RESUME MATERI UCP PEMROGRAMAN WEB

# 1. HTTP (HyperText Transfer Protocol)

• HTTP adalah protokol untuk mentransfer data antara client (browser) dan server.

• Metode: GET (ambil data), POST (kirim data), PUT (update), DELETE (hapus).

• Status Code: 200 (OK), 404 (Not Found), 500 (Internal Server Error).

# 2. HTML (HyperText Markup Language)

Dasar HTML:

<!DOCTYPE html>  
<html>  
 <head><title>Judul</title></head>  
 <body>Konten di sini</body>  
</html>

Elemen penting: heading (<h1>-<h6>), paragraf (<p>), link (<a>), gambar (<img>), list (<ul>/<ol>/<li>).

Kode tersebut menunjukkan struktur dasar sebuah halaman web. Tag <!DOCTYPE html> mendefinisikan dokumen sebagai HTML5, dengan <html> sebagai elemen pembungkus utama. Di dalam <html>, terdapat <head> yang berisi informasi non-visual seperti <title> untuk judul di tab browser, dan <body> yang berisi semua konten yang akan ditampilkan kepada pengguna. Untuk menyusun konten di dalam <body>, kita menggunakan elemen-elemen penting seperti **headings** (<h1> hingga <h6>) untuk membuat judul dengan berbagai tingkatan, **paragraf** (<p>) untuk blok teks, **link** (<a>) untuk membuat tautan ke halaman lain, **gambar** (<img>) untuk menampilkan visual, serta **list** yang bisa tidak berurutan (<ul>) atau berurutan (<ol>), di mana setiap itemnya menggunakan tag <li>.

Form HTML:

<form action="proses.php" method="POST">  
 <input type="text" name="nama" />  
 <input type="submit" value="Kirim" />  
</form>

# 3. CSS (Cascading Style Sheets)

• Digunakan untuk mengatur tampilan HTML.

• Penulisan bisa internal (<style>), eksternal (file .css).

• Layout: margin, padding, border, display, position.

• Flexbox: display: flex, justify-content, align-items.

# 4. JavaScript

Dasar: alert, variabel (let, const), fungsi.

OOP: objek dan class. Contoh:

class Mahasiswa {  
 constructor(nama) {  
 this.nama = nama;  
 }  
 sapa() {  
 return `Halo ${this.nama}`;  
 }  
}

Standard Library: Math, Date, Array (push, map, filter, forEach).

# 5. Bootstrap 5.3

• Framework CSS responsif.

• Tambahkan link CDN ke <head>.

• Gunakan class seperti container, row, col-md, btn, card, navbar, modal.

# 6. PHP + MySQL + Bootstrap (Login & Register)

• Alur: register → simpan DB, login → cek DB, buat session.

• File umum: register.php, login.php, dashboard.php, logout.php, config.php.

Contoh kode register:

<?php  
include "config.php";  
if (isset($\_POST['register'])) {  
 $nama = $\_POST['nama'];  
 $email = $\_POST['email'];  
 $password = password\_hash($\_POST['password'], PASSWORD\_DEFAULT);  
 mysqli\_query($conn, "INSERT INTO users (nama, email, password) VALUES ('$nama', '$email', '$password')");  
} ?>

Contoh login:

<?php  
session\_start();  
include "config.php";  
if (isset($\_POST['login'])) {  
 $email = $\_POST['email'];  
 $password = $\_POST['password'];  
 $result = mysqli\_query($conn, "SELECT \* FROM users WHERE email='$email'");  
 $user = mysqli\_fetch\_assoc($result);  
 if (password\_verify($password, $user['password'])) {  
 $\_SESSION['user'] = $user;  
 header("Location: dashboard.php");  
 } else {  
 echo "Login gagal!";  
 }  
} ?>

# Penutup

• HTTP: Protokol komunikasi data.

• HTML: Struktur halaman.

• CSS: Gaya dan layout.

• JavaScript: Interaktif dan dinamis.

• Bootstrap: Framework desain siap pakai.

• PHP + MySQL: Backend dinamis dan autentikasi.