Hacettepe University
Department of Computer Engineering

BiL236 Programming Lab.
Experiment 2

Subject : Web Programming
Submission Date : 08.03.2012
Due Date : 26.03.2012 (Submission time for **deadline** is **10:00**)
Advisors : R. A. Yalın Yalıç, Dr. Ahmet Burak Can

INTRODUCTION

A web application consists of two parts: server side and client side. At client side the application is displayed on a web browser like Internet Explorer or Firefox. At client side the programming is done with XHMTL, XML, Javascript and their derivatives. At server side, a web server program [1] like IIS [2] or Apache [3] is required. A web server gets requests from clients (generally requests from web browsers of users) ands sends back the requested files to clients. Generally,a traditional web server does not provide any programming support at the server side. However, some technologies allow server side programming like ASP.NET [4], PHP [5] or JSP [6]. ASP.NET and PHP are modules of web servers; on the other hand, JSP is a technology with its own application servers like GlassFish [7].

PHP is a web programming language [8]. You do not need to compile php codes. They are run by php interpreters at run time. The web server runs PHP codes on the server side and sends outputs to client side. PHP also supports object oriented programming. In this experiment you are not supposed to use object oriented features of php. However you will use some predefined objects of PHP.

The dynamic websites require a file or database to store the data that websites use like user names, passwords, or content. **XML** is a markup language [9] to store data at files. It has several rules to define its format but its structure has not any bounds. Hence you can store any structured data. You can also query XML files like a database system to retrieve data. The most easy and efficient way of querying XML files is using **XPATH** [10].

PHP has a **DOM** library [11] to manipulate XML files and validation methods to validate an XML file against a schema file. PHP also supports XPATH queries.

AIM

This experiment is mainly aimed on developing a dynamic website with PHP, creating, managing XML files and querying XML data with XPATH.

DEVELOPMENT ENVIRONMENT

You can develop your experiment on DEV machine [12]. Apache server is installed on DEV with PHP version 5.3.8. You can use this machine for development. You should create a **public_html** folder at your home folder, if it does not exist. Then you can access your website with URL **http://web.cs.hacettepe.edu.tr/~<username>**. You should create a directory named **bil236exp2** and put your PHP files under this folder. You should create XML files in the folder **data**, which is under the folder **bil236exp2**. You cannot change main structure of necessary folders/files and their names.

For security, the folders **public_html** and **bil236exp2** should have read and execution permissions for Apache Web Server. The folder **data** should have read, execution and write permissions for Apache. You can use these commands to set these permissions:

```
mkdir -p ~/public_html/bil236exp2/news
chmod 700 ~/public_html
setfacl -m u:apache:rx ~/public_html
setfacl -m g:apache:rx ~/public_html
setfacl -m u:apache:rwx ~/public_html/bil236exp2/data
setfacl -m g:apache:rwx ~/public_html/bil236exp2/data
setfacl -m u:apache:rwx ~/public_html/bil236exp2/data/*.xml
setfacl -m g:apache:rwx ~/public_html/bil236exp2/data/*.xml
```

You can also work locally while developing your website. EasyPHP [13] and XAMPP [14] are some examples of package software for Windows platform which include Apache web server and PHP together.

EXPERIMENT

In this experiment you will develop an instructor/course list web site. You should implement web pages for **insert, list, update** and **delete** operations for both instructor and course.

- An instructor has two attributes: unique instructor-id, instructor-name.
- A course has four attributes: unique course-id, course-code, course-name and instructor-id.
- Instructor and course id's are unique, given by your program. They can't be updated by user. A new added instructor/course id can't be assigned to previously deleted instructor/course id
- A course can be assigned to only one instructor.
- An attribute (field) can't be empty. So you must check them at the server side after user enters the form fields.
- Instructor data file name is *instructors.xml*
- Course data file name is *courses.xml*. Both data files must be inside *data* folder
    - http://web.cs.hacettepe.edu.tr/~yalinyalic/bil236/data/instructors.xml
    - http://web.cs.hacettepe.edu.tr/~yalinyalic/bil236/data/courses.xml
- You should implement data consistency. This means if an instructor is deleted, all courses assigned that instructor should also be deleted.
- Index page of website is a search page. User can search keywords over course name or instructor. Matched instructors/courses must be returned which contains query string.
- You are allowed to change appearance of web pages but they should have the same functionality.
- All pages have links to search, add/list instructors, add/list courses pages at the top.

An example web site is given at advisor's web: http://web.cs.hacettepe.edu.tr/~yalinyalic/bil236

CONSTRAINTS

PHP page names and form field names must be same as stated below. Otherwise your web program can't be evaluated:

- *index.php* : page for searching instructor name and course names with two different forms
    - form field names: *searchinstructor*, *searchcourse*
- *addinstructor.php* : use for adding instructor
    - form field name: *instructor-name*
- *addcourse.php* : use for adding course

    - form field names: *course-code, course-name, instructor-id*
- *listinstructor.php* : page for listing instructors and links to delete and edit pages of selected instructor
- *listcourse.php* : page for listing courses with codes, names and assigned instructors' names. It has links to delete and edit pages of selected course.
- *editinstructor.php* : page for changing name of instructor
    - form field name: *instructor-name*
    - get parameter by *iid*
- *saveinstructor.php* : page for updating instructor
    - get parameter by *iid*
- *editcourse.php* : page for changing course code, name and instructor
    - form field names: *course-code, course-name, instructor-id*
    - get parameter by *cid*
- *savecourse.php* : page for updating course
    - get parameter by *cid*
- *delinstructor.php* : page for deleting instructor and all courses belongs to him/her
    - get parameter by *iid*
- *delcourse.php* : page for deleting instructor and all courses belongs to him/her
    - get parameter by *cid*
- *searchi.php* : search results of found instructors with courses belongs to him/her
- *searchc.php* : search results of found courses with all attributes.

IMPORTANT NOTES

- **Submission format:**

```
<studentno>.zip
        + report
          | - report.pdf
        + source
          | - <all php files>
          | + data
            | - instructors.xml
            | - courses.xml
```

- Your **REPORT** must be written according to the instructions in the file "ftp://ftp.cs.hacettepe.edu.tr/pub/dersler/genel/FormatForLabReports.doc"
- You will use online submission system to submit your experiments. (https://submit.cs.hacettepe.edu.tr/) Submission time for **deadline is 10:00**. No other submission methods (such as e-mail) will be accepted.
- The assignment must be original, **INDIVIDUAL** work. Duplicate or very similar assignments are both going to be **PUNISHED**. General discussion of the problem is allowed, but **DO NOT SHARE** answers, algorithms or source codes.
- You can ask your questions through course's news group and you are supposed to be aware of everything discussed in the newsgroup: **news://news.cs.hacettepe.edu.tr/dersler.236**
- Respect the office hours of your advisor (on Monday between 9:00-11:00 and on Tuesday 9:00-11:00).

References

[1] Web Server, http://en.wikipedia.org/wiki/Web_server
[2] Internet Information Server, http://www.iis.net/
[3] Apache Web Server, http://httpd.apache.org/
[4] ASP.NET, http://en.wikipedia.org/wiki/ASP.NET
[5] PHP: Hypertext Preprocessor, http://php.net/index.php
[6] JavaServer Pages Technology, http://www.oracle.com/technetwork/java/javaee/jsp/index.html
[7] GlassFish, http://en.wikipedia.org/wiki/GlassFish
[8] PHP Documentation, http://www.php.net/manual/en/index.php
[9] XML, http://en.wikipedia.org/wiki/XML
[10] XPATH, http://en.wikipedia.org/wiki/XPath
[11] PHP DOM, http://php.net/manual/en/book.dom.php
[12] The DEV Machine, ssh://dev.cs.hacettepe.edu.tr
[13] EasyPHP, http://www.easyphp.org/
[14] XAMPP, http://sourceforge.net/projects/xampp/