Dynamic Webpage: Image Uploader

Credit: COURSE CSE3210: SERVER-SIDE SCRIPTING 1

From the Alberta [program of studies](http://www.learnalberta.ca/ProgramOfStudy.aspx?lang=en&ProgramId=74838#757196):

3. use appropriate techniques to design applications for use in a server-side scripting environment

4. write and debug the scripts required to implement a server-side application

Currently, the [project](https://github.com/MrArtym/Image-Hosting-Database-Website) can upload and view images.

Possible Additions:

* Additional attributes to uploaded files saved in the databases
* Different viewers for image files
* Different queries to return different attributes
* A gallery to view all images sequentially.
* A search bar that can search for titles or captions of images in the database.
* Have upload page redirect to same page

There are many possible interpretations of these requirements, and many more enhancements are possible. Be creative!

Not a requirement, but if you do want to use additional tables, present them to your teachers to receive formative feedback.

## Other Resources

Adobe Color Wheel:

<https://color.adobe.com/create/color-wheel/>

<https://www.w3schools.com/sql/>

***Dynamic Webpage Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

## Possible Assessment Criteria

### Basic Feature Requirements (60%)

|  |  |
| --- | --- |
| **Item** | **This/these items are present**  **(30 points for each)** |
| The website includes a feature to view all the images in the database. |  |
| The website includes a feature to view all captions in the database. |  |
|  | **Total**  **/60** |

### Intermediate Feature Requirements (10%) - Add features here!

Choose 2 Features to implement

|  |  |
| --- | --- |
| **Item** | **This/these items are present**  **(5 marks each)** |
| The website includes a feature to create a new image (with custom attributes) through the use of <form> and <input> elements on the page |  |
| The website includes a feature to remove an image from the database. |  |
| The website includes a search feature to search for an image by a field that is NOT their database ID |  |
|  | **Total**  **/10** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ITEM** | **Outstanding**  **3** | **Proficient**  **2** | **Requires improvement**  **1** | **Insufficient evidence**  **0** |
| **Do the HTML pages follow proper style conventions?**  **(x5; 15)**   * HTML tags are all well-formed (all tags are properly closed and encapsulated) * HTML is easy to read (proper indentation) * Comments are included outlining elements and attributes * Document type is included at top of document * Tag elements are all lower case * Empty HTML elements are properly closed * Attributes are contained in quotations * Alt text is provided for all images * Consistency is maintained throughout document | Checklist has been  consistently followed | Checklist has been followed but not  consistently | Checklist has been  followed inconsistently  with several  omissions/  errors |  |
| **Is CSS implemented follows proper style conventions?**  **(x3; 15%)**   * Style is defined in separate, linked style.css document * Semicolons included after every declaration * Indentation of all block content * Consistency is maintained throughout document * ID and class names are short, but descriptive | Checklist has been  consistently followed | Checklist has been followed but not  consistently | Checklist has been  followed inconsistently  with several  omissions/  errors |  |
| **Comments:** | | | | **Overall**  **Total**  **/30** |

### Overall Grade

|  |  |
| --- | --- |
| **Basic Requirements (60%)** | **/60** |
| **Intermediate Requirements**  **(10%)** | **/10** |
| **HTML and CSS Conventions**  **(40%)** | **/30** |
| **Total** | **/100** |
| **Comments** |  |