**Koneksi.php**

1. $conn: Ini adalah variabel yang akan digunakan untuk menyimpan koneksi ke database.

2. mysqli\_connect("localhost","root","","gallery"): Ini adalah pemanggilan fungsi mysqli\_connect() untuk melakukan koneksi ke database MySQL. Parameter-parameter yang digunakan adalah sebagai berikut:

3. "localhost": Merupakan nama atau alamat server database. Dalam hal ini, database berada pada server lokal.

4. "root": Merupakan username yang digunakan untuk mengakses database. Pada kebanyakan instalasi MySQL default, username ini adalah "root".

5. "": Merupakan password yang digunakan untuk mengakses database. Jika tidak ada password, parameter ini diisi dengan string kosong.

6. "gallery": Merupakan nama database yang akan diakses atau digunakan.

Dengan menggunakan fungsi mysqli\_connect(), script tersebut mencoba untuk membuat koneksi ke database MySQL dengan parameter yang diberikan. Jika koneksi berhasil, maka variabel $conn akan berisi objek koneksi yang akan digunakan untuk menjalankan query-query SQL. Jika koneksi gagal, maka fungsi ini akan mengembalikan false, dan biasanya akan memunculkan pesan error terkait koneksi.

**Login.php**

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Halaman Landing</title>

<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css">

</head>

<body>

<div class="container py-5">

<div class="row justify-content-center">

<div class="col-md-4">

<div class="card">

<div class="card-body bg-light">

<div class="text-center">

<h5>Halaman Login</h5>

</div>

<form action="proses\_login.php" method="post">

<label class="form-label">Username</label>

<input type="text" name="username" class="form-control" required>

<label class="form-label">Password</label>

<input type="password" name="password" class="form-control" required>

<div class="d-grid mt-2">

<button class="btn btn-primary" type="submit">Login</button>

</div>

</form>

<hr>

<p>Belum punya akun ? <a href="register.php">Silahkan Register dulu</a></p>

</div>

</div>

</div>

</div>

</div>

<footer class="d-flex justify-content-center border-top mt-3 bg-light fixed-bottom">

<p>Nesya Fadillah</p>

</footer>

<script src="assets/js/bootstrap.min.js"></script>

</body>

</html>

**Penjelasan:**

1. Deklarasi Struktur HTML: Script dimulai dengan deklarasi struktur HTML standar, termasuk deklarasi DOCTYPE, tag <html>, <head>, dan <body>. Ini adalah bagian dasar dari halaman web.
2. Meta Tag: Terdapat meta tag untuk pengaturan charset (pengaturan encoding karakter) dan viewport (pengaturan tampilan responsif untuk perangkat seluler).
3. Judul Halaman: Judul halaman ditetapkan dengan menggunakan tag <title>.
4. Pemanggilan CSS: Terdapat pemanggilan file CSS Bootstrap (bootstrap.min.css) yang digunakan untuk mengatur tata letak dan tampilan elemen-elemen dalam halaman web.
5. Elemen Konten: Konten halaman dimulai dengan tag <div> dengan kelas container untuk mengatur tata letak. Di dalamnya terdapat formulir login yang dibungkus dalam tag <div> dengan kelas row justify-content-center untuk mengatur posisi secara horizontal di tengah halaman.
6. Formulir Login: Formulir login terdiri dari input untuk username dan password. Setiap input diberi atribut name untuk identifikasi saat data dikirimkan. Input juga diberi atribut required untuk memastikan bahwa pengguna harus mengisi kedua kolom sebelum dapat mengirimkan formulir.
7. Button Login: Terdapat tombol "Login" yang akan mengirimkan data formulir ke halaman proses\_login.php saat ditekan.
8. Link Register: Di bagian bawah formulir, terdapat teks yang memberikan opsi untuk melakukan registrasi dengan mengarahkan pengguna ke halaman register.php jika belum memiliki akun.
9. Footer: Di bagian bawah halaman, terdapat footer yang berisi informasi atau keterangan tambahan. Dalam hal ini, terdapat nama "Nesya Fadillah".
10. Pemanggilan JavaScript: Pemanggilan file JavaScript Bootstrap (bootstrap.min.js) untuk menambahkan fungsi-fungsi interaktif pada halaman.

Proses Alur nya adalah sebagai berikut:

1. Pengguna membuka halaman tersebut di browser.
2. Pengguna mengisi formulir dengan username dan password.
3. Ketika tombol "Login" ditekan, data formulir dikirimkan ke proses\_login.php untuk diproses.
4. Jika login berhasil, pengguna akan diarahkan ke halaman yang sesuai (misalnya, halaman admin). Jika gagal, pesan kesalahan akan ditampilkan kembali di halaman login.

**Proses\_login.php**

<?php

include "koneksi.php";

session\_start();

$username = $\_POST['username'];

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

$sql = mysqli\_query($conn, "select \* from user where username='$username' and password='$password'");

$cek = mysqli\_num\_rows($sql);

if ($cek == 1) {

while ($data = mysqli\_fetch\_array($sql)) {

$\_SESSION['userid'] = $data['userid'];

$\_SESSION['namalengkap'] = $data['namalengkap'];

}

echo "<script>

alert('login berhasil');

location.href='index.php';

</script>";

} else {

echo "<script>

alert('login gagal');

location.href='login.php';

</script>";

}

**Penjelasan:**

1. Script tersebut merupakan script PHP yang digunakan untuk proses autentikasi login. Berikut adalah penjelasan dari setiap bagian script:
2. include "koneksi.php";: Ini adalah perintah untuk menyertakan file koneksi.php yang kemungkinan berisi kode untuk menghubungkan ke database.
3. session\_start();: Fungsi ini memulai atau melanjutkan sesi yang sudah ada dalam PHP. Ini diperlukan karena script menggunakan variabel-variabel sesi untuk menyimpan informasi pengguna yang berhasil login.

Pengambilan Data Input:

**$username = $\_POST['username'];**

**$password = $\_POST['password'];**

Ini adalah pengambilan nilai username dan password yang dikirimkan melalui metode HTTP POST dari form login.

1. Query Untuk memeriksa kredensial

**$sql = mysqli\_query($conn, "select \* from user where username='$username' and password='$password'");**

Query ini digunakan untuk memeriksa apakah ada baris dalam tabel pengguna (user) yang memiliki username dan password yang sesuai dengan yang dimasukkan oleh pengguna.

1. Pemeriksaan Hasil Query

**$cek = mysqli\_num\_rows($sql);**

Fungsi mysqli\_num\_rows() digunakan untuk menghitung jumlah baris hasil dari query yang dijalankan sebelumnya.

1. Handling Login

Jika terdapat satu baris yg cocok (login berhasil)

**if ($cek == 1) {**

**while ($data = mysqli\_fetch\_array($sql)) {**

**$\_SESSION['userid'] = $data['userid'];**

**$\_SESSION['namalengkap'] = $data['namalengkap'];**

**}**

**echo "<script>**

**alert('login berhasil');**

**location.href='index.php';**

**</script>";**

Data userid dan namalengkap dari pengguna yang berhasil login disimpan dalam variabel sesi ($\_SESSION).

Pengguna dialihkan ke halaman index.php setelah login berhasil dengan menampilkan pesan alert.

Jika tidak ada baris yg cocok maka (login gagal)

**} else {**

**echo "<script>**

**alert('login gagal');**

**location.href='login.php';**

**</script>";**

Pengguna dialihkan kembali ke halaman login (login.php) dengan menampilkan pesan alert bahwa login gagal.

**Register.php**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Halaman Landing</title>

<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css">

</head>

<body>

<div class="container py-5">

<div class="row justify-content-center">

<div class="col-md-4">

<div class="card">

<div class="card-body bg-light">

<div class="text-center">

<h5>Halaman Register</h5>

</div>

<form action="proses\_register.php" method="post">

<label class="form-label">Username</label>

<input type="text" name="username" class="form-control" required>

<label class="form-label">Password</label>

<input type="password" name="password" class="form-control" required>

<label class="form-label">Email</label>

<input type="email" name="email" class="form-control" required>

<label class="form-label">Nama Lengkap</label>

<input type="text" name="namalengkap" class="form-control" required>

<label class="form-label">Alamat</label>

<input type="text" name="alamat" class="form-control" required>

<div class="d-grid mt-2">

<button class="btn btn-primary" type="submit">Register</button>

</div>

</form>

<hr>

<p>Sudah punya akun ? <a href="login.php">Kembali</a></p>

</div>

</div>

</div>

</div>

</div>

<footer class="d-flex justify-content-center border-top mt-3 bg-light fixed-bottom">

<p>Nesya Fadilla</p>

</footer>

<script src="assets/js/bootstrap.min.js"></script>

</body>

</html>

**Penjelasan:**

1. Deklarasi Struktur HTML: Script dimulai dengan deklarasi struktur HTML standar, termasuk deklarasi DOCTYPE, tag <html>, <head>, dan <body>. Ini adalah bagian dasar dari halaman web.
2. Meta Tag: Terdapat meta tag untuk pengaturan charset (pengaturan encoding karakter) dan viewport (pengaturan tampilan responsif untuk perangkat seluler).
3. Judul Halaman: Judul halaman ditetapkan dengan menggunakan tag <title>.
4. Pemanggilan CSS: Terdapat pemanggilan file CSS Bootstrap (bootstrap.min.css) yang digunakan untuk mengatur tata letak dan tampilan elemen-elemen dalam halaman web.
5. Elemen Konten: Konten halaman dimulai dengan tag <div> dengan kelas container untuk mengatur tata letak. Di dalamnya terdapat formulir pendaftaran yang dibungkus dalam tag <div> dengan kelas row justify-content-center untuk mengatur posisi secara horizontal di tengah halaman.
6. Formulir Pendaftaran: Formulir pendaftaran terdiri dari beberapa input yang mengharuskan pengguna mengisi informasi seperti username, password, email, nama lengkap, dan alamat. Setiap input diberi atribut name untuk identifikasi saat data dikirimkan dan atribut required untuk memastikan bahwa pengguna harus mengisi semua kolom sebelum dapat mengirimkan formulir.
7. Button Register: Terdapat tombol "Register" yang akan mengirimkan data formulir ke halaman proses\_register.php saat ditekan.
8. Link Login: Di bagian bawah formulir, terdapat teks yang memberikan opsi untuk masuk ke halaman login jika pengguna sudah memiliki akun.
9. Footer: Di bagian bawah halaman, terdapat footer yang berisi informasi tambahan atau keterangan. Dalam hal ini, terdapat keterangan "Nesya Fadillah".
10. Pemanggilan JavaScript: Pemanggilan file JavaScript Bootstrap (bootstrap.min.js) untuk menambahkan fungsi-fungsi interaktif pada halaman.

Alur kerja umumnya adalah sebagai berikut:

1. Pengguna membuka halaman pendaftaran tersebut di browser.
2. Pengguna mengisi formulir dengan informasi yang diminta.
3. Ketika tombol "Register" ditekan, data formulir dikirimkan ke halaman proses\_register.php untuk diproses.
4. Script PHP di halaman proses\_register.php akan memproses data yang diterima dan menyimpannya ke dalam database.
5. Pengguna diberi umpan balik berupa pesan alert yang menyatakan apakah pendaftaran berhasil atau tidak. Jika berhasil, pengguna diarahkan ke halaman login untuk masuk. Jika tidak berhasil, pengguna tetap berada di halaman pendaftaran untuk mencoba lagi.

**proses\_register.php**

<?php

include "koneksi.php";

$username = $\_POST['username'];

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

$email = $\_POST['email'];

$namalengkap = $\_POST['namalengkap'];

$alamat = $\_POST['alamat'];

$sql = mysqli\_query($conn, "insert into user values('','$username','$password','$email','$namalengkap','$alamat')");

if ($sql) {

echo "<script>

alert('Pendaftaran akun berhasil');

location.href='login.php';

</script>";

} else {

echo "<script>

alert('Pendaftaran akun tidak berhasil');

location.href='register.php';

</script>";

}

**Penjelasan:**

1. include "koneksi.php";: Ini adalah perintah untuk menyertakan file koneksi.php yang kemungkinan berisi kode untuk menghubungkan ke database.
2. Pengambilan Data Input:

**$username = $\_POST['username'];**

**$password = $\_POST['password'];**

**$email = $\_POST['email'];**

**$namalengkap = $\_POST['namalengkap'];**

**$alamat = $\_POST['alamat'];**

Ini adalah pengambilan nilai-nilai dari form pendaftaran yang dikirimkan melalui metode HTTP POST.

1. Query untuk Menyimpan Data Pengguna Baru:

**$sql = mysqli\_query($conn, "insert into user values('','$username','$password','$email','$namalengkap','$alamat')");**

Pengguna diarahkan ke halaman login (login.php) setelah berhasil mendaftar dengan menampilkan pesan alert.

Jika query gagal dieksekusi (pendaftaran tidak berhasil):

**} else {**

**echo "<script>**

**alert('Pendaftaran akun tidak berhasil');**

**location.href='register.php';**

**</script>";**

Pengguna dialihkan kembali ke halaman pendaftaran (register.php) dengan menampilkan pesan alert bahwa pendaftaran akun tidak berhasil.

Alur kerja umumnya adalah sebagai berikut:

1. Pengguna mengisi formulir pendaftaran dengan informasi yang diminta.
2. Ketika tombol "Daftar" ditekan, data formulir dikirimkan ke halaman yang sama untuk diproses.
3. Script PHP ini memproses data yang diterima, menyimpannya ke dalam database menggunakan query INSERT.
4. Pengguna diberi umpan balik berupa pesan alert yang menyatakan apakah pendaftaran berhasil atau tidak, dan mereka diarahkan kembali ke halaman login jika pendaftaran berhasil atau tetap berada di halaman pendaftaran untuk mencoba lagi jika pendaftaran tidak berhasil.

**Index.php**

<?php

include "koneksi.php";

session\_start();

$userid = $\_SESSION['userid'];

if (!isset($\_SESSION['userid'])) {

header("location:login.php");

}

?>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Halaman Landing</title>

<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css" />

</head>

<body>

<nav class="navbar navbar-expand-lg bg-body-tertiary">

<div class="container">

<a class="navbar-brand" href="index.php">website gallery foto</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNavAltMarkup">

<div class="navbar-nav me-auto">

<?php

if (!isset($\_SESSION['userid'])) {

?>

<a href="register.php" class="btn btn-outline-primary m-1">Register</a>

<a href="login.php" class="btn btn-outline-primary m-1">Login</a>

<?php

} else {

?>

<li class="nav-item">

<a class="nav-link" aria-current="page" href="home.php">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="album.php">Album</a>

</li>

<li class="nav-item">

<a class="nav-link" href="foto.php">Foto</a>

</li>

<li class="nav-item">

<a class="nav-link" href="logout.php">Logout</a>

</li>

<?php

}

?>

</div>

</div>

</div>

</nav>

<div class="container mt-3">

<div class="row">

<?php

$sql = mysqli\_query($conn, "SELECT \* from foto INNER JOIN user ON foto.userid=user.userid INNER JOIN album ON foto.albumid=album.albumid");

while ($data = mysqli\_fetch\_array($sql)) {

?>

<div class="col-md-3">

<a type="button" data-bs-toggle="modal" data-bs-target="#komentar<?php echo $data['fotoid'] ?>">

<div class="card">

<img src="gambar/<?= $data['lokasifile'] ?>" class="card-img-top" title="<?php echo $data['judulfoto'] ?>" style="height: 12rem;" alt="">

<div class="card-footer text-center">

<?php

$fotoid = $data['fotoid'];

$ceksuka = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid' and userid='$userid'");

if (mysqli\_num\_rows($ceksuka) == 1) { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="batalsuka"><i class="fa fa-heart"></i></a>

<?php } else { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="suka"><i class="fa-regular fa-heart"></i></a>

<?php }

$like = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid'");

echo mysqli\_num\_rows($like) . ' Suka';

?>

<a href="#" type="button" data-bs-toggle="modal" data-bs-target="#komentar<?php echo $data['fotoid'] ?>"><i class="fa-regular fa-comment"></i></a>

<?php

$jmlkomentar = mysqli\_query($conn, "select \* from komentarfoto where fotoid='$fotoid'");

echo mysqli\_num\_rows($jmlkomentar) . ' Komentar';

?>

</div>

</div>

</a>

<!-- modal komentar -->

<div class="modal fade" id="komentar<?php echo $data['fotoid'] ?>" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">

<div class="modal-dialog modal-xl">

<div class="modal-content">

<div class="modal-body">

<div class="row">

<div class="col-md-8">

<img src="gambar/<?= $data['lokasifile'] ?>" class="card-img-top" title="<?php echo $data['judulfoto'] ?>">

</div>

<div class="col-md-4">

<div class="m-2">

<div class="overflow-auto">

<div class="sticky-top">

<strong><?php echo $data['judulfoto'] ?></strong><br>

<p>Pengunggah Gambar : <span class="badge bg-secondary"><?php echo $data['namalengkap'] ?></span></p>

<p>Tanggal Unggah : <span class="badge bg-secondary"><?php echo $data['tanggalunggah'] ?></span></p>

<p>Nama Album : <span class="badge bg-primary"><?php echo $data['namaalbum'] ?></span>

</div>

<hr>

<p align="left">

<strong>Deskripsi Gambar</strong><br>

<?php echo $data['deskripsifoto'] ?>

</p>

<hr>

<?php

$fotoid = $data['fotoid'];

$komentar = mysqli\_query($conn, "SELECT \* FROM komentarfoto INNER JOIN user ON komentarfoto.userid=user.userid WHERE komentarfoto.fotoid='$fotoid'");

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

?>

<p align="left">

<strong><?= $row['namalengkap'] ?></strong>

<?= $row['isikomentar'] ?>

</p>

<?php } ?>

<hr>

<div class="sticky-bottom">

<form action="tambah\_komentar.php" method="post">

<input type="text" name="fotoid" value="<?= $data['fotoid'] ?>" hidden>

<input type="text" name="isikomentar" class="form-control" placeholder="tambah komentar">

<div class="input-group">

<button type="submit" class="btn btn-primary">Kirim</button>

</div>

</form>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<?php

}

?>

</div>

</div>

<footer class="d-flex justify-content-center border-top mt-3 bg-light fixed-bottom">

<p>Nesya Fadillah</p>

</footer>

<script src="assets/js/bootstrap.min.js"></script>

</body>

</html>

**Penjelasan:**

1. Pemanggilan koneksi database:

**include "koneksi.php";**

Script ini menyertakan (include) file koneksi.php, yang kemungkinan berisi script untuk menghubungkan ke database MySQL.

1. Memulai Sesi

**session\_start();**

Fungsi session\_start() digunakan untuk memulai atau melanjutkan sesi yang sudah ada dalam PHP. Ini diperlukan karena script menggunakan session untuk menyimpan informasi pengguna yang sudah login.

1. Pengecekan status Login:

**$userid = $\_SESSION['userid'];**

**if (!isset($\_SESSION['userid'])) {**

**header("location:login.php");**

**}**

Script ini memeriksa apakah session userid sudah terisi atau belum.

Jika session userid tidak terisi (artinya pengguna belum login), maka script akan mengarahkan pengguna kembali ke halaman login menggunakan fungsi header("location:login.php");.

Bagian HTML:

1. Struktur Dasar HTML:

Bagian ini merupakan struktur dasar HTML untuk halaman web.

1. Navigasi (Navbar):

Terdapat navigasi (navbar) yang berisi beberapa menu, seperti "Home", "Album", "Foto", dan "Logout".

Menu "Register" dan "Login" akan muncul jika pengguna belum login. Namun, jika pengguna sudah login, menu tersebut akan diganti dengan menu "Home", "Album", "Foto", dan "Logout".

1. Konten Gambar:

Setiap gambar ditampilkan dalam sebuah card.

Ketika gambar diklik, sebuah modal akan muncul yang berisi informasi detail tentang gambar tersebut beserta komentar yang terkait.

1. Footer:

Bagian bawah halaman (footer) menampilkan informasi tambahan, dalam hal ini, nama penulis atau pembuat halaman.

Alur kerja

1. Pengecekan Status Login:

Saat pengguna mengakses halaman ini, script PHP akan melakukan pengecekan apakah pengguna sudah login atau belum.

Jika pengguna belum login (tidak ada session userid), maka pengguna akan diarahkan kembali ke halaman login untuk melakukan autentikasi.

1. Tampilan Halaman:

Jika pengguna sudah login, halaman akan menampilkan konten HTML seperti biasa, termasuk navigasi, gambar, dan footer.

Navigasi akan menyesuaikan tampilannya berdasarkan status login pengguna.

1. Interaksi dengan Gambar:

Setiap gambar memiliki tombol yang memicu munculnya modal saat diklik.

Modal tersebut akan menampilkan informasi detail tentang gambar, seperti judul, pengunggah, tanggal unggah, nama album, deskripsi gambar, serta jumlah suka dan komentar.

Pengguna juga dapat menambahkan komentar pada gambar melalui modal tersebut.

1. Footer:

Bagian bawah halaman menampilkan informasi tambahan, dalam hal ini, nama penulis atau pembuat halaman.

**Home.php**

<?php

include "koneksi.php";

session\_start();

$userid = $\_SESSION['userid'];

if (!isset($\_SESSION['userid'])) {

header("location:login.php");

}

?>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Halaman Landing</title>

<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.1/css/all.min.css" />

</head>

<body>

<nav class="navbar navbar-expand-lg bg-body-tertiary">

<div class="container">

<a class="navbar-brand" href="index.php">website gallery foto</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNavAltMarkup">

<div class="navbar-nav me-auto">

<?php

if (!isset($\_SESSION['userid'])) {

?>

<a href="register.php" class="btn btn-outline-primary m-1">Register</a>

<a href="login.php" class="btn btn-outline-primary m-1">Login</a>

<?php

} else {

?>

<li class="nav-item">

<a class="nav-link" aria-current="page" href="home.php">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="album.php">Album</a>

</li>

<li class="nav-item">

<a class="nav-link" href="foto.php">Foto</a>

</li>

<li class="nav-item">

<a class="nav-link" href="logout.php">Logout</a>

</li>

<?php

}

?>

</div>

</div>

</div>

</nav>

<div class="container mt-3">

Album:

<?php

$album = mysqli\_query($conn, "select \* from album where userid='$userid'");

while ($row = mysqli\_fetch\_array($album)) { ?>

<a href="home.php?albumid=<?php echo $row['albumid'] ?>" class="btn btn-outline-primary"><?php echo $row['namaalbum'] ?></a>

<?php } ?>

<div class="row">

<?php

if (isset($\_GET['albumid'])) {

$albumid = $\_GET['albumid'];

$query = mysqli\_query($conn, "select \* from foto where userid='$userid' and albumid='$albumid'");

while ($data = mysqli\_fetch\_array($query)) { ?>

<div class="col-md-3 mt-2">

<div class="card">

<img src="gambar/<?= $data['lokasifile'] ?>" class="card-img-top" title="" style="height: 12rem;" alt="">

<div class="card-footer text-center">

<?php

$fotoid = $data['fotoid'];

$ceksuka = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid' and userid='$userid'");

if (mysqli\_num\_rows($ceksuka) == 1) { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="batalsuka"><i class="fa fa-heart"></i></a>

<?php } else { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="suka"><i class="fa-regular fa-heart"></i></a>

<?php }

$like = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid'");

echo mysqli\_num\_rows($like) . ' Suka';

?>

<a href="komentar.php?fotoid=<?= $data['fotoid'] ?>"><i class="fa-regular fa-comment"></i></a>Komentar

</div>

</div>

<br>

</div>

<?php }

} else {

$sql = mysqli\_query($conn, "select \* from foto where userid='$userid'");

while ($data = mysqli\_fetch\_array($sql)) {

?>

<div class="col-md-3 mt-2">

<div class="card">

<img src="gambar/<?= $data['lokasifile'] ?>" class="card-img-top" title="" style="height: 12rem;" alt="">

<div class="card-footer text-center">

<?php

$fotoid = $data['fotoid'];

$ceksuka = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid' and userid='$userid'");

if (mysqli\_num\_rows($ceksuka) == 1) { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="batalsuka"><i class="fa fa-heart"></i></a>

<?php } else { ?>

<a href="like.php?fotoid=<?= $data['fotoid'] ?>" name="suka"><i class="fa-regular fa-heart"></i></a>

<?php }

$like = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid'");

echo mysqli\_num\_rows($like) . ' Suka';

?>

<a href="komentar.php?fotoid=<?= $data['fotoid'] ?>"><i class="fa-regular fa-comment"></i></a> Komentar

</div>

</div>

<br>

</div>

<?php

}

}

?>

</div>

</div>

<footer class="d-flex justify-content-center border-top mt-3 bg-light fixed-bottom">

<p>Nesya Fadillah</p>

</footer>

<script src="assets/js/bootstrap.min.js"></script>

</body>

</html>

**Penjelasan:**

1. Pemanggilan koneksi database:

**include "koneksi.php";**

Script ini menyertakan (include) file koneksi.php, yang kemungkinan berisi script untuk menghubungkan ke database MySQL.

1. Memulai Sesi

**session\_start();**

Fungsi session\_start() digunakan untuk memulai atau melanjutkan sesi yang sudah ada dalam PHP. Ini diperlukan karena script menggunakan session untuk menyimpan informasi pengguna yang sudah login.

1. Pengecekan status Login:

**$userid = $\_SESSION['userid'];**

**if (!isset($\_SESSION['userid'])) {**

**header("location:login.php");**

**}**

Script ini memeriksa apakah session userid sudah terisi atau belum.

Jika session userid tidak terisi (artinya pengguna belum login), maka script akan mengarahkan pengguna kembali ke halaman login menggunakan fungsi header("location:login.php");.

Bagian Html

1. Struktur Dasar HTML:

Bagian ini merupakan struktur dasar HTML untuk halaman web.

1. Navigasi (Navbar):

Terdapat navigasi (navbar) yang berisi beberapa menu, seperti "Home", "Album", "Foto", dan "Logout".

Menu "Register" dan "Login" akan muncul jika pengguna belum login. Namun, jika pengguna sudah login, menu tersebut akan diganti dengan menu "Home", "Album", "Foto", dan "Logout".

1. Konten Galeri Foto:

Halaman ini menampilkan galeri foto yang berisi gambar-gambar yang dimiliki oleh pengguna.

Jika terdapat parameter albumid dalam URL, halaman akan menampilkan gambar-gambar yang terkait dengan album tersebut. Jika tidak ada parameter tersebut, halaman akan menampilkan semua gambar yang dimiliki oleh pengguna.

Setiap gambar ditampilkan dalam sebuah card, dengan tombol suka (like) dan komentar.

Tombol suka akan memungkinkan pengguna untuk memberikan suka pada gambar tersebut.

Tombol komentar akan mengarahkan pengguna ke halaman untuk menambahkan komentar pada gambar tersebut.

1. Footer:

Bagian bawah halaman menampilkan informasi tambahan, dalam hal ini, nama penulis atau pembuat halaman.

**Like.php**

<?php

include "koneksi.php";

session\_start();

if (!isset($\_SESSION['userid'])) {

//Untuk bisa like harus login dulu

header("location:index.php");

} else {

$fotoid = $\_GET['fotoid'];

$userid = $\_SESSION['userid'];

//Cek apakah user sudah pernah like foto ini apa belum

$ceksuka = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid' and userid='$userid'");

if (mysqli\_num\_rows($ceksuka) == 1) {

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

$likeid = $row['likeid'];

mysqli\_query($conn, "delete from likefoto where likeid='$likeid'");

header("location:home.php");

}

} else {

$tanggallike = date("Y-m-d");

mysqli\_query($conn, "insert into likefoto values('','$fotoid','$userid','$tanggallike')");

header("location:home.php");

}

}

**Penjelasan:**

1. Memulai sesi **session\_start();**
2. Pengecekkan status login

**if (!isset($\_SESSION['userid'])) {**

**//Untuk bisa like harus login dulu**

**header("location:index.php");**

**}**

Script melakukan pengecekan apakah pengguna telah masuk atau belum dengan memeriksa keberadaan $\_SESSION['userid']. Jika tidak, maka pengguna akan diarahkan kembali ke halaman index atau halaman utama.

1. Proses like dan unlike

**$fotoid = $\_GET['fotoid'];**

**$userid = $\_SESSION['userid'];**

**$ceksuka = mysqli\_query($conn, "select \* from likefoto where fotoid='$fotoid' and userid='$userid'");**

Pengguna yang sudah masuk akan memperoleh akses ke proses selanjutnya. Data foto yang ingin di-'like' atau 'unlike' diperoleh dari parameter GET fotoid dan informasi pengguna dari $\_SESSION['userid'].

1. Cek Status Like

**if (mysqli\_num\_rows($ceksuka) == 1) {**

**// Jika sudah like, maka unlike**

**while ($row = mysqli\_fetch\_array($ceksuka)) {**

**$likeid = $row['likeid'];**

**mysqli\_query($conn, "delete from likefoto where likeid='$likeid'");**

**header("location:home.php");**

**}**

**} else {**

**// Jika belum like, maka like**

**$tanggallike = date("Y-m-d");**

**mysqli\_query($conn, "insert into likefoto values('','$fotoid','$userid','$tanggallike')");**

**header("location:home.php");**

**}**

Script ini melakukan pengecekan apakah pengguna sudah memberikan 'like' pada foto tersebut atau belum. Jika sudah, maka aksi 'unlike' dilakukan dengan menghapus data 'like' dari database. Jika belum, maka aksi 'like' dilakukan dengan memasukkan data 'like' baru ke dalam database.

1. Pengalihan halaman

Setelah proses like atau unlike selesai dilakukan, pengguna akan diarahkan kembali ke halaman home (home.php) menggunakan fungsi header().

**Komentar.php**

<?php

session\_start();

if(!isset($\_SESSION['userid'])){

header("location:login.php");

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Halaman Komentar</title>

</head>

<body>

<h1>Halaman Komentar</h1>

<p>Selamat datang <b><?=$\_SESSION['namalengkap']?></b></p>

<ul>

<li><a href="index.php">Home</a></li>

<li><a href="album.php">Album</a></li>

<li><a href="foto.php">Foto</a></li>

<li><a href="logout.php">Logout</a></li>

</ul>

<form action="tambah\_komentar.php" method="post">

<?php

include "koneksi.php";

$fotoid=$\_GET['fotoid'];

$sql=mysqli\_query($conn,"select \* from foto where fotoid='$fotoid'");

while($data=mysqli\_fetch\_array($sql)){

?>

<input type="text" name="fotoid" value="<?=$data['fotoid']?>" hidden>

<table>

<tr>

<td>Judul</td>

<td><input type="text" name="judulfoto" value="<?=$data['judulfoto']?>"></td>

</tr>

<tr>

<td>Deskripsi</td>

<td><input type="text" name="deskripsifoto" value="<?=$data['deskripsifoto']?>"></td>

</tr>

<tr>

<td>Foto</td>

<td><img src="gambar/<?=$data['lokasifile']?>" width="200px"></td>

</tr>

<tr>

<td>Komentar</td>

<td><input type="text" name="isikomentar"></td>

</tr>

<tr>

<td></td>

<td><input type="submit" value="Tambah"></td>

</tr>

</table>

<?php

}

?>

</form>

<table width="100%" border="1" cellpadding=5 cellspacing=0>

<tr>

<th>ID</th>

<th>Nama</th>

<th>Komentar</th>

<th>Tanggal</th>

</tr>

<?php

include "koneksi.php";

$userid=$\_SESSION['userid'];

$sql=mysqli\_query($conn,"select \* from komentarfoto,user where komentarfoto.userid=user.userid");

while($data=mysqli\_fetch\_array($sql)){

?>

<tr>

<td><?=$data['komentarid']?></td>

<td><?=$data['namalengkap']?></td>

<td><?=$data['isikomentar']?></td>

<td><?=$data['tanggalkomentar']?></td>

</tr>

<?php

}

?>

</table>

</body>

</html>

**Penjelasan:**

1. Pengecekkan Sesi dan direksi

**session\_start();**

**if(!isset($\_SESSION['userid'])){**

**header("location:login.php");**

**}**

Script ini memulai sesi dan memeriksa apakah ada $\_SESSION['userid'] yang tersedia. Jika tidak, maka pengguna akan diarahkan kembali ke halaman login.

1. Struktur Halaman Html

Setelah pengecekkan sesi, halaman html dimulai

1. Informasi pengguna:

**<p>Selamat datang <b><?=$\_SESSION['namalengkap']?></b></p>**

Menampilkan pesan selamat datang dengan nama lengkap pengguna yang sedang masuk.

1. Menu Navigasi

Terdapat menu navigasi yang berisi link untuk mengakses halaman home, album, foto, dan melakukan logout.

1. Form Tambah Komentar:

**<form action="tambah\_komentar.php" method="post">**

Form ini digunakan untuk menambahkan komentar pada foto yang dipilih.

1. Informasi Foto:

Script mengambil informasi mengenai foto yang akan diberi komentar.

Menampilkan judul foto, deskripsi foto, dan gambar foto.

1. Input Komentar:

Terdapat input untuk menambahkan komentar pada foto.

1. Tabel Komentar:

Menampilkan daftar komentar yang telah ada untuk foto yang dipilih.

Informasi yang ditampilkan antara lain ID komentar, nama pengguna yang memberi komentar, isi komentar, dan tanggal komentar.

1. Proses Penambahan Komentar

**$sql=mysqli\_query($conn,"select \* from komentarfoto,user where komentarfoto.userid=user.userid");**

Script ini mengambil data dari tabel komentarfoto dan user, kemudian melakukan join antara kedua tabel tersebut.

1. Iterasi dan tampilan komentar

Data komentar yang didapatkan dari hasil query ditampilkan dalam bentuk tabel.

Setiap baris tabel menunjukkan satu komentar beserta detailnya.

**Create (Tambah Data)**

Pada aplikasi tersebut terdapat Fitur Create atau tambah data memiliki script dan function yg sangat penting, script tersebut memiliki fungsi tersendiri disetiap prosesnya, diantaranya:

1. Pemanggilan koneksi database

**include "koneksi.php";**

**session\_start();**

1. Kemudian terdapat field table kemudian method post:

Script tersebut berguna untuk menerima data dari form

1. Tahap selanjutnya ada sebuah query insert into table query ini berfungsi untuk menambahkan data pada table yg akan dikirim ke database
2. Kemudian terdapat percabangan if else dan diakhiri dengan alert serta alamat untuk mengirimkan data yg telah terkirim tadi untuk ditampilkan pada halaman yg dituju.

**Update**

Pada aplikasi tersebut terdapat Fitur Update atau edit data memiliki script dan function yg sangat penting, script tersebut memiliki fungsi tersendiri disetiap prosesnya, diantaranya:

1. Pemanggilan koneksi database

**include "koneksi.php";**

**session\_start();**

1. Kemudian terdapat field table kemudian method post:

Script tersebut berguna untuk menerima data dari form

1. Kemudian masuk ke proses pengecekan dan pemanggilan data yang akan diupdate untuk ditampilkan pada halaman
2. Tahap selanjutnya ada sebuah query update set query ini berfungsi untuk mengubah data pada table yg telah ditambahkan sebelumnya kemudian akan dikirim ke database
3. Kemudian diakhiri dengan alert serta alamat untuk mengirimkan data yg telah diupdate tadi untuk ditampilkan pada halaman yg dituju.

**Delete**

Pada aplikasi tersebut terdapat Fitur delete atau hapus data yang memiliki script dan function yg sangat penting, script tersebut memiliki fungsi tersendiri disetiap prosesnya, diantaranya:

1. Pemanggilan koneksi database
2. Pemanggilan id data yg akan dihapus, hal ini berguna untuk memanggil dan memilih data dengan id yg akan dihapus
3. Kemudian query delete from, query ini berguna untuk menghapus data yg sudah dipilih tadi melalui id dan method get
4. Kemudian yg terakhir akan muncul alert jika berhasil dihapus.

**Logout**

Pada aplikasi tersebut terdapat menu logout yg didalamnya terdapat function session start dan session destroy, session start memiliki fungsi untuk memulai session, dan session destroy memiliki fungsi untuk mengakhiri session, kemudian logout diarahkan melalui header ke alamat login awal.