Jeena Khaira and Max Segal

COSC 360 Web Programming

Professor: Dr. Scott Fazackerley

April, 2018

**Detailed Description of Implementation**

**Instapets**

**Blog Website for Pet Owners**

1. **User Implementation**

**1.1 Navigation Bar:**

* The navigation bar is the main point of navigation on our site. It is displayed on top of each page and allows the user to get to the main pages of our website. PHP is used to make changes to the navbar depending on what page the user is currently and if the user is logged in or not. HTML and css are used to highlight the page a user is currently on.

**1.2 Account Creation:**

* Users who wish to post content must create an account with Instapets. On the register page, users will be asked to input their name, email, a username, and password of their choice and a profile picture. There is also a confirm password field to ensure are no typos in the users password. All of this information is required by Instapets so if a user leaves a field blank they will be notified and the form will not be sent to our database. Once all of the correct information is inputted the user clicks submit and all the data is sent to the Instapets database and put into our users table. This is accomplished by using a post method on our sign-up form. We then send the posted data to our sign-up php file which connects to the Instapets database and then uses an sql insert statement to put the users information into our database. Once this process is complete the user is redirected to the login page so they can login to their newly created account.

**1.3 Login/Logout:**

* After a user has created an account they will be able to login using their Instapets credentials (username and password) and once a user is logged in they will have the ability to post content and create their own pet blog. On the login page there are two input fields within a form. One for the users username and the other for their password. Both fields are required so if one’s missing the user will be alerted and the form will not be sent. Once there is input in both fields and the user clicks login the form will send the information via a post method to our login php file. This file connects to the Instapets database and then uses an sql select statement to check if there is a user in our users table with credentials that match the username and password that were just inputted. If there is no match the user is alerted and allowed to retry inputting another username, password combination. If there is a match, we start a php session and the user is redirected to their personal Instapets blog page where they will be able to post and comment to their heart's content. To logout, the user just has to click on the logout tab located on that navbar. The logout tab will only appear when the user is logged in. Once clicked it will terminate the current php session and redirect the user to the homepage.

**1.4 User Blog Page:**

* After a user logs in they will be redirected to their Instapets blog page. On this page a user can view their profile information (username and password) and previously created posts. They can create a post and edit their profile picture and/or password. The page connects to our database and fetches the users profile picture location and username by inputting the php session username variable into a sql select statement. Once the information is fetched, we grab the profile picture from our pro pics image file using the fetched image location. The profile picture and user are then displayed on the users Instapets blog page. To display all of the user's posts we grabbed all of the created posts from the posts table whose username field is equivalent to the current php username session variable. We then loop through the each post in the result set and echo each one onto the users blog page.

**1.5 Create Post:**

* To create a post a user must go to their blog page where a create post form will be displayed. The user will be allowed to share photos which will be referenced by a caption and pet type. All fields in the form will be required and if forgotten the user will be prompted. Once submitted, the form will send the data via a post method to our post php page. This page connects to our database and inserts the data into the posts table. The post will be given a postID and will also have a username associated with it.

**1.6 Edit Profile Picture/Password:**

* A user can change their profile picture or password by clicking settings on the bottom of their blog page which will redirect them to their account settings page. Here, the user is given the option to change their profile picture and/or password. To change their profile picture the user must first choose a new picture and click submit. The picture will be sent to our change pro pic php file. This page will connect to our database and move the new profile picture into our pro pics image file. Then an sql update statement will change the users profile picture field to the new profile picture location. Changing a passwords works similarly. The user inputs their old password, a new password and a confirmation of their new password. Once submitted, we check if the old password submitted is equivalent to the users current password. We then check if the new password matches the confirmation password and if all these checks pan out we use an sql update statement to change the users password.

**1.7 Comment on Posts:**

* Users can add comments to a post. To do so a user must first be logged in. If not, the comments of an associated post will just be displayed below the content of the post without the ability to add your own. If you are logged in, a post comment form is displayed at the bottom of each generated post. This is done by checking the logged in session variable. If the logged in variable is equivalent to true we know that there is currently a user logged in and the comment form should be displayed. The user cannot submit a blank comment and if done they will be alerted and the comment will not be posted. Once the user submits a comment, we send it to our database and put it into our comments table using an sql insert statement. We insert the comment with an ID, the username associated with it, the post ID associated with it and the text that will be displayed as a comment. To display the comments below its post we use an sql select all statement to grab all of the comments that are associated with the current post ID. We then loop through the result set and display each comment as a row of a table in the comments section of a post.

**1.8 Search by Category:**

* On a users newsfeed they will have the ability to filter through Instapets posts by pet category. A drop down menu will be displayed at the top of the users newsfeed where they can select a pet category. By default, a users newsfeed will display all posts. Once a category is selected the newsfeed will asynchronously update and only display the pets of the selected category. This was accomplished by creating a jquery function which used ajax to send and grab the information needed. The php file associated with the ajax function used an sql select statement to grab all of the posts of a certain pet type.

**2. Administrator Implementation**

**2.1 Delete Users/Posts:**

* The administrator will have the ability to delete users. All users will be displayed in a table with a delete button located on the right hand side of each user row. The delete button is within a form and once clicked will send the rows username to our delete users php file via a post method. We then delete the user from our database by using an sql delete statement with the username that was sent from the form. The user is deleted and will no longer be displayed within the admin user table.
* The posts will be displayed on a separate page but like the users table there will be delete buttons above each post. Once clicked, like a user, the post will be erased from our database and will no longer be displayed.

**2.2 Search Users by username:**

* Administrators will be able to search for a user by their username. There is a search bar above the delete users table. It uses a php file to grab all of the users that are ‘like’ what the admin searched.

**2.3 Search Posts by Pet Category:**

* Administrators will be able to filter posts by pet category. There is a search bar above the content. It uses a php file to grab all of the posts that are ‘like’ what the admin searched.

**2.4 Edit captions:**

* Administrators have the ability to edit the caption of a selected post. An edit button will appear above each post and once clicked will redirect the admin to a edit caption page where a form will be displayed. Inside the form, administrators choose what they’d like to change the caption to and when submit is clicked the information is sent to a php file that updates the selected post’s caption. This new caption is then displayed on the post.