Assignment 2

Weight: 20% of your final grade

Due: after completing Units 4 and 5

Instructions and General Requirements

This assignment has two parts. In the first part you are required to complete a small Web application development project. For the second part, you will design and implement a web application for e-learning using the web technologies you have learned so far, including HTML5, CSS3, Ajax, XML, databases for the Web, and server-side scripting with PHP for the Web. The general requirements for the assignment are as follows.

- Use a plain text editor such as Notepad++ or TextEdit to create all your web documents pages for this assignment. Do not use MS Word or similar authoring tools to create or edit your web documents because you cannot see the actual codes.
- Plan carefully what to put on each web page and how to lay out everything you want to present. Pay attention also to the visual design of your web pages. Clean, simple designs should work well.
- Each web application or web page you write for each assignment project is expected to be a professional web application or web page. If you want to show course related information or links on the page, they should be placed in such a way that they will not interfere with the content, functionality, or the overall look and feel of your web application or web page.
- You must do your best to write the best web page, or develop the best web application for each and every project in the assignment. When you are asked to develop web applications, a single web page or a few inter-linked web pages may not be sufficient. You should create a web-based system. It should have a welcome page, a banner with the name or title of the application, menus and buttons for navigation, and required functional modules. You must show your effort to make each of your web applications attractive, useful and user-friendly. Your applications may not have a perfect professional look and feel, but you must show your effort to achieve that. Your time and effort in developing the best web applications for the assignment will contribute greatly to your future success as a web developer.
- Use the same external style sheet for all your pages to ensure your assignment site has a consistent look and feel.
- For this assignment, you don't have to validate your web documents, but you must test all your web applications for the assignment on your personal web server to ensure that they work. You also must make your web server accessible for your

tutor to evaluate your work. In most cases, you would only need to open a port on your router provided by your ISP and setup port forwarding within the router. If your ISP doesn't allow you to do port forwarding on the router, you may need to find a hosting service on the internet. To protect your privacy, do not put your student ID on an unprotected web site. When using a hosting service on the internet, you must remove all the documents related to the assignment from the server after your tutor has assessed your work.

- Please refer to Assignment 1 for files organization and composition of main page tma2.htm (**10 marks**). As for Assignment 1, the main page should include the following information:

  – assignment number
  – course number and name
  – your name and student ID number
  – date you began working on the assignment
  – date you completed the assignment
  – estimated hours you spent on the assignment

Following the cover page will be the required documentation on your work for each part of the assignment, which should include:

– the original description and requirements of the project
– your interpretation of the assignment project and requirements
– your analysis and design of the web application
– documentation of your implementation
– a guide for users (your tutor, who will mark the assignment) to set up, to run, and to use the web application
– a hyperlink to the web applications you design for the project
– anything else you want to say to your tutor

Please keep in mind that all the web applications and related files you develop for the assignment must be accessible from the *tma2.htm* page, either directly or indirectly.

Please also keep in mind that each web application must use the same external style sheet to ensure the web application has a consistent look and feel.

Submit your assignment as a zip file called TMA2.zip.

☞ **Important**: All work submitted must be original, and no codes or packages from a third party should be used unless it is explicitly allowed in the assignment instructions. See the Athabasca University policy on intellectual honesty.

Part 1

(20 marks)

For this project, you will be using MySQL and PHP to develop a web application that provides an online bookmarking service to users on the Internet. The requirements are as follows:

- The web application should have a good thoughtful interface, with menus and navigation buttons as needed.
- It should have a name or logo shown across all the pages.
- It should begin with a welcome or greeting message and a list of ten most popular websites that people have bookmarked.
- Once signed in, a list of bookmarks should be displayed, and the user can browse any web site in the list in a new browser tab or window by clicking the URL.
- The user will also be able to add new websites to the list and edit and delete any of the existing ones in the list;
- When adding or editing, user input needs to be validated using JavaScript, to make sure the URLs are correct and active.

Save Part 1 files in the directory TMA2/part1.

Part 2

(70 marks)

In this part of the assignment, you are required to develop a small-scale online learning management system that can be used to deliver online courses to learners.

To that end, you will have to think about what these online courses are, how they can they be developed and how they should be stored on the web server, how they can be retrieved from the server, how they should be delivered to a web browser, and then rendered/presented properly on the web browser.

The development of online courses is the collaboration of efforts of subject matter experts (SMEs) and experts in computing and web technology. It is very common that those SMEs do not know much about computing, and don't know how to use HTML and other web technologies needed to present an attractive course, but they should be quite comfortable with languages and terms used in education. So, the first technical step you need to take is to design a SME-friendly language for marking up educational materials, EML in our term (just another XML like you created for marking up your resume), for the SMEs to use. The EML you design may be a comprehensive one for marking up the contents for an entire course, or several languages in small scale for different components of a course. For example, you may have an EML for a lesson/lecture delivered in just a teaching/learning session, an EML for marking up quizzes, and an EML for marking up assignments, etc. You may look at some existing languages by searching for educational markup language in Google or other search engine.

After SMEs (for this assignment, you will be the SME) have written the course contents in your EML(s), the documents must be stored on the server before they can be delivered to the learners on the web. You may think that you can save each of the documents written in your EML as a file, like the resume file you wrote for assignment 1, but this is not practical. Files may work when there is just one or only a few learning documents, but when there are many, as happens in practical situations, the documents become unmanageable. That's why we need a database.

The next technical step is to design the database table or tables. Over the years, I have seen different designs from students in the course. The most simple and straightforward one is to have a single table. In the table each row contains information for a lesson or quiz, including the actual content written in your EML. To make the learning contents retrievable and manageable, you will need fields to identify the course and the unit the lesson or quiz belongs to. You may also keep the details of the courses and units in the same table for the purpose of this assignment project, though you wouldn't do that in practice because it is really inefficient; in a real world situation is to have separate tables to store information about courses and units.

To deliver the course contents (lessons and quizzes in our context), you will need to navigate through the database to find the right lesson or quiz that the learner has requested, and them retrieve the actual content in EML. For the content to be properly rendered or presented on learner's browser, you need to translate the content in your EML to HTML. We call this process parsing. Instead of using XSLT as you did for the first assignment, parsing EML to HTML needs to be done with PHP. So, the last important technical step for this project is to write a parser in PHP.

When doing this project, or any project for the assignments, you must take a system approach. That's why students are expected to have completed comp361 or a similar course on system analysis and design before taking this course.

The distribution of marks is as follows:

- **System analysis and design –** 10 marks.
  Your analysis and design need to be presented in the assignment report. The details should include, but not be limited to, analysis of requirements, data flow, functional modules of learning management, the relationships of these modules, and system layout within browser's window.
- **Educational Markup language -** 10 marks.
  Your educational markup language should be presented in the assignment report by showing all the tags and explanation of their intended purpose with examples.
- **Database design** - 10 marks.
  Database is used to store information about courses, units, lessons, quizzes, learners, and other learning object such as images, audio and video files, as well as actual learning contents written in your educational markup language. When designing database tables, you have to think about how that information and those actual data will be kept on the server.

- **Parser -** 10 marks.
  On the server side, the parser written in PHP parses the learning contents written in your EML taken from database into HTML documents that can be rendered by web browsers. Don't be afraid of writing a parser for this purpose! It can be as simple as using some PHP regular expression functions to do pattern matching and string replacement.
- **Learning Contents** - 10 marks.
  Since you have written three tutorials or lessons for part 2 of your first assignment, you may take the contents from there, and rewrite them in your EML. You must have enough contents in your system, to make it a real web-based system.
- **Quizzes -** 10 marks
  there should be an online quiz for each teaching lesson.
- **Implementation -** 10 marks.
  The system should be implemented and deployed on a web server, and accessible for your tutor to evaluate. The implementation includes the overall look-and-feel of the learning management system, essential components of such a web-based system, and the functionalities of online learning management.

Save all files related to Part 2 in the directory TMA2/part2.

You must test your web applications developed for the assignment on your own server before you submit them and make the websites hosting the assignments and applications accessible for your tutor to evaluate. For your personal protection you should not put your student ID in any of the documents if hosted on an unprotected web site. After your tutor has finished the evaluation, you must remove all the assignment-related documents from the web site.

You must submit all the files under TMA2 in a zip package through Moodle.