Navigation to my home hosted website:

Navigate to http://68.149.242.146/TMA2/tma2.htm, my computer that is running this is a bit older so it may be slowish. COMP466 Course Password: 123 Admin Password: 123. Bookmark Service a username to load if you dont want to make an account is U:john P:12345.

Schema Overview All Tables:

Part 1:

Table Login:

Username(PK)	Password

Table Bookmarks:

Username(FK, PK)	Bookmark(PK)

Part 2:

Table course_login

courseName(PK)	coursePassword	adminPassword

Table courses:

ID(PK, auto increment)	CourseName(FK)	CourseUnit	Type	xmlMarkup	ContentPath

<u>PART 1:</u>

the original description and requirements of the project:

This part of the assignment involved me getting introduced to PHP and MYSQL by designing a simple bookmark implementation on a webpage. This bookmark implementation should greet user with a list of top 10 most used web pages for bookmarks. Once logged in the user must be able to see his/her bookmarks. The link should open in new tab when the user clicks on a link.

your analysis and design of the web application:

This design was not very complex as we just needed a simple log_in table with a username and a password to track the users. We then used the username as a foreign key in the Bookmarks table which tracked the individual bookmarks. This made it super simple to query all of the user.

documentation of your implementation:

For some aspects of this I found using form with PHP post calls in order to handle a scenario. Other times I used javascript event with an ajax call that triggered PHP. This usually depended on the simplicity of the action. I used modal forms in order to minimize the number of pages for this simple task. I used the common phrase as mentioned in requirements as “Bookmarks are Us”.

a guide for users (your tutor, who will mark the assignment) to set up, to run, and to use the web application:

Please refer to accessing website section above to navigate to this website that will be hosted through a port on my computer at home. Please refer to login information if you want a to log into a user profile that made already for this part or next. This app simple allows a user to view the most popular bookmarks when they are not logged in. When they are logged in they can edit, delete add new bookmarks that will be stored in a database under their profile.

a hyperlink to the web applications you design for the project:

To get to this page from main website simply click on the Bookmark Service.

PART 2:

the original description and requirements of the project:

This part of the assignment involved designing an online course management tool that would enable a SME to create a data file (XML) and upload it with whatever images it uses. The service then must allows student to access this data as a visually friendly webpage. The xml is suppose to be stored in the database and you will have to write a parser that takes the database row and converts it into a html page that will be displayed to user. This parser is to be written in PHP. Once these components are completed package this tool as user friendly webpage that allows teachers and students to upload and view content.

your analysis and design of the web application:

I wanted to allow 3 different types of uploads from the teacher perspective, Assignments, Lessons and Quizzes. In order to minimize the number of files stored on server side the Quizzes and Assignment are not uploaded it more of just an information placeholder. The Quizzes are marked however and correct answers are displayed to user when they submit. There is a schema defined for each file upload which prevents a user from uploading an invalid xml. I converted my assignment 1 XMLs for quiz and lessons into the format I thought more appropriate for assignment 2. These reside in the Part2/Data directory. They are just there so you can see the data structure when marking they are not actually used in app. There are two data tables one for the course login information and one for the course information. Each course row represents either a Assignment, Quiz, or Lesson and it contain all the information my Php Parser needs to know in order to create the web page. When a user uploads an xml they also can upload images that are stored on the server side in a content path that is stored in the database course information row. A course has an admin password and course password. The admin password and course Password are needed to upload file to a course but the course password is only needed to view the information.

documentation of your implementation:

I tried to use as much PHP as possible for this assignment with minimal javascript. All the heavy lifting of this application is done through queries and php. Using two tables for this allowed me to not repeat login information for every single row as a course only has a single Admin Password and single Course Password. These passwords were put in place so a teacher can make available the information to student by simple giving them a password and still have the course integrity protected by having a Admin Password protecting the ability to upload.

a guide for users (your tutor, who will mark the assignment) to set up, to run, and to use the web application:

I already upload quiz 1, quiz 2, quiz 3, lesson 1, lesson 2, lesson 3, from assignment 1 under a **Comp 466 course.** Login information is posted above. Using login information you can upload and view any of the course information. We stash a Session variable that tracks the last course you downloaded and from the “Last Course Loaded” page you can open any of the material for the course you just loade.

a hyperlink to the web applications you design for the project:

To get to this page from main website simply click on the Academic Service.

Educational Markup Language Guide:

Assignment:

- Header: The header of a specific paragraph of information of the assignment.
- Paragraph: The paragraph that goes with the header.
- Grouping: A pair of header and paragraph that implements a single idea for the assignment
- Assignment: The root node for this educational item.

Quiz:

- Question Title: The question that is proposed for a specific question.
- Answer: The correct answer to the question title, matches text identically.
- Option: The list of options for this question.
- OptionList: contains all the options available for a specific question.
- MultipleChoice: A type of question that has a title an option list and a single answer inside the answer tag.
- Selection: A type of question that has question title option list and 1/many answers separated by "," inside the answer tag.
- TrueFalse: A type of question with only a question title and answer. The answer must be either "True" or "False".
- Quiz Details: Contains all the questions for the quiz, the root element.

Lesson:

- Header: The header to appear in the section of tutorial.
- IMGTitle: If an image is used this is the title of the image matched exactly so that the parser loads the correct image.

- Explanation: A paragraph explaining a specific topic.
- Item: A single item in a list item grouping.
- ListItems: A grouping with a series of bullet points.
- FigureExplanation: A grouping that has a Header, IMGTitle and Explanation. Simple Image with a long paragraph explanation.
- List: A grouping that has Header, ListItems, and an Explanation. List with explanation underneath to almost be like a conclusion.
- FigCaption: A grouping with only a short explanation and a IMGTitle.
- TutorialDetails: Contatins all grouping in the lesson, the root element.