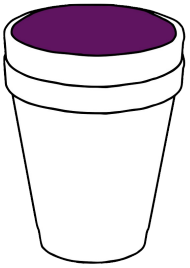
April 11th , 2018



TrapBuzz

Online Blogging Website

GitHub Team Name: Off-White

COSC 360 - 2018

http://cosc360.ok.ubc.ca/44790153/src/server/signup.php

Noman Mohammad - 44790153

Jaryd Fisher - 34065152

Logic Process

Pre and Post Conditions

Sign-Up Process:

- User inputs their first and last name
- User inputs a unique email address
- The user creates a password and confirms in
- User then clicks sign me up and creates their account
- If the user doesn't wish to sign up at the time they may select "Sneak in with minimum accessibility? Enter" and they will be given limited access to the website

Log-in Process:

- User Inputs their unique email and password
- User selects Log in and is logged into their account

Search Process:

- User selects search bar
- User inputs a search result and clicks on search
- User is given Search results

Write Blog Process:

- User selects the new blog post icon
- User can then add a title and content before posting blog

Sign-Out Process:

- User selects the triple bar icon
- User selects sign out and they are signed out of their account and brought back to the sign-up page

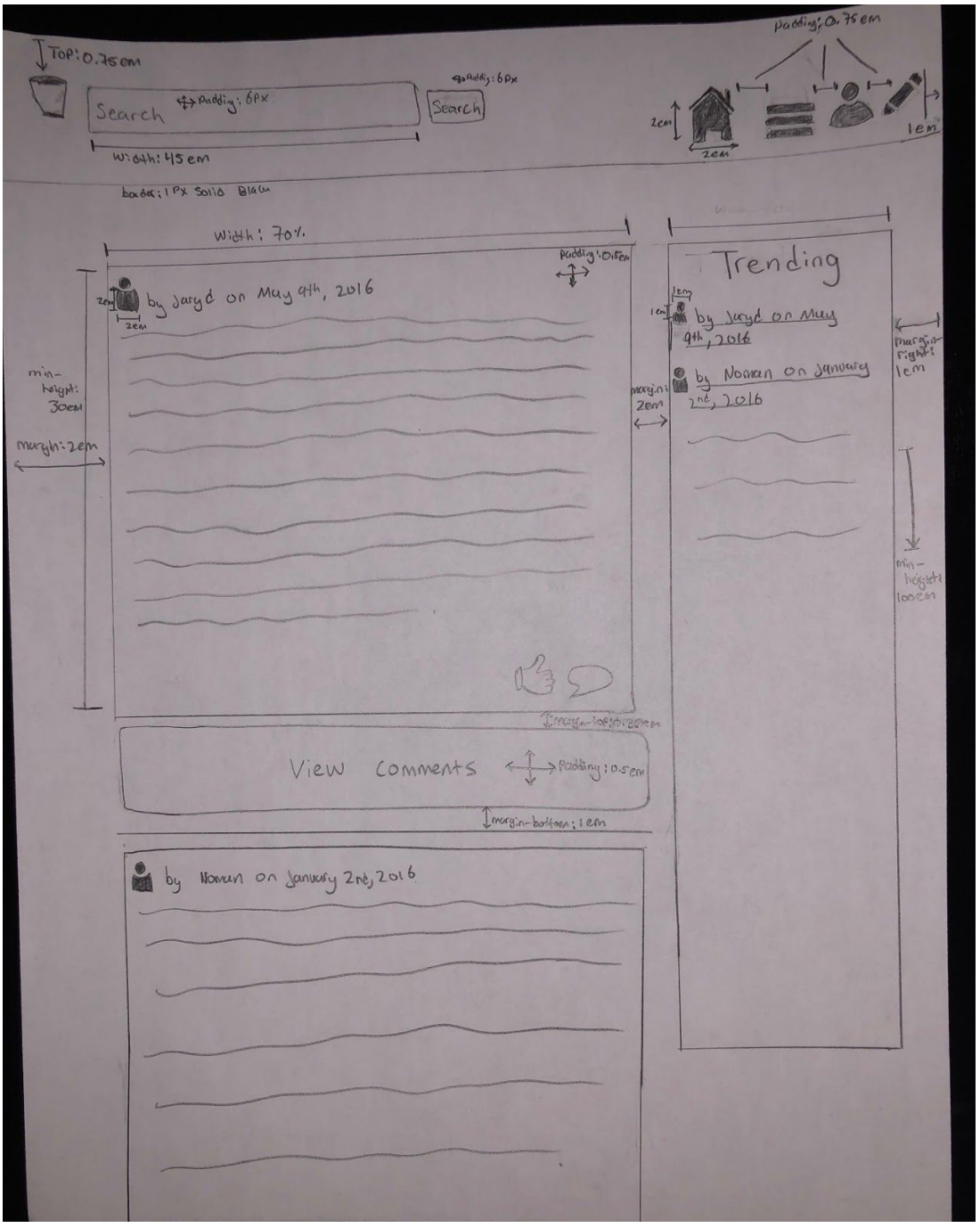
Change Password Process:

- User is on Home page
- User selects change password located on the right side of the page
- User is brought to the change password page
- User inputs new password and confirms it
- User then selects create new password

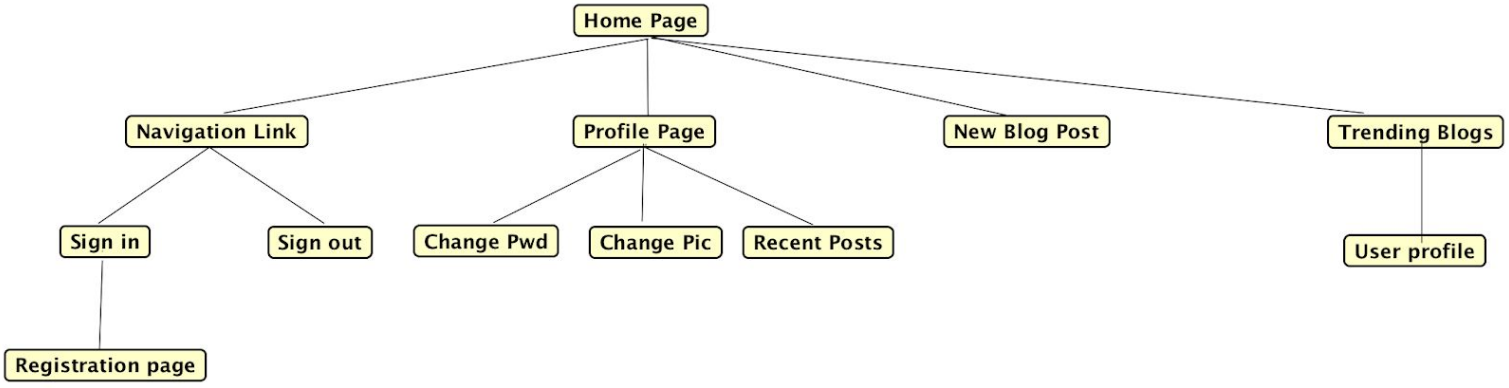
Change Profile Picture Process:

- User selects change profile picture
- User selects new profile picture in their library
- The new profile picture is created

Layout Plan



Site Map



Client-Side Security

When a user registers to the website, javascript will do some basic client side validation including:

- Not being able submit empty data
- Password length being a minimum of 7 characters
- Password and confirm password must match
- If fields left blank, red outline appears
- Numbers not allowed in the name texfields
- Emails must be formatted correctly
- Passwords are inputted using the password input type to prevent others from seeing the password typed
- Post method used for sensitive data transfer to eliminate query strings showing data

Other security issues such as how we will store the password will be addressed moving forward with salting and hashing techniques.

Site Functionality

Sign up

- The site has full registration functionality that allows a user to pick a unique username and prevents other users from picking the same one.
- The password is fully hashed with SHA 256 and is also salted with a randomly produced string of characters.
- The site inserts a new user into the database and they can proceed to login with their username and password.
- The site also dynamically changes the options in the menu icon for login/ logout depending on if the user is signed in or not.
- Once signed in the user can also see their first name displayed in the header to know they are logged in.

Sign up - forgot password

- Registered users can enter their email and get a new password back if they forget their old password

Admin page

- The admin can go into the site and view all posted blogs and delete any blog they would like
- The admin login information is **username: admin password: 1234**

Sneaking in

- The site allows users to check out the site by sneaking in without registering but limits their accessibility on the site
- The users can only view the posted blogs
- The users cannot click on the profile icons or new blog post icons, as well as they cannot browse by categories or view user profiles. If they attempt to they are greeted with a not signed in page or a page overlay.
- Users cannot like or comment

Home Page

- Registered users can view all the posted blogs on the home page and also view by different categories as well.
- Registered users can click on individual blogs which takes them to the profile of the user who posted it. If the user click on their own blog it takes them to their profile.
- Users can view the date and time each blog was posted.
- Registered users can comment on each blog post which will dynamically make an ajax request and display the user comments on the site.
- Registered users can like other users posts which is also dynamically updated through ajax
- Registered users can view trending topics that will be based off of a certain amount of likes.
- Registered users can search anywhere on the site that will bring them results for blogs by title on the home page

Profile

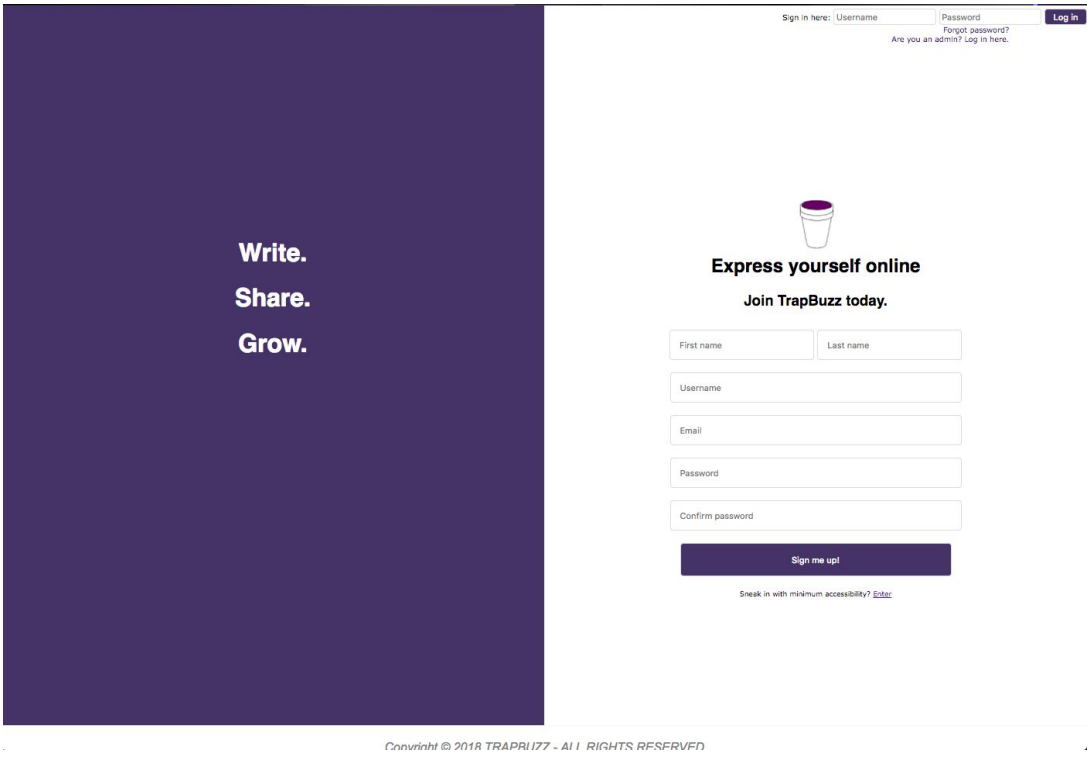
- Registered users have their own profile page where they can view all of their recent blog posts.

- Registered users can see all of their information including first name , last name, email etc.
- Registered users can upload their own personal profile picture, or if they don't a default one is set.
- When a registered user clicks on another users page the options of change profile picture and password appear and disappear depending on if the user clicks on their blog post or not.
- Registered users can edit their profile, being able to change their name, email or password.
- Registered users can change their ‘About me’ bio.
- Registered users can delete their blog posts if they wish to do so.

Blog Post

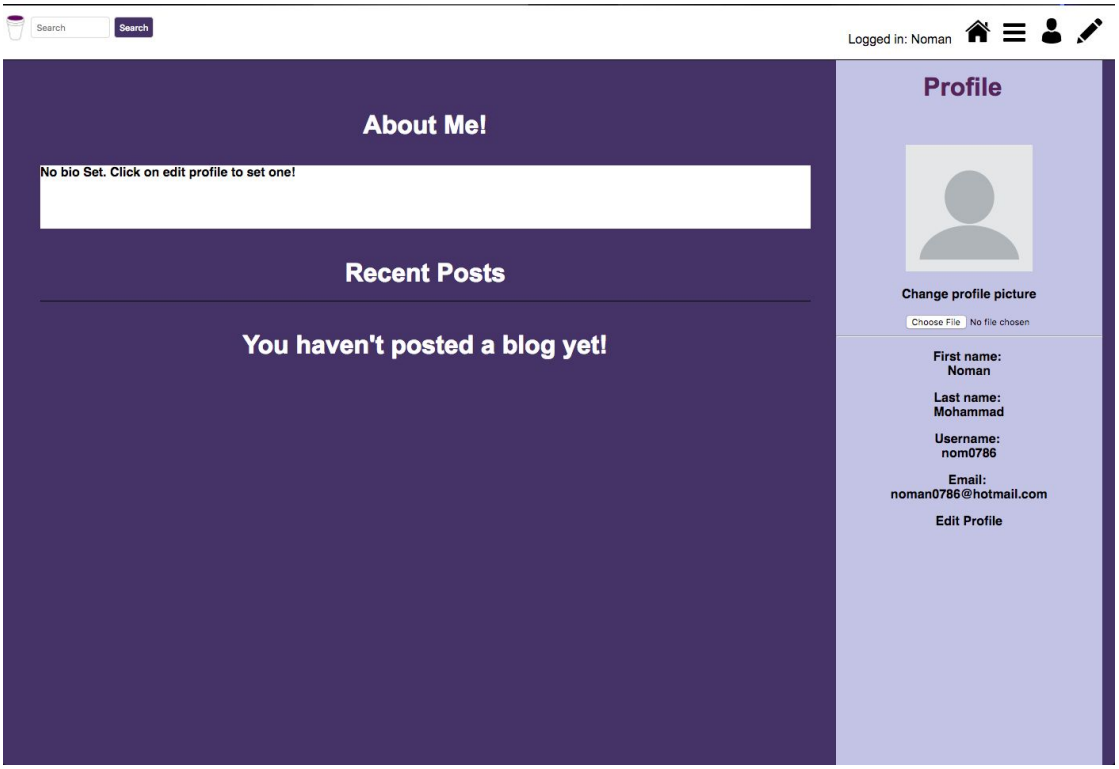
- Registered users can post a new blog by clicking on the pencil icon. The inserted data is then saved in the database where the user id of the poster is saved along with.
- Blogs cannot be posted unless all fields are filled in, **including a picture.**
- Registered users can upload a photo with their blog and the photo is stored as a BLOB type in the database.
- The time and date is also dynamically inserted into the database as well
- Once the blog is posted the Registered user is redirected to the homepage where they can view their newly posted blog.

Site Walkthrough



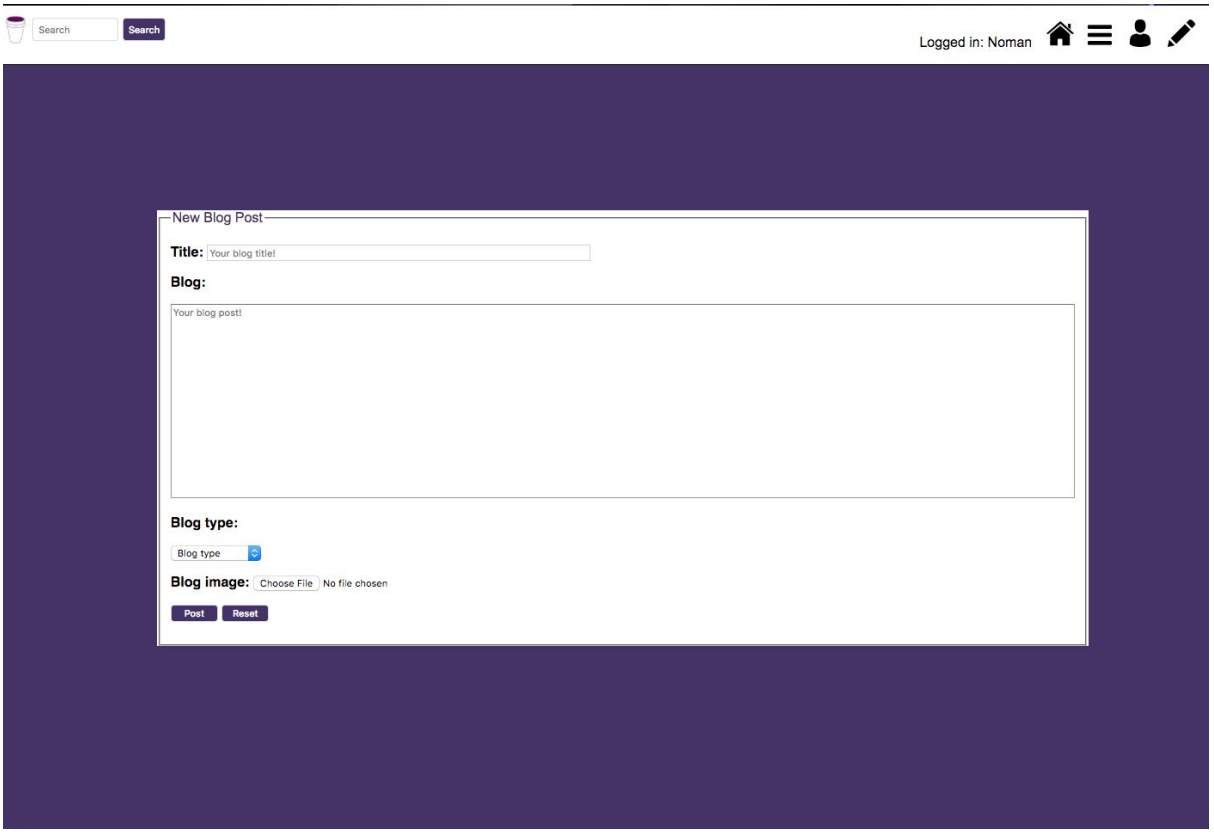
Here a user can sign up to the website. The site is fully validated with client side javascript to let the user know if they missed anything. The user must enter their firstname, lastname, username for the site, email address, and password

Once a user has successfully logged in they are shown a confirmation screen that they have signed up and then can proceed to log in and are then directed to the home page of the website. Here the user can go to their profile, or upload a new blog. They can also search for content anywhere on the site as well as view all categories and any trending posts.



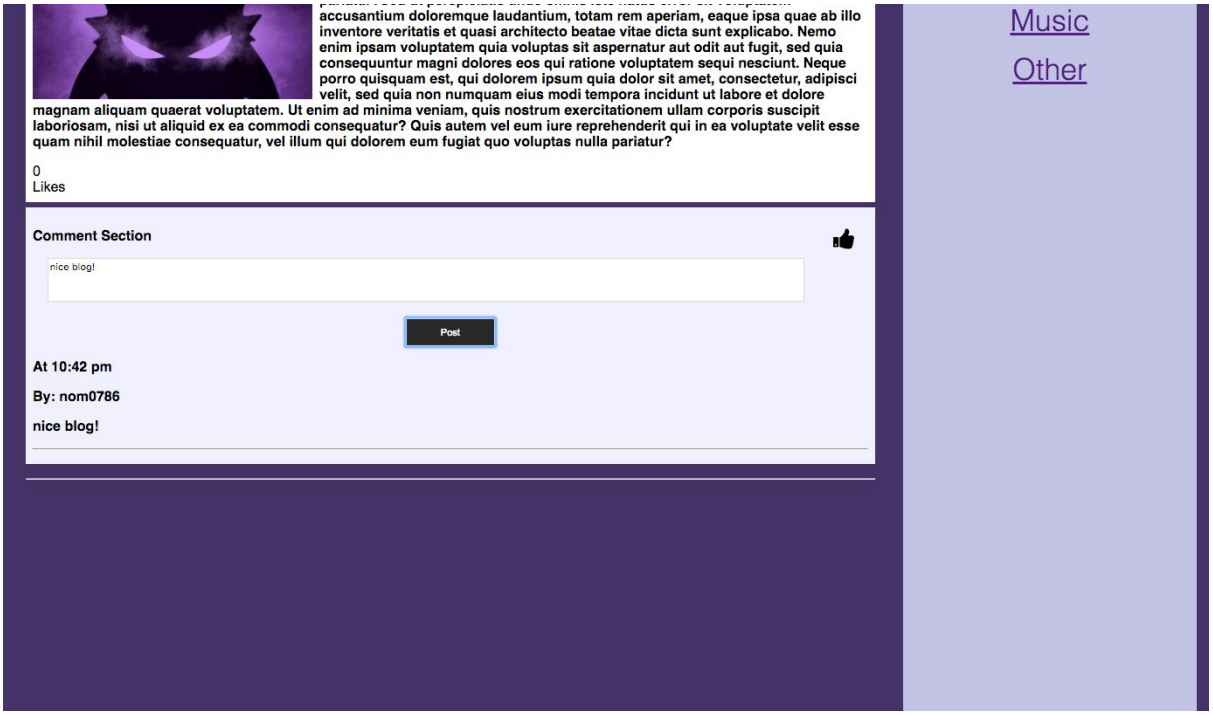
Here the user clicked on the profile button that takes them to their profile. Here they can view all of their recently published blog posts as well as update their profile by clicking on the edit profile button. The user can also set their profile picture.

Once the user clicks on edit profile they can edit their first name, last name, email address or update their password. If the user wishes they can also set their bio to say abit about themselves in case anyone views their profile.

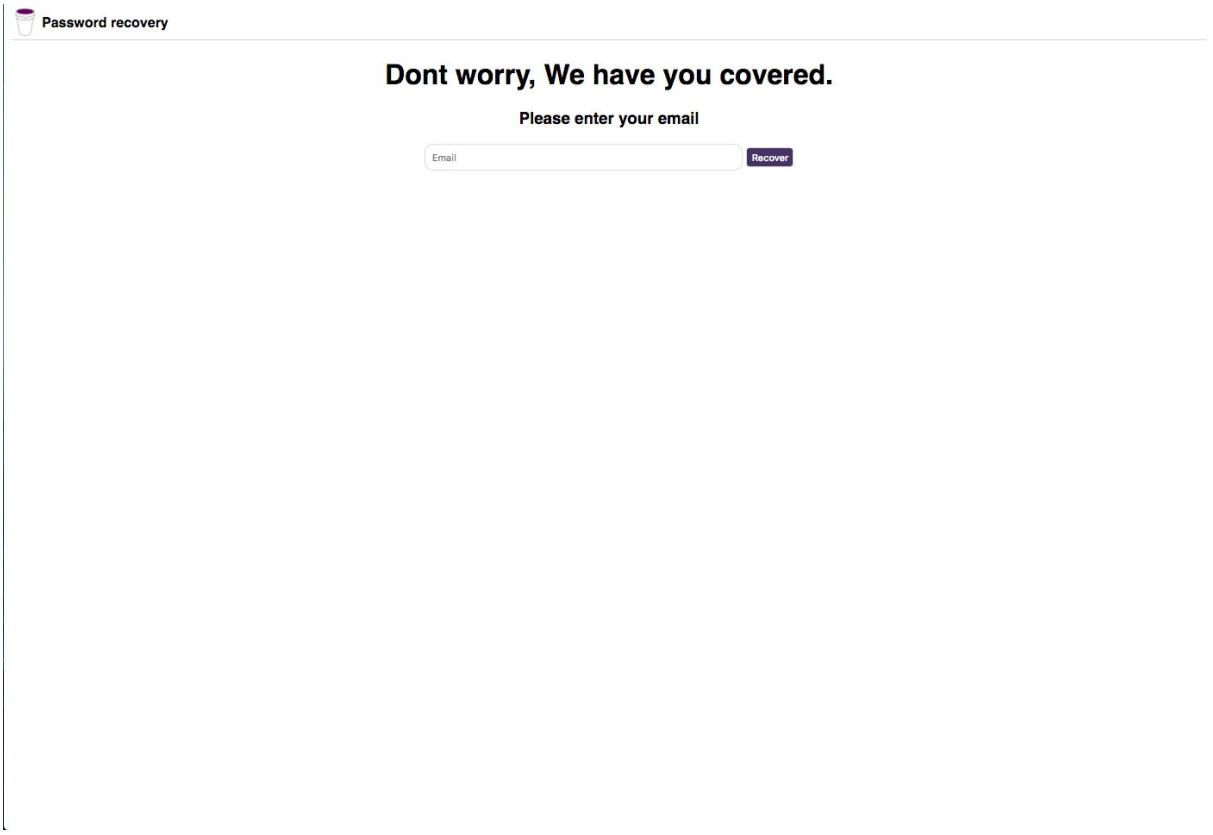


Next, once the user click on the pencil icon they can upload a blog. Here the user must fill out a title, blog content, the blog type and a blog picture as well.

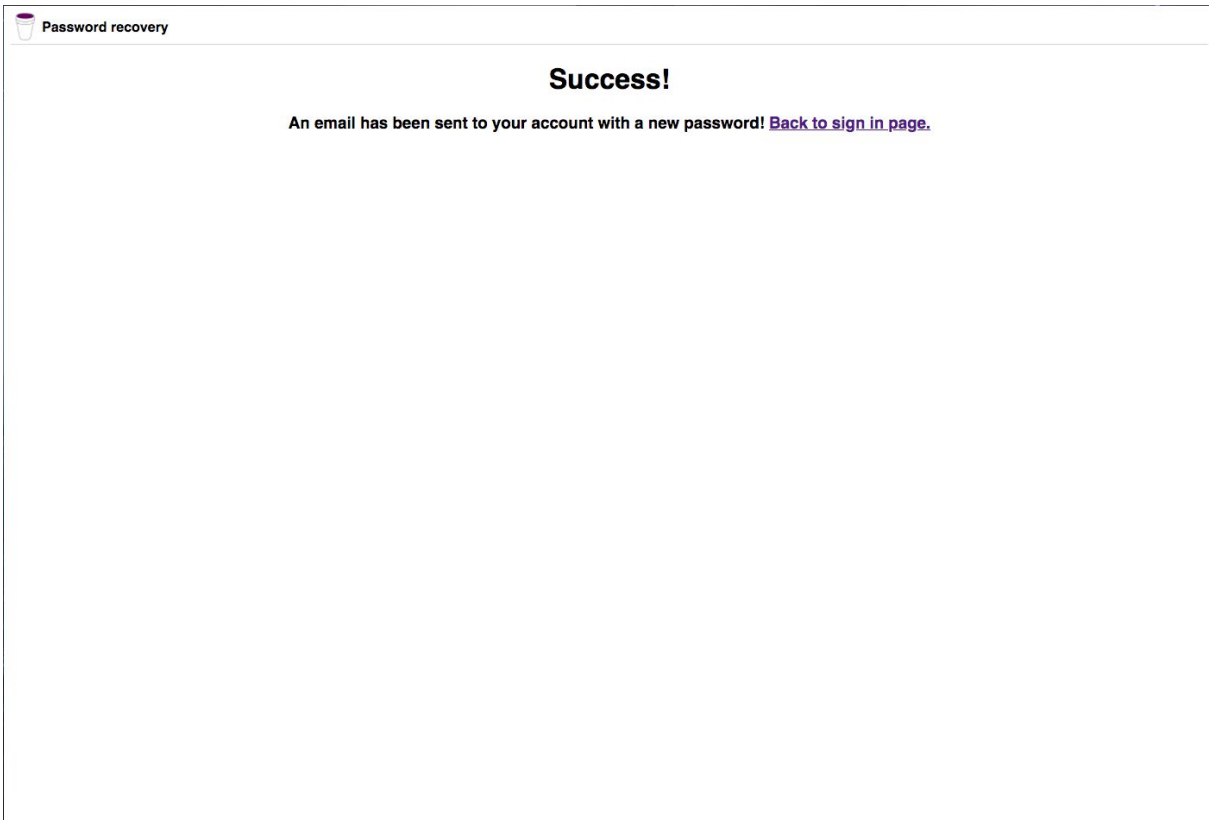
Once they upload a blog they are redirected to the home page where they can view their recently uploaded blog and showcase the rest of the world! From here a user can either like the blog post by clicking the thumbs up button, or comment on their blog post as well.



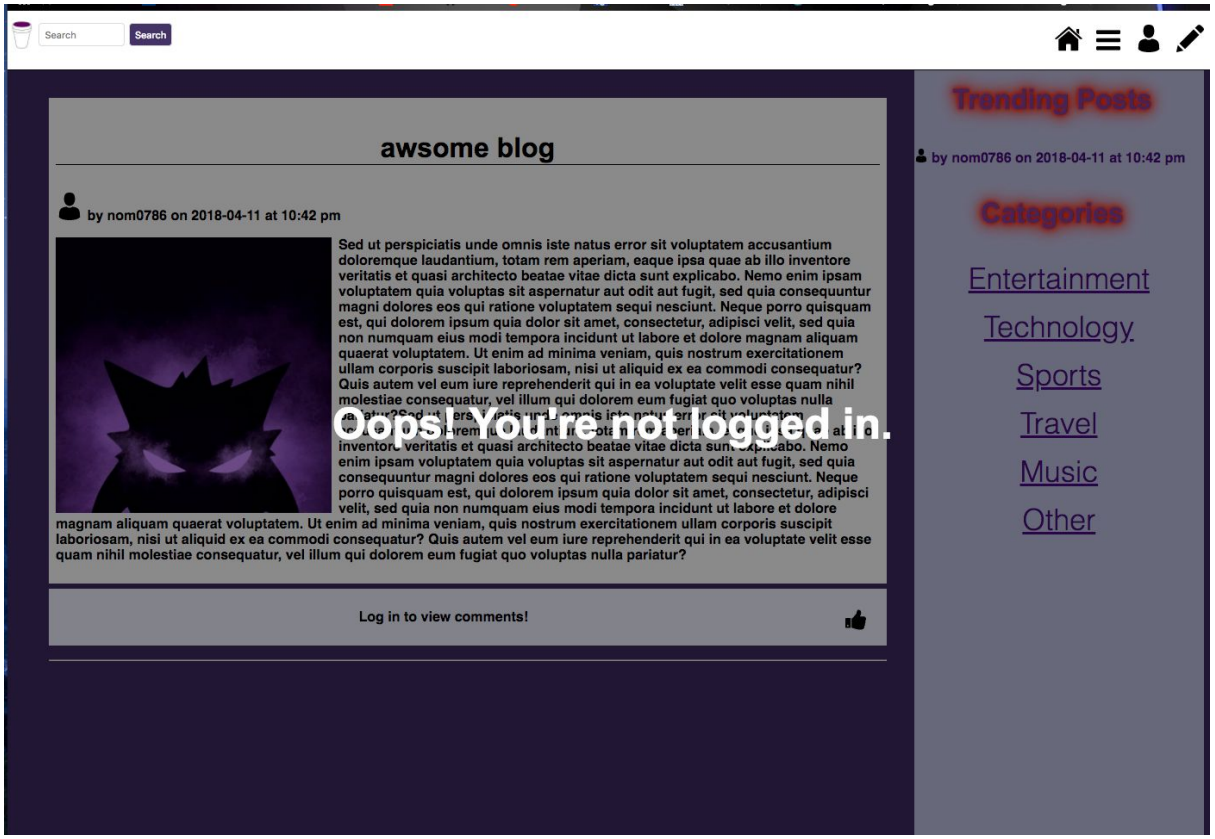
Once a user comments the server will send an ajax request to the comments php page where the comment is then dynamically uploaded to the blog without the need of refreshing. The same goes for likes.



Back in the sign up page, if the user forgets their password they can easily enter their email and request for a new password.



The user is then displayed with a confirmation screen assuring them that an email is sent.



If the user attempts to like, or access the profile or new blog page they are presented with a pop up telling them they are not logged in.

Implementation

Sign up/ login

The sign up page takes in all the fields and send it off to the signup-inc.php page which is kept in the include folder where the information is checked through with the database for duplicate username/emails and then inserted into the database. The passwords are all hashed aswell using sha256 and a randomly generated string of characters to add salt to the password. Then the user is redirected back to the signup page where they can proceed to log in with their recently created credentials. Here the information is then sent off to login-inc.php. Once the information is validated with the database a session is started for the user with all of their information.

Profile

The user can edit their profile where the data is sent to editprofile-inc.php so the information can be sent to the database.

New blogpost

When a user creates a new blog the information is then sent off to newpost-inc.php where the data is entered into the database. Images are saved as blob types.

Home

On the homepage the database is queried for all blog posts and if a user searches the information is sent to the home-search.php where the isset function is used to test if the search submit is pushed. If no blog posts are available we implemented an if statement to check the result check and if it is less than 1 we echo out a message saying no blog posts. On the homepage as well there is a categories bar that allows the user to view by categories. The way it is implemented is that the information is sent through the header where the GET method is used to retrieve the type and then with the SQL where clause we queried the blogs by type. Same is don for trending posts where the database stores the number of likes and we showcase the most recently liked blog posts or the ones with the most likes.

Likes

The thumbs up button sends a post request to the ajaxLikes.js file where it intercepts the post request from the like button. The ajax reads the blogId from the button to determine which post it came from. Then the ajaxLikes.js sends the blogId to the newLikePost.php file where it sends a insert query into the database. The ajaxLikes.js file then refreshes the newLike div tag with the new amount of likes.

Comments

The comment text box reads the text that has been input by the user and sends a post request. The post request is intercepted by the ajax.js file where it prevents default and takes the values from the comment form including blogId and the input from the textbox. The ajax.js file then sends an insert query into the database and uses the loadComment() function to refresh the div containing the comments where it adds the new comment the user has made. Everytime the user loads the page loadComment() is run where it constantly updates comments.

Email Recovery

The email is entered into the form and then validated through the database to ensure that user with that email does in fact exist, if true then an email is sent out to that user using the PHP MAIL function. If no user exists an error message is displayed telling the user.

Admin Page

The admin_signin.php page takes in information and then sends it off to admin-inc.php where the data is validated and then takes them to admin.php where the admin can now delete posts by getting all of the blogs queried on the page and and the blog id is sent through the form as a hidden field.

Testing, deployment and branching

The site had a usability test conducted by peers to ensure that all things on the site were doing what they were supposed to. The site was deployed through filezilla onto the school

server. The development of the site took place on atom and github was used as version control. No branching was done since all merge conflicts were handled individually through atoms github interface.