

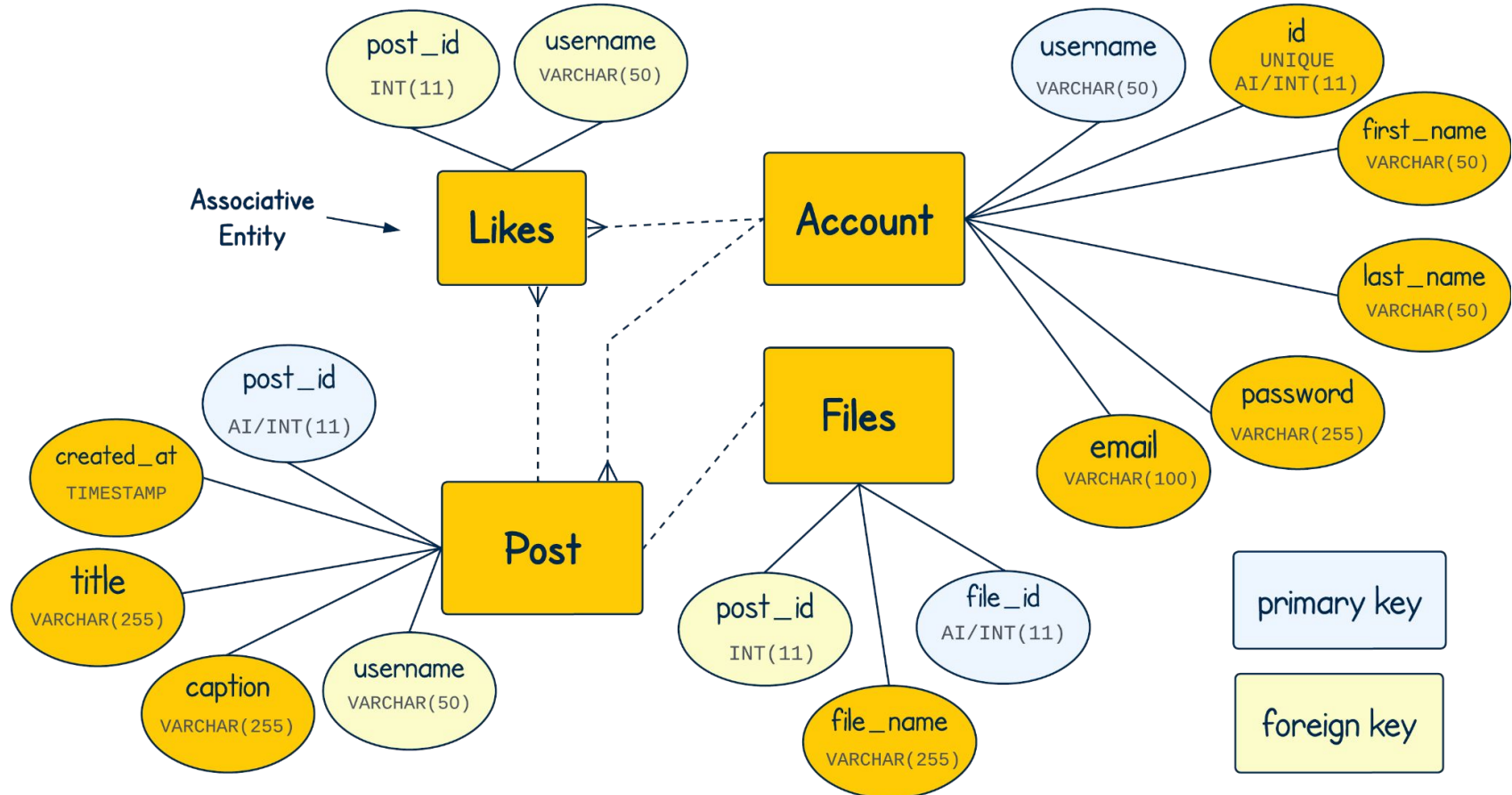
Wolverine Work: Backend

By: Taylor Williams, Mnahil Syed,
Emily Buzle & Kylie Callison

Database Schema Design



WOLVERINE WORK DATABASE SCHEMA




List of Pages & Implementation

The background of the slide is split diagonally from the bottom-left to the top-right. The upper-left portion is a solid yellow color, and the lower-right portion is a solid dark blue color. A thin white line separates the two colors.

Login Page


```
23 // Prepare our SQL, preparing the SQL statement will prevent SQL injection.
24 if ($stmt = $con->prepare('SELECT id, password FROM account WHERE username = ?')) {
25     // Bind parameters (s = string, i = int, b = blob, etc), in our case the username
26     $stmt->bind_param('s', $_POST['username']);
27     $stmt->execute();
28     // Store the result so we can check if the account exists in the database.
29     $stmt->store_result();
30     if ($stmt->num_rows > 0) {
31         $stmt->bind_result($id, $password);
32         $stmt->fetch();
33         $userInput = md5($_POST['password']);
34         // Account exists, now we verify the password.
35         if ($userInput == $password) {
36             // Verification success! User has logged-in!
37             // Create sessions, so we know the user is logged in, they basically act
38
39             session_regenerate_id();
40             $_SESSION['loggedin'] = TRUE;
41             $_SESSION['name'] = $_POST['username'];
42             $_SESSION['id'] = $id;
43
44             //CHANGE NAME TO THAT OF THE HOMEPAGE IF DIFFERENT!!!!!!!!!!
45             //header('Location: ../home/home-page.html');
46             echo 'homepage';
47             exit;
48         } else {
49             // Incorrect password
50             echo 'Incorrect password!';
51         }
52     }
53 }
```


- Pre-existing account users can successfully login
- Connected to wolverine work database username and password is verified via PHP and SQL statements
- Redirects to homepage upon success or else error message is displayed



Welcome Back!

Login





☒ Remember Me? [Forgot Password?](#)

Don't Have an Account? Sign Up Today!

Register Page

```
$email = isset($_POST['email']) ? trim($_POST['email']) : null;

// List of allowed domains
$allowed = [
    'umich.edu',
];

// Make sure the email address is valid
if (filter_var($email, FILTER_VALIDATE_EMAIL)){

    // Separate string by @ characters (there should be only one)
    $parts = explode('@', $email);

    // Remove and return the last part, which should be the domain
    $domain = array_pop($parts);

    // Check if the domain is in our list
    if (!in_array($domain, $allowed))
    {
        // Not allowed
        echo 'Did not input a correct University of Michigan email, try again!';
    }
    else {
        // Finally, register user if there are no errors in the form
        if (count($errors) == 0) {
            $passwordEncrypt = md5($password);//encrypt the password before saving in the database

            $query = "INSERT INTO account (first_name, last_name, username, email, password)
            VALUES('$firstname', '$lastname', '$username', '$email', '$passwordEncrypt')";
```

- Non previously registered users can successfully register an account
- Upon email validation and other correct info via PHP, inputted user data will be stored to the database via SQL insertions.



Sign Up Today!

Register

	<input type="text" value="First Name"/>
	<input type="text" value="Last Name"/>
	<input type="text" value="Email Address"/>
	<input type="text" value="Username"/>
	<input type="password" value="Password"/>

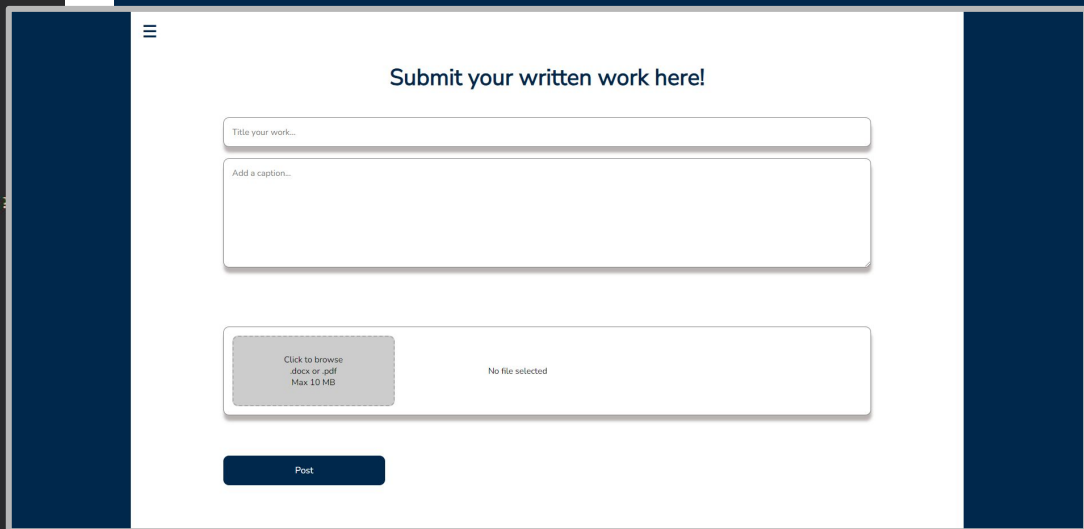
Create An Account

Already Have an Account? Sign In

Create a Post Page

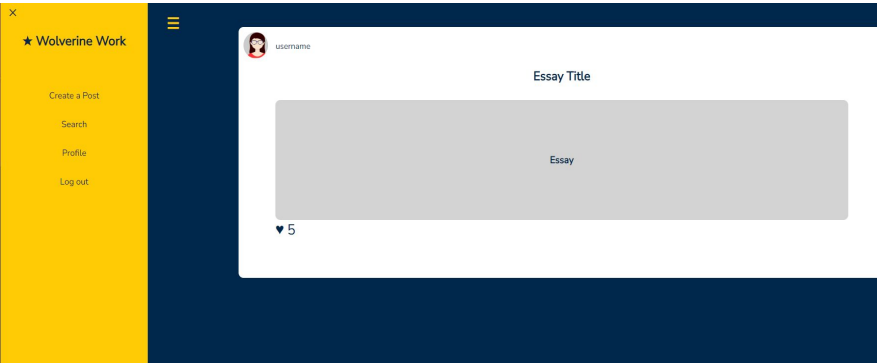
```
CreateAPost > CreatePost.php
You, 1 second ago | 1 author (You)
1 <?php
2 session_start();
3
4 // Check if Logged in You, 12 hours ago * username & file size/type verification
5 if (!isset($_SESSION['loggedin'])) {
6     header("Location: ../login/login-form.php");
7     exit();
8 }
9
10 // Connect to the database
11 include('../database/connection.php');
12
13 if ($_POST) {
14     // Populating variables with form data
15     $title = $_POST['title'];
16     $caption = $_POST['caption'];
17     // Need session variable to complete
18     $username = $_SESSION['name'];
19
20     // Prepared Statement to insert post info into the database
21     $insert_post = "INSERT INTO post(username, title, caption) VALUES (?, ?, ?)";
22     $stmt = $conn->prepare($insert_post);
23     $stmt->bind_param("sss", $username, $title, $caption);
24     $stmt->execute();
25
26     // Getting the Auto Incremented ID from the Last query
27     $post_id = mysqli_insert_id($conn);
28
29     // Close stmt
30     $stmt->close();
31
32     // Upload file to uploads folder
33     include '../includes/upload.php';
34
35     // Redirect back to the homepage
36     header("Location: ../home/homepage.html");
37     exit();
38 }
39 ?>
40
```

- Secure, PHP prepared statement for database insertion from the submission form
- Uploads file to folder
- Redirects to homepage after submission
- Side navigation bar redirects user



The screenshot shows a web interface for submitting a post. It features a dark blue header with a hamburger menu icon on the left. The main content area is white and contains the heading "Submit your written work here!". Below the heading are three input fields: a text field for "Title your work...", a larger text area for "Add a caption...", and a file upload section. The file upload section includes a button that says "Click to browse .docx or .pdf Max 10 MB" and a message "No file selected". At the bottom of the form is a dark blue button labeled "Post".

Home Page



- The profile image and username are displayed with the related essay post.
- The title of the essay and preview of the essay correlate with the each other.
- Clicking on the preview of the essay will direct the user to the whole essay on another page.
- The user needs to be able to search for a specific essay topic and the related essays should be displayed.
- The number of likes per post are accurately displayed.
- The heart icon changes to red once clicked and switched back to blue if clicked again.

Profile Page



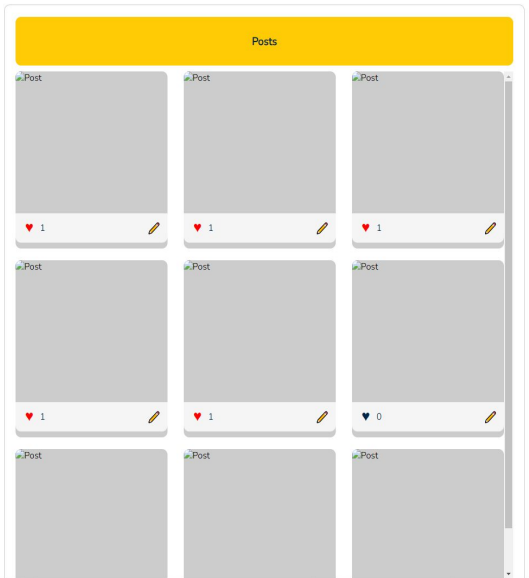
Mnahil Syed

About Me

Write something about yourself...

Contacts

Add your contact information...



- Allows users to manage profile information, including a profile picture, about me section, and contact details.
- Displays a collection of posts with like and edit functionalities.
- The application connects to a database for storing user profiles, post data, and other relevant information.
- Like counter algorithm prevents users from liking a post more than once.
- Profile name editing logic allows users to edit their name with dynamic UI updates.

Roles and Access Control



Roles and Access Control

Registered User

- Can login to the website
- Can post to the website
- Has designated profile page
- Can edit their profile page and user information

Unregistered User

- Can view home page contents
- Can register under a valid umich email address

Administrator

- Can view and edit website contents
 - Backend and Frontend
 - Database material
 - User content (such as removing posts)

Role Implementation

Registered User

- Login through backend PHP and database
 - Login-page.html
 - Authorize.php
 - Database: “wolverinework”
- Can post to website through backend PHP
 - CreateaPost.php
- Can view and edit profile page through embedded Javascript
 - Profile.html

Unregistered User

- Register through backend PHP and database insertion
 - Register-page.html
 - register.php

Administrator

- Edit website contents and permissions through HTML/PHP/Javascript/SQL
 - Database: “wolverinework”

Testing Plan



Login Page

Test Plan

Test Case 1 :

- Input all correct info into fields (username, password)
- Expected result: Successfully login

Test Case 2:

- Input incorrect username, correct password
- Expected result: Error message, cannot login

Test Case 3:

- Input correct username, incorrect password
- Expected result: Error message, cannot login

Test Case 4:

- Input all incorrect info into fields (username, password)
- Expected result: Error message, cannot login

Test Case 5:

- Do not enter any any info into fields (username, password)
- Expected result: Error message, cannot login

Register Page

Test Plan

Test Case 1:

- Input all correct info into fields
- Expected result: Successful registration

Test Case 2:

- Input non umich email
- Expected result: Error message, cannot register

Test Case 3:

- Input no username or password
- Expected result: Error message, cannot register

Test Case 4:

- Input no firstname or lastname
- Expected result: Error message, cannot register

Test Case 5:

- Input no email address
- Expected result: Error message, cannot register

Home Page

Test Plan

Test Case 1:

- Click on the heart icon
- Expected Output: Number of likes increases/decreases and the heart icon turns red/blue

Test Case 2:

- Click on the essay preview
- Expected Output: directed to a new page with the complete essay

Test Case 3:

- Post displayed to the user
- Expected Output: correct username and profile picture are displayed with the related essay

Test Case 4:

- Input essay topic in search feature
- Expected Output: related essays will be displayed for the user to browse

Test Case 5:

- User posts an essay post
- Expected Output: appear on the home page

Create a Post Page

Test Plan:

Test case 1:

- Not adding a title or caption
- Expected Result: This is a required field warning

Test case 2:

- Attaching correct file type and title/caption
- Expected Result: Redirect to homepage and database populated with correct information

Test case 3:

- Attaching an incorrect filetype
- Expected Result: Error message for incorrect file type

Test case 4:

- SQL injection in title
- Expected Result: No error message

Test case 5:

- Attaching an extremely large file type
- Expected Result: Error message for file size

Profile Page

Test Plan

Test Case 1:

- Click on the profile picture.
- Expected Result: Allow user to upload profile photo to their page.

Test Case 2:

- Click on User's Name
- Expected Result: Allow user to change their displayed name.

Test Case 3:

- Click on the About Me and Contact Me section.
- Expected Result: Allow user to edit and add their information to each text input.

Test Case 4:

- Click on heart Icon.
- Expected Result: Change color of icon and make the counter go up by 1.

Test Case 5:

- Click on edit icon.
- Expected Result: Allow user to edit their post.

Thank you!
Any Questions?