

3420 Assignment #4 - Fall 2023

Name(s):Farhana Zahan, Parmeet Kaur, Manpreet Kaur Student Id: 0691212,0732389, 0719583

Live Loki link(s):https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/index.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/search.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/public.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/login.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/register.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/forgot.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/reset.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/checkusername.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/delete_account.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/delete-item.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/edit-item.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/edit_account.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/logout.php
https://loki.trentu.ca/~farhanazahan/3420/ASSIGNMENTS/Assignment_4/view-item.php

Rubric

Component	Grade
Edit List Validation	/4
Register Validation	/4
Delete confirmation	/3
Details modal	/5
Copy Public Link to Clipboard	/3
*Unique Username	/3
*Password Strength	/3
*Show Password	/3
*Limiting Description Field	/3
Star Rating	/3
Code Quality (tidyness, validity, efficiency, etc)	/4
Documentation	/3
Testing	/3

Component	Grade
Bonus	/3.5
Deductions (readability, submission guidelines, originality)	
Total	/35

Things to consider for Bonus Marks (if any)

Code & Testing

Put your code and screenshots here, **with proper heading organization**. You don't need to include html/php code (or testing) for any pages that aren't affected by your javascript for this assignment.

Register

PHP/JS

```
<?php

$errors = array();

// Get user input
$username = $_POST['username'] ?? "";
$name = $_POST['name'] ?? "";
$email = $_POST['email'] ?? "";
$password = $_POST['password'] ?? "";
$confirmPassword = $_POST['confirm_password'] ?? "";
$pswd_match = $_POST['pswd_match'] ?? "";
$list_name = $_POST['list_name'] ?? "";
$list_desc = $_POST['list_desc'] ?? "";
$public_visibility = $_POST['public_visibility'] ?? "N";

// Insert user data into the database
require("./includes/library.php");
$pdo = connectDB();
//build query

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

// Validate that the user has entered a name (since names are a string that
// could be just about anything, its validation is simple)
    if (strlen($name) === 0) {
        $errors['name'] = true;
    }
    if (strlen($username) === 0) {
        $errors['username'] = true;
    }
// Ensure that the user has entered a valid email address
}
```

```
if (filter_var($email, FILTER_VALIDATE_EMAIL) === false) {
    $errors['email'] = true;
}

if (strlen($password) === 0 ){
    $errors['password']= true;
} elseif (strlen($password) < 8) {
    $errors['password_length'] = true;
} elseif (!preg_match('/[a-z]/', $password)) {
    $errors['password_lowercase'] = true;
} elseif (!preg_match('/[A-Z]/', $password)) {
    $errors['password_uppercase'] = true;
} elseif (!preg_match('/\d/', $password)) {
    $errors['password_digit'] = true;
} elseif (!preg_match('/[@$!%*?&]/', $password)) {
    $errors['password_special'] = true;
}

if (strlen($confirmPassword) === 0 ){
    $errors['confirm_password']= true;
}

if ($password!== $confirmPassword && strlen($password) !== 0 &&
strlen($confirmPassword) !== 0){
    $errors['pswd_match']= true;
}

if (strlen($list_name) === 0) {
    $errors['list_name'] = true;
}

if (strlen($list_desc) === 0) {
    $errors['list_desc'] = true;
}

if (count($errors) === 0) {
    // Hash the password

    $hashedPassword = password_hash($password, PASSWORD_DEFAULT);

    $query = "INSERT INTO assn_users(username, password, name, email)
VALUES (?, ?, ?, ?)";
    $stmt = $pdo->prepare($query);
    $stmt->execute([$username,$hashedPassword, $name, $email]);

    $query1 = "INSERT INTO user_lists(username, list_name, list_desc,
public_visibility)
VALUES (?, ?, ?, ?)";
    $stat = $pdo->prepare($query1);
    $stat->execute([$username,$list_name,$list_desc,$public_visibility]);

    // Then redirect:
    header("Location: login.php");
    exit;
}
```

```
}
```

?>

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Create Account</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
    <link rel="stylesheet" href="styles/main.css">
    <script defer src="./scripts/register.js"></script>
    <script defer src="./scripts/password.js"></script>
</head>

<body>
<?php include './includes/header.php' ?>

<main class="page-content">
    <form id="page-form" class="page-form" method="post">
        <div class="page-container">
            <div class="heading">
                <h2> Create Account</h2>
            </div>

            <!-- Username -->
            <div class="page-group">
                <label class="page-label" for="username">Username:</label>
                <input class="page-input" type="text" name="username" id="username"
id="username"
placeholder="Enter your username" value="<?= $username ?>">
                <span class="error <?= !isset($errors['username']) ? 'hidden' : '' ?>">Please enter your username.</span>
            </div>

            <!-- Name -->
            <div class="page-group">
                <label class="page-label" for="name">Name:</label>
                <input class="page-input" type="text" name="name" id="name"
placeholder="Enter your name" value="<?= $name ?>">
                <span class="error <?= !isset($errors['name']) ? 'hidden' : '' ?>">Please fill this field.</span>
            </div>

            <!-- Email -->
            <div class="page-group">
```

```
<label class="page-label" for="email">Email:</label>
<input class="page-input" type="email" name="email" id="email"
placeholder="Enter your email" value=<?= $email ?>
>
    <span class="error <?= !isset($errors['email']) ? 'hidden'
: '' ?>">Please enter your email.</span>
</div>

<!-- Password -->
<div class="page-group">
    <label class="page-label" for="password">Password: </label>
    <input class="page-input" type="password" name="password"
id="password"
        placeholder="Enter your password">
        <i class="fa fa-eye" aria-hidden="true"
id="clickPassword"></i>
        <span class="error <?= !isset($errors['password']) ? 'hidden'
: '' ?>">Please enter your password.</span>
    </div>
    <div id="password-strength-indicator">
        <span id="password-strength-indicator-text"></span>
    </div>

    <div class="page-group">

        <ul class="password-requirements">
            <li class="error <?= !isset($errors['password_length']) ? 'hidden'
: '' ?>">Password must be at least 8 characters long.</li>
            <li class="error <?= !isset($errors['password_lowercase']) ? 'hidden'
: '' ?>">Password must contain at least one lowercase letter.</li>
                <li class="error <?= !isset($errors['password_uppercase']) ? 'hidden'
: '' ?>">Password must contain at least one uppercase letter.</li>
                <li class="error <?= !isset($errors['password_digit']) ? 'hidden'
: '' ?>">Password must contain at least one digit.</li>
                <li class="error <?= !isset($errors['password_special']) ? 'hidden'
: '' ?>">Password must contain at least one special character (@$!%*?&).
            </li>
        </ul>

    </div>

    <!-- Confirm Password -->
    <div class="page-group">
        <label class="page-label" for="confirm_password">Confirm
        Password:</label>
        <input class="page-input" type="password"
name="confirm_password" id="confirm_password"
            placeholder="Confirm your password" >
            <span class="error <?= !isset($errors['confirm_password']) ? 'hidden'
: '' ?>">Please confirm your password.</span>
            <span class="error <?= !isset($errors['pswd_match']) ? 'hidden'
: '' ?>">Password does not match.</span>
    </div>
```

```
<div class = "page-divider">
    <!-- "In your list???" item -->
<div class="page-group">
    <h2 class="in-your-list">Include a Fresh Entry in Your
List</h2>
</div>
<!-- List Information Fieldset -->
<fieldset id="list-info-fieldset" class="page-fieldset">
    <legend class="page-legend"><strong>List Information</strong>
</legend>
    <!-- List Name -->
    <div class="page-group">
        <label class="page-label" for="list_name">List Name:
</label>
        <input class="page-input" type="text" name="list_name"
id="list_name"
            placeholder="Enter list name" value=<?= $list_name ?>">
        <span class="error <?= !isset($errors['list_name']) ? 'hidden' : '' ?>">You haven't specified the list name.</span>
    </div>
    <!-- List Description -->
    <div class="page-group">
        <label class="page-label" for="list_desc">List
Description:</label>
        <br>
        <textarea class="page-textarea" name="list_desc"
id="list_desc"
            placeholder="Enter list description" rows="5"
maxlength="2500"> <?= $list_desc ?> </textarea>
        <span class="error <?= !isset($errors['list_desc']) ? 'hidden' : '' ?>">Give the description for the list created.</span>
        <span id="charCounter">2500 characters
remaining</span>
    </div>
    <!-- Public Visibility Checkbox -->
    <div class="page-group">
        <input class="page-checkbox" type="checkbox"
name="public_visibility" id="public_visibility"
            value="Y" <?= $public_visibility === 'Y' ? 'checked'
: '' ?>>
        <label for="public_visibility">Make List Publicly
Viewable</label>
    </div>
</fieldset>
</div>
<!-- Display errors if any -->
<?php
if (!empty($errors)) {
    echo '<div class="error-message">';
    if (isset($errors['empty_fields'])) {
```

```

        echo 'Please fill in all fields.';
    }
    if (isset($errors['email_invalid'])) {
        echo 'Invalid email address.';
    }
    if (isset($errors['password_mismatch'])) {
        echo 'Passwords do not match.';
    }
    // Add more error messages as needed
    echo '</div>';
}
?>
<!-- Submit Button -->
<button class="page-button" type="submit"
name="submit">Register</button>

<div class="back-to-link">
    <h3><i class="fa fa-angle-right" aria-hidden="true"></i><a
href=". ./login.php">Back to Login</a></h3>
    </div>
</div>
</form>
</main>
<?php include './includes/footer.php' ?>
</body>

</html>

```

```

//register.js

document.addEventListener("DOMContentLoaded", function () {
    const usernameInput = document.getElementById("username");
    const xhr = new XMLHttpRequest();

    usernameInput.addEventListener("blur", (event) => {
        const errorExists = document.getElementById("username_error");
        if (errorExists) {
            errorExists.remove();
        }

        const usernameValue = usernameInput.value.trim(); // Trim to remove
leading/trailing spaces

        if (usernameValue !== "") {
            xhr.open("GET", `checkusername.php?
username=${encodeURIComponent(usernameValue)} `);
            xhr.addEventListener("load", (ev) => {
                if (xhr.status === 200) {
                    if (xhr.responseText === "error" || xhr.responseText === "true")
{

```

```

        const errorSpan = document.createElement("span");
        errorSpan.classList.add("error");
        errorSpan.id = "username_error";
        errorSpan.innerText = "Username already exists. Please
choose a different one.";
        usernameInput.insertAdjacentElement('afterend', errorSpan);
    }
} else {
    console.error("Connection Failed");
}
});
xhr.send();
}

});

const clickPassword = document.getElementById("clickPassword");
const passwordInput = document.getElementById("password");

clickPassword.addEventListener("click", (ev) => {
    passwordInput.type = (passwordInput.type === "password") ? "text" :
"password";
});

const descInput = document.getElementById("list_desc");
const count = document.getElementById("charCounter");
const maxChars = 2500;

descInput.addEventListener("input", function () {
    const remainingChars = maxChars - descInput.value.length;
    count.innerText = `${remainingChars} characters remaining`;

    if (remainingChars < 50) {
        count.style.color = "red";
    } else {
        count.style.color = "blue";
    }
});
});
});

```

```

// password.js
function updatePasswordStrengthIndicator(password) {
    const strengthTextElement = document.getElementById("password-strength-
indicator-text");

    if (password.length === 0) {
        strengthTextElement.textContent = '';
    } else {
        const strength = checkPasswordStrength(password);

        if (strength.length === 0) {
            strengthTextElement.textContent = 'Strong';
        }
    }
}

```

```
        strengthTextElement.style.color = 'green';
    } else if (password.length < 6) {
        strengthTextElement.textContent = 'Weak';
        strengthTextElement.style.color = 'red';
    } else if (password.length < 8) {
        strengthTextElement.textContent = 'Moderate';
        strengthTextElement.style.color = 'orange';
    } else {
        strengthTextElement.textContent = 'Strong';
        strengthTextElement.style.color = 'green';
    }
}

function checkPasswordStrength(password) {
    const requirements = [
        {
            condition: password.length >= 8,
            message: 'Password must be at least 8 characters long.'
        },
        {
            condition: /[a-z]/.test(password),
            message: 'Password must contain at least one lowercase letter.'
        },
        {
            condition: /[A-Z]/.test(password),
            message: 'Password must contain at least one uppercase letter.'
        },
        {
            condition: /\d/.test(password),
            message: 'Password must contain at least one digit.'
        },
        {
            condition: /[@$!%*?&]/.test(password),
            message: 'Password must contain at least one special character (@$!%*?&).'
        }
    ];

    return requirements.filter(req => !req.condition).map(req => req.message);
}

document.addEventListener("DOMContentLoaded", function () {
    const passwordInput = document.getElementById("password");

    passwordInput.addEventListener('input', function () {
        updatePasswordStrengthIndicator(this.value);
    });
});
```

Browser Testing

Windows

Password Strength

weak Password

Create Account

Username: Abay

Name: abhy

Email: abhy@gmail.com

Password: .. 

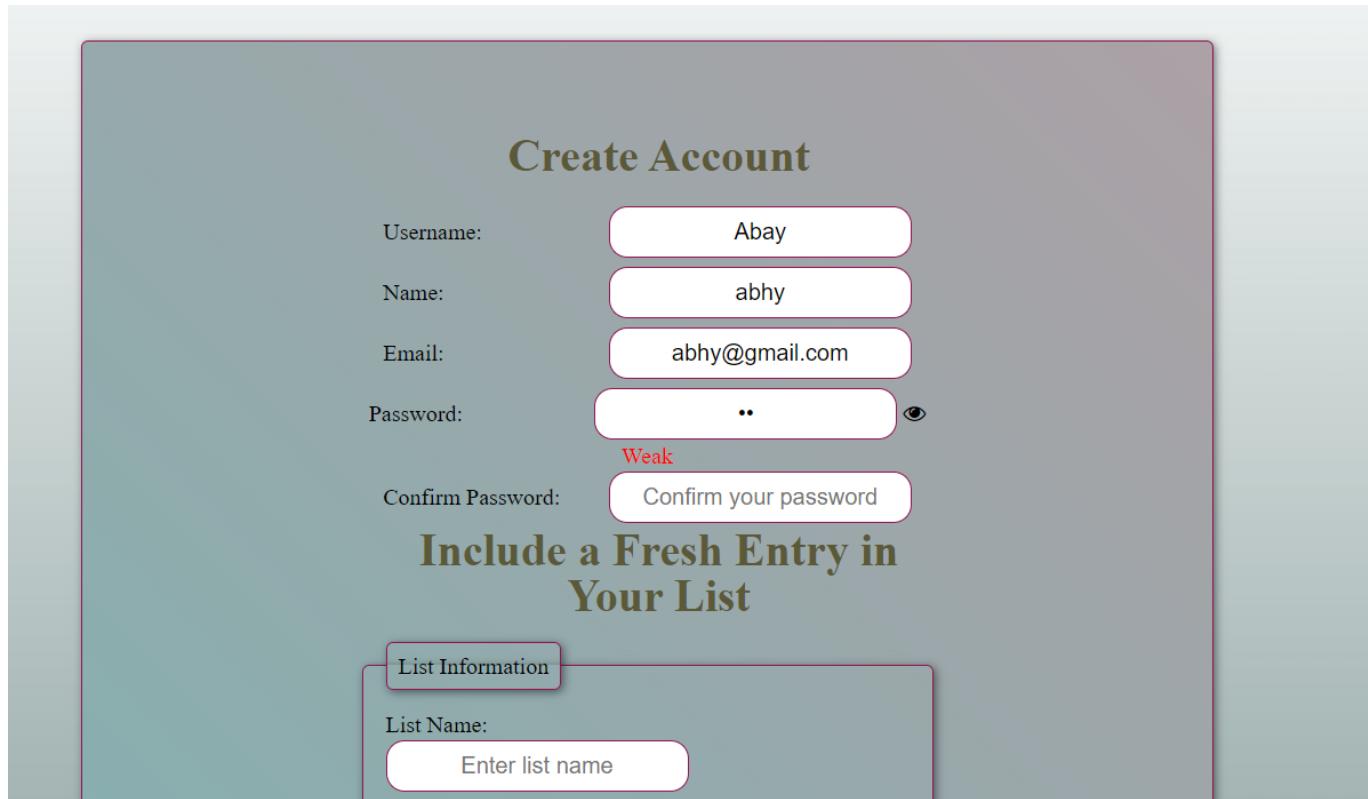
Weak

Confirm Password: Confirm your password

Include a Fresh Entry in Your List

List Information

List Name: Enter list name



Create Account

Username: Abay

Name: abhy

Email: abhy@gmail.com

Password: 

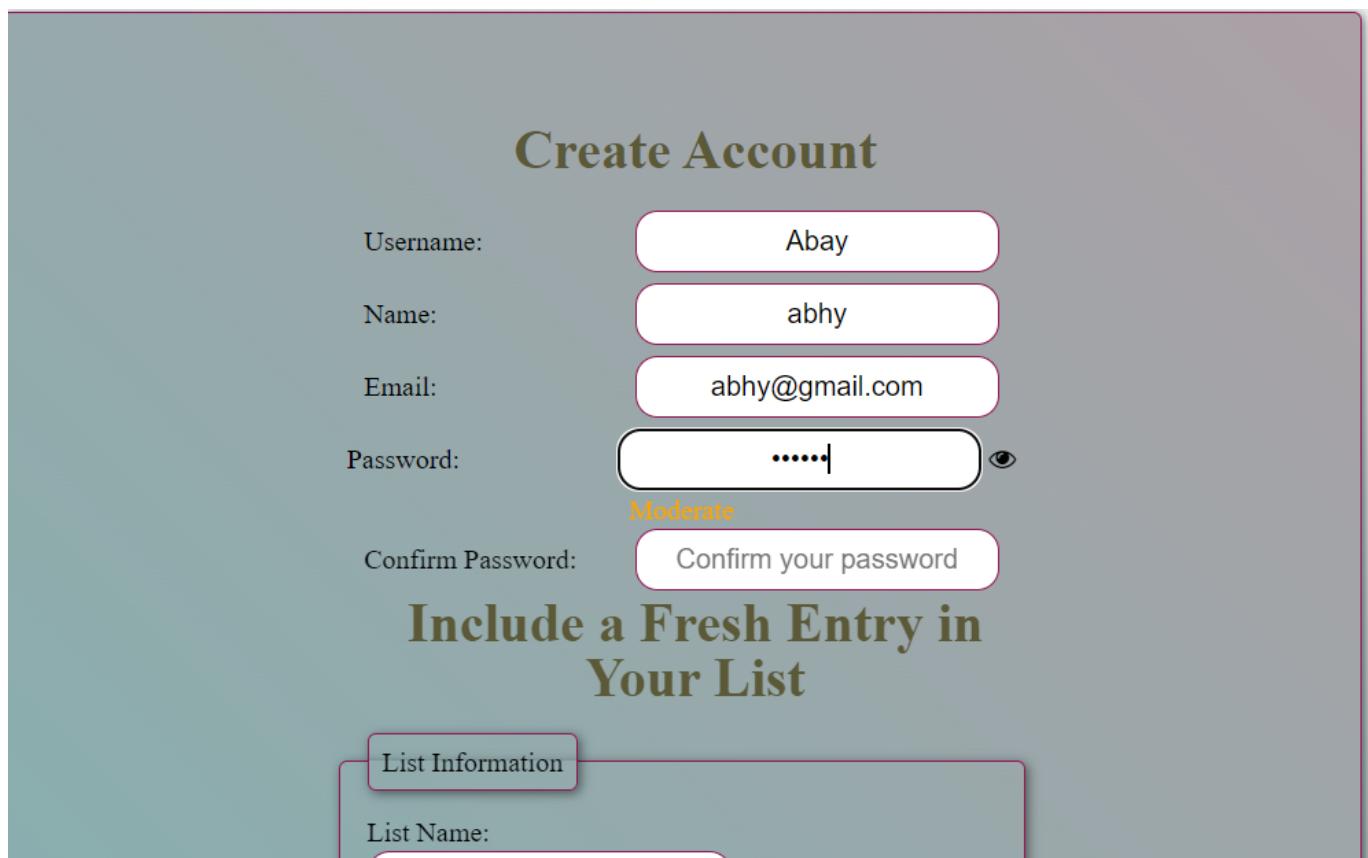
Moderate

Confirm Password: Confirm your password

Include a Fresh Entry in Your List

List Information

List Name:



Create Account

Username: Abay

Name: abhy

Email: abhy@gmail.com

Password: Strong (eye icon)

Confirm Password: Confirm your password

Include a Fresh Entry in Your List

List Information

List Name: Enter list name

Validation if the user exists

Create Account

Username: Username already exists. Please choose a different one.

Name:

Email:

Password: (eye icon)

Confirm Password:

Include a Fresh Entry in Your List

List Information

List Name:

List Description:

Extra options

	<input type="button" value="←"/>	<input type="button" value="→"/>	<input type="button" value="▼"/>	id	username	password	name	email	reset_token	reset_to
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	1	farha	\$2y\$10\$ceuNHoCkWVPHhq6Vc3dB3erBopJpAAasmkwz0ZFPqsr...	farha	farha@trentu.ca	e33845f4cc2394164eb825e50f8816	2023-11-11
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	6	Manpreet	\$2y\$10\$R9v21/g72LduPCERkPdYGeJrYaTFzBrclz3HZBYcTnk...	manpreet	manprkaur@trentu.ca	bcdacaa091e174ff31405960343867c1	2023-11-11
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	7	Afrin	\$2y\$10\$1.8eHtFMR1PSl02bihq.aANTQ0XcsjOWRFoFVSAR/...	afrin	Afrin@trentu.ca	NULL	NULL
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	9	farhana	\$2y\$10\$MEJoi0dzO_73FoluXInMtU5.l66o.bm4TZJlrHTrLue...	Farhana	farhanazahan@trentu.ca	01a0a2a0d1263b828f4337bedf5896	2023-11-11
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	11	Alif	\$2y\$10\$Ed5ETev7taPp8Q/toqaM6Om9KXAXz24NT5zyxdCrkb...	Alif12	Alif12@gmail.com	NULL	NULL
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	12	Asdf	\$2y\$10\$yCMCPI2uUn86sQJSrHCireS1jw3AwREDQ9bVOxG0AqU...	Asdf	Asdf@gmail.com	NULL	NULL
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	13	Alisa	\$2y\$10\$vHGVpHo.BamSHQYVW0ORH./ct2DSh9xtbF.zTyMFkp...	Alisa	Alisa@trentu.ca	NULL	NULL
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	14	ALI	\$2y\$10\$BKbd7XnruE0S/f1CzUGMbe8IAcTnj94QXSWQOPvvXQ...	ALI	ALI@gmail.com	NULL	NULL
<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	15	Anu	\$2y\$10\$1xpEoKGAAAaUNyBi8v9pPOyS5nDhdOJ86ZA.NG1.Rb5...	Anu	Anu@gmail.com	NULL	NULL

**Validation

Create Account

Username: Enter your username
Please enter your username.

Name: anu

Email: xsxsd@gmail.com
Please enter your email.

Password: Enter your password 

Password must be at least 8 characters long.

Confirm Password: Confirm your password
Password does not match.

Include a Fresh Entry in Your List

List Information

List Name: Enter list name
You haven't specified the list name.

List Description:
 2490 characters remaining

Make List Publicly Viewable

[Register](#)

> [Back to Login](#)

Details field limited to 2500 characters,

Strong

Password must be at least 8 characters long.

Confirm Password: ······
Password does not match.

Include a Fresh Entry in Your List

List Information

List Name:
You haven't specified the list name.

List Description:
 51 characters remaining

Make List Publicly
 Viewable

Register

> [Back to Login](#)

Password does not match.

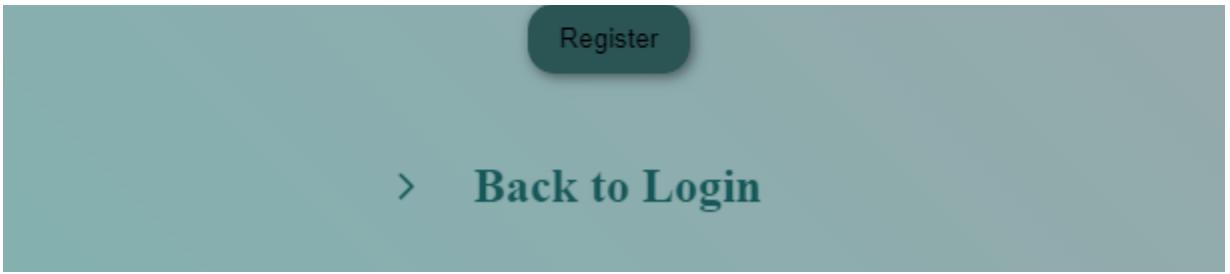
Include a Fresh Entry in Your List

List Information

List Name:
You haven't specified the list name.

List Description:
 47 characters remaining

Make List Publicly
 Viewable



Register

> Back to Login

Show Password

```
const clickPassword = document.getElementById("clickPassword");
const passwordInput = document.getElementById("password");

clickPassword.addEventListener("click", (ev) => {
    passwordInput.type = (passwordInput.type === "password") ? "text" :
    "password";
});
```

**Validation

Create Account

Username:	<input type="text" value="bob"/>
Name:	<input type="text" value="Enter your name"/>
Email:	<input type="text" value="Enter your email"/>
Password:	<input type="password" value="*****"/> Moderate
Confirm Password:	<input type="text" value="Confirm your password"/>

Include a Fresh Entry in Your List

Create Account

Username:	<input type="text" value="bob"/>
Name:	<input type="text" value="Enter your name"/>
Email:	<input type="text" value="Enter your email"/>
Password:	<input type="password" value="ffdfdf"/> Moderate
Confirm Password:	<input type="text" value="Confirm your password"/>

Include a Fresh Entry in Your List

EditList

PHP/Javascript

```
<?php
$errors = array();
// Start the session to check user authentication
session_start();
require("./includes/library.php");
$pdo = connectDB();

$category = $_POST['options'] ?? null;
$state = $_POST['state'] ?? null;
$starting_date = $_POST ['starting_date'] ?? null;
$completion_date = $_POST ['completion_date'] ?? null;

// Check if the user is logged in
if (!isset($_SESSION['username'])) {
    // Redirect to the login page if not logged in
    header("Location: login.php");
    exit();
```

```
}
```

```
$item_id = $_GET['id'];
echo "Received item_id: $item_id";
```

```
// Check if the item id is provided in the URL
if (isset($_GET['id'])) {
    $item_id = $_GET['id'];

    // Fetch item details from the database based on the item id
    $query = "SELECT * FROM user_items WHERE id = ?";
    $stmt = $pdo->prepare($query);
    $stmt->execute([$item_id]);
    $row = $stmt->fetch();

    // Check if the item belongs to the logged-in user
    if ($row) {
        // Fetch existing data
        $title = $row['item_name'];
        $description = $row['item_description'];
        $category = $row['category'];
        $state = $row['state'];
        $starting_date = $row['starting_date'];
        $completion_date = $row['completion_date'];

        // Handle form submission
        if (isset($_POST['submit'])) {
            $title = $_POST['title'] ?? "";
            $state = $_POST['state'] ?? "";
            $description = $_POST['description'] ?? "";
            $category = $_POST['options'] ?? "";
            $starting_date = $_POST['starting_date'] ? $_POST['starting_date'] :
null;
            $completion_date = $_POST['completion_date'] ?
$_POST['completion_date'] : null;

            $valid_state = ['Completed', 'In Progress', 'Not Yet Started'];
            if (!in_array($state, $valid_state)) {
                $errors['state'] = true;
            }
        }

        // Image handling
        $image = $_FILES['proof'];

        if ($image['size'] > 0) {
            // Check if the file is an image
            $allowedTypes = ['jpg', 'jpeg', 'png', 'gif'];
            $imageType = strtolower(pathinfo($image['name'],
PATHINFO_EXTENSION));

            if (in_array($imageType, $allowedTypes)) {
                // Read image data
                $imageData = file_get_contents($image['tmp_name']);
            } else {

```

```
        $errors['image'] = true;
    }
} else {
    // If no new image is uploaded, retain the existing image data
    $imageData = $row['image'];
    $imageType = $row['image_type'];
}

if ($state == 'Completed' && (!$starting_date || !$completion_date)) {
    $errors['missing_dates'] = true;
} elseif ($state == 'In Progress' && !$starting_date) {
    $errors['missing_start_date'] = true;
} elseif ($state == 'Completed' && strtotime($completion_date) <
strtotime($starting_date)) {
    $errors['wrong_date'] = true;
} else {
    // Update the item in the database
    $update_query = "UPDATE user_items SET item_name = ?,
item_description = ?, category = ?, state = ?, starting_date = ?, completion_date = ?,
image = ?, image_type = ? WHERE id = ?";
    $stmt_query = $pdo->prepare($update_query);

    if ($state == 'Not Yet Started') {
        $params = [$title, $description, $category, $state, null,
null, null, null, $item_id];
        if ($stmt_query->execute($params)) {
            // Handle image upload if necessary
            // Redirect to the main page after successful update
            header("Location: index.php");
            exit();
        } else {
            // Handle error
            echo "Error updating item.";
        }
    } else {
        $params = [$title, $description, $category, $state,
$starting_date, $completion_date, $imageData, $imageType, $item_id];
        if ($stmt_query->execute($params)) {
            // Handle image upload if necessary
            // Redirect to the main page after successful update
            header("Location: index.php");
            exit();
        } else {
            // Handle error
            echo "Error updating item.";
        }
    }
}
} else {
    // Redirect to the main page if the item doesn't belong to the user
    header("Location: index.php");
    exit();
}
```

```
    } else {
        // Redirect to the main page if no item id is provided
        header("Location: index.php");
        exit();
    }
?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Edit List Item</title>
    <link rel="stylesheet" href="styles/main.css">
    <script defer src="./scripts/edit_item.js"></script>
    <script defer src="./scripts/edit_des.js"></script>
</head>

<body>
    <?php include './includes/header.php' ?>

    <h2 id="edit-title">Edit Bucket List Item</h2>

    <div class="container">
        <?php include './includes/nav.php' ?>
        <main class="page-content">
            <!-- Error container for displaying general error messages -->
            <div id="error-container" class="error-container">
                <form id="edit-form" method="post" enctype="multipart/form-data">

                    <div class="form-element">
                        <label for="title">Title:</label>
                        <input type="text" name="title" id="title"
placeholder="Adventure and Travel" value="<?= $title ?>" >
                        <!-- Error span for title -->
                        <span class="error hidden" id="title-error"></span>
                    </div>

                    <fieldset class="form-element">
                        <legend>Category</legend>
                        <div>
                            <input type="radio" name="options"
id="travel_adventure" value="Travel_and_Adventure" <?= $category ===
'Travel_and_Adventure' ? 'checked' : '' ?>/>
                            <label for="travel_adventure">Travel and
Adventure</label>
                        </div>
                        <div>
                            <input type="radio" name="options"
id="unique_experiences" value="Unique_Experiences" <?= $category ===
'Unique_Experiences' ? 'checked' : '' ?>/>
                            <label for="unique_experiences">Unique
Experiences</label>
                        </div>
                    </fieldset>
                </form>
            </div>
        </main>
    </div>
</body>
```

```
</div>
<div>
    <input type="radio" name="options"
id="financial_milestones" value="Financial_Milestones" <?= $category ===
'Financial_Milestones' ? 'checked' : '' ?>/>
    <label
for="financial_milestones">FinancialMilestones</label>
</div>
<div>
    <input type="radio" name="options" id="career_success"
value="Career_Success" <?= $category === 'Career_Success' ? 'checked' : '' ?>/>
    <label for="career_success">Career Success</label>
</div>
</fieldset>

<fieldset class="form-element">
    <legend>State</legend>
    <div>
        <input type="radio" name="state" id="completed"
value="Completed" <?= $state == 'Completed' ? 'checked' : '' ?>>
        <label for="completed">Completed</label>
    </div>

    <div>
        <input type="radio" name="state" id="in-progress"
value="In Progress" <?= $state == 'In Progress' ? 'checked' : '' ?>>
        <label for="in-progress">In Progress</label>
    </div>
    <div>
        <input type="radio" name="state" id="not-started"
value="Not Yet Started" <?= $state == 'Not Yet Started' ? 'checked' : '' ?> >
        <label for="not-started">Not Started Yet</label>
    </div>
</fieldset>
<div class="form-element">
    <label for="starting-date">Starting Date: </label>
    <input type="date" id="starting_date"
name="starting_date" value="<?= $starting_date ?>">
    <!-- Error span for starting date -->
    <span class="error hidden" id="starting-date-error">
</span>
</div>

<div class="form-element">
    <label for="completion-date">Completion Date: </label>
    <input type="date" id="completion_date"
name="completion_date" value="<?= $completion_date ?>">
    <!-- Error span for completion date -->
    <span class="error hidden" id="completion-date-error">
</span>

</div>
<span class="error <?=
```

```

!isset($errors['missing_start_date']) ? 'hidden' : '' ?>">Please insert a valid
Start Date.</span>


```

```

//edit_js
"use strict";
document.addEventListener('DOMContentLoaded', function () {

```

```
const editForm = document.getElementById('edit-form');
editForm.addEventListener('submit', function (event) {
    if (!validateForm()) {
        event.preventDefault(); // Prevent form submission if validation fails
    }
});

function validateForm() {
    resetErrorMessages();

    const titleInput = document.getElementById('title');
    const descriptionInput = document.getElementById('description');
    const startDateInput = document.getElementById('starting_date');
    const completionDateInput = document.getElementById('completion_date');
    const stateInput = document.querySelector('input[name="state"]:checked');
    const proofInput = document.getElementById('proof');

    let isValid = true;

    if (titleInput.value.trim() === '') {
        displayError(titleInput, 'Please enter a title.');
        isValid = false;
    }

    if (descriptionInput.value.trim() === '') {
        displayError(descriptionInput, 'Please enter a description.');
        isValid = false;
    }

    if (!stateInput) {
        const stateRadioGroup = document.querySelector('input[name="state"]');
        displayError(stateRadioGroup, 'Please select a state.');
        isValid = false;
    }

    if (stateInput && stateInput.value === 'In Progress' &&
startDateInput.value.trim() === '') {
        displayError(startDateInput, 'Please enter a starting date.');
        isValid = false;
    }

    if (stateInput && stateInput.value === 'Completed' &&
(startDateInput.value.trim() === '' || completionDateInput.value.trim() === '')) {
        displayError(startDateInput, 'Please enter a starting date.');
        displayError(completionDateInput, 'Please enter a completion date.');
        isValid = false;
    }

    // Additional date validation
    const today = new Date().toISOString().split('T')[0];

    if (startDateInput.value.trim() !== '' &&
!isValidDateFormat(startDateInput.value)) {
        displayError(startDateInput, 'Invalid date format. Please use YYYY-MM-
```

```
DD. ');
    isValid = false;
}

if (completionDateInput.value.trim() !== '' &&
!isValidDateFormat(completionDateInput.value)) {
    displayError(completionDateInput, 'Invalid date format. Please use
YYYY-MM-DD.');
    isValid = false;
}

const todayDate = new Date(today);

if (startDateInput.value.trim() !== '') {
    const startDate = new Date(startDateInput.value);

    if (startDate > todayDate) {
        displayError(startDateInput, 'Starting date can\'t be after
today.');
        isValid = false;
    }
}

if (completionDateInput.value.trim() !== '') {
    const completionDate = new Date(completionDateInput.value);

    if (completionDate > todayDate) {
        displayError(completionDateInput, 'Completion date can\'t be
after today.');
        isValid = false;
    }
}

if (startDateInput.value.trim() !== '' &&
completionDateInput.value.trim() !== '') {
    const startDate = new Date(startDateInput.value);
    const completionDate = new Date(completionDateInput.value);

    if (startDate > completionDate) {
        displayError(startDateInput, 'Starting date can\'t be after
the completion date.');
        displayError(completionDateInput, 'Completion date can\'t be
before the starting date.');
        isValid = false;
    }
}

else if (stateInput.value === 'Completed') {
    // If state is 'Completed' and no completion date is provided, show an
error
    displayError(completionDateInput, 'Completion date is required for
Completed state.');
    isValid = false;
} else if (stateInput.value === 'In Progress' &&
```

```

startDateInput.value.trim() === '') {
    // If state is 'In Progress' and no starting date is provided, show an
error
    displayError(startDateInput, 'Starting date is required for In
Progress state.');
    isValid = false;
}

// Add more validation for proofInput if needed (e.g., file type, size)

return isValid;
}

function resetErrorMessages() {
const errorMessages = document.querySelectorAll('.error');
errorMessages.forEach(function (error) {
    error.classList.add('hidden');
});
}

function displayError(inputElement, errorMessage) {
const errorSpan = inputElement.nextElementSibling;
errorSpan.innerText = errorMessage;
errorSpan.classList.remove('hidden');
}

function isValidDateFormat(dateString) {
// Check if the date string is in the format YYYY-MM-DD
const dateRegex = /^\d{4}-\d{2}-\d{2}$/;
return dateRegex.test(dateString);
}
});
});
```

```

//edit_des.js
"use strict";
document.addEventListener("DOMContentLoaded", function () {
const descInput = document.getElementById("description");
const charCountSpan = document.getElementById("charCount");
const maxChars = 2500;

descInput.addEventListener("input", function () {
const remainingChars = maxChars - descInput.value.length;
charCountSpan.innerText = `Characters left: ${remainingChars}`;

if (remainingChars < 50) {
    charCountSpan.style.color = "red";
} else {
    charCountSpan.style.color = "blue";
}

// Validate word limit
});
```

```
const words = descInput.value.trim().split(/\s+/).filter(word =>
word.length > 0);
const wordCount = words.length;
const wordLimit = 200; // You can adjust the word limit

if (wordCount > wordLimit) {
    showError("description-error", "Description exceeds the word limit");
} else {
    resetError("description-error");
}
});

function showError(spanId, errorMessage) {
    const errorSpan = document.getElementById(spanId);
    errorSpan.innerText = errorMessage;
    errorSpan.style.color = "red";
}

function resetError(spanId) {
    const errorSpan = document.getElementById(spanId);
    errorSpan.innerText = "";
}
});
```

Browser Testing

editlist

Date validation

My List •

Search •

Public View •

Edit Bucket List Item

Title: List1

Category:

- Travel and Adventure
- Unique Experiences
- FinancialMilestones
- Career Success

State:

- Completed
- In Progress
- Not Started Yet

Starting Date: 12/28/2023 Starting date can't be after today.

Completion Date: 11/27/2023 Completion date can't be after today.

Description: stfddfdffff Characters left: 2489

Proof (Image Upload): Choose File No file chosen

Log Out Edit Account Delete Account

© FarhanaZahan & ManpreetKaur & ParmeetKaur

Inbox - farhanasatu454@gmail.com

Inbox (1) - farhanasatu@gmail.com

Edit List Item

+

My List •

Search •

Public View •

Edit Bucket List Item

Title: List1

Category:

- Travel and Adventure
- Unique Experiences
- FinancialMilestones
- Career Success

State:

- Completed
- In Progress
- Not Started Yet

Starting Date: 01/06/2024 Starting date can't be after today.

Completion Date: 01/06/2024 Completion date can't be after today.

Description: stfddfdffff Characters left: 2489

Proof (Image Upload): Choose File No file chosen

Log Out Edit Account Delete Account

© FarhanaZahan & ManpreetKaur & ParmeetKaur

DeleteList

PHP/JS

```
<?php
session_start();

// Check if the user is logged in
if (!isset($_SESSION['username'])) {
    header("Location: login.php");
    exit();
}

// Get item ID from the query string
```

```
$item_id = $_GET['id'] ?? null;

echo "Item ID: ";
var_dump($item_id);

// Check if item ID is provided
if (!$item_id) {
    echo "user doesn't exist";
    //header("Location: index.php");
    //exit();
}

// Connect to the database
require("./includes/library.php");
$pdo = connectDB();

// Fetch item details from the database based on the item ID
$query = "SELECT item_name FROM user_items WHERE id = ?";
$stmt = $pdo->prepare($query);
$stmt->execute([$item_id]);
$item = $stmt->fetch();

// Check if the item exists and belongs to the logged-in user
if (!$item || $item['item_name'] !== $_SESSION['username']) {
    // Redirect if the item doesn't exist or doesn't belong to the user

    // Delete the item from the database
    $delete_query = "DELETE FROM user_items WHERE id = ?";
    $delete_stmt = $pdo->prepare($delete_query);

    if ($delete_stmt->execute([$item_id])) {
        // Redirect to the main page after successful deletion
        echo "success deletion";
        //header("Location: index.php");
        //exit();
    } else {
        // Handle error if deletion fails
        echo "Error deleting item: ";
    }
}

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

?>
```

```
"use strict";
document.addEventListener('DOMContentLoaded', function () {
    const deleteItemLinks = document.querySelectorAll('.delete-item-link');
```

```
deleteItemLinks.forEach(link => {
    link.addEventListener('click', function (event) {
        event.preventDefault();
        const itemId = link.getAttribute('data-item-id');

        // Show a confirmation dialog
        const confirmed = confirmDelete();

        console.log('Delete link clicked for item ID:', itemId);

        if (confirmed) {
            // Redirect to delete-item.php with the item ID
            console.log('User confirmed deletion. Redirecting...');

            window.location.href = 'delete-item.php?item_id=' + itemId;
        } else {
            console.log('Deletion canceled by the user.');
        }
    });
});

function confirmDelete() {
    // Show a custom confirmation dialog
    console.log('Showing confirmation dialog...');

    return window.confirm('Are you sure you want to delete this item?');
}
});
```

Delete Item

Delete

Title:
List2
Category:
Unique_Experiences
State:
Completed
Starting Date:
2024-01-31
Completion Date:
2024-04-18
Description:

View

Edit

Delete

box (1) - farhanasatuu@gmail.com x Bucket List x +

han/3420/ASSIGNMENTS/Assignment_4/index.php

loki.trentu.ca says

Are you sure you want to delete this item?

OK Cancel

Unique_Experiences
State:
Completed
Starting Date:
2024-01-31
Completion Date:
2024-04-18
Description:

View

The screenshot shows a list item with the following details:

- Title: titke 1
- Category: Travel_and_Adventure
- State: Not Yet Started
- Starting Date:
- Completion Date:
- Description: sdsfsf

Below the item are three buttons: "View", "Edit", and "Delete".

Add New Item

The "Add New Item" form has the following fields:

- Title: (input field)
- Category: (button labeled "Category")
- Options for Category (radio buttons):
 - Travel and Adventure (selected)
 - Unique
 - Experiences
 - FinancialMilestones
 - Career
 - Success

Mainpage

PHP/JS

```
<?php

session_start();
// Check if the user is logged in and the session variable is set
//connecting using pdo using connectDB()
require("./includes/library.php");
$pdo = connectDB();

$errors = array();

// Check if the username is in the session (user is logged in)
if (isset($_SESSION['username'])) {
    $currentUsername = $_SESSION['username'];

    // Fetch user's lists based on the username
    $userListsQuery = "SELECT * FROM user_lists WHERE username = ?";
    $userListsStmt = $pdo->prepare($userListsQuery);
    $userListsStmt->execute([$currentUsername]);

    // Fetch the user's lists
}
```

```
$userLists = $userListsStmt->fetchAll(PDO::FETCH_ASSOC);

    // Output the user's lists
    foreach ($userLists as $list) {
        echo '<h3>' . htmlspecialchars($list['list_name']) . '</h3>';
        // Fetch and display items for each list if needed
    }
} else {
    echo "User not found.";
}

$state = $_POST['state'] ?? "";
$starting_date = $_POST['starting_date'] ?? null;
$completion_date = $_POST['completion_date'] ?? null;

// Fetch user lists for the dropdown
$userListsQueryForDropdown = "SELECT * FROM user_lists WHERE username = ?";
$userListsStmtForDropdown = $pdo->prepare($userListsQueryForDropdown);
$userListsStmtForDropdown->execute([$currentUsername]);

// Fetch the user's lists for the dropdown
$userListsForDropdown = $userListsStmtForDropdown->fetchAll(PDO::FETCH_ASSOC);

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Get form data
    $newItemTitle = $_POST["newItemTitle"];
    $category = $_POST["options"];
    $state = $_POST['state'] ?? "";
    $newItemDescription = $_POST["newItemDescription"] ?? "";
    $selectedList = $_POST["selectedList"] ?? "";
    $starting_date = $_POST['starting_date'] ?? null;
    $completion_date = $_POST['completion_date'] ?? null;

    // Check if a new list is being created
    if ($selectedList === "new_list") {
        $newListName = $_POST["newListName"];
        $newListDesc = $_POST["newListDescription"];
        $publicVisibility = isset($_POST["publicVisibility"]) ?
            $_POST["publicVisibility"] : 'N';

        // Insert the new list into the user_lists table
        $query = "INSERT INTO user_lists (username, list_name, list_desc,
public_visibility) VALUES (?, ?, ?, ?)";
        $stmt = $pdo->prepare($query);
        $stmt->execute([$currentUsername, $newListName, $newListDesc,
$publicVisibility]);

        // After creating the new list, fetch its ID
        $selectedList = $pdo->lastInsertId();
    }
}
```

```
if ($state == 'Completed' && (!$starting_date || !$completion_date)) {
    $errors['missing_dates'] = true;
} elseif ($state == 'In Progress' && !$starting_date) {
    $errors['missing_start_date'] = true;
} elseif ($state == 'Completed' && strtotime($completion_date) <
strtotime($starting_date)) {
    $errors['wrong_date'] = true;
}

// Image handling
$image = $_FILES['image'];

if ($image['size'] > 0) {
    // Read image data
    // Insert data into the database, including the image
    $query = "INSERT INTO user_items (list_id, title, category, state,
starting_date, completion_date, description, image, image_type) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)";
    $stmt = $pdo->prepare($query);

    if ($state == 'Not Yet Started') {
        // Insert without starting and completion dates
        $stmt->execute([$selectedList, $newItemTitle, $category, $state, null,
null, $newItemDescription, $imageData, $imageType]);
    } else {
        // Insert with starting and completion dates
        $stmt->execute([$selectedList, $newItemTitle, $category, $state,
$starting_date, $completion_date, $newItemDescription, $imageData, $imageType]);
    }
} else {
    // Insert data into the database without the image
    $query = "INSERT INTO user_items (list_id, item_name, category, state,
starting_date, completion_date, item_description) VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
    $stmt = $pdo->prepare($query);

    if ($state == 'Not Yet Started') {
        // Insert without starting and completion dates
        $stmt->execute([$selectedList, $newItemTitle, $category, $state, null,
null, $newItemDescription]);
    } else {
        // Insert with starting and completion dates
        $stmt->execute([$selectedList, $newItemTitle, $category, $state,
$starting_date, $completion_date, $newItemDescription]);
    }
}

header("Location:index.php");
exit();
}
?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Bucket List</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
    <link rel="stylesheet" href="styles/main.css">
    <script src="./scripts/index_2.js"></script>
    <script src="./scripts/modal.js"></script>
    <script src="./scripts/Index_3.js"></script>
    <script src="./scripts/index_des.js"></script>

</head>

<body>
    <header>
        <h1>Bucket List</h1>
    </header>
    <div class="container">

        <?php include './includes/nav.php' ?>

        <main class="page-content">
            <div class="page-container">
                <!-- Modal Container -->
                <div id="viewItemModal" class="modal">
                    <div class="modal-content">
                        <!-- Content will be loaded here dynamically -->
                    <span class="close-modal"
                        onclick="closeModal()">&times;</span>
                    <div id="modal-content"></div>
                </div>
                <!-- Display user lists from the database -->
                <div id="existing_items">
                    <?php if (!empty($userLists)) : ?>
                    <?php foreach ($userLists as $list) : ?>
                        <h3><?php echo htmlspecialchars($list['list_name']); ?>
                    <?php
                    // Fetch list items for the current user list
                    $listItemsQuery = "SELECT * FROM user_items WHERE
list_id = ? ";
                    $listItemsStmt = $pdo->prepare($listItemsQuery);
                    $listItemsStmt->execute([$list['id']]);
                    $listItems = $listItemsStmt-
                >fetchAll(PDO::FETCH_ASSOC);
                    ?>
                <?php if (!empty($listItems)) : ?>
```

```

        <ul class="list-items">
            <?php foreach ($listItems as $item) : ?>
                <li>
                    <strong>Title:</strong> <?php echo
htmlspecialchars($item['item_name']); ?>
                    <strong>Category:</strong> <?php echo
htmlspecialchars($item['category']); ?>
                    <strong>State:</strong> <?php echo
htmlspecialchars($item['state']); ?>
                    <strong>Starting Date:</strong> <?php
echo htmlspecialchars($item['starting_date']); ?>
                    <strong>Completion Date:</strong> <?
php echo htmlspecialchars($item['completion_date']); ?>
                    <strong>Description:</strong><span
class="description"> <?php echo htmlspecialchars($item['item_description']); ?>

                    <!-- Display Image -->
                    <?php if ($item['image']) : ?>
                        <label for="itemImage">Image:
</label>
                        
                    <?php endif; ?>
                    <a <?php echo ($item['state'] ===
'Completed') ? 'href="view-item.php?id=' . $item['id'] . '" class="view-item-link"' data-modal-target="viewItemModal" : 'href="view-item.php?id=' . $item['id'] . '''; ?>>
                        View
                    </a>
                </li>
            <?php endforeach; ?>
        </ul>
        <?php else : ?>
            <p>No items found for this list.</p>
        <?php endif; ?>
        <?php endforeach; ?>
    <?php else : ?>
        <p>No user lists found.</p>
    <?php endif; ?>
</div>

<div id="adding_new_item">
```

```
<h2>Add New Item</h2>
<!-- Form for adding a new item -->
<form id="addlist_form" method="POST" action="">
    <label for="newItemTitle">Title:</label>
    <input type="text" name="newItemTitle" id="newItemTitle"
required>
    <span class="error hidden" id="newItemTitleError"></span>

    <fieldset class="category_list">
        <legend>Category</legend>
        <div>
            <input type="radio" name="options"
id="travel_adventure" value="Travel_and_Adventure" />
            <label for="travel_adventure">Travel and
Adventure</label>
        </div>
        <div>
            <input type="radio" name="options"
id="unique_experiences" value="Unique_Experiences" />
            <label for="unique_experiences">Unique
Experiences</label>
        </div>
        <div>
            <input type="radio" name="options"
id="financial_milestones"
                value="Financial_Milestones" />
            <label
for="financial_milestones">FinancialMilestones</label>
        </div>
        <div>
            <input type="radio" name="options"
id="career_success" value="Career_Success" />
            <label for="career_success">Career Success</label>
        </div>
    </fieldset>

    <fieldset class="form-element">
        <legend>State</legend>
        <div>
            <input type="radio" name="state" id="completed"
value="Completed" <?= $state == 'Completed' ? 'checked' : '' ?>>
            <label for="completed">Completed</label>
        </div>

        <div>
            <input type="radio" name="state" id="in-progress"
value="In Progress" <?= $state == 'In Progress' ? 'checked' : '' ?>>
            <label for="in-progress">In Progress</label>
        </div>
        <div>
            <input type="radio" name="state" id="not-started"
value="Not Yet Started" <?= $state == 'Not Yet Started' ? 'checked' : '' ?> >
            <label for="not-started">Not Started Yet</label>
        </div>
    </fieldset>
</form>
```

```
</div>
<span class="error hidden" id="stateError"></span>
</fieldset>

<fieldset class="form-element">
<legend>Dates</legend>
<div>
    <label for="startingDate">Starting Date:</label>
    <input type="date" name="starting_date"
id="starting_date" value=<?= $starting_date ?>">
        <span class="error hidden" id="startingDateError"></span>
</div>

<div>
    <label for="completionDate">Completion Date:</label>
    <input type="date" name="completion_date"
id="completion_date" value=<?= $completion_date ?>">
        <span class="error hidden" id="completionDateError"></span>
</div>

    <label for="newItemDescription">Description:</label>
    <textarea name="newItemDescription"
id="newItemDescription" required></textarea>
        <span class="error hidden" id="newItemDescriptionError"></span>
    <span id="charCount">2500 characters remaining</span>
    <!-- Dropdown for selecting the list -->
    <div>
        <label for="selectedList">Select List:</label>
        <select name="selectedList" id="selectedList"
required>
            <?php foreach ($userListsForDropdown as $list) : ?>
            <option value=<?php echo $list['id']; ?>><?php echo htmlspecialchars($list['list_name']); ?></option>
            <?php endforeach; ?>
            <option value="new_list">Create New List</option>
        </select>
        <span class="error hidden" id="selectedListError"></span>
    </div>
    <!-- Input fields for the new list (initially hidden) -->
    <label for=" newListName" id=" newListNameLabel"
style="display: none;">New List Name:</label>
        <input type="text" name=" newListName" id=" newListName"
style="display: none;">
        <span class="error hidden" id=" newListNameError"></span>
    <!-- Checkbox for public visibility -->
    <label for="publicVisibility" id="publicVisibilityLabel">
```

```
style="display: none;">>Publicly Accessible:</label>
          <input type="checkbox" name="publicVisibility"
id="publicVisibility" value="Y" style="display: none;">

          <button type="submit">Add</button>
        </form>
      </div>

    </div>
  </main>

</body>

</html>
```

```
//Index_1
"use strict";
document.addEventListener('DOMContentLoaded', function () {
  const newListNameLabel = document.getElementById('newListNameLabel');
  const newListNameInput = document.getElementById('newListName');
  const newListDescriptionLabel =
document.getElementById('newListDescriptionLabel');
  const newListDescriptionInput = document.getElementById('newListDescription');
  const publicVisibilityLabel =
document.getElementById('publicVisibilityLabel');
  const publicVisibilityCheckbox = document.getElementById('publicVisibility');

  const selectedListDropdown = document.getElementById('selectedList');
  selectedListDropdown.addEventListener('change', function () {
    const selectedValue = selectedListDropdown.value;

    // Check if the selected value is 'new_list'
    if (selectedValue === 'new_list') {
      // Show the input fields for creating a new list
      newListNameLabel.style.display = 'block';
      newListNameInput.style.display = 'block';
      newListDescriptionLabel.style.display = 'block';
      newListDescriptionInput.style.display = 'block';
      publicVisibilityLabel.style.display = 'block';
      publicVisibilityCheckbox.style.display = 'block';
    } else {
      // Hide the input fields for creating a new list
      newListNameLabel.style.display = 'none';
      newListNameInput.style.display = 'none';
      newListDescriptionLabel.style.display = 'none';
      newListDescriptionInput.style.display = 'none';
      publicVisibilityLabel.style.display = 'none';
      publicVisibilityCheckbox.style.display = 'none';
    }
  })
})
```

```
});  
});  
`  
###  
```js  
//index_2.js
"use strict";

document.addEventListener("DOMContentLoaded", function () {
 // Select all anchor tags for viewing completed items
 const viewItemLinks = document.querySelectorAll(".view-item-link");

 // Add event listener to each link
 viewItemLinks.forEach(function (link) {
 link.addEventListener("click", function (event) {
 event.preventDefault();

 // Fetch the URL from the link's href attribute
 const url = this.getAttribute("href");

 // Fetch the item details and display in the modal window
 fetch(url)
 .then(function (response) {
 return response.text();
 })
 .then(function (data) {
 // Display the content in the modal
 document.getElementById("modal-content").innerHTML = data;

 // Show the modal
 document.getElementById("viewItemModal").style.display = "block";

 // Hide the header
 const header = document.getElementById("main-header");
 if (header) {
 header.style.display = "none";
 }

 // Hide the footer
 const footer = document.querySelector("footer");
 if (footer) {
 footer.style.display = "none";
 }
 })
 .catch(function (error) {
 console.error("Error fetching item details:", error);
 });
 });
 });

 // Function to close the modal
 window.closeModal = function () {
 document.getElementById("viewItemModal").style.display = "none";
 };
});
```

```
// Show the header
const header = document.getElementById("main-header");
if (header) {
 header.style.display = "block";
}

// Show the footer
const footer = document.querySelector("footer");
if (footer) {
 footer.style.display = "block";
}
};

});
```

```
//modal.js
"use strict";
document.addEventListener('DOMContentLoaded', function () {
 const modal = document.getElementById('myModal');
 const openModalBtn = document.getElementById('openModalBtn');
 const closeModalBtn = document.querySelector('.close');
 const modalContent = document.getElementById('modalContent');

 openModalBtn.addEventListener('click', function () {
 // Use AJAX to load content into the modal
 const xhr = new XMLHttpRequest();
 xhr.onreadystatechange = function () {
 if (this.readyState === 4 && this.status === 200) {
 modalContent.innerHTML = this.responseText;
 modal.style.display = 'block';
 }
 };
 xhr.open("GET", "modal-content.php", true);
 xhr.send();
 });

 closeModalBtn.addEventListener('click', function () {
 modal.style.display = 'none';
 });

 // Close the modal if the user clicks outside the modal content
 window.addEventListener('click', function (event) {
 if (event.target === modal) {
 modal.style.display = 'none';
 }
 });
});
```

```
//index_des.js
document.addEventListener("DOMContentLoaded", function () {
```

```
const descInput = document.getElementById("newItemDescription");
const charCountSpan = document.getElementById("charCount");
const maxChars = 2500;

descInput.addEventListener("input", function () {
 const remainingChars = maxChars - descInput.value.length;
 charCountSpan.innerText = `Characters left: ${remainingChars}`;

 if (remainingChars < 50) {
 charCountSpan.style.color = "red";
 } else {
 charCountSpan.style.color = "blue";
 }

 // Validate word limit
 const words = descInput.value.trim().split(/\s+/).filter(word =>
word.length > 0);
 const wordCount = words.length;
 const wordLimit = 200; // You can adjust the word limit

 if (wordCount > wordLimit) {
 showError("newItemDescriptionError", "Description exceeds the word
limit");
 } else {
 resetError("newItemDescriptionError");
 }
});

function showError(spanId, errorMessage) {
 const errorSpan = document.getElementById(spanId);
 errorSpan.innerText = errorMessage;
 errorSpan.style.color = "red";
}

function resetError(spanId) {
 const errorSpan = document.getElementById(spanId);
 errorSpan.innerText = "";
}
});
```

Modal Window pop up

Before clicking view

**Bucket List**

**ddjenjdnednejdn**

Title: List1  
Category: Travel\_and\_Adventure  
State: Completed  
Starting Date: 2023-12-28  
Completion Date: 2024-02-08  
Description:

[View](#) [Edit](#) [Delete](#)

**My List** • **Search** • **Public View** •

[Log Out](#) [Edit Account](#) [Delete Account](#)

After clicking view

**Bucket List**

**List1**

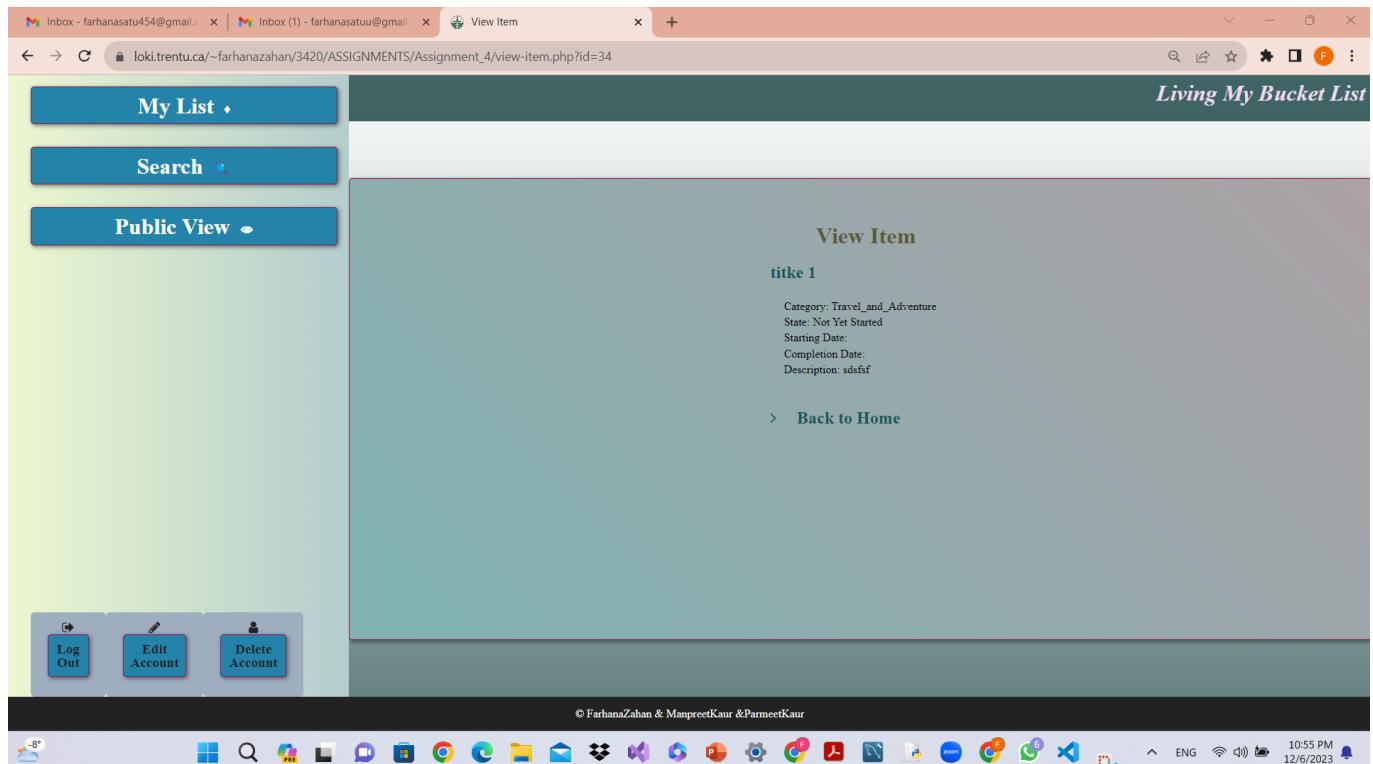
Category: Travel\_and\_Adventure  
State: Completed  
Starting Date: 2023-12-28  
Completion Date: 2024-02-08  
Description:

[Back to Home](#)

**My List** • **Search** • **Public View** •

[Log Out](#) [Edit Account](#) [Delete Account](#)

list which not in Completed state



\*\*Public page

```

<?php
session_start();
require("./includes/library.php");
$pdo = connectDB();

// Fetch public lists from the database
$publicListsQuery = "SELECT * FROM user_lists WHERE public_visibility = 'Y'";
$publicListsStmt = $pdo->query($publicListsQuery);
$publicLists = $publicListsStmt->fetchAll(PDO::FETCH_ASSOC);

?>

<!DOCTYPE html>
<html lang="en">

<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Bucket List</title>
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
 <link rel="stylesheet" href="styles/main.css">
 <script src="./scripts/public.js"></script>
</head>

<body>
 <?php include './includes/header.php' ?>

 <h2 id="view-as-public-title">View As Public</h2>

```

```
<div class="container">
 <?php include './includes/nav.php' ?>
 <main class="page-content">
 <div class="page-container">
 <!-- Modal Container -->
 <div id="viewItemModal" class="modal">
 <div class="modal-content">
 <!-- Content will be loaded here dynamically -->
 ×;
 </div>
 <div id="modal-content"></div>
 </div>
 <?php foreach ($publicLists as $list) : ?>
 <details>
 <summary>
 <h2><?php echo htmlspecialchars($list['list_name']); ?>
< i class="fa fa-caret-down"
 aria-hidden="true"></i></h2>
 </summary>
 <section id="section-<?php echo strtolower(str_replace(
', '_', $list['list_name'])) . '_' . $list['id']; ?>">
 <h3>Details of List</h3>
 <p><?php echo htmlspecialchars($list['list_desc']); ?>
</p>
<?php
// Fetch list items for the current user list
$listItemsQuery = "SELECT * FROM user_items WHERE
list_id = ?";>
 $listItemsStmt = $pdo->prepare($listItemsQuery);
 $listItemsStmt->execute([$list['id']]);
 $listItems = $listItemsStmt-
>fetchAll(PDO::FETCH_ASSOC);
 ?>

 <?php if (!empty($listItems)) : ?>
 <ul class="list-items">
 <?php foreach ($listItems as $item) : ?>

 <?php echo
htmlspecialchars($item['item_name']); ?>
 Category: <?php echo
htmlspecialchars($item['category']); ?>
 State: <?php echo
htmlspecialchars($item['state']); ?>
 Starting Date: <?php
echo htmlspecialchars($item['starting_date']); ?>
 Completion Date: <?
php echo htmlspecialchars($item['completion_date']); ?>
 Description: <?php echo
htmlspecialchars($item['item_description']); ?>
 <?php if ($item['state'] ===

```

```

'Completed') : ?>
 <a href="#" class="view-item-link" data-modal-target="viewItemModal"
 onclick="openModal(<?php echo $item['id']; ?>)">View
<?php else : ?>
 <a href="view-item.php?id=<?php echo $item['id']; ?>">View
<?php endif; ?>

 <?php if ($item['image']) : ?>
 <label for="itemImage_<?php echo
$item['id']; ?>">Image:</label>
 <input type="file"
 id="itemImage_<?php echo $item['id']; ?>"
 name="itemImage">

 <?php endif; ?>

 <?php endforeach; ?>

<?php else : ?>
 <p>No items found for this list.</p>
<?php endif; ?>
</section>
</details>
<?php endforeach; ?>
</div>
</main>
</div>
<?php include './includes/footer.php'?>
</body>

</html>

```

```

//publis.js

"use strict";
function openModal(itemId) {
 // Use AJAX to load content into the modal
 const xhr = new XMLHttpRequest();
 xhr.onreadystatechange = function () {
 if (this.readyState === 4 && this.status === 200) {
 document.getElementById('modal-content').innerHTML =
 this.responseText;
 document.getElementById('viewItemModal').style.display = 'block';
 }
 };
 xhr.open("GET", "view-item.php?id=" + itemId, true);
}

```

```
xhr.send();
}
```

## \*\*Delete Account

```
<?php
// Start the session to check user authentication
session_start();
// Include your database connection file
require("./includes/library.php");
$pdo = connectDB();
// Check if the user is logged in
if (!isset($_SESSION['username'])) {
 // Redirect to the login page if not logged in
 header("Location: login.php");
 exit();
}
// Check if the form is submitted
if (isset($_POST['delete'])) {
 // Fetch the username of the logged-in user
 $username = $_SESSION['username'];

 // Delete related items first
 $delete_items_query = "DELETE FROM user_items WHERE list_id IN (SELECT id FROM
user_lists WHERE username = ?);
 $stmt_items = $pdo->prepare($delete_items_query);

 if ($stmt_items->execute([$username])) {

 // Delete related records first
 $delete_lists_query = "DELETE FROM user_lists WHERE username = ?";
 $stmt_lists = $pdo->prepare($delete_lists_query);

 if ($stmt_lists->execute([$username])) {
 // Now delete the user account
 $delete_query = "DELETE FROM assn_users WHERE username = ?";
 $stmt = $pdo->prepare($delete_query);

 if ($stmt->execute([$username])) {
 // Logout the user and redirect to the login page after successful
 deletion
 session_destroy();
 echo "Account deleted";
 header("Location: login.php");
 exit();
 } else {
 // Handle error
 echo "Error deleting user account.";
 }
 } else {
 // Handle error
 }
 } else {
 // Handle error
 }
}
```

```

 echo "Error deleting related records.";
 }
}

}

?>
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Delete Account</title>
 <link rel="stylesheet" href="styles/main.css">
 <script src=".//scripts/delete-account.js"></script>

</head>
<body>
 <header>
 <h1>Back to Login</h1>
 </header>
 <main class="page-content">
 <form id="page-form" class="page-form" method="post">
 <div class="page-container">
 <div class="heading">
 <h2> Delete Account</h2>
 </div>
 <p>Are you sure you want to delete your account?</p>
 <input type="submit" name="delete" value="Delete Account">

 </div>
 </form>
 </main>

</body>
<?php include './includes/footer.php' ?>
</html>

```

```

//delete-account.js
"use strict";
document.addEventListener("DOMContentLoaded", function () {
 var deleteForm = document.getElementById("page-form");

 if (deleteForm) {
 deleteForm.addEventListener("submit", function (event) {
 // Display a confirmation dialog before submitting the form
 var isConfirmed = confirm("Are you sure you want to delete your account?");

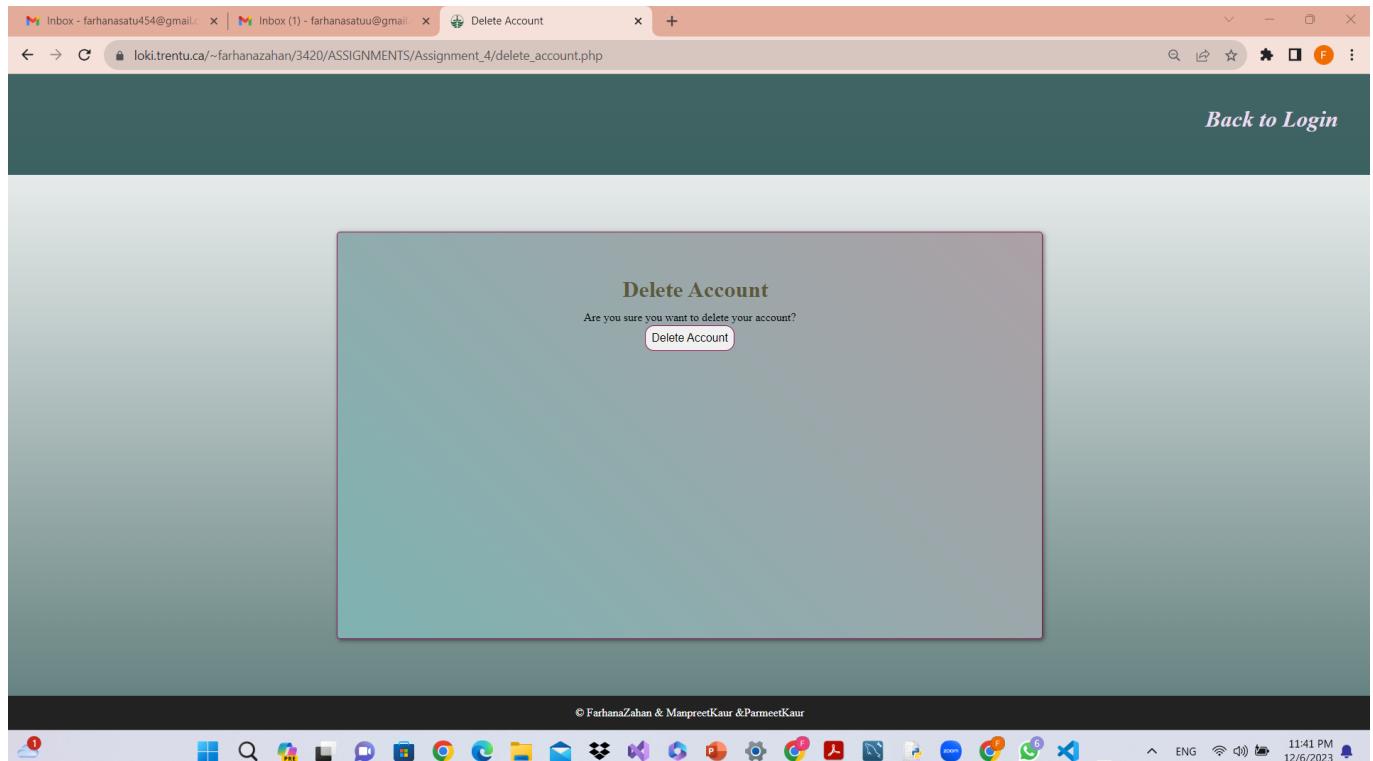
 if (!isConfirmed) {
 // Prevent the form from being submitted if the user cancels
 event.preventDefault();
 }
 });
 }
});

```

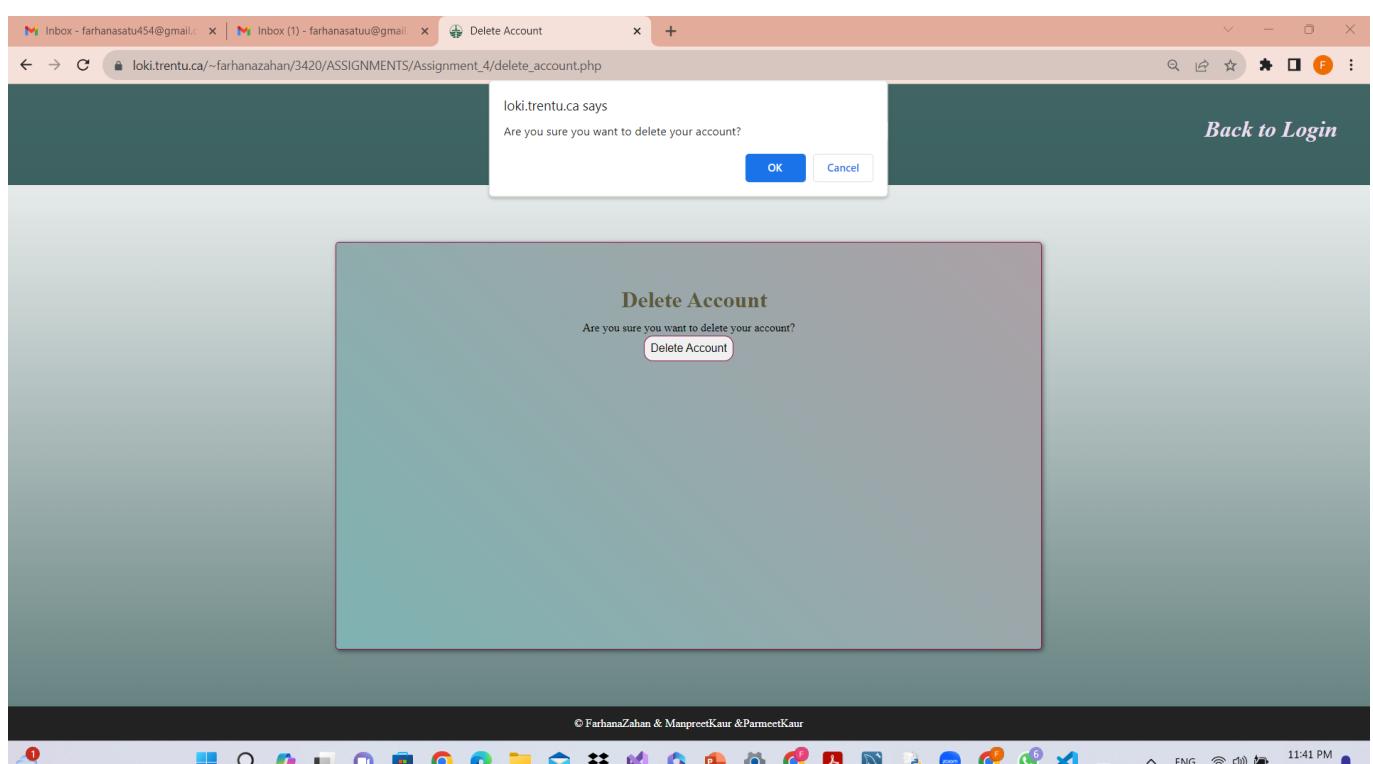
```
 }
 });
}
});
```

## validation

### before delete



### before when click to delete



## database after deleting

The screenshot shows the phpMyAdmin interface for the 'farhanazahan' database. The left sidebar lists tables: 3420\_candies, 3420\_users, 3420\_votes, assn\_users, user\_items, and user\_lists. The 'assn\_users' table is selected, displaying 9 rows of data.

	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">id</a>	<a href="#">username</a>	<a href="#">password</a>	<a href="#">name</a>	<a href="#">email</a>	<a href="#">reset_token</a>	<a href="#">reset_token_expiration</a>
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">1</a>	farha	\$2y\$10\$ceuNHoCkWVPHq0Vc3dB3erBopJpAAsmkzw0ZFPqr...	farha	farha@trentu.ca	e3384f4ccc23941648eb825e50f8816	2023-11-21 13:01:53	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">6</a>	Manpreet	\$2y\$10\$R9v21g72LduPCERkPdYGeJrYaTf2Brclz3HZBYcTnk...	manpreet	manprkaur@trentu.ca	bcdacaa091e174ff31405960343867c1	2023-11-22 23:50:26	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">7</a>	Afrin	\$2y\$10\$1.8eHfMR1fPSl02bvvhq.aANTQDXcsjOWRFcVSAR/...	Afrin	Afrin@trentu.ca	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">9</a>	farhana	\$2y\$10\$MElojodzO73FoluXHnltu5.l66o.bm4T2JlrlHTLue...	Farhana	farhanazahan@trentu.ca	01a0a2a0a0d1263b828f4337bedf5896	2023-11-23 16:16:50	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">11</a>	Alif	\$2y\$10\$Ed55ETev7taTpP8Qrtqalm8Om9KXAXz24NT5zyxcdCrkb...	Alif12	Alif12@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">12</a>	Asdf	\$2y\$10\$yCMCPI2uUn86sQJSyHCireStJw3AwREDQ9bVOxG0AqU...	Asdf	Asdf@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">13</a>	Alisa	\$2y\$10\$ViHOgvHbBamSHQYVV0ORH/c2DSH9xbTzTyMFkp...	Alisa	Alisa@trentu.ca	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">14</a>	ALI	\$2y\$10\$BKbd7XnruE0S/f1CzUGMbe8AsCtnj94QXSWQOPvXQ...	ALI	ALI@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">15</a>	Anu	\$2y\$10\$1xpEoKGaaaaUNyBi8v9pPoYS5nDhdOJ86ZA.NG1.Rb5...	Anu	Anu@gmail.com	NULL	NULL	

The screenshot shows a web application interface. On the left, there are buttons for 'My List', 'Search', and 'Public View'. The main content area displays a list item with the title 'titke 1'. Below the title, there is some descriptive text: Category: Travel\_and\_Adventure, State: Not Yet Started, Starting Date: , Completion Date: , Description: sdfsdf. At the bottom right of the content area is a 'Back to Home' link. The footer contains a copyright notice: © FarhanaZahan & ManpreetKaur & ParmeetKaur.

All the validation in mac

The screenshot shows the phpMyAdmin interface for the 'farhanazahan' database. The left sidebar lists tables: New, 3420\_candies, 3420\_users, 3420\_votes, asn\_users, user\_items, user\_lists, and information\_schema. The 'asn\_users' table is selected. The main area displays the table structure with columns: id, username, password, name, email, reset\_token, and reset\_token\_expiration. There are 9 rows of data. The bottom navigation bar includes links for Print, Copy to clipboard, Export, Display chart, Create view, and Console.

	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	<a href="#">id</a>	<a href="#">username</a>	<a href="#">password</a>	<a href="#">name</a>	<a href="#">email</a>	<a href="#">reset_token</a>	<a href="#">reset_token_expiration</a>
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	1	farha	\$2y\$10\$ceuNHoCkWVPHhq6Vc3dB3erBopJpAAsmkzw0ZFPqsr...	farha	farha@trentu.ca	e33845f4ccc23941648eb825e50f8816	2023-11-21 13:01:53	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	6	Manpreet	\$2y\$10\$R9v21/g72LduPCERkPdYGeJrYaTFzBrdz3HZBYcTnk...	manpreet	manprkaur@trentu.ca	bcdacaa091e174ff31405980343867c1	2023-11-22 23:50:26	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	7	Afrin	\$2y\$10\$1.8eHfMR1PSli02bvhq.aANTQ0XcsjOWRFgFVSAR/...	afrin	Afrin@trentu.ca	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	9	Farhana	\$2y\$10\$MEJoi0dzO.73FoluXInMft5!66o bm4TzJrHTrLue...	Farhana	farhanazahan@trentu.ca	01a0a2a0a0d1263b8284337bedf5896	2023-11-23 16:16:50	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	11	Alif	\$2y\$10\$SEcd5ETev7taPp8QItopaM6Om9KXAX24NT5zyxdCrb...	Alif12	Alif12@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	12	Asdf	\$2y\$10\$yCMCPi2uJn86sQJSRHcreStjw3AwREDQ9bV0xG0AqU...	Asdf	Asdf@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	13	Alisa	\$2y\$10\$vHGvpHo BamSHQYVW0ORH /c2DSh9xbFzTyMFkp...	Alisa	Alisa@trentu.ca	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	14	ALI	\$2y\$10\$BKbd7XnuEU0S/1CzUGMbe8AsClnj94QXSWQOPvvXQ...	ALI	ALI@gmail.com	NULL	NULL	
<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	15	Anu	\$2y\$10\$xpEoKGAAaaaUNyBi0v9pPOyS5nDhdOJ86ZA NG1.Rb5...	Anu	Anu@gmail.com	NULL	NULL	