



PATUAHALI SCIENCE AND TECHNOLOGY UNIVERSITY

COURSE CODE CCE-224

SUBMITTED TO:

Dr. Md. Samsuzzaman Sobuz

Professor, Department of Computer and Communication
Engineering
Faculty of Computer Science and Engineering

SUBMITTED BY:

Name: MD Noushad Bhuiyan

ID: 2102038

Reg No: 10165

Assignment Topic: Database Project Details

Title: A Food Donation and Distribution Platform

Description: This project is a web-based platform designed to connect food donors—such as individuals, restaurants, and supermarkets—with nonprofit organizations and volunteers working to fight hunger and reduce food waste. The system allows users to register, post available food items, chat with organizations, and rate services. It aims to simplify the process of food redistribution, promote community welfare, and ensure that excess food reaches those who need it most.

Core Features

1. User Registration and Login

- Separate sign-up and login system for users and organizations.
 - Profile management (profile image, contact info, etc.).
 - Session handling and authentication.
-

2. Organization Listing Page

- Displays list of registered food-distributing organizations.
 - Real-time search and pagination.
 - Organization details: name, image, location, and contact info.
-

3. Messaging System (Chat Room)

- Users can send and receive messages from organizations.
- Each user sees a list of users or organizations they've chatted with.
- Chat UI with message alignment (sender vs. receiver).

- Message history fetched on page load.
-

4. Rating and Review System

- Users can rate organizations (1 to 5 stars).
 - Average rating and total ratings displayed dynamically.
 - Re-rating updates previous rating without changing count.
 - Ratings are fetched and updated in real time when visiting the org page.
-

5. Image Slider on Homepage

- Rotating banner of 8 images with donation awareness messages.
 - Slides every 3–5 seconds automatically.
-

6. Admin / Organization Panel (Optional / Extendable)

- Organizations can manage their profile and communication.
 - Can view ratings they've received.
-

7. Responsive Frontend Design

- Organized UI using HTML, CSS, and JavaScript.
- Navigation bar, chat interface, rating panel, image slider, etc.
- Mobile-friendly layout.

8. Backend Integration (PHP + MySQL)

- Uses prepared statements to prevent SQL injection.
 - MySQL relational database with tables: users, organization, messages, ratings.
-

9. Error Handling & Validations

- Form validation for empty inputs.
- Prevent sending messages to self.
- Handles edge cases like no chat history or no rating.

All The Table Creation Operation Codes for MySQL Database:

```
create database donation_website;
use donation_website;
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    full_name VARCHAR(100) NOT NULL,
    username VARCHAR(50) NOT NULL,
    email VARCHAR(100) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,
    address VARCHAR(255) NOT NULL,
    dob DATE NOT NULL,
    user_type ENUM('donator', 'organization') NOT NULL,
    org_name VARCHAR(100),
    org_location VARCHAR(255),
    org_phone VARCHAR(20),
    profile_picture VARCHAR(255),
    org_image VARCHAR(255),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```
CREATE TABLE messages (
    id INT AUTO_INCREMENT PRIMARY KEY,
    sender_id INT NOT NULL,
    receiver_id INT NOT NULL,
    message TEXT NOT NULL,
    is_read TINYINT(1) DEFAULT 0,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
);
```

```
CREATE TABLE posts (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    content TEXT NOT NULL,
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);
```

```
CREATE TABLE organization (
    id INT AUTO_INCREMENT PRIMARY KEY,
```

```
user_id INT NOT NULL,  
org_name VARCHAR(255) NOT NULL,  
org_location VARCHAR(255) NOT NULL,  
org_phone VARCHAR(20) NOT NULL,  
keyword VARCHAR(255),  
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE  
);
```

```
ALTER TABLE users ADD profile_picture VARCHAR(255) DEFAULT NULL;
```

```
ALTER TABLE users ADD COLUMN org_image VARCHAR(255) DEFAULT NULL;
```

```
ALTER TABLE organization ADD COLUMN org_image VARCHAR(255);
```

```
CREATE TABLE messages (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    sender_id INT NOT NULL,  
    receiver_id INT NOT NULL,  
    message TEXT NOT NULL,  
    is_read TINYINT(1) DEFAULT 0,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
);
```

```
INSERT INTO messages (sender_id, receiver_id, message)  
VALUES (1, 2, 'Hello from user 4 to user 1');
```

```
CREATE TABLE messages (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    sender_id INT NOT NULL,  
    receiver_id INT NOT NULL,  
    message TEXT NOT NULL,  
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,  
    is_read TINYINT DEFAULT 0  
);
```

```
ALTER TABLE organization ADD COLUMN rating_total INT DEFAULT 0;  
ALTER TABLE organization ADD COLUMN rating_count INT DEFAULT 0;
```

```

-- Drop old columns if present (be sure you want this!)
ALTER TABLE organization DROP COLUMN rating_total;
ALTER TABLE organization DROP COLUMN rating_count;

-- Create the ratings table
CREATE TABLE ratings (
    id INT AUTO_INCREMENT PRIMARY KEY,
    org_id INT NOT NULL,
    rater_id INT NOT NULL,
    user_rating INT NOT NULL CHECK (user_rating BETWEEN 1 AND 5),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (org_id) REFERENCES organization(id),
    FOREIGN KEY (rater_id) REFERENCES users(id)
);
ALTER TABLE messages CHANGE message message TEXT;

CREATE TABLE user_chats (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    chat_user_id INT NOT NULL,
    UNIQUE (user_id, chat_user_id)
);
INSERT IGNORE INTO user_chats (user_id, chat_user_id) VALUES ($receiver_id,
$sender_id);

```

All Data fetching and updating code from database:

db.php:

```
<?php  
$host = 'localhost';  
$user = 'root';  
$pass = '';  
$db = 'donation_website';  
  
$conn = new mysqli($host, $user, $pass, $db);  
if ($conn->connect_error) {  
    die("Connection failed: " . $conn->connect_error);  
}  
?>
```

fetch_messages.php:

```
<?php  
session_start();  
include 'db.php';  
$sender_id = $_SESSION['user_id'];  
$receiver_id = $_GET['user_id'];  
  
$result = $conn->query("SELECT * FROM messages  
WHERE (sender_id = $sender_id AND receiver_id = $receiver_id)  
    OR (sender_id = $receiver_id AND receiver_id = $sender_id)  
ORDER BY created_at ASC");  
  
$update_query = "UPDATE messages  
    SET is_read = 1  
    WHERE sender_id = $receiver_id  
        AND receiver_id = $sender_id  
        AND is_read = 0";  
$conn->query($update_query);  
  
while ($row = $result->fetch_assoc()) {  
    $isMine = $row['sender_id'] == $sender_id;  
    $bubbleClass = $isMine ? "message-bubble message-mine" : "message-bubble  
message-other";
```

```

$icon = $isMine ? "❶" : "❷";

echo "<div class='bubbleClass'>
    <b>$icon</b> " . htmlspecialchars($row['message']) . "
</div>";
}

?>

```

Send messages.php:

```

<?php
session_start();
include 'db.php';
$sender_id = $_SESSION['user_id'];
$receiver_id = $_POST['receiver_id'];
$message = $_POST['message'];

$stmt = $conn->prepare("INSERT INTO messages (sender_id, receiver_id,
message) VALUES (?, ?, ?)");
$stmt->bind_param("iis", $sender_id, $receiver_id, $message);
$stmt->execute();

```

check_organization.php:

```

<?php
session_start();
require 'db_connect.php';

if (!isset($_SESSION['email'])) {
    echo json_encode(["success" => false, "error" => "User not logged in"]);
    exit;
}

// $email = $_SESSION['user_email'];

$email = $_SESSION['email'];

// Retrieve the email from the 'users' table based on the session email
$query = "SELECT email FROM users WHERE id = ?";
$stmt = $conn->prepare($query);
$stmt->bind_param("s", $id);

```

```

$stmt->execute();
$stmt->bind_result($userEmail);
$stmt->fetch();

// Use the fetched email from the 'users' table
$email = $_POST[$userEmail]; // Now $email is set with the value from the
users table

$orgName = $_POST['orgName'];
$orgDescription = $_POST['orgDescription'];
$donationType = $_POST['donationType'];
$contactNumber = $_POST['contactNumber'];
$orgImage = "";

if (!empty($_FILES['orgImage']['name'])) {
    $orgImage = "uploads/organization_images/" . time() . "_" .
$_FILES['orgImage']['name'];
    move_uploaded_file($_FILES['orgImage']['tmp_name'], "../" . $orgImage);
}

$query = "INSERT INTO organizationDetails (user_email, orgName,
orgDescription, donationType, contactNumber, orgImage)
VALUES (?, ?, ?, ?, ?, ?) ON DUPLICATE KEY UPDATE orgName=?,
orgDescription=?, donationType=?, contactNumber=?, orgImage=?";
$stmt = $conn->prepare($query);
$stmt->bind_param("sssssss", $email, $orgName, $orgDescription,
$donationType, $contactNumber, $orgImage, $orgName, $orgDescription,
$donationType, $contactNumber, $orgImage);

echo json_encode(["success" => $stmt->execute()]);
?>

```

create_post.php:

```

<?php
session_start();
require 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: ../pages/login.html");
    exit();
}

```

```

}

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $user_id = $_SESSION['user_id'];
    $content = trim($_POST['content']);

    if (!empty($content)) {
        $stmt = $conn->prepare("INSERT INTO posts (user_id, content) VALUES (?, ?)");
        $stmt->bind_param("is", $user_id, $content);
        $stmt->execute();
        $stmt->close();
    }

    header("Location: ../pages/home.html");
    exit();
}

?>

```

Delete_post.php:

```

<?php
session_start();
require 'db.php';

if (!isset($_SESSION['user_id'])) {
    echo "Unauthorized";
    exit();
}

if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['post_id'])) {
    $post_id = $_POST['post_id'];
    $user_id = $_SESSION['user_id'];

    $stmt = $conn->prepare("DELETE FROM posts WHERE id = ? AND user_id = ?");
    $stmt->bind_param("ii", $post_id, $user_id);
    $stmt->execute();

    if ($stmt->affected_rows > 0) {
        echo "Post deleted successfully.";
    }
}

```

```

} else {
    echo "Failed to delete post.";
}

$stmt->close();
$conn->close();
}

?>

```

Fetch_posts.php:

```

<?php
session_start();
require 'db.php';

$query = "SELECT posts.id, posts.content, posts.user_id, posts.created_at,
users.username
        FROM posts
        JOIN users ON posts.user_id = users.id
        ORDER BY posts.created_at DESC";

$result = $conn->query($query);

$posts = [];

while ($row = $result->fetch_assoc()) {
    $posts[] = $row;
}

echo json_encode($posts);
?>

```

Fetch_rating.php

```

<?php
require 'db.php';

header('Content-Type: application/json');

if (!isset($_GET['org_id'])) {
    echo json_encode(['error' => 'Organization ID is required']);
}

```

```

    exit;
}

$org_id = intval($_GET['org_id']);

// Get average rating and total rating count
$stmt = $conn->prepare("
    SELECT
        AVG(user_rating) AS avg_rating,
        COUNT(*) AS total_ratings
    FROM ratings
    WHERE org_id = ?
");

$stmt->bind_param("i", $org_id);
$stmt->execute();
$result = $stmt->get_result();

if ($row = $result->fetch_assoc()) {
    $avg_rating = round($row['avg_rating'], 1); // e.g., 4.3
    $total_ratings = $row['total_ratings'];

    echo json_encode([
        'avg_rating' => $avg_rating ? $avg_rating : 0,
        'total_ratings' => $total_ratings
    ]);
} else {
    echo json_encode([
        'avg_rating' => 0,
        'total_ratings' => 0
    ]);
}
?>
```

fetch_only_user_posts.php

```

<?php
session_start();
require 'db.php';

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

```

echo json_encode([]);
exit();
}

$user_id = $_SESSION['user_id'];

$query = "SELECT posts.id, posts.content, posts.created_at, users.username
          FROM posts
          JOIN users ON posts.user_id = users.id
          WHERE posts.user_id = ?
          ORDER BY posts.created_at DESC";

$stmt = $conn->prepare($query);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();

$posts = [];

while ($row = $result->fetch_assoc()) {
    $posts[] = $row;
}

echo json_encode($posts);
?>

```

```

get_profile.php
<?php
session_start();
require 'db.php';

if (!isset($_SESSION['user_id'])) {
    echo json_encode(["success" => false, "message" => "Not logged in"]);
    exit;
}

$user_id = $_SESSION['user_id'];

$stmt = $conn->prepare("SELECT * FROM users WHERE id = ?");

```

```

$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();

if ($result->num_rows === 1) {
    $user = $result->fetch_assoc();
    echo json_encode(["success" => true, "user" => $user]);
} else {
    echo json_encode(["success" => false, "message" => "User not found"]);
}

$stmt->close();
$conn->close();

```

get_user_data.php:

```

<?php
session_start();
require 'db.php';

$user_id = $_SESSION['user_id'] ?? null;

if (!$user_id) {
    echo json_encode(["error" => "User not logged in"]);
    exit;
}

$stmt = $conn->prepare("SELECT profile_picture FROM users WHERE id = ?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$user = $result->fetch_assoc();

echo json_encode($user);

```

check_session.php:

```

<?php
session_start();
header('Content-Type: application/json');

```

```

if (isset($_SESSION['user_id'])) {
    echo json_encode(["loggedIn" => true, "username" =>
$_SESSION['username']]);
} else {
    echo json_encode(["loggedIn" => false]);
}
?>

```

submit_rating.php:

```

<?php
session_start();
include 'db.php';

$data = json_decode(file_get_contents("php://input"), true);
$org_id = $data['org_id'];
$rating = $data['rating'];
$rater_id = $_SESSION['user_id']; // current logged-in user

// Check if the user already rated this org
$check = $conn->prepare("SELECT * FROM ratings WHERE org_id = ? AND rater_id
= ?");
$check->bind_param("ii", $org_id, $rater_id);
$check->execute();
$result = $check->get_result();

if ($result->num_rows > 0) {
    // Update existing rating
    $existing = $result->fetch_assoc();
    $old_rating = $existing['user_rating'];

    $update = $conn->prepare("UPDATE ratings SET user_rating = ? WHERE org_id
= ? AND rater_id = ?");
    $update->bind_param("iii", $rating, $org_id, $rater_id);
    $update->execute();

    // Recalculate total rating
    $recalc = $conn->prepare("SELECT SUM(user_rating) AS total, COUNT(*) AS
count FROM ratings WHERE org_id = ?");

```

```

$recalc->bind_param("i", $org_id);
$recalc->execute();
$sumData = $recalc->get_result()->fetch_assoc();

$new_avg = $sumData['total'] / $sumData['count'];

} else {
    // Insert new rating
    $insert = $conn->prepare("INSERT INTO ratings (org_id, rater_id,
user_rating) VALUES (?, ?, ?)");
    $insert->bind_param("iii", $org_id, $rater_id, $rating);
    $insert->execute();

    // Update total and count
    $recalc = $conn->prepare("SELECT SUM(user_rating) AS total, COUNT(*) AS
count FROM ratings WHERE org_id = ?");
    $recalc->bind_param("i", $org_id);
    $recalc->execute();
    $sumData = $recalc->get_result()->fetch_assoc();

    $new_avg = $sumData['total'] / $sumData['count'];
}

// Return result
echo json_encode([
    'success' => true,
    'avg_rating' => $new_avg,
    'rating_count' => $sumData['count']
]);
?>

```

Update_org_image.php:

```

<?php
session_start();
require 'db.php';

$user_id = $_SESSION['user_id'];

if (isset($_FILES['new_org_image']) && $_FILES['new_org_image']['error'] ===

```

```

0) {
    $upload_dir = "../uploads/";
    if (!file_exists($upload_dir)) mkdir($upload_dir);
    $filename = uniqid() . "_" . basename($_FILES["new_org_image"]["name"]);
    $target_file = $upload_dir . $filename;
    if (move_uploaded_file($_FILES["new_org_image"]["tmp_name"],
    $target_file)) {
        $stmt = $conn->prepare("UPDATE users SET org_image = ? WHERE id =
        ?");
        $stmt->bind_param("si", $filename, $user_id);

        $stmt->execute();
        $stmt->close();
        header("Location: ../pages/profile.html");
        exit;
    }
}

echo "Failed to update Organization picture.";
?>

```

update profile picture.php:

```

<?php
session_start();
require 'db.php';

$user_id = $_SESSION['user_id'];

if (isset($_FILES['new_profile_picture']) &&
$_FILES['new_profile_picture']['error'] === 0) {
    $upload_dir = "../uploads/";
    if (!file_exists($upload_dir)) mkdir($upload_dir);
    $filename = uniqid() . "_" .
basename($_FILES["new_profile_picture"]["name"]);
    $target_file = $upload_dir . $filename;
    if (move_uploaded_file($_FILES["new_profile_picture"]["tmp_name"],
    $target_file)) {
        $stmt = $conn->prepare("UPDATE users SET profile_picture = ? WHERE id
= ?");
        $stmt->bind_param("si", $filename, $user_id);

```

```

$stmt->execute();
$stmt->close();
header("Location: ../pages/profile.html");
exit;
}

echo "Failed to update profile picture.";
?>

```

Update_profile.php:

```

<?php
session_start();
require 'db.php';

if (!isset($_SESSION['user_id'])) {
    echo "Unauthorized";
    exit;
}

$user_id = $_SESSION['user_id'];
$full_name = $_POST['full_name'];
$username = $_POST['username'];
$address = $_POST['address'];
$dob = $_POST['dob'];

$org_name = isset($_POST['org_name']) ? $_POST['org_name'] : null;
$org_location = isset($_POST['org_location']) ? $_POST['org_location'] : null;
$org_phone = isset($_POST['org_phone']) ? $_POST['org_phone'] : null;

$stmt = $conn->prepare("UPDATE users SET full_name=?, username=?, address=?,
dob=?, org_name=?, org_location=?, org_phone=? WHERE id=?");
$stmt->bind_param("sssssssi", $full_name, $username, $address, $dob,
$org_name, $org_location, $org_phone, $user_id);

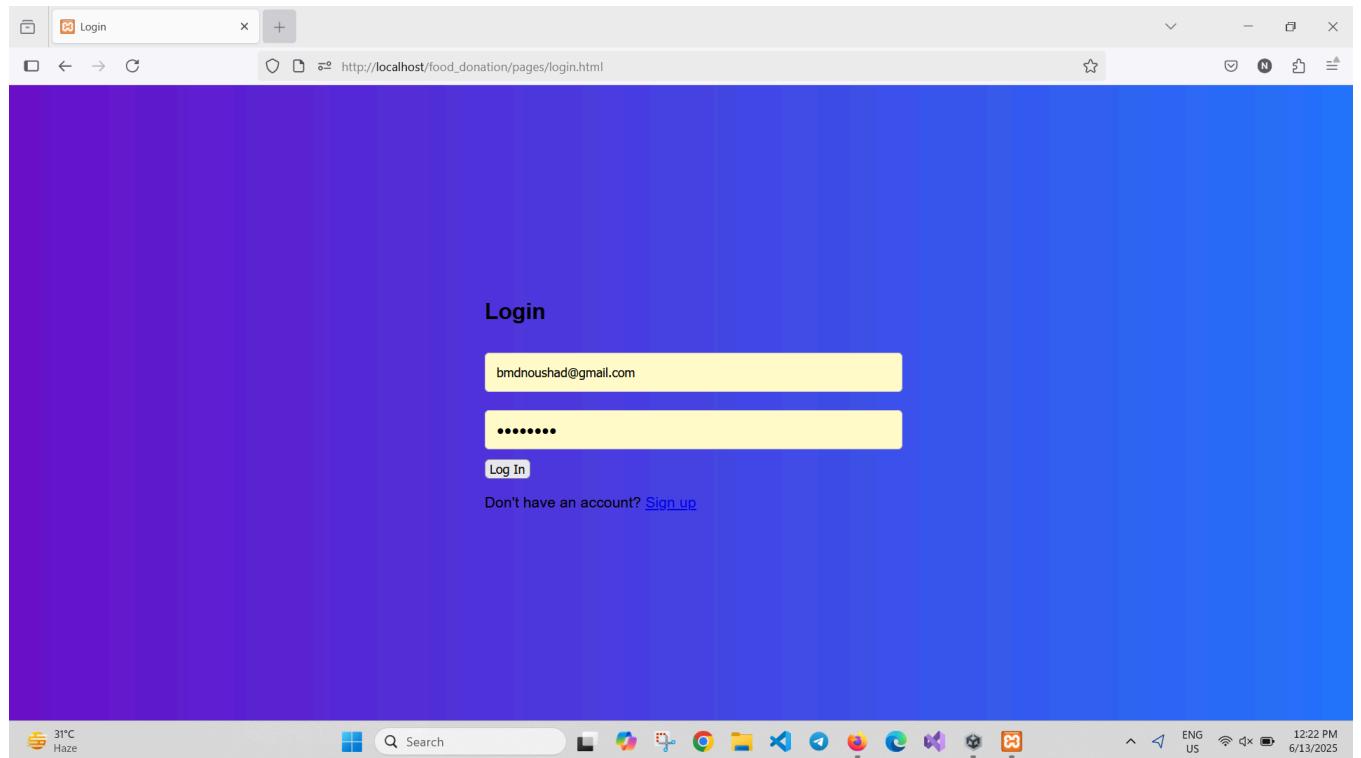
if ($stmt->execute()) {
    header("Location: ../pages/profile.html");
} else {

```

```
        echo "Update failed.";  
    }  
  
    $stmt->close();  
    $conn->close();
```

Page's And Their Details:

Login Page:



Login.HTML Code:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8" />  
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>  
    <title>Login</title>  
    <link rel="stylesheet" href="../../css/login.css" />
```

```

</head>
<body>

    <div class="signup-container">
        <h2>Login</h2>
        <form action="../php/login.php" method="POST">
            <input type="email" name="email" placeholder="Email" required />
            <input type="password" name="password" placeholder="Password" required
/>
            <button type="submit">Log In</button>
        </form>
        <p>Don't have an account? <a href="signup.html">Sign up</a></p>
    </div>
</body>
</html>

```

Login.CSS:

```

/* General Styles */
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    height: 100vh;
    display: flex;
    justify-content: center;
    overflow-y: scroll;
    overflow-x: hidden;
    align-items: center;
    background: linear-gradient(to right, #6a11cb, #2575fc);
    animation: fadeIn 1s ease-in-out;
}

/* Login Container */
.login-container {
    background: white;
    padding: 30px;
    border-radius: 10px;
    box-shadow: 0px 5px 15px rgba(0, 0, 0, 0.3);
}

```

```
width: 400px;
text-align: center;
animation: slideIn 1s ease-in-out;
}

/* Form Inputs */
input {
    width: 100%;
    padding: 12px;
    margin: 10px 0;
    border: 1px solid #ccc;
    border-radius: 5px;
    transition: all 0.3s ease;
}

input:focus {
    border-color: #6a11cb;
    transform: scale(1.05);
}

/* Login Button */
.btn {
    width: 100%;
    background: #6a11cb;
    color: white;
    padding: 12px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    font-size: 16px;
    transition: all 0.3s ease;
    animation: glowEffect 1.5s infinite alternate;
}

.btn:hover {
    background: #2575fc;
    transform: scale(1.05);
}

/* Switch to Signup */

```

```

.switch {
    margin-top: 15px;
}

.switch p {
    color: #666;
}

.signup-instead {
    display: inline-block;
    color: #6a11cb;
    font-weight: bold;
    text-decoration: none;
    transition: all 0.3s ease;
}

.signup-instead:hover {
    text-decoration: underline;
    transform: scale(1.1);
}

/* Animations */

@keyframes fadeIn {
    from { opacity: 0; transform: translateY(-20px); }
    to { opacity: 1; transform: translateY(0); }
}

@keyframes slideIn {
    from { opacity: 0; transform: translateY(-50px); }
    to { opacity: 1; transform: translateY(0); }
}

@keyframes glowEffect {
    from { box-shadow: 0px 0px 5px #6a11cb; }
    to { box-shadow: 0px 0px 15px #6a11cb; }
}

```

Login.PHP:

```

<?php
require 'db.php';

```

```
session_start();

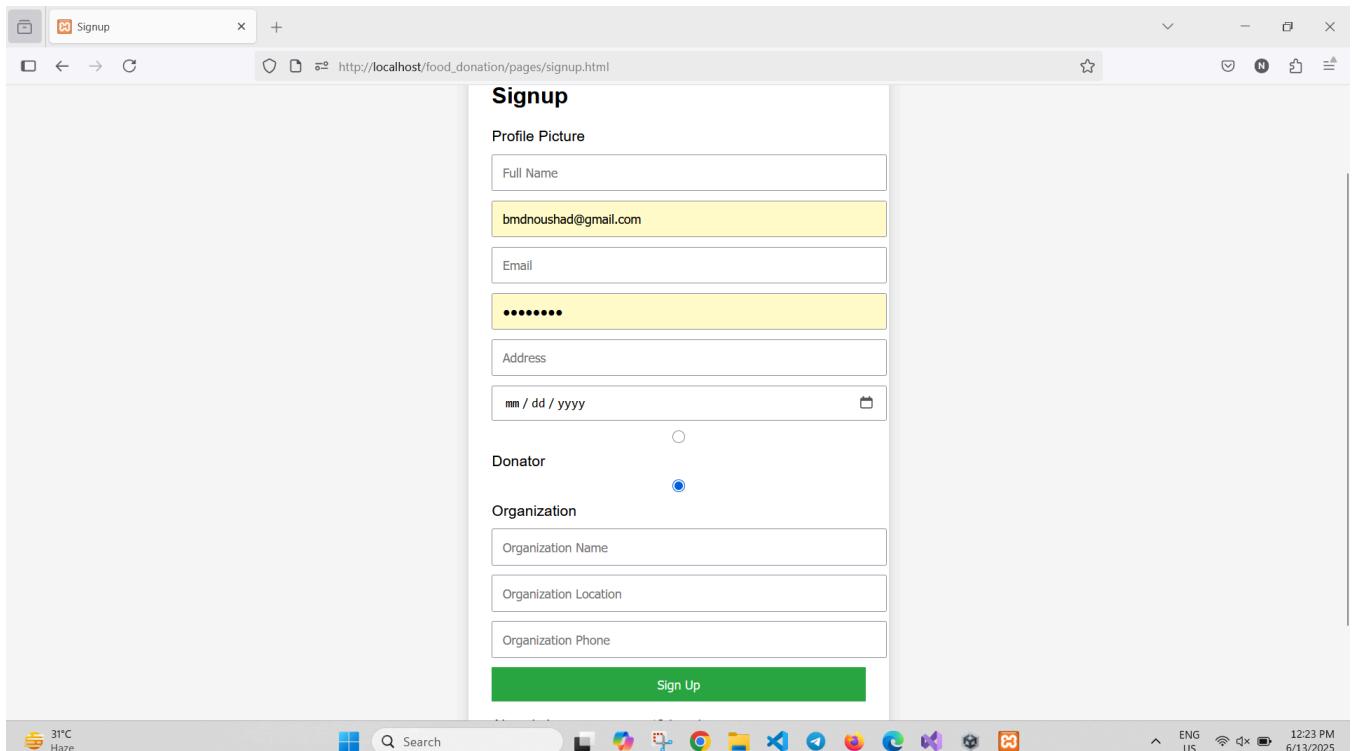
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = $_POST['email'];
    $password = $_POST['password'];

    $stmt = $conn->prepare("SELECT id, password FROM users WHERE email = ?");
    $stmt->bind_param("s", $email);
    $stmt->execute();
    $stmt->store_result();

    if ($stmt->num_rows == 1) {
        $stmt->bind_result($id, $hashed_password);
        $stmt->fetch();
        if (password_verify($password, $hashed_password)) {
            $_SESSION['user_id'] = $id;
            header("Location: ../pages/home.html"); // Redirect to home page
        } else {
            echo "Invalid password.";
        }
    } else {
        echo "No user found with that email.";
    }

    $stmt->close();
    $conn->close();
} else {
    echo "Invalid request.";
}
?>
```

Signup Page:



SignUp.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>Signup</title>
    <link rel="stylesheet" href="../css/signups.css" />
</head>
<body>
    <div class="signup-container">
```

```
<h2>Signup</h2>
<form id="signupForm" action="../php/signup.php" method="POST">
    <div>Profile Picture</div>
    <!-- <input type="file" name="profile_picture" accept="image/*" required /> -->
    <input type="text" name="full_name" placeholder="Full Name" required />
    <input type="text" name="username" placeholder="Username" required />
    <input type="email" name="email" placeholder="Email" required />
    <input type="password" name="password" placeholder="Password" required />
    <input type="text" name="address" placeholder="Address" required />
    <input type="date" name="dob" required />

    <label>
        <input type="radio" name="user_type" value="donator" checked onclick="toggleOrgFields(false)"> Donator
    </label>
    <label>
        <input type="radio" name="user_type" value="organization" onclick="toggleOrgFields(true)"> Organization
    </label>

    <div id="organizationFields" style="display: none;">
        <input type="text" name="org_name" placeholder="Organization Name" />
        <input type="text" name="org_location" placeholder="Organization Location" />
        <input type="text" name="org_phone" placeholder="Organization Phone" />
    </div>

    <button type="submit">Sign Up</button>
</form>
<p>Already have an account? <a href="login.html">Log in</a></p>
</div>

<script src="../js/signup.js"></script>
</body>
</html>
```

Signup.css

```
body {  
    font-family: Arial, sans-serif;  
    background-color: #f5f5f5;  
    overflow-y: scroll;  
    overflow-x: hidden;  
}  
  
.signup-container {  
    width: 400px;  
    margin: 60px auto;  
    background: white;  
    padding: 25px;  
    border-radius: 10px;  
    box-shadow: 0 0 10px #ccc;  
}  
  
input, button {  
    display: block;  
    width: 100%;  
    margin: 10px 0;  
    padding: 10px;  
}  
  
button {  
    background-color: #28a745;  
    color: white;  
    border: none;  
    cursor: pointer;  
}  
  
button:hover {  
    background-color: #218838;  
}
```

Signup.js:

```
function toggleOrgFields(show) {  
    document.getElementById('organizationFields').style.display = show ?  
    'block' : 'none';
```

```
}
```

Signup.php:

```
<?php  
require 'db.php';  
  
// user info  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $full_name = $_POST['full_name'];  
    $username = $_POST['username'];  
    $email = $_POST['email'];  
    $password = password_hash($_POST['password'], PASSWORD_BCRYPT);  
    $address = $_POST['address'];  
    $dob = $_POST['dob'];  
    $user_type = $_POST['user_type'];  
  
    $org_name=null;  
    $org_location=null;  
    $org_phone=null;  
    $org_image = null;  
  
    // Handle organization-specific data  
    if ($user_type == 'organization') {  
        $org_name = $_POST['org_name'];  
        $org_location = $_POST['org_location'];  
        $org_phone = $_POST['org_phone'];  
  
        if (isset($_FILES['org_image']) && $_FILES['org_image']['error'] ===  
0) {  
            $target_dir = "../uploads/";  
            $org_image = time() . "_" .  
basename($_FILES["org_image"]["name"]);  
            $target_file = $target_dir . $org_image;  
            move_uploaded_file($_FILES["org_image"]["tmp_name"],  
$target_file);  
        }  
    }  
    // profile picture info  
    $profile_picture = null;  
    if (isset($_FILES['profile_picture']) &&
```

```

$_FILES['profile_picture']['error'] === 0) {
    $upload_dir = "../uploads/";
    if (!file_exists($upload_dir)) mkdir($upload_dir);
    $filename = uniqid() . "_" .
basename($_FILES["profile_picture"]["name"]);
    $target_file = $upload_dir . $filename;
    if (move_uploaded_file($_FILES["profile_picture"]["tmp_name"],
$target_file)) {
        $profile_picture = $filename;
    }
}

// Check if email exists
$stmt = $conn->prepare("SELECT id FROM users WHERE email = ?");
$stmt->bind_param("s", $email);
$stmt->execute();
$stmt->store_result();

if ($stmt->num_rows > 0) {
    echo "Email already exists.";
    exit;
}

// Insert into users table
$stmt = $conn->prepare("INSERT INTO users (full_name, username, email,
password, address, dob, user_type, org_name, org_location, org_phone,
profile_picture,org_image)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");
$stmt->bind_param("ssssssssss", $full_name, $username, $email,
$password, $address, $dob, $user_type, $org_name, $org_location, $org_phone,
$profile_picture,$org_image);

if ($stmt->execute()) {
    $user_id = $stmt->insert_id;

    // Insert into organization table
    if ($user_type === 'organization') {
        $stmt2 = $conn->prepare("INSERT INTO organization (user_id,
org_name, org_location, org_phone, org_image, keyword) VALUES (?, ?, ?, ?, ?, ?,
?)");
        $keyword = strtolower($org_location); // Example keyword, can be
modified
    }
}

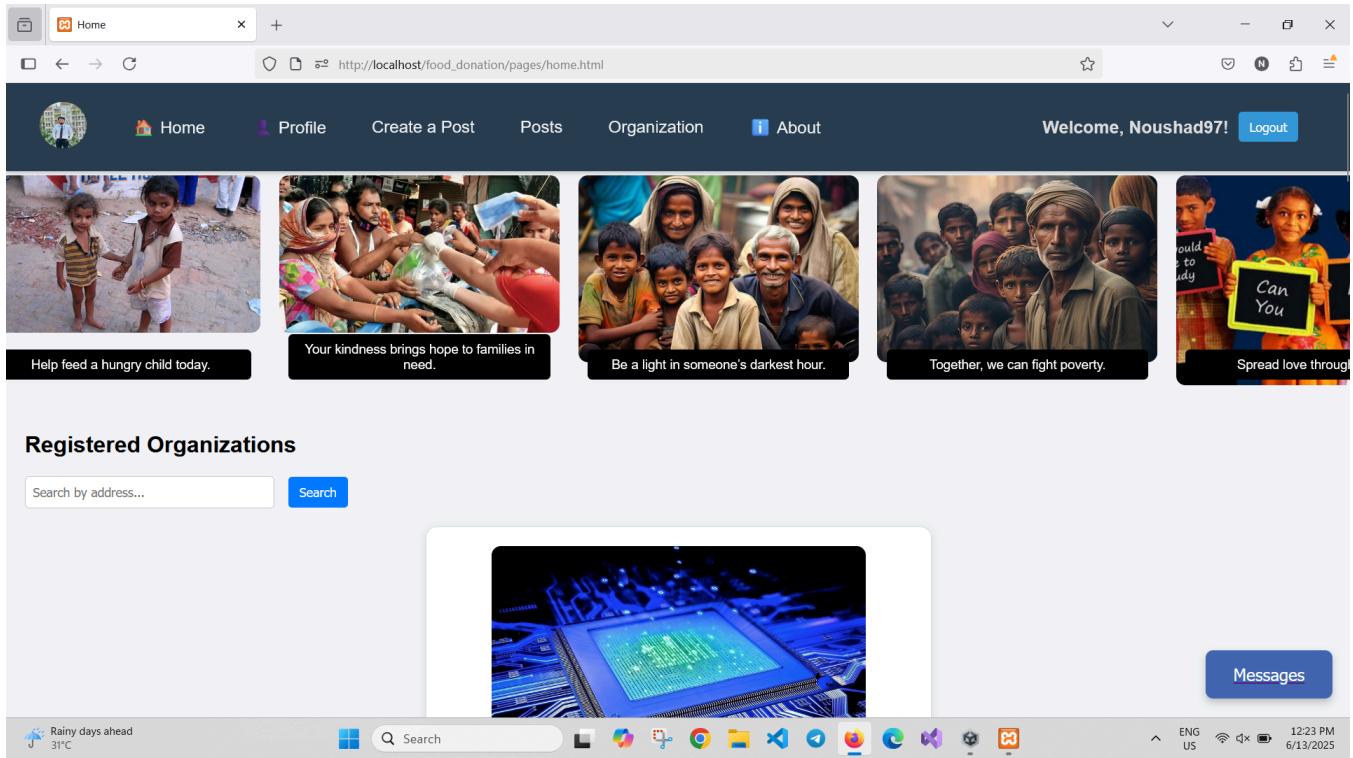
```

```
        $stmt2->bind_param("isssss", $user_id, $org_name, $org_location,
$org_phone, $org_image, $keyword);
        $stmt2->execute();
        $stmt2->close();
    }

    header("Location: ../pages/login.html");
} else {
    echo "Signup failed. Please try again.";
}

$stmt->close();
$conn->close();
} else {
    echo "Invalid request.";
}
?>
```

HomePage:



HomePage.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>Home</title>
    <link rel="stylesheet" href="../css/home.css" />
    <link rel="stylesheet" href="../css/header.css" />
    <link rel="stylesheet" href="../css/message.css" />
    <link rel="stylesheet" href="../css/home_organization.css" />
<style>
    .red-dot {
        width: 10px;
        height: 10px;
        background: red;
        border-radius: 50%;
        display: inline-block;
        position: absolute;
        top: 5px;
        right: 5px;
    }

```

```
</style>

</head>
<body>

<section style="background:rgb(0, 153, 255)">
    <!-- Top Navigation Bar -->

    <div class="top-navbar">

        <a href="profile.html">
            <img id="profilePic" src="" alt="Profile" style="width: 50px; height: 50px; border-radius: 50%;"/>
        </a>
        <a href="home.html"> Home</a>
        <a href="profile.html"> Profile</a>
        <a href="create-post.html">Create a Post</a>
        <a href="posts.html">Posts</a>
        <a href="organization.html">Organization</a>
        <a href="about.html"> About</a>

        <h3 style="margin-left: 200px; color:#ddd">Welcome, <span id="username">Guest</span>!</h3>

        <div class="nav-right" id="navButtons">
            <!-- Buttons will be injected here -->
        </div>
        <!-- <button onclick="toggleChatPopup()" id="messageIcon">
            <img alt="Message icon" style="vertical-align: middle;"/> <span id="messageDot" class="red-dot" style="display: none;"/></span>
        </button> -->
        <!-- Add more buttons here -->
    </div>
</section>
```

```
<!-- image slider panel -->
<div class="background-overlay">
  <div class="slider-container">
    <div class="slider-track">
      <div class="slide">
        
        <div class="slide-message">Every small donation makes a big difference!</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Help feed a hungry child today.</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Your kindness brings hope to families in need.</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Be a light in someone's darkest hour.</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Together, we can fight poverty.</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Spread love through giving.</div>
      </div>
      <div class="slide">
        
        <div class="slide-message">Your support changes lives.</div>
      </div>
    <div class="slide">
```

```

<div class="slide-message">Join us in helping the poor.</div>
</div>
</div>
</div>

<!-- organization section -->
<main style="padding: 20px;">
    <h2>Registered Organizations</h2>

    <div class="search-box">
        <input type="text" id="searchInput" placeholder="Search by address...">
        <button onclick="searchOrganizations()">Search</button>
    </div>

    <div id="orgList"></div>

    <div class="pagination">
        <button onclick="loadPreviousPage()">Previous Page</button>
        <button onclick="loadNextPage()">Next Page</button>
    </div>
</main>

<!-- home newsfeed -->
<div class="newsfeed-container">
    <h2>Newsfeed</h2>
    <div id="postsContainer" class="post"></div>
</div>

<!-- Popup Toggle Button -->

<a href="../chatting/chatroom.php" style="position: fixed; bottom: 20px; right: 20px;">
    <button style="
        padding: 15px 30px;
        font-size: 18px;
        border: none;
        background-color: transparent;
        color: inherit;
        text-decoration: none;
    ">
        Chat
    </button>
</a>

```

```
border-radius: 8px;
background-color: #4267B2;
color: white;
cursor: pointer;
box-shadow: 0 4px 8px rgba(0,0,0,0.2);
display: flex;
align-items: center;
">


Messages



</button>
</a>

<script src="../js/home.js"></script>
<script src="../js/message.js"></script>
<script src="../js/organization.js"></script>
<script>
fetch('../php/get_user_data.php')
.then(res => res.json())
.then(data => {
  const pic = data.profile_picture ?
`../uploads/${data.profile_picture}` : '../assets/default.jpg';
  document.getElementById('profilePic').src = pic;
});

//slider image
const track = document.querySelector('.slider-track');
const slides = Array.from(track.children);
slides.forEach(slide => {
  const clone = slide.cloneNode(true);
  track.appendChild(clone);
});
```

```
        });

</script>

<script>

// Global listener for sending messages directly to organization boxes
document.addEventListener('click', function (e) {
    if (e.target.classList.contains('sendMessageBtn')) {
        const orgId = e.target.getAttribute('data-org-id');
        const input =
document.querySelector(`.messageInput[data-org-id="${orgId}"]`);
        const message = input.value.trim();

        if (message === "") {
            alert("Please enter a message.");
            return;
        }

        fetch('../php/send_message.php', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify({
                receiver_id: orgId,
                message: message
            })
        })
        .then(res => res.json())
        .then(data => {
            if (data.success) {
                alert("Message sent!");
                input.value = "";
            } else {
                alert("Failed to send message.");
            }
        })
        .catch(err => {
            console.error(err);
            alert("Error sending message.");
        });
    }
})
```

```
});  
</script>
```

```
</body>
```

```
</html>
```

HomePage.css:

```
body {  
    margin: 0;  
    overflow-y: scroll;  
    overflow-x: hidden;  
    background-color: #f0f2f5;  
  
    padding-top: 70px; /* Adjust to match navbar height */  
}  
  
.navbar {  
    display: flex;  
    justify-content: space-between;  
    padding: 15px 30px;  
    background-color: #2c3e50;  
    color: white;  
}  
  
.nav-right button {  
    margin-left: 10px;  
    padding: 8px 12px;  
    background-color: #3498db;  
    border: none;  
    color: white;  
    cursor: pointer;  
    border-radius: 4px;  
}  
  
.nav-right button:hover {  
    background-color: #2980b9;  
}
```

```
.container {
    padding: 30px;
}

/* Hamburger Icon */

#menuToggle {
    position: fixed;
    top: 15px;
    left: 15px;
    font-size: 30px;
    cursor: pointer;
    z-index: 1002;
    color: white;
    background-color: #2c3e50;
    padding: 8px 12px;
    border-radius: 4px;
}

/* Overlay background */

.overlay {
    position: fixed;
    display: none;
    top: 0;
    left: 0;
    height: 100%;
    width: 100%;
    background: rgba(0,0,0,0.5);
    z-index: 1000;
}

/* Top Navbar Styling */

.top-navbar {
    display: flex;
    justify-content: left;
    align-items: center;
    background-color: #2c3e50; /* DARK BLUE BACKGROUND */
    padding: 12px 24px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.2);
    position: fixed;
    top: 0;
    left: 0;
    right: 0;
```

```
z-index: 100;
}

/* Navbar Links */
.top-navbar a {
    color: #ecf0f1;
    text-decoration: none;
    margin-right: 25px;
    font-size: 18px;
    transition: all 0.3s ease;
    padding: 8px 12px;
    border-radius: 4px;
}

.top-navbar a:hover {
    background-color: #34495e;
    transform: translateY(-2px);
}

/* Push content below the navbar */
/* body {
    margin: 0;
    padding-top: 70px;
    font-family: Arial, sans-serif;

    padding: 0;
    height: 70%;
} */
/* body {
    font-family: Arial, sans-serif;
    background: #f2f2f2;
    padding: 20px;
} */
/*
.newsfeed-container {
    max-width: 600px;
    margin: auto;
    background: white;
    padding: 20px;
    border-radius: 8px;
}
```

```
}

/* Common Post Styling */

.post {
    background: #ffffff;
    border-radius: 10px;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.08);
    margin: 20px auto;
    padding: 20px;
    max-width: 600px;
    transition: transform 0.2s ease, box-shadow 0.3s ease;
    border-left: 5px solid #4caf50;
}

.post:hover {
    transform: translateY(-5px);
    box-shadow: 0 6px 20px rgba(0, 0, 0, 0.1);
}

.post h4 {
    margin: 0 0 10px;
    color: #333;
    font-size: 1.2em;
    font-weight: 600;
}

.post p {
    font-size: 1em;
    color: #444;
    margin: 0 0 10px;
    line-height: 1.5;
}

.post small {
    display: block;
    text-align: right;
    color: #888;
    font-size: 0.85em;
}
```

```
.post form {
    text-align: right;
    margin-top: 10px;
}

.post form button {
    background: #e53935;
    color: white;
    border: none;
    padding: 6px 12px;
    border-radius: 4px;
    cursor: pointer;
    transition: background 0.2s ease;
}

.post form button:hover {
    background: #c62828;
}

/* Popup container */
.chat-popup {
    display: none;
    position: fixed;
    bottom: 60px;
    right: 20px;
    width: 400px;
    height: 500px;
    background: #fff;
    border-radius: 12px;
    box-shadow: 0 8px 20px rgba(0,0,0,0.3);
    overflow: hidden;
    display: flex;
    flex-direction: column;
    animation: slideUpFade 0.4s ease forwards;
    font-family: Arial, sans-serif;
}

@keyframes slideUpFade {
    from { opacity: 0; transform: translateY(20px); }
    to { opacity: 1; transform: translateY(0); }
}
```

```
/* Header */

.chat-header {
    background: #007BFF;
    color: #fff;
    padding: 12px;
    font-size: 18px;
    font-weight: bold;
    display: flex;
    justify-content: space-between;
    align-items: center;
}

.chat-close-btn {
    background: transparent;
    border: none;
    color: white;
    font-size: 18px;
    cursor: pointer;
}

/* Body */

.chat-body {
    display: flex;
    flex: 1;
}

/* User list */

.chat-user-list {
    width: 40%;
    border-right: 1px solid #ddd;
    overflow-y: auto;
    background: #f9f9f9;
    padding: 8px;
}

.chat-user {
    background: #fff;
    padding: 8px 10px;
    margin-bottom: 8px;
}
```

```
border-radius: 8px;
cursor: pointer;
transition: background 0.2s, transform 0.1s;
text-align: center;
}

.chat-user:hover {
background: #e8f0fe;
transform: scale(1.03);
}

/* Main chat area */

.chat-main {
flex: 1;
display: flex;
flex-direction: column;
}

.chat-title {
padding: 8px;
font-weight: bold;
background: #f1f1f1;
border-bottom: 1px solid #ddd;
}

.chat-area {
flex: 1;
padding: 10px;
overflow-y: auto;
background: #fafafa;
display: flex;
flex-direction: column;
}

.chat-placeholder {
text-align: center;
color: gray;
}

.chat-message {
```

```
max-width: 70%;  
margin: 6px 0;  
padding: 8px 12px;  
border-radius: 16px;  
word-wrap: break-word;  
animation: messageFadeIn 0.2s ease-out;  
}  
  
@keyframes messageFadeIn {  
from { opacity: 0; transform: translateY(10px); }  
to { opacity: 1; transform: translateY(0); }  
}  
  
.chat-message.sender {  
background: #007BFF;  
color: #fff;  
align-self: flex-end;  
}  
  
.chat-message.receiver {  
background: #e0e0e0;  
color: #333;  
align-self: flex-start;  
}  
  
/* Input area */  
.chat-input-area {  
display: flex;  
border-top: 1px solid #ddd;  
padding: 8px;  
background: #fff;  
}  
  
.chat-input-area input {  
flex: 1;  
padding: 8px 12px;  
border: 1px solid #ccc;  
border-radius: 20px;  
outline: none;  
}
```

```
.chat-input-area button {  
    margin-left: 8px;  
    padding: 8px 16px;  
    border: none;  
    background: #007BFF;  
    color: #fff;  
    border-radius: 20px;  
    cursor: pointer;  
    transition: background 0.2s;  
}  
  
.chat-input-area button:hover {  
    background: #0056b3;  
}  
  
/* image slider */  
body, html {  
    margin: 0;  
    padding: 0;  
    height: 70%;  
    font-family: Arial, sans-serif;  
}  
  
.background-overlay {  
    background-size: cover;  
    background-position: center;  
    width: 100%;  
    height: 100%;  
    position: relative;  
}  
  
.slider-container {  
    width: 100%;  
    overflow: hidden;  
    position: absolute;  
    bottom: 5px; /* adjust vertical position */  
}  
  
.slider-track {
```

```
display: flex;
width: calc(300px * 20); /* 8 slides * 2 for looping */
animation: slide 100s linear infinite;
}

.slide {
width: 300px;
margin-right: 20px;
position: relative;
}

.slide img {
width: 100%;
height: auto;
border-radius: 10px;
}

.slide-message {
position: absolute;
bottom: 10px;
left: 10px;
right: 10px;
padding: 8px;
background: rgb(0, 0, 0);
color: #fff;
font-size: 14px;
border-radius: 5px;
text-align: center;
}

@keyframes slide {
0% {
transform: translateX(0);
}
100% {
transform: translateX(-50%);
}
}
```

[Homepage.js](#):

```
// Fetch user info from PHP session
fetch('../php/get_user.php')
  .then(response => response.json())
  .then(data => {
    const nav = document.getElementById("navButtons");
    const userSpan = document.getElementById("username");

    if (data.loggedIn) {
      userSpan.textContent = data.username;

      const logoutBtn = document.createElement("button");
      logoutBtn.textContent = "Logout";
      logoutBtn.onclick = () => {
        window.location.href = "../php/logout.php";
      };
      nav.appendChild(logoutBtn);
    } else {
      userSpan.textContent = "Guest";

      const loginBtn = document.createElement("button");
      loginBtn.textContent = "Login";
      loginBtn.onclick = () => {
        window.location.href = "login.html";
      };

      const signupBtn = document.createElement("button");
      signupBtn.textContent = "Signup";
      signupBtn.onclick = () => {
        window.location.href = "signup.html";
      };

      nav.appendChild(loginBtn);
      nav.appendChild(signupBtn);
    }
  });
}

fetch("../php/fetch_posts.php")
  .then(res => res.json())
```

```

.then(posts => {
  const container = document.getElementById("postsContainer");
  posts.forEach(post => {
    const div = document.createElement("div");
    console.log('Checking ');
    div.className = "post";
    div.innerHTML =
      `

#### ${post.username}


      <p>${post.content}</p>
      <small>${new Date(post.created_at).toLocaleString()}</small>
    `;
    container.appendChild(div);
  });
}

function checkUnread() {
  fetch("php/check_unread.php")
    .then(res => res.json())
    .then(data => {
      document.getElementById("messageDot").style.display = data.unread
        > 0 ? "inline-block" : "none";
    });
}
checkUnread();
setInterval(checkUnread, 10000);

```

Profile Page:

Your Profile

Profile Picture

Browse... No file selected.

Update Profile Picture

Full Name
Md Nouhad Bhuiyan

Username
Noushad97

Email (cannot be changed)
bmdnoushad@gmail.com

Address
Dhaka, Narshingdi, Shibpur

Date of Birth
12 / 31 / 2004

Your Posts

Noushad97
hey everyone
5/27/2025, 6:12:02 PM

Delete

Profile.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <title>Profile</title>
    <link rel="stylesheet" href="../../css/profile.css" />
```

```
<style>

/* Common Post Styling */

.post {
    background: #ffffff;
    border-radius: 10px;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.08);
    margin: 20px auto;
    padding: 20px;
    max-width: 600px;
    transition: transform 0.2s ease, box-shadow 0.3s ease;
    border-left: 5px solid #4caf50;
}

.post:hover {
    transform: translateY(-5px);
    box-shadow: 0 6px 20px rgba(0, 0, 0, 0.1);
}

.post h4 {
    margin: 0 0 10px;
    color: #333;
    font-size: 1.2em;
    font-weight: 600;
}

.post p {
    font-size: 1em;
    color: #444;
    margin: 0 0 10px;
    line-height: 1.5;
}

.post small {
    display: block;
    text-align: right;
    color: #888;
    font-size: 0.85em;
}

.post form {
```

```
        text-align: right;
        margin-top: 10px;
    }

.post form button {
    background: #e53935;
    color: white;
    border: none;
    padding: 6px 12px;
    border-radius: 4px;
    cursor: pointer;
    transition: background 0.2s ease;
}

.post form button:hover {
    background: #c62828;
}

</style>
<link rel="stylesheet" href="../../css/header.css" />
</head>
<body>
    <header>
        <section style="background:rgb(0, 153, 255)">
            <!-- Top Navigation Bar -->
            <div class="top-navbar">
                <a href="profile.html">
                    <img id="profilePic" src="" alt="Profile" style="width: 50px; height: 50px; border-radius: 50%;"/>
                </a>
                <a href="home.html"> Home</a>
                <a href="create-post.html">Create a Post</a>
                <a href="posts.html">Posts</a>
                <a href="organization.html">Organization</a>
                <a href="about.html"> About us</a>
            <!-- Add more buttons here -->
        </div>
    </section>
</header>
<div class="profile-container">
```

```
<h2>Your Profile</h2>

<!-- profile picture update section -->
<dir>Profile Picture</dir>
<form action="../php/update_profile_picture.php" method="POST"
enctype="multipart/form-data">
    <input type="file" name="new_profile_picture" accept="image/*" required
/>
    <button type="submit">Update Profile Picture</button>
</form>
<!-- profile update section -->
<form id="profileForm" method="POST" action="../php/update_profile.php">
<div>
    <label for="full_name">Full Name</label>
    <input type="text" name="full_name" id="full_name" required />
</div>
<div>
    <label for="username">Username</label>
    <input type="text" name="username" id="username" required />
</div>
<div>
    <label for="email">Email (cannot be changed)</label>
    <input type="email" name="email" id="email" readonly />
</div>
<div>
    <label for="address">Address</label>
    <input type="text" name="address" id="address" required />
</div>
<div>
    <label for="dob">Date of Birth</label>
    <input type="date" name="dob" id="dob" required />
</div>

<button type="submit">Update Profile</button>

<div>
    <label for="user_type">User Type</label>
    <input type="text" name="user_type" id="user_type" readonly />
</div>
```

```

        </div>

        <div>
            <div id="organizationFields" style="display: none;">
                <label for="org_name">Organization Name</label>
                <!-- <input type="text" name="org_name" id="org_name" /> -->
                <input type="text" name="org_name" id="org_name" readonly />
            </div>
            <div>
                <label for="org_location">Organization Location</label>
                <!-- <input type="text" name="org_location" id="org_location" /> -->
                <input type="text" name="org_location" id="org_location" readonly />
            </div>
            <div>
                <!-- organization update section -->
                <label for="org_phone">Organization Phone</label>
                <!-- <input type="text" name="org_phone" id="org_phone" /> -->
                <input type="text" name="org_phone" id="org_phone" readonly />
            </div>
        </div>
    <!-- end of profile details and organization details -->
    </form>
    <form action="../php/update_org_image.php" method="POST"
enctype="multipart/form-data">
        <input type="file" name="new_org_image" accept="image/*" required />
        <button type="submit">Update Organization Picture</button>
    </form>
</div>

<script>
fetch("../php/get_profile.php")
    .then(res => res.json())
    .then(data => {
        if (data.success) {
            const user = data.user;
            document.getElementById("full_name").value = user.full_name;
            document.getElementById("username").value = user.username;
            document.getElementById("email").value = user.email;
            document.getElementById("address").value = user.address;
            document.getElementById("dob").value = user.dob;
        }
    })
</script>

```

```
document.getElementById("user_type").value = user.user_type;

if (user.user_type === 'organization') {
    document.getElementById("organizationFields").style.display =
"block";
    document.getElementById("org_name").value = user.org_name;
    document.getElementById("org_location").value =
user.org_location;
    document.getElementById("org_phone").value = user.org_phone;
    document.getElementById("org_image").value = user.org_image;
}
} else {
    alert("Error loading profile");
}
}) ;
</script>

<div class="profile-container">
<h2>Your Posts</h2>
<div id="userPosts" class="post"></div>
</div>

<script>
fetch("../php/fetch_user_posts.php")
.then(res => res.json())
.then(posts => {
    const container = document.getElementById("userPosts");
    posts.forEach(post => {
        const div = document.createElement("div");
        div.className = "post";
        div.innerHTML = `
            <h4>${post.username}</h4>
            <p>${post.content}</p>
            <small>${new Date(post.created_at).toLocaleString()}</small>
            <form method="POST" action="../php/delete_post.php">
                <input type="hidden" name="post_id" value="${post.id}" />
                <button type="submit">Delete</button>
            </form>
        `;
        container.appendChild(div);
    });
});
</script>
```

```

        }) ;
    }) ;
</script>
<script>
    fetch('.../php/get_user_data.php')
        .then(res => res.json())
        .then(data => {
            const pic = data.profile_picture ?
`.../uploads/${data.profile_picture}` : '../assets/default.jpg';
            document.getElementById('profilePic').src = pic;
        }) ;
</script>
</body>
</html>

```

Profile.php:

```

<?php
session_start();
include("db.php");

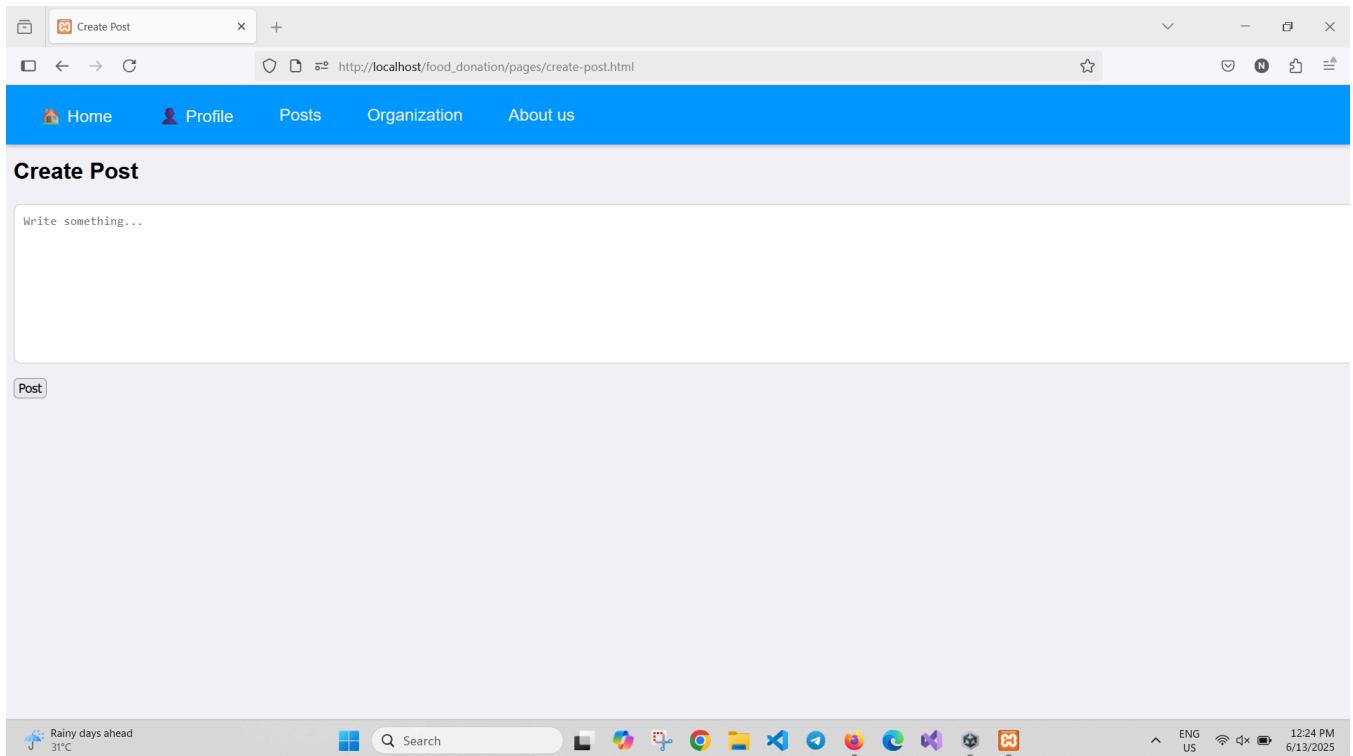
if (!isset($_SESSION['user_id'])) {
    header("Location: .../auth/login.html");
    exit();
}

$id = $_SESSION['user_id'];
$query = "SELECT * FROM users WHERE id = $id";
$result = mysqli_query($conn, $query);
$user = mysqli_fetch_assoc($result);

// Display profile
echo "<p><strong>Full Name:</strong> {$user['fullname']}</p>";
echo "<p><strong>Username:</strong> {$user['username']}</p>";
echo "<p><strong>Email:</strong> {$user['gmail']}</p>";
echo "<p><strong>Address:</strong> {$user['address']}</p>";
echo "<p><strong>Date of Birth:</strong> {$user['dob']}</p>";
?>

```

Create A Post Page:



CreatePost.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <title>Create Post</title>
    <link rel="stylesheet" href="../css/create_post.css" />
    <link rel="stylesheet" href="../css/header.css" />
</head>
<body>

<header>
    <div style="background-color: rgb(0, 153, 255)" class="top-navbar">
        <a href="home.html">🏠 Home</a>
        <a href="profile.html">👤 Profile</a>
        <a href="posts.html">Posts</a>
        <a href="organization.html">Organization</a>
        <a href="about.html">About us</a>
        <!-- Add more buttons here -->
    </div>
</header>
<div class="post-container">
    <h2>Create Post</h2>
```

```

<form id="postForm" action="../php/create_post.php" method="POST">
    <textarea name="content" placeholder="Write something..." required></textarea>
    <button type="submit">Post</button>
</form>
</div>
<script src="../js/create_post.js"></script>
</body>
</html>

```

CreatePost.php:

```

<?php
session_start();
require 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: ../pages/login.html");
    exit();
}

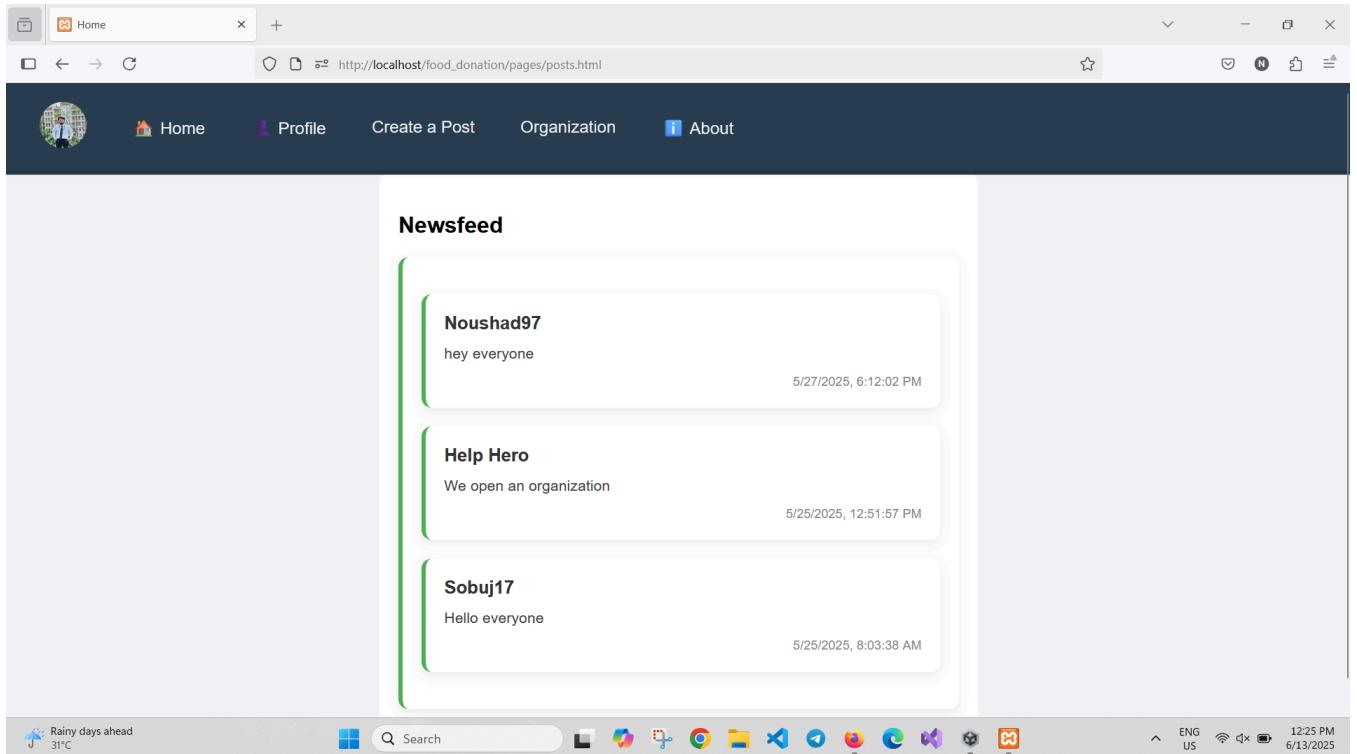
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $user_id = $_SESSION['user_id'];
    $content = trim($_POST['content']);

    if (!empty($content)) {
        $stmt = $conn->prepare("INSERT INTO posts (user_id, content) VALUES (?, ?)");
        $stmt->bind_param("is", $user_id, $content);
        $stmt->execute();
        $stmt->close();
    }

    header("Location: ../pages/home.html");
    exit();
}
?>

```

See Post Page:



posts.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>Home</title>
    <link rel="stylesheet" href="../../css/home.css" />
    <link rel="stylesheet" href="../../css/header.css" />
    <style>
        /* Common Post Styling */
        .post {
            background: #ffffff;
            border-radius: 10px;
            box-shadow: 0 4px 10px rgba(0, 0, 0, 0.08);
            margin: 20px auto;
            padding: 20px;
            max-width: 600px;
            transition: transform 0.2s ease, box-shadow 0.3s ease;
            border-left: 5px solid #4caf50;
        }
        .post:hover {
            transform: translateY(-5px);
        }
    </style>

```

```
    box-shadow: 0 6px 20px rgba(0, 0, 0, 0.1);  
}  
  
.post h4 {  
    margin: 0 0 10px;  
    color: #333;  
    font-size: 1.2em;  
    font-weight: 600;  
}  
  
.post p {  
    font-size: 1em;  
    color: #444;  
    margin: 0 0 10px;  
    line-height: 1.5;  
}  
  
.post small {  
    display: block;  
    text-align: right;  
    color: #888;  
    font-size: 0.85em;  
}  
  
.post form {  
    text-align: right;  
    margin-top: 10px;  
}  
  
.post form button {  
    background: #e53935;  
    color: white;  
    border: none;  
    padding: 6px 12px;  
    border-radius: 4px;  
    cursor: pointer;  
    transition: background 0.2s ease;  
}  
  
.post form button:hover {
```

```
        background: #c62828;
    }

</style>
</head>
<body>

<section style="background:rgb(0, 153, 255)">
    <!-- Top Navigation Bar -->
<div class="top-navbar">
    <a href="profile.html">
        <img id="profilePic" src="" alt="Profile" style="width: 50px; height: 50px; border-radius: 50%;"/>
    </a>
    <a href="home.html"> Home</a>
    <a href="profile.html"> Profile</a>
    <a href="create-post.html">Create a Post</a>
    <a href="organization.html">Organization</a>
    <a href="about.html"> About</a>
    <!-- Add more buttons here -->
</div>
</section>

<div class="navbar">
    <div class="nav-left">
        <h2>Posts</h2>
    </div>
    <div class="nav-right" id="navButtons">
        <!-- Buttons will be injected here -->
    </div>
</div>

<!-- home newsfeed -->
<div class="newsfeed-container">
    <h2>Newsfeed</h2>
    <div id="postsContainer" class="post"></div>
```

```

</div>

<script src="../js/posts.js"></script>
<script>
    fetch('../php/get_user_data.php')
        .then(res => res.json())
        .then(data => {
            const pic = data.profile_picture ?
`../uploads/${data.profile_picture}` : '../assets/default.jpg';
            document.getElementById('profilePic').src = pic;
        });
</script>
</body>
</html>

```

Posts.php:

```

<?php
session_start();
require 'db.php';

$query = "SELECT posts.id, posts.content, posts.user_id, posts.created_at,
users.username
        FROM posts
        JOIN users ON posts.user_id = users.id
        ORDER BY posts.created_at DESC";

$result = $conn->query($query);

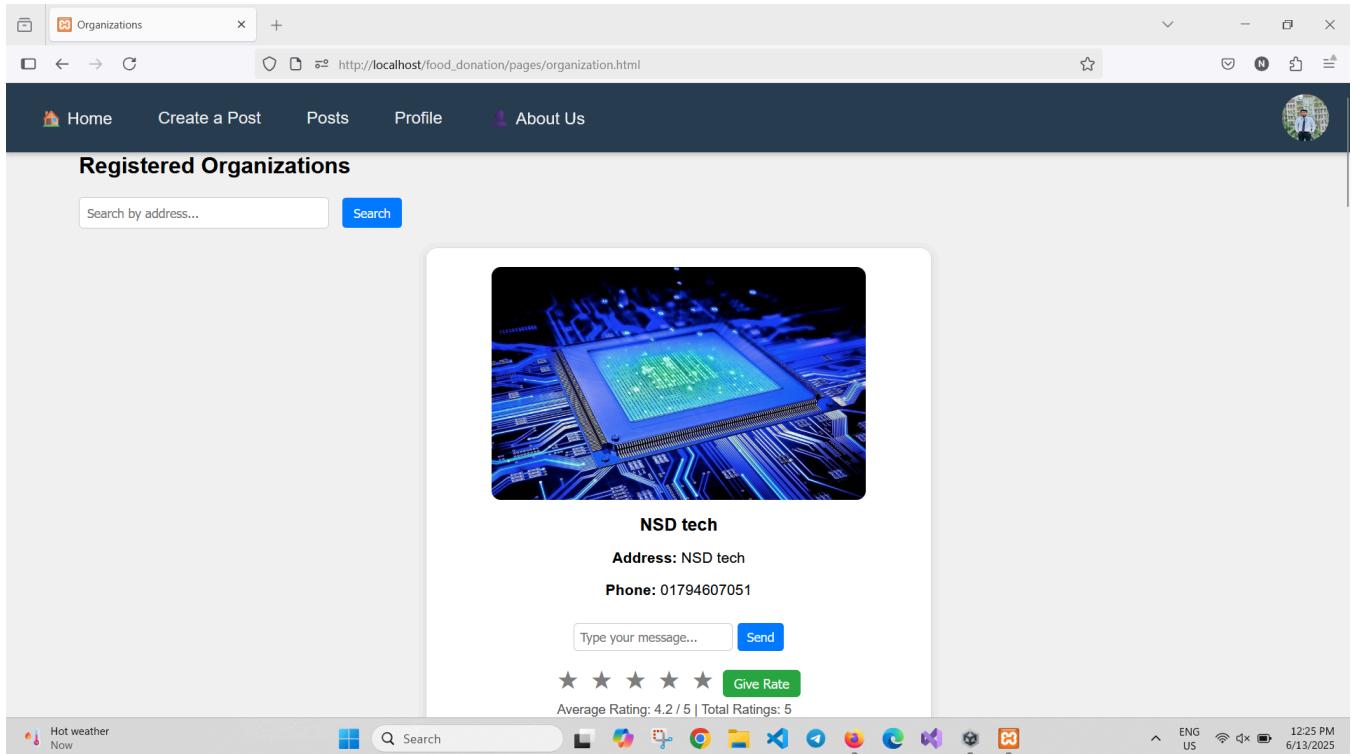
$posts = [];

while ($row = $result->fetch_assoc()) {
    $posts[] = $row;
}

echo json_encode($posts);
?>

```

All Organization Page:



Organization.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Organizations</title>

    <link rel="stylesheet" href="../css/organizations.css" />
    <link rel="stylesheet" href="../css/header.css" />
    <link rel="stylesheet" href="../css/message.css" />

    <style>
        .background {
            background-color: #007bff;
            color: white;
            padding: 10px;
        }
    </style>
</head>

<body>
<header>
    <section style="background:rgb(0, 153, 255)">
        <div class="top-navbar">
            <a href="home.html">🏠 Home</a>
            <a href="create-post.html">Create a Post</a>
            <a href="posts.html">Posts</a>
            <a href="profile.html">Profile</a>
        </div>
    </section>
</header>
```

```
<a href="about.html">👤 About Us</a>
    
</div>
</section>
</header>

<main style="padding: 20px;">
    <h2>Registered Organizations</h2>

    <div class="search-box">
        <input type="text" id="searchInput" placeholder="Search by address...">
        <button onclick="searchOrganizations()">Search</button>
    </div>

    <div id="orgList"></div>

    <div class="pagination">
        <button onclick="loadPreviousPage()">Previous Page</button>
        <button onclick="loadNextPage()">Next Page</button>
    </div>
</main>

<script src="../js/message.js"></script>
<script src="../js/organization.js"></script>
<script>

// Global listener for sending messages directly to organization boxes
document.addEventListener('click', function (e) {
    if (e.target.classList.contains('sendMessageBtn')) {
        const orgId = e.target.getAttribute('data-org-id');
        const input =
document.querySelector(`.messageInput[data-org-id="${orgId}"]`);
        const message = input.value.trim();

        if (message === "") {
            alert("Please enter a message.");
            return;
        }
    }
})
```

```

    fetch('../php/send_message.php', {
      method: 'POST',
      headers: { 'Content-Type': 'application/json' },
      body: JSON.stringify({
        receiver_id: orgId,
        message: message
      })
    })
    .then(res => res.json())
    .then(data => {
      if (data.success) {
        alert("Message sent!");
        input.value = "";
      } else {
        alert("Failed to send message.");
      }
    })
    .catch(err => {
      console.error(err);
      alert("Error sending message.");
    });
  }
);
</script>
</body>
</html>

```

organization.php:

```

<?php
require 'db.php';

$page = isset($_GET['page']) ? intval($_GET['page']) : 1;
$search = isset($_GET['search']) ? "%" . $_GET['search'] . "%" : "%";
$limit = 10;
$offset = ($page - 1) * $limit;

// JOIN users table using organization.user_id = users.id
$stmt = $conn->prepare("
  SELECT
    organization.org_name,

```

```
        organization.org_location,
        organization.org_phone,
        organization.user_id,
        users.org_image
    FROM organization
    JOIN users ON organization.user_id = users.id
    WHERE organization.org_location LIKE ?
    LIMIT ? OFFSET ?

");

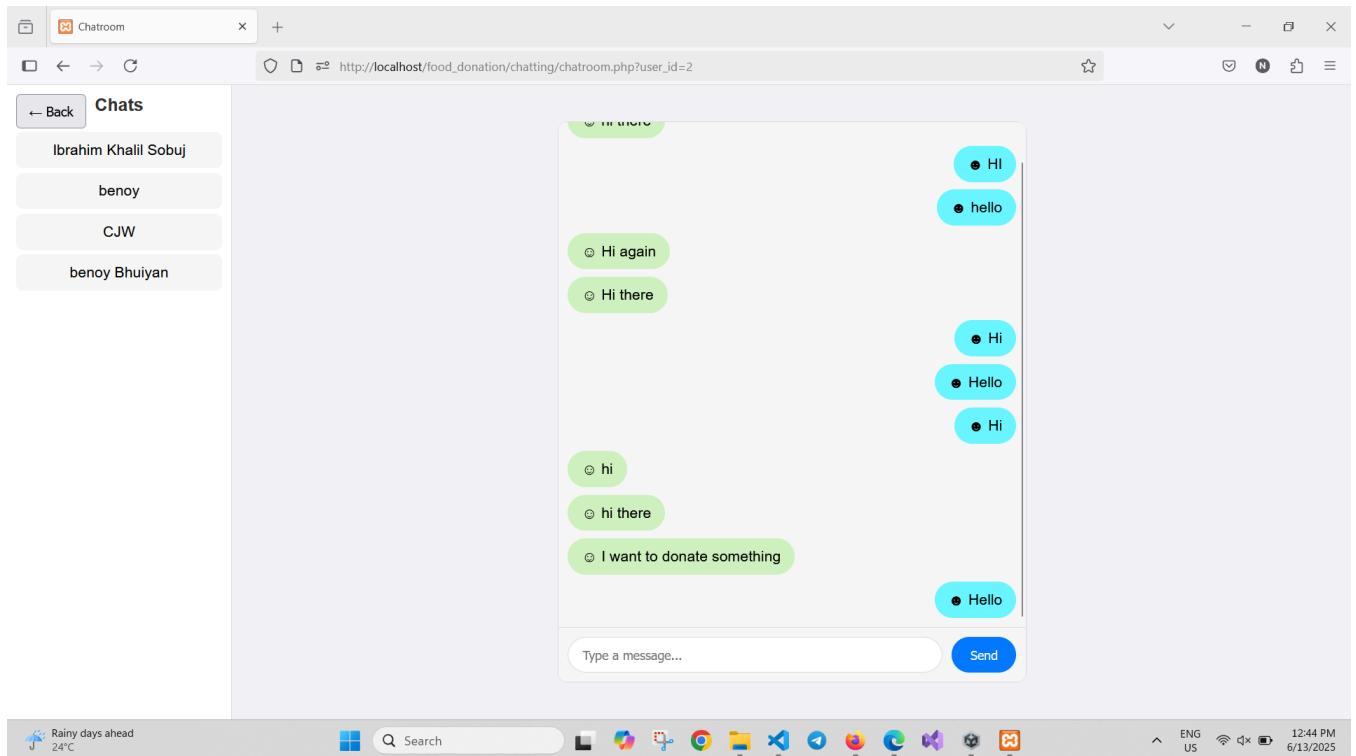
$stmt->bind_param("sii", $search, $limit, $offset);
$stmt->execute();

$result = $stmt->get_result();
$organizations = [];

while ($row = $result->fetch_assoc()) {
    $organizations[] = $row;
}

echo json_encode($organizations);
```

Chatting Page:



Chatroom.php:

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$current_user_id = $_SESSION['user_id'];
$selected_user_id = isset($_GET['user_id']) ? $_GET['user_id'] : null;

// Fetch conversation users (who messaged me or I messaged)
$users_result = $conn->query("
    SELECT DISTINCT u.id, u.full_name
    FROM users u
    JOIN messages m ON (u.id = m.sender_id OR u.id = m.receiver_id)
    WHERE (m.sender_id = $current_user_id OR m.receiver_id =
$current_user_id)
        AND u.id != $current_user_id
");
```

```
// Mark all messages from chat_user_id to current_user_id as read

?>

<!DOCTYPE html>
<html>
<head>
    <title>Chatroom</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f0f2f5;
            margin: 0;
            display: flex;
            height: 100vh;
        }
        .sidebar {
            width: 220px;
            background-color: #fff;
            border-right: 1px solid #ddd;
            overflow-y: auto;
            padding: 10px;
        }
        .sidebar h3 {
            margin-top: 0;
            text-align: center;
            color: #333;
        }
        .user-item {
            padding: 10px;
            margin: 5px 0;
            background-color: #f7f7f7;
            border-radius: 8px;
            cursor: pointer;
            transition: background 0.2s;
            text-align: center;
        }
        .user-item:hover {
            background-color: #e2e2e2;
        }
    </style>
</head>
<body>
    <div class="sidebar">
        <h3>Recent Users</h3>
        <ul>
            <li>User 1</li>
            <li>User 2</li>
            <li>User 3</li>
            <li>User 4</li>
            <li>User 5</li>
        </ul>
    </div>
    <div class="content">
        <h2>Chatroom</h2>
        <p>Welcome to the Chatroom!</p>
        <ul>
            <li>Message 1</li>
            <li>Message 2</li>
            <li>Message 3</li>
            <li>Message 4</li>
            <li>Message 5</li>
        </ul>
    </div>
</body>
</html>
```

```
.chat-container {
    flex: 1;
    display: flex;
    justify-content: center;
    align-items: center;
}

.chat-box {
    width: 500px;
    height: 600px;
    background-color: #fff;
    border: 1px solid #ddd;
    border-radius: 10px;
    display: flex;
    flex-direction: column;
    overflow: hidden;
}

.messages {
    overflow-y: scroll;
    background-color: #f9f9f9;
    display: flex;
    flex-direction: column;
    padding: 10px;
    row-gap: 8px;
}

.message-bubble {
    max-width: 70%;
    padding: 10px 15px;
    border-radius: 20px;
    word-wrap: break-word;
    opacity: 100;
    /* animation: fadeIn 0.3s forwards; */
}

.message-mine {
    align-self: flex-end;
    background-color: #6CF9FF;
}
```

```
.message-other {
    align-self: flex-start;
    background-color: #d0f0c0;
}

form {
    display: flex;
    padding: 10px;
    border-top: 1px solid #ddd;
    background-color: #f7f7f7;
}

form input[type="text"] {
    flex: 1;
    padding: 10px 15px;
    border: 1px solid #ddd;
    border-radius: 20px;
    margin-right: 10px;
    outline: none;
}

form button {
    padding: 10px 20px;
    border: none;
    background-color: #007bff;
    color: #fff;
    border-radius: 20px;
    cursor: pointer;
    transition: background 0.2s;
}

form button:hover {
    background-color: #0056b3;
}

@keyframes fadeIn {
    from { opacity: 0; transform: translateY(10px); }
    to { opacity: 1; transform: translateY(0); }
}

</style>

</head>
<body>
<div style="position: fixed; top: 10px; left: 10px;">
    <button onclick="goBack()" style="padding: 8px 12px; font-size: 14px; background-color: #f7f7f7; border: 1px solid #ddd; border-radius: 20px; outline: none; color: inherit; text-decoration: none; font-weight: bold; font-family: inherit; width: 100%; height: 100%;">Back</button>
</div>

```

```

cursor: pointer;">
    ← Back
</button>
</div>

<div class="sidebar">
    <h3>Chats</h3>
    <?php while ($u = $users_result->fetch_assoc()): ?>
        <a href="?user_id=<?php echo $u['id']; ?>" style="text-decoration: none; color: inherit; display: block;">
            <div class="user-item">
                <?php echo htmlspecialchars($u['full_name']); ?>
            </div>
        </a>

        <?php endwhile; ?>
</div>

<div class="chat-container">
    <div class="chat-box" >
        <?php if ($selected_user_id): ?>
            <div id="messages" class="messages" ></div>

            <form id="sendForm">
                <input type="hidden" name="receiver_id" value="<?php echo $selected_user_id; ?>">
                <input type="text" name="message" placeholder="Type a message..." required>
                <button type="submit">Send</button>
            </form>
        <?php else: ?>
            <div style="text-align: center; margin: auto; color: #777;">
                Select a user on the left to start chatting.
            </div>
        <?php endif; ?>
    </div>
</div>

<script>
function fetchMessages() {
    const xhr = new XMLHttpRequest();

```

```

xhr.onload = function() {
    const messagesDiv = document.getElementById('messages');

    // Save current scroll position
    const scrollTopBefore = messagesDiv.scrollTop;
    const scrollHeightBefore = messagesDiv.scrollHeight;

    messagesDiv.innerHTML = this.responseText;

    // Calculate how much the scrollHeight changed
    const scrollHeightAfter = messagesDiv.scrollHeight;
    const scrollDifference = scrollHeightAfter - scrollHeightBefore;

    // Keep the scroll position stable (no automatic jump)
    messagesDiv.scrollTop = scrollTopBefore + scrollDifference;
};

xhr.open("GET", "fetch_messages.php?user_id=<?php echo $selected_user_id; ?>", true);
xhr.send();
}

function goBack() {
    window.location.href = '../pages/home.html'; // change 'home.php' to your
actual homepage file
}

<?php if ($selected_user_id): ?>
setInterval(fetchMessages, 200);
fetchMessages();

document.getElementById('sendForm').addEventListener('submit', function(e) {
    e.preventDefault();
    const xhr = new XMLHttpRequest();
    xhr.onload = function() {
        fetchMessages();
        e.target.message.value = '';
    };
    xhr.open("POST", "send_message.php", true);
    xhr.setRequestHeader("Content-Type",
"application/x-www-form-urlencoded");
    const params = new URLSearchParams(new FormData(e.target)).toString();

```

```
    xhr.send(params);  
});  
<?php endif; ?>  
  
</script>  
</body>  
</html>
```