**Buatlah suatu sistem menggunakan CRUD Table sederhana dengan ketentuan berikut**

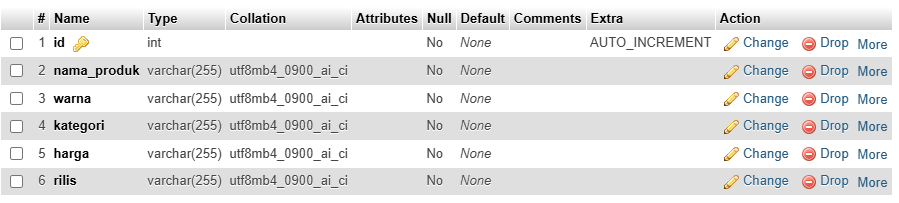
1. Di usahakan berpikir secara mandiri (backend php)
2. Buat suatu tabel database
3. Jelaskan secara rinci apa yang dibuat
4. Pembuatan sistem ini bebas yang penting nanti hasilnya menjadi CRUD Table
5. Agar bisa buat mandiri, di usahakan baca meteri sebelum-sebelumnya.
6. Agar lebih cepat, Styling/frontend dipersilahkan di ambil di mana saja.

**Teknologi yang digunakan:**

1. PHP
2. HTML & Tailwind CSS

**Tema** : Manajemen stok toko komputer.

**Struktur Tabel**



Terdapat id sebagai primary key dan auto increment, nama produk, warna, kategori, harga, rilis yang berupa (varchar) dengan length/panjang 255 dan juga nama path gambar.

**Insert (Front End)**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tambah Data</title>

    <script src="https://cdn.jsdelivr.net/npm/@tailwindcss/browser@4"></script>

</head>

<body class="flex items-center justify-center min-h-screen">

    <div class="bg-white shadow-lg rounded-2xl p-8 w-full max-w-md">

        <h2 class="text-2xl font-bold mb-6 text-gray-800 text-center">Tambah Data</h2>

        <form action="../backend/tambah.php" method="POST" class="space-y-5">

            <div>

                <label for="nama\_produk" class="block text-sm font-medium text-gray-700 mb-1">Nama Produk</label>

                <input type="text" id="nama\_produk" name="nama\_produk"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan nama produk" required>

            </div>

            <div>

                <label for="warna" class="block text-sm font-medium text-gray-700 mb-1">Warna</label>

                <input type="text" id="warna" name="warna"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan warna" required>

            </div>

            <div>

                <label for="kategori" class="block text-sm font-medium text-gray-700 mb-1">Kategori</label>

                <input type="text" id="kategori" name="kategori"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan kategori" required>

            </div>

            <div>

                <label for="harga" class="block text-sm font-medium text-gray-700 mb-1">Harga</label>

                <input type="text" id="harga" name="harga"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan harga" required>

            </div>

            <div>

                <label for="rilis" class="block text-sm font-medium text-gray-700 mb-1">Tahun Rilis</label>

                <input type="rilis" id="rilis" name="rilis"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan tahun rilis" required>

            </div>

            <button type="submit"

                class="w-full bg-blue-600 text-white py-2 px-4 rounded-lg hover:bg-blue-700 focus:ring-2 focus:ring-blue-400 transition">

                Submit

            </button>

        </form>

    </div>

</body>

</html>

Hal yang terpenting di sini, saya menggunakan name yang akan di isi dengan suatu value, name ini berupa post yang akan di bawa ke back end dengan action ../backend/tambah.php

**Insert (Backend)**

<?php

require\_once '../koneksi.php';

// Mengambil nilai form dari forntend

$nama\_produk = $\_POST['nama\_produk'];

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

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

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

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

// Menjalankan sql

$sqlquery = "INSERT INTO stok VALUES(NULL, '$nama\_produk', '$warna', '$kategori', '$harga', '$rilis')";

$eksekusi = mysqli\_query($koneksi, $sqlquery);

// Pengecekan

if (mysqli\_affected\_rows($koneksi) > 0) {

    echo "

    <script>

        alert('Data Berhasil Ditambahkan');

        window.location.href = '../frontend/index.php';

    </script>

    ";

} else {

    echo "Data gagal ditambahkan";

}

?>

Pada kode ini terdapat

1. Pengambil kode / require\_once di koneksi php. Ini agar dari sisi php akan tersambung ke sql.
2. Pengambilan data bedasarkan html **name** di fornt end, yang menggunakan metod post (tidak kelihatan di url), terdapat data berupa nama\_produk, warna, kategori, harga, rilis
3. Penyimpanan sqlsintaks yakni **INSERT INTO stok VALUES (…..),** yang berati : menambahkan data ke tabel stok dengan values dari hasil penangkapan data dari forntend (post)
4. Eksekusi query dengan mysqli\_query yang terhubung di koneksi.php yang menyambungkan kode php dengan Mysql
5. Pengecekan menggunakan mysqli\_affected\_rows bedasarkan perubahan di sql (dilihat dari $koneksi). **Jika 0 = tidak ada perubahan, jika -1 eror, jika >0 / 1 berhasil ditambahkan/ada perubahan**

**SELECT (FRONT END DI GABUNG BACKEND)**

<?php

require\_once '../koneksi.php';

$sqlsintaks = "SELECT \* FROM stok";

$eksekusi = mysqli\_query($koneksi, $sqlsintaks);

?>

Pada kode ini terdapat

1. Pengambil kode / require\_once di koneksi php. Ini agar dari sisi php akan tersambung ke sql.
2. Penyimpanan kode sintaks sql yang berisi “SELECT \* FORM stok” yang berarti memilih semua yang ada di tabel stok
3. Pengeksekusian sql sintaks bersama koneksi

            <tbody>

                <tr class="bg-white border-b dark:bg-gray-800 dark:border-gray-700 border-gray-200">

                    <?php  while( $datastok = mysqli\_fetch\_assoc($eksekusi)){ ?>

                    <th scope="row" class="px-6 py-4 font-medium text-gray-900 whitespace-nowrap dark:text-white">

                        <?php echo $datastok['id']; ?>

                    </th>

                    <th scope="row" class="px-6 py-4 font-medium text-gray-900 whitespace-nowrap dark:text-white">

                        <?php echo $datastok['nama\_produk']; ?>

                    </th>

                    <td class="px-6 py-4">

                        <?php echo $datastok['warna']; ?>

                    </td>

                    <td class="px-6 py-4">

                        <?php echo $datastok['kategori']; ?>

                    </td>

                    <td class="px-6 py-4">

                        <?php echo $datastok['harga']; ?>

                    </td>

                    <td class="px-6 py-4">

                        <?php echo $datastok['rilis']; ?>

                    </td>

                    <td>

                        <a href=" ../backend/hapus.php?id=<?php echo $datastok["id"]; ?>">Hapus |</a>

                        <a href="../backend/hapus.php"> Edit</a>

                    </td>

                    <?php }?>

                </tr>

            </tbody>

Pada Kode ini terdapat :

1. Pengambilan data menggunakan while. Dikarenakan data di mysql itu banyak, maka akan menggunakan while, yang di mana akan menjankan kode tersebut selama nilainya true (ada data). Yang dimana di dalam parameter while terdapat mysqli fetch assoc / berupa array asosiatif
2. Karena berupa asosiatif maka yang ditambpikan berupa string

**Delete (Front END)**

                        <a href=" ../backend/hapus.php?id=<?php echo $datastok["id"]; ?>">Hapus |</a>

Agar kita mengetahui primary key dari suatu kode dan di taruh di link, kita harus pakai ?id=$datastok[“id”] , yang nanti di tangkap di backend menggunakan metod get (karena melaui link)

**Delete (Backend END)**

require\_once '../koneksi.php';

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

$sqlsintaks = "DELETE FROM stok WHERE id = '$id'";

$eksekusi = mysqli\_query($koneksi, $sqlsintaks);

// Pengecekan eror

if (mysqli\_affected\_rows($koneksi) > 0) {

    echo "

    <script>

        alert('Data Berhasil dihapus');

        window.location.href = '../frontend/index.php';

    </script>

    ";

} else {

    echo "Data gagal dihapus". mysqli\_errno($koneksi);

}

Pada keode ini terdapat

1. Pengambil koneksi.php
2. Pengambilan id dari link (get)
3. Sintaks sql yang berisi Tentang menghapus data di tabel stok bedasarkan id (di dapat dari get)
4. Ekseukusi sintaks sql yang melibatkan koneksi
5. Pengecekan apakah ada eror, ketika nilai 1, data berhasil di hapus, 0 tidak berubah, -1 eror

**UPADATE / EDIT (FRONT END)**

<?php

require\_once '../koneksi.php';

$id = $\_GET["id"];

$sqlsintaks = "SELECT \* FROM stok WHERE id = '$id' ";

$eksekusi = mysqli\_query($koneksi, $sqlsintaks);

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Tambah Data</title>

    <script src="https://cdn.jsdelivr.net/npm/@tailwindcss/browser@4"></script>

</head>

<body class="flex items-center justify-center min-h-screen">

    <div class="bg-white shadow-lg rounded-2xl p-8 w-full max-w-md">

        <h2 class="text-2xl font-bold mb-6 text-gray-800 text-center">Update Data</h2>

        <form action="../backend/update.php" method="POST" class="space-y-5">

            <?php  while ( $datastok = mysqli\_fetch\_assoc($eksekusi)){?>

            <input type="hidden" name="id" value="<?php echo $datastok["id"]; ?>">

            <div>

                <label for="nama\_produk" class="block text-sm font-medium text-gray-700 mb-1">Nama Produk</label>

                <input type="text" id="nama\_produk" name="nama\_produk"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan nama produk" value="<?php echo $datastok["nama\_produk"]; ?>">

            </div>

            <div>

                <label for="warna" class="block text-sm font-medium text-gray-700 mb-1">Warna</label>

                <input type="text" id="warna" name="warna"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan warna" value="<?php echo $datastok["warna"]; ?>">

            </div>

            <div>

                <label for="kategori" class="block text-sm font-medium text-gray-700 mb-1">Kategori</label>

                <input type="text" id="kategori" name="kategori"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan kategori" value="<?php echo $datastok["kategori"]; ?>">

            </div>

            <div>

                <label for="harga" class="block text-sm font-medium text-gray-700 mb-1">Harga</label>

                <input type="text" id="harga" name="harga"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan harga" value="<?php echo $datastok["harga"]; ?>">

            </div>

            <div>

                <label for="rilis" class="block text-sm font-medium text-gray-700 mb-1">Tahun Rilis</label>

                <input type="rilis" id="rilis" name="rilis"

                    class="w-full px-4 py-2 border border-gray-300 rounded-lg shadow-sm focus:ring-2 focus:ring-blue-500 focus:outline-none"

                    placeholder="Masukkan tahun rilis" value="<?php echo $datastok["rilis"]; ?>">

            </div>

            <button type="submit"

                class="w-full bg-blue-600 text-white py-2 px-4 rounded-lg hover:bg-blue-700 focus:ring-2 focus:ring-blue-400 transition">

                Submit

            </button>

            <?php }?>

        </form>

    </div>

</body>

</html>

**Pada kode ini terdapat**

1. Bagian Pengambilan data:
2. Pengambil kode / require\_once di koneksi php. Ini agar dari sisi php akan tersambung ke sql.
3. Penyimpanan kode sintaks sql yang berisi “SELECT \* FORM stok” yang berarti memilih semua yang ada di tabel stok
4. Pengeksekusian sql sintaks bersama koneksi
5. Bagian Penampilan Menggunakan prinsip select
6. Pengambilan data menggunakan while. Dikarenakan data di mysql itu banyak, maka akan menggunakan while, yang di mana akan menjankan kode tersebut selama nilainya true (ada data). Yang dimana di dalam parameter while terdapat mysqli fetch assoc / berupa array asosiatif
7. Karena berupa asosiatif maka yang ditambpikan berupa string
8. Di tampilkan di value di html

**UPDATE / EDIT (BACK END)**

<?php

require\_once '../koneksi.php';

// Pengambilan nilai

$nama\_produk = $\_POST['nama\_produk'];

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

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

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

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

// Perintah ke SQL

$sintakssql = "UPDATE stok SET nama\_produk = '$nama\_produk', warna = '$warna', kategori = '$kategori', harga = '$harga', rilis = '$rilis' WHERE id = '$id'";

$eksekusi = mysqli\_query($koneksi, $sintakssql);

// Pengecekan ERORR

if (mysqli\_affected\_rows($koneksi) > 0) {

    echo "

    <script>

        alert('Data Berhasil diedit');

        window.location.href = '../frontend/index.php';

    </script>

    ";

} else {

    echo "Data gagal diedit". mysqli\_errno($koneksi);

}

?>

Dalam kode ini terdapat :

1. Pengambilan kode dari koneksi.php
2. Pengambilan nilai bedasarkan metod post dari sisi frontend.
3. Perintah sql yang berupa UPDATE namatabel SET nama\_kolom = ‘value yang di ubah’ WHERE id = ‘$id’, ini bedasarkan id per baris
4. Jika hasil nya 1 (berhasil), 0 (gagal), -1 eror

**SISTEM UPLOAD GAMBAR**

Penambahan pada Front End yakni tambahPage.php di bagian form sebagai berikut:

enctype="multipart/form-data"

Encoding type ini digunakan agar bisa membaca uploadan file dari form page pada halaman tersebut.

Penambahan input type file dengan name dan id gambar.

            <div>

                <input type="file" name="gambar" id="gambar">

            </div>

Penambahan sistem gambar di backend di **tambah.php**

// File Uploading sistem

$namaGambar = $\_FILES['gambar']['name'];

$tmpGambar = $\_FILES['gambar']['tmp\_name'];

$error = $\_FILES['gambar']['error'];

// Pengecekan files jika tidak ada file

if($error === 4){

    echo "

    <script>

        alert('Upload dulu gambarnya');

        window.location.href = '../frontend/tambahPage.php';

    </script>

    ";

    return false;

}

// Pengecekan ekstensi gambar

$ekstensiDibolehkan = ["png", "jpg", "jpeg"];

$ekstensiGambar = explode(".", $namaGambar);

$ekstensiGambar = strtolower(end($ekstensiGambar));

if (!in\_array($ekstensiGambar, $ekstensiDibolehkan)){

    echo "

    <script>

        alert('eror ekstensi gambar, note : ini bukan gambar!');

        window.location.href = '../frontend/tambahPage.php';

    </script>

    ";

    return false;

}

move\_uploaded\_file($tmpGambar, "../img/" . $namaGambar);

Pada kode tersebut terdapat:

1. Pengembian data $\_FILES dengan array assoiatif gambar yang memiliki array lagi di dalamnya seperti name,tmp\_name, erorr (array multidimensi), data-data tersebut disimpan ke suatu variabel.
2. Pengecekan error 4 , yakni value yang berfungsi untuk mengecek apakah user sudah mengungah file? Jika belum maka akan alert, dan kode berikutnya di hentikan alias return false atau pun bisa menggunakan die.
3. Beriisi variabel yang berisi array seperti png, jpg, jpeg yang berbentuk string.
4. Memecah nama gambar menjadi beberapa bagian dengan patokan “.” , Contoh : [“Manusia”, “Jpg”]
5. Merubah semua huruf menjadi huruf kecil dan mengambil data paling belakang dari array yang tadi contoh : jpg.
6. Mengecek apakah array yang tadi (manusia.jpg yg diambil jpg) itu ada di sekumpulan array $ekstensiDibolehkan ?, Dikarenakan ada tanda “!” yang berarti not berarti ngecek jika **tidak ada.** Jika ada maka kode insert akan di jalankan.
7. Memindahkan file sementara alias $tmpGambar ke folder ../img/ dengan nama gambar.

**SISTEM UBAH GAMBAR**

1. **Mencari data gambar sebelum di update**

$sqlGambarLama = mysqli\_query($koneksi, "SELECT gambar FROM stok WHERE id = '$id'");

$dataLama = mysqli\_fetch\_assoc($sqlGambarLama);

$gambarLama = $dataLama['gambar'];

1. Kita mencari gambar yang ada di tabel stok di database bedasarkan id dari $\_POST
2. Eksekusi perintah di bungkusui variabel data lama
3. dataLama skearang sudah asosiatif, ambil lah data bedasarkan kolom gambar. Ini akan dibungkus dengan variabel gambar lama
4. **Pengecekan apakah user memang ingin meng update data atau hanya teks?**

$error = $\_FILES['gambar']['error'];

if($error === 4){

    $namaFile = $gambarLama;

}

Mengecek apakah tidak ada gambar yang di uplado? Jika tidak maka pakai saja gambar lama sebagai acuan upload.

1. **Proses pengupdatean data**

else {

    $namaGambar = $\_FILES['gambar']['name'];

    $tmpGambar = $\_FILES['gambar']['tmp\_name'];

    $size = $\_FILES['gambar']['size'];

    $ekstensiDibolehkan = ["png", "jpg", "jpeg"];

    $ekstensiGambar = explode(".", $namaGambar);

    $ekstensiGambar = strtolower(end($ekstensiGambar));

    if (!in\_array($ekstensiGambar, $ekstensiDibolehkan)){

        echo "

        <script>

            alert('eror ekstensi gambar, note : ini bukan gambar!');

            window.location.href = '../frontend/index.php';

        </script>

        ";

        return false;

    }

    // Bikin nama unik biar tidak nabrak

    $namaGambarBaru = uniqid() . "." . $ekstensiGambar;

    // Upload gambar baru

    move\_uploaded\_file($tmpGambar, "../img/" . $namaGambarBaru);

    // Hapus gambar lama kalau ada

    if ($gambarLama && file\_exists("../img/" . $gambarLama)) {

        unlink("../img/" . $gambarLama);

    }

}

Pada kode di atas dijelaskan bahwa:

1. Pengambilan nama gambar bedasarkan array multidimensi dari $\_FILES gambar yang berisi index name, tmp\_name, size.
2. Membuat suatu tampungan yang berbentuk array yakni file gambar apa saja di bolehkan ?
3. Memecah nama gambar dengan patokan “.” Contoh = Manusia.jpg = “Manusia” “jpg”
4. Mengubah semua kata menjadi huruf kecil , Contoh = Manusia => manusia. Serta mengambil index paling belakang dari array yang sudah di pecah.
5. Mengecek apakah index paling belakang array yang tadi itu ada di sekumpulan array “esktensi yang diperbolehkan”. Metode “mencari jarum pada jerami”
6. Kerena menggunakan metode flase maka jika tidak ada maka tampilkan eror dan hentikan kode.
7. Jika kode yang tadi aman maka jalankan :
8. Mengeubah nama file dengan id unik serta di tempelkan “.” Dan juga ekstensi gambar yang tadi
9. Memindahkan file dari tmp ke ../img/ dengna nama file.
10. Jika ada gambar lama dan ada di file maka unlink alias hapus.

**SISTEM MENGHAPUS GAMBAR**

**1. Pengambilan data gambar dari mysql.**

Data ini di ambil dari mysql dari tabel stok dengan kolom gambar bedasarkan id yang ada. Ini dijalankan dengan mysqli\_query. Setelah diambil, data akan dijadikan menjadi array asosisatif. Ini akan menjadikan $dataGambar[‘gambar’]

// Ambil nama file gambar dulu

$queryGambar = mysqli\_query($koneksi, "SELECT gambar FROM stok WHERE id = '$id'");

$dataGambar = mysqli\_fetch\_assoc($queryGambar);

**2. Penghapusan**

    if ($dataGambar && file\_exists("../img/" . $dataGambar['gambar'])) {

        unlink("../img/" . $dataGambar['gambar']);

    }

Pada kode ini terdapat jika data gambar ada, dan jika file ../img/ yang disambungkan dengan dataGambar[‘gambar] **terdapat di mysqli** maka akan mengahpus data gambar yang ada di komputer server bedasarkan data gambar yang sama dengan yang ada di sql.