

Web Development

Group 07



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
Design	5
Entity Relationship Diagram	5
Website Navigational Structure	5
Data Dictionary	6
Wire Frame	7
Before Login	7
Home Page.....	7
Event Page	8
Login Page	9
Register Page	10
After Login.....	11
User Home Page	11
User Event Page.....	12
Donate Page.....	13
Feedback Page	14
Comment Page.....	16
Admin	17
Admin View User Page	17
Admin View Comment Page	18
Admin View Payment Page.....	19
Admin Edit User Page	20
Admin Add Contact Page	21
Implementation	23
Before Login	23
Login Page	23
Register Page	25
Log Out\.....	27
Home Page.....	27
Event Page	29
Web Header	31
Delete page	32
User Home Page (After user Login).....	33

User Event Page (After user Login)	34
Donation Page.....	35
User Profile Page	36
Feedback Page	39
Comments Page	40
Edit Page.....	42
Update php function	44
User Guidance	45
Conclusion	57
References	58
Workload Matrix	59

Introduction

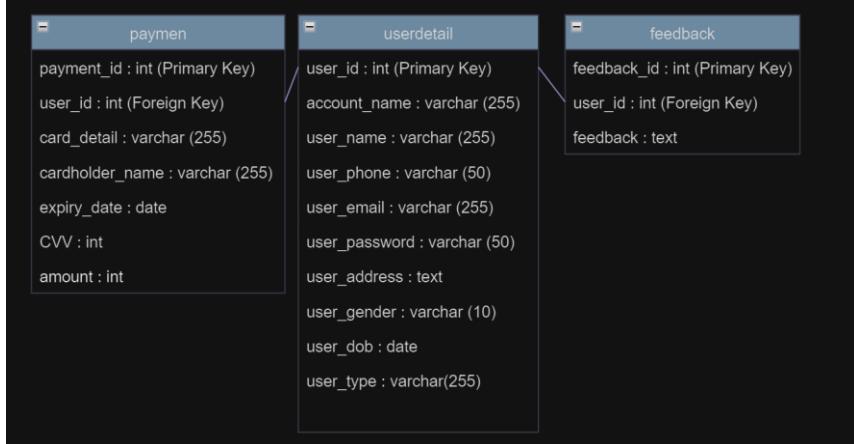
Charity Ease foundation is a charity foundation that is aimed towards helping those in their golden years live worry free and in comfort. We do this by collecting and donating funds to various old folks' homes with the intention of lifting their financial burdens and providing them with enough funds 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.

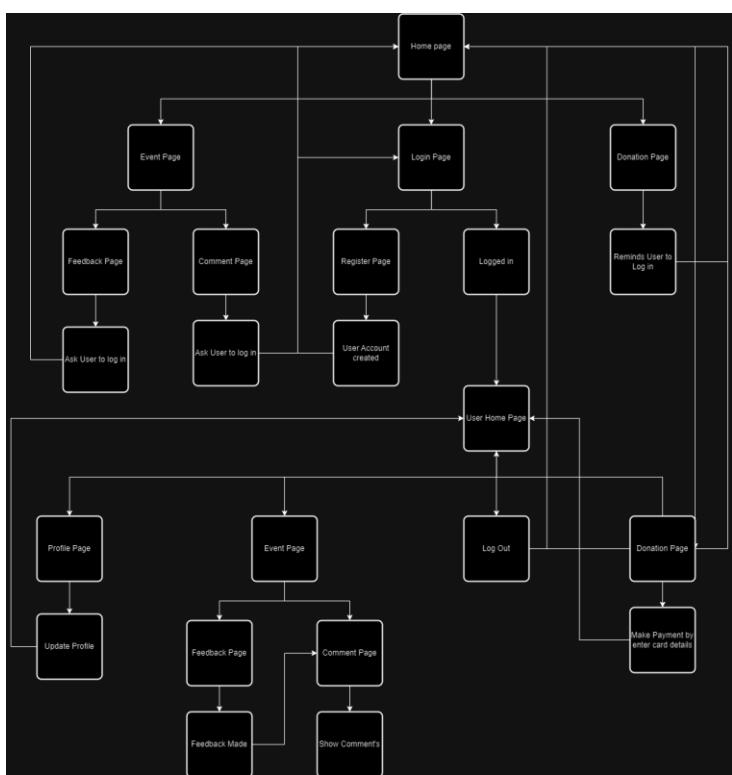
Registered customers can deposit funds into our website which will then be channelled to various old folks' homes. Customers will be able to change and edit their account profiles whenever they wish. Customers will also be able to provide feedback which will then be reviewed and used to improve our services.

Design

Entity Relationship Diagram



Website Navigational Structure



Data Dictionary

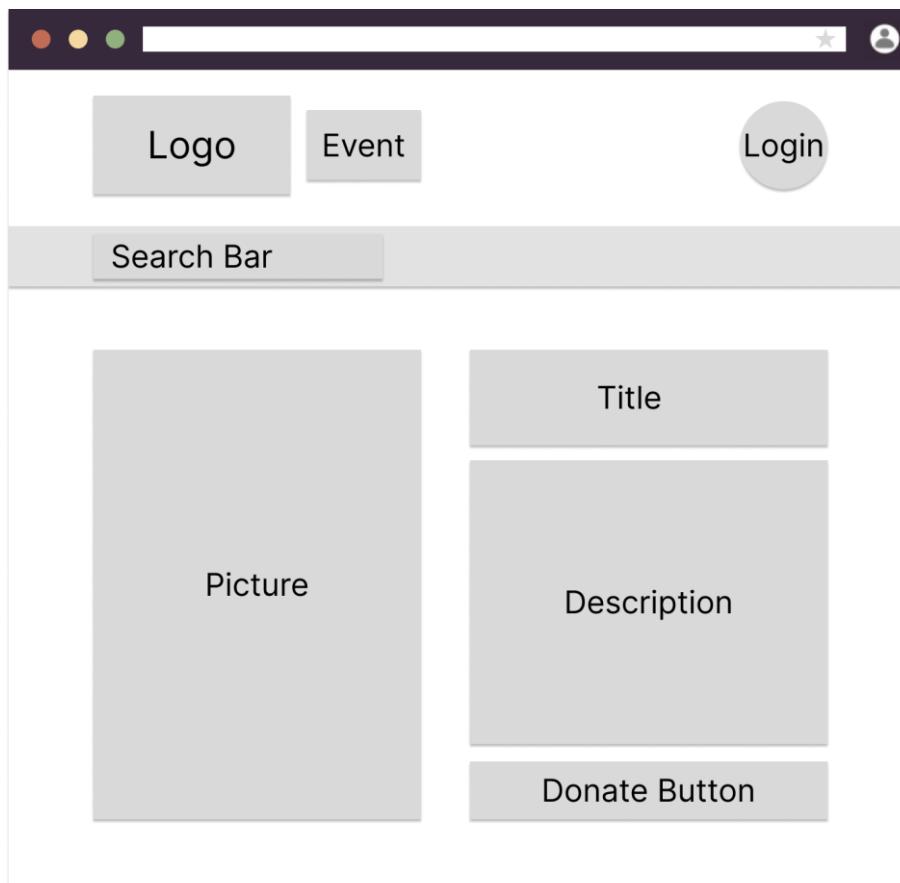
Field Name	Data Type	Field Length	Constraint	Description
user_id	Int		Public Key	Id number of user
account_name	Varchar	255	Not Null	The identification that the user would prefer in the website
user_name	Varchar	255	Not Null	Name of user
user_phone	Varchar	50	Not Null	Phone number of user
user_email	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.

Wire Frame

Before Login

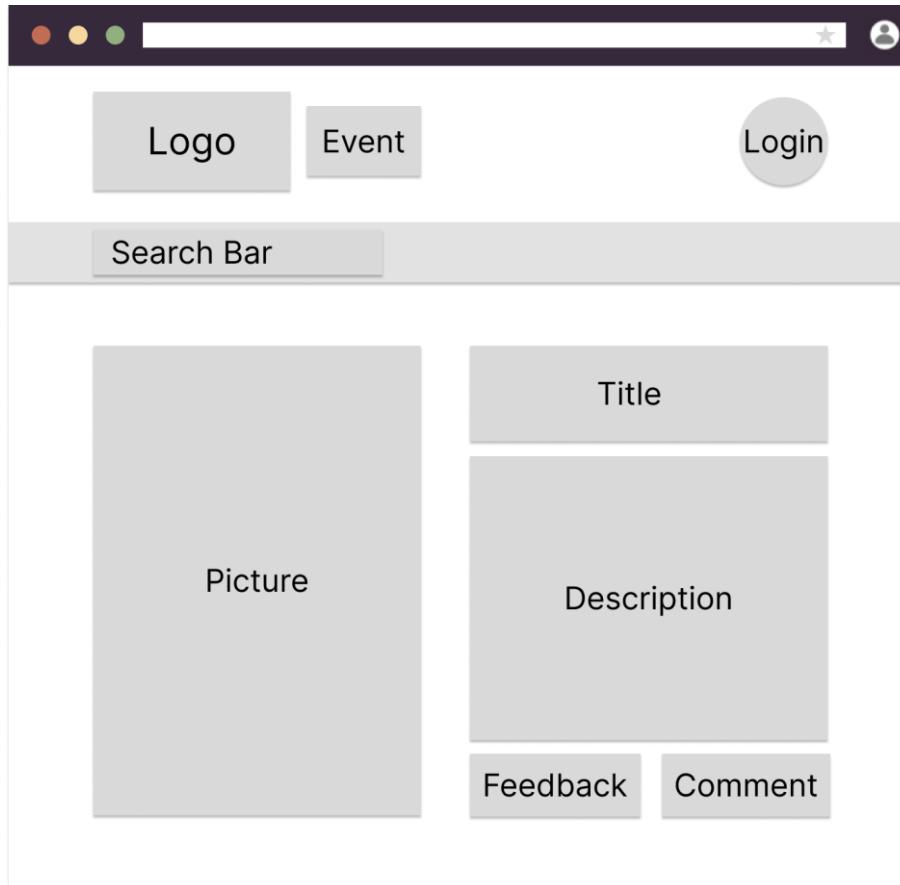
In this section, the following shows a rough sketch of what the UI UX of the website, showing the location of all the buttons would have access to before the user have created an account in our website

Home Page

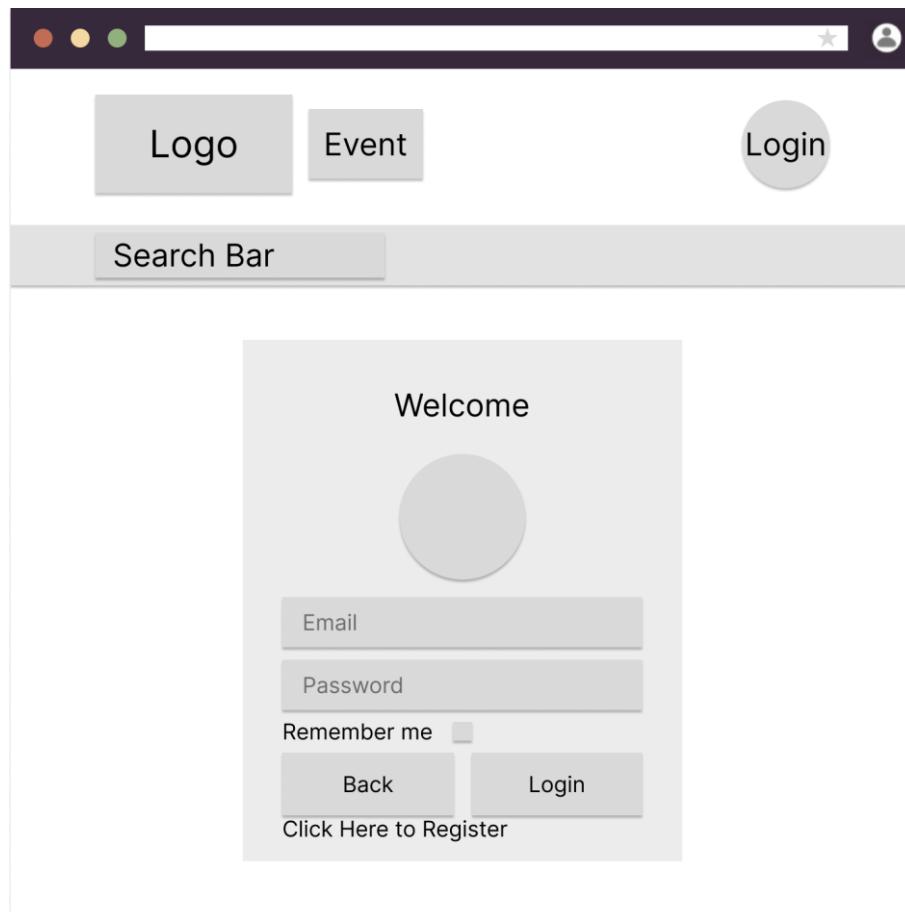


In our home page, it will display buttons that redirect you to other function pages, such as event, login and donate page.

Event Page



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

Our 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

Welcome

Name
Username
Phone Number
Email Address
Password
Home Address

Gender
 Male
 Female

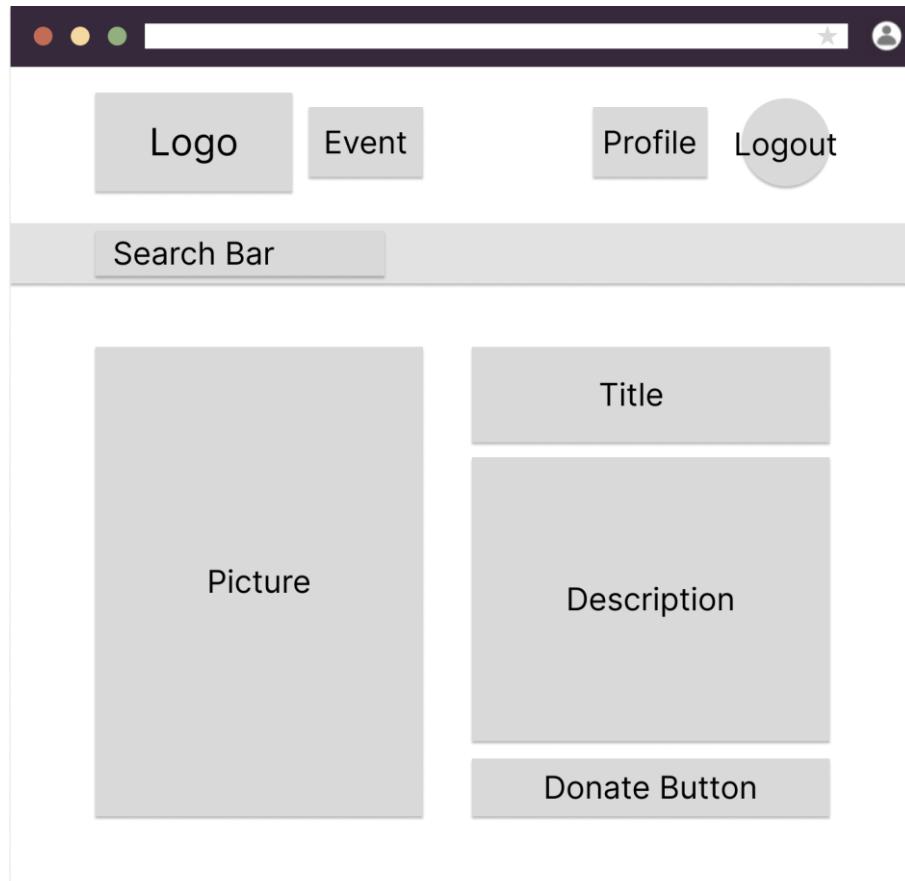
Date of Birth

Submit
Reset

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.

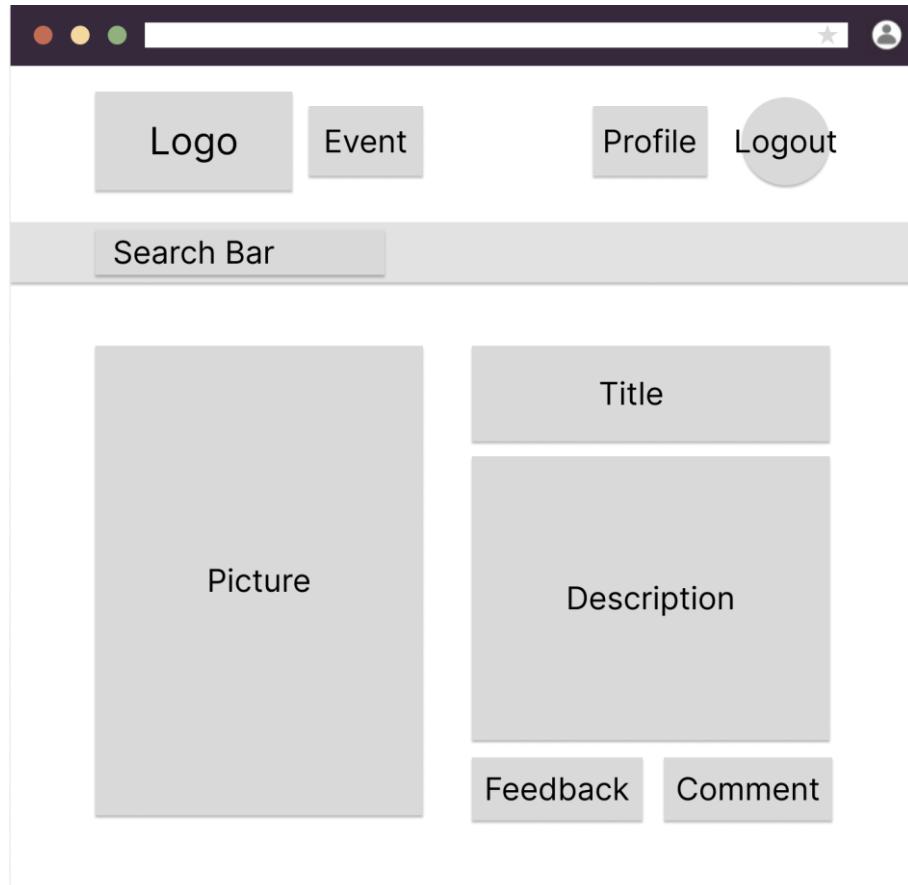
After Login

User Home Page



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



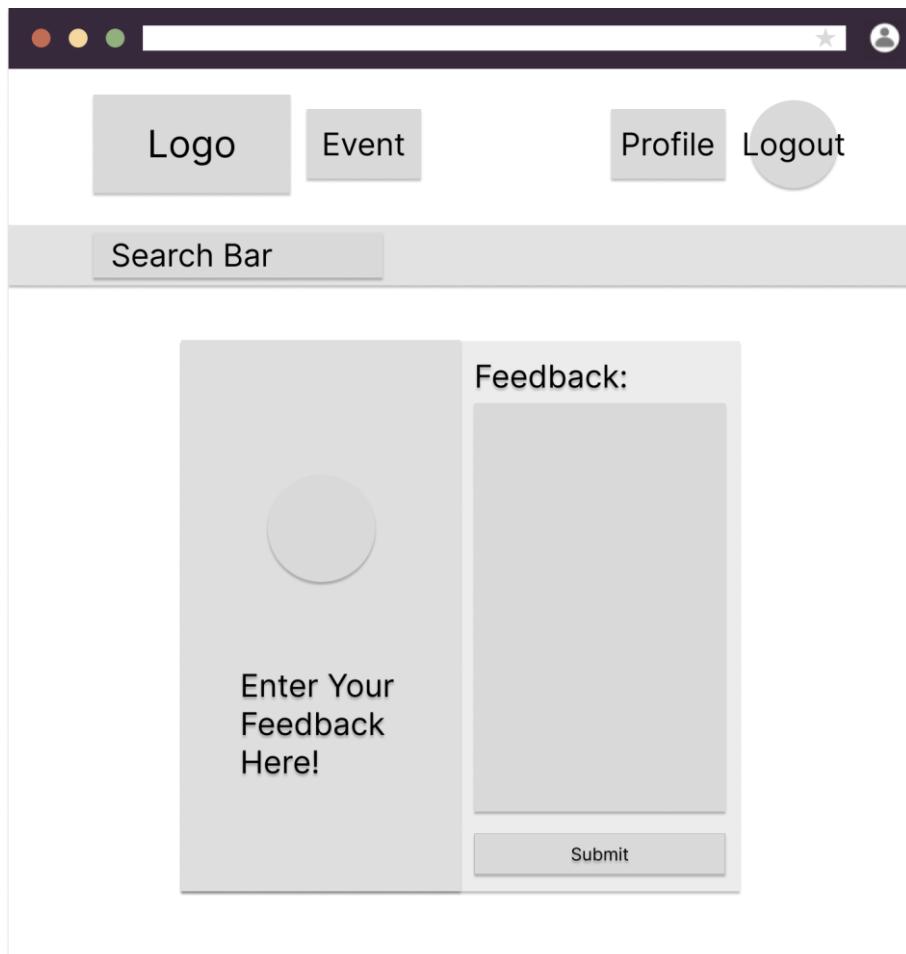
The user event page displays every detail about our event. The feedback button allows the user to leave a feedback. The comment button brings the user to the comment section where they can review their comment or other people's comment about our event.

Donate Page

The screenshot shows a web application interface. At the top, there is a dark header bar with three colored dots (red, yellow, green) on the left and a user icon on the right. Below the header, there are four buttons: 'Logo', 'Event', 'Profile', and 'Logout'. A 'Search Bar' is positioned below these buttons. The main content area is titled 'Payment Section' and contains five input fields: 'Card Details', 'Cardholder Name', 'Expiry Date', 'CVV', and 'Amount'. At the bottom of this section is a 'Submit' button.

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

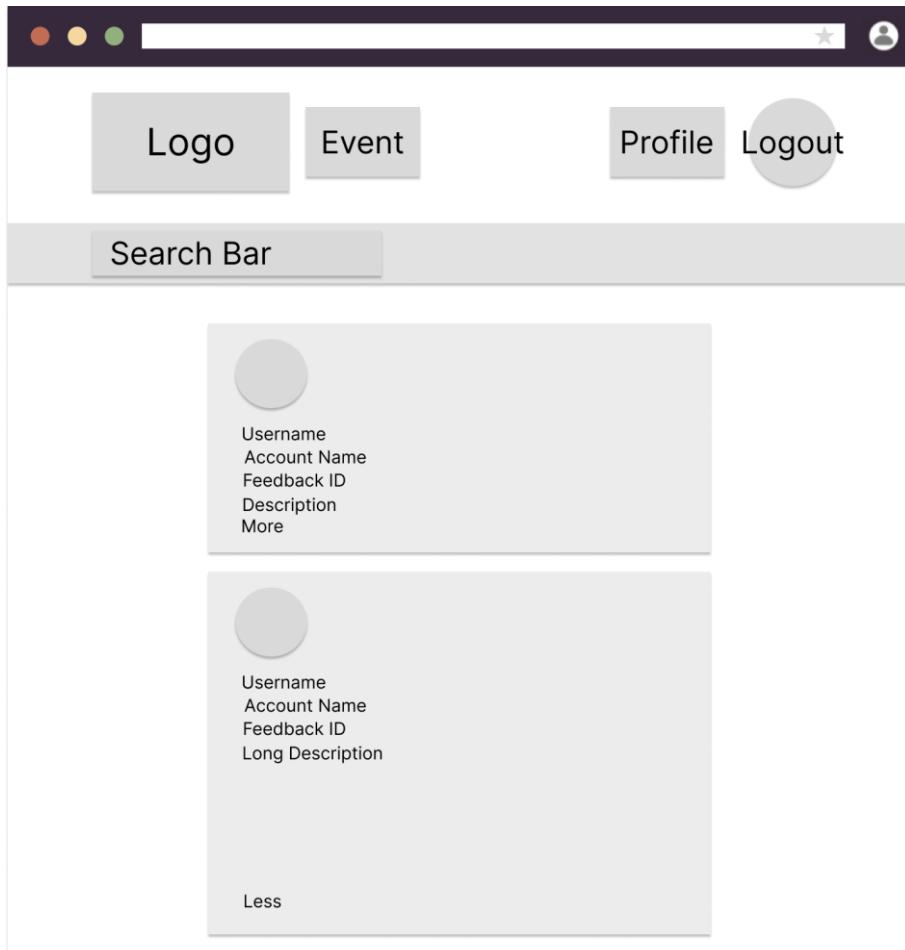


After entering the feedback page, the user is allowed to leave feedbacks to the events that they have thoughts about. Pressing submit will send the feedback to database which will be explained more technically in the future sections.

Profile Page

The screenshot shows a web browser window with a dark header bar containing three dots (red, yellow, green) and a user icon. Below the header is a navigation bar with four buttons: 'Logo', 'Event', 'Profile', and 'Logout'. A 'Search Bar' is positioned below the navigation bar. The main content area contains a form for updating profile information. The fields include:

- Name (text input)
- Username (text input)
- Phone Number (text input)
- Email Address (text input)
- Password (text input)
- Home Address (text input)
- Gender (radio buttons for Male and Female, with Male selected)
- Date of Birth (text input)
- Update Profile (button)

Comment Page

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

Add New Contact	View Comments	View Payments				
<p>Search: <input type="text"/> <input type="button" value="Search"/></p> <hr/> <table border="1"><tr><td> Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/></td><td> Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/></td><td> Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/></td></tr><tr><td> Username Account Name Phone Number Email Password</td><td> Username Account Name Phone Number Email Password</td><td> Username Account Name Phone Number Email Password</td></tr></table>	 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>	 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>	 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>	 Username Account Name Phone Number Email Password	 Username Account Name Phone Number Email Password	 Username Account Name Phone Number Email Password
 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>	 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>	 Username Account Name Phone Number Email Password Address Gender Date of Birth <input type="button" value="Delete"/> <input type="button" value="Edit"/>				
 Username Account Name Phone Number Email Password	 Username Account Name Phone Number Email Password	 Username Account Name Phone Number Email Password				

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

The screenshot shows a web page titled "Admin View Comment Page". At the top, there are two buttons: "View User" and "View Comments". Below them is a search bar with the placeholder "Search:" and a "Search" button. The main content area displays two comments, each with a user profile picture and a list of fields: Username, Account Name, Feedback ID, Description, and More. The second comment also includes a Long Description field. A "Less" link is visible at the bottom of the second comment's description section.

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

View User View Comments

Search:

 Username Account Name Card Detail Card Holder_name Expiry Date CVV Amount <input type="button" value="Delete"/>	 Username Account Name Card Detail Card Holder_name Expiry Date CVV Amount <input type="button" value="Delete"/>	 Username Account Name Card Detail Card Holder_name Expiry Date CVV Amount <input type="button" value="Delete"/>
 Username Account Name Card Detail Card Holder_name	 Username Account Name Card Detail Card Holder_name	 Username Account Name Card Detail Card Holder_name

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

Edit Contact

Name

Password

Phone Number

Email Address

Home Address

Gender
 Male
 Female

Date of Birth

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

Edit Contact

Name

Password

Phone Number

Email Address

Home Address

Gender
 Male
 Female

Date of Birth

Add New User

Name

Password

Phone Number

Email Address

Home Address

Gender
 Male
 Female

Date of Birth

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('Successfull!'); 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'">  
        
    </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'">  
        
    </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'">  
    
</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'">  
    
</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">
    <!-- 
        we cancel it-->
</div>
<div class="preview-image">
    
</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">
    <!-- 
        we cancel it-->
</div>
</div>
```

Home Button (homebtn)

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

Formatted: Heading 3

Login Button (loginbtn)

```
<button class="loginbtn" onclick="window.location.href='login.php'">
    
</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 user_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" >
    
  </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'">
  
</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">
    
  </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>
  <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 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.
2. Database Connection:
The script includes the **conn.php** file to connect to the database.
3. 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.
4. Sanitize User Input:
The script sanitizes the user input to prevent SQL injection using **mysqli_real_escape_string()**.
5. Insert Payment Information:
It constructs and executes an SQL query to insert the payment information into the database.
6. 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.
7. 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.

2. Retrieve User ID:

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

3. Form Submission Check:

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

4. 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);
}
```

5. 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.";
}
```

6. 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.

2. 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');
}
```

3. Sanitize User Input:

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

4. 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>";
}
```

5. 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 '';
        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 . '">More</span>';
        echo '</div>';
    } else {
        echo '<div id="box">';
        echo '';
        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'];**).

2. Database Connection:

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

3. 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%'";**).

4. 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.

5. Close Database Connection:

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

6. 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.

2. Retrieve User ID:

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

3. 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);**.

4. 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 -->
```

5. 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'";
```

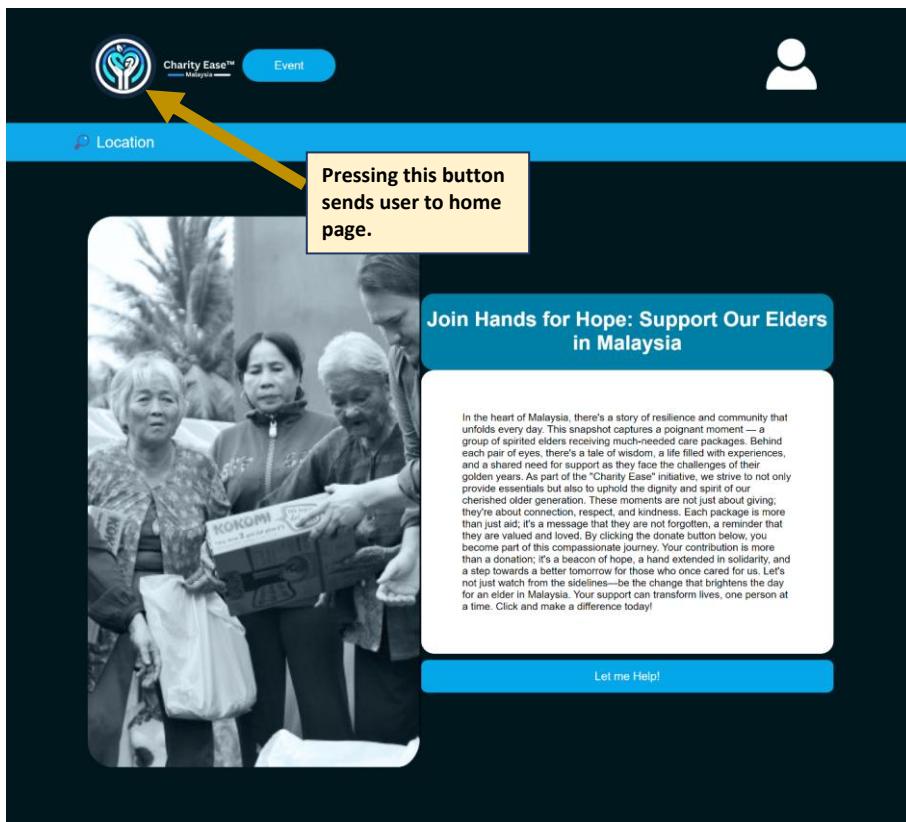
Update to the database.

```
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);
?>
```

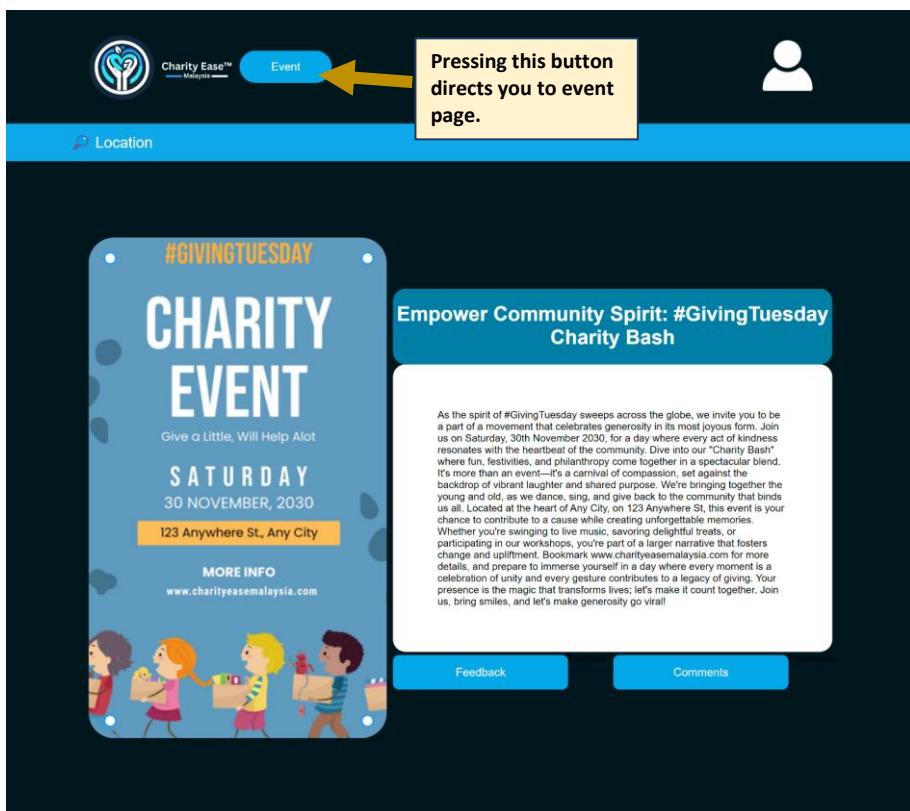
User Guidance

Before Login

Home Page



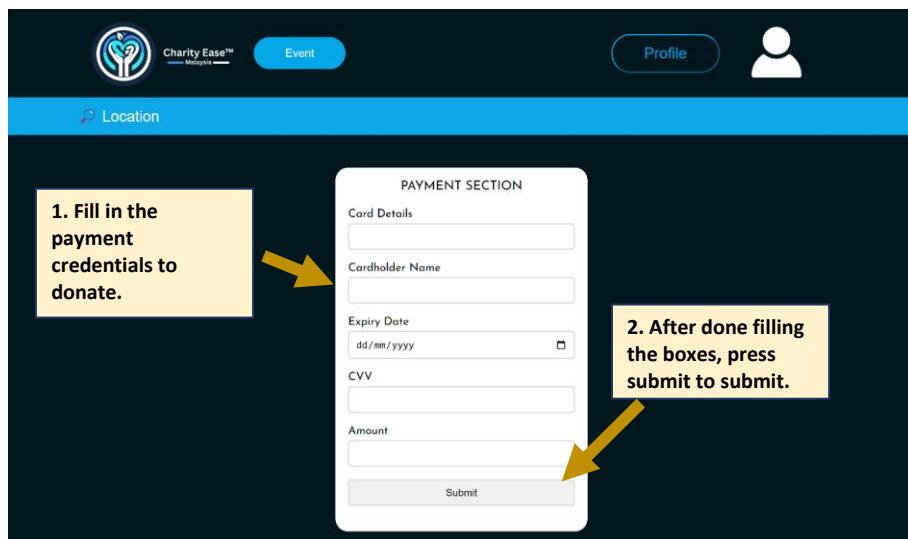
Event Page



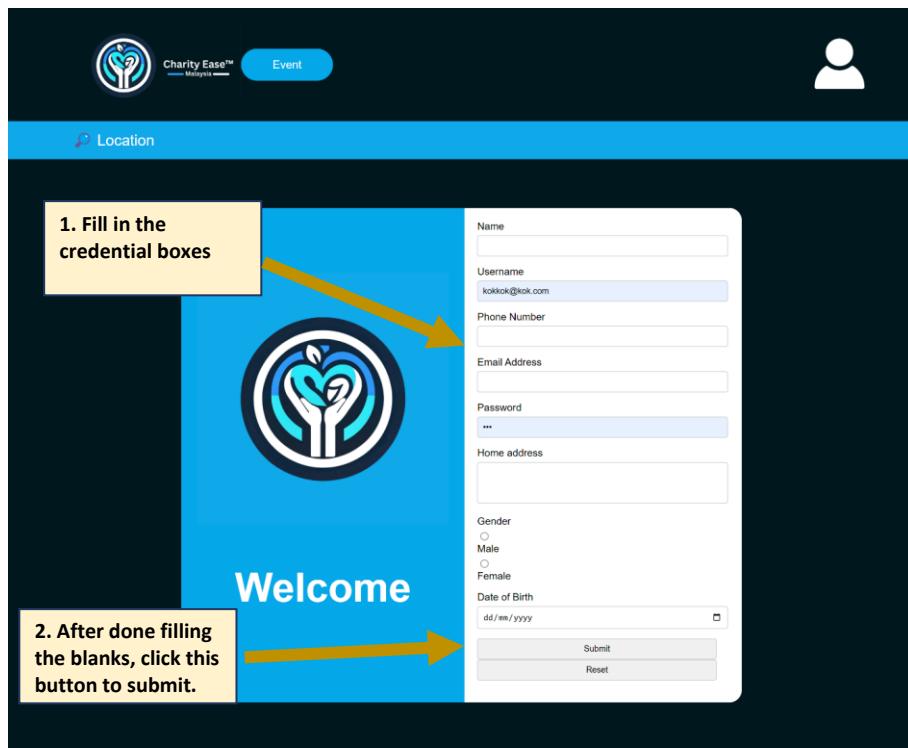
Login Page



Donation Page



Register Page



The screenshot shows a registration form for 'Charity Ease™ Malaysia'. At the top left is the logo with the text 'Charity Ease™' and 'Malaysia'. To its right are buttons for 'Event' and a user profile icon. Below the header is a blue bar with a location pin icon and the word 'Location'. The main area has a dark background with a large blue 'Welcome' banner containing a circular logo with hands holding a heart. To the left of the banner is a yellow callout box with the text '1. Fill in the credential boxes'. A yellow arrow points from this box to the credential input fields on the right. To the right of the banner is another yellow callout box with the text '2. After done filling the blanks, click this button to submit.' A yellow arrow points from this box to the 'Submit' button at the bottom of the form. The form itself contains fields for Name, Username (with 'kokkok@kok.com' entered), Phone Number, Email Address, Password, Home address, Gender (Male and Female options), Date of Birth (dd/mm/yyyy format), and two buttons at the bottom labeled 'Submit' and 'Reset'.

1. Fill in the credential boxes

Welcome

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

Name

Username
kokkok@kok.com

Phone Number

Email Address

Password

Home address

Gender

Male

Female

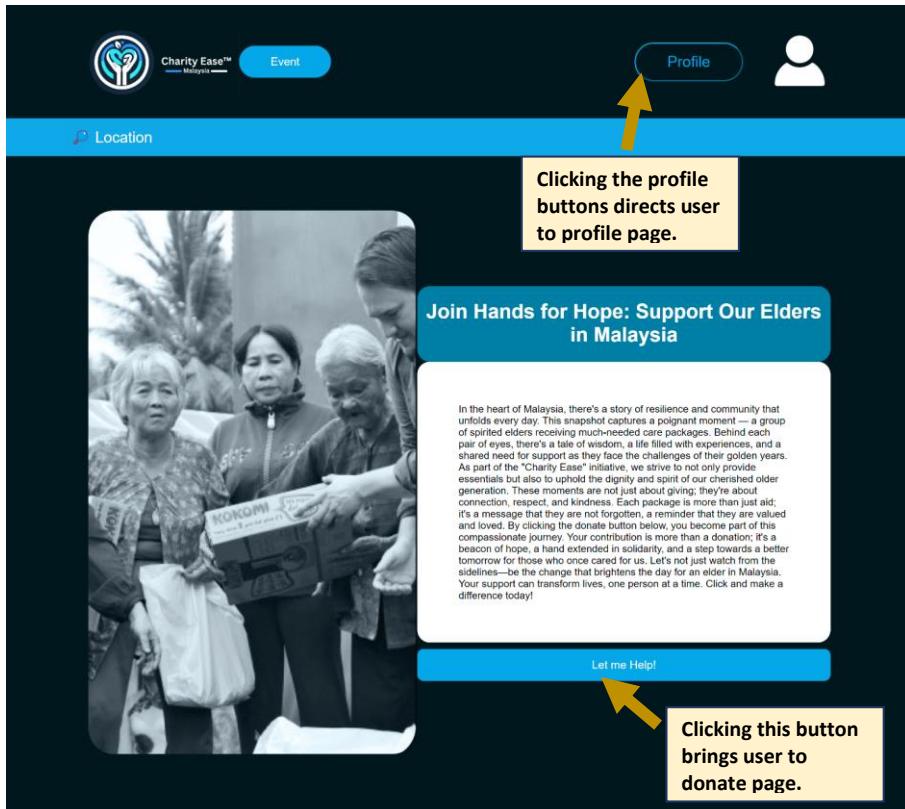
Date of Birth
dd/mm/yyyy

Submit

Reset

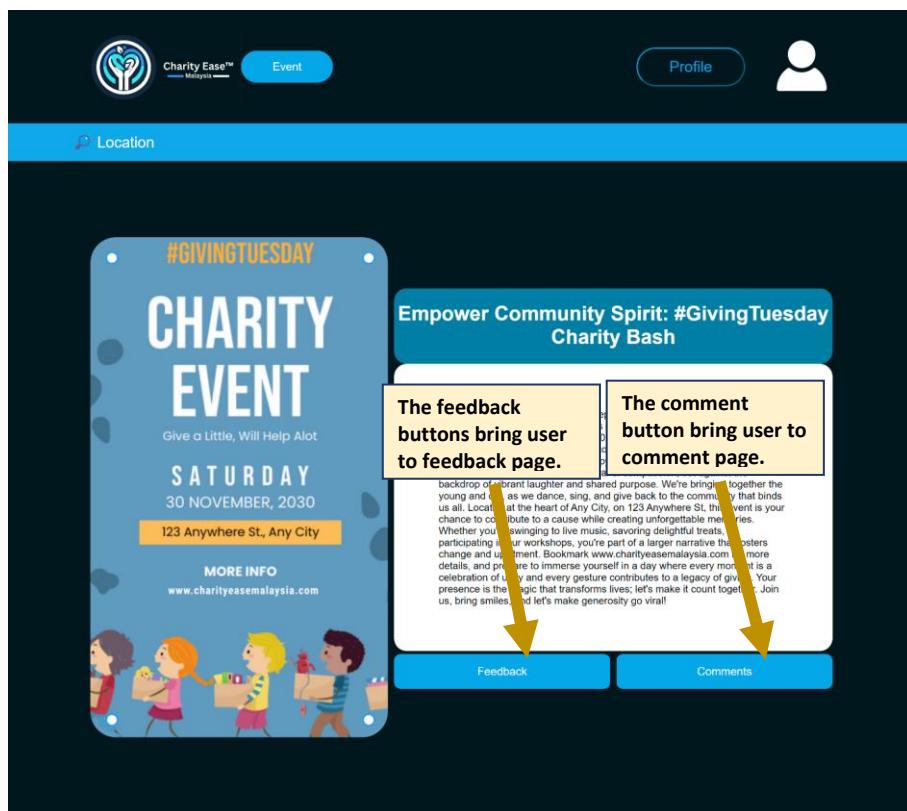
After Login

User Home Page

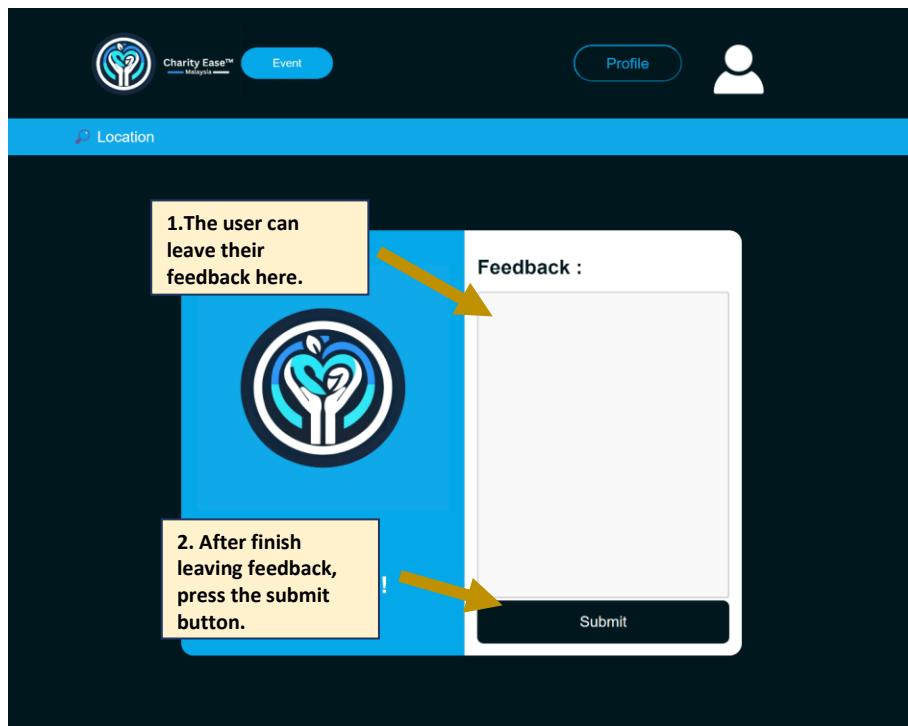


The profile button only will display after the user has login to an account, and the let me help button only will function after the user has login to an account. If the user has not logged in, it will direct the user to an alert page reminding the user to create an account.

User Event Page



Feedback Page



Comment Page

The screenshot shows a mobile application interface for the "Charity Ease™ Malaysia" platform. At the top, there is a navigation bar with a logo, a "Location" button, a "Search" bar, a "Profile" button, and a user icon. Below the navigation bar, the main content area displays three user comments in separate boxes:

- User 1:** bro1
Account Name: bro
3
hi
- User 2:** ahxianah
Account Name: Enoch Kok Jia Xian
4
Ah xian ah, 吃饱了没有啊 ah xian,
- User 3:** ahxianah
Account Name: Enoch Kok Jia Xian
9
AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH,
More

Two yellow arrows point to specific elements: one arrow points to the "Search" button at the top, with the text "The search buttons allow user to search feedbacks."; another arrow points to the third comment box, with the text "These are the feedbacks that user left."

Admin

Admin View User Page

The screenshot shows a web page titled "Admin View User Page". At the top, there is a search bar labeled "Search:" and three buttons: "Add New Contact", "View Comments", and "View Payment". Below the search bar, there is a table displaying user account information.

User Account Data:

- Row 1:** Not.John04, Account Name: Tham King Joe, Phone: 0127507092, Email: thamkingjoe9@gmail.com, Password: acca, Address: 29, Jalan 2/149L 2 Jalan 2/32, Gender: Male, Date of Birth: 2000-00-00. Buttons: Delete, Edit.
- Row 2:** Not.John04, Account Name: Tham King Joe, Phone: 0127507092, Email: thamkingjoe9@gmail.com, Password: efev, Address: 29, Jalan 2/149L 2 Jalan 2/32, Gender: Male, Date of Birth: 2000-00-00. Buttons: Delete, Edit.
- Row 3:** Not.John04, Account Name: Tham King Joe, Phone: 0127507092, Email: thamkingjoe9@gmail.com, Password: sdyvd, Address: 29, Jalan 2/149L 2 Jalan 2/32, Gender: Male, Date of Birth: 2000-00-00. Buttons: Delete, Edit.
- Row 4:** Not.John04, Account Name: Tham King Joe, Phone: 0127507092, Email: thamkingjoe9@gmail.com, Password: 123, Address: 29, Jalan 2/149L 2 Jalan 2/32, Gender: Male, Date of Birth: 2024-06-01. Buttons: Delete, Edit.
- Row 5:** WaifuProtector, Account Name: Pencil, Phone: 0128013120, Email: thehighgroundmaster3@gmail.com, Password: damn, Address: damn, Gender: Male, Date of Birth: 2024-06-01. Buttons: Delete, Edit.
- Row 6:** bro1, Account Name: bro, Phone: 123, Email: bro@gmail.com, Password: bro, Address: bro, Gender: Male, Date of Birth: 2024-06-01. Buttons: Delete, Edit.
- Row 7:** Daddy Wil, Account Name: Wilson Tan Jun Yuan, Phone: 1231289371298, Email: dhoncnacocwo@gmail.com, Password: 123, Address: cayuivbiawbinusabow, Gender: Female, Date of Birth: 2024-06-01. Buttons: Delete, Edit.
- Row 8:** ahxianah, Account Name: Enoch Kok Jia Xian, Phone: 0122104323, Email: kokkok@kok.com, Password: 123, Address: 29, Jalan 2/149L 2 Jalan 2/32, Gender: Male, Date of Birth: 2024-06-02. Buttons: Delete, Edit.

Annotations:

- A yellow box with a double-headed arrow between the "View Comments" and "View Payment" buttons contains the text: "This button allows admin to add new contact".
- A yellow box containing the text: "The search bar allows admin to search accounts" has a double-headed arrow pointing to the search bar.
- A yellow box containing the text: "This button allows admin to view comments." has a double-headed arrow pointing to the "View Comments" button.
- A yellow box containing the text: "View payment button allows admin to view user's payment info." has a double-headed arrow pointing to the "View Payment" button.
- A yellow box containing the text: "The delete button deletes the user account." has a double-headed arrow pointing to the "Delete" button in the fourth row.
- A yellow box containing the text: "The edit button edits the user account's credential." has a double-headed arrow pointing to the "Edit" button in the fourth row.

Admin View Payment Page

The screenshot shows a web page titled "Admin View Payment Page". At the top, there are two links: "View User" and "View Comments". Below these is a search bar with the placeholder "Search:" and a magnifying glass icon. A yellow arrow points from this search bar to a callout box that reads: "The search bar allows admin to search user's account". The main content area displays four payment account entries, each in a blue card-like box:

- bro1**
Account Name: bro
Card Detail: 123
Card Holder_name: aeuihfuqeqe
Expiry Date: 2024-06-07
CVV: 0
Amount: 123
[Delete](#)
- bro1**
Account Name: bro
Card Detail: 123
Card Holder_name: aeunafe
Expiry Date: 2024-06-14
CVV: 0
Amount: 123
[Delete](#)
- ahxianah**
Account Name: Enoch Kok Jia Xian
Card Detail: 0971033571108924
Card Holder_name: ahxianah
Expiry Date: 2024-06-11
CVV: 0
Amount: 124214
[Delete](#)
- ahxianah**
Account Name: Enoch Kok Jia Xian
Card Detail: 089239012821904
Card Holder_name: kokkok
Expiry Date: 2024-06-02
CVV: 0
Amount: 123
[Delete](#)

A yellow arrow points from the "Delete" button in the third account entry to a callout box that reads: "The delete button deletes the user's account."

Admin View Comment Page

View User View Payment

Search:

The screenshot shows a list of users with their account names, comments, and a 'Delete' button. A yellow arrow points from a callout box to the 'Delete' button in the first profile.

User	Comments	Action
bro1	3 hi	<input type="button" value="Delete"/>
ahxianah	4 Ah xian ah, 吃饱了没有啊 ah xian,	<input type="button" value="Delete"/>
ahxianah	9 AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, AHXIANAH, <input type="button" value="More"/> <input type="button" value="Delete"/>	<input type="button" value="Delete"/>

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

The delete button deletes the user's account.

Admin Edit User Page

Edit Contact

Name	NotJohn04
Password	*****
Phone Number	0127507092
Email Address	thekingof9@gmail.com
Home Address	29, Jalan 2/149L 2 Jalan 2/32z,
Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Date Of Birth	23/11/2004 <input type="button" value="Change"/>
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

The submit button saves the changed to the account.

The reset button clears data in the boxes.

These are the credential to the user's account.

Admin Add New Contact Page

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.

ADD NEW USER

Name

Username
kokko@rok.com

Phone Number

Email Address

Password
...

Home address

Gender
 Male
 Female

Date of Birth
dd / mm / yyyy

Submit Reset

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 Joe Tp072535	•	•			29
Enoch Kok Jia Xian Xian Tp068941	•				11
Wilson Tan Jun Yuan Yuan Tp072665	•				9
Tancel Raj Tp072214	•				1
Total					100%