

**LAPORAN PRAKTIKUM  
PERANCANGAN DAN PEMROGRAMAN WEB**

**MODUL 10**

**PHP & MYSQL**



Oleh:

Dhiemas Tulus Ikhsan - 2311104046

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK  
DIREKTORAT KAMPUS PURWOKERTO  
UNIVERSITAS TELKOM  
2025**

## BAB I

### PENDAHULUAN

#### 1.1 Dasar Teori

Pembangunan situs web dinamis sangat bergantung pada sinergi antara bahasa pemrograman server-side dan sistem manajemen basis data untuk memastikan pengelolaan informasi yang fleksibel dan berkesinambungan. Berbeda dengan situs statis yang kaku karena pembaruan kontennya harus dilakukan langsung pada kode sumber HTML dan CSS, penggunaan PHP dan MySQL menawarkan solusi yang lebih adaptif dalam memproses data secara real-time. Dalam ekosistem ini, PHP berperan sebagai otak yang mengolah logika aplikasi dan menjembatani komunikasi antara input pengguna dengan basis data. Di sisi lain, MySQL berfungsi sebagai repositori relasional yang menyimpan data secara sistematis dalam struktur tabel melalui perintah SQL. Integrasi keduanya memungkinkan penyimpanan data secara permanen sekaligus penyajian informasi secara otomatis sesuai instruksi sistem.

Inti dari manajemen data dalam aplikasi berbasis database terletak pada implementasi konsep CRUD (Create, Read, Update, Delete). Operasi Create memfasilitasi penambahan entitas baru, Read digunakan untuk pengambilan dan penyajian data, Update untuk sinkronisasi perubahan informasi, serta Delete untuk meniadakan data yang sudah tidak relevan. Penerapan siklus CRUD ini menjamin sistem informasi tetap komprehensif, terorganisir, dan mudah dikelola dalam jangka panjang. Selain itu, mekanisme pertukaran data antara pengguna dan peladen didukung oleh metode HTTP GET untuk kueri pencarian dan POST untuk transmisi data formulir yang lebih aman. Optimasi fitur pencarian sering kali mengandalkan perintah SQL LIKE guna memfilter data berdasarkan kata kunci tertentu. Dengan sentuhan CSS atau framework desain sebagai penunjang visual, antarmuka aplikasi menjadi lebih intuitif dan estetis, yang secara keseluruhan memenuhi tujuan utama praktikum dalam membangun sistem pengelolaan data yang dinamis.

#### 1.2 Tujuan

1. Mahasiswa mampu memahami konsep database dan integrasinya dengan PHP
2. Mahasiswa mampu melakukan koneksi PHP dengan MySQL
3. Mahasiswa mampu membuat aplikasi CRUD (Create, Read, Update, Delete) menggunakan PHP dan MySQL
4. Mahasiswa mampu menerapkan validasi dan keamanan data pada aplikasi web

## **BAB II**

### **HASIL PRAKTIKUM**

#### **2.1 GUIDED (Praktikum Terbimbing)**

Pada tahap ini dilakukan persiapan database menggunakan phpMyAdmin. Database digunakan sebagai tempat penyimpanan data agar dapat dikelola secara terstruktur. phpMyAdmin memudahkan pengelolaan database MySQL melalui antarmuka web.

Database yang dibuat bernama akademik, kemudian dibuat tabel mahasiswa untuk menyimpan data mahasiswa seperti NIM, nama, jurusan, email, dan tanggal lahir. NIM digunakan sebagai primary key agar setiap data bersifat unik.

```
CREATE DATABASE akademik;

CREATE TABLE mahasiswa(
    nim VARCHAR(10) PRIMARY KEY,
    nama VARCHAR(50) NOT NULL,
    jurusan VARCHAR(30) NOT NULL,
    email VARCHAR(50),
    tanggal_lahir DATE
);
```

#### **Teori HTTP Method: GET dan POST**

HTTP Method digunakan untuk mengirim data dari form ke server. Pada praktikum ini digunakan metode GET dan POST.

Metode GET digunakan untuk pencarian data karena data dikirim melalui URL dan mudah diproses. Sedangkan metode POST digunakan untuk input data karena lebih aman dan data tidak terlihat di URL.

```
<form method="GET">

<input type="text" name="keyword" placeholder="Cari data">

<button type="submit">Cari</button>

</form>

<form method="POST" action="proses_tambah.php">
```

```
<input type="text" name="nama">  
  
<button type="submit">Simpan</button>  
  
</form>
```

## Koneksi PHP dengan MySQL

Koneksi database dibuat menggunakan ekstensi MySQLi. File koneksi dipisahkan agar dapat digunakan kembali oleh banyak file, sehingga kode lebih efisien dan terstruktur.

Koneksi diuji untuk memastikan aplikasi dapat terhubung dengan database MySQL

```
<?php  
  
$host = "localhost";  
  
$username = "root";  
  
$password = "";  
  
$database = "akademik";  
  
$conn = mysqli_connect($host, $username, $password, $database);  
  
if (!$conn) {  
  
die("Koneksi gagal: " . mysqli_connect_error());  
  
}  
  
mysqli_set_charset($conn, "utf8");  
  
?>  
  
<?php  
  
include "koneksi.php";  
  
if ($conn) {  
  
echo "Koneksi ke database berhasil!";  
  
} else {  
  
echo "Koneksi ke database gagal!";  
  
}  
  
?>
```

## Operasi CRUD dengan PHP dan MySQL

### CREATE (Menambah Data)

Digunakan untuk menambahkan data mahasiswa ke dalam database. Data diambil dari form kemudian diproses menggunakan query INSERT. Sebelum menyimpan data, dilakukan pengecekan agar NIM tidak duplikat.

```
<form method="POST" action="proses_tambah.php">

<input type="text" name="nim" placeholder="NIM" required>

<input type="text" name="nama" placeholder="Nama" required>

<input type="text" name="jurusan" placeholder="Jurusan" required>

<input type="email" name="email">

<input type="date" name="tanggal_lahir">

<button type="submit">Simpan</button>

</form>

<?php

include "koneksi.php";

$nim = $_POST['nim'];

$nama = $_POST['nama'];

$jurusan = $_POST['jurusan'];

$email = $_POST['email'];

$tanggal_lahir = $_POST['tanggal_lahir'];

$query = "INSERT INTO mahasiswa VALUES

('$nim','$nama','$jurusan','$email','$tanggal_lahir')";

mysqli_query($conn, $query);

header("Location: tampil_data.php");
```

### READ (Menampilkan Data)

Digunakan untuk menampilkan data mahasiswa dari database ke dalam tabel HTML menggunakan query SELECT.

```
tampil_data.php
```

```

<?php
include "koneksi.php";
$query = "SELECT * FROM mahasiswa";
$result = mysqli_query($conn, $query);
?>


| NIM                  | Nama                  | Jurusan                  | Email                  | Tanggal Lahir                  |
|----------------------|-----------------------|--------------------------|------------------------|--------------------------------|
| <?= \$row['nim']; ?> | <?= \$row['nama']; ?> | <?= \$row['jurusan']; ?> | <?= \$row['email']; ?> | <?= \$row['tanggal_lahir']; ?> |

.

```

## **UPDATE (Mengubah Data)**

Digunakan untuk mengubah data mahasiswa. Data lama ditampilkan kembali ke form edit, kemudian diperbarui menggunakan query UPDATE.

```

$query = "UPDATE mahasiswa SET
nama='$nama',

```

```
jurusan='$jurusan',  
email='$email',  
tanggal_lahir='$tanggal_lahir'  
WHERE nim='$nim';
```

### **DELETE (Menghapus Data)**

Digunakan untuk menghapus data mahasiswa berdasarkan NIM menggunakan query DELETE.

#### Source Code

```
<?php  
  
include "koneksi.php";  
  
$nim = $_GET['nim'];  
  
mysqli_query($conn, "DELETE FROM mahasiswa WHERE nim='$nim');  
  
header("Location: tampil_data.php");  
  
?>
```

## **2.2 UNGUIDED (Tugas Mandiri)**

1. Integrasikan PHP dengan MySQL pada Tugas Pertemuan kedua (Bootstrap).
  - Tambahkan menu di sidebar dengan nama Kelola Produk
  - Terapkan CRUD (Create Read Update Delete) pada halaman Kelola Produk
  - Pastikan seluruh produk diambil dari Database
  - Tambahkan fungsi pencarian produk

Jawaban:

### **koneksi.php**

```
<?php  
  
$conn = mysqli_connect("localhost", "root", "", "multifandom");  
  
if (! $conn) {  
  
die(
```

### **admin.php**

```
<!DOCTYPE html>
```

```
<html lang="id">

<head>

<meta charset="UTF-8">

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

<title>Halaman Administrator</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header>

<h1> Panel Administrator</h1>

</header>

<main class="admin-layout">

<nav class="menu-samping">

<h3>Menu</h3>

<ul>

<li><a href="#">Data User</a></li>

<li><a href="produk.php">Kelola Produk</a></li>

<li><a href="#">Edit Password</a></li>

<li><a href="#">Logout</a></li>

</ul>

</nav>

<section class="konten-admin">

<h2>Selamat Datang, Admin!</h2>

<p>Pilih menu di sebelah kiri untuk mengelola data website.</p>

<p>Anda dapat mengatur data user, mengelola produk, dan melakukan perubahan lain sesuai kebutuhan sistem.</p>

<p>Pastikan keamanan data selalu terjaga dan lakukan <strong>backup</strong> secara rutin.</p>

<div class="info-box">

<h3> Statistik Singkat</h3>

<ul>

<li>Total User Terdaftar: <span>145</span></li>

<li>Produk Aktif: <span>38</span></li>
```

```

<li>Transaksi Hari Ini: <span>12</span></li>
</ul>
</div>
</section>
</main>
<footer>
<p>© 2025 Multi Fandom Society</p>
<p>Unite. Create. Celebrate.</p>
<p>Follow the fandom journey → <a href="#">@multifandomsociety</a></p>
</footer>
</body>
</html>

```

## produk.php

```

<?php
include "koneksi.php";
$keyword = "";
$query = "SELECT * FROM produk";
if (isset($_GET['keyword'])) {
$keyword = $_GET['keyword'];
$query = "SELECT * FROM produk
WHERE nama_produk LIKE '%$keyword%'
OR kategori LIKE '%$keyword%'";
}
$result = mysqli_query($conn, $query);
?>
<!DOCTYPE html>
<html lang="id">
<head>
<meta charset="UTF-8">
<title>Kelola Produk</title>
<link rel="stylesheet" href="style.css">
</head>

```

```
<body>

<header>

<h1> Panel Administrator</h1>

</header>

<main class="admin-layout">

<nav class="menu-samping">

<h3>Menu</h3>

<ul>

<li><a href="admin.php">Dashboard</a></li>

<li><a href="produk.php">Kelola Produk</a></li>

</ul>

</nav>

<section class="konten-admin">

<h2>Kelola Produk</h2>

<form method="GET">

<input type="text" name="keyword" placeholder="Cari produk..." value="<?= $keyword ?>">

<button type="submit">Cari</button>

</form>

<br>

<a href="form_tambah.php">+ Tambah Produk</a>

<table border="1" cellpadding="8" cellspacing="0">

<tr>

<th>No</th>

<th>Nama</th>

<th>Kategori</th>

<th>Harga</th>

<th>Stok</th>

<th>Aksi</th>

</tr>

<?php $no=1; while($row = mysqli_fetch_assoc($result)) { ?>

<tr>

<td><?= $no++ ?></td>
```

```

<td><?= $row['nama_produk'] ?></td>
<td><?= $row['kategori'] ?></td>
<td><?= $row['harga'] ?></td>
<td><?= $row['stok'] ?></td>
<td>
<a href="form_edit.php?id=<?= $row['id'] ?>">Edit</a> |
<a href="proses_hapus.php?id=<?= $row['id'] ?>" onclick="return confirm('Hapus?')">Hapus</a>
</td>
</tr>
<?php } ?>
</table>
</section>
</main>
</body>
</html>

```

### **Proses\_tambah.php**

```

<?php
include "koneksi.php";
mysqli_query($conn, "INSERT INTO produk VALUES (
NULL,
'$_POST[nama_produk]',
'$_POST[kategori]',
'$_POST[harga]',
'$_POST[stok]'
) ");
header("Location: produk.php");
?>

```

### **Proses\_edit.php**

```

<?php
include "koneksi.php";
mysqli_query($conn, "UPDATE produk SET

```

```

nama_produk='$_POST[nama_produk]',
kategori='$_POST[kategori]',
harga='$_POST[harga]',
stok='$_POST[stok]'

WHERE id='$_POST[id]'");

header("Location: produk.php");

?>

```

## **Form\_tambah.php**

```

<form method="POST" action="proses_tambah.php">

<input type="text" name="nama_produk" placeholder="Nama Produk" required><br><br>

<input type="text" name="kategori" placeholder="Kategori" required><br><br>

<input type="number" name="harga" placeholder="Harga" required><br><br>

<input type="number" name="stok" placeholder="Stok" required><br><br>

<button type="submit">Simpan</button>

</form>

```

## **Form\_edit.php**

```

<?php

include "koneksi.php";

$id = $_GET['id'];

$data = mysqli_fetch_assoc(mysqli_query($conn,
"SELECT * FROM produk WHERE id=$id"));

?>

<form method="POST" action="proses_edit.php">

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

<input type="text" name="nama_produk" value="<?= $data['nama_produk'] ?>"><br><br>

<input type="text" name="kategori" value="<?= $data['kategori'] ?>"><br><br>

<input type="number" name="harga" value="<?= $data['harga'] ?>"><br><br>

<input type="number" name="stok" value="<?= $data['stok'] ?>"><br><br>

```

```

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

```

## Delete.php

```

<?php
include "koneksi.php";
mysqli_query($conn, "DELETE FROM produk WHERE id=$_GET[id]");
header("Location: index.php");

```

## Output:

No	Kategori	Harga	Stok	Tugas Minngu In	Aksi
1	Smartphone X Pro	Gadget	9	<a href="#">Edit   Hapus</a>	
2	Smartphone X Pro	Rp 8.900.000	55	50	
2	Smartwatch Series 5	Rp 3.200.000	120	<a href="#">Edit   Hapus</a>	
3	Keysourd Mekanik	Rp Komputer	30	<a href="#">Edit   Hapus</a>	

Website yang awalnya bersifat statis telah dikembangkan menjadi sistem dinamis dengan mengintegrasikan PHP dan MySQL, khususnya pada modul Manajemen Data Produk. Pengembangan ini berpusat pada penggunaan file koneksi.php sebagai jembatan komunikasi antara aplikasi dan basis data. Melalui dashboard admin.php, administrator dapat mengoperasikan sistem pengelolaan data secara terstruktur menggunakan konsep CRUD (Create, Read, Update, Delete) yang lengkap.

## **BAB III**

### **PENUTUP**

#### **KESIMPULAN**

Berdasarkan rangkaian praktikum dan penyelesaian tugas mandiri, dapat disimpulkan bahwa integrasi antara PHP dan MySQL merupakan pondasi utama dalam mentransformasi halaman web statis menjadi sistem informasi dinamis yang mampu mengorganisir data secara sistematis. Praktikum ini berhasil memberikan pemahaman komprehensif mengenai siklus operasional CRUD (Create, Read, Update, Delete), efektivitas metode transmisi data melalui GET dan POST, serta urgensi stabilitas koneksi basis data dalam pengelolaan inventaris produk.

Meskipun terdapat kendala teknis dalam proses pengembangan—seperti kegagalan otentikasi database, ketidaksesuaian struktur direktori, hingga kesalahan sintaksis pada kueri SQL, permasalahan tersebut dapat diatasi melalui audit konfigurasi pada file koneksi.php, penyelarasan file path, serta validasi kueri secara berkala menggunakan perangkat phpMyAdmin. Secara keseluruhan, praktikum ini telah mencapai target pembelajaran dalam membangun aplikasi web berbasis database yang fungsional, terstruktur, dan adaptif terhadap kebutuhan data.