CPCS403 – Final Project Description (Individual Project)

Submit Wednesday 15 May 2024 @ 11:59 pm

Submit via BlackBoard

Once completed, you should submit the zip file on the BlackBoard

Objective

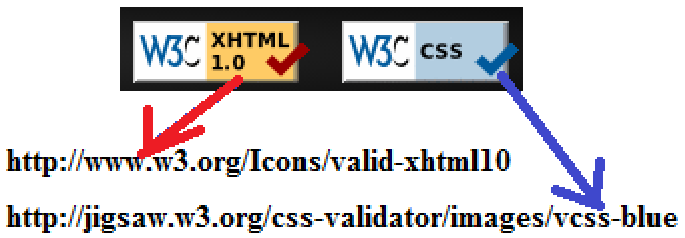
The result of the final project will be a fully developed web application containing both front- and back-end that will be available online at public domain.

1. Number of Pages

- The web application must have a minimum of **10** total pages.
- The web application must build on the idea you submitted in assignment 1.
- All pages must have proper content with appropriate title/headings.
- Your website must contain the following pages (discussed in this document):
  - Home page
  - Table (Schedule) or any table suitable for your website idea
  - Picture Gallery
  - Video
  - Feedback form or another kind of form suitable to your website idea
  - Resume
  - Contact Us
- Interactive function (get input from user and website display appropriate action)
- Database
- Add any other page(s) of your own choice with proper content according to your website idea
- Make sure that your website has good usability (the presentation of information and choices in a clear and concise way, a lack of ambiguity and the placement of important items in appropriate areas)

2. PHP

- The backend of the web application must be developed in PHP. The use of specialized/advanced PHP frameworks is not allowed.
- ALL pages must be PHP pages (saved as *.php) in order to use PHP *includes*.
- ALL pages MUST pass XHTML (Strict) and CSS validation via W3C.
- ALL pages must use at least TWO PHP includes for the header and footer.
  - **Header**: all pages should have a common “header” using PHP *includes*
    - The header should have your logo and the links of the main menu.
    - This header should be an include page within all other pages of the Web site
  - **Footer**: all pages should have a common “footer” section using PHP *includes*
    - Your footer should have BOTH w3c validation images, and both images should be “clickable”.


W3C XHTML 1.0
W3C css
http://www.w3.org/Icons/valid-xhtml10
http://jigsaw.w3.org/css-validator/images/vcss-blue

3. Directory Structure

- All files for your Web site should be organized into appropriate directories:
    - All images should be saved into an “image” folder.
    - All PHP *include* files should be saved into “includes” folder/
    - All videos should be saved into a “videos” folder/
    - All HTML/PHP files should be saved into a “pages” folder.
- You can choose the names of these folders at your own choice, just make sure your files are organized appropriately.

4. Look (HTML, layout, and CSS)

- All pages should have a unified theme and color scheme which is set in external CSS files.
- You MUST use the DIV approach to layout different components of your website (such as header, footer, content area, etc.)
- Your Web site should follow the hierarchical approach to Web site design
    - Depending on the content/purpose of your site, some of the internal pages may use a linear approach
- All layout/design should be done in CSS
    - Including using containers for layout
- You should have one CSS file for your entire site
    - Should be saved within the “global” directory
    - So if your CSS file is called main.css, it should be stored in global/main.css
    - Each page of the site should link to this CSS file

5. Table

- You should have at least one page with a table.
- This table should not be used for the layout of the page but should be used to display some tabular data, such as a schedule or some other type of content that is best laid out within a table suitable to your website idea.
- Your table should have multiple columns and rows, and it MUST use *colspan* and *rowspan*
- Printable table: in addition to the main CSS file, create a separate CSS (**print.css**) for customized print options. It should print only **table contents** (with caption) and nothing else.

6. Picture Gallery

- You should include a page containing an image gallery consisting of some thumbnails. Clicking one of thumbnails causes the corresponding image to be displayed in bigger size. Use JavaScript to implement this feature.

7. Video

- You should have a page with a Video. You can use the <video> or <object> elements. For example, you can embed a Youtube video using the object tag.

8. Feedback Form (or another Form suitable to your idea)

1. Look

- The web application should have a form. The form should be divided into at least two sections. Use *fieldsets* to divide the sections.
- Proper labels must be assigned to all form elements.
- Your form should contain AT LEAST the following elements:
    - 4 textboxes (including email and name) and depending on your website idea.
    - 2 radio/option inputs.
    - 2 checkboxes.
    - 1 dropdown selection lists.
    - 1 Textarea box for feedback.
- Apply CSS for the form elements.

2. JavaScript Validation

- Validation must be applied by using JavaScript to the form which include:
    - Input validation: the form should not submit with invalid data
    - Mandatory fields: the form should not submit with empty mandatory fields.
- Add all validation code in a separate file **validation.js**

3. Server Processing and Validation

- Store all data on the server in the database - create a designated table(s) for this form.
- Apply validation on email ID so that there should be an only single entry for an email ID.
- In case the email ID is new, store the feedback data; otherwise give an error message to user without saving it.

9. Resume

- Include PDF version of your resume. The pdf resume must be displayed within a page without the need to download it or open it in a new tab by using the <object> tag for it.

10. Database (PHP + MySQL)

- Create a MySQL database on your hosting service.
- Add at least two tables to the database (other than "form" table, as mentioned in section 7).
    - Add appropriate tables that are related to your project idea.
- There must be a relationship between these two tables (primary/foreign keys).
- Ensure that you have proper PHP pages that add data to these tables.
- Ensure that you have proper PHP pages that fetch/display data from these tables.

11. Security

- Ensure that all validations done at the client-side (via JavaScript) are also performed at the server side via PHP.
- Ensure that SQL injection attacks are handled properly.
- Ensure that Cross-Site Scripting attacks are handled properly.
- Ensure that you incorporate a secure design.

Deliverables (what to submit)

- You must submit one ZIP file containing all the files and folders of your Web Application to Blackboard.
- The project must also be uploaded to your domain so that the web application becomes live and accessible via a URL. The project (including backend) must work correctly on the host.
- Word file that explains your website idea and design. You can build on the file submitted for assignment 1.
- Submit a script for creating the various tables used in the project.

Deadline

Submit on Wednesday 15 May, 2024 @ 11:59 pm. The presentations will be on the same week.