**Source code files for the SurveyEngine project:**

**admin.php**

<SCRIPT LANGUAGE="JavaScript">

function copyToClipboard(text) {

window.prompt("Copy to clipboard: Ctrl+C, Enter", text);

}

</SCRIPT>

<?php

$page\_title = 'Admin Page';

include ('../includes/header.php');

require ('../config.php'); //Connect DB

//If not an admin, redirect to user home page

if(!isset($\_SESSION["admin"]) && $\_SESSION["admin"] !== true) {

header("location: /SurveyEngine/home.php");

exit;

} else {

$adminID = $\_SESSION["id"]; //Get user ID

}

?>

<div class="page-header">

<h1><b>Admin Panel</b></h1>

</div>

<div class="text-center">

<a href="create.html" style="width:150px;" class="btn btn-primary">Create Survey</a><br><br>

<a href="view\_user.php" style="width:150px;" class="btn btn-info">View Users</a><br><br>

<div id="createdSurveys" style="margin-top:5%;">

<h4>View created surveys:</h4>

<?php

//Survey list

$query = "SELECT `entity\_user`.`user\_id`, `entity\_survey`.`title`, `entity\_survey`.`unique\_id` FROM `survey`.`xref\_user\_survey` AS `xref\_user\_survey`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_user` AS `entity\_user` WHERE `xref\_user\_survey`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_user\_survey`.`user\_id` = `entity\_user`.`user\_id` AND `entity\_user`.`user\_id` = '$adminID' AND `entity\_user`.`type` = '0'";

$rs = mysqli\_query($link, $query); //Get result set

$numSurvey = mysqli\_num\_rows($rs); //Get number of surveys created

if ($numSurvey > 0) {

echo "<table border='1' align='center'>";

echo "<tr><th>".'Survey Title'."</th>";

echo "<th>".'Link to Results'."</th>";

echo "<th>".'Survey Invite Link'."</th>";

echo "<th>".'Delete Survey'."</th></tr>";

while($re = mysqli\_fetch\_array($rs, MYSQLI\_BOTH)){

$surveyID = $re['unique\_id'];

$string = 'http://' . $\_SERVER['HTTP\_HOST'] . "/SurveyEngine/survey.php?id=" . $surveyID;

echo "<tr><td>" . $re['title'] . "</td>";

echo "<td> <a href='./result.php?id=$surveyID'>Link</a> </td>";

echo "<td> <button id='reg' onclick='copyToClipboard(\"$string\")'>Copy</button> </td>";

echo '<td> <a href="delete\_survey.php?id=' . $surveyID . '">Delete</a></td></tr>';

}

echo "</table>";

}

else {

echo "None";

}

?>

</div>

<div class="settings">

<h3><span class="glyphicon glyphicon-cog"></span> Settings</h3>

<a href="../reset\_password.php" class="btn btn-warning">Reset password</a>

</div>

</div>

</body>

</html>

**create.php**

<?php

$page\_title = 'Survey Created';

include ('../includes/header.php');

require\_once ('../config.php');

//Check if user is admin

if(!isset($\_SESSION["admin"]) && $\_SESSION["admin"] !== true) {

header("location: home.php");

exit;

}

?>

<style>

body {text-align:center;}

</style>

<SCRIPT LANGUAGE="JavaScript">

function copyToClipboard(text) {

window.prompt("Copy to clipboard: Ctrl+C, Enter", text);

}

</SCRIPT>

<div class="page-header">

<h1>Survey Creation Success</h1>

</div>

<?php

// Decode the survey object

$obj = json\_decode($\_COOKIE["survey"]);

////Output example of decoded object

//echo "Survey&nbspTitle: " , $obj->title , "<br>";

//echo "Description: " , $obj->desc , "<br><br>";

//

//$counter = 0; //Counter for question number

//foreach ($obj->qstnArr as $qstn) {

// $counter++;

// echo $qstn->qstn , "<br>";

// foreach ($qstn->answers as $ans) {

// echo '<input type="radio" name=q' . $counter . ' value=' . $ans . '>' . $ans . "<br>";

// }

// echo "<br>";

//}

//Primary keys for table insertions are all auto-incremented

//Insert survey title and description into database as well as unique id identifier

$string = bin2hex(random\_bytes(16)); //Create random unique hash/string

$query = "INSERT INTO `entity\_survey` (title, description, unique\_id) VALUES ('$obj->title', '$obj->desc', '$string')";

mysqli\_query($link,$query); //Input query

$surveyID = mysqli\_insert\_id($link);

$adminID = $\_SESSION["id"];

//Insert xref\_admin\_survey entry

$query = "INSERT INTO `xref\_user\_survey` (user\_id, survey\_id) VALUES ('$adminID', '$surveyID')";

mysqli\_query($link, $query);

//For each question

foreach ($obj->qstnArr as $qstn) {

//Insert question into database

$query = "INSERT INTO `entity\_question` (question) VALUES ('$qstn->qstn')";

mysqli\_query($link, $query);

$qstnID = mysqli\_insert\_id($link);

//Insert xref\_survey\_question entry

$query = "INSERT INTO `xref\_survey\_question` (survey\_id, question\_id) VALUES ('$surveyID', '$qstnID')";

mysqli\_query($link, $query);

//For each answer to a question

foreach ($qstn->answers as $ans) {

//Insert answer into database

$query = "INSERT INTO `entity\_answer` (answer) VALUES ('$ans')";

mysqli\_query($link, $query);

$ansID = mysqli\_insert\_id($link);

//Insert xref\_question\_answer entry

$query = "INSERT INTO `xref\_question\_answer` (question\_id, answer\_id) VALUES ('$qstnID', '$ansID')";

mysqli\_query($link, $query);

}

}

//Link to survey invitation

$copyLink = 'http://' . $\_SERVER['HTTP\_HOST'] . "/SurveyEngine/survey.php?id=" . $string;

//Output message

echo "<p>Survey has been created!</p>";

echo "<p>You can view your surveys at your account page.</p><br>";

echo "<p>Send the following survey invite link to any participants:</p>";

echo "<button id='reg' style='max-width:100px;' onclick='copyToClipboard(\"$copyLink\")'>Copy</button>";

mysqli\_close($link);

?>

**delete\_survey.php**

<style type="text/css">

body {text-align: center;}

</style>

<?php

$page\_title = 'Delete Survey';

include ('../includes/header.php');

require\_once "../config.php";

echo '<div class="page-header"><h2><b>Delete Survey</b></h2></div>';

// Check for a valid survey unique ID, through GET or POST:

if (isset($\_GET['id'])) {

$id = $\_GET['id'];

} elseif (isset($\_POST['id'])) { // Form submission.

$id = $\_POST['id'];

} else { // No valid ID, kill the script.

echo '<p class="error">This page has been accessed in error.</p>';

exit();

}

//Only if form has been submitted on the page

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

//Delete record if yes is chosen

if ($\_POST['sure'] == 'Yes') {

//Query to get survey ID

$q = "SELECT `survey\_id`, `unique\_id` FROM `survey`.`entity\_survey` AS `entity\_survey` WHERE `unique\_id` = '$id'";

$r = mysqli\_query($link, $q);

if (mysqli\_num\_rows($r) == 1) {

$row = mysqli\_fetch\_array($r, MYSQLI\_NUM);

$surveyID = $row[0]; //Get survey ID

} else {

echo '<h4 class="text-center" style="color:green;">The survey could not be found.</h4>';

echo '<a href="admin.php" class="btn btn-default">Go back to Admin Panel</button>';

}

//Array to hold answers, questions for later deletion

$ansArray = array();

$qstnArray = array();

//Delete xref\_survey\_question

$q2 = "SELECT `xref\_survey\_question`.`survey\_id`, `xref\_survey\_question`.`question\_id` FROM `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_question` AS `entity\_question`, `survey`.`entity\_survey` AS `entity\_survey` WHERE `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`survey\_id` = '$surveyID'";

$r2 = mysqli\_query($link, $q2);

$numQstn = mysqli\_num\_rows($r2);

for ($i=0; $i<$numQstn; $i++) {

mysqli\_data\_seek($r2, $i);

$rowQstn = $r2->fetch\_assoc();

$qstnID = $rowQstn['question\_id']; //Get question ID for each row

$q3 = "SELECT `xref\_question\_answer`.`question\_id`, `xref\_question\_answer`.`answer\_id` FROM `survey`.`xref\_question\_answer` AS `xref\_question\_answer`, `survey`.`entity\_question` AS `entity\_question`, `survey`.`entity\_answer` AS `entity\_answer` WHERE `xref\_question\_answer`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_question\_answer`.`answer\_id` = `entity\_answer`.`answer\_id` AND `xref\_question\_answer`.`question\_id` = '$qstnID'";

$r3 = mysqli\_query($link, $q3);

if (mysqli\_num\_rows($r3) > 0) {

$numAns = mysqli\_num\_rows($r3);

for ($j=0; $j<$numAns; $j++) {

mysqli\_data\_seek($r3, $j);

$rowAns = $r3->fetch\_assoc();

$ansID = $rowAns['answer\_id']; //Get question ID for each row

array\_push($ansArray, $ansID);

//Delete xref\_answer\_user rows

$query\_delete\_xref\_answer\_user = "DELETE FROM xref\_answer\_user WHERE answer\_id = '$ansID'";

mysqli\_query($link, $query\_delete\_xref\_answer\_user);

}

}

array\_push($qstnArray, $qstnID);

//Delete xref\_question\_answer rows

$query\_delete\_xref\_question\_answer = "DELETE FROM xref\_question\_answer WHERE question\_id = '$qstnID'";

mysqli\_query($link, $query\_delete\_xref\_question\_answer);

}

//Delete xref\_survey\_question rows

$query\_delete\_xref\_survey\_question = "DELETE FROM xref\_survey\_question WHERE survey\_id = '$surveyID'";

mysqli\_query($link, $query\_delete\_xref\_survey\_question);

//Delete entity\_answer

for ($i=0; $i<count($ansArray); $i++) {

$query\_delete\_answer = "DELETE FROM entity\_answer WHERE answer\_id = '$ansArray[$i]'";

mysqli\_query($link, $query\_delete\_answer);

}

//Delete entity\_question

for ($i=0; $i<count($qstnArray); $i++) {

$query\_delete\_question = "DELETE FROM entity\_question WHERE question\_id = '$qstnArray[$i]'";

mysqli\_query($link, $query\_delete\_question);

}

//Delete\_xref\_user\_survey rows

$query\_delete\_xref\_user\_survey = "DELETE FROM xref\_user\_survey WHERE survey\_id = '$surveyID'";

mysqli\_query($link, $query\_delete\_xref\_user\_survey);

//Delete entity\_survey row

$query\_delete\_survey = "DELETE FROM entity\_survey WHERE survey\_id = '$surveyID'";

mysqli\_query($link, $query\_delete\_survey);

echo '<h4 class="text-center" style="color:green;">The survey has been deleted.</h4>';

echo '<a href="admin.php" class="btn btn-default">Go back to Admin Panel</button>';

} else {

echo '<h4 class="text-center">The survey has NOT been deleted.</h4>';

echo '<a href="admin.php" class="btn btn-default">Go back to Admin Panel</button>';

}

}

//Show form

else {

$sql = "SELECT `survey\_id`, `unique\_id`, `title`, `description` FROM `survey`.`entity\_survey` AS `entity\_survey` WHERE `unique\_id` = '$id'";

$rs = mysqli\_query($link,$sql);

if (mysqli\_num\_rows($rs) == 1) {

//Get product information:

$row = mysqli\_fetch\_array ($rs, MYSQLI\_NUM);

// Display the record being deleted:

echo "<div class='text-center'>";

echo "<h3>Survey title: $row[2]</h3>

<h3>Description: $row[3]</h3>";

echo "<br>Are you sure you want to delete this survey?<br><br>";

// Create the form:

echo '<form action="delete\_survey.php" method="post">

<input type="radio" name="sure" value="Yes"> Yes<br>

<input type="radio" name="sure" value="No" checked="checked"> No<br><br>

<input type="submit" name="submit" value="Submit">

<input type="hidden" name="id" value="' . $id . '">

</form>';

echo "</div>";

} else {

echo '<p class="error">Error, something went wrong. Survey not found.</p>';

}

}

?>

**delete\_user.php**

<?php

$page\_title = 'Delete User';

include ('../includes/header.php');

require ('../config.php'); //Connect DB

//Check for a valid user ID, through GET or POST

if ( (isset($\_GET['id'])) && (is\_numeric($\_GET['id'])) ) {

$id = $\_GET['id'];

} elseif ( (isset($\_POST['id'])) && (is\_numeric($\_POST['id'])) ) {

$id = $\_POST['id'];

} else {

echo '<p class="error text-center">This page has been accessed in error.</p>';

exit;

}

?>

<div class="page-header">

<h1><b>Delete user</b></h1>

</div>

<?php

//Check if form has been submitted

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

if ($\_POST['sure'] == 'Yes') {

//Delete xrefs with foreign keys first

$q = "DELETE FROM xref\_answer\_user WHERE user\_id = $id";

mysqli\_query($link, $q);

$q2 = "DELETE FROM xref\_user\_survey WHERE user\_id = $id";

mysqli\_query($link, $q2);

//Delete user

$q3 = "DELETE FROM entity\_user WHERE user\_id = $id LIMIT 1";

$r = mysqli\_query ($link, $q3);

//If deletion is successful

if (mysqli\_affected\_rows($link) == 1) {

echo '<div class="text-center">

<p>The user has been deleted.</p>

<a href="view\_user.php" class="btn btn-secondary">Go back</button>

</div>';

} else {

echo '<div class="text-center">

<p class="error">The user could not be deleted due to a system error.</p>

<p>' . mysqli\_error($link) . '<br />Query: ' . $q . '</p>

<a href="view\_user.php" class="btn btn-secondary">Go back</button>

</div>';

}

} else {

echo '<div class="text-center">

<p>The user has NOT been deleted.</p>

<a href="view\_user.php" class="btn btn-secondary">Go back</button>

</div>';

}

}

//Show form

else {

// Retrieve the user's information:

$query = "SELECT \* FROM entity\_user WHERE user\_id= '$id'";

$rs = mysqli\_query ($link, $query);

//If query is successful

if (mysqli\_num\_rows($rs) == 1) {

// Get the user's information:

$row = mysqli\_fetch\_array ($rs, MYSQLI\_NUM);

$accType = ($row[3] == 0) ? "Admin" : "User"; //Get user account type

// Display the record being deleted:

echo "<div class='text-center'>

<h3>Username: $row[1]</h3>

<h3>Account type: $accType</h3>

</div>";

// Create the form:

echo '<div class="text-center">

Are you sure you want to delete this user?<br>

<form action="delete\_user.php" method="post">

<input type="radio" name="sure" value="Yes" /> Yes<br>

<input type="radio" name="sure" value="No" checked="checked" /> No<br><br>

<input type="submit" name="submit" value="Submit" />

<input type="hidden" name="id" value="' . $id . '" />

</form>

</div>';

} else { // Not a valid user ID.

echo '<p class="error">This page has been accessed in error.</p>';

}

}

mysqli\_close($link);

?>

**register\_admin.php**

<?php

$page\_title = 'Register Admin';

include ('../includes/header.php');

require\_once ('../config.php');

// Define variables and initialize with empty values

$username = $password = $confirm\_password = "";

$username\_err = $password\_err = $confirm\_password\_err = "";

// Processing form data when form is submitted

if($\_SERVER["REQUEST\_METHOD"] == "POST"){

// Validate username

if(empty(trim($\_POST["username"]))){

$username\_err = "Please enter a username.";

}

else{

// Prepare a select statement

$sql = "SELECT user\_id FROM entity\_user WHERE username = ?";

if($stmt = mysqli\_prepare($link, $sql)){

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "s", $param\_username);

// Set parameters

$param\_username = trim($\_POST["username"]);

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)){

/\* store result \*/

mysqli\_stmt\_store\_result($stmt);

if(mysqli\_stmt\_num\_rows($stmt) == 1){

$username\_err = "This username is already taken.";

} else{

$username = trim($\_POST["username"]);

}

} else{

echo "Oops! Something went wrong. Please try again later.";

}

// Close statement

mysqli\_stmt\_close($stmt);

}

}

// Validate password

if(empty(trim($\_POST["password"]))){

$password\_err = "Please enter a password.";

} else{

$password = trim($\_POST["password"]);

}

// Validate confirm password

if(empty(trim($\_POST["confirm\_password"]))){

$confirm\_password\_err = "Please confirm password.";

} else{

$confirm\_password = trim($\_POST["confirm\_password"]);

if(empty($password\_err) && ($password != $confirm\_password)){

$confirm\_password\_err = "Password does not match.";

}

}

// Check input errors before inserting in database

if(empty($username\_err) && empty($password\_err) && empty($confirm\_password\_err)){

$admin = 0; //Admin code

// Prepare an insert statement

$sql = "INSERT INTO entity\_user (username, password, type) VALUES (?, ?, $admin)";

if($stmt = mysqli\_prepare($link, $sql)){

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "ss", $param\_username, $param\_password);

// Set parameters

$param\_username = $username;

$param\_password = password\_hash($password, PASSWORD\_DEFAULT); // Creates a password hash

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)){

// Redirect to login page

header("location: admin.php");

} else{

echo "Something went wrong. Please try again later.";

}

// Close statement

mysqli\_stmt\_close($stmt);

}

}

// Close connection

mysqli\_close($link);

}

?>

<div id="accountForm">

<form action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]); ?>" method="post">

<div class="container">

<h2>Register Admin</h2>

<p>Please enter the following information to create an account</p>

<hr>

<div class="form-group <?php echo (!empty($username\_err)) ? 'has-error' : ''; ?>">

<label>Username</label>

<input type="text" name="username" class="form-control" value="<?php echo $username; ?>" placeholder="Enter username here">

<span class="help-block"><?php echo $username\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($password\_err)) ? 'has-error' : ''; ?>">

<label>Password</label>

<input type="password" name="password" class="form-control" value="<?php echo $password; ?>" placeholder="Enter password here">

<span class="help-block"><?php echo $password\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($confirm\_password\_err)) ? 'has-error' : ''; ?>">

<label>Confirm Password</label>

<input type="password" name="confirm\_password" class="form-control" value="<?php echo $confirm\_password; ?>" placeholder="Repeat password here">

<span class="help-block"><?php echo $confirm\_password\_err; ?></span>

</div>

<div class="form-group">

<br><button type="submit">Register</button>

</div>

</div>

</form>

</div>

</body>

</html>

**result.php**

<?php

$page\_title = 'Survey Results';

include ('../includes/header.php');

require\_once ('../config.php');

// Use GET to get the survey id

if (!isset($\_GET['id'])) {

echo '<h3 class="error text-center">Survey results not found</h3>';

exit;

} else {

$surveyID = $\_GET["id"];

}

$query = "SELECT `entity\_survey`.`title`, `entity\_survey`.`description`, `entity\_question`.`question` FROM `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_question` AS `entity\_question` WHERE `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `entity\_survey`.`unique\_id` = '$surveyID'";

$qstnRS = mysqli\_query($link, $query);

$numQstn = mysqli\_num\_rows($qstnRS);

//If survey result page is valid

if ($numQstn > 0) {

mysqli\_data\_seek($qstnRS, 0); //Seek data

$row = $qstnRS->fetch\_assoc(); //Fetch data

$title = $row['title']; //Get survey title

$desc = $row['description']; //Get survey description

}

else {

$error = "No results found.";

}

?>

<div class="page-header">

<h1>Survey Results</h1>

</div>

<div class="text-center" id="result">

<?php

if ($numQstn > 0) {

//Header for page

echo "<h3>Title: " . $title . "</h3>";

echo "<h4>Description: " . $desc . "</h4><br><br>";

//Table headers

echo "<table border='0' align='center'>";

echo "<tr><th>" . 'Question' . "</th>";

echo "<th>" . 'Answer' . "</th>";

echo "<th>" . 'Count' . "</th></tr>";

//Output question, answer, and tally for each answer from users

//For each question

for ($i=0; $i<$numQstn; $i++) {

mysqli\_data\_seek($qstnRS, $i); //Seek data

$rowQ = $qstnRS->fetch\_assoc(); //Fetch data

$qstn = $rowQ['question']; //Get question from row

//Output question

echo "<tr><td>" . $qstn . "</td>";

//Query answer for the question

$query = "SELECT `entity\_survey`.`unique\_id`, `entity\_question`.`question`, `entity\_answer`.`answer` FROM `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_question` AS `entity\_question`, `survey`.`xref\_question\_answer` AS `xref\_question\_answer`, `survey`.`entity\_answer` AS `entity\_answer` WHERE `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_question\_answer`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_question\_answer`.`answer\_id` = `entity\_answer`.`answer\_id` AND `entity\_survey`.`unique\_id` = '$surveyID' AND `entity\_question`.`question` = '$qstn'";

$ansRS = mysqli\_query($link, $query);

$numAns = mysqli\_num\_rows($ansRS);

//For each answer

for ($j=0; $j<$numAns; $j++) {

if ($j>0) { echo "<td></td>"; } //Output nothing in table for the question column

mysqli\_data\_seek($ansRS, $j); //Seek data

$rowA = $ansRS->fetch\_assoc(); //Fetch data

$answer = $rowA['answer']; //Get answer from row

//Output answer

echo "<td>" . $answer . "</td>";

$query = "SELECT `entity\_survey`.`unique\_id`, `entity\_user`.`username`, `entity\_question`.`question`, `entity\_answer`.`answer` FROM `survey`.`xref\_answer\_user` AS `xref\_answer\_user`, `survey`.`entity\_user` AS `entity\_user`, `survey`.`entity\_answer` AS `entity\_answer`, `survey`.`xref\_question\_answer` AS `xref\_question\_answer`, `survey`.`entity\_question` AS `entity\_question`, `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`xref\_user\_survey` AS `xref\_user\_survey` WHERE `xref\_answer\_user`.`user\_id` = `entity\_user`.`user\_id` AND `xref\_answer\_user`.`answer\_id` = `entity\_answer`.`answer\_id` AND `xref\_question\_answer`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_question\_answer`.`answer\_id` = `entity\_answer`.`answer\_id` AND `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_user\_survey`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_user\_survey`.`user\_id` = `entity\_user`.`user\_id` AND `entity\_survey`.`unique\_id` = '$surveyID' AND `entity\_question`.`question` = '$qstn' AND `entity\_answer`.`answer` = '$answer'";

$countRS = mysqli\_query($link, $query);

$count = mysqli\_num\_rows($countRS);

//Output count for answer

echo "<td>" . $count . "</td></tr>";

}

echo "<tr><td></td><td></td><td></td></tr>";

}

echo "</table>";

} else {

echo "<div id='error'>" . $error . "</div>";

}

?>

</div>

</body>

</html>

**view\_user.php**

<?php

$page\_title = 'View Users';

include ('../includes/header.php');

require ('../config.php'); //Connect DB

?>

<div class="page-header">

<h1><b>List of Users</b></h1>

</div>

<?php

//Number of records to show per page

$display = 5;

//Determine how many pages there are

if (isset($\_GET['p']) && is\_numeric($\_GET['p'])) { // Already been determined.

$pages = $\_GET['p'];

} else { //Need to determine pages

//Count number of records

$q = "SELECT COUNT(user\_id) FROM entity\_user";

$r = mysqli\_query ($link, $q);

$row = mysqli\_fetch\_array($r, MYSQLI\_NUM);

$records = $row[0];

//Calculate number of pages

if ($records > $display) { //More than 1 page

$pages = ceil ($records/$display);

} else {

$pages = 1;

}

}

// Determine where in the database to start returning results...

if (isset($\_GET['s']) && is\_numeric($\_GET['s'])) {

$start = $\_GET['s'];

} else {

$start = 0;

}

//Determine the sort

//Default is by registration date

$sort = (isset($\_GET['sort'])) ? $\_GET['sort'] : 'rd';

//Determine the sorting order

switch ($sort) {

case 'un':

$order\_by = 'username ASC';

break;

case 'rd':

$order\_by = 'registration\_date ASC';

break;

default:

$order\_by = 'registration\_date ASC';

$sort = 'rd';

break;

}

// Define the query:

$q = "SELECT `user\_id`, `username`, `registration\_date` FROM `survey`.`entity\_user` AS `entity\_user` ORDER BY $order\_by LIMIT $start, $display";

$r = mysqli\_query($link, $q); //Run the query

//Table header:

echo '<table align="center" style="width:30%;">

<tr>

<th width=5%></td>

<th width=5%><b><a href="view\_user.php?sort=un">Username</a></b></td>

<th width=5%><b><a href="view\_user.php?sort=rd">Date Registered</a></b></td>

</tr>';

// Fetch and print all the records....

while ($row = mysqli\_fetch\_array($r, MYSQLI\_ASSOC)) {

echo '<tr>

<td><a href="delete\_user.php?id=' . $row['user\_id'] . '">Delete</a></td>

<td>' . $row['username'] . '</td>

<td>' . $row['registration\_date'] . '</td>

</tr>';

}

echo '</table>';

mysqli\_free\_result ($r);

mysqli\_close($link);

//Pagination

if ($pages > 1) {

echo '<div class="text-center">';

echo '<br /><p>';

$current\_page = ($start/$display) + 1;

//If it's not the first page, make a previous button:

if ($current\_page != 1) {

echo '<a href="view\_user.php?s=' . ($start - $display) . '&p=' . $pages . '&sort=' . $sort . '">Previous</a> ';

}

//Make all the numbered pages:

for ($i = 1; $i <= $pages; $i++) {

if ($i != $current\_page) {

echo '<a href="view\_user.php?s=' . (($display \* ($i - 1))) . '&p=' . $pages . '&sort=' . $sort . '">' . $i . '</a> ';

} else {

echo $i . ' ';

}

}

//If it's not the last page, make a next button:

if ($current\_page != $pages) {

echo '<a href="view\_user.php?s=' . ($start + $display) . '&p=' . $pages . '&sort=' . $sort . '">Next</a>';

}

echo '</p>';

echo '</div>';

}

?>

**header.php**

<?php

session\_start();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title><?php echo $page\_title; ?></title>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">

<link href="https://fonts.googleapis.com/css?family=Roboto+Slab" rel="stylesheet">

<link href="/SurveyEngine/stylesheet.css" rel="stylesheet">

<script type="text/javascript" src="/SurveyEngine/scripts/survey.js"></script>

</head>

<body>

<nav class="navbar navbar-default">

<div class="container-fluid">

<ul class="nav navbar-nav">

<li class="active"><a href="/SurveyEngine/home.php"><span class="glyphicon glyphicon-home"></span> Home</a></li>

</ul>

<ul class="nav navbar-nav navbar-right">

<?php

if(!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true) {

echo "<li><a href='/SurveyEngine/register.php'><span class='glyphicon glyphicon-user'></span> Register</a></li>";

echo "<li><a href='/SurveyEngine/login.php'><span class='glyphicon glyphicon-log-in'></span> Log in</a></li>";

} else {

if (isset($\_SESSION['admin']) && $\_SESSION['admin'] == true) {

echo "<li><a href='/SurveyEngine/admin/admin.php'><span class='glyphicon glyphicon-user'></span> Account</a></li>";

} else {

echo "<li><a href='/SurveyEngine/account.php'><span class='glyphicon glyphicon-user'></span> Account</a></li>";

}

echo "<li><a href='/SurveyEngine/logout.php'><span class='glyphicon glyphicon-log-out'></span> Log Out</a></li>";

}

?>

</ul>

</div>

</nav>

**create.js**

//Question class

class Question {

constructor() {

this.qstn; //Question

this.numAns = 2; //Number of answers

this.answers = []; //Store answers

}

};

//Survey class

class Survey {

constructor() {

this.title = "";

this.desc = "";

this.numQstn = 0; //Number of questions

this.qstnArr = []; //Array of question objects

}

//Get number of questions from survey

getQ() {

return this.numQstn;

}

//Set title

setTitle(name) {

this.title = name;

}

//Set description

setDesc(detail) {

this.desc = detail;

}

//Add question

addQ() {

this.numQstn++; //Increase question count

this.qstnArr.push(new Question()); //Add new question to question array

}

//Remove question from survey

delQ() {

this.numQstn--; //Decrease question count

this.qstnArr.pop(); //Delete last question in array

}

};

//Add question to survey form

function addQstn() {

var numQ = srvy.getQ(); //Number of questions in survey

numQ++; //Increase count

//Add new question element to html page

var q = qTemp.cloneNode(true);

q.id = "qstn" + numQ;

q.querySelector(".txt").innerHTML = "Question " + numQ;

q.querySelector("#addBtn").setAttribute("onclick", "addAns(" + numQ + ")");

q.querySelector("#delBtn").setAttribute("onclick", "delAns(" + numQ + ")");

srvy.addQ(); //Add question to survey

document.getElementById("survey").appendChild(q);

};

//Delete question from survey from

function delQstn() {

//Only allowed if number of question on survey form exceeds 1

if (srvy.numQstn > 1) {

//Get node and destroy

let node = document.getElementById("qstn" + srvy.numQstn);

if (node.parentNode) {

node.parentNode.removeChild(node);

}

srvy.delQ(); //Delete question from survey

}

};

//Add an answer to a question (q = question number)

function addAns(q) {

q = q-1; //Convert to element value

//Increase number of answers in the question

srvy.qstnArr[q].numAns++;

//Get answers class from specified question #

var temp = document.querySelector("#qstn" + (q+1) + " > #answers");

var ans = aTemp.cloneNode(true);

ans.id = "ans" + srvy.qstnArr[q].numAns;

ans.placeholder = "Answer " + srvy.qstnArr[q].numAns;

temp.appendChild(ans); //Add answer

};

//Delete an answer to a question

function delAns(q) {

if (srvy.qstnArr[q-1].numAns > 2) {

var cont = document.querySelector("#qstn" + q + " > #answers");

var ans = document.querySelector("#qstn" + q + " #ans" + srvy.qstnArr[q-1].numAns);

cont.removeChild(ans);

srvy.qstnArr[q-1].numAns--;

}

};

//Display error message when required fields are not shown

function showErr() {

var err = document.getElementById("error");

err.style.display = "block"; //Toggle css display on

err.innerHTML = "\*All fields required"; //Add & display error message

}

//When create survey button is clicked, save the survey values

function create() {

var error = false;

var sName = document.getElementById("surveyTitle").value.trim(); //Get survey title

var sDesc = document.getElementById("surveyDesc").value.trim(); //Get survey description

//Check for empty input text

if (sName === "" || sDesc === "") {

error = true;

if (sName === "") {

var titleErr = document.getElementById("surveyTitle");

titleErr.style.border = "2px solid red";

}

if (sDesc === "") {

var descErr = document.getElementById("surveyDesc");

descErr.style.border = "2px solid red";

}

}

else {

srvy.setTitle(sName); //Set survey title

srvy.setDesc(sDesc); //Set survey description

}

//Loop every question

for (var i=0; i<srvy.numQstn; i++) {

//Get and set question text

var q = document.querySelector("#qstn" + (i+1) + " > #qCtnt").value.trim();

//Check for empty input text

if (q === "") {

error = true;

var qErr = document.querySelector("#qstn" + (i+1) + " > #qCtnt");

qErr.style.border = "2px solid red";

}

else {

srvy.qstnArr[i].qstn = q; //Set question text

}

//Loop every answer

for (var j=0; j<srvy.qstnArr[i].numAns; j++) {

//Get and set answer text

var a = document.querySelector("#qstn" + (i+1) + " #ans" + (j+1)).value.trim();

//Check for empty input text

if (a === "") {

error = true;

var aErr = document.querySelector("#qstn" + (i+1) + " #ans" + (j+1));

aErr.style.border = "2px solid red";

}

else {

srvy.qstnArr[i].answers.push(a); //Set answer text

}

}

}

//If there are no errors, save data

if (!error) {

var str = JSON.stringify(srvy); //Convert object to JSON string

//localStorage.setItem("Survey", str); //Save to local storage

document.cookie = "survey=" + str; //Save to cookie

window.location = "/SurveyEngine/admin/create.php"; //Redirect page

}

//Else show error message

else {

showErr();

//Delete answer values that were successfuly pushed previously

for (var i=0; i<srvy.numQstn; i++) {

for (var j=0; j<srvy.qstnArr[i].numAns; j++) {

srvy.qstnArr[i].answers.pop();

}

}

}

//Remove error markings if corrected by user

if (sName !== "") { //Title

document.getElementById("surveyTitle").style.border = "none";

}

if (sDesc !== "") { //Description

document.getElementById("surveyDesc").style.border = "none";

}

for (var i=0; i<srvy.numQstn; i++) { //Question

var q = document.querySelector("#qstn" + (i+1) + " > #qCtnt");

if (q.value !== "") {

q.style.border = "none";

}

for (var j=0; j<srvy.qstnArr[i].numAns; j++) { //Answer

var a = document.querySelector("#qstn" + (i+1) + " #ans" + (j+1));

if (a.value !== "") {

a.style.border = "none";

}

}

}

};

//Run when page loads

window.onload = function () {

//In order to add default questions/answers later

qTemp = document.getElementById("qstn1"); //Default question

aTemp = document.getElementById("ans1"); //Default answer

qTemp.parentNode.removeChild(qTemp); //Remove

//Add first question

addQstn();

};

var srvy = new Survey(); //Create survey

var qTemp; //Store question

var aTemp; //Store answer

**survey.js**

function validForm() {

//Flag to check if all questions have been answered

var valid = true;

//For each question

var questions = document.querySelectorAll(".qstn");

for (var i=0; i<questions.length; i++) {

//Flag to check if answer has been checked

var answered = false;

//For each answer

var ans = questions[i].querySelectorAll("input[type='radio']");

for (var j=0; j<ans.length; j++) {

//Check if radio box is checked

if (ans[j].checked === true) {

console.log(ans[j]);

answered = true;

}

}

//If a question is not answered, output error sign for that question

if (!answered) {

var qErr = document.getElementById("q"+(i+1));

qErr.style.border = "2px solid red";

//Survey is incomplete

valid = false;

}

if (answered) {

var q = document.getElementById("q"+(i+1));

q.style.border = "none";

}

}

//Submit form if valid

if (valid) {

var form = document.getElementById("survey");

form.submit();

}

else {

//Display an error

var err = document.getElementById("error");

err.style.display = "block"; //Toggle css display on

err.innerHTML = "\*Please answer all questions";

}

}

**account.php**

<?php

$page\_title = 'Account';

include ('./includes/header.php');

require\_once ('./config.php');

//Check if user or admin, redirect accordingly

if(!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true){

header("location: login.php");

exit;

} elseif (isset($\_SESSION["admin"]) && $\_SESSION["admin"] === true) {

header("location: admin.php");

exit;

} else {

$userID = $\_SESSION["id"]; //Get user ID

}

?>

<style type="text/css">

body {text-align: center;}

</style>

<div class="page-header">

<h1>Welcome, <b><?php echo htmlspecialchars($\_SESSION["username"]); ?></b></h1>

</div>

<div class="text-center">

<h3>Surveys taken:</h3>

<?php

//Surveys taken

$query = "SELECT `entity\_survey`.`title`, `entity\_survey`.`description` FROM `survey`.`xref\_user\_survey` AS `xref\_user\_survey`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_user` AS `entity\_user` WHERE `xref\_user\_survey`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_user\_survey`.`user\_id` = `entity\_user`.`user\_id` AND `entity\_user`.`user\_id` = '$userID'";

$rs = mysqli\_query($link, $query); //Get result set

$numSurvey = mysqli\_num\_rows($rs); //Get number of surveys taken

if ($numSurvey > 0) {

echo "<table border='1' align='center'>";

echo "<tr><th>".'Survey Title'."</th>";

echo "<th>".'Description'."</th></tr>";

while($re = mysqli\_fetch\_array($rs, MYSQLI\_BOTH)){

echo "<tr><td>".$re['title']."</td>";

echo "<td>".$re['description']."</td></tr>";

}

echo "</table>";

} else {

echo "No surveys found";

}

?>

<div class="settings">

<h3><span class="glyphicon glyphicon-cog"></span> Settings</h3>

<a href="reset\_password.php" class="btn btn-warning">Reset password</a>

</div>

</div>

</body>

</html>

**config.php**

<?php

//This file contains the database access information.

//This file also establishes a connection to MySQL and selects the database.

//Database credentials

define('DB\_HOST', 'localhost');

define('DB\_USER', 'root');

define('DB\_PASSWORD', '');

define('DB\_NAME', 'survey');

//Attempt to connect to MySQL database

$link = mysqli\_connect(DB\_HOST, DB\_USER, DB\_PASSWORD, DB\_NAME);

//Check connection

if ($link === false){

die ("Could not connect to MySQL: " . mysqli\_connect\_error());

}

//Set encoding

mysqli\_set\_charset($link, 'utf8');

?>

**home.php**

<?php

$page\_title = 'Home';

include ('./includes/header.php');

?>

<div class="text-center">

<h2>Survey Engine Website</h2><br>

<p>Admins can create surveys and view their results</p>

<p>Users can take surveys sent to them through a unique survey link</p>

</div>

</body>

</html>

**login.php**

<?php

$page\_title = 'Login';

include ('./includes/header.php');

require\_once ('./config.php');

// Check if the user is already logged in, if yes then redirect him to welcome page

if(isset($\_SESSION["loggedin"]) && $\_SESSION["loggedin"] === true) {

if(isset($\_SESSION["admin"]) && $\_SESSION["admin"] === true) {

header("location: admin.php");

} else {

header("location: account.php");

}

exit;

}

// Define variables and initialize with empty values

$username = $password = "";

$username\_err = $password\_err = "";

// Processing form data when form is submitted

if($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Check if username is empty

if(empty(trim($\_POST["username"]))) {

$username\_err = "Please enter username.";

} else {

$username = trim($\_POST["username"]);

}

// Check if password is empty

if(empty(trim($\_POST["password"]))) {

$password\_err = "Please enter your password.";

} else {

$password = trim($\_POST["password"]);

}

// Validate credentials

if(empty($username\_err) && empty($password\_err)) {

// Prepare a select statement

$sql = "SELECT user\_id, username, password, type FROM entity\_user WHERE username = ?";

if($stmt = mysqli\_prepare($link, $sql)) {

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "s", $param\_username);

// Set parameters

$param\_username = $username;

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)) {

// Store result

mysqli\_stmt\_store\_result($stmt);

// Check if username exists, if yes then verify password

if(mysqli\_stmt\_num\_rows($stmt) == 1) {

// Bind result variables

mysqli\_stmt\_bind\_result($stmt, $id, $username, $hashed\_password, $type);

if(mysqli\_stmt\_fetch($stmt)) {

if(password\_verify($password, $hashed\_password)) {

// Password is correct, so start a new session

session\_start();

// Store data in session variables

$\_SESSION["loggedin"] = true;

$\_SESSION["id"] = $id;

$\_SESSION["username"] = $username;

if(isset($\_SESSION["redirect\_to"])) {

header("location: survey.php?id=" . $\_SESSION["redirect\_to"]);

exit;

}

if($type == 0) { //If admin

$\_SESSION["admin"] = true;

//Redirect to admin page

header("location: admin/admin.php");

} else { //If regular user

$\_SESSION["admin"] = false;

// Redirect user to welcome page

header("location: account.php");

}

} else {

// Display an error message if password is not valid

$password\_err = "The password you entered was not valid.";

}

}

} else {

// Display an error message if username doesn't exist

$username\_err = "No account found with that username.";

}

} else {

echo "Oops! Something went wrong. Please try again later.";

}

// Close statement

mysqli\_stmt\_close($stmt);

}

}

// Close connection

mysqli\_close($link);

}

if(isset($\_SESSION["redirect\_to"])) {

echo "<p style='text-align:center;'>You have been invited to take a survey. Please login or register.</p>";

}

?>

<div id="accountForm">

<form action="" method="post">

<div class="container">

<h2>Login</h2>

<hr>

<div class="form-group <?php echo (!empty($username\_err)) ? 'has-error' : ''; ?>" >

<label>Username</label>

<input type="text" name="username" class="form-control" value="<?php echo $username; ?>" placeholder="Enter Username">

<span class="help-block"><?php echo $username\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($password\_err)) ? 'has-error' : ''; ?>" >

<label>Password</label>

<input type="password" name="password" class="form-control" placeholder="Enter Password">

<span class="help-block"><?php echo $password\_err; ?></span>

</div>

<p>No account? <a href="./register.php">Register here.</a></p>

<div class="form-group" style="padding-top:25px">

<button type="submit">Login</button>

</div>

</div>

</form>

</div>

</body>

</html>

**logout.php**

<?php

//Initialize session

session\_start();

//Unset all session variables

$\_SESSION = array();

//Destroy session

session\_destroy();

//Redirect

header("location: home.php");

exit;

?>

**register.php**

<?php

$page\_title = 'Register';

include ('./includes/header.php');

require\_once ('./config.php');

// Define variables and initialize with empty values

$username = $password = $confirm\_password = "";

$username\_err = $password\_err = $confirm\_password\_err = "";

//Username regex must consist of 3-15 alphanumeric characters

function validate\_user($user) {

if( !preg\_match("/^[a-z0-9\_-]{3,15}$/", $user) ) {

return false;

}

return true;

}

//Password regex must be 6-20 characters including: a digit, a lower case char, a upper case char, a special symbol

function validate\_password($pass) {

if( !preg\_match("/^((?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[!@#$%^&\*]).{6,20})$/", $pass) ) {

return false;

}

return true;

}

// Processing form data when form is submitted

if($\_SERVER["REQUEST\_METHOD"] == "POST"){

// Validate username

if(empty(trim($\_POST["username"]))){

$username\_err = "Please enter a username.";

}

else if (!validate\_user(trim($\_POST["username"]))) {

$username\_err = "Username must be 3 to 15 alphanumeric characters";

}

else{

// Prepare a select statement

$sql = "SELECT user\_id FROM entity\_user WHERE username = ?";

if($stmt = mysqli\_prepare($link, $sql)){

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "s", $param\_username);

// Set parameters

$param\_username = trim($\_POST["username"]);

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)){

/\* store result \*/

mysqli\_stmt\_store\_result($stmt);

if(mysqli\_stmt\_num\_rows($stmt) == 1){

$username\_err = "This username is already taken.";

} else{

$username = trim($\_POST["username"]);

}

} else{

echo "Oops! Something went wrong. Please try again later.";

}

// Close statement

mysqli\_stmt\_close($stmt);

}

}

// Validate password

if(empty(trim($\_POST["password"]))){

$password\_err = "Please enter a password.";

} elseif(strlen(trim($\_POST["password"])) < 6){

$password\_err = "Password must have at least 6 characters.";

} elseif(!validate\_password(trim($\_POST["password"]))) {

$password\_err = "Password must have each of the following: digit, lowercase character, uppercase character, and a special symbol.";

} else{

$password = trim($\_POST["password"]);

}

// Validate confirm password

if(empty(trim($\_POST["confirm\_password"]))){

$confirm\_password\_err = "Please confirm password.";

} else{

$confirm\_password = trim($\_POST["confirm\_password"]);

if(empty($password\_err) && ($password != $confirm\_password)){

$confirm\_password\_err = "Password does not match.";

}

}

// Check input errors before inserting in database

if(empty($username\_err) && empty($password\_err) && empty($confirm\_password\_err)){

$user = 1; //Regular user code

// Prepare an insert statement

$sql = "INSERT INTO entity\_user (username, password, type) VALUES (?, ?, $user)";

if($stmt = mysqli\_prepare($link, $sql)){

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "ss", $param\_username, $param\_password);

// Set parameters

$param\_username = $username;

$param\_password = password\_hash($password, PASSWORD\_DEFAULT); // Creates a password hash

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)){

// Redirect to login page

header("location: login.php");

} else{

echo "Something went wrong. Please try again later.";

}

// Close statement

mysqli\_stmt\_close($stmt);

}

}

// Close connection

mysqli\_close($link);

}

?>

<div id="accountForm">

<form action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]); ?>" method="post">

<div class="container">

<h2>Register</h2>

<p>Please enter the following information to create an account</p>

<hr>

<div class="form-group <?php echo (!empty($username\_err)) ? 'has-error' : ''; ?>">

<label>Username</label>

<input type="text" name="username" class="form-control" value="<?php echo $username; ?>" placeholder="Enter username here">

<span class="help-block"><?php echo $username\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($password\_err)) ? 'has-error' : ''; ?>">

<label>Password</label>

<input type="password" name="password" class="form-control" value="<?php echo $password; ?>" placeholder="Enter password here">

<span class="help-block"><?php echo $password\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($confirm\_password\_err)) ? 'has-error' : ''; ?>">

<label>Confirm Password</label>

<input type="password" name="confirm\_password" class="form-control" value="<?php echo $confirm\_password; ?>" placeholder="Repeat password here">

<span class="help-block"><?php echo $confirm\_password\_err; ?></span>

</div>

<div class="form-group">

<br><button type="submit">Register</button>

</div>

</div>

</form>

</div>

</body>

</html>

**reset\_password.php**

<?php

$page\_title = 'Reset Password';

include ('./includes/header.php');

require\_once ('./config.php');

// Check if the user is logged in, if not then redirect to login page

if(!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true) {

header("location: login.php");

exit;

}

// Define variables and initialize with empty values

$new\_password = $confirm\_password = "";

$new\_password\_err = $confirm\_password\_err = "";

// Processing form data when form is submitted

if($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Validate new password

if(empty(trim($\_POST["new\_password"]))) {

$new\_password\_err = "Please enter the new password.";

} elseif(strlen(trim($\_POST["new\_password"])) < 6) {

$new\_password\_err = "Password must have atleast 6 characters.";

} else {

$new\_password = trim($\_POST["new\_password"]);

}

// Validate confirm password

if(empty(trim($\_POST["confirm\_password"]))) {

$confirm\_password\_err = "Please confirm the password.";

} else {

$confirm\_password = trim($\_POST["confirm\_password"]);

if(empty($new\_password\_err) && ($new\_password != $confirm\_password)) {

$confirm\_password\_err = "Password did not match.";

}

}

// Check input errors before updating the database

if(empty($new\_password\_err) && empty($confirm\_password\_err)) {

// Prepare an update statement

$sql = "UPDATE entity\_user SET password = ? WHERE user\_id = ?";

if($stmt = mysqli\_prepare($link, $sql)) {

// Bind variables to the prepared statement as parameters

mysqli\_stmt\_bind\_param($stmt, "si", $param\_password, $param\_id);

// Set parameters

$param\_password = password\_hash($new\_password, PASSWORD\_DEFAULT);

$param\_id = $\_SESSION["id"];

// Attempt to execute the prepared statement

if(mysqli\_stmt\_execute($stmt)) {

// Password updated successfully. Destroy the session, and redirect to login page

session\_destroy();

header("location: login.php");

exit();

} else {

echo "Oops! Something went wrong. Please try again later.";

}

}

// Close statement

mysqli\_stmt\_close($stmt);

}

// Close connection

mysqli\_close($link);

}

?>

<div id="accountForm">

<div class="container">

<h2>Reset Password</h2>

<hr>

<form action="<?php echo htmlspecialchars($\_SERVER["PHP\_SELF"]); ?>" method="post">

<div class="form-group <?php echo (!empty($new\_password\_err)) ? 'has-error' : ''; ?>">

<label>New Password</label>

<input type="password" name="new\_password" class="form-control" value="<?php echo $new\_password; ?>">

<span class="help-block"><?php echo $new\_password\_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($confirm\_password\_err)) ? 'has-error' : ''; ?>">

<label>Confirm Password</label>

<input type="password" name="confirm\_password" class="form-control">

<span class="help-block"><?php echo $confirm\_password\_err; ?></span>

</div>

<div class="form-group">

<input type="submit" class="btn btn-primary" value="Submit">

<?php

if (isset($\_SESSION['admin']) && $\_SESSION['admin'] == true)

$string = "/SurveyEngine/admin/admin.php";

else

$string = "/SurveyEngine/account.php";

?>

<a class="btn btn-link" href="<?php echo $string ?>">Cancel</a>

</div>

</form>

</div>

</div>

</body>

</html>

**stylesheet.css**

body {

font-family: 'Roboto Slab', serif;

padding-bottom: 100px;

}

\* {

box-sizing: border-box

}

#title {

margin: auto;

}

input[type=text], input[type=password], input[type=box], textarea {

width: 100%;

padding: 15px;

margin: 5px 0 10px 0;

display: inline-block;

border: none;

background: #f1f1f1;

}

input[type=text]:focus, input[type=password]:focus, textarea:focus {

background-color: #ddd;

outline: none;

}

hr {

border: 1px solid #f1f1f1;

margin-bottom: 30px;

}

button {

background-color: #4CAF50;

color: white;

padding: 15px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 100%;

opacity: 0.9;

border-radius: 20px;

}

#botBtns {

position: relative;

margin: auto;

top: 0;

right: 0;

bottom: 0;

left: 0;

max-width: 1000px;

padding: 50px;

text-align: center;

}

.submitBtns {

padding: 200px;

}

.createBtn {

padding-top: 5%;

margin: auto;

max-width: 600px;

}

#addQBtn {

max-width: 300px;

background-color: #63a2f8;

margin: auto;

}

#delQBtn {

max-width: 300px;

background-color: #fa647a;

margin: auto;

}

#addBtn {

width: 100px;

background-color: #63a2f8;

}

#delBtn {

width: 100px;

background-color: #fa647a;

}

button:hover {

opacity: 1;

}

.container {

padding: 15px;

max-width: 450px;

}

.container2 {

padding: 16px;

text-align: center;

background-color: #f1f1f1;

}

#accountForm {

position: relative;

margin: auto;

padding: 20px;

top: 0;

right: 0;

bottom: 50px;

left: 0;

max-width: 500px;

border-radius: 3px;

border: 1px solid #ccc;

}

.container-fluid {

padding-left: 25%;

padding-right: 25%;

}

#survey {

position: relative;

margin: 0 auto;

max-width: 50%;

padding: 50px;

border: 2px solid gray;

border-style: groove;

border-radius: 15px;

}

.questions {

border-style: groove;

border-color: gray;

border-width: 1px;

border-radius: 5px;

padding: 35px 35px;

}

#error {

color: red;

border-style: solid;

display: none;

padding: 10px;

margin-bottom: 30px;

}

.settings {

padding-top: 10%;

padding-bottom: 5%;

}

.text-center {

margin-top: 5%;

padding-bottom: 1%;

}

table {

border-collapse: collapse;

width: 40%;

}

th, td {

text-align: center;

height: 50px;

}

tr:nth-child(even) {background-color: #f2f2f2;}

#reg {

background-color: #4CAF50;

color: white;

border: none;

line-height: 5px;

cursor: pointer;

width: 50%;

opacity: 0.9;

border-radius: 20px;

}

#radioDiv {

margin: 0 auto;

width: auto;

text-align: left;

display: table;

}

input[type="radio"] {

margin-top: -1px;

vertical-align: middle;

margin-right: 5px;

}

.error {

color: red;

}

.page-header {

text-align: center;

}

body {

font-family: 'Roboto Slab', serif;

padding-bottom: 100px;

}

\* {

box-sizing: border-box

}

#title {

margin: auto;

}

input[type=text], input[type=password], input[type=box], textarea {

width: 100%;

padding: 15px;

margin: 5px 0 10px 0;

display: inline-block;

border: none;

background: #f1f1f1;

}

input[type=text]:focus, input[type=password]:focus, textarea:focus {

background-color: #ddd;

outline: none;

}

hr {

border: 1px solid #f1f1f1;

margin-bottom: 30px;

}

button {

background-color: #4CAF50;

color: white;

padding: 15px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 100%;

opacity: 0.9;

border-radius: 20px;

}

#botBtns {

position: relative;

margin: auto;

top: 0;

right: 0;

bottom: 0;

left: 0;

max-width: 1000px;

padding: 50px;

text-align: center;

}

.submitBtns {

padding: 200px;

}

.createBtn {

padding-top: 5%;

margin: auto;

max-width: 600px;

}

#addQBtn {

max-width: 300px;

background-color: #63a2f8;

margin: auto;

}

#delQBtn {

max-width: 300px;

background-color: #fa647a;

margin: auto;

}

#addBtn {

width: 100px;

background-color: #63a2f8;

}

#delBtn {

width: 100px;

background-color: #fa647a;

}

button:hover {

opacity: 1;

}

.container {

padding: 15px;

max-width: 450px;

}

.container2 {

padding: 16px;

text-align: center;

background-color: #f1f1f1;

}

#accountForm {

position: relative;

margin: auto;

padding: 20px;

top: 0;

right: 0;

bottom: 50px;

left: 0;

max-width: 500px;

border-radius: 3px;

border: 1px solid #ccc;

}

.container-fluid {

padding-left: 25%;

padding-right: 25%;

}

#survey {

position: relative;

margin: 0 auto;

max-width: 50%;

padding: 50px;

border: 2px solid gray;

border-style: groove;

border-radius: 15px;

}

.questions {

border-style: groove;

border-color: gray;

border-width: 1px;

border-radius: 5px;

padding: 35px 35px;

}

#error {

color: red;

border-style: solid;

display: none;

padding: 10px;

margin-bottom: 30px;

}

.settings {

padding-top: 10%;

padding-bottom: 5%;

}

.text-center {

margin-top: 5%;

padding-bottom: 1%;

}

table {

border-collapse: collapse;

width: 40%;

}

th, td {

text-align: center;

height: 50px;

}

tr:nth-child(even) {background-color: #f2f2f2;}

#reg {

background-color: #4CAF50;

color: white;

border: none;

line-height: 5px;

cursor: pointer;

width: 50%;

opacity: 0.9;

border-radius: 20px;

}

#radioDiv {

margin: 0 auto;

width: auto;

text-align: left;

display: table;

}

input[type="radio"] {

margin-top: -1px;

vertical-align: middle;

margin-right: 5px;

}

.error {

color: red;

}

.page-header {

text-align: center;

}

**submit.php**

<?php

$page\_title = 'Survey Submission';

include ('./includes/header.php');

require\_once ('./config.php');

if(!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true) {

header("location: login.php");

exit;

}

$userID = $\_SESSION["id"]; //Store user ID

$survID = $\_SESSION["surveyID"]; //Store survey ID

//Query server to see if user has already taken survey or not

$query = "SELECT `entity\_user`.`user\_id`, `entity\_survey`.`unique\_id` FROM `survey`.`xref\_user\_survey` AS `xref\_user\_survey`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_user` AS `entity\_user` WHERE `xref\_user\_survey`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_user\_survey`.`user\_id` = `entity\_user`.`user\_id` AND `entity\_user`.`user\_id` = '$userID' AND `entity\_survey`.`unique\_id` = '$survID'";

$rs = mysqli\_query($link, $query); //Result set to see if user is connected with this particular survey

$nRows = mysqli\_num\_rows($rs); //Number of rows from query should return 0 if survey hasn't been taken

//If user has already taken survey, create error message

if ($nRows > 0) {

$error = "You have already taken this survey.";

echo "<h3 class='error text-center'>" . $error . "</h3>";

exit;

}

//Otherwise, if post method was utilized, read form results

else if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

//Query survey PID, question\_id

$query = "SELECT `entity\_survey`.`survey\_id`, `entity\_question`.`question\_id`, `entity\_question`.`question` FROM `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_question` AS `entity\_question` WHERE `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `entity\_survey`.`unique\_id` = '$survID'";

$qstnRS = mysqli\_query($link, $query); //Question result set

$numQstn = mysqli\_num\_rows($qstnRS); //Get number of questions

//Get survey primary key ID

mysqli\_data\_seek($qstnRS, 0); //Seek data

$row = $qstnRS->fetch\_assoc(); //Fetch data

$survPID = $row['survey\_id']; //Get survey\_id primary key to put in xref

//Get answer to each question and input into database using queries

for ($i=0; $i<$numQstn; $i++) {

mysqli\_data\_seek($qstnRS, $i); //Seek data

$row = $qstnRS->fetch\_assoc(); //Fetch data

$qstnID = $row["question\_id"]; //Get question id (Name value for answers for question in previous form)

$ansID = $\_POST['q'.$qstnID]; //Get answer id value number specified in previous form

//Input entry into xref\_answer\_user

$query = "INSERT INTO `xref\_answer\_user` (answer\_id, user\_id) VALUES ('$ansID', '$userID')";

mysqli\_query($link, $query);

}

//Input entry into xref\_user\_survey

$query = "INSERT INTO `xref\_user\_survey` (user\_id, survey\_id) VALUES ('$userID', '$survPID')";

mysqli\_query($link, $query);

}

?>

<div class="text-center">

<h3>Your survey has been submitted.</h3>

<h3>Thank you!</h3>

</div>

</body>

</html>

**survey.php**

<?php

//This file needs survey.js (located in header)

$page\_title = 'Take Survey';

include ('./includes/header.php');

require\_once ('./config.php');

if (!isset($\_GET['id'])) {

echo '<h2 class="error text-center">Error: Survey not found</h2>';

exit;

} elseif (!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true) {

$\_SESSION["redirect\_to"] = $\_GET['id'];

header("location: login.php");

exit;

} else {

$id = $\_GET["id"];

$\_SESSION["surveyID"] = $id;

}

//Query the survey id

$query = "SELECT `unique\_id`, `title`, `description` FROM `survey`.`entity\_survey` AS `entity\_survey` WHERE `unique\_id` = '$id'";

$rs = mysqli\_query($link, $query);

//Check if the query returned a row or if it nothing exists

if (mysqli\_num\_rows($rs) > 0) {

$row = $rs->fetch\_assoc();

$title = $row['title'];

$desc = $row['description'];

}

else {

echo '<h2 class="error text-center">Error: Survey not found</h2>';

exit;

}

?>

<h2 class="page-header">Survey</h2>

<form id="survey" action="submit.php" method="post">

<span id="error"></span>

<?php

//Output survey title and description

echo "<h2 id='title'>" . $title . "</h2><br>";

echo "<p id='desc'>" . $desc . "</p>";

//Get questions

$query = "SELECT `entity\_question`.`question\_id`, `entity\_question`.`question` FROM `survey`.`xref\_survey\_question` AS `xref\_survey\_question`, `survey`.`entity\_survey` AS `entity\_survey`, `survey`.`entity\_question` AS `entity\_question` WHERE `xref\_survey\_question`.`survey\_id` = `entity\_survey`.`survey\_id` AND `xref\_survey\_question`.`question\_id` = `entity\_question`.`question\_id` AND `entity\_survey`.`unique\_id` = '$id'";

$qstnRS = mysqli\_query($link, $query); //Question result set

$numQstn = mysqli\_num\_rows($qstnRS); //Get number of questions

//For each question

echo "<hr>";

for ($i=0; $i<$numQstn; $i++) {

$num = $i + 1; //Counter

//Output question

mysqli\_data\_seek($qstnRS, $i); //Seek row

$row = $qstnRS->fetch\_assoc(); //Fetch row

echo "<div class='qstn'> <p id='q$num'>" . $row["question"] . "<br><br>";

//Output answers to question

$qstnID = $row["question\_id"];

$ansQuery = "SELECT `entity\_answer`.`answer`, `entity\_question`.`question\_id`, `entity\_answer`.`answer\_id` FROM `survey`.`xref\_question\_answer` AS `xref\_question\_answer`, `survey`.`entity\_question` AS `entity\_question`, `survey`.`entity\_answer` AS `entity\_answer` WHERE `xref\_question\_answer`.`question\_id` = `entity\_question`.`question\_id` AND `xref\_question\_answer`.`answer\_id` = `entity\_answer`.`answer\_id` AND `entity\_question`.`question\_id` = $qstnID";

$ansRS = mysqli\_query($link, $ansQuery);

$numAns = mysqli\_num\_rows($ansRS);

//For each answer

for ($ans=0; $ans<$numAns; $ans++) {

$cntrAns = $ans + 1;

mysqli\_data\_seek($ansRS, $ans); //Seek row

$rowA = $ansRS->fetch\_assoc(); //Fetch row

$ansID = $rowA["answer\_id"];

echo "<input type='radio' name=q" . $qstnID . " id='q". $qstnID ."a". $ansID ."' value='" . $ansID . "'></input>

<label for='q". $qstnID ."a". $ansID ."'>" . $rowA["answer"] . "</label><br>";

}

echo "</div>";

echo "<hr>";

}

echo "</div>";

echo '<button onclick="validForm()" type="button">Submit</button>';

?>

</form>

</body>

</html>