용을 절감시킬 수 있다[3, 4].

특히, 수강신청의 경우는 최근 들어 대부분의 국내 대학들이 교육여건의 다변화와 시장경쟁의 원리에 따라 경쟁력 강화차원에서 많은 부분의 변혁이 일어나고 있다. 즉, 부전공, 복수전공 등의 문제와 학과간의 장벽을 허물고 있으며, 학부제, 최소학점인정 등의 특성화된 교육제도 시행에 의해 수강신청의 업무가 점차 다른 업무에 비해 복잡하고 다양해지고 있는 것이다[5, 6].

따라서, 이와 같은 업무지원을 위한 시스템구축은 서버에서 해석하고 동적인 웹 문서를 생성할 수 있는 서버차원의 언어가 유용하다[7].

이에, 본 논문에서는 서버차원의 웹 개발언어인 ASP(Active Server Page)를 이용하여 수강신청 및 성적조회를 할 수 있는 학사행정지원시스템을 구축하였다. 또한, 원활한 데이터 처리를 위하여 MS_ACCESS를 이용한 데이터베이스 시스템을 연동 하였으며, 데이터의 효과적인 관리 및 처리를 위해 웹 데이터 관리의 표준인 XML(eXtensible Markup Language)를 이용하여 ASP 프로그래밍과 연계하였다. 그리고, 웹 페이지의 효율적인 운용을 위하여 Java Script언어를 사용하였으며, 학생과 관리자의 접근모드를 따로 두어 수강신청 및 성적조회 업무를 제어하도록 하였으며, 시스템 분석 및 설계를 위하여 UML을 이용하였다.

2. 관련연구

2.1. 웹 문서 개발

2.1.1. ASP

ASP란 동적인 웹 문서를 만들 수 있는 페이지를 말하며 Windows NT와 Windows 2000에서의 IIS(Internet Information Server)나 Windows 95/98의 PWS(Personal

Web Server)상에서 지원되는 ISAPI를 기본으로 하는 솔루션이다[8].

즉, 클라이언트로부터 웹 문서의 요청이 왔을 경우, 서버는 해당 요구사항을 분석한 후 이를 HTML문서로 만들어 클라이언트에 보내준다. 이는 ASP이용한 서버와 클라이언트 사이의 동적인 구성으로서, 보다 효율적인 서버구축을 위하여 ASP와 데이터베이스의 연동을 구현하여 동적인 서비스 요청이 요구되어질 때 보다 효율적이고 유동적인 처리를 가능하게 한다.

2.1.2. XML

XML은 1996년 W3C(World Wide Web Consortium)의 후원으로 형성된 XML Working Group에 의해 개발되었다. XML의 특징은 다음과 같다.

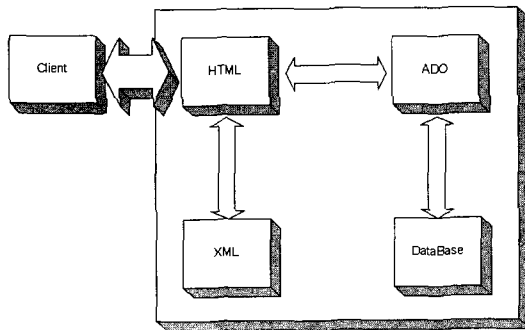
- 1) XML 파일을 작성하는 사용자는 내용에 관련된 태그를 직접 만들어 사용할 수 있다.
- 2) XML 파일은 오직 문서의 구조와 의미에 관한 정보만 들어가며, 요소를 꾸미는 부분은 스타일 시트로 분리되어 데이터로서 활용할 수 있다.
- 3) 태그가 문서와 밀접하게 관련되어있기 때문에 XML 파일자체가 잘 설계된 데이터베이스 역할을 할 수 있다. 따라서, XML로 표현된 인터넷 문서로부터 데이터를 뽑아 활용할 수 있으며 검색로봇을 이용하여 전 세계의 웹 페이지들로부터 정보를 모아 활용할 수 있다.
- 4) HTTP의 범용 프로토콜을 사용하기 때문에 응용범위가 넓다.
- 5) TEXT 기반이어서 평범한 TEXT 편집기로 XML을 만들 수 있다.
- 6) 특정 개발도구나 소프트웨어에 종속되지 않는다[11, 12, 13].

2.2 데이터베이스 연동

ASP와 데이터베이스의 연동부분에서는 사용자 질의가 웹 브라우저를 통하여 웹 서버에 전달되면, 서버측 ASP가 SQL을 이용하여 데이터베이스엔진으로 전송한다[9].

이를 위하여, ASP에서는 ODBC를 통하여 ADO(ActiveX Data Object) 객체를 이용한다. 즉, ADO객체의 구성 중에서 Connection 객체, Command 객체, Recordset 객체를 사용하여 데이터베이스를 처리한다[10]. 즉, Connection 객체는

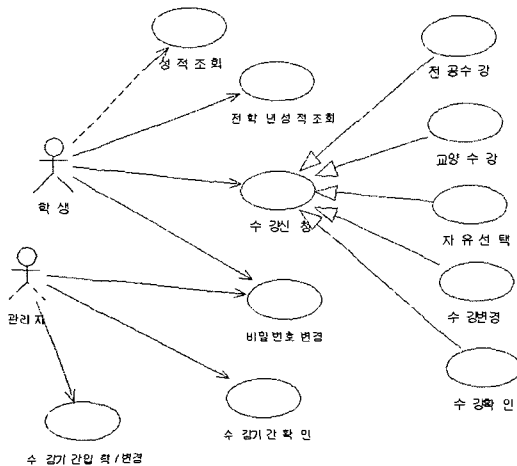
데이터베이스에 대한 연결정보를 갖고, Command 객체는 SQL을 실행하며, Recordset 객체는 데이터베이스에 SQL을 통하여 명령을 주고 그 결과 값을 받는 객체를 말한다. 그러나, 사실상 이들 객체들은 데이터베이스의 연결에서 실행까지 모두 할 수 있는 것이다. 다음 <그림 1>는 웹에서의 연동 관계를 나타낸 것이다.



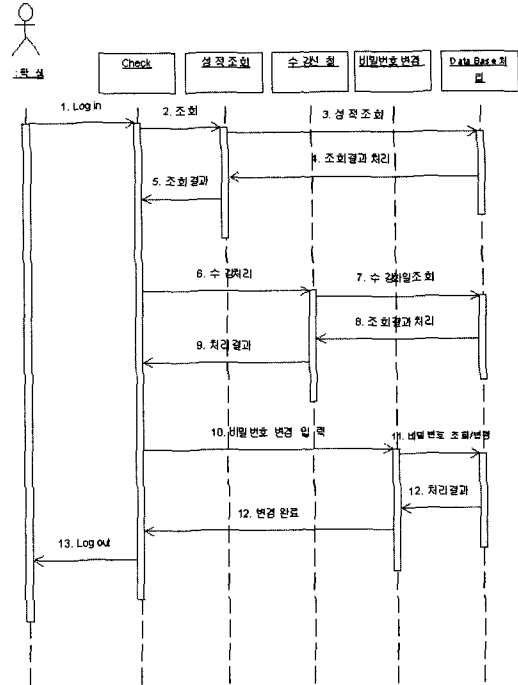
<그림 1> 웹 데이터베이스 연동

3. 시스템 분석 및 설계

시스템 분석을 위하여 먼저 사용자의 요구사항 및 필요구성 요소들을 분석하였다. 즉, 학생과 관리자 모드를 구별하여 해당 업무를 수행토록 하였으며, 수강신청은 전공, 교양, 자유선택과목을 포함하도록 하였다. 이에, <그림 2>는 본 시스템의 Usecase Diagram을 나타낸다.

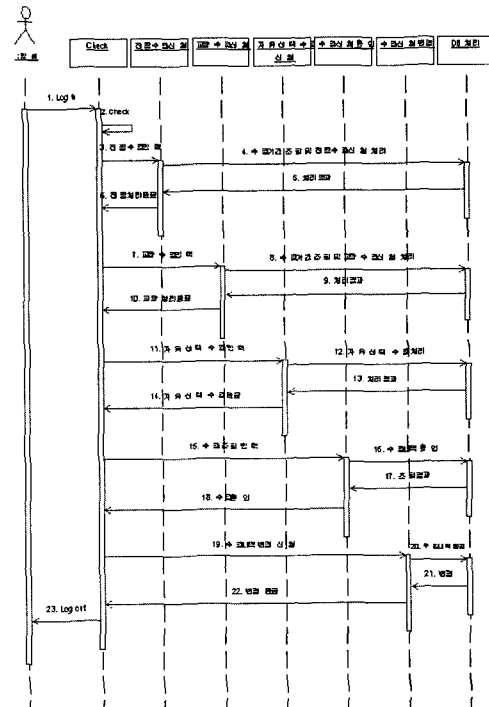


<그림 2> 본 시스템의 Usecase Diagram



<그림 3> 학생모드 Sequence Diagram

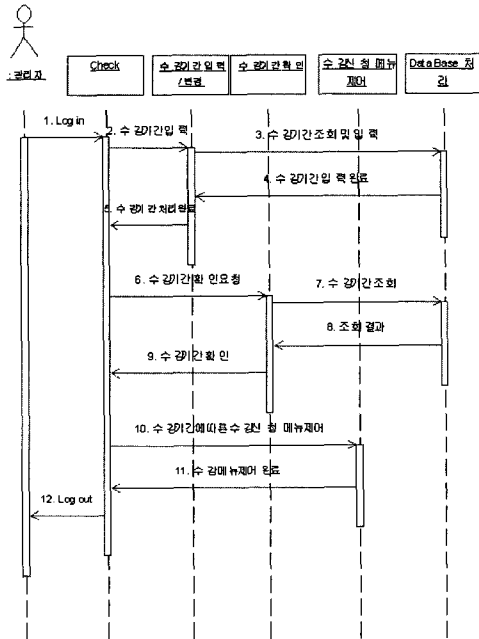
학생모드에서 로그-인을 하였을 경우 로그아웃까지의 순차도는 <그림 3>과 같이 나타내며, <그림 4>는 학생모드에



<그림 4> 수강신청 Sequence Diagram

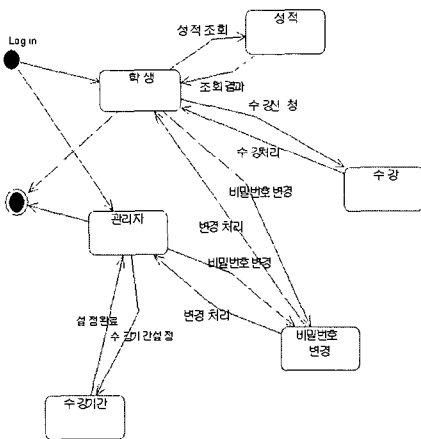
서 수강신청업무에 관한 순차도를 나타낸다. 또한, <그림 5>는 관리자모드에서의 순차도를 나타낸다.

즉, 관리자모드에서 수강기간을 입력/변경하면 해당기간에 따라 학생모드에서는 수강신청메뉴를 제어하도록 하였다.

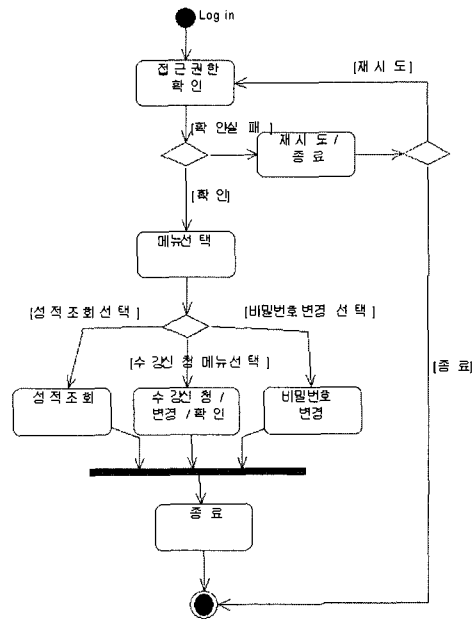


<그림 5> 관리자모드 Sequence Diagram

또한, 로그-인에서 로그-아웃까지의 작업흐름에 관한 상태도(State Diagram)는 <그림 6>과 같으며, <그림 7>은 활동도(Activity Diagram)를 나타낸다.



<그림 6> 본 시스템의 State Diagram

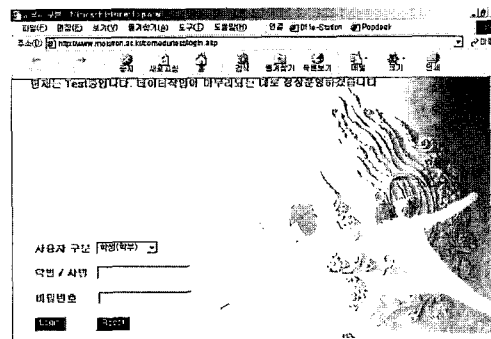


<그림 7> 본 시스템의 Activity Diagram

4. 구현

본 시스템의 구현은 Windows 2000 Server 환경의 IIS기반에서 ASP를 이용하였다. 데이터베이스는 MS-ACCESS를 이용하였고, ODBC에 학사데이터베이스를 등록한 후 ASP의 ADO객체를 이용하였다.

또한, 메뉴방식은 Java Script를 이용하여 구현하였으며, 화면구성은 Namo 3.0을 이용하였다. 그리고, 그래픽 처리를 위하여 Photoshop 6.0 과 Paint Shop 6을 이용하였다. 이에 따라, <그림 8>은 초기 로그인 화면을 나타낸다.



<그림 8> 초기 화면

로그인 후, 각 사용자 구분에 따라 아래와 같이 사용자 로그인을 구현하였으며, 대학원, 교수, 상사, 직원의 로그인은 학부의 인증 프로세스에서 데이터베이스의 검색 테이블 부분과 로그인 확인 후의 프로세스인 "Response.Redirect" 부분만이 다르다.

```

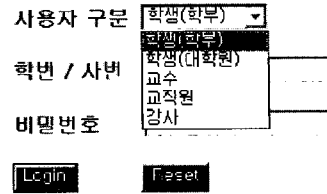
<!-- #include file="ADOVBS.INC" -->
<%
    Dim InputID, InputPass, Gubun, DBName
    DBName = "2001"
    InputID = Request("UserID")
    InputPass = Request("UserPassword")
    Gubun = Request("User")
    Set DBConn = Server.CreateObject
        ("ADODB.Connection")
    DBConn.Open DBName
    Select case Gubun
    case "학부"
        Set DBRect = Server.CreateObject
            ("ADODB.Recordset")
        sql = "Select * From JaeHak Where hakbun
            ='" & InputID & "'"
        DBRect.Open sql, DBConn
        If Not DBRect.EOF Then
            If UCase(DBRect("Password")) =
                UCase(InputPass) Then
                Session("Member") = True
                session("ID") = DBRect("hakbun")
                session("HakKwa") = DBRect("hakkwa")
                session("Name") = DBRect("name")
                session("Job") = DBRect("job")
                session("Hakryun") = DBRect("haknyun")
                session("Hakki") = DBRect("hakki")
                Response.Redirect "Success_St_frame.asp"
            Else
                Session("Member") = False
            session("ID") = DBRect("hakbun")
                Response.Redirect "Err_psw.htm"
            End if
        Else
            Session("Member") = False
            Response.Redirect "Err_ID.htm"
        End If
    case "대학원"
        *** 대학원생 로그인 인증 ***
    case "교수"
        *** 교수 로그인 인증 ***
    case "교직원"
        *** 교직원 로그인 인증 ***
    case "강사"
        *** 강사 로그인 인증 ***
    End Select

    DBRect.Close
    DBConn.Close
%>

```

다음 <그림 9>는 이를 구현한 사용자의 구분화면을 나타낸다. 또한, 본 시스템에서는 교직원이 관리자의 역할을 하

도록 하였다.



<그림 9> 사용자 구분

해당 학기의 성적 조회는 현재의 "Month" 를 검색 후 3월에서 7월 사이면 1학기, 아니면 2학기로 자동 산출하여 현재의 학기에 맞는 성적을 열람하도록 하였으며, 이는 아래와 같이 구현하였다.

```

<%
    Dim Serial_No, Current_Year, Current_Hakki
    Set DBRect_sung = Server.CreateObject
        ("ADODB.Recordset")
    sql_sung = "Select * From Sungjuk Where hakbun
        ='" & UserID & "'"
    DBRect_sung.Open sql_sung, DBConn
    DBRect_sung.MoveFirst
    Serial_No = 0
    Current_Year = Year(now)
    if Month(now) (<= 2 And Month(now) >= 8) then
        Current_Hakki = 2
    else
        Current_Hakki = 1
    end if
    do while Not DBRect_sung.eof
        Serial_No = Serial_No + 1
        Set DBRect_Kwamok = Server.CreateObject
            ("ADODB.Recordset")
        sql_Kwamok = "Select * From Kwamokmyung
            Where haksunbunho ='" & DBRect_sung
            ("haksunbunho") & "'"
        DBRect_Kwamok.Open sql_Kwamok, DBConn
        if Current_Year = DBRect_sung("year") And
            Current_Hakki = DBRect_sung("hakki") then
        %>
        **** 검색된 성적 내용 표시****

    <% end if
    DBRect_sung.MoveNext
    loop

    DBRect_Kwamok.Close
    DBRect_sung.Close
    DBRect.Close
    DBConn.Close %>

```

또한, <그림 10>은 이에 따라 구현된 해당 학기 성적 조회

를 나타내며, <그림 11>은 전체 학년/학기에 관한 성적조회 결과를 나타낸다.

학과 : 컴퓨터교육과		전공 :					
상업 : 컴퓨터		학번 : 200014818					
[금학기성적조회]							
No	학수번호	분반	교과목명	학 기	점수	평점	등급
1	CK1105A	5	컴퓨터영어	2	97		A+
2	AG1201A	1	정보통신기반	2	95		A+

<그림 10> 성적조회

학과 : 컴퓨터교육과		전공 :					
상업 : 컴퓨터		학번 : 200014818					
[전학년 성적조회]							
년도	학수번호	분반	교과목명	학 기	점수	평점	등급
2000년	CK1105A	5	한국사 0188	2	95		A+
	CK1105A	5	컴퓨터입문 및 실습1	2	90		A
	AG1103A	1	한미어(미인간의 이해)	3	94		A
	AG1205A	1	역사습자 1	2	95		A+
2001년	AG1105A	1	컴퓨터 프로그래밍 1	2	94		A
	CK1103A	5	영어와 문학	2	90		A
	CK1101A	4	상형문자	2	95		A+
2002년			교과목명	학 기	점수	평점	등급

<그림 11> 전체 성적조회

수강신청에 있어서는 전공과목인 경우 현재의 재학생에서 해당 학생을 검색한 후 과목명 테이블과 개설과목 테이블을 비교하여 현재 설강되는 과목에 대하여 수강신청이 이루어지도록 하였다.

```

(%)
Dim Serial_No, Current_Year, Current_Hakki,
    Current_Hakkwa, Current_hakryun
Dim Current_time, Current_Prof, Current_Location
Dim Kaesul_Bigo
UserID = Session("ID")
if(Session("Member") = True) then
Serial_No = 0
RecCount = 0
Current_Year = Year(now)
Set DBRect_JaeHak = Server.CreateObject
    ("ADODB.Recordset")
sql_JaeHak = "Select * From JaeHak Where hakbun
    =" & UserID &""
DBRect_JaeHak.Open sql_JaeHak, DBConn
Current_Hakkwa = DBRect_JaeHak("hakkwa")
Current_hakryun = DBRect_JaeHak("hakryun")
Current_Hakki = DBRect_JaeHak("hakki")
DBRect_JaeHak.Close
Set DBRect_Kwamok = Server.CreateObject
    ("ADODB.Recordset")
    
```

```

sql_Kwamok = "Select * From Kwamokmyung
    Where hakryun =" & sql_Kwamok = sql_Kwamok
    & Current_hakryun & " and hakki =" &
    Current_Hakki
sql_Kwamok = sql_Kwamok & " and hakkwa=" &
    Current_Hakkwa &"" Order by kwamokgubun
    DESC"
DBRect_Kwamok.Open sql_Kwamok, DBConn
DBRect_Kwamok.MoveFirst
Set DBRect_Kaesul = Server.CreateObject
    ("ADODB.Recordset")
sql_Kaesul = "Select * From KaesulKwaMok Where
    year="
sql_Kaesul = sql_Kaesul & Current_Year
    & " and hakki =" &
    Current_Hakki
DBRect_Kaesul.Open sql_Kaesul, DBConn
Kaesul_Bigo = 0
if Not DBRect_Kwamok.eof then
do while Not DBRect_Kwamok.eof
    Serial_No = Serial_No + 1
    DBRect_Kaesul.MoveFirst
do while Not DBRect_Kaesul.eof
    ifDBRect_Kwamok("haksunbunho") =
        DBRect_Kaesul("haksunbunho") and
        DBRect_Kwamok("bunban") =
        DBRect_Kaesul("bunban") then
Kaesul_Bigo = 1
        Exit Do
    end if
    DBRect_Kaesul.MoveNext
loop
if Kaesul_Bigo = 1 then
    Current_time = DBRect_Kaesul("time")
    Current_Prof = DBRect_Kaesul("prof")
    Current_Location = DBRect_Kaesul("location")
else
    Current_time = DBRect_Kwamok("time")
    Current_Prof = DBRect_Kwamok("prof")
    Current_Location = DBRect_Kwamok("location")
end if
    
```

**** 수강신청 처리된 내용 표시****

```

(%)
Kaesul_Bigo = 0
DBRect_Kwamok.MoveNext
loop
end if
(%)
    
```

수강관련 데이터베이스 정보를 XML양식으로 나타낸 것은 다음 <그림 12>과 같으며, 각 엘리먼트의 속성 및 구성은 다음 <그림 13>과 같다.

```

(%)
fname = rs("fname")
.....
fspec = rs("fg")
Dim strData
strData = "<년도" & taglt("학년도", fname,"")
.....
    
```

```

strData = strData & "<년도>"
function tagIt(strTag, strData, strAttr)
if strData (<) "" then
tagIt = "<" & strTag & "" & strAttr
& ">" & strData & "</" & strTag & ">"
else
tagIt = "<" & strTag & "" & strAttr & ">"
end if
end function
    
```

<그림 12> 데이터베이스 정보의 XML 출력

Element	종류	구성요소
학생	Element Attlist	<학생> 학번, 이름
교과목	Element Attlist	<교과목> 과목구분, 학수번호, 분 반, 과목명, 학점
설강	Element	<설강> <연도>, <학기>, <요일>, <시간>, <교수명>, <강의실>

<그림 13> 각 엘리먼트의 속성과 구성

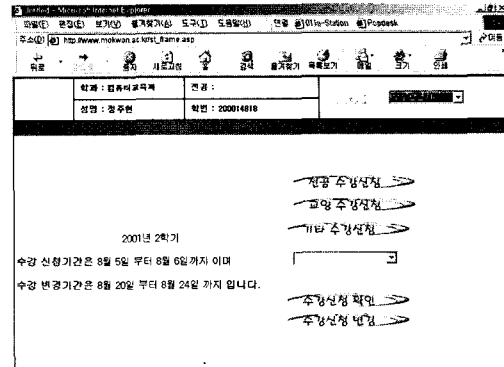
이에 따라, 실제 문서부를 작성하면 다음 <그림 14>와 같다.

```

<학생 학번="9942052" 이름="김희중">
<교과목 과목구분="전공선택"
학수번호=AA9200 분반="1"
과목명="데이터베이스" 학점="3">
<설강>
<연도>2000</연도>
<학기>1</학기>
<요일>수</요일>
<시간>1, 2</시간>
<요일>금</요일>
<시간>2</시간>
<교수명>윤호균</교수명>
<강의실>실습실301</강의실>
</설강>
    
```

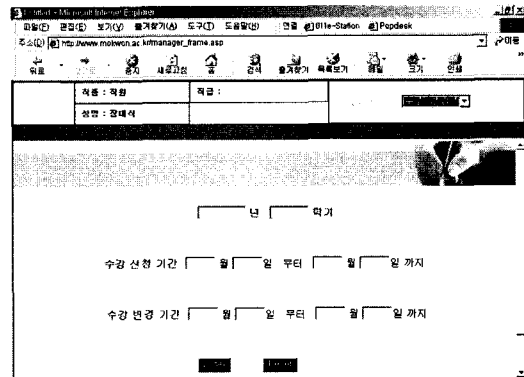
<그림 14> 수강관련 실제 문서부

<그림 15>은 학생모드로 로그인시의 메뉴를 나타내는 것으로, 해당 수강신청기간이 만료되면 모든 버튼이 자동으로 비활성화된다.



<그림 15> 수강신청

또한, <그림 16>은 관리자 모드로 로그인 후의 수강기간 입력화면을 나타낸다.



<그림 16> 수강기간

그리고, 수강 및 성적조회에 관한 본인만의 작업을 위하여 비밀번호를 스스로 관리할 수 있도록 하였다. 즉, 로그인 후 자신의 비밀번호 변경메뉴에서 이를 변경할 수 있도록 한 것이다.

(%)

UserID = Session("ID")

```

if(Session("Member") = True) then
Set DBConn = Server.CreateObject
("ADODB.Connection")
    
```

```

DBConn.Open "2001"
Set DBRect = Server.CreateObject
("ADODB.Recordset")
sql = "Select * From JaeHak Where hakbun =" &
UserID & ""
DBRect.Open sql, DBConn
    
```

```

End if

if Not DBRect_eof then
if Request("UserPassword") = DBRect
    ("Password") then
SQL = "Update JaeHak set Password =" &
    Request("NewPassword") & " Where hakbun
    =" & UserID & ""
DBConn.Execute SQL
Response.Write "<b>비밀번호</b> <b>변
    경이 완료되었습니다.</b> </p>"
else
Response.Write "<b>비밀번호</b> <b>가
    맞지 않습니다. 다시 확인하여 주십시
    요.<b> </b> </p>"
end if
else
Response.Write "<b>등록</b> <b>을 먼저
    하여야 합니다.</b> </p>"
end if
%)

```

4. 결론

현재의 많은 학교에서는 학사업무 도입에서 가장 먼저 고려하는 부분이 학사관리 시스템이며, 이를 웹에서 운용 가능하도록 요구하고 있다.

따라서, 본 논문은 인터넷을 이용하여 일선 학교에서 운영할 수 있는 학사관리 시스템 중에서 수강관련과 성적관련 처리를 하는 시스템을 구현 및 개발하였다. 이는, 로그인에서의 사용자 인증은 데이터베이스내의 재학생테이블에서 ID와 암호를 검색하도록 하였다. 특히, 성적조회는 급변학기 뿐만 아니라 전체학기의 성적조회가 가능하도록 하였으며, 수강신청은 신청에서부터 변경, 확인에 이르기까지 웹을 통하여 할 수 있도록 하였다. 또한, 관리자모드에서 수강기간을 입력 및 변경할 경우, 해당기간에 따라 학생모드의 수강메뉴를 활성화 또는 비 활성화하도록 하였다. 따라서, 관리자는 비 수강기간에 학생들의 수강입력 및 변경작업으로 인하여 중복 또는 임의변경 할 수 없도록 주기적으로 수강기간에 따라 관리하여야 하는 불편함을 제거하였다.

또한, XML을 이용함으로써 보다 효율적인 수강 데이터 관리를 할 수 있도록 하였으며, 수강 관련 데이터의 공통된 양식을 제공함으로써 각 프로세스가 이를 사용하도록 하였다. 시스템 분석 및

설계부분에서는 UML을 이용함으로써 시스템 구축의 효율성을 높였다.

그러나, 향후 본 시스템의 효율적인 사용을 위하여는, 전

체 학생들의 많은 데이터를 MS- ACCESS에 의존한다는 것은 비효율적이므로 보다 큰 데이터베이스 엔진을 사용하여야 할 것이며, 관리자모드의 다양한 메뉴가 추가적으로 필요함으로 이를 보완한다면 완전한 학사관리 서비스가 가능할 것이다.

참고문헌

- [1] 최유순, 김변곤, 김정옥, 한성국, 박종구, "일반 텍스트 문서를 XML로 변환하기 위한 T2XG 시스템 설계 및 구현", 컴퓨터산업교육기술학회논문지 VOL. 03 NO.03, 2002, 3.
- [2] 이충기, "웹기반의 자바 프로그래밍 강의 교안 개발", 공학교육연구 논문지 제5권 제1호, 2002.
- [3] R, Atkinson, "Preparing an Infrastructure for Virtual Campus Operations", "http://www,curtin edu.au, 1997.
- [4] 문진용, 구용환, "인터넷상에서 PHP를 이용한 학사관리시스템의 설계 및 구현", 한국정보처리학회 논문지 제7권 제10호, 2000, 10.
- [5] 황대준, "가상대학의 현황과 발전방향", 정보과학회지 제16권 제10호, 1998.
- [6] 문진용, 구용환외 3인, "원격교육을 위한 확장 가능한 VOD서버에 관한 연구" 한국정보과학회 봄 학술대회 제26권 제1호, 1999.
- [7] M, A, Sepulveda, "MySQL data Base Server", Linux Focus, 1998.
- [8] Alex Hornor, "Professional ASP Techniques for Webmaster", WORX, 1999.
- [9] J. Rowe, "Building Internet Database Servers with CGI", New Riders Publishing, 1996.
- [10] 박진호, 정향녕, 박진석, 최 성, "ASP System 구축을 위한 액티브 서버 환경 연구", 정보처리학회 춘계학술대회 제8권 제1호, 2001.
- [11] 강제구, "B-to-B 전자상거래 시스템을 위한 XML/EDI 구축방안에 관한 연구", 세종대, 1999.
- [12] 신행자, 박경환, "웹 기반 교육 시스템을 위한 XML문서지원 기법에 관한 연구", 한국멀티미디어학회 춘계 학술발표 논문집 제1권 2호, 1999.

- [13] 유진이, 하얀, 김용성, “메타 데이터를 이용한 학교문
서 통합관리 시스템의 설계”, 한국정보처리학회 추계
학술논문집 제6권 2호, 1999.



윤호군

1961년~1968년 한양대학교 산업공학과 공학사
 1980년~1982년 한양대학교 전자계산학과 공학석사
 1986년~1992년 동국대학교 전산통계학과 이학박사
 1984년~1995년 목원대학교 컴퓨터공학과 교수
 1984년~1987년 목원대학교 전자계산소장
 1994년~1996년 목원대학교 이공대학장
 1996년~1997년 목원대학교 학생처장
 1992년~현재 한국조사통계연구소 이사
 1997년~현재 대전광역시 소프트웨어 자문위원
 1997년~1999년 한국 컴퓨터 교육학회 부회장
 1999년~현재 한국 컴퓨터 교육학회 이사
 1996년~현재 목원대학교 컴퓨터교육과 교수
 관심분야 : 데이터베이스, XML, 데이터마이닝, 객체-관계모델링,
 Dataware House, Data Structure, Data Search Algorithm, WBI, Web
 Courseware, ICT 활용기법.