Homework 8 – An image sharing site (Part II: Photo uploader)

Goal

Design and implement the image uploader functionality. This will be accomplished in two steps:

(1) a "basic uploader" form, which should **post pictures (and descriptions) to the wall** designed in HW7.

(2) an "improved uploader", which should include the **ability to apply pre-existing image filters to an image before posting it to the wall.**

Recommended Procedure:

Milestone 3: basic image uploader (starter code: 'BasicImageUploader')

1. **Create a table for storing photo information**. It should contain 5 fields per record, as follows: USER_USERNAME, STATUS_TEXT, STATUS_TITLE, IMAGE_NAME, TIME_STAMP

```
database.sql

 1   -- Uncomment the following line if recreating the table
 2   -- DROP TABLE WALL;
 3   CREATE TABLE WALL(
 4       USER_USERNAME VARCHAR(15) NOT NULL,
 5       STATUS_TEXT VARCHAR(140) NOT NULL,
 6       STATUS_TITLE VARCHAR(140) NOT NULL,
 7       IMAGE_NAME VARCHAR(50) NOT NULL,
 8       TIME_STAMP VARCHAR(50) NOT NULL,
 9       PRIMARY KEY (TIME_STAMP)
10   );
```

2. Design a **form** for image uploading. It should contain an input for the uploader's username, the image title and caption, and a button (for file selection before upload).
3. Design the **script** that processes the form action, i.e., uploads the picture, saves it with a specific name, stores associated information (username, title, and text) in the WALL table, reads the contents of the WALL table, and displays them in reverse chronological order.
4. (OPTIONAL) See example of desired functionality at:
   http://lamp.cse.fau.edu/~gbudden2014/project5/index.php (form) and
   http://lamp.cse.fau.edu/~gbudden2014/project5/postcard.php (wall)
5. *Integrate the latest code with previous milestones, i.e., ensure that the form and the wall are only accessible to authenticated users during the course of a session.*

Milestone 4: improved image uploader

6. Modify the form to allow applying **image filters** (implemented using CSS) to an image before uploading.
7. **Change the WALL table** to include a field that will store information about applied/selected filter.
8. **Change the PHP scripts** accordingly.

9. See example of desired functionality[1] at:
   http://lamp.cse.fau.edu/~rhubelbank2014/project5/index.php (form) and
   http://lamp.cse.fau.edu/~rhubelbank2014/project5/postcard.php (wall)
10. ***Integrate the latest code with previous milestone(s).***
11. **Test your app** after every significant change / addition.
12. **IMPORTANT: NAMING CONVENTIONS**
    o   Name your starting page (with login/signup screen) *index.php*
    o   Name the upload form (which should only be accessible to registered users) *form.php*
    o   Name your wall file (which should only be accessible to registered users) *wall.php*
    o   Make all files available at your *hw8* folder.
    o   In summary, my grader and I should be able to go to
        http://lamp.cse.fau.edu/~username/hw8/index.php to signup/login, then to
        http://lamp.cse.fau.edu/~username/hw8/form.php to test the upload functionality,
        and to http://lamp.cse.fau.edu/~username/hw8/wall.php to see the results (and other
        people's photos, too).
13. Once you've reached a point where your app is complete and fully functional in the browser of
    your choice (Chrome, Firefox, Opera, or Safari), **prepare the final package** (single zip, all that is
    needed, and nothing else).
14. **Submit the final package** via Blackboard.

Minimum requirements:

• Your app **must** be your own work. If you use a site, textbook example or any other source as
  "inspiration" along the way, please make a note of it in your report.
• Your app should demonstrate a basic understanding of PHP, MySQL, HTML5, CSS, and
  JavaScript.
• Your app must be fully functional.

Deliverables

• **A single zip file** containing **all** files (.php, .sql, .css, .jpg, .js, etc.) associated with your project
  and your report (see below).
• A **report** (2-4 pages) describing the entire process, including: the learning curve (for PHP and
  MySQL, especially), the design process, coding troubles, functionality highlights, future
  improvements, etc.

  Please name your file using your FAU / Blackboard username as a filename, e.g., jsmith85.zip.

---

[1] This example also includes sliders for brightness and contrast adjustments. These are **not** required.

- **Grading rubric:**
    - **PHP code – correctness:** **25 %**
    - **PHP code – functionality:** **25 %**
    - **PHP code – quality of code (and inline comments)** **10 %**
    - **Overall presentation (alignment, colors, fonts, images, etc.):** **20 %**
    - **Report** **20 %**

- **Submission**: Please **submit your files via Blackboard** using *Blackboard's assignment submission option*. **Please don't hand in any assignment by email.**

- **Deadline**: November 18, 2015 – 11:59 p.m. ET

- These guidelines may be updated. Please check your email and Blackboard often.