

|  |  |  |
| --- | --- | --- |
| Module Code | : | AAPP009-4-2-WDT – WEB DEVELOPMENT |
| Intake Code | : | UCDF2208ICT(SE) |
| Lecturer Name | : | MR MOHAMAD FIRDAUS CHE ABDUL RANI |
| Tutorial No. | : | T20 |
| Group Leader | : | Alvan Sew (TP072788) |
| Project Title | : | Charity Ease Malaysia |

|  |  |
| --- | --- |
| Student ID | Student Name |
| TP072535 | Tham King Joe |
| TP068941 | Enoch Kok Jia Xian |
| TP072665 | Wilson Tan Jun Yuan |
| TP072214 | Tancel Raj |

Table of Contents

[Introduction 4](#_Toc168348791)

[Design 5](#_Toc168348792)

[Entity Relationship Diagram 5](#_Toc168348793)

[Website Navigational Structure 5](#_Toc168348794)

[Data Dictionary 6](#_Toc168348795)

[Wire Frame 7](#_Toc168348796)

[Before Login 7](#_Toc168348797)

[Home Page 7](#_Toc168348798)

[Event Page 8](#_Toc168348799)

[Login Page 9](#_Toc168348800)

[Register Page 10](#_Toc168348801)

[After Login 11](#_Toc168348802)

[User Home Page 11](#_Toc168348803)

[User Event Page 12](#_Toc168348804)

[Donate Page 13](#_Toc168348805)

[Feedback Page 14](#_Toc168348806)

[Comment Page 16](#_Toc168348807)

[Admin 17](#_Toc168348808)

[Admin View User Page 17](#_Toc168348809)

[Admin View Comment Page 18](#_Toc168348810)

[Admin View Payment Page 19](#_Toc168348811)

[Admin Edit User Page 20](#_Toc168348812)

[Admin Add Contact Page 21](#_Toc168348813)

[Implementation 22](#_Toc168348814)

[Before Login 22](#_Toc168348815)

[Login Page 22](#_Toc168348816)

[Register Page 24](#_Toc168348817)

[Log Out\ 26](#_Toc168348818)

[Home Page 26](#_Toc168348819)

[Event Page 28](#_Toc168348820)

[Web Header 30](#_Toc168348821)

[Delete page 31](#_Toc168348822)

[User Home Page (After user Login) 32](#_Toc168348823)

[User Event Page (After user Login) 33](#_Toc168348824)

[Donation Page 34](#_Toc168348825)

[User Profile Page 35](#_Toc168348826)

[Feedback Page 38](#_Toc168348827)

[Comments Page 39](#_Toc168348828)

[Edit Page 41](#_Toc168348829)

[Update php function 43](#_Toc168348830)

[User Guidance 44](#_Toc168348831)

[Conclusion 56](#_Toc168348832)

[References 57](#_Toc168348833)

[Workload Matrix 58](#_Toc168348834)

# Introduction

Charity Ease foundation is a charity foundation that is aimed towards helping those in their golden years live worry free and in comfort. We do this by collecting and donating funds to various old folks' homes with the intention of lifting their financial burdens and providing them with enough funds to better their facilities and spare no luxury from those who are nearing their end of life.

Our website allows users to view our main page which displays all information about our charity. There is register option which creates profile for users to unlock more options like feedback and the function to donate. Without logging into a registered account, the user is unable to leave comments and donate. Users can create their account in registration page by providing their name, username, phone number, email, gender, address and password.

Registered customers can deposit funds into our website which will then be channelled to various old folks' homes. Customers will be able to change and edit their account profiles whenever they wish. Customers will also be able to provide feedback which will then be reviewed and used to improve our services.

# Design

## Entity Relationship Diagram

A screenshot of a computer

Description automatically generated

## Website Navigational Structure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Field Length** | **Constraint** | **Description** |
|  |  |  |  |  |
| user\_id | Int |  | Public Key | Id number of user |
| account\_name | Varchar | 255 | Not Null | The identification that the user would prefer in the website |
| user\_name | Varchar | 255 | Not Null | Name of user |
| user\_phone | Varchar | 50 | Not Null | Phone number of user |
| user\_email | Varchar | 255 | Not Null | Email address of user |
| user\_password | Varchar | 50 | Not Null | Password of user |
| user\_address | Text |  | Not Null | Address of user |
| user\_gender | Varchar | 10 | Not Null | Gender of user |
| user\_dob | Date |  | Not Null | Date of birth of user |
| user\_type | varchar | 255 | Not Null | Use to identify whether user have logged in or not |
|  |  |  |  |  |
| payment\_id | Int |  | Primary key | Id of payment made by user |
| user\_id | Int |  | Foreign Key | Id number of user |
| card\_detail | Varchar | 255 | Not Null | Card detail’s of user |
| cardholder\_name | Varchar | 255 | Not Null | The name of the person who owns the card |
| expiry\_date | Date |  | Not Null | Expiry date of the card |
| CVV | Int |  | Not Null | Card Verification Value of the card |
| amount | int |  | Not Null | The amount that the user would like to donate |
|  |  |  |  |  |
| feedback\_id | int |  | Primary key | Id number of the comment made |
| feedback | text |  | Not Null | Contains the feedback that is given by the user. |

## Data Dictionary

# Wire Frame

## Before Login

In this section, the following shows a rough sketch of what the UI UX of the website, showing the location of all the buttons would have access to before the user have created an account in our website

### Home Page

A screenshot of a computer

Description automatically generated

In our home page, it will display buttons that redirect you to other function pages, such as event, login and donate page.

### Event Page

A screenshot of a computer

Description automatically generated

In our event page, it shows information about our event, also it displays function buttons like feedback and comments. It allows the user to provide feedback to the organizer. The comment button allows the user to view at the comments.

### Login Page

A screenshot of a computer

Description automatically generatedOur login page allows user to login into their accounts by inserting their email and password. The remember me check box behaves when it is clicked, the next time they enter our website it will automatically login into their account without having to insert their account credentials again. The login page only works if the user has already created an account.

### Register Page

A screenshot of a login form

Description automatically generated

The register page is a page where you create an account if you haven’t. This page can be accessed through pressing ‘click here to register’ button on the login page. The user can create an account by providing their name, username, phone number, email, password, house address, gender and birthday date and submit.

## After Login

### User Home Page

A screenshot of a computer

Description automatically generated

After logging in, the user will be in the user home page. In this page the user has access to the event page, search bar and donate page. The user can also review their profile details by pressing the profile button on the top right and logout with the logout button.

### User Event Page

A screenshot of a computer

Description automatically generated

The user event page displays every detail about our event. The feedback button allows the user to leave a feedback. The comment button brings the user to the comment section where they can review their comment or other people’s comment about our event.

### Donate Page

A screenshot of a web page

Description automatically generated

The donate page allows user to donate after pressing the donate button. They can pay via card by providing their card details, name and other card credentials. Press submit to donate immediately.

### Feedback Page

A screenshot of a web page

Description automatically generated

After entering the feedback page, the user is allowed to leave feedbacks to the events that they have thoughts about. Pressing submit will send the feedback to database which will be explained more technically in the future sections.

Profile Page

A screenshot of a web page

Description automatically generated

### Comment Page

A screenshot of a computer

Description automatically generated

In the comment page, the user is able to review their feedback that they left, and also other users feedback. The comments here is public to any one that has an account.

## Admin

### Admin View User Page

A screenshot of a contact form

Description automatically generated

The admin view user page is for admins to view at every user that has created an account in our website, the admins also has access to delete or edit user’s information. This page also has function buttons such as view comments, search bar and view payments.

### Admin View Comment Page

A screenshot of a computer

Description automatically generated

The admin view comment page allows admins to view at comments that any user left, it also provides the user’s description.

### Admin View Payment Page

A screenshot of a computer

Description automatically generated

The admin view payment page displays user’s payment credentials, and the option to delete it. The search bar allows the admin to search user’s name or credentials for more convenience.

### Admin Edit User Page

The admin edit user page gives admin access to edit user’s credentials such as name, password, phone number and more.

### Admin Add Contact Page

A screenshot of a contact form

Description automatically generated

The admin add contact page allows the admin to add contact by providing credentials about the user.

# Implementation

The implementation provides a walkthrough over every single line of code to describe how each individual features of the website works in technical terms.

## Before Login

### Login Page

<?php

session\_start();

if(isset($\_POST["LoginBtn"])){

$conn = mysqli\_connect("localhost","root","","assignment");

$sql = "SELECT \* FROM userdetail WHERE user\_email = '".$\_POST["user\_email"]."' AND user\_password ='".$\_POST["user\_password"]."'";

$result = mysqli\_query($conn,$sql);

$row = mysqli\_fetch\_array($result);

if($row){

$\_SESSION["user\_id"] = $row["user\_id"];

if(isset($\_POST["remember"])){

setcookie("user\_login",$\_POST["user\_email"],time() + (7 \* 24 \* 60 \* 60),"/");

} else {

unset($\_COOKIE['user\_login']);

setcookie('user\_login', null, -1, '/');

}

header("location:userhome.php");

} else {

echo"<script>alert('Invalid username/ password!');</script>";

}

}

**Handling User Login**

if(isset($\_POST["LoginBtn"])){

$conn = mysqli\_connect("localhost","root","","assignment");

$sql = "SELECT \* FROM userdetail WHERE user\_email = '".$\_POST["user\_email"]."' AND user\_password ='".$\_POST["user\_password"]."'";

$result = mysqli\_query($conn,$sql);

$row = mysqli\_fetch\_array($result);

This block checks if the login button (**LoginBtn**) is pressed and processes the login credentials. It establishes a connection to the MySQL database using **mysqli\_connect** and checks the credentials with an SQL query. If the query returns a row, it means the login is successful.

**Email Input Field**

<input name="user\_email" class="emaillogin" placeholder="Email" type="text" value="<?php if(isset($\_COOKIE['user\_login'])) { echo $\_COOKIE['user\_login']; } ?>">

Collects the user's email for login. Users enter their email address, which is used to identify their account.

**Password Input Field**

<input name="user\_password" type="password" class="passwordlogin" placeholder="Password" value="">

Collects the user's password for login. Users enter their password, which is used to authenticate their identity.

**Remember Me Checkbox**

            if(isset($\_POST["remember"])){

                setcookie("user\_login",$\_POST["user\_email"],time() + (7 \* 24 \* 60 \* 60),"/");

            } else {

                unset($\_COOKIE['user\_login']);

                setcookie('user\_login', null, -1, '/');

            }

Allows users to remain logged in for a specified duration. If checked, it sets a cookie named ‘user login’ with the user's email for 7 days. If not, it deletes the cookie by setting its expiration time to the past.

**Back Button (backlogin)**

<input type="button" name="backlogin" class="backlogin" value="Back" onclick="window.location.href='home.php'">

Redirects the user to the home page (**home.php**). Provides a quick way for users to return to the home page from the login page.

**Login Button (nextlogin)**

<input type="submit" id="LoginBtn" name="LoginBtn" class="nextlogin" value="Login">

Submits the login form. When clicked, the form data (email and password) is submitted to the server for authentication.

**Register Link**

<a href="register.php">Click Here to Register</a>

Redirects the user to the registration page (**register.php**). Allows new users to create an account.

## Register Page

<input type="submit" id="submitID" name="submitBtn" value="Submit">

If no matching user is found, it shows an alert message indicating invalid credentials.

This is the submit button for the registration form. When clicked, it triggers the form submission.

<?php

if(isset($\_POST['submitBtn']))

This checks if the form has been submitted by checking if the ‘submitBtn’ button is set in the ‘POST’ data.

{

include('conn.php');

This includes a file ‘conn.php’ which presumably contains the database connection code.

  $name = mysqli\_real\_escape\_string($con, $\_POST['name']);

    $username = mysqli\_real\_escape\_string($con, $\_POST['username']);

    $phonenumber = mysqli\_real\_escape\_string($con, $\_POST['phonenumber']);

    $emailaddress = mysqli\_real\_escape\_string($con, $\_POST['emailaddress']);

    $password = mysqli\_real\_escape\_string($con, $\_POST['password']);

    $homeaddress = mysqli\_real\_escape\_string($con, $\_POST['homeaddress']);

    $gender = mysqli\_real\_escape\_string($con, $\_POST['gender']);

    $dateofbirth = mysqli\_real\_escape\_string($con, $\_POST['dateofbirth']);

$user\_type = 'member';

This sanitizes the user input to prevent SQL injection by escaping special characters and set default user type.

    $sql = "INSERT INTO userdetail (account\_name, user\_name, user\_phone, user\_email, user\_password, user\_address, user\_gender, user\_dob, user\_type)

            VALUES ('$name', '$username', '$phonenumber', '$emailaddress', '$password', '$homeaddress', '$gender', '$dateofbirth','$user\_type')";

This constructs an SQL query to insert the user details into the ‘userdetail’ table.

    if (!mysqli\_query($con, $sql)){

        die('Error: ' . mysqli\_error($con));

    }

    else{

        echo "<script>alert('Successful!'); window.location.href='login.php';</script>";

}

This executes the query. If it fails, it shows an error message. If it succeeds, it shows a success message and redirects to the login page.

    mysqli\_close($con);

}

else{

    echo "<script>alert('Please fill in the form'); window.location.href='register.php';</script>";

}

?>

If the form is not submitted, it shows an alert and redirects back to the registration page.

### Log Out\

<?php

session\_start();

session\_destroy();

echo"<script>alert('Successfully Logout.');window.location.href='home.php';</script>";

?>

The logout functionality in **logout.php** ensures a secure and effective logout process for users. When a user clicks the logout button, this PHP script is executed. It starts by including the necessary database connection file (**conn.php**) and initiating a session.

The script checks if a user is logged in by verifying the existence of **$\_SESSION['user\_id']**. If the user is logged in, it retrieves the user ID from the session and updates the **user\_type** field in the database to 'none'. This effectively logs the user out by changing their status. The session is then destroyed, and the user is redirected to the home page (**home.php**) with a success message. If the user is not logged in, an alert is displayed, and the user is redirected to the home page. (PHP MySQL select data, n.d.)

### Home Page

<div class="header">

<div class="box1">

<button class="homebtn" onclick="window.location.href='home.php'">

<img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

</button>

<button id="eventbtn" onclick="window.location.href='event.php'">Event</button>

</div>

<div class="box3">

<button class="loginbtn" onclick="window.location.href='login.php'">

<img src="images/bananaman.png" alt="user" id="user">

</button>

</div>

</div>

<div class="header2">

<input type="text" name="location-search" id="location-search" placeholder="🔎 Location">

</div>

<button id="Lmh" onclick="window.location.href='pleaseLogin.php'">Let me Help!</button>

**Home Button (homebtn)**

<button class="homebtn" onclick="window.location.href='home.php'">

<img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

</button>

Redirects the user to the home page (**home.php**). The **onclick** event triggers **window.location.href='home.php'**, navigating the user to the main landing page.

**Event Button (eventbtn)**

<button id="eventbtn" onclick="window.location.href='event.php'">Event</button>

Redirects the user to the events page (**event.php**).

The **onclick** event triggers **window.location.href='event.php'**, navigating the user to the events page.

**Login Button (loginbtn)**

<button class="loginbtn" onclick="window.location.href='login.php'">

<img src="images/bananaman.png" alt="user" id="user">

</button>

Redirects the user to the login page (**login.php**).

The **onclick** event triggers **window.location.href='login.php'**, navigating the user to the login page.

**Search Box (#location-search)**

<input type="text" name="location-search" id="location-search" placeholder="🔎 Location">

Allows users to search for events or services by location. Users type a location name or keyword into the search box to filter events or services relevant to their search criteria.

**Let Me Help Button (#Lmh)**

<button id="Lmh" onclick="window.location.href='pleaseLogin.php'">Let me Help!</button>

Prompts the user to log in before they can offer help. The **onclick** event triggers **window.location.href='pleaseLogin.php'**, navigating the user to a login prompt page.

### Event Page

    <div class="header3">

        <div class="leftarrow">

                <!-- <img src="images/urfather.png" alt="left-arrow">   
 we cancel it-->

        </div>

        <div class="preview-image">

            <img src="images/charity\_giving\_poster.png" alt="preview-image" id="preimg">

        </div>

        <div class="description">

            <div class="des-title">

                <h1 id="des-title">Empower Community Spirit: #GivingTuesday Charity Bash</h1>

            </div>

            <div class="des-content">

                Detail of the Event

            </div>

                <div class="row">

                <button id="pl" onclick="window.location.href='pleaseLogin.php'">Feedback</button>

                    Feedback

                </button>

                <button id="pl" onclick="window.location.href='pleaseLogin.php'">Comments</button>

                    Comments

                </button>

    </div>

        </div>

        <div class="rightarrow">

            <!-- <img src="images/urmother.png" alt="left-arrow">

we cancel it-->

        </div>

    </div>

**Home Button (homebtn)**

<button class="homebtn" onclick="window.location.href='home.php'">

<img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

</button>

Redirects the user to the home page (**home.php**). Provides an easy way for users to navigate back to the main page from any other page on the site. Users simply click this button to return to the home page.

**Event Button (eventbtn)**

<button id="eventbtn" onclick="window.location.href='event.php'">Event</button>

Redirects the user to the events page (**event.php**). Allows users to view all events organized by the community service platform, keeping them informed about upcoming activities and initiatives. Users click this button to view the events page.

**Login Button (loginbtn)**

<button class="loginbtn" onclick="window.location.href='login.php'">

<img src="images/bananaman.png" alt="user" id="user">

</button>

Redirects the user to the login page (**login.php**). Ensures users can access the login page from various parts of the site. Users click this button to navigate to the login page.

**Search Box**

<input type="text" name="location-search" id="location-search" placeholder="🔎 Location">

Allows users to search for events or services by location. Users type in a location name or keyword to filter events or services relevant to their search criteria.

**Feedback Button (#pl)**

<button id="pl" onclick="window.location.href='pleaseLogin.php'">Feedback</button>

Prompts the user to log in before they can leave feedback. When clicked, this button redirects the user to **pleaseLogin.php** if they are not logged in, ensuring only authenticated users can provide feedback.

**Comments Button (#pl)**

<button id="pl" onclick="window.location.href='pleaseLogin.php'">Comments</button>

Prompts the user to log in before they can leave comments. When clicked, this button redirects the user to **pleaseLogin.php** if they are not logged in, ensuring only authenticated users can leave comments.

### Web Header

    <div class="header">

        <div class="box1">

            <button class="homebtn" onclick="window.location.href='home.php'">

                <img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

            </button>

            <button id="eventbtn" onclick="window.location.href='event.php'">Event</button>

        </div>

        <div class="box3">

            <button class="loginbtn" onclick="window,location.href='login.php'">

                <img src="images/bananaman.png" alt="user" id="user">

            </button>

        </div>

    </div>

    <div class="header2">

        <input type="text" name="location-search" id="location-search" placeholder="🔎 Location">

</div>

Home Button (homebtn)

<button class="homebtn" onclick="window.location.href='home.php'">

<img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

</button>

Redirects the user to the home page (**home.php**).

The **onclick** event triggers **window.location.href='home.php'**, navigating the user to the main landing page.

Event Button (eventbtn)

<button id="eventbtn" onclick="window.location.href='event.php'">Event</button>

Redirects the user to the events page (**event.php**).

The **onclick** event triggers **window.location.href='event.php'**, navigating the user to the events page.

**Login Button (loginbtn)**

<button class="loginbtn" onclick="window.location.href='login.php'">

<img src="images/bananaman.png" alt="user" id="user">

</button>

Redirects the user to the login page (**login.php**).

The **onclick** event triggers **window.location.href='login.php'**, navigating the user to the login page.

**Search Box (#location-search)**

<input type="text" name="location-search" id="location-search" placeholder="🔎 Location"

Allows users to search for events or services by location.

Users type a location name or keyword into the search box to filter events or services relevant to their search criteria.

### Delete page

<?php

include("conn.php");

if(isset($\_GET['user\_id'])){

    $user\_id = $\_GET['user\_id'];

$sql= "DELETE FROM userdetail WHERE user\_id =$user\_id";

Delete the user which link by the uer\_id from the data base

    $result = mysqli\_query($con,$sql);

    mysqli\_close ($con);

    echo "<script>alert('Record Deleted!');

    window.location.href='admin\_view.php';</script>";}

else{

    echo "<script>alert('Please select the record to delete');

    window.location.href='admin\_view.php';</script>";}

?>

When the delete function work, show the inform to admin

(We can using this same method when deleting comment and payment in admin page like changing the user\_id to payment\_id or feedback\_id.)

### User Home Page (After user Login)

<div class="header3">

<div class="leftarrow"></div>

<div class="preview-image">

<img src="images/WhatsApp Image 2024-05-01 at 20.15.20\_6e98cadd.jpg" alt="preview-image" id="preimg">

</div>

<div class="description">

<div class="des-title">

<h1 id="des-title">Join Hands for Hope: Support Our Elders in Malaysia</h1>

</div>

<div class="des-content">

<!-- Donation details go here -->

</div>

<button id="Lmh" onclick="window.location.href='donation.php'">Let me Help!</button>

</div>

<div class="rightarrow"></div>

</div>

**Home Button (homebtn)**

<button class="homebtn" onclick="window.location.href='userhome.php'">

<img src="images/Charityeaselogo.jpg" alt="chairityeaselogo" id="charitylogo">

</button>

Redirects the user to the user home page (**userhome.php**).

The **onclick** event triggers **window.location.href='userhome.php'**, navigating the user to their home page.

**Let Me Help Button (#Lmh)**

<button id="Lmh" onclick="window.location.href='donation.php'">Let me Help!</button>

Redirects the user to the donation page (**donation.php**).

The **onclick** event triggers **window.location.href='donation.php'**, navigating the user to the donation page.

### User Event Page (After user Login)

<div class="header3">

<div class="leftarrow"></div>

<div class="preview-image">

<img src="images/charity\_giving\_poster.png" alt="preview-image" id="preimg">

</div>

<div class="description">

<div class="des-title">

<h1 id="des-title">Empower Community Spirit: #GivingTuesday Charity Bash</h1>

</div>

<div class="des-content">

<!-- Event details go here -->

</div>

<div class="row">

<button id="Lmh" onclick="window.location.href='feedback.php'">Feedback</button>

<button id="Lmh" onclick="window.location.href='comments.php'">Comments</button>

</div>

</div>

<div class="rightarrow"></div>

</div>

**Feedback Button (#Lmh)**

<button id="Lmh" onclick="window.location.href='feedback.php'">Feedback</button>

Redirects the user to the feedback page (**feedback.php**).

The **onclick** event triggers **window.location.href='feedback.php'**, navigating the user to the feedback page.

**Comments Button (#Lmh)**

<button id="Lmh" onclick="window.location.href='comments.php'">Comments</button>

Redirects the user to the comments page (**comments.php**).

The **onclick** event triggers **window.location.href='comments.php'**, navigating the user to the comments page.

### Donation Page

<?php

session\_start();

if(isset($\_POST['pmsubmitBtn']))

{

include('conn.php');

if(isset($\_SESSION['user\_id'])) {

$user\_id = $\_SESSION['user\_id'];

echo "User ID: ". $user\_id;}

else {die('Error: User not logged in');}

$cardDetail = mysqli\_real\_escape\_string($con, $\_POST['cardDetail']);

$cardholderName = mysqli\_real\_escape\_string($con, $\_POST['cardholderName']);

$expiryDate = mysqli\_real\_escape\_string($con, $\_POST['expiryDate']);

$cvv = mysqli\_real\_escape\_string($con, $\_POST['cvv']);

$amount = mysqli\_real\_escape\_string($con, $\_POST['amount']);

$sql = "INSERT INTO payment (user\_id, card\_detail, cardholder\_name, expiry\_date, CVV, amount)

VALUES ('$user\_id', '$cardDetail', '$cardholderName', '$expiryDate', '$cvv', '$amount')";

if (!mysqli\_query($con, $sql)) {die('Error: ' . mysqli\_error($con));}

else { echo "<script>alert('Payment Successful!');

window.location.href='userhome.php';</script>";}

mysqli\_close($con);}

else {echo "<script>alert('Form submission failed'); window.location.href='donation.php';</script>";}

?>

1. Session Start and Form Check:

The script starts a session and checks if the form has been submitted by verifying the presence of the 'pmsubmitBtn' in the **$\_POST** data.

1. Database Connection:

The script includes the **conn.php** file to connect to the database.

1. User Login Verification:

It checks if the user is logged in by verifying **$\_SESSION['user\_id']**. If not logged in, it stops execution with an error message.

1. Sanitize User Input:

The script sanitizes the user input to prevent SQL injection using **mysqli\_real\_escape\_string()**.

1. Insert Payment Information:

It constructs and executes an SQL query to insert the payment information into the database.

1. Error Handling and Redirection:

If the query fails, it stops execution and shows an error. If successful, it shows a success message and redirects to the user home page. The database connection is then closed.

1. Form Submission Failure:

If the form was not submitted, it alerts the user of the failure and redirects back to the donation page.

Php to connect with database

<?php

$con = mysqli\_connect("localhost","root","","assignment");

if(mysqli\_connect\_errno()){

echo"Can't connect to the DB:". mysqli\_connect\_error();

}

?>

### User Profile Page

<?php

session\_start();

include("conn.php"); // Connect to the database

$user\_id = $\_SESSION['user\_id']; // Get user ID from session

// Check if the form is submitted

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

$user\_name = $\_POST['user\_name'];

$account\_name = $\_POST['account\_name'];

$user\_phone = $\_POST['user\_phone'];

$user\_email = $\_POST['user\_email'];

$user\_password = $\_POST['user\_password'];

$user\_address = $\_POST['user\_address'];

$user\_gender = $\_POST['user\_gender'];

$user\_dob = $\_POST['user\_dob'];

// Update user details in the database

$update\_sql = "UPDATE userdetail SET

user\_name='$user\_name',

account\_name='$account\_name',

user\_phone='$user\_phone',

user\_email='$user\_email',

user\_password='$user\_password',

user\_address='$user\_address',

user\_gender='$user\_gender',

user\_dob='$user\_dob'

WHERE user\_id='$user\_id'";

// Execute the update query and handle success or failure

if (mysqli\_query($con, $update\_sql)) {

echo "<script>alert('Upload Successful.'); window.location.href = 'userhome.php';</script>";

exit;

} else {

echo "Error updating profile: " . mysqli\_error($con);

}

}

// Retrieve user details to display in the form

$sql = "SELECT \* FROM userdetail WHERE user\_id='$user\_id'";

$result = mysqli\_query($con, $sql);

if ($row = mysqli\_fetch\_array($result)) {

?>

<!-- HTML form to display user details for editing -->

<!-- (HTML form code would go here) -->

<?php

} else {

echo "No user found.";

}

mysqli\_close($con); // Close the database connection

?>

1. Session and Database Connection:

The script starts a session using **session\_start();** and includes the **conn.php** file to establish a connection to the database.

1. Retrieve User ID:

It retrieves the user ID from the session with **$user\_id = $\_SESSION['user\_id'];**.

1. Form Submission Check:

If the form is submitted (**if ($\_SERVER['REQUEST\_METHOD'] == 'POST')**), it collects the user input from the POST request.

1. Update User Details:

The script constructs an SQL query to update the user details in the database with the provided information (**$update\_sql = "UPDATE userdetail SET ..."**). It executes the query and handles the success or failure of the update operation:

if (mysqli\_query($con, $update\_sql)) {

echo "<script>alert('Upload Successful.'); window.location.href = 'userhome.php';</script>";

exit;

} else {

echo "Error updating profile: " . mysqli\_error($con);

}

1. Fetch User Details:

It constructs an SQL query to fetch the user details from the database based on the user ID (**$sql = "SELECT \* FROM userdetail WHERE user\_id='$user\_id'";**). If the user is found, it displays the user details in an HTML form for editing. If no user is found, it displays an error message:

if ($row = mysqli\_fetch\_array($result)) {

// Display form with user details

} else {

echo "No user found.";

}

1. Close Database Connection:

The script closes the database connection with **mysqli\_close($con);**.

### Feedback Page

<?php

session\_start();

if(isset($\_POST['FBsubmitBtn']))

{

    include('conn.php');

    if(isset($\_SESSION['user\_id'])) {

        $user\_id = $\_SESSION['user\_id'];

        echo "User ID: ". $user\_id;}

else {die('Error: User not logged in');}

    $feedback = mysqli\_real\_escape\_string($con, $\_POST['feedback']);

    $sql = "INSERT INTO feedback (user\_id, feedback)

            VALUES ('$user\_id', '$feedback')";

    if (!mysqli\_query($con, $sql)){

        die('Error: ' . mysqli\_error($con));}

    else{echo "<script>alert('Thanks For Your Feedback!'); window.location.href='comments.php';</script>";}

    mysqli\_close($con);

}

else{echo "<script>alert('error'); window.location.href='feedback.php';</script>";}

?>

1. Session and Database Connection:

The script starts a session using **session\_start();** and includes the **conn.php** file to establish a connection to the database.

1. User Login Check:

It checks if the user is logged in by verifying the presence of **$\_SESSION['user\_id']**. If the user is logged in, it assigns the user ID to **$user\_id**. If not, it stops execution with an error message:

if(isset($\_SESSION['user\_id'])) {

$user\_id = $\_SESSION['user\_id'];

echo "User ID: ". $user\_id;

} else {

die('Error: User not logged in');

}

1. Sanitize User Input:

The script sanitizes the user input to prevent SQL injection using **mysqli\_real\_escape\_string($con, $\_POST['feedback']);**.

1. Insert Feedback:

The script constructs an SQL query to insert the feedback into the database (**$sql = "INSERT INTO feedback (user\_id, feedback) VALUES ('$user\_id', '$feedback')";**). It executes the query and handles the success or failure of the insert operation:

if (!mysqli\_query($con, $sql)) {

die('Error: ' . mysqli\_error($con));

} else {

echo "<script>alert('Thanks For Your Feedback!'); window.location.href='comments.php';</script>";

}

1. Close Database Connection:

The script closes the database connection with **mysqli\_close($con);**.

### Comments Page

<?php

$search\_keyword = "";

if(isset($\_POST['searchBtn'])){

    $search\_keyword = $\_POST['search\_keyword'];

}

include("conn.php");

$sql = "SELECT \*

        FROM feedback

        INNER JOIN userdetail ON feedback.user\_id = userdetail.user\_id

        WHERE userdetail.user\_name LIKE '%$search\_keyword%'";

$result = mysqli\_query($con, $sql);

while ($row = mysqli\_fetch\_array($result)) {

    $feedback = $row['feedback'];

    $feedback\_id = $row['feedback\_id'];

    $gender\_image = ($row['user\_gender'] == "Male") ? "images/male.png" : "images/female.png";

    if (strlen($feedback) > 100) {

        $visible\_text = substr($feedback, 0, 100);

        $hidden\_text = substr($feedback, 100);

        echo '<div id="box">';

        echo '<img src="'.$gender\_image.'" width="50">';

        echo '<h3>'. $row['user\_name'].'</h3>';

        echo '<p><strong>Account Name:</strong> '. $row['account\_name'].'</p>';

        echo '<h3>'. $feedback\_id .'</h3>';

        echo '<p>'. $visible\_text .'<span class="hidden-text" id="hidden-'.$feedback\_id.'">'. $hidden\_text .'</span></p>';

        echo '<span class="more" id="more-'.$feedback\_id.'" onclick="toggleMore('.$feedback\_id.')">More</span>';

        echo '</div>';

    } else {

        echo '<div id="box">';

        echo '<img src="'.$gender\_image.'" width="50">';

        echo '<h3>'. $row['user\_name'].'</h3>';

        echo '<p><strong>Account Name:</strong> '. $row['account\_name'].'</p>';

        echo '<h3>'. $feedback\_id .'</h3>';

        echo '<p>'. $feedback .'</p>';

        echo '</div>';

    }

}

mysqli\_close($con);

?>

<script>

    function toggleMore(id) {

        var hiddenText = document.getElementById('hidden-' + id);

        var moreLink = document.getElementById('more-' + id);

        if (hiddenText.style.display === 'none' || hiddenText.style.display === '') {hiddenText.style.display='inline'; moreLink.textContent='Less';}

        else {hiddenText.style.display='none'; moreLink.textContent='More';}

    }

</script>

1. Search Keyword Handling:

The script starts by checking if the search button is pressed and retrieves the search keyword from the POST request (**$search\_keyword = $\_POST['search\_keyword'];**).

1. Database Connection:

It includes the **conn.php** file to establish a connection to the database.

1. SQL Query:

The script constructs an SQL query to join the **feedback** and **userdetail** tables based on the user ID and filters the results based on the search keyword (**$sql = "SELECT \* FROM feedback INNER JOIN userdetail ON feedback.user\_id = userdetail.user\_id WHERE userdetail.user\_name LIKE '%$search\_keyword%'";**).

1. Fetch and Display Feedback:

It executes the query and fetches the results using **mysqli\_fetch\_array($result)**. For each feedback entry, it checks if the feedback text is longer than 100 characters. If so, it splits the text into visible and hidden parts and displays it in a formatted HTML structure.

1. Close Database Connection:

The script closes the database connection with **mysqli\_close($con);**.

1. JavaScript for Toggle Functionality:

The script includes a JavaScript function to toggle the display of the hidden feedback text.

### Edit Page

    <?php

        include("conn.php");

        $id = $\_GET["id"];

        $sql = "SELECT \* FROM userdetail WHERE user\_id=$id";

        $result = mysqli\_query($con, $sql);

        $row = mysqli\_fetch\_array($result);

    ?>

    <div class="container">

        <h2>Edit Contact</h2>

        <form action="update.php" method="post">

            <input type="hidden" name="id" value="<?php echo $row['user\_id']; ?>">

            <div class="field">

                <label for="name">Name</label>

                <input type="text" name="name" id="name" required value="<?php echo $row['user\_name']; ?>">

            </div>

            <input type="hidden" name="password" value="<?php echo $row['user\_password']; ?>">

            <div class="field">

                <label for="name">Password</label>

                <input type="password" name="password" id="password" required value="<?php echo $row['user\_password']; ?>">

            </div>

            <div class="field">

                <label for="phone">Phone Number</label>

                <input type="tel" name="phone\_number" id="phone" required value="<?php echo $row['user\_phone']; ?>">

            </div>

            <div class="field">

                <label for="email">Email Address</label>

                <input type="email" name="email\_address" id="email" required value="<?php echo $row['user\_email']; ?>">

            </div>

            <div class="field">

                <label for="address">Home Address</label>

                <textarea name="home\_address" id="address" required><?php echo $row['user\_address']; ?></textarea>

            </div>

            <div class="field">

                <label>Gender</label>

                <input type="radio" name="gender" id="male" value="Male" <?php if ($row['user\_gender'] == 'Male') echo 'checked'; ?>>

                <label for="male">Male</label>

                <input type="radio" name="gender" id="female" value="Female" <?php if ($row['user\_gender'] == 'Female') echo 'checked'; ?>>

                <label for="female">Female</label>

            </div>

            <div class="field">

                <label for="dob">Date Of Birth</label>

                <input type="date" name="dob" id="dob" value="<?php echo $row['user\_dob']; ?>">

            </div>

            <div class="field">

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

                <input type="reset" value="Reset">

            </div>

        </form>

    </div>

    <?php

        mysqli\_close($con);

    ?>

Explanation:

1. Database Connection:

The script includes the **conn.php** file to establish a connection to the database.

1. Retrieve User ID:

It gets the user ID from the URL parameter using **$id = $\_GET["id"];**.

1. Fetch User Details:

The script constructs an SQL query to select user details based on the user ID (**$sql = "SELECT \* FROM userdetail WHERE user\_id=$id";**). It executes the query and fetches the result using **mysqli\_fetch\_array($result);**.

1. Display User Details:

If the user details are found, it displays the details in an HTML form for editing. Each form field is populated with the corresponding value from the database:

<input type="hidden" name="id" value="<?php echo $row['user\_id']; ?>"> <!-- Hidden field for user ID -->

<div class="field">

<label for="name">Name</label>

<input type="text" name="name" id="name" required value="<?php echo $row['user\_name']; ?>">

</div>

<input type="hidden" name="password" value="<?php echo $row['user\_password']; ?>">

<div class="field">

<label for="name">Password</label>

<input type="password" name="password" id="password" required value="<?php echo $row['user\_password']; ?>">

</div>

<!-- Additional form fields for phone, email, address, gender, and date of birth -->

1. Close Database Connection:

The script closes the database connection with **mysqli\_close($con);**.

### Update php function

<?php

include("conn.php");

Connect to the database.

    $user\_id = $\_POST['id'];

    $user\_name = $\_POST['name'];

    $user\_password = $\_POST['password'];

    $user\_phone = $\_POST['phone\_number'];

    $user\_email = $\_POST['email\_address'];

    $user\_address = $\_POST['home\_address'];

    $user\_gender = $\_POST['gender'];

$user\_dob = $\_POST['dob'];

Get information from edit page

    $sql = "UPDATE userdetail SET

    user\_name='$user\_name',

    user\_phone='$user\_phone',

    user\_password='$user\_password',

    user\_email='$user\_email',

    user\_address='$user\_address',

    user\_gender='$user\_gender',

    user\_dob='$user\_dob'

WHERE user\_id='$user\_id'";

Update to the database.

    if (mysqli\_query($con, $sql)) {

        echo "<script>alert('Record updated!');

        window.location.href='admin\_view.php';</script>";}

else {echo "Error updating record: " . mysqli\_error($con);}

    mysqli\_close($con);

?>

# User Guidance

Before Login

Home Page

A screenshot of a phone

Description automatically generated

**Pressing this button sends user to home page.**

Event Page

A screenshot of a computer

Description automatically generated

**Pressing this button directs you to event page.**

Login Page

A screenshot of a login page

Description automatically generated

**3. Click this button to enter register page.**

**1.Pressing this button lets the user to login.**

**2.Fill in these boxes and click login.**

Donation Page

A screenshot of a computer

Description automatically generated

**1. Fill in the payment credentials to donate.**

**2. After done filling the boxes, press submit to submit.**

Register Page

A screenshot of a computer

Description automatically generated

**2. After done filling the blanks, click this button to submit.**

**1. Fill in the credential boxes**

After Login

User Home Page

A screenshot of a social media page

Description automatically generated

**Clicking this button brings user to donate page.**

**Clicking the profile buttons directs user to profile page.**

The profile button only will display after the user has login to an account, and the let me help button only will function after the user has login to an account. If the user has not logged in, it will direct the user to an alert page reminding the user to create an account.

User Event Page

A screenshot of a phone

Description automatically generated

**The feedback buttons bring user to feedback page.**

**The comment button bring user to comment page.**

Feedback Page

A screenshot of a computer

Description automatically generated

**2. After finish leaving feedback, press the submit button.**

**1.The user can leave their feedback here.**

Comment Page

A screenshot of a computer

Description automatically generated

**These are the feedbacks that user left.**

**The search buttons allow user to search feedbacks.**

Admin

Admin View User Page

A screenshot of a computer

Description automatically generated

**View payment button allows admin to view user’s payment info.**

**This button allows admin to view comments.**

**The search bar allows admin to search accounts**

**This button allows admin to add new contact.**

**The edit button edits the user account’s credential.**

**The delete button deletes the user account.**

Admin View Payment Page

A screenshot of a computer

Description automatically generated

**The search bar allows admin to search user’s account**

**The delete button deletes the user’s account.**

Admin View Comment Page

A screenshot of a chat

Description automatically generated

**These are comments left by users, and admin can view them.**

**The delete button deletes the user’s account.**

Admin Edit User Page

A screen shot of a contact

Description automatically generated

**The reset button clears data in the boxes.**

**These are the credential to the user’s account.**

**The submit button saves the changed to the account.**

Admin Add New Contact Page

A screenshot of a computer

Description automatically generated

**After inserting the boxes, click submit to create and account.**

**Admin can fill in these boxes to create new account.**

**The reset button clears the data in the boxes.**

# Conclusion

In conclusion, our website Charity Ease foundation is a charity foundation that is aimed towards helping those in their golden years live worry free and in comfort. Our website includes information about our organization and allows users to participate in this charity. Users can create their own personal accounts to be able to donate and leave feedbacks about their thoughts of our charity. Admins has access to all pages and user’s profile data and is able to edit and delete any user’s account. They also has the ability to edit and delete comments.

# References

*PHP MySQL select data*. (n.d.). Retrieved from https://www.w3schools.com/php/php\_mysql\_select.asp

# Workload Matrix

Group Number : 7

Group Leader Name : Alvan Sew

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Member Name and Student ID (including Group Leader) | Task Completed | | | | | Overall Contribution % |
| Documentation | Web App (Client-side) | Web (Server Side) | Database | |
| Alvan Sew  Tp072788 |  |  |  |  | | 50 |
| Tham King Joe  Tp072535 |  |  |  |  | | 29 |
| Enoch Kok Jia Xian  Tp068941 |  |  |  |  | | 11 |
| Wilson Tan Jun Yuan  Tp072665 |  |  |  |  | | 9 |
| Tancel Raj  Tp072214 |  |  |  |  | | 1 |
| Total | | | | |  | 100% |