

RESUME MATERI UCP

1. HTTP (Hypertext Transfer Protocol)

HTTP (Hypertext Transfer Protocol) adalah protokol yang digunakan untuk mentransfer data antara klien dan server melalui web. HTTP merupakan protokol **stateless**, yang berarti setiap permintaan (request) diperlakukan secara terpisah tanpa mempengaruhi permintaan sebelumnya. Protokol ini dirancang untuk komunikasi antara browser web (klien) dan server web. HTTP memfasilitasi pengiriman berbagai jenis data, termasuk HTML, JavaScript, CSS, gambar, dan media lainnya.

Komponen HTTP:

- **Request:** Klien mengirimkan HTTP request untuk meminta atau mengirim informasi ke server. Request ini dapat menggunakan metode seperti GET, POST, PUT, dan DELETE.
- **Response:** Setelah menerima request, server mengirimkan HTTP response yang berisi data atau informasi yang diminta oleh klien.
- **Stateless:** Setiap permintaan HTTP tidak bergantung pada permintaan sebelumnya, yang memungkinkan permintaan yang lebih fleksibel tanpa urutan tertentu.
- **HTTPS:** Versi aman dari HTTP, di mana data dienkripsi menggunakan protokol SSL/TLS untuk meningkatkan keamanan transaksi online.

HTTP versi terbaru adalah **HTTP/2** dan **HTTP/3**, yang mengoptimalkan pengiriman data dengan kompresi dan multiplexing untuk meningkatkan kecepatan dan efisiensi komunikasi antara klien dan server.

2. HTML (HyperText Markup Language)

HTML adalah bahasa markup yang digunakan untuk menyusun struktur halaman web. HTML menyediakan cara untuk menampilkan berbagai elemen web seperti teks, gambar, video, dan elemen formulir. HTML terdiri dari **tags** yang membungkus konten dan menentukan bagaimana elemen tersebut akan ditampilkan di browser.

HTML Dasar:

- **Tags:** HTML terdiri dari tag pembuka dan penutup. Contoh: `<p>Konten Teks</p>`.
- **Elemen:** Tag HTML yang berisi konten dan atribut tertentu. Contoh: ``.
- **Atribut:** Mengatur properti tambahan pada elemen, seperti `src` untuk gambar, `href` untuk tautan, dll.

HTML Form:

HTML Forms memungkinkan pengguna untuk memasukkan data yang dapat diproses oleh server. Elemen form mencakup berbagai jenis input, seperti teks, password, dan pilihan.

- **Tag `<form>`:** Digunakan untuk membuat form yang dapat menerima input.
- **Input Types:** Jenis input yang dapat digunakan dalam form, seperti `text`, `password`, `checkbox`, dan `radio`.
- **Metode Pengiriman:** Pengguna dapat memilih metode pengiriman data, seperti GET atau POST, sesuai dengan jenis form yang diinginkan.

3. CSS (Cascading Style Sheets)

CSS digunakan untuk mengatur **gaya** atau **layout** elemen-elemen HTML. CSS mengubah tampilan visual konten web dengan cara yang lebih efisien daripada mengubah HTML langsung. CSS memisahkan konten dari tampilan untuk menjaga struktur halaman web yang lebih rapi dan mudah dikelola.

CSS Dasar:

- **Selektor:** Memilih elemen HTML yang akan diberi gaya, seperti tag, id, dan class.
- **Properti dan Nilai:** Aturan CSS ditulis dalam format property: value;. Misalnya, color: red; akan memberi warna teks merah.
- **Posisi dan Layout:** CSS memungkinkan pengaturan posisi elemen menggunakan properti seperti display, position, dan float.

CSS Layout:

- **Normal Flow:** Ini adalah cara browser menampilkan elemen berdasarkan urutan HTML secara default. Namun, untuk tata letak yang lebih kompleks, kita menggunakan teknik layout lainnya.
- **Flexbox:** Layout berbasis flex memungkinkan kita untuk mendistribusikan ruang secara fleksibel dan merapikan elemen di dalam kontainer. Flexbox sangat cocok untuk tata letak skala kecil dan elemen responsif.
- **Grid Layout:** Digunakan untuk desain tata letak yang lebih kompleks dengan membagi halaman menjadi grid dan mengatur elemen-elemen dalam grid tersebut.

4. JavaScript

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web lebih interaktif. Dengan JavaScript, kita dapat mengubah konten secara dinamis, mengelola data, dan membuat aplikasi web yang lebih kaya.

JavaScript Dasar:

- **Tipe Data:** JavaScript mendukung tipe data seperti Number, String, dan Boolean.
- **Variabel:** Data disimpan dalam variabel yang dapat diubah selama eksekusi program.
- **Fungsi:** Fungsi digunakan untuk mengorganisir kode menjadi blok yang lebih kecil dan dapat dipanggil kapan saja.

Object-Oriented Programming (OOP):

JavaScript juga mendukung konsep **OOP**, yang memungkinkan penggunaan objek dan kelas untuk struktur data dan fungsionalitas yang lebih modular. **Kelas** berfungsi sebagai cetakan untuk objek, dan **Inheritance** memungkinkan objek mewarisi properti dan metode dari objek lain.

Standard Library:

JavaScript menyediakan berbagai fungsi standar untuk manipulasi data seperti metode pada **Array**, **String**, **Number**, dan objek lainnya. Misalnya, fungsi map(), filter(), dan reduce() digunakan untuk memanipulasi array secara efisien.

5. Bootstrap

Bootstrap adalah **framework CSS** yang menyediakan **template desain** siap pakai untuk aplikasi web responsif. Dengan menggunakan Bootstrap, pengembang dapat dengan cepat membuat layout yang beradaptasi dengan berbagai ukuran layar tanpa menulis banyak CSS dari awal.

Pengenalan Bootstrap:

Bootstrap menggunakan **grid system** untuk mengatur elemen-elemen di halaman. Framework ini juga menyediakan berbagai komponen UI seperti tombol, formulir, dan navigasi. Bootstrap memanfaatkan teknologi **CSS Flexbox** dan **Grid** untuk mempermudah desain responsif.

Menggunakan Bootstrap:

- **Instalasi:** Bootstrap dapat diinstal dengan menambahkan link CSS dan JavaScript ke dalam halaman HTML, atau dengan mengunduh file sumbernya.
- **Komponen:** Bootstrap menyediakan berbagai komponen seperti carousel, modals, dan alerts yang siap digunakan tanpa memerlukan penulisan banyak kode.
- **Customisasi:** Pengguna dapat menyesuaikan tema Bootstrap menggunakan file Sass untuk mengubah warna, font, dan tata letak sesuai kebutuhan.

6. Materi MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang digunakan untuk menyimpan dan mengelola data dalam aplikasi web. MySQL menggunakan bahasa query **SQL (Structured Query Language)** untuk berinteraksi dengan data. Sebagai database open-source, MySQL banyak digunakan bersama dengan PHP untuk mengelola data dalam aplikasi web dinamis.

Fungsi dan Kegunaan MySQL:

1. **Menyimpan Data:** MySQL digunakan untuk menyimpan data aplikasi web dalam tabel-tabel yang saling berhubungan.
2. **Mengelola Data:** MySQL memungkinkan pengguna untuk melakukan berbagai operasi seperti **insert**, **select**, **update**, dan **delete** data melalui query SQL.
3. **Keamanan dan Pengelolaan Pengguna:** MySQL menyediakan fitur untuk mengatur hak akses dan kontrol keamanan untuk melindungi data.

Struktur Data MySQL:

- **Database:** Basis data adalah wadah untuk menyimpan tabel-tabel yang berisi data.
- **Tabel:** Tabel terdiri dari baris (record) dan kolom (fields) yang digunakan untuk menyimpan data. Setiap tabel harus memiliki **primary key** untuk mengidentifikasi data secara unik.
- **Query SQL:** SQL digunakan untuk mengakses dan memodifikasi data dalam MySQL.
SELECT: Mengambil data dari database. **INSERT:** Menambah data ke dalam tabel.
UPDATE: Memperbarui data yang ada. **DELETE:** Menghapus data.

7. Materi PHP

PHP (Hypertext Preprocessor) adalah bahasa pemrograman yang digunakan untuk pengembangan aplikasi web dinamis. PHP adalah bahasa pemrograman server-side, yang artinya kode PHP dijalankan di server sebelum hasilnya dikirimkan ke browser pengguna. PHP banyak digunakan untuk membuat aplikasi berbasis web yang dapat berinteraksi dengan basis data dan menghasilkan konten dinamis secara real-time.

Fungsi dan Kegunaan PHP:

1. **Server-Side Scripting:** PHP memungkinkan pembuatan aplikasi web dinamis dengan menangani logika di server. Kode PHP mengatur interaksi antara server dan pengguna.
2. **Pengelolaan Formulir:** PHP sangat efisien dalam menangani data dari formulir HTML. PHP dapat memproses data yang dimasukkan pengguna, menyimpannya ke dalam database, atau mengirimkan kembali hasilnya ke halaman web.
3. **Pembuatan Aplikasi CRUD:** PHP sering digunakan untuk membangun aplikasi CRUD (Create, Read, Update, Delete) yang mengelola data dalam basis data.
4. **Keamanan:** PHP menyediakan berbagai cara untuk menangani data sensitif dan mencegah serangan umum seperti SQL injection dan Cross-Site Scripting (XSS).