

COIS 3420
Winter 2020
Final Project

Name: **Ahmed Ahmed & Mansur Mansur**

Live Web Link:

<https://loki.trentu.ca/~ahmedahmed/3420/project/index.php>

Credentials for User Account in your application with Data:

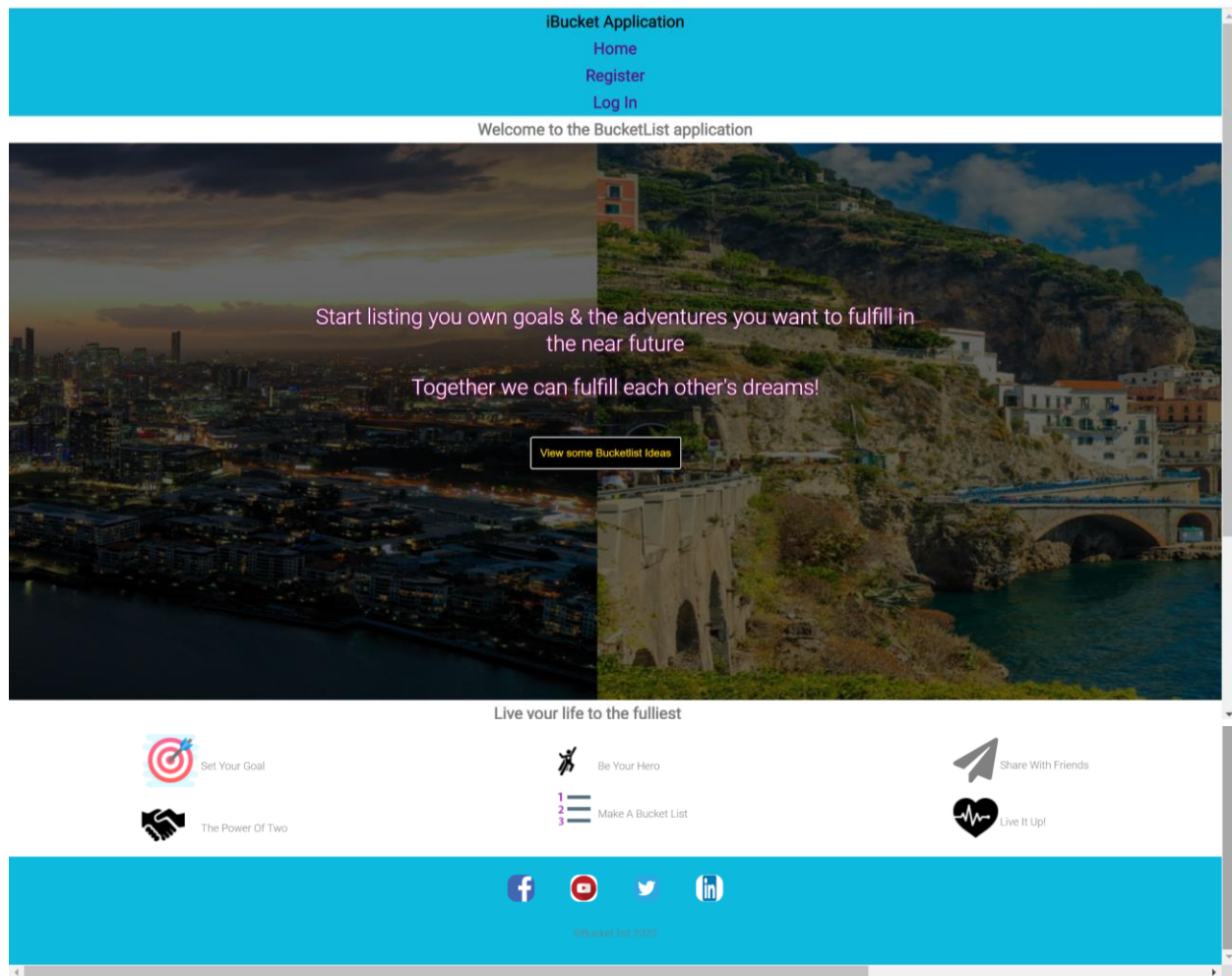
Username: Ahmed

Password: COIS3420h!

Functionality	/20
---------------	-----

Design	/10
Complexity	/10
Code Quality	/5
Documentation & Testing	/5
Deductions (readability, submission guidelines)	
Total	/50

Main page



Testing:

1. Registration process:

a. Trying invalid username, email, and passwords:

Registration Form

invalid username

ahmed

yourname@example.com*

Your Password Must Contain At Least 1 Uppercase Letter

The passwords should match

Confirm Password*

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Create an account

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

b. Using an already registered username:

Registration Form

username already in use

ahmed

yourname@example.com*

Confirm Password*

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Create an account

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

c. Validating a strong password:

Registration Form

Your Password Must Contain At Least 8 Characters

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

Registration Form

Your Password Must Contain At Least 1 Uppercase Letter

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

Registration Form

Your Password Must Contain At Least 1 Lowercase Letter

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

Registration Form

Your Password Must Contain At Least 1 special character e.g '!, @ etc.'

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Already have an account? [Sign in](#)

Designed by Mansur & Ahmed

Registration Form

Your Password Must Contain At Least 1 Number

☒ Remember me

By creating an account, you agree to our [Terms of Service & Privacy Policy](#).

Already have an account? [Sign in](#)

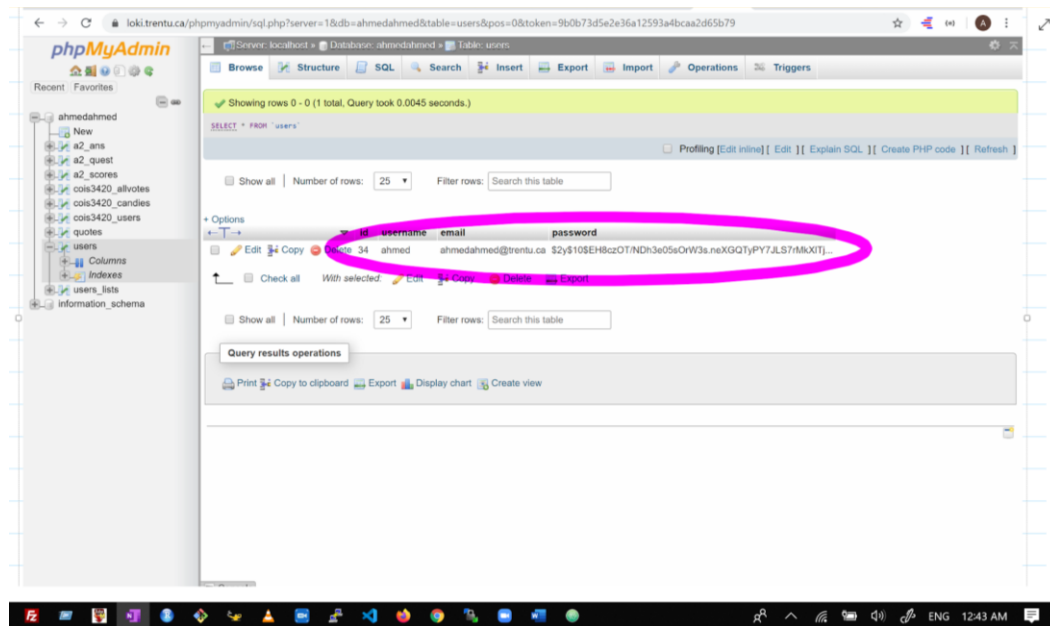
Designed by Mansur & Ahmed

d. A successful registration:

SUCCESS!

Hi there! Thank you for signing up! Your Bucketlist account is now active 😊

e. Proof of new user registration (stored in the database)



2. Login Process:

a. Login in with an Invalid username or password

Login Form

Invalid username or Password

[Forgot your password?](#)

☒ Remember Me

Not registered? [Create an account](#)

Designed by Mansur & Ahmed

b. Successful Login (redirects the user to the 'display' page)

← → ↻

loki.trentu.ca/~ahmedahmed/3420/project/display.php

☆ [4] A ⋮

Bucket List

Home

Add Goal

My List

Setting

Log out

Welcome Ahmed

Public Lists (15 in total)

by said

Goal: go to canada

Category: travel

Description: amazing

by said

Goal: Play soccer

Category: sports

Description: get healthy

← → ↻

loki.trentu.ca/~ahmedahmed/3420/project/display.php

☆ [4] A ⋮

Bucket List

Home

Add Goal

My List

Setting

Log out

Welcome Kody

Public Lists (15 in total)

by said

Goal: go to canada

Category: travel

Description: amazing

by said

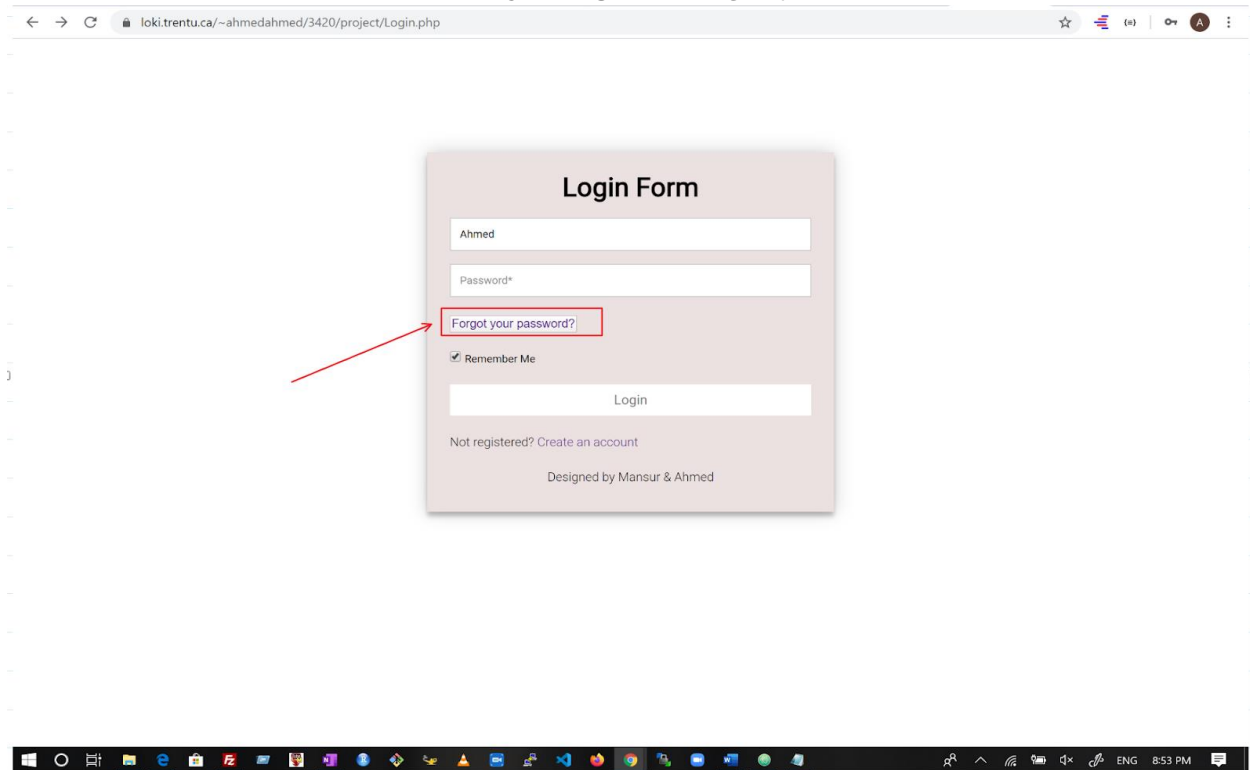
Goal: Play soccer

Category: sports

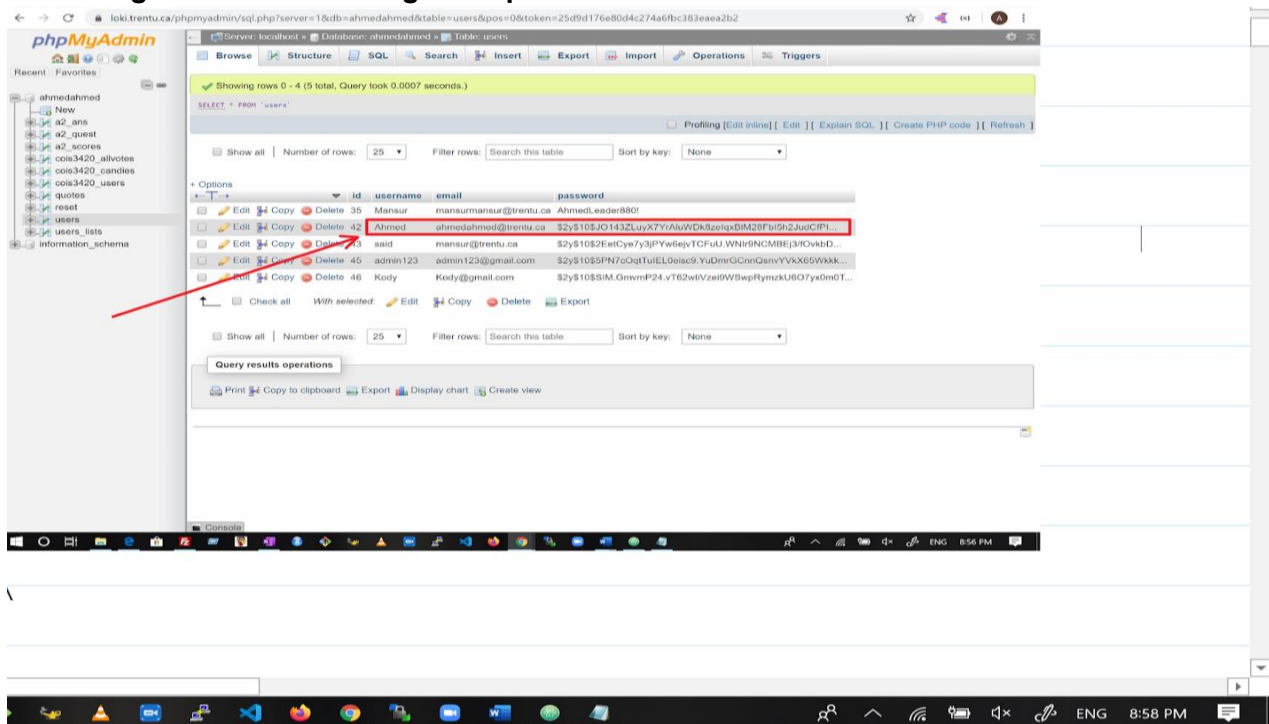
Description: get healthy

3. Password reset process:

The user can reset their password by using the “*forgot your password?*” link

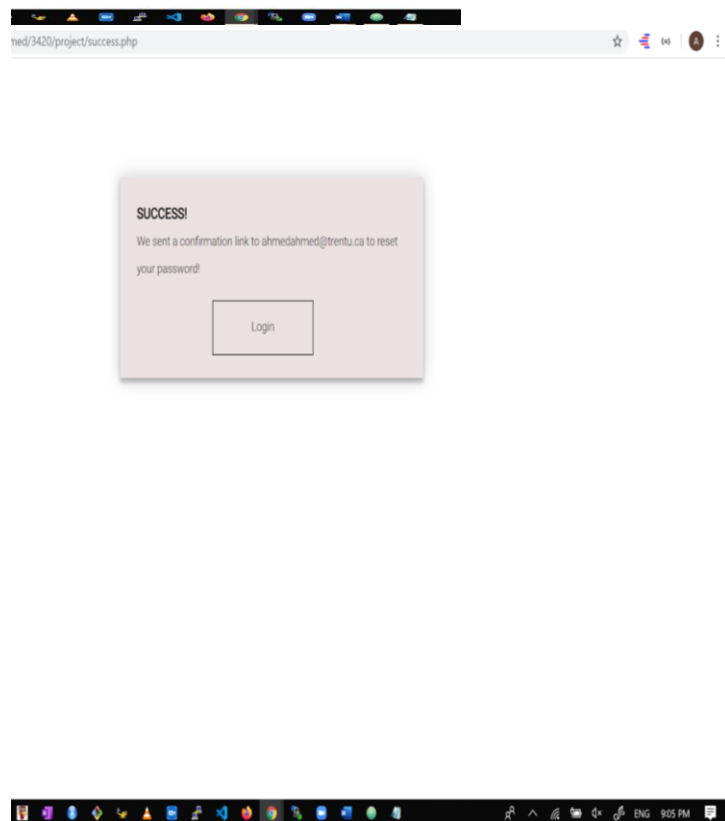
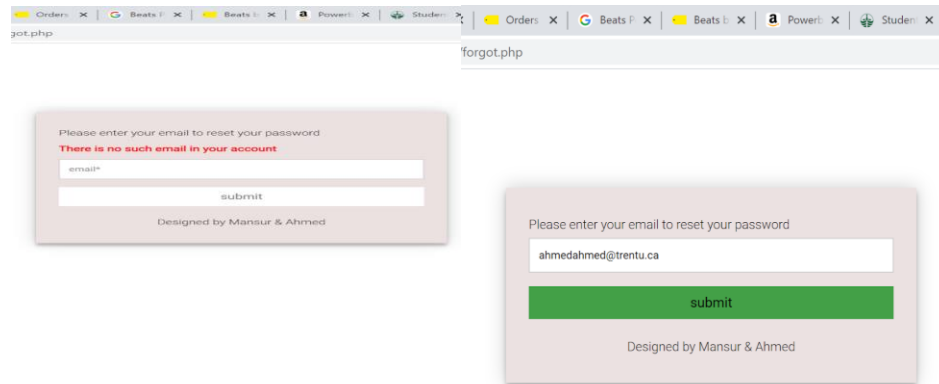


Assuming user “Ahmed” forgot his password:

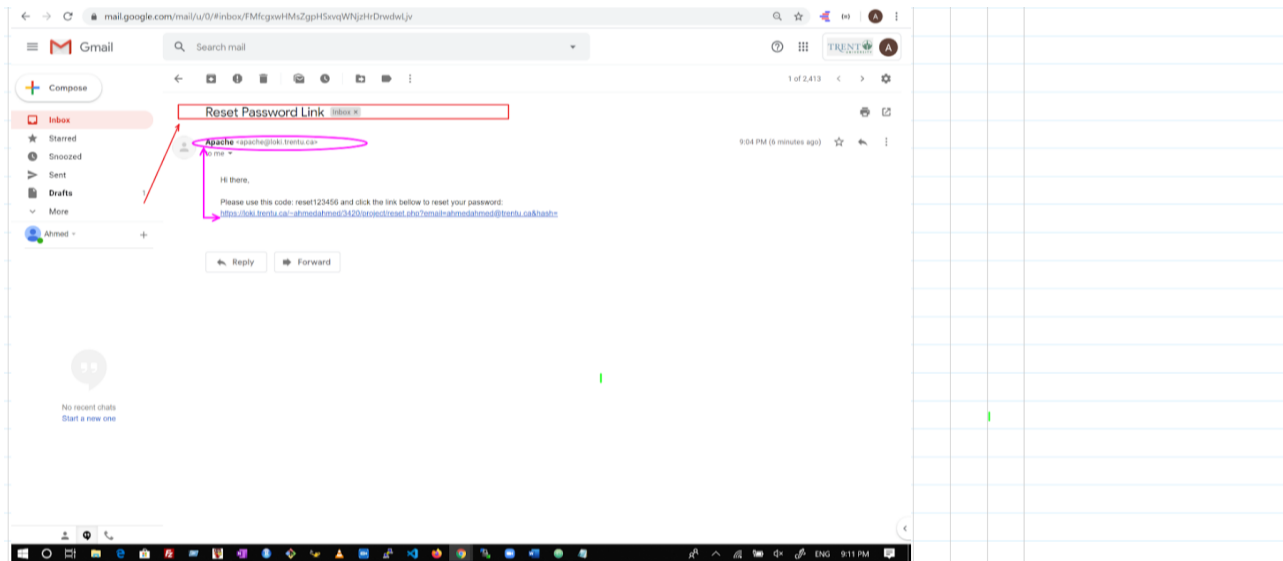


Giving a wrong email address

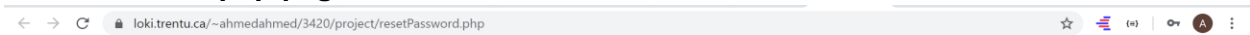
providing the correct email address



Checking the email address for the reset link:



Once the user clicks on the link provided in the reset password email, it directs to the *resetPassword.php* page



Reset Password

Designed by Mansur & Ahmed



4. My list details page:

Hello Ahmed!

Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Visit the highest mountain in Africa (Mount Kilimanjaro)	nature	This will be my first time to visit Mount Kilimanjaro	Tanzania	2020-08-31	Edit Delete
visit Wuhan	travel	I want to visit Wuhan in order to know where COVID 19 originated from	China	2020-04-30	Edit Delete
Learn how to fly a drone	entertainment	I want to start flying drones	Remote area	2020-04-25	Edit Delete
Ride Horses on the beach	entertainment	I want to experience riding a horse in a beautiful place like the Vibrant coastal city	Vancouver	2021-01-01	Edit Delete
Take a hot air balloon ride	entertainment	This will be an amazing experience for me since I never had a hot air balloon ride.	Ottawa	2020-06-05	Edit Delete
Visit the Great Wall of China	travel	This is an interesting experience since it is one of the Eight Wonders of the World	China	2020-04-12	Edit Delete
Receive Dean's Honour Roll	education	This will make me feel towards my academic and career skills	Trent University	2020-06-30	Edit Delete
Cook all my foods for 2 consecutive weeks	health	This will help care for my health and wellness	Home	2020-08-02	Edit Delete
Visit remote forests	nature	There is no experience better than enjoying in forests	Rain forest	2020-05-24	Edit Delete

Hello Kody!

Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Attend the Olympics	sports	I want to attend the Olympic for my first time in Japan.	Japan	2020-07-01	Edit Delete
Ride a mechanical bull	entertainment	This will be an amazing experience for me	New York City	2020-10-04	Edit Delete
Swim in the Major Oceans and Seas	entertainment	I like to swim in the major oceans and seas	Indian and Atlantic Ocean	2020-04-26	Edit Delete
Make a video that goes viral	entertainment	This will be my first video to go viral	Online	2020-04-19	Edit Delete

5. Add a bucket list:

I am adding a new goal and sharing with friends

← → ↻ loki.trentu.ca/~ahmedahmed/3420/project/create.php ☆ (n) A

Add Goal

Finish a class novel

Education ▾

Home

2020-06-14

This will sharpen my reading ability

☒ Share with friends

Save

Cancel

Designed by Mansur & Ahmed

To prove a new list is added (with a public privacy)

← → ↻ loki.trentu.ca/~ahmedahmed/3420/project/display.php ☆ (n) A

Bucket List

Home

Add Goal

My List

Setting

Log out

Welcome Ahmed

Public Lists (16 in total)

by said

Goal: go to canada

Category: travel

Description: amazing

by said

Goal: Play soccer

Category: sports

Description: get healthy

Windows taskbar: ENG 7:43 PM

Before Adding:

Hello Ahmed!

Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Visit the highest mountain in Africa (Mount Kilimanjaro)	nature	This will be my first time to visit Mount Kilimanjaro	Tanzania	2020-08-31	Edit Delete
visit Wuhan	travel	I want to visit Wuhan in order to know where COVID 19 originated from	China	2020-04-30	Edit Delete
Learn how to fly a drone	entertainment	I want to start flying drones	Remote area	2020-04-25	Edit Delete
Ride Horses on the beach	entertainment	I want to experience riding a horse in a beautiful place like the Vibrant coastal city	Vancouver	2021-01-01	Edit Delete
Take a hot air balloon ride	entertainment	This will be an amazing experience for me since I never had a hot air balloon ride.	Ottawa	2020-06-05	Edit Delete
Visit the Great Wall of China	travel	This is an interesting experience since it is one of the Eight Wonders of the World	China	2020-04-12	Edit Delete
Receive Dean's Honour Roll	education	This will make me feel towards my academic and career skills	Trent University	2020-06-30	Edit Delete
Cook all my foods for 2 consecutive weeks	health	This will help care for my health and wellness	Home	2020-08-02	Edit Delete
Visit remote forests	nature	There is no experience better than enjoying in forests	Rain forest	2020-05-24	Edit Delete

After Adding:

Hello Ahmed!

Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Visit the highest mountain in Africa (Mount Kilimanjaro)	nature	This will be my first time to visit Mount Kilimanjaro	Tanzania	2020-08-31	Edit Delete
visit Wuhan	travel	I want to visit Wuhan in order to know where COVID 19 originated from	China	2020-04-30	Edit Delete
Learn how to fly a drone	entertainment	I want to start flying drones	Remote area	2020-04-25	Edit Delete
Ride Horses on the beach	entertainment	I want to experience riding a horse in a beautiful place like the Vibrant coastal city	Vancouver	2021-01-01	Edit Delete
Take a hot air balloon ride	entertainment	This will be an amazing experience for me since I never had a hot air balloon ride.	Ottawa	2020-06-05	Edit Delete
Visit the Great Wall of China	travel	This is an interesting experience since it is one of the Eight Wonders of the World	China	2020-04-12	Edit Delete
Receive Dean's Honour Roll	education	This will make me feel towards my academic and career skills	Trent University	2020-06-30	Edit Delete
Cook all my foods for 2 consecutive weeks	health	This will help care for my health and wellness	Home	2020-08-02	Edit Delete
Visit remote forests	nature	There is no experience better than enjoying in forests	Rain forest	2020-05-24	Edit Delete
Finish a class novel	education	This will sharpen my reading ability	Home	2020-06-14	Edit Delete

6. Edit a list:

Before editing:

loli.trentu.ca/~ahmedahmed/3420/project/myList.php

Hello Ahmed!
Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Visit the highest mountain in Africa (Mount Kilimanjaro)	nature	This will be my first time to visit Mount Kilimanjaro	Tanzania	2020-08-31	Edit Delete
visit Wuhan	travel	I want to visit Wuhan in order to know where COVID 19 originated from	China	2020-04-30	Edit Delete
Learn how to fly a drone	entertainment	I want to start flying drones	Remote area	2020-04-25	Edit Delete
Ride Horses on the beach	entertainment	I want to experience riding a horse in a beautiful place like the Vibrant coastal city	Vancouver	2021-01-01	Edit Delete
Take a hot air balloon ride	entertainment	This will be an amazing experience for me since I never had a hot air balloon ride.	Ottawa	2020-06-05	Edit Delete
Visit the Great Wall of China	travel	This is an interesting experience since it is one of the Eight Wonders of the World	China	2020-04-12	Edit Delete
Receive Dean's Honour Roll	education	This will make me feel towards my academic and career skills	Trent University	2020-06-30	Edit Delete
Cook all my foods for 2 consecutive weeks	health	This will help care for my health and wellness	Home	2020-08-02	Edit Delete
Visit remote forests	nature	There is no experience better than enjoying in forests	Rain forest	2020-05-24	Edit Delete
Finish a class novel	education	This will sharpen my reading ability	Home	2020-06-14	Edit Delete

Windows taskbar: 7:47 PM

After Editing:

loli.trentu.ca/~ahmedahmed/3420/project/myList.php

Hello Ahmed!
Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Visit the highest mountain in Africa (Mount Kilimanjaro)	nature	This will be my first time to visit Mount Kilimanjaro	Tanzania	2020-08-31	Edit Delete
visit Wuhan	travel	I want to visit Wuhan in order to know where COVID 19 originated from	China	2020-04-30	Edit Delete
Learn how to fly a drone	entertainment	I want to start flying drones	Remote area	2020-04-25	Edit Delete
Ride Horses on the beach	entertainment	I want to experience riding a horse in a beautiful place like the Vibrant coastal city	Vancouver	2021-01-01	Edit Delete
Take a hot air balloon ride	entertainment	This will be an amazing experience for me since I never had a hot air balloon ride.	Ottawa	2020-06-05	Edit Delete
Visit the Great Wall of China	travel	This is an interesting experience since it is one of the Eight Wonders of the World	China	2020-04-12	Edit Delete
Receive Dean's Honour Roll	education	This will make me feel towards my academic and career skills	Trent University	2020-06-30	Edit Delete
Cook all my foods for 2 consecutive weeks	health	This will help care for my health and wellness	Home	2020-08-02	Edit Delete
Visit remote forests	nature	There is no experience better than enjoying in forests	Rain forest	2020-05-24	Edit Delete
Finish a classic novel	entertainment	This will sharpen my reading ability	At summer camp	2020-06-14	Edit Delete

Annotations: A pink box highlights 'Finish a classic novel'. A blue circle highlights 'entertainment' with an arrow pointing to it. A green triangle highlights 'At summer camp' with an arrow pointing to it.

Windows taskbar: 7:54 PM

7. Delete a list:

The program first ask a confirmation before doing the delete option:

loki.trentu.ca/~ahmedahmed/3420/project/myList.php

loki.trentu.ca says
Are you sure You want to delete?

OK Cancel

Hello Kody!
Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Attend the Olympics	sports	I want to attend the Olympic for my first time in Japan.	Japan	2020-07-01	Edit Delete
Ride a mechanical bull	entertainment	This will be an amazing experience for me	New York City	2020-10-04	Edit Delete
Swim in the Major Oceans and Seas	entertainment	I like to swim in the major oceans and seas	Indian and Atlantic Ocean	2020-04-26	Edit Delete
Make a video that goes viral	entertainment	This will be my first video to go viral	Online	2020-04-19	Edit Delete

https://loki.trentu.ca/~ahmedahmed/3420/project/deleteList.php?id=22

After deleting:

loki.trentu.ca/~ahmedahmed/3420/project/myList.php

Hello Kody!
Your list information can be found in the table below

Goal	Category	Description	Location	Date	Edit/Delete
Attend the Olympics	sports	I want to attend the Olympic for my first time in Japan.	Japan	2020-07-01	Edit Delete
Ride a mechanical bull	entertainment	This will be an amazing experience for me	New York City	2020-10-04	Edit Delete
Swim in the Major Oceans and Seas	entertainment	I like to swim in the major oceans and seas	Indian and Atlantic Ocean	2020-04-26	Edit Delete

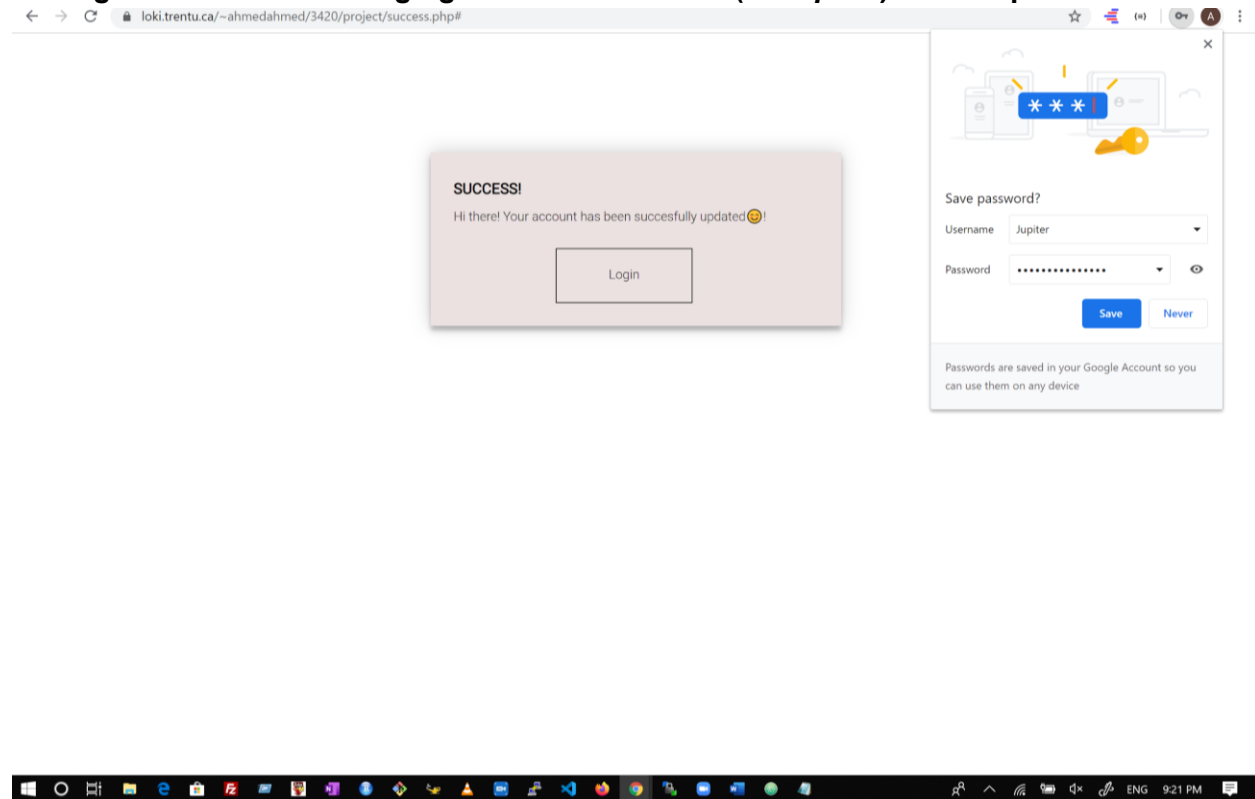
The list is gone

ENG 8:08 PM

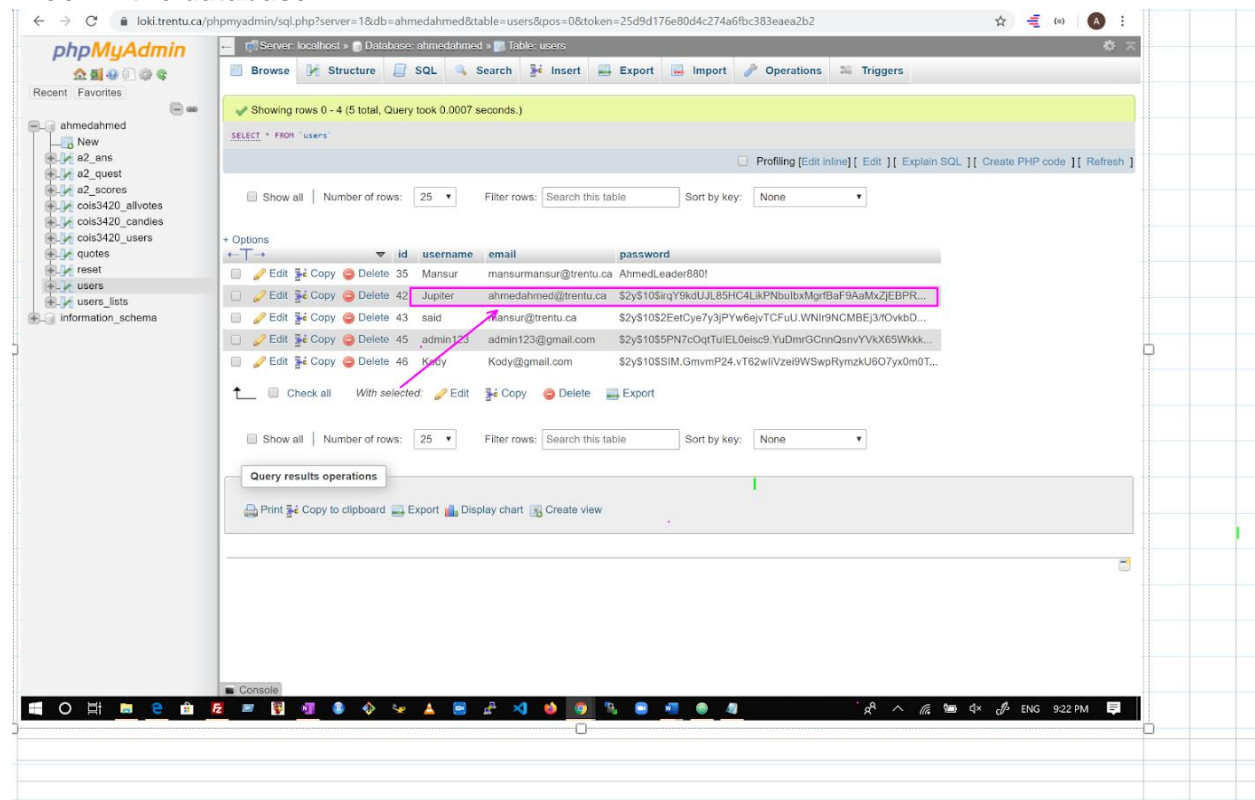
8. Settings page (allows edit account and delete account):

Edit Account:

Editing user “Ahmed” changing both the username (to *Jupiter*) and the password

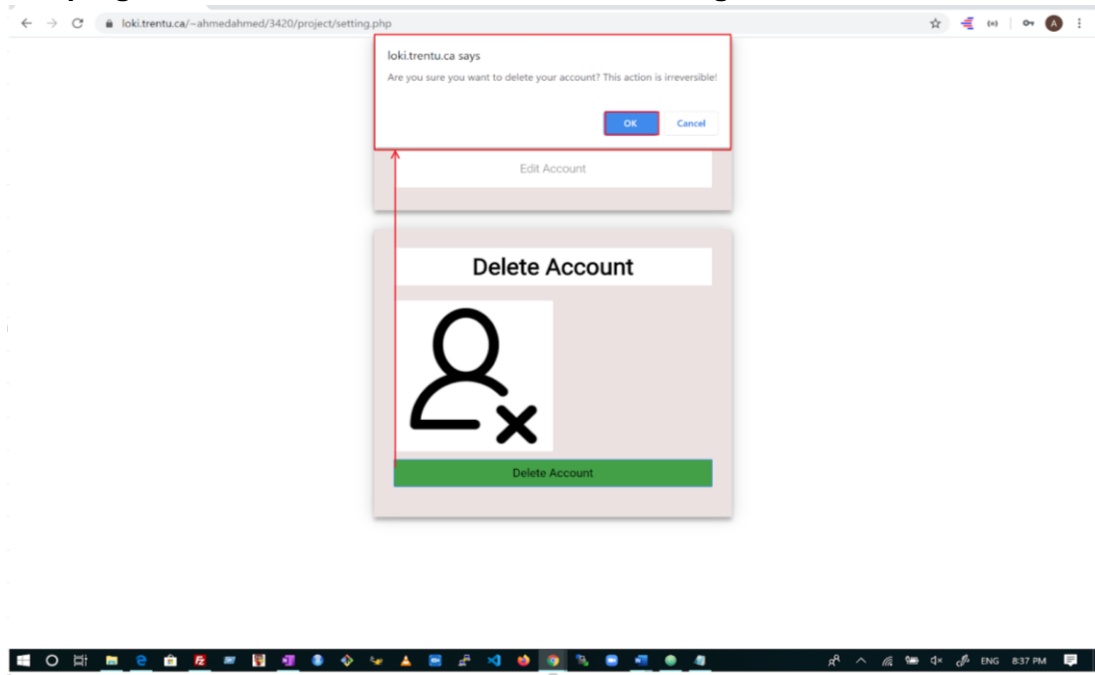


Proof in the database

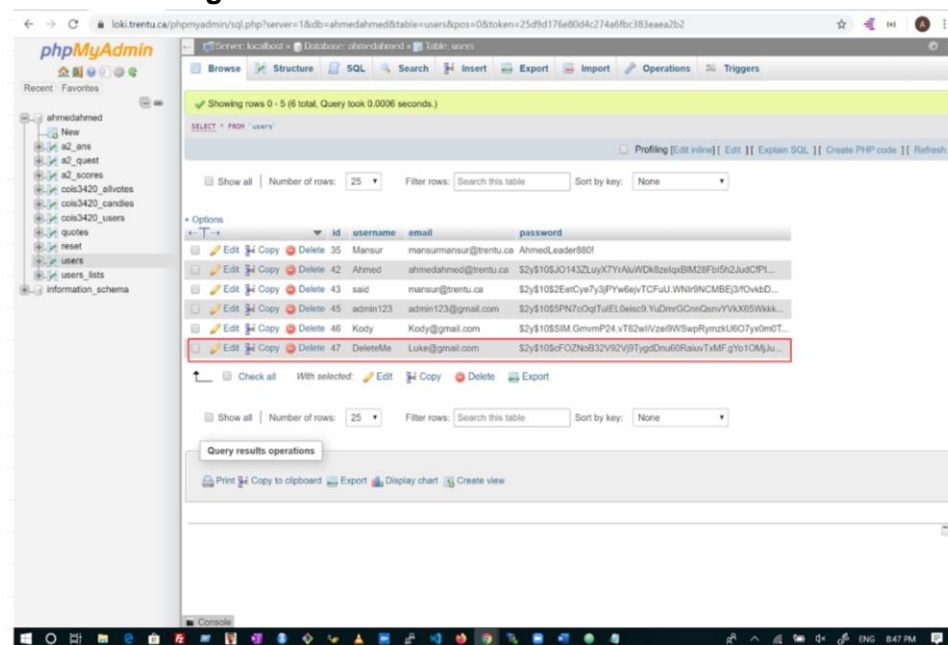


Delete Account:

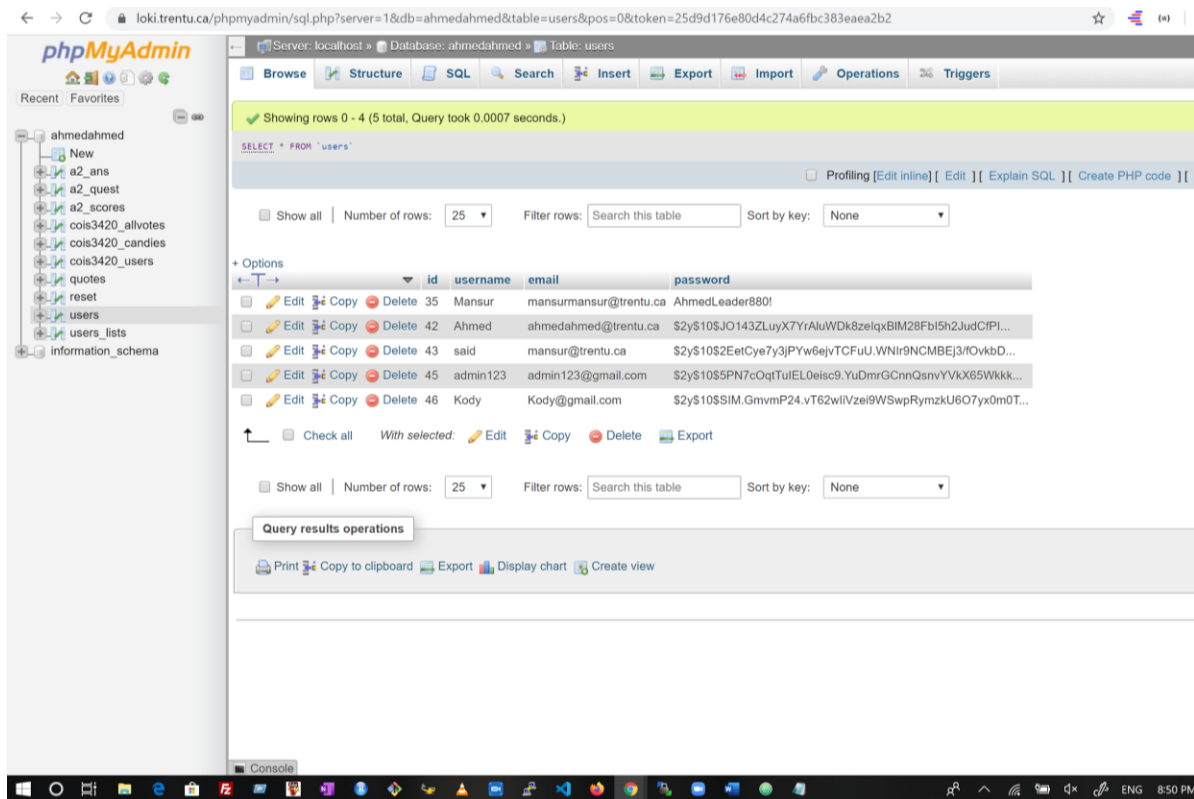
The program confirms from the user before deleting the account



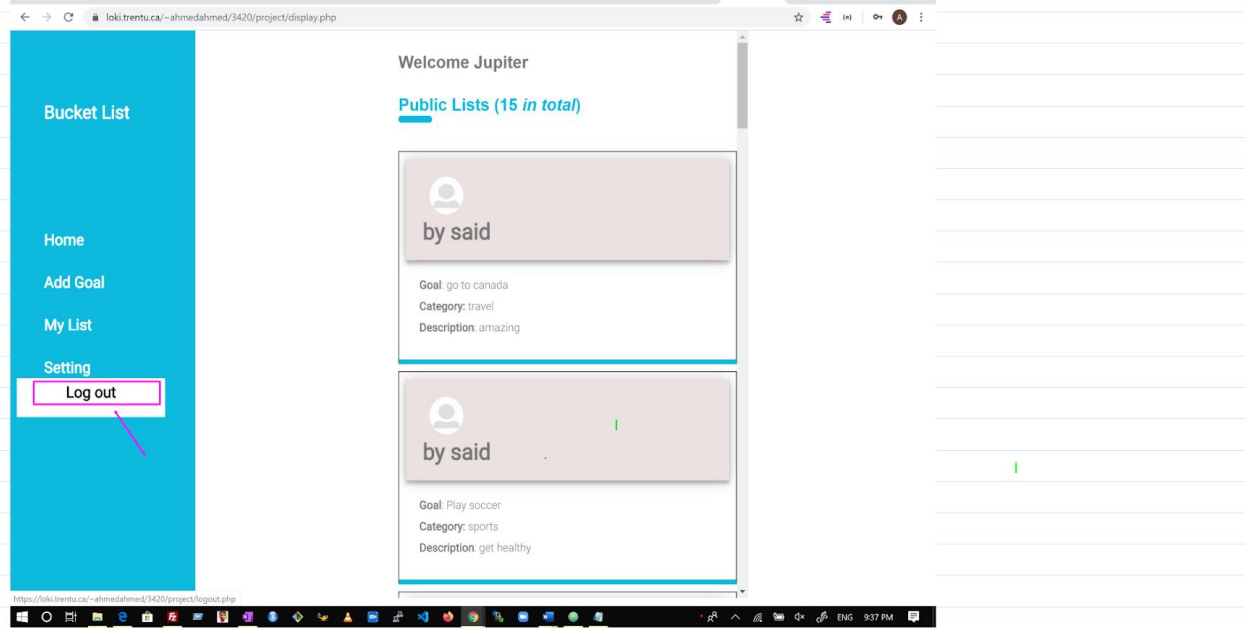
Before Deleting the account



After Deleting:

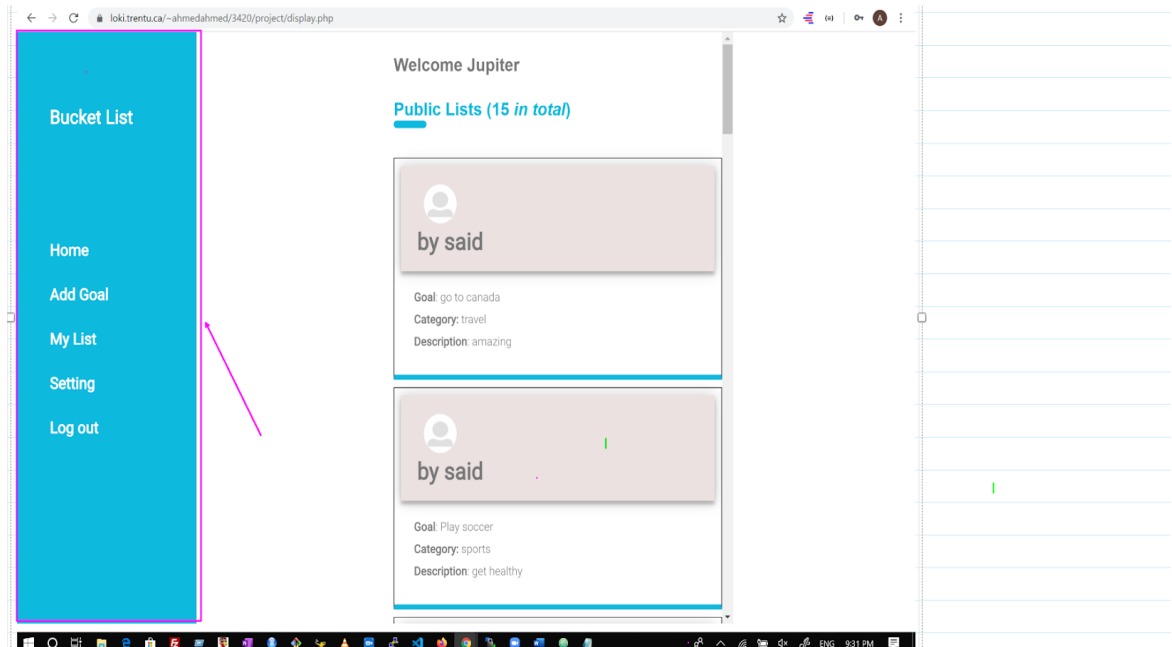


9. Logout functionality: it destroys the *SESSION* & redirects the user to index page

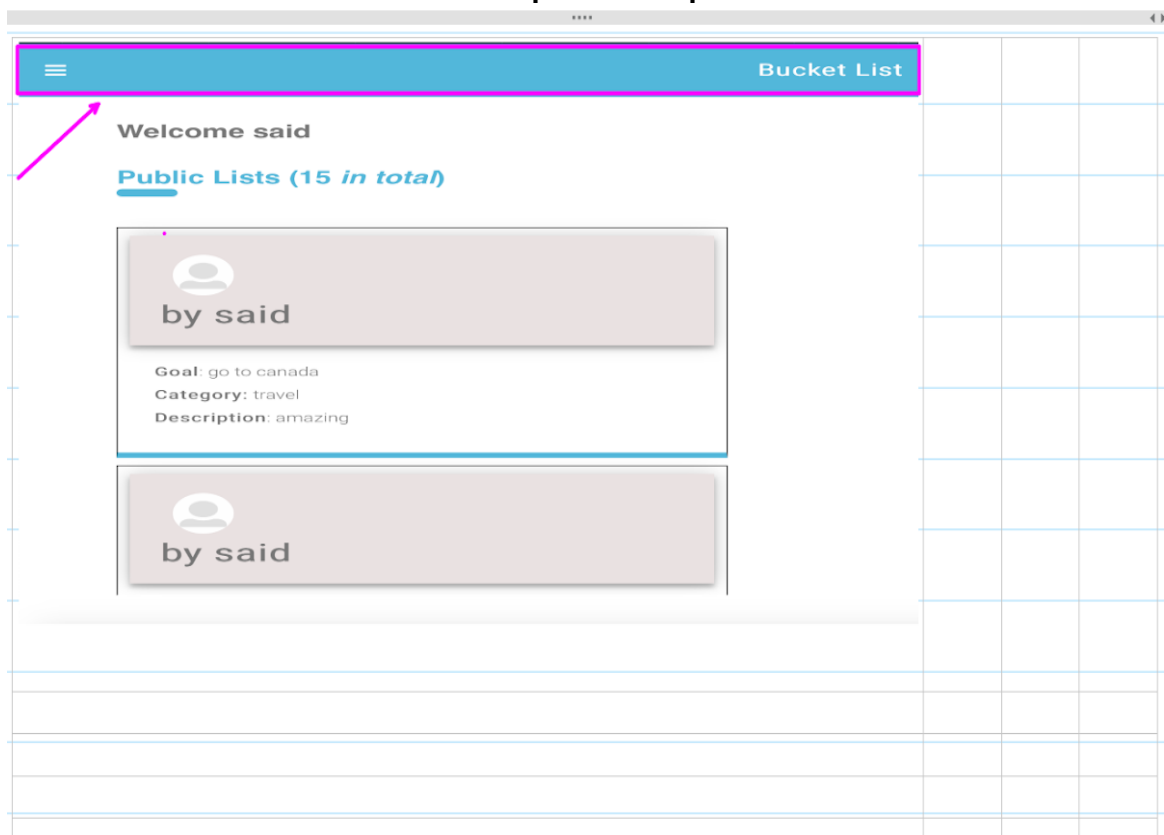


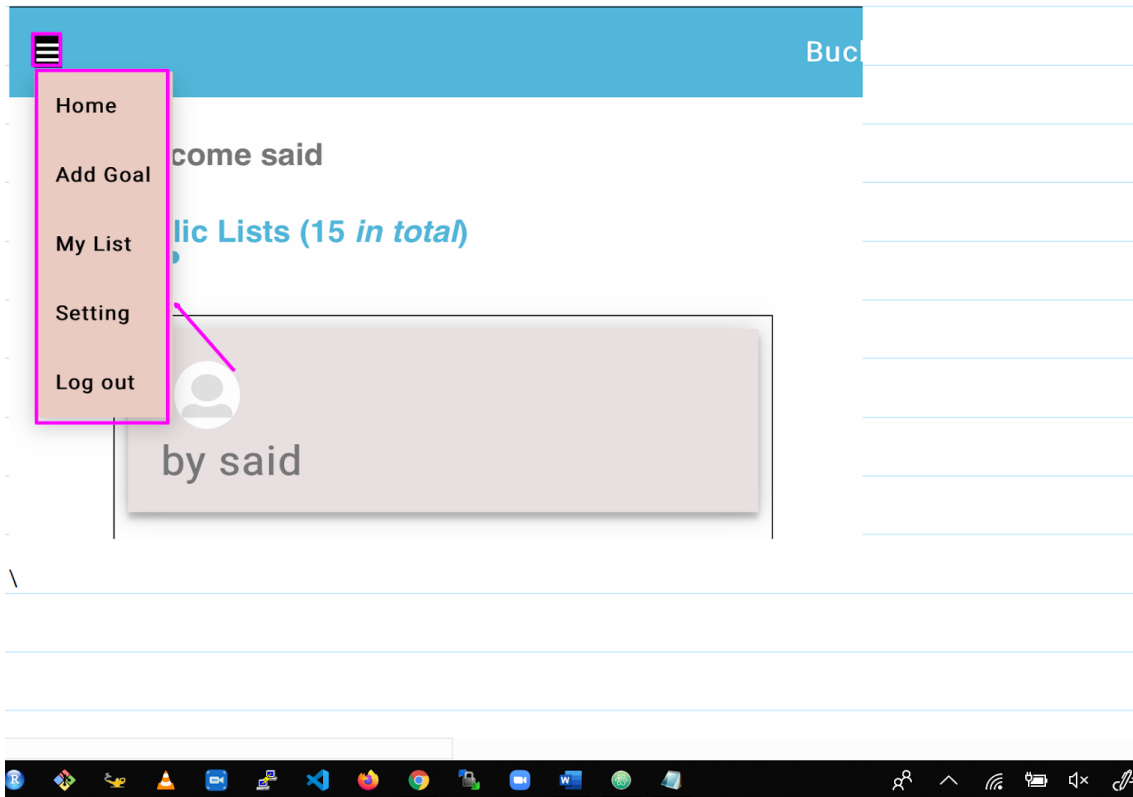
10. Other features:

- Responsive navigation menu



- When screen minimized less than 950px with drop down menu





Project Code (for readability purposes we use hightled the pages as: **INDEX PAGE,**
LOGIN PAGE, etc....)

```
<!--=====INDEX(MAIN) PAGE=====-->
<!--
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the MAIN page. It contains:
    1- A HTML that displays some welcoming message for the user
    II- A Header that contains Registration link, a sign in
link
    II- other useful information for the user.

-->
<!DOCTYPE html>
<html lang="en">
    <!--meta data-->
    <head>
        <?php
        $PAGE_TITLE = "Home page|welcome";
        include "includes/metadata.php";
```

```

?>
</head>
<!--content-->
<body>
  <!-- header -->
  <div id="header">
    <?php include "includes/header.php"; ?>
  </div>
  <!-- Main Content -->
  <main>

    <!-- this section will contain a background photo and one or two
statements -->
    <section id="showarea">

      <h2>Welcome to the BucketList application</h2>
      <figure>
        
        
      </figure>
      <div class="caption1">Start listing you own goals & the
adventures you want to fulfill in the near future</div>
      <div class="caption2">Together we can fulfill each other's
dreams!</div>
      <button class="indexButton1" type="button" name="SignUp"><a
href="https://bestow.com/blog/bucket-list-ideas/">View some Bucketlist
Ideas</a></button>
    </section>
    <!-- a nother section -->
    <section class="functionality">
      <h2>Live your life to the fullest</h2>
      <!-- Icons -->
      <div class="Icons">
        <div class="row">
          <table>
            <tr>
              <td></td>
              <td><div class="iconHeader">Set Your Goal</div> </td>
            </tr>
            <tr>
              <td></td>

```

```

        <td> <div class="iconHeader">The Power Of Two</div>
</td>

    </tr>
</table>
</div>
<div class="row">
    <table>
        <tr>
            <td>  </td>
            <td><div class="iconHeader">Be Your Hero</div></td>
        </tr>
        <tr>
            <td></td>
            <td><div class="iconHeader">Make A Bucket
List</div></td>
        </tr>
    </table>
</div>
<div class="row">
    <table>
        <tr>
            <td>
                <div class="share">
                    <svg aria-hidden="true" focusable="false" data-
prefix="fas" data-icon="paper-plane" class="svg-inline--fa fa-paper-plane
fa-w-16" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512
512"><path fill="gray" d="M476 3.2L12.5 270.6c-18.1 10.4-15.8 35.6 2.2
43.2L121 358.4l287.3-253.2c5.5-4.9 13.3 2.6 8.6 8.3L176 407v80.5c0 23.6
28.5 32.9 42.5 15.8L282 426l124.6 52.2c14.2 6 30.4-2.9 33-18.2l72-432C515
7.8 493.3-6.8 476 3.2z"></path></svg>
                </div>
            </td>
            <td><div class="iconHeader">Share With
Friends</div></td>
        </tr>
        <tr>
            <td></td>
            <td> <div class="iconHeader">Live It Up!</div> </td>
        </tr>
    </table>
</div>

```

```

        </div>
    </section>
</main>
<!--footer-->
<?php include "includes/footer.php"; ?>
</body>
</html>

<!--=====SIGN UP (REGISTRATION) PAGE=====-->
<!--
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the registration page. It contains:
    1- A HTML form in which a user can fill the required
information.
    2- A PHP code that:
        I. Validates the user's information in the HTML form
given
        II. Prints the error messages if the user's
informaiton are not valid, or already in use
        III. Processes the registration process for successful
user's informaiton
        IV. Stores the user's data into the database talbe
        V. Redirects the user to the index page
REFERENCE:
    https://www.codexworld.com/how-to/validate-password-strength-in-php/
-->
<?php //start of the PHP code
include ("includes/database.php"); // Include library file
session_start(); //start the session
$pdo = connectDB(); // Get database connection

//An array to store errors
$PswErrors = []; //for password errors
$userErrors = []; // for username errors
$emErrors = []; //for email errors
$PswMErrors = []; //for password mismatch errors

// Validate on submit (register)
if (isset($_POST['register'])) {

    /* Get everything from $_POST */

```

```

$username = $_POST['username'] ?? NULL; //for username
$email = $_POST['email'] ?? NULL; //for email address
$psw = $_POST['psw'] ?? NULL; //for password

// compare the username given with the ones in the database table
$query = "SELECT 1 username FROM users WHERE username = ?";
$stmt = $pdo->prepare($query);
$stmt->execute([$_POST['username']]);
$result = $stmt->fetchAll();

// check if the username is already in use
if (!empty($result)) {
    // set the flag to be false if the username is in use
    $newUsername = false;
}

// compare the email given with the ones already in register in the
table
$query = "SELECT 1 email FROM users WHERE email = ?";
$stmt = $pdo->prepare($query);
$stmt->execute([$_POST['email']]);
$result = $stmt->fetchAll();

// check if the email is already in use
if (!empty($result)) {
    // set the flag to be false if the email is already registered
    $newEmail = false;
}

//=====USERNAME VALIDATION=====//
/*
    Validate Username with the following conditions:
        1- Do not register a username with less than 3 character and more
than 10 characters
        2- Do not allow usernmae with special characters other than the
regualr ones and numbers
        3- Do not allow a username that has already been in use to
registred again
*/
if (!( strlen($_POST['username']) >= 3 && strlen($_POST['username'])

```



```

<= 10 && preg_match("/^[a-zA-Z0-9]*$/", $_POST['username']) ))
array_push($userErrors, "invalid username");
    if (isset($newUsername) && !$newUsername) array_push($userErrors,
"username already in use"); //allow only unique username

//===== EMAIL VALIDATION=====//
/*
    Validate email with the following conditions:
        1- Do not allow an invalid email, the email should be valid.
        2- Do not allow an already existing email to be registered
*/
    if (!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL))
array_push($emErrors, "Invalid email address");
    if (isset($newEmail) && !$newEmail) array_push($emErrors, "email
already in use"); //allow only unique emails

//===== PASSWORD
VALIDATION=====//
/*
    For the password to be strong enough it should have the following
conditons before it can be accepted as a valid password:
        1- It should be atleast 8 characters long
        2- It must contain at least 1 Uppercase letter
        3- It must contain at leaset 1 lowercase Letter
        4- It must contain at least 1 number
        5- It must contain at least 1 special character
*/
    if (strlen($psw) < '8') array_push ($PswErrors, "Your Password Must
Contain At Least 8 Characters");
    elseif(!preg_match("#[A-Z]+#", $psw)) array_push ($PswErrors, "Your
Password Must Contain At Least 1 Uppercase Letter");
    elseif(!preg_match("#[a-z]+#", $psw)) array_push ($PswErrors, "Your
Password Must Contain At Least 1 Lowercase Letter");
    elseif(!preg_match("#[0-9]+#", $psw)) array_push ($PswErrors, "Your
Password Must Contain At Least 1 Number");
    elseif(!preg_match('@[^\w]@', $psw)) array_push ($PswErrors, "Your
Password Must Contain At Least 1 special character e.g '!, @ etc.'");

    // Validate Password Confirm (equals password)
    if (!( $psw === $_POST['psw_repeat'] ))array_push($PswMErrors, "The
passowrds should match");

```

```

//=====PROCESS
REGISTRATION=====//
/*
    The following piece of code processes registraton process by:
    =>First ensuring there is no valiation errors
    =>Hashing the passowrd for security purposes
    =>Inserting all the user information such as username, email, and
    passowrd(hashed password) INTO the database table
*/

// Check if there is no validation errors
if (count($userErrors) === 0 && count($PswErrors) === 0 &&
count($emErrors) === 0 && count($PswMErrors) === 0) {

    // Hash Password
    $HashPsw = password_hash($psw, PASSWORD_DEFAULT);

    /*
        Insert all the user information intot he tabel
    */
    $query = "INSERT INTO users (username, email, password) VALUES
    (?, ?, ?)";
    $pdo->prepare($query)->execute([$ _POST['username'], $ _POST['email'],
    $HashPsw]);

    //get the id of the user from the database table
    $query = "SELECT id FROM users WHERE username = ? LIMIT 1";
    $stmt = $pdo->prepare($query);
    $stmt->execute([$ _POST['username']]);
    $result = $stmt->fetchAll();

    // Get the user id retrieved from the table
    $userId = $result[0]['id'];

    // Set session variable to userid
    $_SESSION['id'] = $userId;

    $_SESSION['message'] = "Hi there! Thank you for signing up! Your
    Bucketlist account is now active😊!";
    // Redirect the user to index page

```

```

        header("Location: ./success.php");
        die();

    }
}
?> <!--END OF PHP CODE-->

<!--START OF HTML DOCUMENT-->
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <title>Create account|welcome</title>
        <link rel="stylesheet" href="css/master.css">
    </head>
    <body>
        <!-- main content -->
        <div class="container">
            <form id="signupForm" action="SignUp.php" method="post">
                <h3>Registration Form</h3>
                <span class='error' id="UsernameError"></span>
                <!--echo out username ERRORS-->
                <?php foreach ($userErrors as $error): ?>
                    <li id="error"><?= $error ?></li>
                <?php endforeach; ?>
                <fieldset>
                    <input type="text" placeholder="username*" name="username"
required="">
                </fieldset>
                <span class='error' id="EmailError"></span>
                <!--echo out email ERRORS-->
                <?php foreach ($emErrors as $error): ?>
                    <li id="error"><?= $error ?></li>
                <?php endforeach; ?>
                <fieldset>
                    <input type="email" placeholder="yourname@example.com*"
name="email" required="">
                </fieldset>
                <span class='error' id="PasswordError">
                <!--echo out password strength ERRORS-->
                <?php foreach ($PswErrors as $error): ?>
                    <li id="error"><?= $error ?></li>

```

```

        <?php endforeach; ?>
        <fieldset>
            <input type="password" placeholder="Password*" name="psw"
required="">
        </fieldset>
        <span class='error' id="PasswordError">
        <!--echo out password mismatch ERRORS-->
        <?php foreach ($PswMErrors as $error): ?>
            <li id ="error"><?= $error ?></li>
        <?php endforeach; ?>
        <fieldset>
            <input type="password" placeholder="Confirm Password*"
name="psw_repeat" required="">
        </fieldset>
        <fieldset>
            <input type="checkbox" id="checkbox" checked name="remember">
            <label for="checkbox">Remember me</label>
        </fieldset>
        <fieldset>
            <div >By creating an account, you agree to our <a class="terms"
href="#">Terms of Service & Privacy Policy.</a></div>
        </fieldset>
        <fieldset>
            <button type="submit" class="btn" name="register">Create an
account</button>
        </fieldset>
        <fieldset>
            <div >Already have an account? <a
class="mainform"href="Login.php">Sign in</a></div>
        </fieldset>
        <p class="copyright">Designed by Mansur & Ahmed</p>
    </form>
</body>
</html>

```

```

<!--===== LOGIN PAGE =====>
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the logIn page. It contains:
    1- A HTML form in which a user can fill the required
information for the login process.
    2- A PHP code that:
        I.  Validates the user's information in the HTML form

```

given

II. Prints the error messages if the user's
information are not valid by comparing
the username & password with the ones in record (in the database)
III. Processes the login process for correct username
and password
IV. Redirects the user to the bucketlist information
page

REFERENCES:

<https://www.dreamincode.net/forums/topic/286198-simple-login-page-with-only-3-password-attempts/>

-->

```
<?php //START OF PHP CODE
include('includes/database.php'); //requires database connection
session_start(); //start the session

// Get database connection
$pdo = connectDB();

//variable to exit the login page if number of attempts exceeded
$exit = false; //set to false for the first time

//array to store errors during validation
$errors = [];

//boolean variable to verify passwords
$passExists = false;
//=====LOG IN
VALIDATION=====//
/*
    Validates the user's Username and password by:
        1- Setting the number of attempts at 0
        2- Getting the username and password from the $_POST
        3- Gets the username and password from the database and
compares with the ones obtained from the $_POST
        4- validates the username and password
*/

// Validate on submit (login)
if (isset($_POST['login'])){
```

```

        // Check if the attempts value is set
        if(!isset($_SESSION['Attempts'])) {
            // if not set it at zero
            $_SESSION['Attempts'] = 0;
        }
        /* Get everything from $_POST */
        $username = $_POST['username'] ?? NULL;
        $psw = $_POST['psw'] ?? NULL;

        // Get the username and password from the database
        $statement = $pdo->prepare("SELECT id, username, password
FROM users WHERE username = ?");
        $statement->execute([$username]);
        $results = $statement->fetch();

        //verify the password
        if(password_verify($psw, $results['password']))
        {
            $passExists = true;
        }

        //check if the username & the password given is not valid by
        comparing the ones in the database table
        if ($results['username'] != $username || !$passExists) {
            //store the an ERROR message in the array
            array_push($errors, "Invalid username or Password");
            $_SESSION['Attempts']++; //increment the number of
attempts in session
            //check if the number of attempts has reached 3 or more
            if ($_SESSION['Attempts'] >= 3){
                //set the flag of the exit to be true
                $exit = true;
            }
        }

        // Sets cookies when remember box is checked on login
        if(isset($_POST['remember']) ) {
            //sets the cookies for the user's username
            setcookie('username', $_POST['username']);
            //sets the cookies for the user's password

```

```

        setcookie('psw', $_POST['psw']);
    }

    //=====PROCESS LOG
IN=====//
    /*
        The following piece of code processes log in process by:
        =>First ensuring there is no valiation errors
        =>unsetting the remaining number of attempts in session
        =>Redirect the user to the BucketList page
    */
    if (count($errors) == 0)
    {

        //unset the remaining attempts on
successful login

        unset($_SESSION['Attempts']);

        // Set the userid to session variable.
        $_SESSION['user'] =

$results['username'];

        $_SESSION['userid'] = $results['id'];
        //print a welcome message with the
user's username

        // Redirect to index.
        header("Location: ./display.php");
        echo $results['username'];

        die();
    }

    //=====EXIT THE LOG IN
PAGE=====//
    /*
        The follwing code exits the user from the login page if the
number of unsuccessful
        attempts on the login process exceeds 3 and prints an alert
message to indicate that.
        It also unsets the attempts in session.
    */
    //check if the exit flag is set (3 or more unsuccessful attempts)
    if($exit){
        //print an alert message to the user

```

```

        $_SESSION['message'] = "Sorry 😞! you have exceeded the
maximum number of login attempts allowed. You can login after sometime.";
        //unset the remaining attempts in session
        unset($_SESSION['Attempts']);
        header("Location: error.php");
    }

}

//forgot password code
if(isset($_POST['forgot'])){
    header("Location: forgot.php");
    die();
}

?><!--END OF PHP CODE-->

<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>Login|welcome</title>
        <link rel="stylesheet" href="css/master.css">
    </head>
    <body>
        <div class="container">
            <!-- Login form -->
            <form id="loginForm" action="Login.php" method="post">
                <h3>Login Form </h3>
                <!--echo out username and passowrd ERRORS-->
                <?php foreach ($errors as $error): ?>
                    <div id="error"><?= $error ?></div>
                <?php endforeach; ?>
                <fieldset>
                    <input type="text"
placeholder="Username*" name="username" required="">
                </fieldset>
                <fieldset>
                    <input type="password"
placeholder="Password*" name="psw" required="" >
                </fieldset>
                <fieldset>
                    <div><button name="forgot"><a class="terms"
href="forgot.php" >Forgot your password?</a></button></div>

```



```

        </fieldset>
        <div class="Logintext"></div>
        <fieldset>
            <input type="checkbox" id="checkbox"
checked="checked" name="remember" style="margin-bottom:15px">
            <label for="remember">Remember Me</label>
        </fieldset>
        <fieldset>
            <button type="submit" name="login"
value="submit">Login</button>
        </fieldset>
        <fieldset>
            <div >Not registered? <a
class="mainform"href="SignUp.php">Create an account</a></div>
        </fieldset>
        <p class="copyright">Designed by Mansur &
Ahmed</p>
    </form>
</div>
</body>
</html>

```

```

<!--=====DISPLAY PAGE=====
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the Display page. It contains:
    1- A HTML which shows the structure of the user dashboard.
when the user logs in, they will have navigation
    menu which will take them to create goal, or setting
page, or mylist page, or logout page, and home page which shows all public
list
    of other users.
    2- A PHP code that:
        I. checks if the user logged in. if no then redirects
back to the Login page
        II. it also displays all the list of other users in the
news feed which are public

-->
<?php
    include('includes/database.php'); //requires database connection
    //connect to the database
    $pdo = connectDB();

```

```

//session_start
session_start();
//check if the user has logged in
if(!isset($_SESSION['user'])){
    //user has not logged
    header("Location:Login.php"); //redirect the user back to login
    exit();
}

//get all the values in the users_lists
$stmt = $pdo->prepare("SELECT * FROM users_lists WHERE share = ?");
$stmt->execute([1]);
$results = $stmt->fetchAll();

$count = count($results);
?>
<!DOCTYPE html>
<html lang="en" dir="ltr">
    <head>
        <meta charset="utf-8">
        <title>Display|welcome</title>
        <link rel="stylesheet" href="css/master.css">
        <script defer src="script/main.js"></script>
    </head>
    <body>
        <!--sidebar/menu-->
        <nav class="sidebar">
            <h3><br>Bucket List <br></h3>
            <form class="home" action="<?php echo $_SERVER['PHP_SELF']; ?>"
method="post">
                <a href="display.php" name="home">Home</a>
            </form>
            <a href="create.php">Add Goal</a>
            <form class="myList" action="<?php echo $_SERVER['PHP_SELF']; ?>"
method="post">
                <a href="myList.php" name="list">My List</a>
            </form>
            <a href="setting.php">Setting</a>
            <a href="logout.php">Log out</a>
        </nav>

        <!--
            dropdown-menu when the screen is minimized.

```

```

-->
<header class="header">
  <div class="dropdown">
    <a href="javascript:void(0)" class="menuButton">≡</a>
    <div class="dropdown-content">
      <a href="display.php" name="home">Home</a>
      <a href="create.php" >Add Goal</a>
      <a href="myList.php" name="list">My List</a>
      <a href="setting.php">Setting</a>
      <a href="logout.php">Log out</a>
    </div>
  </div>
  <span>Bucket List</span>
</header>

<!--page content-->
<div class="main">
  <!--Header-->
  <div class="title">
    <h1><br> Welcome <?php echo $_SESSION['user']; ?><br></h1>
    <h1 id="h1"><br>Public Lists (<?php echo $count?> <em>in
total</em>)<br></h1>
    <hr class="line">
  </div>
  <!--container display that holds list of other users-->
  <div class="display">
    <div id="shared">
      <!--the loop that goes through all the user list that has been
fetched.-->
      <?php foreach ($results as $result): ?>
        <fieldset>
          <div class="userdiv">
            <div class="bar1">
              <div>
                <p class="username">
                  <?php
                      // get the user name from the database
                      $statement = $pdo->prepare("SELECT username FROM
users WHERE id = ?");
                      $statement->execute([$result['user_id']]);
                      $results = $statement->fetch();
                  ?> </p>

```

```

        </div>
        <div>
            
<h1>by <?php echo $results['username']; ?></h1>
        </div>
    </div>
    <div class="wrapper">
        <div><span class="span">Goal</span>: <?php echo
$result['goal']; //get goal ?></div>
        <div><span class="span">Category:</span> <?php echo
$result['category']; //get the category?></div>
        <div><span class="span">Description</span>: <?php echo
$result['description']; //get the description?></div>
    </div>
</div>
</fieldset>
    <?php endforeach; ?> <!--// end of the loop-->
</div>
</div>
</div>
</body>
</html>

```

<!--===== **ADD A LIST PAGE** =====>

Student names: Ahmed Ahmed & Mansur Mansur

COURSE: COIS 3420H

PURPOSE: This is the create page. It contains:

1- A HTML (the html body contains only the form) which contains a create goal form. which allows the user to add a goal to their list.

2- A PHP code that:

I. it checks first whether the user has already logged in. if so

the user can proceed with add goal or else it will redirect back to login page.

II. when the save button is clicked while all the form is filled,

it insert the goal in users_lists table in the database.

III. if the insertion was success it redirects the user back to

the display page.

-->

```

<?php
include ("includes/database.php"); // Include library file
session_start(); //start the session
$pdo = connectDB(); // Get database connection

//check if the user has logged in
if(!isset($_SESSION['user'])){
    //user has not logged
    header("Location:Login.php"); //redirect the user back to login
    exit();
}

//if already logged in and clicked save button

if(isset($_POST['button'])){

    //get all the values from the post
    $goal = $_POST['goal'] ?? NULL;
    $category = $_POST['category'] ?? NULL;
    $location = $_POST['location'] ?? NULL;
    $description = $_POST['description'] ?? NULL;
    $share = $_POST['share'] ?? NULL;
    $date = $_POST['date'] ?? NULL;
    $submit = $_POST['save_createGoal'] ?? NULL;

    /*
        Insert all the user information into the table
    */
    $user_id = $_SESSION['userid'];
    $query = "INSERT INTO users_lists (user_id, goal, category, description,
location, goalDate, share) VALUES (?, ?, ?, ?, ?, ?, ?)";
    if($_POST['share'] == NULL){ //if user did not click the share radio
button set the share column to 0. which is make it private.
        $pdo->prepare($query)-
>execute([$user_id, $_POST['goal'], $_POST['category'], $_POST['description'],
$_POST['location'], $_POST['date'], 0]);
    } else { //else share will be set to 1.
        $pdo->prepare($query)-
>execute([$user_id, $_POST['goal'], $_POST['category'], $_POST['description'],
$_POST['location'], $_POST['date'], $_POST['share']]);
    }
}

```

```

//take back to display pages
header("Location:display.php");
exit();
}
?>
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Create a goal|welcome</title>
    <link rel="stylesheet" href="css/master.css">
    <script defer src="script/main.js"></script>
  </head>
  <body>
    <!--Content of the page-->
    <div class="container">
      <!--create a goal page-->
      <form id="createGoal" name="createGoal" class="createGoal"
action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
        <h3>Add Goal</h3>
        <fieldset>
          <input type="text" id="goal" name="goal" placeholder="Type your
goal here" required="">
        </fieldset>
        <fieldset>
          <select id="category" name="category">
            <option value="">--Please choose category</option>
            <option value="travel">Travel</option>
            <option value="entertainment">Entertainment</option>
            <option value="food">Food</option>
            <option value="health">Health</option>
            <option value="education">Education</option>
            <option value="sports">Sports</option>
            <option value="nature">Nature</option>
          </select>
        </fieldset>
        <fieldset>
          <input type="text" name="location" placeholder="Enter a location"
required="">
        </fieldset>
        <fieldset>
          <input type="date" id="date" name="date" placeholder="Enter

```

```

target date" required="">
    </fieldset>
    <fieldset>
        <textarea id="description" name="description" rows="8" cols="60"
placeholder="Describe your goal" required=""></textarea>
    </fieldset>
    <fieldset>
        <input type="checkbox" name="share" value="1">
        <label for="share">Share with friends</label>
    </fieldset>
    <fieldset>
        <button type="submit" name="button" id="save_createGoal" data-
submit="...sending">Save</button>
        <button type="reset" name="button" id="cancel_createGoal" data-
submit="...canceling">Cancel</button>
    </fieldset>
    <p class="copyright">Designed by Mansur & Ahmed</p>
</form>
</div>
</body>
</html>

```

```

<!--=====MYLIST PAGE=====
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the myList page, that contains:
    I- Html part, that list the bucketlist of the user in a
table, and allows edit/delete fucntionality
    II- PHP code that lists all the content of the user's
bucketlist from the database table in the Html table.
-->

```

```

<?php
    include('includes/database.php'); //requires database connection
    //connect to the database
    $pdo = connectDB();
    //session_start
    session_start();
    //check if the user has logged in
    if(!isset($_SESSION['user'])){
        //user has not logged
        header("Location:Login.php"); //redirect the user back to login
        exit();
    }

```

```

//get all the values in the users_lists
$stmtement = $pdo->prepare("SELECT * FROM users_lists WHERE user_id =
?");
$stmtement->execute([$SESSION['userid']]);

//a while loop to store all the results from the above query in an array
while ($results = $statement->fetch()){
    if(!empty($results)){
        //an array list to store all the content of the bukcetlist
        $List[] = array('id' => $results['id'], 'goal' => $results['goal'],
'category' => $results['category'], 'description' =>
$results['description'], 'location' => $results['location'], 'goalDate' =>
$results['goalDate']));}
    }
?><!--END OF PHP CODE-->

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Display|welcome</title>
    <link rel="stylesheet" href="css/master.css">
    <script defer src="script/main.js"></script>
</head>
<body id="myList">
    <div>
        <div class="list">Hello <?php echo $SESSION['user']?>!<br/>
        Your list information can be found in the table bellow</div>
        <table id="table">
            <tr class="tr1">
                <td>Goal</td>
                <td>Category</td>
                <td>Description</td>
                <td>Location</td>
                <td>Date</td>
                <td>Edit/Delete</td>
            </tr>
            <tr class="tr2">
                <!--foreach loop to list all the lists of the user and their
content-->
                <?php foreach ($List as $myList): ?>
                    <td><?php echo $myList['goal']; ?></td>

```



```

        <td><?php echo $myList['category']; ?></td>
        <td><?php echo $myList['description']; ?></td>
        <td><?php echo $myList['location']; ?></td>
        <td><?php echo $myList['goalDate']; ?></td>
        <?php $id = $myList['id']; ?><!--a php code to store the id of each
list in order to allow edit/delete fucntionalty-->
        <td><button name="edit"><a href="edit.php?id=<?=$id
?>">Edit</a></button> &nbsp;
        <button name="delete"><a href="deleteList.php?id=<?=$id?>"
onclick="return confirm('Are you sure you want to
delete?')">Delete</a></button>
        </td>
    </tr>
<?php endforeach; //end of foreach loop?>
</table>
</div>
</body></html>

```

```

<!--=====SETTING PAGE=====-->
<!--

```

Student names: Ahmed Ahmed & Mansur Mansur

COURSE: COIS 3420H

PURPOSE: This is the Setting page. It contains:

- 1- A HTML form in which a user can edit their account, such as editing username, passowrd etc.
- 2- A link to completely delete thier account.

```
-->
```

```
<?php //START OF PHP CODE
```

```
include ("includes/database.php"); // Include library file
```

```
session_start(); //start the session
```

```
$pdo = connectDB(); // Get database connection
```

```
//An arry to store errors
```

```
$PswErrors = []; //for password errors
```

```
$userErrors = []; // for username errors
```

```
$PswMErrors = []; //for password mismatch errors
```

```
//Check if the user is not logged in
```

```
if ( !isset($_SESSION['user']) ) {
```

```
// if, then redirect to the login page
```

```
header("Location:Login.php");
```

```
exit();
```

```

    }

    //check if the user wants to delete the account
    if (isset($_POST['delete'])) {
        //prepare and execute the delete query of the user based with the userid
        in session
        $query = "DELETE FROM users WHERE id = ?";
        $stmt = $pdo->prepare($query);
        $stmt->execute([$_SESSION['userid']]);

        // Unset and destroy user's session
        session_unset();
        session_destroy();

        // redirect the user to main index page
        header("Location: ./index.php");
        die();
    }

    //Get the info of the user in session from the database
    $stmt = $pdo->prepare("SELECT * FROM users WHERE id =
?");

    $stmt->execute([$_SESSION['userid']]);
    $info = $stmt->fetchAll();

    if (isset($_POST['update']))
    {

        /* Get everything from $_POST */
        $username = $_POST['username'] ?? NULL; //for Username
        $current_psw = $_POST['current_psw'] ?? NULL; //for
the user's existing password
        $psw = $_POST['psw'] ?? NULL; //for the new
password

        // compare the username given with the ones in the database table
        $stmt = $pdo->prepare("SELECT * FROM users WHERE username = ?");
        $stmt->execute([$_POST['username']]);
        $result = $stmt->fetch();

        // check if the username is already in use

```

```

        if (!empty($result) && $result['id'] != $_SESSION['userid']) {
            // set the flag to be false if the username is in use
            array_push($userErrors,
"username already in use"); //allow only unique username
        }

        //=====USERNAME
VALIDATION=====//
        /*
            Validate Username with the following conditions:
            1- Do not register a username with less than 3
character and more than 10 characters
            2- Do not allow usermae with special characters
other than the regualr ones and numbers
            3- Do not allow a username that has already been
in use to registred again
        */
        if (!( strlen($_POST['username']) >= 3 &&
strlen($_POST['username']) <= 10 && preg_match("/^[a-zA-Z0-9]*$/",
$_POST['username']) )) array_push($userErrors, "invalid username");

        //===== PASSWORD
VALIDATION=====//
        /*
            For the password to be strong enough it should have the
following conditons before it can be accepted as a valid password:
            1- It should be atleast 8 characters long
            2- It must contain at least 1 Upercase letter
            3- It must contain at leaset 1 lowercase Letter
            4- It must contain at least 1 number
            5- It must contain at least 1 special character
        */
        if (strlen($psw) < '8') array_push ($PswErrors, "Your
Password Must Contain At Least 8 Characters");
        elseif(!preg_match("#[A-Z]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Uppercase Letter");
        elseif(!preg_match("#[a-z]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Lowercase Letter");
        elseif(!preg_match("#[0-9]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Number");
        elseif(!preg_match('@[^\w]@', $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 special character e.g
'!, @ etc.'");

```

```

        // Validate Password Confirm (equals password)
        if (!( $psw === $_POST['psw_repeat']
    ))array_push($PswMErrors, "The passowrds should match");

    //=====UPDATE USER
ACCOUNT=====//
    /*
    The following piece of
code processes log in process by:

    =>First ensuring there is no valiation errors

    =>Update the user's informaiton and store in the database

    =>Display a successful message and redirect the user to
the index page

    */
    if (count($PswErrors) == 0 &&
count($userErrors) == 0 && count($PswMErrors) == 0 ){

        // Hash Password
        $HashPsw =
password_hash($_POST['psw'], PASSWORD_DEFAULT);

        // UPDATE username in the users table.
        $stm = "UPDATE users SET username = ? WHERE id =
?";

        $pdo->prepare($stm)-
>execute([$_POST['username'], $_SESSION['userid']]);

        // UPDATE password in the
users table.

        $stm2 = "UPDATE users SET
password = ? WHERE id = ?";

        $pdo->prepare($stm2)-
>execute([$HashPsw, $_SESSION['userid']]);

        //display a successful mesage then redirect to the logging
page

```

```

        $_SESSION['message'] = "Hi there! Your account has been
successfully updated😊!";
        // Redirect the user to index page
        header("Location: ./success.php");
        die();
    }
}

?><!--END OF PHP CODE-->
<!DOCTYPE html>
<html lang="en" dir="ltr">
    <head>
        <meta charset="utf-8">
        <title>setting|welcome</title>
        <link rel="stylesheet" href="css/master.css">
        <script defer src="script/main.js"></script>
    </head>
    <body>
        <div class="container">
            <div class="bar1">
                <div>
                    
                </div>
                <div>
                    <div>
                        <p class="username"> <?php echo
$_SESSION['user']; ?> </p>
                        <h2>Setting</h2>
                    </div>
                    </div>
                    <div id="edit">
                        <p class="editAccount">Edit Account</p>
                    </div>
                    <div id="delete">
                        <p class="deleteAccount">Delete Account</p>
                    </div>
                </div>
            <div class="container">

                <div class="container">

```

```

        <!-- update account form -->
        <form id="updateForm" action="#" method="post">
            <h3 id="edit">Edit Account</h3>
            <!--echo out username
and passowrd ERRORS-->

            <span class='error'
id="UsernameError"></span>

            <!--echo out username ERRORS-->
            <?php foreach ($userErrors as
$error): ?>

                <li id="error"><?= $error ?></li>
            <?php endforeach; ?>
            <fieldset>
                <label
for="username">Username</label>

                <input id="name" type="text"
placeholder="username*" name="username" required="">
            </fieldset>

            <span class='error'
id="PasswordError">

                <!--echo
out password strength ERRORS-->

                <?php foreach ($PswErrors as
$error): ?>

                    <li id ="error"><?=
$error ?></li>

                <?php endforeach; ?>
                <fieldset>
                    <label for="psw"> new password:
</label>

                    <input id="psw" type="password"
placeholder="password*" name="psw" required="" >
                </fieldset>

                <span class='error'
id="PasswordError">

                    <!--echo out password strength ERRORS-->
                    <?php foreach ($PswMErrors as $error): ?>
                        <li id ="error"><?= $error ?></li>
                    <?php endforeach; ?>
                    <fieldset>
                        <label for="psw_repeat">confirm

```

```

password: </label>
                <input id="psw" type="password"
placeholder="confirm password*" name="psw_repeat" required="" >
                </fieldset>
            <fieldset>
                <button type="submit" name="update"
value="submit">Update</button>
            </fieldset>
        </form>
    </div>
    <div class="container">
        <!-- update account form -->
        <form id="deleteForm" action="#"
method="post">
            <h3 id="delete">Delete Account</h3>
            
            <fieldset>
                <button type="submit"
name="delete" onclick="return confirm('Are you sure you want to delete your
account? This action is irreversible!')">Delete Account</button>
            </fieldset>
        </form>
    </div>
</div>
</body>
</html>

```

```

<!--===== LOGOUT PAGE =====-->
<!--
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the log Out page that allows the user to log output
their account.
-->
<?PHP
    // Starts session
    session_start();
    // Unsets the session variables
    session_unset();
    // Destroys session
    session_destroy();
    // Redirects user to the index page

```

```

    header('Location: index.php');
?>

<!--=====RESET PAGE=====
Makes sure only the user with the reset link can access the RESET PASSWORD
PAGE
-->
<?php //START OF PHP CODE
include ("includes/database.php"); // Include library file
session_start(); //start the session
$pdo = connectDB(); // Get database connection

//An array to store errors
$PswErrors = []; //for password errors
$emErrors = []; // for username errors
$PswMErrors = []; //for password mismatch errors

if (isset($_GET['email']) && !empty($_GET['email']))
{
    // Select the list that is going to be edited
    $stmt = $pdo->prepare('SELECT * FROM users WHERE email = ?');
    $stmt->execute([$_GET['email']]);
    $results = $stmt->fetchAll();

    if(!empty($results)){

        $_SESSION['email'] = $_GET['email'];
        header("Location: resetPassword.php");
        die();
    }
}
else {
    $_SESSION['message'] = "Invalid URL access";
    header("Location: error.php");
    die();
}
?>

<!--=====RESET PASSWORD PAGE=====
This page allows the user to create a new password
-->
<?php //START OF PHP CODE
include ("includes/database.php"); // Include library file

```



```

session_start(); //start the session
$pdo = connectDB(); // Get database connection

//An array to store errors
$PswErrors = []; //for password errors
$codErrors = []; //for validation code
$PswMErrors = []; //for password mismatch errors

//boolean to verify email
$validEmail = false;
if (isset($_POST['submit']))
{
    /* Get everything from $_POST */
    $code = $_POST['code'] ?? NULL; //for the user's
existing password
    $psw = $_POST['psw'] ?? NULL; //for the new
password

    // compare the code given with the ones in the database table
    $stmt = $pdo->prepare("SELECT * FROM reset WHERE reset_code = ?");
    $stmt->execute([$_POST['code']]);
    $result = $stmt->fetch();

    //verify the code
    if($result['reset_code'] != $_POST['code']){
        array_push($codErrors, "Invalid code");
    }

    //===== PASSWORD
VALIDATION=====//
    /*
        For the password to be strong enough it should have the
        following conditons before it can be accepted as a valid password:
        1- It should be atleast 8 characters long
        2- It must contain at least 1 Upercase letter
        3- It must contain at leaset 1 lowercase Letter
        4- It must contain at least 1 number
        5- It must contain at least 1 special character
    */
    if (strlen($psw) < '8') array_push ($PswErrors, "Your

```

```

Password Must Contain At Least 8 Characters");
        elseif(!preg_match("#[A-Z]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Uppercase Letter");
        elseif(!preg_match("#[a-z]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Lowercase Letter");
        elseif(!preg_match("#[0-9]+#", $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 Number");
        elseif(!preg_match('@[^\w]@', $psw)) array_push
($PswErrors, "Your Password Must Contain At Least 1 special character e.g
'!, @ etc.'");

        // Validate Password Confirm (equals password)
        if (!( $psw == $_POST['psw_repeat']
))array_push($PswMErrors, "The passowrds should match");

//=====UPDATE USER
ACCOUNT=====//
/*
The following piece of
code processes log in process by:

=>First ensuring there is no valiation errors

=>Update the user's informaiton and store in the database

=>Display a successful message and redirect the user to
the index page

*/
        if (count($PswErrors) == 0 &&
count($codErrors) == 0 && count($PswMErrors) == 0) {

// Hash Password
$HashPsw =
password_hash($_POST['psw'], PASSWORD_DEFAULT);

// UPDATE username in the users table.
$stmt = "UPDATE users SET password = ? WHERE email =
?";

        $pdo->prepare($stmt)->execute([$HashPsw,
$_SESSION['email']]);

```

```

        //display a successful message then redirect to the login
page
        $_SESSION['message'] = "Hi there! You have reset your
password successfully😊!";
        // Redirect the user to index page
        header("Location: ./success.php");
        die();
    }
}

```

```

?><!--END OF PHP CODE-->
<!doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>Reset Password</title>
        <link rel="stylesheet" href="css/master.css">
    </head>
    <body>
        <div class="container">
            <!-- Login form -->
            <form id="loginForm" action="resetPassword.php"
method="post">
                <h3>Reset Password </h3>
                <!--echo out username and password ERRORS-->
                <?php foreach ($codErrors as $error): ?>
                    <div id="error"><?= $error ?></div>
                <?php endforeach; ?>
                <fieldset>
                    <input type="text" placeholder="code"
name="code" required="" >
                </fieldset>
                <!--echo out username and password ERRORS-->
                <?php foreach ($PswErrors as $error): ?>
                    <div id="error"><?= $error ?></div>
                <?php endforeach; ?>
                <fieldset>
                    <input type="password" placeholder="new
password*" name="psw" required="" >

```

```

        </fieldset>
        <!--echo out username and passowrd ERRORS-->
        <?php foreach ($PswMErrors as $error): ?>
            <div id="error"><?= $error ?></div>
        <?php endforeach; ?>
        <fieldset>
            <input type="password" placeholder="confirm password*"
name="psw_repeat" required="" >
        </fieldset>

        <div class="Logintext"></div>
        <fieldset>
            <button type="submit" name="submit"
value="submit">submit</button>
        </fieldset>
        <p class="copyright">Designed by Mansur &
Ahmed</p>

    </form>
</div>
</body>
</html>

<!--=====EDIT/UPDATE USER ACCOUNT PAGE=====
Student names: Ahmed Ahmed & Mansur Mansur
COURSE: COIS 3420H
PURPOSE: This is the UPDATE page. It contains:
    1- A HTML which shows form in which the user can fill in
order to edit an existing list
    2- A PHP code that updates the selected list with the
information provided in the edit form

-->
<?php
    include('includes/database.php'); //requires database connection
    $pdo = connectDB(); // Get database connection

    //session_start
    session_start();
    //check if the user has logged in
    if(!isset($_SESSION['user'])){
        //user has not logged
        header("Location:Login.php"); //redirect the user back to login
        exit();
    }

```

```

//if already logged in and clicked save button
if(isset($_POST['update'])){
    //get all the values from the post
    $goal = $_POST['goal'] ?? NULL;
    $category = $_POST['category'] ?? NULL;
    $location = $_POST['location'] ?? NULL;
    $description = $_POST['description'] ?? NULL;
    $share = $_POST['share'] ?? NULL;
    $date = $_POST['date'] ?? NULL;

    // UPDATE Goal in the users_list table.
    $stm = "UPDATE users_lists SET goal = ? WHERE id = ?";
    $pdo->prepare($stm)->execute([$_POST['goal'], $_SESSION['list']]);

    // UPDATE Goal in the users_list table.
    $stm = "UPDATE users_lists SET category = ? WHERE id = ?";
    $pdo->prepare($stm)->execute([$_POST['category'], $_SESSION['list']]);

    // UPDATE location in the users_list table.
    $stm = "UPDATE users_lists SET location = ? WHERE id = ?";
    $pdo->prepare($stm)->execute([$_POST['location'], $_SESSION['list']]);

    // UPDATE description in the users_list table.
    $stm = "UPDATE users_lists SET description = ? WHERE id = ?";
    $pdo->prepare($stm)->execute([$_POST['description'], $_SESSION['list']]);

    // UPDATE privacy setting in the users_list table.
    $stm = "UPDATE users_lists SET share = ? WHERE id = ?";
    if ($_POST['share'] == NULL){
        //if the user opts not to share, make it private by inserting 0
        $pdo->prepare($stm)->execute([0, $_SESSION['list']]);
    }else{
        //if the users wishes to share, make it public
        $pdo->prepare($stm)->execute([$_POST['share'], $_SESSION['list']]);
    }

    // UPDATE Date in the users_list table.
    $stm = "UPDATE users_lists SET goalDate = ? WHERE id = ?";
    $pdo->prepare($stm)->execute([$_POST['date'], $_SESSION['list']]);
}

```

```

    header("Location: myList.php");
    exit();
}

?>
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Update List|welcome</title>
    <link rel="stylesheet" href="css/master.css">
    <script defer src="script/main.js"></script>
  </head>
  <body>
    <div class="container">
      <form id="createGoal" name="createGoal" class="createGoal"
action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
        <h3>Update Your List</h3>
        <fieldset>
          <input type="text" id="goal" name="goal" placeholder="Type your
goal here" required="">
        </fieldset>
        <fieldset>
          <select id="category" name="category">
            <option value="">--Please choose category</option>
            <option value="travel">Travel</option>
            <option value="entertainment">Entertainment</option>
            <option value="food">Food</option>
            <option value="health">Health</option>
            <option value="education">Education</option>
            <option value="sports">Sports</option>
            <option value="nature">Nature</option>
          </select>
        </fieldset>
        <fieldset>
          <input type="text" name="location" placeholder="Enter a location"
required="">
        </fieldset>
        <fieldset>
          <input type="date" id="date" name="date" placeholder="Enter
target date" required="">

```

```

        </fieldset>
        <fieldset>
            <textarea id="description" name="description" rows="8" cols="60"
placeholder="Describe your goal" required=""></textarea>
        </fieldset>
        <fieldset>
            <input type="checkbox" name="share" value="1">
            <label for="share">Share with friends</label>
        </fieldset>
        <fieldset>
            <button type="submit" name="update" id="save_createGoal" data-
submit="...sending">Save</button>
            <button type="reset" name="button" id="cancel_createGoal" data-
submit="...canceling">Cancel</button>
        </fieldset>
        <p class="copyright">Designed by Mansur & Ahmed</p>
    </form>
</div>
</body>
</html>

```

```

<!--=====SUCCESS PAGE=====
Displays all the successfull messages
-->
<?php
//Displays all the successfull messages
session_start();
?>
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>success|welcome</title>
    <link rel="stylesheet" href="css/master.css">
    <script defer src="script/main.js"></script>
</head>
<body>
    <div class="container">
        <form id="msgform">
            <h3>SUCCESS!</h3>
            <p>
                <?php
                if (isset($_SESSION['message']) AND !empty($_SESSION['message'])):

```

```

        echo $_SESSION['message'];
    else:
        header("Location: index.php");
    endif;
    ?>
</p>
<a class="msglink" href="Login.php">Login</a>
</form>
</div>
</body>
</html>

<!--=====ERROR PAGE=====
Displays all the unsuccessfull messages
-->
<?php
//Displays all the successfull messages
session_start();
?>
<!Doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>success|welcome</title>
    <link rel="stylesheet" href="css/master.css">
    <script defer src="script/main.js"></script>
</head>
<body>
    <div class="container">
        <form id="msgform">
            <h3>SUCCESS!</h3>
            <p>
                <?php
                if (isset($_SESSION['message']) AND !empty($_SESSION['message'])):
                    echo $_SESSION['message'];
                else:
                    header("Location: index.php");
                endif;
                ?>
            </p>
            <a class="msglink" href="Login.php">Login</a>
        </form>
    </div>

```



```

</body>
</html>

<?php
//Displays all the error messages
session_start();
?>
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>error|welcome</title>
  <link rel="stylesheet" href="css/master.css">
  <script defer src="script/main.js"></script>
</head>
<body>
  <div class="container">
    <form id="errorMsgForm">
      <h3>ERROR!</h3>
      <p>
        <?php
        if (isset($_SESSION['message']) AND !empty($_SESSION['message'])):
          echo $_SESSION['message'];
        else:
          header("Location: index.php");
        endif;
        ?>
      </p>
      <a class="msglink" href="index.php">HOME PAGE</a>
    </form>
  </div>
</body>
</html>

<!--=====FORGOT PAGE=====
VERIFIES AND SENDS A RESET EMAIL TO THE USER TO RESET THEIR PASSWORD
-->
<!-- REFERENCE:
https://www.youtube.com/watch?time_continue=1768&v=Pz5CbLqdGwM&feature=emb_
logo-->
<?php //start of the PHP code
include ("includes/database.php"); // Include library file
session_start(); //start the session

```

```

$pdo = connectDB(); // Get database connection

//An array to store errors
$emErrors = []; //for email errors

//a boolean to verify email address
$validEmail = false;
// Validate on submit (register)
if (isset($_POST['submit'])) {

    /* Get everything from $_POST */
    $email = $_POST['email'] ?? NULL; //for email address

    // compare the email given with the ones already in register in the
    table
    $query = "SELECT 1 email FROM users WHERE email = ?";
    $stmt = $pdo->prepare($query);
    $stmt->execute([$_POST['email']]);
    $result = $stmt->fetchAll();

    // check if the email is already in use
    if (!empty($result)) {
        // set the flag to be false if the email is already registered
        $validEmail = true;
    }

    //===== EMAIL
    VALIDATION=====//
    /*
        Validate email with the following conditions:
        1- Do not allow an invalid email, the email should be valid.
        2- Do not allow an already existing email to be registered
    */
    if (!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL))
    array_push($emErrors, "Invalid email address");
    if (!$validEmail){ array_push($emErrors, "There is no such email in
    your account");} //allow only unique emails

    // Check if there is no validation errors
    if (count($emErrors) == 0 ) {

```

```

        $_SESSION['message'] = "We sent a confirmation link to $email to
reset your password!";

        //sent an email with a confirmation link to reset password
        $to = $email;
        $hash = $user['password'];
        $subject = 'Reset Password Link';
        $body = '
            Hi there,

            Please use this code: reset123456 and click the link bellow to reset
your password:

https://loki.trentu.ca/~ahmedahmed/3420/project/reset.php?email='.\$email.'&
hash='.\$hash;

            mail($to, $subject, $body);

        // Redirect the user to index page
        header("Location: ./success.php");
        die();
    }
}
?> <!--END OF PHP CODE-->

<!Doctype html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>Forgot Password</title>
        <link rel="stylesheet" href="css/master.css">
    </head>
    <body>
        <div class="container">
            <!-- Login form -->
            <form id="loginForm" action="forgot.php" method="post">
                <p>Please enter your email to reset your
password</p>

                <!--echo out username and passowrd ERRORS-->
                <?php foreach ($emErrors as $error): ?>
                    <div id="error"><?= $error ?></div>
                <?php endforeach; ?>
            </form>
        </div>
    </body>
</html>

```

```

        <fieldset>
            <input type="text" placeholder="email*"
name="email" required="">
        </fieldset>
    </div class="Logintext"></div>
    <fieldset>
        <button type="submit" name="submit"
value="submit">submit</button>
    </fieldset>
    <p class="copyright">Designed by Mansur &
Ahmed</p>
</form>
</div>
</body>
</html>

```

<!--===== **EDIT LIST PAGE** =====>
 Student names: Ahmed Ahmed & Mansur Mansur
 COURSE: COIS 3420H
 PURPOSE: This is the EDIT LIST page. It contains only a PHP code
 that gets the id of a list from the bucketlist contents of the user,
 it then stored the id of the selected list in SESSION to
 allow the user to edit the selected list in the UPDATE PAGE.

Note: This page works behind the scene

```

-->
<?php
include ("includes/database.php"); // Include library file
session_start(); //start the session
$pdo = connectDB(); // Get database connection

//check if the user has logged in
if(!isset($_SESSION['user'])){
    //user has not logged
    header("Location:Login.php"); //redirect the user back to login
    exit();
}
// Check if the list id exists, for example update.php?id=1 will get the
list with the id of 1
if (isset($_GET['id'])) {
    // Select the list that is going to be edited
    $stmt = $pdo->prepare('SELECT * FROM users_lists WHERE id = ?');
    $stmt->execute([$_GET['id']]);
}

```

```

$results = $stmt->fetchAll();

//check if the id given returns anything other than NULL
if (!empty($results)) {
    //store the id of the list in SESSION
    $_SESSION['list'] = $_GET['id'];
    ///redirec the user to the UPDATE page
    header("Location: update.php");
}
}
?>

<!--=====DELETE LIST PAGE=====
Student names: Ahmed Ahmed & Mansur Mansur
        COURSE: COIS 3420H
        PURPOSE: This is the DELETE LIST page. It contains only a PHP code
that gets the id of a list from the bucketlist contents of the user,
                it then removes/deletes the list from the bucketlist

Note: This page works behind the scene
-->
<?php
    include('includes/database.php'); //requires database connection
    //connect to the database
    $pdo = connectDB();
    //session_start
    session_start();
    //check if the user has logged in
    if(!isset($_SESSION['user'])){
        //user has not logged
        header("Location:Login.php"); //redirect the user back to login
        exit();
    }
    // Check that the list id exists
    if (isset($_GET['id'])) {
        // Select the list that is going to be deleted
        $stmt = $pdo->prepare('SELECT * FROM users_lists WHERE id = ?');
        $stmt->execute([$_GET['id']]);
        $list = $stmt->fetchAll();
        if (empty($list)) {
            exit('There is no List with that ID!');

```

```

    }
    // Delete the content in thelist if it is not empty
    if(!empty($list)) {

        // Run the query to delete the list from the database
        $stmt = $pdo->prepare('DELETE FROM users_lists WHERE id = ?');
        $stmt->execute([$GET['id']]);

        //redirect the user to myList page
        header('Location: myList.php');
        exit(); //exit the program
    }
}
?>

<?php
include('includes/database.php'); //requires database connection
//connect to the database
$pdo = connectDB();
//session_start
session_start();
//check if the user has logged in
if(!isset($_SESSION['user'])) {
    //user has not logged
    header("Location:Login.php"); //redirect the user back to login
    exit();
}
// Check that the contact ID exists
if (isset($_GET['id'])) {
    // Select the record that is going to be deleted
    $stmt = $pdo->prepare('SELECT * FROM users_lists WHERE id = ?');
    $stmt->execute($_GET['id']);
    $contact = $stmt->fetchAll();
    if (empty($contact)) {
        exit('List doesn\'t exist with that ID!');
    }
    // Make sure the user confirms before deletion
    if(!empty($contact)) {
        // User clicked the "Yes" button, delete record
        $stmt = $pdo->prepare('DELETE FROM users_lists WHERE id = ?');
        $stmt->execute($_GET['id']);
        $msg = 'You have deleted the List!';
        header('Location: myList.php');
    }
}

```

```

        exit;
    }
}
?>

//Java Script CODE
"use strict";
//setting
const edit = document.querySelector('.editAccount');
const deleteAccount = document.querySelector('.deleteAccount');
const editform = document.querySelector('#updateForm');
const deleteForm = document.querySelector('#deleteForm');
const buttonUpdate = document.querySelector('#updateForm button');
const buttonDelete = document.querySelector('#deleteForm button');
const createPage = document.querySelector('#createPage');

// MAKE THE EDIT PAGE RESPONSIVE ON CLICK
edit.addEventListener('click', ev=>{
    ev.srcElement.parentElement.style.display = "none";
    edit.style.display = "none";
    editform.style.display = "block";
});
//REVEAL THE CONTENT OF THE DELETE PAE ON CLICK
deleteAccount.addEventListener('click', ev=>{
    ev.srcElement.parentElement.style.display = "none";
    deleteAccount.style.display = "none";
    deleteForm.style.display = "block";
});
//WHAT HAPPENS ONCE THE USER CLICKS THE UPDATE BUTTON ON THE EDIT PAGE
buttonUpdate.addEventListener('click', ev=>{
    editform.style.display = "none";
    edit.parentElement.style.display = "block";
    edit.style.display = "block";
});
// WHAT HAPPENS WHEN THE USER PRESSES THE DELETE ACCOUNT BUTTON
buttonDelete.addEventListener('click', ev=>{
    deleteForm.style.display = "none";
    deleteAccount.parentElement.style.display = "block";
    deleteAccount.style.display = " block";
});

//add event listener to menu icon

```

```

document.querySelector('#home').addEventListener('click',ev=>{
    createPage.style.display = "block";
});

<!--=====DATABASE CONNECTION PAGE=====-->
<?php
// Get the actual document and webroot path for virtual directories
$direx = explode('/', getcwd());
define('DOCR00T', "/$direx[1]/$direx[2]/"); // /home/username/
define('WEBROOT', "/$direx[1]/$direx[2]/$direx[3]/");
//home/username/public_html

function connectDB() {
    // Load configuration as an array.
    $config = parse_ini_file(DOCR00T."pwd/config.ini");
    $dsn =
    "mysql:host=$config[domain];dbname=$config[dbname];charset=utf8mb4";

    try {
        $pdo = new PDO($dsn, $config['username'], $config['password'], [
            PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
            PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
            PDO::ATTR_EMULATE_PREPARES => false,
        ]);
    } catch (\PDOException $e) {
        throw new \PDOException($e->getMessage(), (int)$e->getCode());
    }

    return $pdo;
}
?>

<!--=====METADATA PAGE=====-->
<meta charset="utf-8">
<title> <?php $PAGE_TITLE ?> </title>
<meta name="authors" content="Mansur Mansur && Ahmed Ahmed"> <!-- the
authors are : Mansur Mansur and Ahmed Ahmed -->
<meta name="description" content="This is a web application that allows
users create, organize and manage one (ormore) lists of bucket items. The
use can add date, details and a picture to thebucket list items when
completed.">
<link rel="stylesheet" href="reset.css">
<link rel="stylesheet" href="css/master.css">

```



```

<meta name="viewport" content="width=device-width, initial-scale=1">

<!--=====HEADER PAGE=====-->
<header>
    <div>iBu<span>cket</span> Application</div>

    <!-- Navigation bar -->
    <div id="header">
        <!-- menu for list, detail, About us -->
        <ul class="topMenu">
            <li class="active"><a href="./index.php">Home</a></li>
            <li><a href="./SignUp.php">Register</a></li>
            <li><a href="./Login.php">Log In</a></li>
        </ul>
    </div>
</header>

<!--=====FOOTER PAGE=====-->
<footer>
    <!-- class that contains the social media platforms-->
    <div class="row">
        <div>
            <form action="https://www.facebook.com" method="post">
                <input type="image" src="img/facebooklogo.png" alt="facebook">
            </form>
        </div>
        <div>
            <form action="https://www.youtube.com" method="post">
                <input type="image" src="img/youtubelogo.png" alt="youtube">
            </form>
        </div>
        <div>
            <form action="https://www.twitter.com" method="post">
                <input type="image" src="img/twitterlogo.png" alt="twitter">
            </form>
        </div>
        <div>
            <form action="https://www.linkedin.com" method="post">
                <input type="image" src="img/LinkedinIcon.png" alt="linkedin">
            </form>
        </div>
    </div>
    <p class="copyright">&copy;Bucket list 2020</p>

```

</footer>