CPSC 2030 Final Project

Your task is to create a dynamic website using PHP, a MySQL Database, CSS, and JavaScript/jQuery, that is based on a topic of your choice. Choose a topic that interests you and that you will enjoy working for the remainder of the term. Ideally, this site will be a good addition to your portfolio. Think of this as a project that you can use to showcase your web development skills to potential employers.

Dynamic PHP Website From Scratch

You will write all code from scratch and design your own database. Your application should be designed in a way that is easy to update and maintain, with the logic, presentation and data components relatively distinct from one another.

Your site should include the following:

- Dynamic content that is stored in and fetched from the database;
- At least three pages
- At least **one form**, with **validation** (client and server-side. Feel free to use a JavaScript plugin, but implement your own PHP validation);
- Some way for the user to update records in the database or add new records to the database;
- Security measures that prevent malicious activity (eg. prepared statements to prevent SQL injection attacks);
- At least one element that uses **CSS transitions**, and at least one **@keyframes animation**.
- A layout that uses **CSS Grid** and/or **Flexbox**. (No float-based layouts allowed!)
- Some **JavaScript** and/or **jQuery functionality**. This could be a keyboard-accessible menu, collapsible content, an image slider, etc.

Your website should also satisfy the following criteria:

- **Responsive, mobile-first design:** Your site should respond to any kind of viewport, from a small mobile phone to a large desktop screen.
- **Tested and free from errors:** Your HTML and CSS should validate. (This means the HTML that is output by your PHP scripts.) In your documentation, include screenshots of validation results for your CSS styles and each page on your site. You should try out your site in a few different browsers and a few different devices.
- **Accessibility features:** You should have features such as alt text on images, sufficient text/background contrast, labels on forms, screen-reader text for icon links, semantic tags such as <nav>, <header>, etc. You may also implement ARIA roles and attributes.
- **Well-designed:** It should be easy to use your page. Navigation should be easy to figure out, text should be easy to read, and one shouldn't have to hunt around for what they're looking for. Use a sensible organization that makes it easy to accomplish tasks with minimal difficulty.

One of the pages on your site should be called **Documentation**, and should include instructions for how to use your site, in addition to a list of features that you would like to contribute to your grade. If there's something you've implemented that you're proud of, mention it here, and include a link to that part of the site. A good practice would be to go through the checklist at the end of this document and make sure you address every point. Also include the screenshots of your validation results on this page. (see below.)

- A few other things:
    - Text should be attractive and easy to read:
        - Titles and headings should be easy to differentiate from paragraph text.
        - The fonts should be appropriate to the subject matter, and simple fonts should be used for paragraph text.
        - Most text (except for headings, perhaps) should be left-aligned.
        - Make sure there is adequate contrast between text and backgrounds.
    - Elements should be aligned.
    - Related elements should be grouped together.

Content:

You are **not** required to create original content for this site, although you may choose to do so anyway. Feel free to use text from Wikipedia or some other source that provides content under a Creative Commons license. You may use any images that are either in the **public domain** or licensed for use under **Creative Commons**. Of course, you are free to use your own images and writing if you choose.

Some sources of free images and video:

- https://unsplash.com/
- https://morguefile.com/photos/morguefile/ (images and video)
- https://videos.pexels.com/ (video)
- https://www.pexels.com/
- https://pixabay.com/
- http://nos.twnsnd.co/
- http://jaymantri.com/
- http://pickupimage.com/
- https://picography.co/
- http://www.lifeofpix.com/

Please note that if you use any content created by someone else, including text, images and videos, **you must provide proper attribution; failure to do so will result in a mark of 0 on the project. You must cite every source by providing a link to the exact, specific page where you got the content, even if the source does not require attribution.**

You may not use any code libraries or frameworks (other than jQuery, JavaScript form validation plugin) without permission from me. I encourage you to explore other tools, but please discuss it with me first (via the project proposal, email or during the lab).

Notes on Sources and Attribution:

- **Google Images is *not* a valid source.**
- **Rather than simply providing a link to a stock image site, provide a link to the page for the *specific* image you used. For example, providing a link to "pexels.com" would *not* be sufficient.**
- **If an image does not have any copyright or license information, you must assume that it is *not* free to use.**

Failure to follow these practices will result in a mark of 0 on the project and will be reported to the college as academic misconduct. Ask me if you have any doubts; I'd be happy to help!

You should have a page called **Sources** that lists all sources of images, text, code, and other content used by you that was created by someone else.

Bonus:

I may award a small bonus for functionality that goes beyond what we've learned in the class, or represents a high difficulty level. **If you have any ideas for bonus functionality, feel free to ask me**.

Check List

- [1 mark] Organization/Architecture
- [1 mark] Dynamic content that is stored in and fetched from the database;
- [2 marks] At least one form, with validation;
- [1 mark] Some way for the user to update the database or add new records to the database;
- [2 mark] Documentation and Sources pages
- [1 mark] Animations
- [4 mark] Grid/Flexbox Layout
- [4 mark] JavaScript/jQuery functionality
- [1 marks] Accessible and responsive
- [2 marks] Well-tested and free from errors (with screenshots of validation.)
- [1 marks] Well-designed

- Bonus
  - [+1-2 marks] Approved bonus functionality

Total: 20 marks

Hand In:

- Export your database using PhpMyAdmin and include it in your project files.
- Remove any files that are not being used in the final version of your site, and make sure your code has been cleaned up/formatted/commented.
- Zip all necessary files, rename the folder to firstname-lastname, and hand in the archive to D2L.
- If there are any special instructions, please include them in the comments.

*Please note that I may deduct up to -10% for poorly organized/commented/formatted code and for improper hand in.

Acknowledgement/References:

- Jordan Millier, Langara College