

**Muhammad Haikal Azka-E-**  
**20230140242**

## **RANGKUMAN MATERI PENGEMBANGAN DESAIN WEB**

---

### **1. HTTP**

HTTP (Hypertext Transfer Protocol) adalah protokol komunikasi utama di web antara client dan server.

#### **Karakteristik**

- Stateless: setiap request berdiri sendiri.
- Bersifat teks (mudah dibaca manusia), tapi HTTP/2 dan HTTP/3 mendukung format biner.
- Digunakan oleh semua website modern.

#### **HTTP Method dan Fungsi**

- GET: meminta data dari server (digunakan saat membuka halaman).
- POST: mengirim data ke server (digunakan saat login, kirim form).
- PUT: mengganti data di server.
- DELETE: menghapus data di server.

#### **Header dan Body**

- HTTP Header: berisi metadata permintaan (misal: jenis konten, panjang data, user-agent).
- HTTP Body: berisi isi permintaan (khusus POST/PUT).

#### **Perbedaan HTTP dan HTTPS**

- HTTP = tidak aman, semua data dikirim terbuka.
- HTTPS = menggunakan SSL/TLS untuk enkripsi, lebih aman untuk login, transaksi.

#### **Contoh Kode**

GET /index.html HTTP/1.1

Host: www.contoh.com

User-Agent: Mozilla/5.0

POST /login HTTP/1.1

Content-Type: application/x-www-form-urlencoded

username=kai&password=1234

### **Kesimpulan**

HTTP menjadi tulang punggung semua komunikasi data web. HTTPS digunakan untuk keamanan.

### **2. HTML (Struktur & Form)**

HTML adalah bahasa markup yang menjadi pondasi struktur halaman web.

#### **Struktur Umum HTML**

- <html> adalah elemen pembungkus seluruh dokumen.
- <head> berisi metadata, <body> berisi konten yang dilihat user.
- Tag memiliki pembuka dan penutup. Contoh: <p>...</p>

#### **Tag Penting**

- <h1> - <h6>: Judul (struktur utama konten, penting untuk SEO dan tampilan).
- <p>: Paragraf teks, otomatis beri jarak antar paragraf.
- <a>: Tautan ke halaman lain/situs eksternal.
- <img>: Menampilkan gambar, wajib pakai src dan alt.
- <ul>, <ol>, <li>: Membuat daftar (unordered/ordered).
- <br>: Ganti baris, tidak memulai paragraf baru.
- <strong>/<b>: Menebalkan teks (semantik vs visual).
- <em>/<i>: Memiringkan teks (semantik vs visual).

#### **Form & Input**

- Digunakan untuk menerima data dari user.
- Elemen penting: <form>, <input>, <textarea>, <select>, <label>, <button>.
- Atribut penting: name, type, required, placeholder, value.

#### **Contoh Kode**

```
<form action='/login' method='post'>
```

```
  <label for='user'>Username:</label>
```

```
  <input type='text' id='user' name='username' required>
```

```
  <input type='submit' value='Login'>
```

```
</form>
```

## Kesimpulan

HTML menyusun konten web dalam format terstruktur yang dapat dibaca dan ditampilkan oleh browser.

HTTP digunakan untuk mengirim data dari browser ke internet, namun tidak aman karena data bisa disadap; cocok untuk situs informasi umum seperti blog atau berita. Sebaliknya, HTTPS adalah versi aman karena data dienkripsi dengan SSL, sehingga cocok untuk situs yang mengelola data penting seperti login, formulir, dan transaksi. Maka, HTTP cocok untuk situs biasa tanpa data penting, sedangkan HTTPS wajib digunakan untuk menjaga keamanan data dan meningkatkan kepercayaan pengguna.

## 3. CSS (Style & Layout)

CSS digunakan untuk memberi gaya visual pada elemen HTML seperti warna, posisi, dan bentuk.

### Cara Menulis CSS

- • Inline: langsung di tag HTML (style=color:red;)
- • Internal: dalam tag <style>
- • Eksternal: file .css terpisah (direkomendasikan)

### Selector dan Properti

- • Selector: menentukan elemen mana yang diatur (p, .class, #id)
- • Property: color, font-size, margin, padding, border, background

### Layout dengan Flexbox

- • display: flex membuat container fleksibel
- • justify-content untuk mengatur posisi horizontal
- • align-items untuk vertikal, flex-direction untuk arah baris/kolom

### Contoh Kode

```
p { color: blue; font-size: 16px; }
```

```
#judul { font-weight: bold; }
```

```
.kotak { background: lightgray; padding: 10px; }
```

```
div.container { display: flex; justify-content: space-between; }
```

## Kesimpulan

CSS membuat web tampil menarik dan responsif, serta mengatur tata letak konten dengan efisien.

## 4. JavaScript (Dasar, OOP, dan Library)

JavaScript membuat halaman web menjadi hidup dengan kemampuan interaktif dan pemrosesan dinamis di sisi browser.

### Struktur Dasar JavaScript

- • Digunakan untuk logika dan interaksi dalam web.
- • Dapat disisipkan langsung di HTML (<script>) atau file eksternal .js.
- • Bahasa case-sensitive: 'Nama' ≠ 'nama'.

### Tipe Data Umum

- • String: Teks, contoh "Halo".
- • Number: Angka, contoh 10, 3.14.
- • Boolean: true atau false.
- • Array: Kumpulan nilai, contoh [1, 2, 3].
- • Object: Pasangan key dan value, contoh {nama: 'Kai', umur: 20}.

### Variabel dan Fungsi

- • let dan const untuk deklarasi variabel.
- • function namaFungsi() {} untuk membuat fungsi.
- • Arrow function => digunakan sebagai versi singkat.

### Contoh Kode

```
let nama = "Kai";
```

```
function sapa() {  
    alert("Halo " + nama);  
}
```

```
sapa();
```

### OOP (Object-Oriented Programming)

- • Class adalah template object.
- • Constructor digunakan untuk memberi nilai awal.
- • Inheritance (pewarisan) menggunakan 'extends'.

### Contoh Kode OOP

```
class Mahasiswa {  
    constructor(nama) { this.nama = nama; }  
  
    sapa() { console.log(`Halo, saya ${this.nama}`); }  
}  
  
const kai = new Mahasiswa("Kai");
```

```
kai.sapa(); // Output: Halo, saya Kai
```

### Library & Built-in Function

- Array: `.map()`, `.filter()`, `.reduce()`, `.push()`, `.pop()`
- String: `.toUpperCase()`, `.slice()`, `.split()`
- JSON: `JSON.stringify()`, `JSON.parse()`

### Kesimpulan

JavaScript sangat penting untuk membuat web menjadi interaktif dan dinamis, baik secara dasar maupun lanjutan seperti OOP.

## 5. Bootstrap

Bootstrap adalah framework CSS modern yang menyederhanakan pembuatan tampilan responsif dan menarik.

### Grid System

- Menggunakan container, row, dan col untuk menyusun layout.
- 12 kolom per baris, bisa disesuaikan ukurannya. Contoh: `col-6 = 50%`.
- Layout otomatis menyesuaikan ukuran layar HP, tablet, dan PC.

### Komponen Penting

- `btn` → Tombol
- `card` → Kotak informasi
- `navbar` → Navigasi atas
- `modal` → Popup kotak dialog
- `form-control` → Tampilan form yang rapi

### Utility Class

- Text: `text-center`, `text-white`, `text-muted`
- Background: `bg-primary`, `bg-light`, `bg-dark`
- Spacing: `m-3`, `p-2`, `mt-1` (margin dan padding)

### Contoh Kode

```
<div class="container">

  <div class="row">

    <div class="col-md-6">Kiri</div>

    <div class="col-md-6">Kanan</div>

  </div>

</div>
```

### Kesimpulan

Bootstrap mempercepat proses membuat UI web yang rapi, menarik, dan otomatis responsif di semua ukuran layar.

## 6. PHP + MySQL

PHP digunakan di sisi server untuk memproses permintaan, sedangkan MySQL menyimpan data pengguna dalam database.

### Fungsi PHP

- Memproses data dari form HTML.
- Mengakses dan memodifikasi database.
- Mengatur sesi pengguna (login/logout).

### CRUD (Create, Read, Update, Delete)

- Create → Tambah data: `INSERT`.
- Read → Ambil data: `SELECT`.
- Update → Ubah data: `UPDATE`.
- Delete → Hapus data: `DELETE`.

### Contoh Kode Koneksi dan Query

```
<?php
```

```
$conn = mysqli_connect("localhost", "root", "",
"db_kampus");
```

```
$sql = "SELECT * FROM users WHERE
username='kai'";
```

```
$result = mysqli_query($conn, $sql);
```

```
?>
```

### Proses Login Singkat

- User isi form login.
- PHP terima data `$_POST['username']`.
- Query ke database untuk cocokkan data.
- Jika cocok → simpan data user ke `$_SESSION`.

### Kesimpulan

PHP dan MySQL bekerja sama untuk membuat web yang bisa menyimpan dan memproses data pengguna seperti login, pendaftaran, dan dashboard.