<http://ec2-3-15-201-223.us-east-2.compute.amazonaws.com/final/>

My website is a photo-sharing website where users that are signed in can upload photos with an optional caption and if it is successful, it will post the image on the front page and the caption will be shown when the photo is clicked on, along with the users username. The purpose is to share photos, each with their own url, so it can also be used as an image hosting website. The target audience is anyone that enjoys photography, memes, or any types of photos. The entire website is mobile responsive.

1. All of my pages contain the 5 required HTML tags and I did not use any tables in my application. For example in greetings.php, I used !DOCTYPE, html,head,title, and body throughout the page.
2. On each page I include the header and when navigating through other pages, the header stays the same. Once someone is logged in, the log in link in the header changes to logout. The color scheme I chose is also consistent throughout and does not change. You can see on header.php, that I include on every page, is the same for each page.
3. I do not change the navbar in each page when it changes. I change it dynamically with php, so if a user is logged in, then the login link changes to logout by using php. This can be seen in lines 29-33 in my header.php, which I php include in each page.
4. My page upload.php is only accessible for logged in users. On lines 8-11 of upload.php you can see that I used !isset to determine if the user is logged in and if they are not it gives a JS alert and redirects them to the log in page. When a user logs in, they are redirected to a greetings page to indicate they are logged in and the navbar changes the log in and register links to logout, letting them know they are logged in. When a user logs out, they cannot access upload.php because my function still applies to redirect them to the log in page.
5. test and pass is registered in the database as a user.
6. Every single page utilizes PHP, one example is backendreg.php on line 81-83, which checks if an account exists and if not, sends the data to the database.
7. I properly use GET and POST. For GET, I included it on upload.php on line 14 to send a message if a file was uploaded, because this isn’t sensitive data and I wanted it to show a message on the URL changing. For POST, I included it on login.php line 47, because login information is sensitive data and I do not want it showing the information in the URL.
8. A form element I use besides the login page is the upload image form on my upload.php line 41 – 63. I also used a form on report.php line 43 -55, to send an email to me, reporting a post.
9. I give proper feedback for error messages. For example, on backendlog.php line 26- 27, I check if the username slot is empty and if it is it sends a message saying “Please enter your Username”.
10. On the home page, index.php, it is an image gallery for multiple user uploaded photos.
11. On about.php line 23, I included a YouTube video that explains AWS Ec2 servers.
12. I used proper JavaScript. On web.js line 40-57, I used a JavaScript function for having an image preview before upload.
13. I used JQuery. On web.js line 31 -33, I used JQuery to toggle an image flipping to the caption when a photo is clicked on.
14. I use bootstrap interface elements. For example on login.php line 39, I used bootstrap’s col-md class that sizes elements responsively.
15. I utilize AJAX on web.js line 6-24, it retrieves a message in my message.xml and prints it on my about.php saying “message from the developer:” with the xml data appended. This is so I can write easy messages about my website and it updates immediately.
16. This goes beyond the requirements, for example, using a database, making an icon with my initials for the tab, making it mobile friendly, and having a report system.
17. This is my document.

References:

I learned how to do responsive bootstrap navbars on <https://www.w3schools.com/bootstrap/bootstrap_navbar.asp>

And from my web development internship.

I had help with my log in page and register page from the code from the lectures and from

https://www.tutorialrepublic.com/php-tutorial/php-mysql-login-system.php

I had help with my flip cards on index.php from

<https://www.w3schools.com/howto/howto_css_flip_card.asp> but I changed it so it flips on click and not on hover.

I had help with my javascript image preview function from

<https://stackoverflow.com/questions/4459379/preview-an-image-before-it-is-uploaded>