COMP 3015

Introduction to Web Applications

with PHP

Assignment 2

Assignment 2

Start with the starter kit for Assignment 2. This is essentially the solution to Assignment 1.

index.php and thankyou.php

No changes required.

login.php

- if a cookie for `error_message` is set, display this message
  - after displaying this message, expire this cookie

signup.php

- if a cookie for `error_message` is set, display this message
  - after displaying this message, expire this cookie

redirect.php

- if the login validation fails,
  - create an error message indicating so and set that as a cookie with the name `error_message`
  - the cookie expires immediately
  - redirect to `login.php`
- if the signup validation fails,
  - create an error message indicating so and set that as a cookie with the name `error_message`
  - the cookie expires immediately
  - redirect to `signup.php`
- if login or signup validation passes,
  - start a session
  - set a session variable to flag the user is logged in
  - set a session variable to hold the user's username
- instead working with a text file, work with a MySQL database – a `users` table

profiles.php

`profiles.php` is protected by sessions. This page allows the logged in user to submit a new profile picture and view existing users

- if the session flag for a logged in user is not set, redirect to `index.php`
- read all rows from the `profiles` table. Each row represents a profile to be displayed; for each row in the table, display a profile panel
- there are already 3 example panels in the template. Be sure to remove these 3 examples before you submit
    - replace Username with the `username` from the row
    - replace Profile Picture with an `<img>` of the `picture` from the table row
    - update the `delete.php` link with the `id` from the table row
        - this link only shows up if the logged in user is the same as the post author
- populate the `username` input box with the session variable for username
- update the input form to POST to `profiles.php`
- upon a POST,
    - validate the input field for `picture` to be JPGs only and under 4 megabytes
        - store pictures in `profiles/` folder
        - give a unique name for each uploaded picture
    - if validation is successful, save the data to the `profiles` table
        - the `username` is from the session
        - the `picture` field is the filename of the uploaded picture
            - use an absolute path, take a 0; store only the filename

delete.php

- if the session flag for a logged in user is not set, redirect to `index.php`
- grab the `id` from the GET parameter
    - validate this `id`
- grab the `username` from the session
- check if the `profiles` table has an entry where
    - the `id` matches the id from the GET parameter
    - the `username` matches the session username
    - if both are matched, remove this entry from `profiles` table
    - redirect to `profiles.php`

logout.php

- destroy all sessions and cookies
- redirect to `index.php`

Database schema

users

```
create table users (
    id bigint primary key auto_increment,
    username varchar(15),
    password varchar(60)
);
```

profiles

```
create table profiles (
    id bigint primary key auto_increment,
    username varchar(15),
    picture varchar(60)
);
```

Bonus

Got time? Deal with duplicates – prevent users from submitting multiple profiles, prevent duplicate usernames. +1 bonus mark.

Submission

Zip up all your files and submit it to the D2L dropbox. (http://learn.bcit.ca)